

subversion 使用 mysql 认证的原理:

- 1.由 httpd 判断是否需要 location 进行认证(require 指令)
- 2.然后通过配置文件中指定的认证模块完成这个认证过程, 在这里, 认证的模块是 auth\_mysql。
- 3.auth\_mysql 模块通过访问指定的表及字段, 检查用户提交的用户名及密码是否匹配用户表.
- 4.如果有需要, auth\_mysql 模块还会验证组(group)
- 5.接下来通过 dav\_svn 模块检查 AuthzSVNAccessFile 文件进行存取控制。

#### 一.安装 mysql

```
# groupadd mysql
```

```
# useradd -g mysql -d /usr/local/mysql/data -M mysql
```

```
# tar -zxvf mysql-5.0.17.tar.gz
```

```
# cd mysql-5.0.17
```

```
./configure --prefix=/usr/local/mysql \           指定安装目录
> --sysconfdir=/etc \                             配置文件的路径
> --localstatedir=/usr/local/mysql/data \          数据库存放的路径
> --enable-assembly \                             使用一些字符函数的汇编版本
> --with-mysqld-ldflags=-all-static \              以纯静态方式编译服务端
> --with-charset=utf8 \                           添加 utf8 字符支持
> --with-extra-charsets=all                       添加所有字符支持
```

```
./configure --prefix=/usr/local/mysql
```

```
--sysconfdir=/etc
```

```
--localstatedir=/usr/local/mysql/data
```

```
--enable-assembly
```

```
--with-mysqld-ldflags=-all-static
```

```
--with-charset=utf8
```

```
--with-extra-charsets=all
```

```
# make
```

```
# make install
```

```
# cp /usr/local/mysql/share/mysql/mysql.server /etc/rc.d/init.d/mysqld //复制 mysql 服务启动项文件。
```

```
# chkconfig --add mysqld //把 mysql 放到启动项列表里
```

```
# cp /usr/local/mysql/share/mysql/my-medium.cnf /etc/my.cnf
```

```
# /usr/local/mysql/bin/mysql_install_db //在/usr/local/mysql/data 里建立好 MYSQL 数据库
```

```
# chown -R mysql:mysql /usr/local/mysql/data
```

```
service mysqld restart
```

创建 root 用户密码:

```
/usr/local/mysql/bin/mysqladmin -u root password '111111'
```

修改 root 用户密码:

```
/usr/local/mysql/bin/mysqladmin -u root -p password 123456
```

提示输入原来 root 密码（111111）输入原来密码后 root 密码变成 123456

进入 MYSQL 数据库

```
/usr/local/mysql/bin/mysql -u root -p
```

也可以-p 后直接接密码

```
/usr/local/mysql/bin/mysql -u root -p123456 注意: 123456 为 root 密码
```

## 二.安装 apache

```
# tar -zxvf httpd-2.2.14.tar.gz
```

```
# cd httpd.2.2.14
```

配置 apache 源代码树

```
./configure --prefix=/usr/local/httpd --enable-mods-shared=most --enable-so --enable-dav --enable-dav-fs
```

```
#make && make install
```

### 三.安装 svn

解压 subversion-1.5.7.tar.gz

```
#tar -zxvf subversion-1.5.7.tar.gz
```

```
#cd subversion-1.5.7
```

检查所需的 build tools 是否符合编译 subversion 的要求:

```
#sh autogen.sh 查看结果。
```

检查所依赖的第三方库是否安装:

```
libapr libapr-util zlib
```

配置源代码树

```
./configure --with-apr=/usr/local/httpd --with-apr-util=/usr/local/httpd
```

```
--with-apxs=/usr/local/httpd/bin/apxs
```

```
--with-ssl
```

```
--with-zlib=/usr/local/zlib
```

编译与安装

```
make && make install
```

编译过程中的错误 1.

```
/usr/bin/ld: /usr/local/zlib/lib/libz.a(deflate.o): relocation R_X86_64_32S against `a local symbol' can not be used when making a shared object; recompile with -fPIC
```

```
/usr/local/zlib/lib/libz.a: could not read symbols: Bad value
```

```
collect2: ld returned 1 exit status
```

```
make: *** [subversion/libsvn_subr/libsvn_subr-1.la] Error 1
```

解决办法 : 重新安装 zlib-1.2.3.tar.gz

```
tar -zxvf zlib-1.2.3.tar.gz
```

```
cd zlib-1.2.3
```

```
./configure
```

```
vi Makefile
```

找到 CFLAGS=-O3 -DUSE\_MMAP

在后面加入-fPIC，即变成 CFLAGS=-O3 -DUSE\_MMAP -fPIC

接下面步骤

```
make
```

```
make install
```

编译过程中的错误 2.:

```
ttpd/lib/libapr-1.la -lrt -lcrypt -lpthread -ldl
```

```
/usr/bin/ld: cannot find -lexpat
```

```
collect2: ld returned 1 exit status
```

```
make: *** [subversion/svn/svn] Error 1
```

是缺少-lexpat 库文件，找安装盘中的

```
expat-x.x.x.rpm
```

```
expat-devel-x.x.x.rpm
```

```
rpm -ivh 安装上就可以了
```

五. 安装 apache 的 mod\_auth\_mysql 模块实现 mysql 认证

在 <http://modauthmysql.sourceforge.net/> 下载 mod\_auth\_mysql

下载后，然后进入该目录解压。根据 BUILD 文件的指示，安装步骤如下：

安装前需要打个补丁：

[http://sourceforge.net/tracker/?func=detail&aid=1437139&group\\_id=60218&atid=493464](http://sourceforge.net/tracker/?func=detail&aid=1437139&group_id=60218&atid=493464) 下载  
apache22.diff，它是 mod\_auth\_mysql 专门针对 apache2.2 的一个补丁。

