# Linux SVN版本控制

简介：

Subversion(简称SVN)是一款自由开放的版本控制软件，Subversion可以管理文件、文件夹以及记录他们的修改状况。SVN常用来帮助我们管理软件开发的源代码或是公司手册文档。SVN通过将文档到入至版本库中进行管理，版本库类似于文件服务器，但比文件服务器更强大的是，它可以记录用户每一次对文件或目录的修改状况，并提供还原数据至老版本的功能。

## SVN服务器对比

客户端访问服务器时可以通过四种方式访问：svnserve访问、svnserve+SSH访问、Apache间接访问 和 本地访问的方式。

svnserve访问：服务器启动一个svnserve服务，svnserve是一个小巧、轻便的服务器程序，客户端通过使用Subversion专用的协议对它进行访问，其特点简单。

svnserve+SSH访问：由于svnserve本身并不提供数据加密通信的功能，所以，如果你需要更加安全的连接，可以使用svnserver+ssh的方式与服务器建立SSH隧道连接，再通过SSH调用svnserve程序，实现数据的安全加密传输。

Apache访问：Apache通过mod\_dav\_svn模块也可以访问版本库，这样客户端可以通过访问Apache取得版本库资料，从而完成对数据的版本控制，其优点是用户可以直接使用浏览器访问版本库

本地访问：使用本地方式连接服务器版本库，这种方式最大的好处是不需要联网，即可对服务器做直接的操作，但实际工作中，客户端和服务器往往不在相同的主机，所以应用相对较少。

**功能 Apache+DAV svnserve svnserve+SSH**

认证 HTTP基本认证、LDAP、证书 支持MD5认证 SSH认证

权限 可对版本库整体设置权限， 可对版本库整体设置权限， 仅对版本库整体

也可对指定目录设置权限 也可对指定目录设置权限 设置权限

加密 支持SSL加密 无 支持SSH加密

日志 完善的Apache日志 无 无

速度 慢 快 快

设置 相对复杂 简单 相对简单

## 安装SVN软件

有两种方式安装Subversion软件。

第一种：yum方式（rpm方式）安装

优点：傻瓜式安装，安装快

缺点：无法进行自定义设置

第二种：源码安装

官方地址：<http://subversion.apache.org/>

镜像源码下载：<http://mirrors.hust.edu.cn/apache/subversion/>



## SVN服务器搭建

### 1，创建服务器版本

利用svnadmin命令可创建服务器版本库。

描述：subversion版本库管理工具，通过svnadmin help可以查看命令帮助。

用法：svnadmin 命令 /版本库路径 [选项]

命令： create 创建一个新的版本库

hotcopy 版本库热备

lslocks 打印所有的锁描述

例如：

[root@centos7 ~]# svnadmin create /var/svn/project1

[root@centos7 ~]# svnadmin hotcopy /var/svn/project1 /var/svn/project\_copy

[root@centos7 ~]# ls /var/svn/

注：新创建的仓库是空的，需要导入数据到仓库

svn import 导入文件至SVN仓库

用法：svn import 本地目录/文件 仓库地址 -m "描述，必填"

例如：

[root@centos7 ~]# svnadmin create /var/svn/web\_project

[root@centos7 ~]# svn import /var/web/ [file:///var/svn/web\_porject](file://var\\svn\\web_porject)/ -m "add data"

上面指令是把本地目录/var/web/导入至本地SVN仓库/var/svn/web\_porject

svn list 列出仓库中的资料列表

用法：svn list 仓库地址

例如：

[root@centos7 ~]# svn list [file:///var/svn/web\_project](file://var\\svn\\web_project)

### 2，认证与授权

使用SVN内置的认证机制可以有效的增强客户端访问版本库的安全性，当客户端访问版本库服务器时，服务器会根据版本库目录下的conf/svnserve.conf文件中定义的认证与授权策略实现权限的控制。

