LABORATORIO 21 GENNAIO 2025 S7-L2

Exploit Telnet con Metasploit

Traccia: Utilizzare Metasploit per sfruttare la vulnerabilità relativa a Telnet con il modulo auxiliary telnet_version sulla macchina Metasploitable.

Requisito: Seguire gli step visti in lezione teorica. Prima, configurare l'IP della macchina Kali con 192.168.1.25 e l'IP della macchina Metasploitable con 192.168.1.40

Svolgimento: Iniziamo avviando le macchine e procediamo con le modifiche dei relativi indirizzi IP.

```
msfadmin@metasploitable:~$ sudo ifconfig eth0 192.168.1.40/24
[sudo] password for msfadmin:
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:3f:28:90
          inet addr:192.168.1.40  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe3f:2890/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:2 errors:0 dropped:0 overruns:0 frame:0
          TX packets:64 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:128 (128.0 B)  TX bytes:4752 (4.6 KB)
          Base address:0xd020 Memory:f0200000-f0220000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:115 errors:0 dropped:0 overruns:0 frame:0
          TX packets:115 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:23317 (22.7 KB)  TX bytes:23317 (22.7 KB)
```

```
┌──(kali㉿kali)-[~]
└─$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:ad:25:87 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.5/24 brd 192.168.1.255 scope global noprefixroute eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::3ade:9ec0:6541:130e/64 scope link noprefixroute
       valid_lft forever preferred_lft forever

┌──(kali㉿kali)-[~]
└─$ sudo ifconfig eth0 192.168.1.25/24
[sudo] password for kali:

┌──(kali㉿kali)-[~]
└─$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:ad:25:87 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.25/24 brd 192.168.1.255 scope global noprefixroute eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::3ade:9ec0:6541:130e/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
```

Una volta modificati gli indirizzi IP accertiamoci che le due macchine comunichino tra loro:

```
┌──(kali㉿kali)-[~]
└─$ ping 192.168.1.40
PING 192.168.1.40 (192.168.1.40) 56(84) bytes of data.
64 bytes from 192.168.1.40: icmp_seq=1 ttl=64 time=0.523 ms
64 bytes from 192.168.1.40: icmp_seq=2 ttl=64 time=0.513 ms
64 bytes from 192.168.1.40: icmp_seq=3 ttl=64 time=0.459 ms
64 bytes from 192.168.1.40: icmp_seq=4 ttl=64 time=7.14 ms
64 bytes from 192.168.1.40: icmp_seq=5 ttl=64 time=0.456 ms
64 bytes from 192.168.1.40: icmp_seq=6 ttl=64 time=0.393 ms
64 bytes from 192.168.1.40: icmp_seq=7 ttl=64 time=0.419 ms
^C
── 192.168.1.40 ping statistics ──
7 packets transmitted, 7 received, 0% packet loss, time 6086ms
rtt min/avg/max/mdev = 0.393/1.415/7.142/2.338 ms

┌──(kali㉿kali)-[~]
```

```
Metaspoitable2 [In esecuzione] - Oracle VirtualBox
File  Macchina  Visualizza  Inserimento  Dispositivi  Aiuto

msfadmin@metasploitable:~$ ping 192.168.1.25
PING 192.168.1.25 (192.168.1.25) 56(84) bytes of data.
64 bytes from 192.168.1.25: icmp_seq=1 ttl=64 time=0.413 ms
64 bytes from 192.168.1.25: icmp_seq=2 ttl=64 time=0.465 ms
64 bytes from 192.168.1.25: icmp_seq=3 ttl=64 time=0.976 ms
64 bytes from 192.168.1.25: icmp_seq=4 ttl=64 time=0.233 ms

--- 192.168.1.25 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 2997ms
rtt min/avg/max/mdev = 0.233/0.521/0.976/0.277 ms
msfadmin@metasploitable:~$ _
```

Tramite scansione di nmap procediamo a controllare lo stato del servizio Telnet sulla porta 23 di metasploitable:

