Project Submission | Hack Like A Pro | Basic Pentesting

Summary

used netdiscover and nmap to locate the target IP and identify open ports, then focused on port 80 to enumerate a WordPress site using gobuster, revealing a login page at /secret/wp-login.php. You then used niktoandhydrato gather more information and brute-force credentials. Finally, you exploited the WordPress admin panel using thewp_admin_shell_upload module in msfconsole, gained shell access, and cracked a hashed password using johntheripper to escalate to root privileges.

Steps:

1.Recon and Scanning

ip of target:192.168.200.128

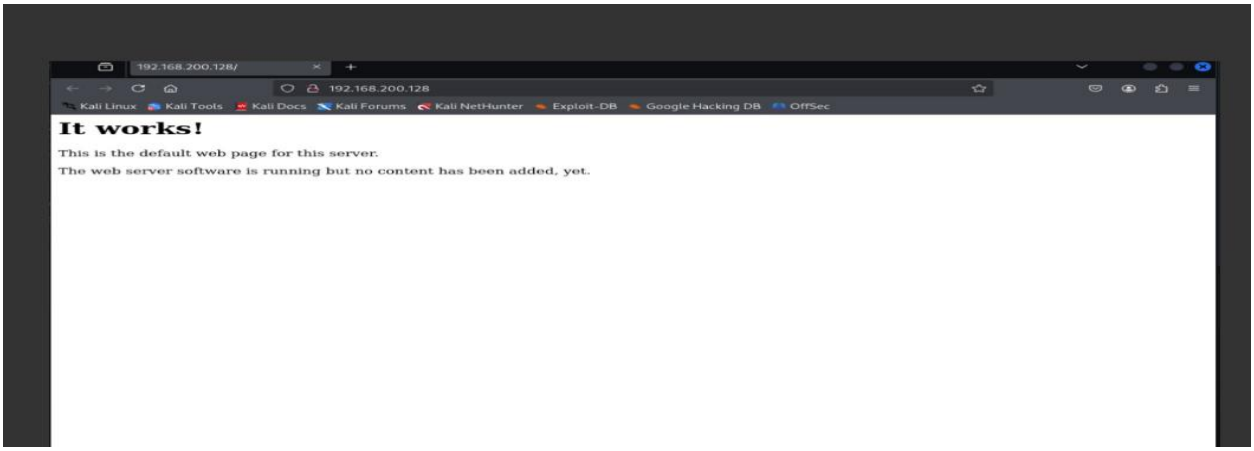
```
┌──(root㉿kali)-[/home/benzalot]
└─# nmap -sC -sV -oN basicpentest_nmap.txt 192.168.200.128
Starting Nmap 7.95 ( https://nmap.org ) at 2025-06-03 07:33 EDT
Nmap scan report for 192.168.200.128
Host is up (0.00026s latency).
Not shown: 997 closed tcp ports (reset)
PORT   STATE SERVICE VERSION
21/tcp open  ftp     ProFTPD 1.3.3c
22/tcp open  ssh     OpenSSH 7.2p2 Ubuntu 4ubuntu2.2 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 d6:01:90:39:2d:8f:46:fb:03:86:73:b3:3c:54:7e:54 (RSA)
|   256 f1:f3:c0:dd:ba:a4:85:f7:13:9a:da:3a:bb:4d:93:04 (ECDSA)
|_  256 12:e2:98:d2:a3:e7:36:4f:be:6b:ce:36:6b:7e:0d:9e (ED25519)
80/tcp open  http    Apache httpd 2.4.18 ((Ubuntu))
|_http-title: Site doesn't have a title (text/html).
|_http-server-header: Apache/2.4.18 (Ubuntu)
MAC Address: 00:0C:29:25:9C:69 (VMware)
Service Info: OSs: Unix, 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 61.98 seconds
```

