Key-based SELinux-enabled Chroot SCP User

Verified on RHEL 6 and RHEL 7

 This configuration was created to fill the requirement of having an account to scp data from a remote system. There are two packages *(scponly and rssh)* that are recommended when researching this on the net. Both provide the functionality described below, the goal of this undertaking was to understand the in-depth requirements and configuration for a very customized implementation of scp for secure environments.

This document assumes the firewall rules and ssh configurations have already been put into place on the server as well as other necessary operating system hardening configurations. The demo chroot location is ***/data/lockdown*** and the demo account is ***demo***.

## User Account Creation/Configuration

All actions below in this section are to be performed as the root user, or using *sudo* command structure.

* **Create a local user account**

useradd -s /bin/bash -m -d /data/lockdown/demo demo

* **Create the password for the user**

passwd demo

* **Ensure ownership and permissions are set for chroot jailing the user**

chmod 0755 /data

chmod 0755 /data/lockdown

chmod 0755 /data/lockdown/demo

chown root:root /data/lockdown/demo

## Chroot Jail Creation/Configuration

* **Create the necessary directories in the chroot jail.**

mkdir -p /data/lockdown/demo/{.ssh,bin,dev,etc,incoming,lib64}

* **Fix ownership/permissions/SELinux contexts of the newly-created directories.**

chmod 0750 /data/lockdown/demo/{.ssh,incoming}

chown demo:demo /data/lockdown/demo/{.ssh,incoming}

chmod 0755 /data/lockdown/demo/{bin,dev,etc,lib64}

semanage fcontext -a -t bin\_t /data/lockdown/demo/bin

restorecon /data/lockdown/demo/bin

semanage fcontext -a -t device\_t /data/lockdown/demo/dev

restorecon /data/lockdown/demo/dev

semanage fcontext -a -t etc\_t /data/lockdown/demo/etc

restorecon /data/lockdown/demo/etc

semanage fcontext -a -t lib\_t /data/lockdown/demo/lib64

restorecon /data/lockdown/demo/lib64

**\*\*\* For RHEL 7 the following steps must also be performed \*\*\***

mkdir -p /data/lockdown/demo/usr/bin

chmod -R 0755 /data/lockdown/usr

semanage fcontext -a -t usr\_t /data/lockdown/demo/usr

restorecon /data/lockdown/demo/usr

semanage fcontext -a -t bin\_t /data/lockdown/demo/usr/bin

restorecon /data/lockdown/demo/usr/bin

**\*\*\* END SECTION SPECIFIC TO RHEL 7 \*\*\***

* **Copy the binaries for bash, scp, and ssh in to the necessary folder**

cp -a /bin/bash /data/lockdown/demo/bin

cp -a /usr/bin/{scp,ssh} /data/lockdown/demo/bin

**\*\*\* For RHEL 7 the scp command must be placed in** */data/lockdown/demo/usr/bin* **\*\*\***

cp -a /usr/bin/scp /data/lockdown/demo/usr/bin

**\*\*\* END SECTION SPECIFIC TO RHEL 7 \*\*\***

* **Copy the necessary special files for chroot bash**

cp -ar /dev/{null,ptmx,pts,random,stderr,stdin,stdout,systty,tty,tty0,urandom,zero}

/data/lockdown/demo/dev (There's a space after “zero}” but text-wrapping broke it)

* **Determine which shared libraries are required for chroot bash**

ldd /bin/bash

The output should read:

 linux-vdso.so.1 => (0x00007ffeb6ff4000)

 libtinfo.so.5 => /lib64/libtinfo.so.5 (0x00007f8988e14000)

 libdl.so.2 => /lib64/libdl.so.2 (0x00007f8988c10000)

 libc.so.6 => /lib64/libc.so.6 (0x00007f898884c000)

 /lib64/ld-linux-x86-64.so.2 (0x000055d3be1c8000)

To copy the shared libraries ignore any line that does not point to another file (i.e., the “l*inux-vsdo.so.1*” line above) using the following command structure:

cp -p /lib64/{libtinfo.so.5,libdl.so.2,libc.so.6,ld-linux-x86-64.so.2} /data/lockdown/demo/lib64 (There's a space after “64.so.2}” but text-wrapping broke it)

* **Copy one extra shared library required for the chroot environment**

If this step is not done, the user will be able to ssh, but not scp!

cp -p /usr/lib64/libnss\_files.so.2 /data/lockdown/demo

* **Determine which shared libraries are required for chroot scp**

ldd /usr/bin/scp

Use the same procedure which was just used above for bash, but there will be many more files to work with in the output. Also make note that some of the output files will point to */usr/lib64* instead of */lib64*. **Regardless of their location, they must all be copied to “/data/lockdown/demo/lib64”!** This means there is no need to create a “*/usr/lib64*” in /data/lockdown/demo. **For RHEL 7 all files will be found under /lib64 .**

* **Determine which shared libraries are required for chroot ssh**

ldd /usr/bin/ssh

Use the same procedure which was just used above for bash, but there will be many more files to work with in the output. Also make note that some of the output files will point to */usr/lib64* instead of */lib64*. **Regardless of their location, they must all be copied to “/data/lockdown/demo/lib64”!** This means there is no need to create a “*/usr/lib64*” in /data/lockdown/demo. **For RHEL 7 all files will be found under /lib64 .**

* **Populate**  *etc/passwd* **in the chroot jail**

grep demo /etc/passwd > /data/lockdown/demo/etc/passwd

chmod 0644 /data/lockdown/demo/etc/passwd

semanage fcontext -a -t etc\_t /data/lockdown/demo/etc/passwd

restorecon /data/lockdown/demo/etc/passwd

* **Populate** *etc/nsswitch.conf* **in the chroot jail**

cp -p /etc/nsswitch.conf /data/lockdown/demo/etc

Edit */data/lockdown/demo/etc/nsswitch.conf* so that the passwd, shadow, and group lines read as follows:

passwd: files

shadow: files

group: files

* **Edit** */etc/ssh/sshd\_config* **to chroot the new user account**

Add the following text block **BEFORE ANY “Match” BLOCK** since this will be username specific

Match User demo

 ChrootDirectory /data/lockdown/%u

 X11forwarding no

 AllowTcpForwarding no

* **Restart the sshd service**

## **Public/Private Key Generation/Configuration**

* **Switch to user account**

su – demo

* **Generate a keypair**

ssh-keygen (when prompted, set the key name to: /data/lockdown/demo/.ssh/demo\_rsa)

Press the Enter key twice to generate the keypair with no password (or specify one if you want).

* **Copy the contents of the public key in the the** *authorized\_keys* **file**

cat /data/lockdown/demo/.ssh/demo\_rsa.pub > /data/lockdown/demo/.ssh/authorized\_keys

(There's a space after “>” but text-wrapping broke it)

* **Ensure permissions on the newly-created** *authorized\_keys* **file are set**

chmod 0600 /data/lockdown/demo/.ssh/authorized\_keys

semanage fcontext -a -t user\_home\_t /data/lockdown/demo/.ssh/authorized\_keys

restorecon /data/lockdown/demo/.ssh/authorized\_keys

* **Switch back to the root account**

exit

* **Provide the username and PRIVATE KEY** *(/data/lockdown/demo/.ssh/demo\_rsa)* **to whatever POC or system owner requires the account.**