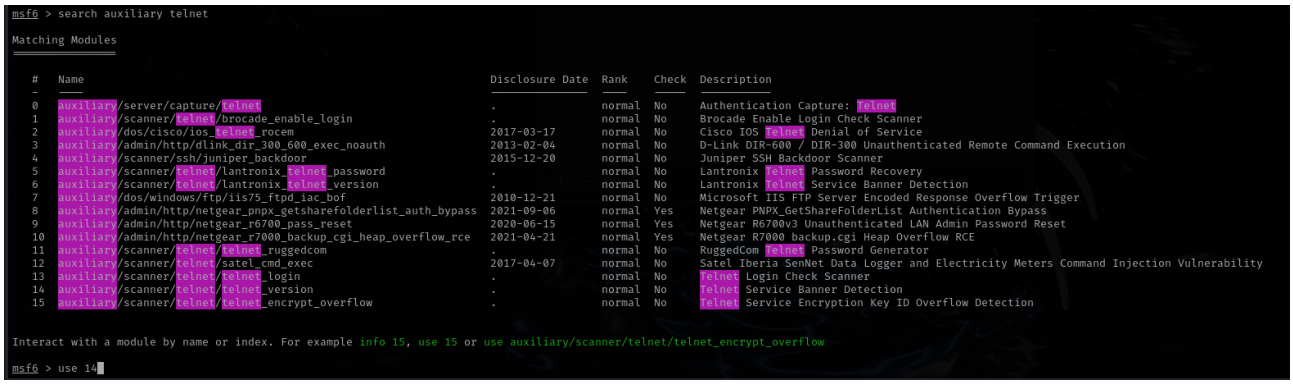
```
┌──(kali㉿kali)-[~]
└─$ nmap -p 23 192.168.1.40
Starting Nmap 7.95 ( https://nmap.org ) at 2025-01-21 09:21 EST
Nmap scan report for 192.168.1.40
Host is up (0.0049s latency).

PORT   STATE SERVICE
23/tcp open  telnet
MAC Address: 08:00:27:3F:28:90 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 6.81 seconds

┌──(kali㉿kali)-[~]
└─$
```

A questo punto avviamo la console di Metasploit framework con il comando msfconsole:

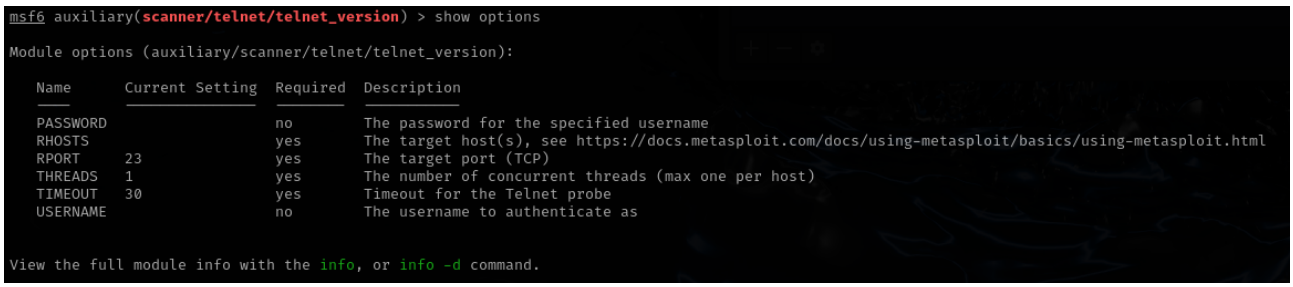
```
┌──(kali㉿kali)-[~]
└─$ msfconsole
Metasploit tip: Enable verbose logging with set VERBOSE true

                .;lxO0KXXXK0Oxl:.
            ,o0WMMMMMMMMMMMMMMMMMMMMMMKd,
         'xNMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMWx,
       :KMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMK:
     .KMMMMMMMMMMMMMMMMMMWNNNWMMMMMMMMMMMMMMMMMMMX,
    lWMMMMMMMMMMMMXd:..      ..;dKMMMMMMMMMMMMMMo
   xMMMMMMMMMMMWd.                .oNMMMMMMMMMMMk
  oMMMMMMMMMMx.                      dMMMMMMMMMMx
 .WMMMMMMMMM:                         :MMMMMMMMMM,
 xMMMMMMMMMo                           lMMMMMMMMMO
 NMMMMMMMMW                       ,cccccoMMMMMMMMWlccccc;
 MMMMMMMMMX                       ;KMMMMMMMMMMMMMMMMMMMX:
 NMMMMMMMMW.                        ;KMMMMMMMMMMMMMMMX:
 xMMMMMMMMMd                          ,0MMMMMMMMMMK;
 .WMMMMMMMMMc                           'OMMMMMM0,
  lMMMMMMMMMMk.                           .kMMO'
   dMMMMMMMMMMWd'                           ..
    cWMMMMMMMMMMMNxc'.                    ##########
     .0MMMMMMMMMMMMMMMMWc                #+#    #+#
       ;0MMMMMMMMMMMMMMMo.               +:+
          .dNMMMMMMMMMMMMo               +#++:++#+
              'oOWMMMMMMMMo                     +:+
                  .,cdkO0K;          :+:    :+:
                                     :::::::+:
                      Metasploit

       =[ metasploit v6.4.38-dev                          ]
+ -- --=[ 2467 exploits - 1273 auxiliary - 431 post       ]
+ -- --=[ 1478 payloads - 49 encoders - 13 nops           ]
+ -- --=[ 9 evasion                                       ]

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

msf6 >
```