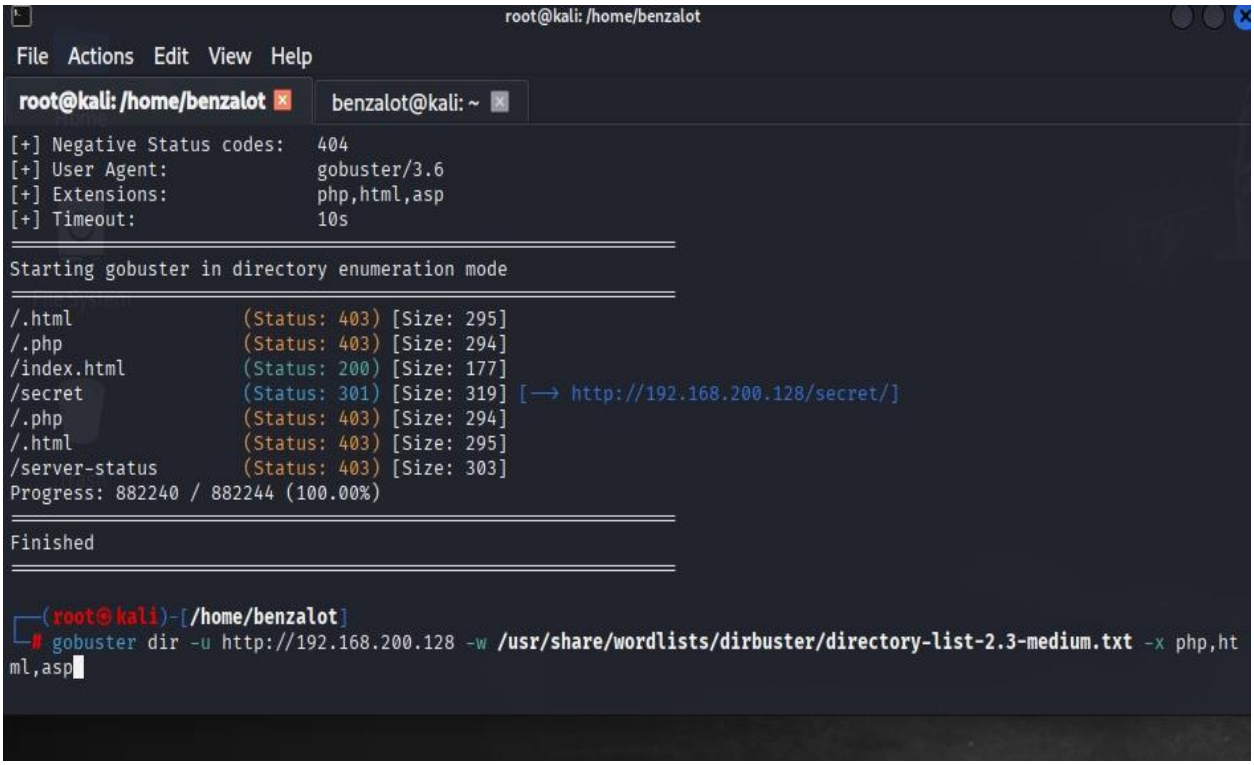
nmap result

2.Enumeration

- **FOUND THE WEBSITE**



- **Used Gobuster to find hidden page**



- Went to the /secret page to investigate

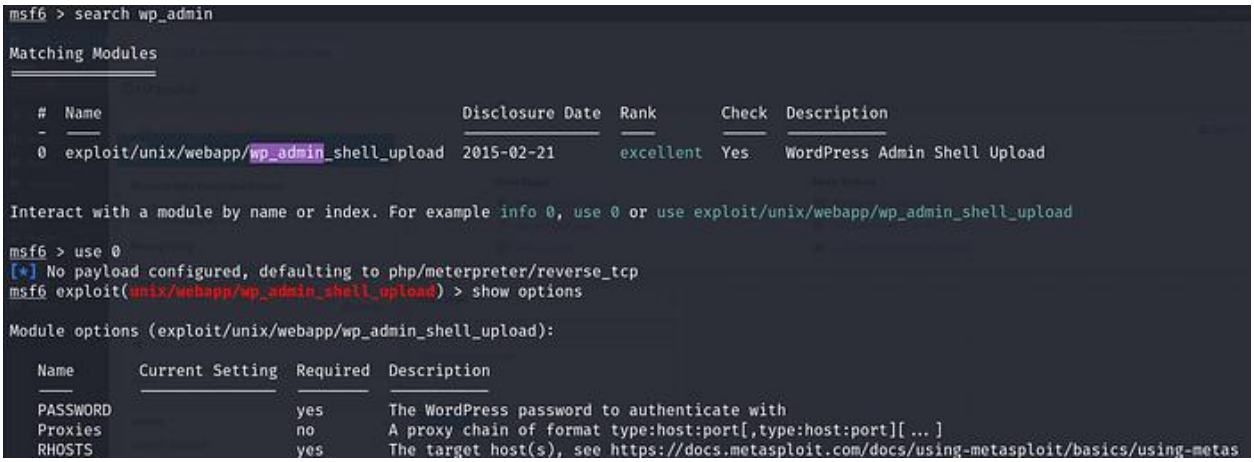
links point to a domain named “vtcsec.” However, it appears to be offline so I added “vtcsec” to my hosts file and attempted access again


```
┌──(root㉿kali)-[/home/benzalot]
└─# nikto -h http://192.168.200.128/secret
- Nikto v2.5.0
---------------------------------------------------------------------------
+ Target IP:          192.168.200.128
+ Target Hostname:    192.168.200.128
+ Target Port:        80
+ Start Time:         2025-06-03 08:21:33 (GMT-4)
---------------------------------------------------------------------------
+ Server: Apache/2.4.18 (Ubuntu)
+ /secret/: The anti-clickjacking X-Frame-Options header
 is not present. See: https://developer.mozilla.org/en-U
S/docs/Web/HTTP/Headers/X-Frame-Options
+ /secret/: Drupal Link header found with value: <http:/
/vtcsec/secret/index.php/wp-json/>; rel="https://api.w.o
rg/". See: https://www.drupal.org/
+ /secret/: The X-Content-Type-Options header is not set
. This could allow the user agent to render the content
of the site in a different fashion to the MIME type. See
: https://www.netsparker.com/web-vulnerability-scanner/v
ulnerabilities/missing-content-type-header/
+ No CGI Directories found (use '-C all' to force check
all possible dirs)