例如：

[root@centos7 ~]# cat /var/svn/web\_porject/conf/svnserve.conf

[general]

anon-access = read

# 设置拒绝匿名账户访问，此处可以设置为none,read,write

auth-access = write

# 经过认证的账户权限为可写

password-db = passwd

# 指定账户名称与密码的存放文件名，该文件在conf目录下

authz-db = authz

# 指定基于路径的访问控制名（可以对文件或目录设置权限）

realm = My First Repository

# 设置版本库域，如果两个版本库的认证域相同，他们将使用相同的密码数据

force-username-case = none

# 对用户名进行大小写转换。可选upper,lower,none

在svnserver.conf中已经配置好了行乎密码文件，默认该文件存放在版本库的conf目录下。在passwd文件中需要设置行乎信息，在authz文件中需要设置访问控制权限。

[root@centos7 ~]# cat /var/svn/web/conf/psswd

....

[users]

# harry = harryssecret # 用户名为harry，密码为harryssecret

# sally = sallyssecret # 用户名为sally, 密码为 sallyssecret

[root@centos7 ~]# cat /var/svn/web/conf/authz

....

[groups]

# harry\_and\_sally = harry,sally # 定义组，组成员有harry,sally

# harry\_sally\_and\_joe = harry,sally,&joe

# [/foo/bar] # 对版本库根路径设置权限，可以设置为需要控制的路径

# harry = rw

# &joe = r

# \* =

# [repository:/baz/fuz]

# @harry\_and\_sally = rw # harry\_and\_sally组中的用户可读，可写权限

# \* = r # 其他所有的人只读，权限可以设置为只读('r')，读写('rw'),无权限('')

### 3，启动svnserve服务

svnserve命令的描述和用法：

描述：SVN服务器程序

用法：svnserve [选项]

选项： -d 以守护进程方式运行svnserve

--listen-port=port 指定监听的端口，默认监听端口号为3690

-r 路径 为版本库指定一个虚拟路径，默认客户端要指定绝对路径访问库。

直接运行svnserve命令即可启动SVN服务进程，但如果需要该服务作为后台程序持续监听客户端访问，可以使用-d选项使该程序以守护进程的方式启动svnserve服务，SVN服务默认监听端口3690，如果防火墙处于开启状态，需要注意对防火墙的正确设置。

例如：

客户端全路径访问：svn://192.168.0.106/var/svn/web

此时需要在authz文件中设置相应权限：

[web:/] 表示web版本库的根

[project:/test] 表示project下的test目录

客户端使用相对路径访问：

服务器启动指定虚拟路径：svnserver -d -r /var/sv/

客户端访问：svn://192.168.0.106/web

### 4,客户端访问

客户端访问SVN服务器有5种方式：

URL格式 含义

[file:///](file://) 直接访问本地磁盘上的版本库（客户端与服务端在一台机器上）

http:// 配置Apache的WebDAV协议，通过网页访问版本库

https:// 与http://相似。但使用了SSL进行数据加密

svn:// 通过svnserve定义的协议版本访问库

svn+ssh:// 与svn://相似，但使用了SSH封装加密数据

在客户端访问服务器版本库的众多方法中，命令行的方式使高效、功能完善、无须安装第三方软件的一种简单方式，SVN软件为我们提供了一个名为svn的命令行程序。

1. svn命令

描述：subversion 客户端命令行工具

用法：svn 命令 [选项]

选项：--password 密码

--username 用户名

--revision(-r) 指定要检查的特定版本

命令：add 添加文件、目录或符号连接

cat 输出特定文件内容

checkout URL[@REV][PATH]

checkout可以从服务器版本库中复制一份副本至本地，URL定位版本库，通过REV可以下载特定版本的数据，PATH为本地工作副本路径。checkout可以简写为co。

commit 将本地工作副本修改后的内容发布到版本中，简写ci。

copy SRC DST 将工作副本中的一个文件或目录复制至版本库

delete PATH 从本地工作副本中删除一个项目

delete URL 从版本库中删除一个项目

diff 对比两个版本之间的差别

import 提交一个路径的副本至版本库

info 显示本地或远程版本信息

list 列出服务器参考中的数据

update 拉取服务器最新数据

svn实例：

[root@centos ~]# svn checkout [file:///var/svn/project](file://var\\svn\\project) mine

[root@centos ~]# svn co svn://192.168.0.106/project mine

[root@centos ~]# svn co svn://192.168.0.106/web /mine

[root@centos ~]# svn commit -m "描述...."