Partiamo con la ricerca degli exploit tramite il comando **search auxiliary telnet** e scegliamo quello richiesto dalla traccia, ovvero il numero 14: **auxiliary/scanner/telnet/telnet_version**

Diamo il comando **use 14**

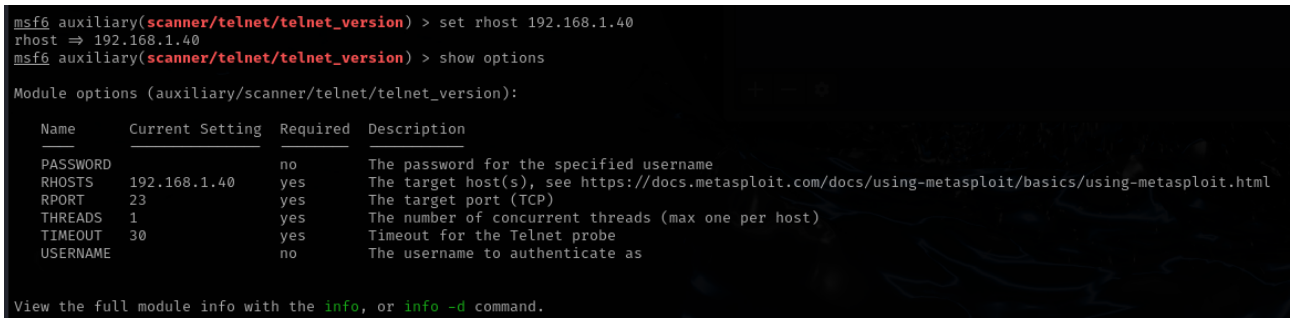


Chiediamo di mostrare le opzioni a disposizione con il comando: **show options**



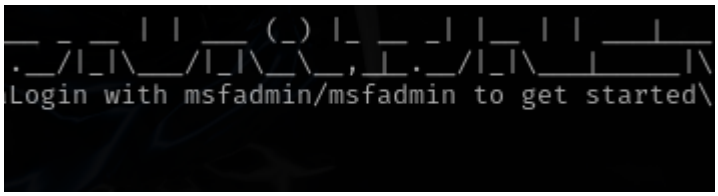
Possiamo notare che è stata impostata automaticamente la porta 23 ovvero quella relativa al Telnet.

Tramite il comando **set rhost** e inseriamo l'indirizzo IP del Target Metasploitable.

