**Fwknop w/ single packet Authorization (SPA)**

**source: https://www.cipherdyne.org/fwknop/docs/fwknop-tutorial.html**

sudo apt-cache search fwknop

sudo apt-get install fwknop-server \*install fwknop if you haven’t

sudo netstat -alnp | grep ' LISTEN '

sudo apt-cache search openssh

sudo apt-get install openssh-server \*install openssh server if you haven’t

ifconfig \*get IP

sudo apt-cache search fwknop

sudo apt-get install fwknop-client \*install fwknop client

ifconfig

client=138.47.128.35 \*get client IP

server=138.47.134.184 \*get server IP

ssh\_port=22 \*port #

fwknop -A tcp/$ssh\_port -a $client -D $server --key-gen --use-hmac --save-rc-stanza

grep KEY ~/.fwknoprc \*generate fwknop key

scp ~/.fwknoprc $server:~/access.conf \*copy key to server

**on server**

sudo vim ~/access.conf /etc/fwknop/access.conf \*

**note the format of access.conf should be as follows (make sure to comment [default]):**

SOURCE <ip>

OPEN\_PORTS <prot>/<port>

KEY\_BASE64 <key>

HMAC\_KEY\_BASE64 <key>

rm ~/access.conf

sudo vim /etc/default/fwknop-server \*enable fwknop

**change START\_DAEMON="no" to START\_DAEMON="yes"**

sudo vim /etc/fwknop/fwknopd.conf \*specifiy listening interface

**make sure that the interface is correct: PCAP\_INTF <interface> \*may need to uncomment**

sudo service fwknop-server restart \*restart fwknop server

tail /var/log/syslog \*checks log

sudo service fwknop-server status \*checks fwknop status

**on client**

ssh $server \*test SSH

**on server**

ifconfig \*get interface

**add firewall rules**

sudo iptables -L –nv

int=enp0s3

ssh\_port=22 sudo iptables -A INPUT -i $int -p tcp --dport $ssh\_port -j DROP

sudo iptables -L –nv

**on client**

ssh $server \*test SSH

sudo apt-get install nmap \*install nmap

sudo nmap -sS -p $ssh\_port $server \*check if SSH is visible on server

fwknop -n $server \*send SPA packet to server

**on server**

sudo iptables -L –nv \*checks firewalls rules

**on client**

sudo nmap -sS -p $ssh\_port $server

TEST SSH

**NOTE: may need to resend SPA packet to the server (if the above takes too long)**

**on a different machine:**

test SSH now (shouldn't work):

server=138.47.134.184

ssh $server

**on server:**

we still get locked out once the knocking window closes; so, allow related and established connections in iptables:

sudo iptables -D INPUT -i $int -p tcp --dport $ssh\_port -j DROP

sudo iptables -A INPUT -i $int -p tcp --dport $ssh\_port -m state --state RELATED,ESTABLISHED -j ACCEPT sudo iptables -A INPUT -i $int -p tcp --dport $ssh\_port -j DROP

**Chroot Jail**

**http://allanfeid.com/content/creating-chroot-jail-ssh-access**

**https://linuxconfig.org/how-to-automatically-chroot-jail-selected-ssh-user-logins https://krisko210.blogspot.com/2014/04/ssh-chroot-jail.html**

sudo –i \*need to be at root

cd ~

mkdir /var/jail \*make jail/can be anything

ldd /bin/bash \*see what bash has

cd /var/jail/

mkdir bin lib lib64

cp /lib/x86\_64-linux-gnu/libtinfo.so.5 lib

cp /lib/x86\_64-linux-gnu/libdl.so.2 lib

cp /lib/x86\_64-linux-gnu/libc.so.6 lib

cp /lib64/ld-linux-x86-64.so.2 lib64

cp /bin/bash bin

chroot /var/jail/ \*got jail/or any file you save it as

ls –al

**source: http://linuxcareer.com**

**make sure the file (jail.sh) is in ~ (/root) and that it is executable (chmod u+x jail.sh)**

**GET SCRIPT AND COPY AND PASTE IT!!!**

**Try it with the script**

cd ~

rm -rf /var/jail/ ./jail.sh /bin/{ls,cat,echo,rm,mkdir,bash} /usr/bin/vim /usr/bin/whoami /usr/bin/scp /etc/hosts

mkdir /var/jail/dev \*add useful “special” files

mknod -m 0666 /var/jail/dev/null c 1 3

mknod -m 0666 /var/jail/dev/random c 1 8

mknod -m 0444 /var/jail/dev/urandom c 1 9

**the user will be sphincter; its password will be sphincter we'll add a group to place all jailed users (chrootjail)**

groupadd chrootjail

adduser sphincter

adduser sphincter chrootjail

ls -alh /home \*check user that is added

cp /etc/passwd /etc/group /var/jail/etc/ \*copy sys pw and group file to jail/file

vim /var/jail/etc/passwd /var/jail/etc/group \*remove users and group in jail/file

mkdir /var/jail/home /var/jail/home/sphincter \*add user’s home dir in the jail

cp -r /etc/skel/ /var/jail/home/sphincter \*copy default shell and config files

mv /var/jail/home/sphincter/skel/.\* /var/jail/home/sphincter

rm -rf /var/jail/home/sphincter/skel/

**copy useful bash and vim config files**

cp /home/jgourd/.bashrc /home/jgourd/.bash\_aliases /home/jgourd/.vimrc /var/jail/home/sphincter/

chown -R sphincter:sphincter /var/jail/home/sphincter \*make sure new user owns it’s home dir

vim /etc/ssh/sshd\_config \*alter config to jail users in chrootjail group

**add the following to the configuration file:**

**Match group chrootjail**

**ChrootDirectory /var/jail/**

**X11Forwarding no**

**AllowTcpForwarding no**

service ssh restart

ssh sphincter@localhost \*log in

cp /lib/x86\_64-linux-gnu/libnss\_\* /var/jail/lib/x86\_64-linux-gnu \*if shell doesn’t recognize new user

**log in again**

**Denying password logins and only allowing keys**

***create a key on client(need to log in as user we wish to allow access to): when prompted for pw, hit Enter***

ssh-keygen -t rsa

cat ~/.ssh/id\_rsa.pub >> ~/.ssh/authorized\_keys

vim /etc/ssh/sshd\_config \*change SSH config

**change the following lines (if necessary) as follows:**

PasswordAuthentication no

RSAAuthentication yes

PubkeyAuthentication yes

AuthorizedKeysFile %h/.ssh/authorized\_keys

service ssh restart

ssh user@localhost \*user=what you named

**To undo everything we did**

rm -rf /var/jail

rm ~/jail.sh

deluser --remove-home sphincter \*remove new user

delgroup --only-if-empty chrootjail \*remove chrootjail group

sudo apt-get remove --purge openssh-server \*remove SSH server(or fix config)