[root@centos ~]# svn commit -m "ddddd" /mine

[root@centos ~]# svn delete testfile.txt

[root@centos ~]# svn diff

[root@centos ~]# svn import -m "new project" /etc [file:///var/svn/porject](file://var\\svn\\porject)

[root@centos ~]# svn info

[root@centos ~]# svn info svn://192.168.0.106/var/svn/project

[root@centos ~]# svn list svn://192.168.0.106/var/svn/project

1. svnlook命令

除了核心的svn命令，SVN软件包还提供了一个用于对版本库数据信息进行简单查询的工具，svnlook命令可以帮助用户完成这些工作。

描述：subversion检查工具，通过svnlook help 可以查看命令帮助

用法：svnlook 命令 /版本库路径 [选项]

选项：--revision(-r) 指定要检查的特定版本

命令：author 显示作者

cat 显示版本库中的文件内容

date 显示时间标记

log 显示日志消息

tree 显示版本库资料

svnlook实例：

[root@centos ~]# svnlook author /var/svn/project

[root@centos ~]# svnlook author -r 2 /var/svn/project # 查看版本2的作者信息

[root@centos ~]# svnlook author -r 2 /var/svn/project test

# 查看版本2中的test文件内容

## svnserve+SSH服务器搭建

由于svnserve并不支持加密，所以对公司的数据安全危害比较答，基于SSH的svnserve使得客户端可以通过SSH服务调用SVN服务程序。客户端需要使用ssh程序连接远程服务器的SSHD服务，通过SSH认证机制验证账户身份后，在启动svnserve服务。所以服务端不需要提前启动SVN服务。在这种模式下，svnserve.conf配置文件依然可以进行权限控制，服务器部署流程使启动SSH服务，在服务器上创建版本库，向版本库导入数据，最后客户端使用命令或图形方式访问服务器。

具体步骤如下：

### 1，服务器端启动SSHD服务

[root@centos ~]# systemctl resatrt sshd

[root@centos ~]# systemctl enable sshd

### 2，服务器端创建版本库

[root@centos ~]# svnadmin create /var/svn/web

[root@centos ~]# svn import /var/code2 [file:///var/svn/web](file://var\\svn\\web) -m 'web code

### 3，客户端访问

[root@centos ~]# svn co svn+ssh://192.168.0.106/var/svn/web /web\_code2

## Apache+SVN服务器搭建

Apache使目前非常流行的Web服务器软件，它允许在几乎所有的计算机平台上运行，目前觉大多数系统平台都可以部署Apache Web Server，使用Apache Web Server访问版本库需要加载mod\_dav 与 mod\_dav\_svn模块，Apache需要通过这两个模块才可以管理subversion版本库。

好处：

a) subversion 可以使用Apache的多种认证方式

b) 不需要创建系统账户

c) Apache提供了完善的日志功能

d) 可以通过TLS进行数据加密

e) HTTP以及HTTPD可以穿越企业防火墙

f) 客户端简单的通过浏览器访问版本库

搭建步骤：

### 创建服务器版本库

[root@centos ~]# svnadmin create /var/svn/web

[root@centos ~]# svn import /var/web\_code [file://var/svn/web](file://var\\svn\\web) -m 'term 3'

### 安装Apache及相关模块软件

[root@centos ~]# yum -y install httpd

[root@centos ~]# yum -y install mod\_dav\_svn

### 修改Apache配置

为了让Apache Web Server可以读取SVN版本库中的数据，我们需要修改httpd针对SVN的配置文件subversion.conf，在该文件中至少要确保mod\_dav 和 mod\_authz\_svn 两个版本会被加载。另外，为了满足用户对数据安全的需求，我们还可以利用httpd的认证模块实现基于账户和密码的访问机制。