+ Apache/2.4.18 appears to be outdated (current is at le
ast Apache/2.4.54). Apache 2.2.34 is the EOL for the 2.x
 branch.
+ OPTIONS: Allowed HTTP Methods: OPTIONS, GET, HEAD, POS
T .
+ /: Web Server returns a valid response with junk HTTP
methods which may cause false positives.
+ /: DEBUG HTTP verb may show server debugging informati
on. See: https://docs.microsoft.com/en-us/visualstudio/d
ebugger/how-to-enable-debugging-for-aspnet-applications?
view=vs-2017
+ /secret/wp-content/plugins/akismet/readme.txt: The WordPress Akismet pl
ugin 'Tested up to' version usually matches the WordPress version.
+ /secret/wp-links-opml.php: This WordPress script reveals the installed
version.
+ /secret/license.txt: License file found may identify site software.
+ /secret/: A Wordpress installation was found.
+ /secret/wp-login.php?action=register: Cookie wordpress_test_cookie crea
ted without the httponly flag. See: https://developer.mozilla.org/en-US/d
ocs/Web/HTTP/Cookies
+ /secret/wp-login.php: Wordpress login found.
+ 8102 requests: 0 error(s) and 13 item(s) reported on remote host
+ End Time:           2025-06-03 08:21:55 (GMT-4) (22 seconds)
---------------------------------------------------------------------------
```

Found a login page at the bottom of the page used **nikito** to find the vulnerabilities ,**hydra** to bruteforce and finally got admin as username and further used wpscan to find the password: **username:admin** AND **password:admin**