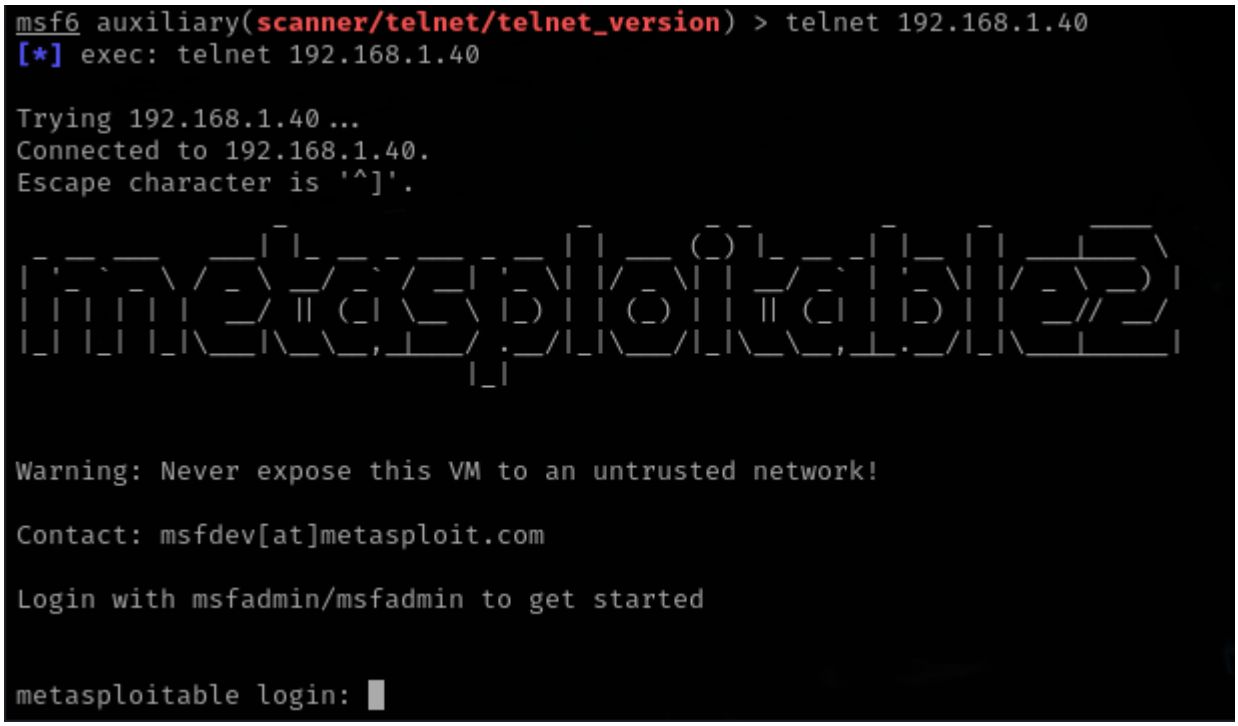


Siamo pronti a lanciare il nostro exploit:

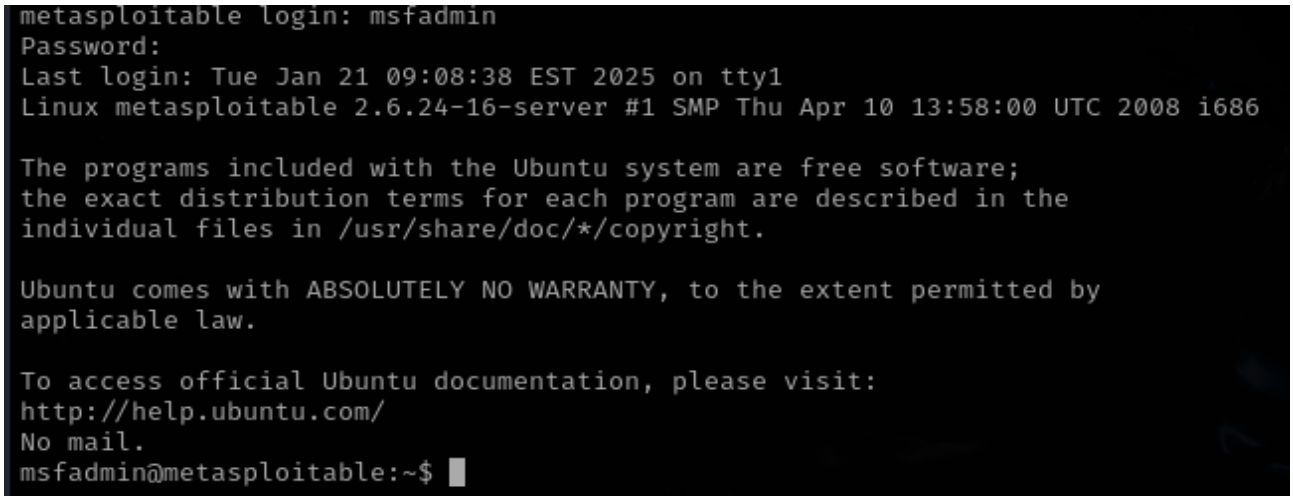
Possiamo notare che ci sono state restituite in output anche le credenziali di accesso a Metasploitable:



A questo punto effettuiamo un test per verificare la correttezza delle informazioni utilizzando il comando **telnet** seguito dall'IP di metasploitable, sarà quindi **telnet 192.168.1.40**



Ci chiede dunque di inserire le credenziali che ci aveva fornito in precedenza: **msfadmin/msfadmin**



Possiamo constatare che l'attacco è andato a buon fine e abbiamo sfruttato la vulnerabilità del servizio Telnet. Siamo infatti riusciti ad ottenere l'accesso non autorizzato.