[root@centos ~]# vim /etc/httpd/conf.d/subversion.conf

LoadModule dav\_svn\_module modules /mod\_dav\_svn.so

# 加载dav\_svn模块

LoadModule authz\_svn\_module modules /mode\_authz\_svn.so

# 加载权限设置模块

# 除了这两个模块，还要确保/etc/httpd/conf/httpd.conf文件中dav\_module模块也会被加载

# a) readable and writable by the 'apache' user, and

# 默认Apache的启动用户为apache，所以需要apache用户对版本库有读写权限

# b) labelled with the 'httpd\_sys\_content\_t' context if using SELinux

# ruguo kaiqi le SELinux，则版本库目录需要有 httpd\_sys\_content\_t安全上下文标签

......

# 以下内容是配置文件的操作步骤

# # cd /var/www/svn

# # svnadmin create stuff

# # chown -R apache.apache stuff

# # chcon -R -t httpd\_sys\_content\_t stuff

#

<Location /svn>

# 当用户的URL是以/svn开始(http://hostname/svn/...)，Apache会将控制权交于DAV处理

DAV svn

SVNParentPath /var/svn

# 通过SVNParentPath 指定所有版本库的主目录（发布所有的版本库）

# 也可以通过SVNPath 指定某个特定版本库的路径（发布特定版本库）

AuthType Basic # Apache认证方式

AuthName "Please input passwd" # 提示字符

AuthUserFile /var/svn/.pass # 账户文件路径，为了安全，应隐藏该文件

Require valid-user # 设置仅为有效用户可以访问

</Location>

[root@centos ~]# chown -R apache.apache /var/svn # 修改权限

[root@centos ~]# chcon -R -t httpd\_sys\_content\_t /var/svn

# 仅当 SELinux开启时使用

[root@centos ~]# htpasswd -c /var/svn/.pass jerry

# 常见账户文件，添加jerry用户

[root@centos ~]# systemctl start httpd

[root@centos ~]# systemctl enable httpd

### 客户端访问

在浏览器输入网址：192.168.0.106/svn/web 访问客户端

输入用户名和密码即可svn查看版本库的数据。

## 常见问题

### 1，访问版本库路径错误

如果通过svnserver命令启动服务时指定了发布的具体版本库，则客户端访问时就可以直接使用相对路径访问，如果服务端使用了svnserver -d -r /var/svn/启动服务后台，客户端依然使用svn co svn://192.168.0.106/var/svn/web，系统将提示svn:No repostiory found，说明没有找到该版本库，正确的写法是svn co svn://172.168.0.118/web

### 2，每次想访问版本库时，进程就会挂起

首先确保版本库没有被损坏，数据也没有丢失。当进程直接访问版本库时，进程将通过Berkeley DB来实现。Berkeley DB包含日志系统，也就是说，所有的操作在执行前都被记录在日志中。当进行奔溃时，遗留下文件锁，并记录了所有未完成操作的信息，从而导致所有试图访问数据库的进程将因为要访问文件锁而被挂起。若想解除文件锁，可以回滚到前一个正常状态。

### 3，subversion 提示工作副本已过时

这可能是提交失败导致了你得工作副本被破坏，可以使用svn revert回滚、svn uodate更新来解决该问题。

### 4，Windows XP 下，subversion服务器有时会发布一些错误数据

如果你是在Windows平台下搭建subversion服务器，可以尝试通过安装SP1补丁包解决该问题。

### 5，Apache拒绝访问

通过浏览器访问版本库时提示：You don't have permission to access /svn on this server。出现这种提示说明没有权限读取相应目录。

如用户在这里输入：<http://192.168.0.106/svn> ,就会得到这样的提示

正确的输入应为：http://192.168.0.106/svn/web。