将 apache22.diff 复制到 mod\_auth\_mysql.c 同目录下，执行命令 `patch -p0 < apache22.diff`。

安装 mod\_auth\_mysql

```
/usr/local/httpd/bin/apxs -c -lmysqlclient -L/usr/local/mysql/lib/mysql
```

```
-I/usr/local/mysql/include/mysql -lm -lz mod_auth_mysql.c
```

安装 mod\_auth\_mysql 到 /usr/local/httpd/modules 中

```
/usr/local/httpd/bin/apxs -i mod_auth_mysql.la
```

然后把下面这行加入 httpd.conf

```
LoadModule mysql_auth_module modules/mod_auth_mysql.so
```

## 六. 配置 apache

修改 apache 主配置文件

```
vi /usr/local/httpd/conf/httpd.conf
```

添加如下行

```
LoadModule dav_module          modules/mod_dav.so
LoadModule dav_fs_module        modules/mod_dav_fs.so
LoadModule dav_svn_module       modules/mod_dav_svn.so
LoadModule authz_svn_module     modules/mod_authz_svn.so
LoadModule mysql_auth_module    modules/mod_auth_mysql.so
```

检查上面的模块在 /usr/local/httpd/modules/ 中存在。

## 七. 创建 SVN 用户的认证数据库

### 1. 登陆 svn 认证的数据库

```
#!/usr/local/mysql/bin/mysql -u root -p
```

### 2. 创建认证数据库名，名字在 AuthMySQLDB manager 指定

```
#mysql>create database manager;
```

### 3. # mysql>use manager;

### 4. 创建数据库表

```
#mysql> create table sysusers (
    -> username char(30) not null,
    -> password char(30) not null,
    -> primary key (username)
```

```
-> );
```

#### 5.向数据库表中插入数据

```
#mysql>insert into sysusers values ('test1',encrypt('123456'));
```

```
#mysql>insert into sysusers values ('test2',encrypt('123456'));
```

```
#mysql>insert into sysusers values ('test3',encrypt('123456'));
```

#### 八.修改访问权限管理文件.auth 文件

```
#vi /var/www/svn/.auth
```

```
[groups]
```

```
group1 = test1
```

```
group2 = test2
```

```
group3 = test3
```

```
[exmple1:/]
```

```
@group1 = rw
```

```
[exmple2:/]
```

```
@group2 = rw
```

```
[exmple3:/]
```

```
@group3 = rw
```

在 httpd.conf 文件末尾添加下面一行:

```
Include conf/extra/httpd-dav.conf
```

修改 httpd-dav.conf 文件为如下:

```
NameVirtualHost *:80
```

```
<VirtualHost *:80>
```

```
LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\"" combined
```

```
ErrorLog logs/svn-error_log
```

```
CustomLog logs/svn-access_log combined
```

<Location /svn/>

AuthType Basic	用户认证类型
AuthName "exmple subversion repository"	
Require valid-user	所有有效用户都可以访问此目录
AuthBasicAuthoritative Off	
AuthUserFile /dev/null	
AuthMySQLEnable On	
AuthMySQLHost localhost	
AuthMySQLPort 3306	
AuthMySQLUser root	连接 mysql 的用户名
AuthMySQLPassword woyo.com.linux	连接 mysql 的密码
AuthMySQLDB manager	验证使用的数据库库名
AuthMySQLUserTable sysusers	用户验证的数据表
AuthMySQLNameField username	用户验证的数据表的用户名字段
AuthMySQLPasswordField password	用户验证的数据表的密码字段
AuthMySQLPwEncryption crypt	用户验证的数据表的密码加密方式
DAV svn	
SVNListParentPath on	
SVNParentPath /var/www/svn/	版本库父目录
AuthzSVNAccessFile /var/www/svn/.auth	权限控制

</Location>

</VirtualHost>

重启 apache，测试。

另外一种认证方式：

使用 mod\_authz\_svn 进行目录访问控制

1. 在 httpd.conf 配置文件末尾添加如下行：

<Location /svn>

DAV svn

SVNParentPath /var/www/svn/ // svn 父目录

AuthzSVNAccessFile /var/www/svn/authzfile // 权限配置文件

AuthType Basic

AuthName "Subversion login" // 连接框提示

AuthUserFile /var/www/svn/authfile // 用户配置文件

Require valid-user //采用何种认证

</Location>

/"Require valid-user"告诉 apache 在 authfile 中所有的用户都可以访问。如果没有它，则只能第一个用户可以访问新建库

## 2. 权限管理

### 1)增加用户 svn 和 svn1

```
# htpasswd -c /var/www/svn/authfile svn
```

//第一次设置用户时使用-c 表示新建一个用户文件。回车后输入用户密码，

```
#htpasswd [-c] /var/www/svn/authfile svn1
```

### 3.权限分配

```
# vi /var/www/wvn/authzfile
```

[groups] //这个表示群组设置

svn1-developers = svn //这个表示某群组里的成员

[example1:/] //这表示，仓库 example1 的根目录下的访问权限

svn = rw

svn1 = r

@svn1-developers = rw //如果在前面加上@符号,则表示这是个群组权限设置

[example2:/] //example2 仓库根目录下的访问权限

svn = rw

svn1 =

[/] //这个表示在所有仓库的根目录下

\* = r //这个表示对所有的用户都具有读权限