ESERCIZIO BONUS:

Studiare cos'è il servizio **distcc** e spiegarne il funzionamento. Spiegare inoltre la motivazione dell'esistenza della vulnerabilità e il motivo per il quale si tiene aperta la relativa porta e spiegare se questa è facilmente accessibile.

Effettuare quindi l'attacco al servizio **distccd** ed aprire una shell nella macchina bersaglio.

Spiegazione del servizio e delle vulnerabilità:

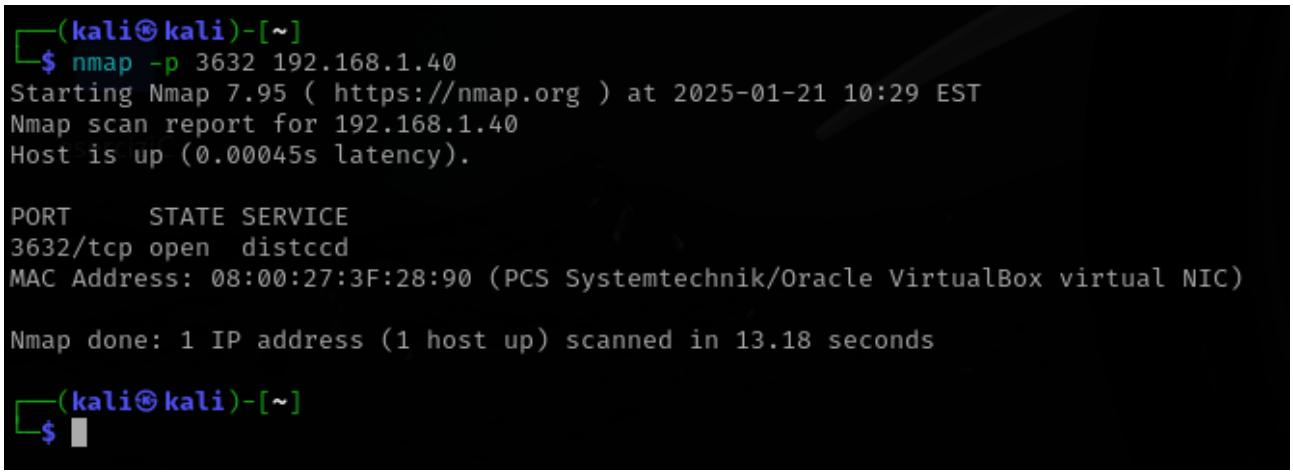
Il servizio **distcc** è un tool per la compilazione distribuita di codice, progettato per velocizzare il processo di build distribuendo il lavoro su più macchine in rete. Funziona coordinando i processi di compilazione: una macchina principale suddivide il lavoro e lo invia ai nodi configurati, che compilano il codice e restituiscono i risultati.

La vulnerabilità del servizio spesso deriva da configurazioni errate che consentono accessi non autenticati o da una mancata restrizione degli IP autorizzati a utilizzare il servizio. La porta standard utilizzata da **distcc** è la **3632**, che rimane aperta per consentire il traffico tra i nodi.

Se lasciata aperta senza restrizioni, questa porta può essere facilmente accessibile da attori malintenzionati, che potrebbero sfruttare il servizio per eseguire comandi arbitrari, installare malware o compromettere ulteriormente il sistema. Per mitigare i rischi, è fondamentale limitare l'accesso alla porta tramite firewall e configurare correttamente le regole di accesso.

Attacco al servizio:

