**Copied almost entirely from:**

**http://v2.robbyt.com/2008/howto/chrooted-sftp-with-openssh-5/**

**Install OpenSSH 5.0 and libedit (RPM’s in ~cjohnson/SSH\_5.0/**

Yum install gtk2 pkgconfig

rpm -ivh libedit-2.10-4.20070831cvs.i386.rpm libedit-devel-2.10-4.20070831cvs.i386.rpm

Rpm –Uvh libedit-2.10-4.20070831cvs.i386.rpm

Rpm –Uvh openssh-askpass-5.0p1-1.i386.rpm

Rpm –Uvh libedit-devel-2.10-4.20070831cvs.i386.rpm

Rpm –Uvh openssh-clients-5.0p1-1.i386.rpm

Rpm –Uvh openssh-5.0p1-1.i386.rpm

Rpm –Uvh openssh-server-5.0p1-1.i386.rpm

Or, if you want to back-port OpenSSH5.0 from FC9, follow these steps on a separate machine:

### Compile fedora openssh5

wget ftp://mirror.nyi.net/fedora/linux/releases/9/Fedora/source/SRPMS/openssh-5.0p1-1.fc9.src.rpm  
wget ftp://mirror.nyi.net/fedora/linux/releases/9/Fedora/source/SRPMS/  
libedit-2.10-4.20070831cvs.fc9.src.rpm  
mkdir -p /usr/src/redhat/RPMS/i386 && mkdir /usr/src/redhat/BUILD  
rpm -ivh openssh-5.0p1-1.fc9.src.rpm && rpm -ivh libedit-2.10-4.20070831cvs.fc9.src.rpm  
yum -y install gtk2-devel libX11-devel ncurses-devel nss-devel xauth rpm-build (and any other dependencies)  
rpmbuild -bb /usr/src/redhat/SPECS/libedit.spec  
cd /usr/src/redhat/RPMS/i386  
rpm -ivh libedit-2.10-4.20070831cvs.i386.rpm && rpm -ivh libedit-devel-2.10-4.20070831cvs.i386.rpm  
rpmbuild -bb /usr/src/redhat/SPECS/openssh.spec  
rpm -Uvh /usr/src/redhat/RPMS/i386/openssh\*

**Copy pam\_script.so into /lib/security/**

(this file can be found in my home directory)

Cp ~cjohnson/pam\_script.so /lib/security

**Modify /etc/ssh/sshd\_config to include the following lines (your environment may vary):**

# Use the following line to \*replace\* any existing 'Subsystem' line

Subsystem sftp internal-sftp

# These lines must appear at the \*end\* of sshd\_config

Match Group sftponly

ChrootDirectory %h

ForceCommand internal-sftp

AllowTcpForwarding no

This means that all users you add to the 'sftponly' group will be chroot'd to their home directory, and will only be able to run the internal SFTP process.

**Create a new user group to add users to (this determines whether they are chroot'd or not), using:**

groupadd sftponly.

**Service sshd restart after sshd\_config changes!**

**Now, configure your users' accounts as follows (don't forget to do this with new users as well):**

**useradd –g sftponly –m –s /bin/false**

* Set their group (usermod -g) to sftponly (the group you created in the previous step)
* Set their shell (usermod -s) to /bin/false (to deny them shell access)
* Set their home directory (usermod -d) as you prefer
* **Important** (OpenSSH tests for this condition): ensure their home directory is owned by root, and is not writable by any other user or group. This must also be the case for each directory in the path up to the root of your system.
* Ensure the default UMASK is set appropriately - I found my users got confused if their files didn't automatically set themselves to rwxr-xr-x (755), so I added CMASK=022 to /etc/default/login

**Drop this system-auth config into /etc/pam.d/system-auth (backup t he original!)**

#**cat /etc/pam.d/system-auth**

#%PAM-1.0

# This file is auto-generated.

# User changes will be destroyed the next time authconfig is run.

auth required pam\_env.so

auth sufficient pam\_unix.so nullok try\_first\_pass

auth requisite pam\_succeed\_if.so uid >= 500 quiet

auth sufficient pam\_winbind.so cached\_login use\_first\_pass

auth required pam\_deny.so

account required pam\_unix.so broken\_shadow

account sufficient pam\_localuser.so

account sufficient pam\_succeed\_if.so uid < 500 quiet

account [default=bad success=ok user\_unknown=ignore] pam\_winbind.so cached\_login

account required pam\_permit.so

#this creates the homedir if it doesn't exist

session required pam\_mkhomedir.so skel=/etc/skel/

#this changes permissions of user's home to root so chroot will work

session required pam\_script.so runas=root onsessionopen=/root/scripts/chown\_home.sh

password requisite pam\_cracklib.so try\_first\_pass retry=3

password sufficient pam\_unix.so md5 shadow nullok try\_first\_pass use\_authtok

password sufficient pam\_winbind.so cached\_login use\_authtok

password required pam\_deny.so

session optional pam\_keyinit.so revoke

session required pam\_limits.so

session [success=1 default=ignore] pam\_succeed\_if.so service in crond quiet use\_uid

session required pam\_unix.so

**Create this script:**

[root@dl1rwe210 security]# vi /root/scripts/chown\_home.sh

#!/bin/bash

/usr/bin/id $1 | /bin/grep -q "sftponly"

if [ $? -eq 0 ]

then

/bin/chown root `eval echo ~$1`

exit 0

fi

exit 0

**So when a user logs in, it creates the home directory and changes the ownership to the following example:**

drwxr-xr-x 4 root sftponly 4.0K Sep 3 13:36 ftptest

**Add file staging areas to skel:**

In this tutorial, we’re going to add incoming and outgoing dirs in /etc/skel

mkdir /etc/skel/from\_client && mkdir /etc/skel/to\_client

This way when new user accounts are created, they’ll always have the “to” and “from” directories.