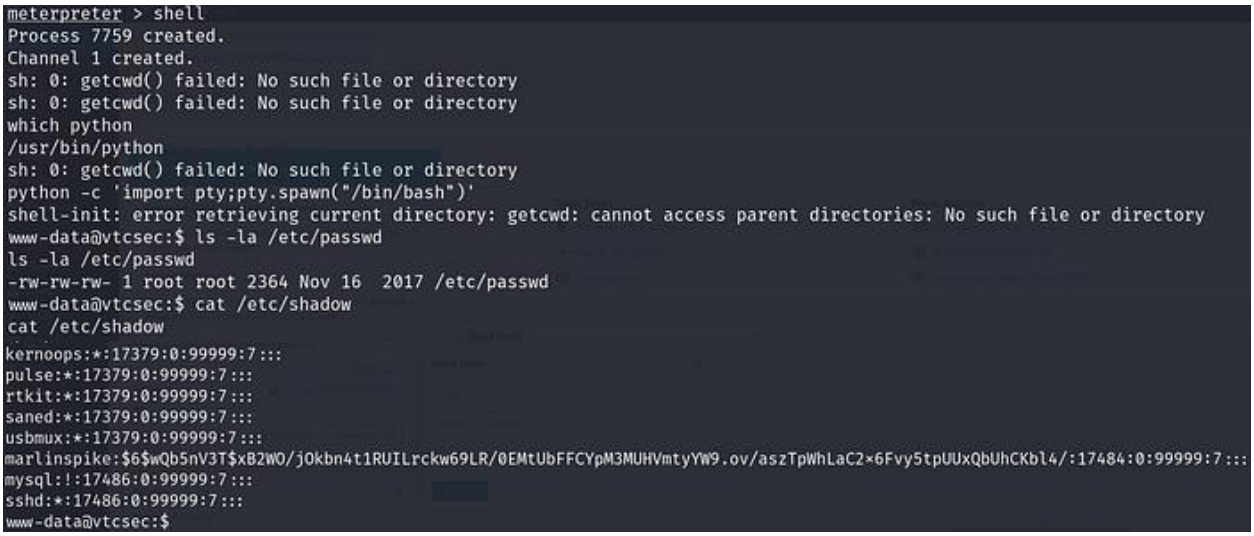
3. Exploitation

Used metaexploit to find the exploit found there's an exploit called **wp_admin**

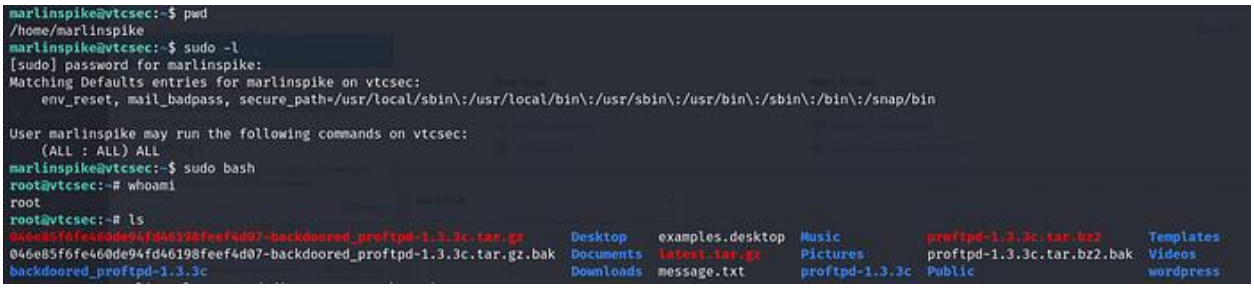


- Set the required options(LHOST,RHOSTS/TARGETURI) and runned the exploit

Upgraded using **python3 -c 'import pty;pty.spawn("/bin/bash")'**



Decrypted the encrypted or hashed password for '**marlinspike'** by using **John The Ripper**.
The password for the Marlinspike has been discovered, and it is '**marlinspike'**.



Lessons Learned

This penetration test on the Basic Pentesting 1 machine provided valuable hands-on experience in real-world ethical hacking techniques. I learned how to effectively use tools like Nmap , netdiscover , Hydra , WPScan , and Metasploit to locate, scan, enumerate, and exploit vulnerabilities in a system.
I gained practical knowledge of how misconfigured or outdated services — such as WordPress login pages and ProFTPD — can be exploited to gain unauthorized access. Additionally, I practiced post-exploitation techniques using John the Ripper to crack hashes and escalate privileges.
Most importantly, this lab reinforced the importance of strong security practices, including:

- Using non-default credentials
- Keeping software up to date
- Securing web applications with firewalls and WAFs

These experiences improved my understanding of both offensive and defensive cybersecurity strategies, preparing me for more advanced penetration testing tasks.

Suggestions for defense

- Use **strong, non-default usernames and passwords** to prevent brute-force attacks.
- **Disable or remove** unnecessary services like FTP (ProFTPD) if not required.
- Keep all software up to date, including **Apache** , **WordPress** , and **OpenSSH** , to avoid exploitation of known vulnerabilities.

- Harden WordPress by restricting access to sensitive areas such as /wp-admin.
- Implement a **Web Application Firewall (WAF)** to detect and block malicious requests.
- Set proper file and directory permissions to prevent unauthorized access or uploads

THANK YOU.

Report by :
Benson Eldho