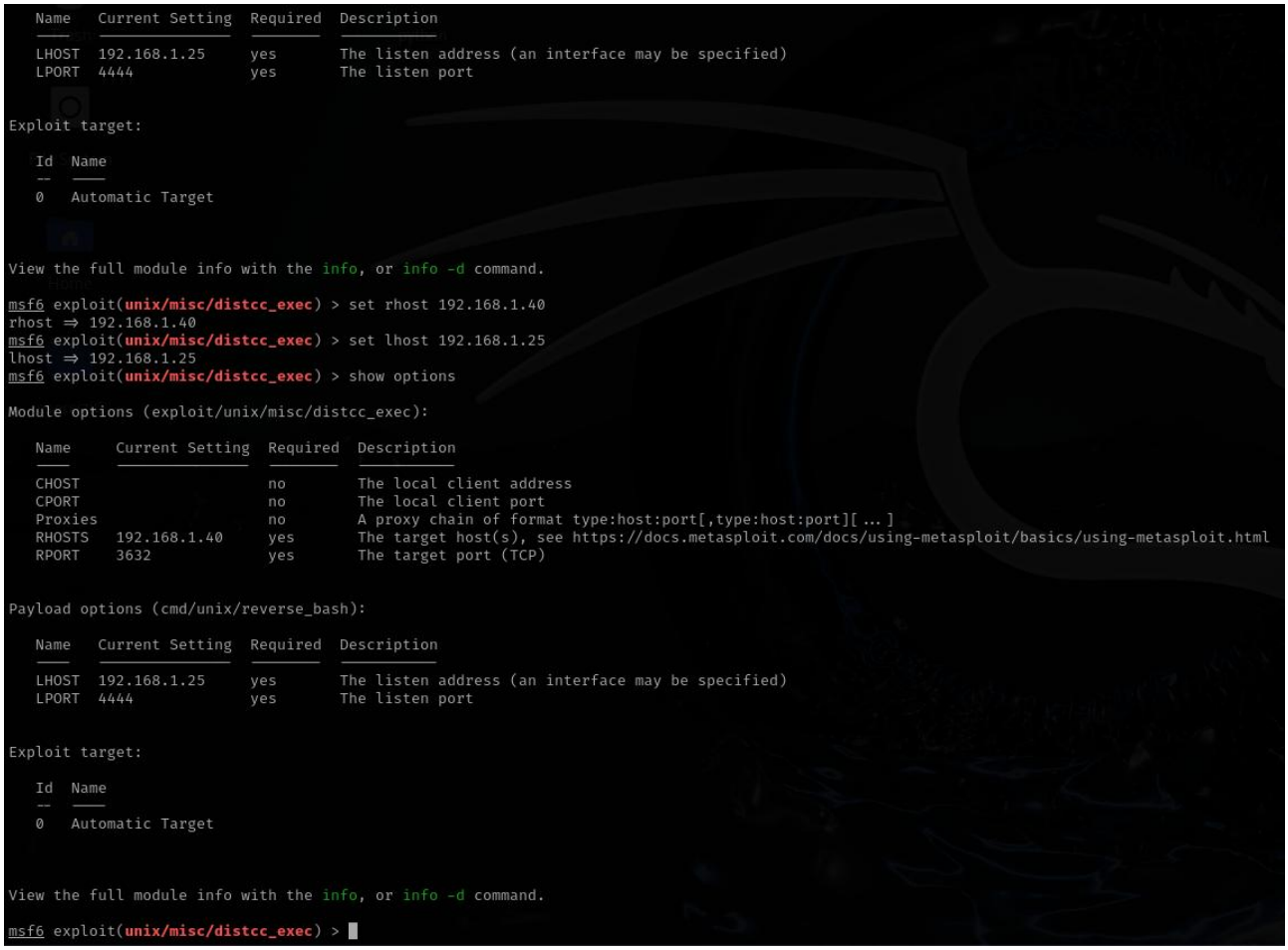
Procediamo ora con l'attacco. Innanzitutto effettuiamo una scansione sulla porta interessata, quindi la **3632.**



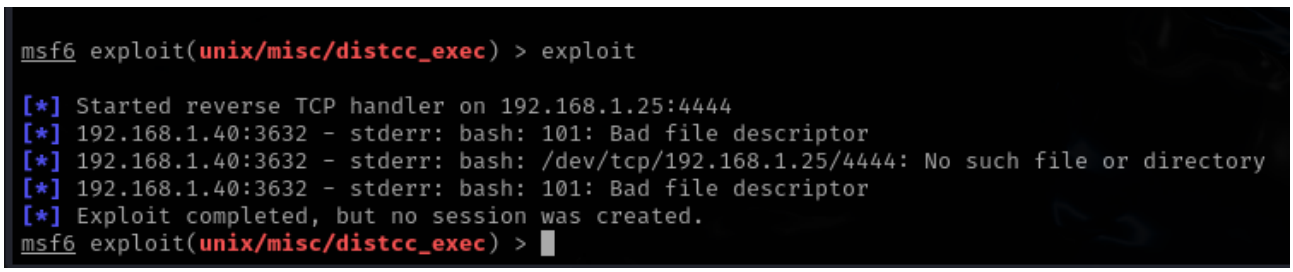
Apriamo di nuovo il tool metasploit framework e cerchiamo un exploit adatto a questo servizio.

Scegliamo l’unico modulo disponibile con il comando **use 0**

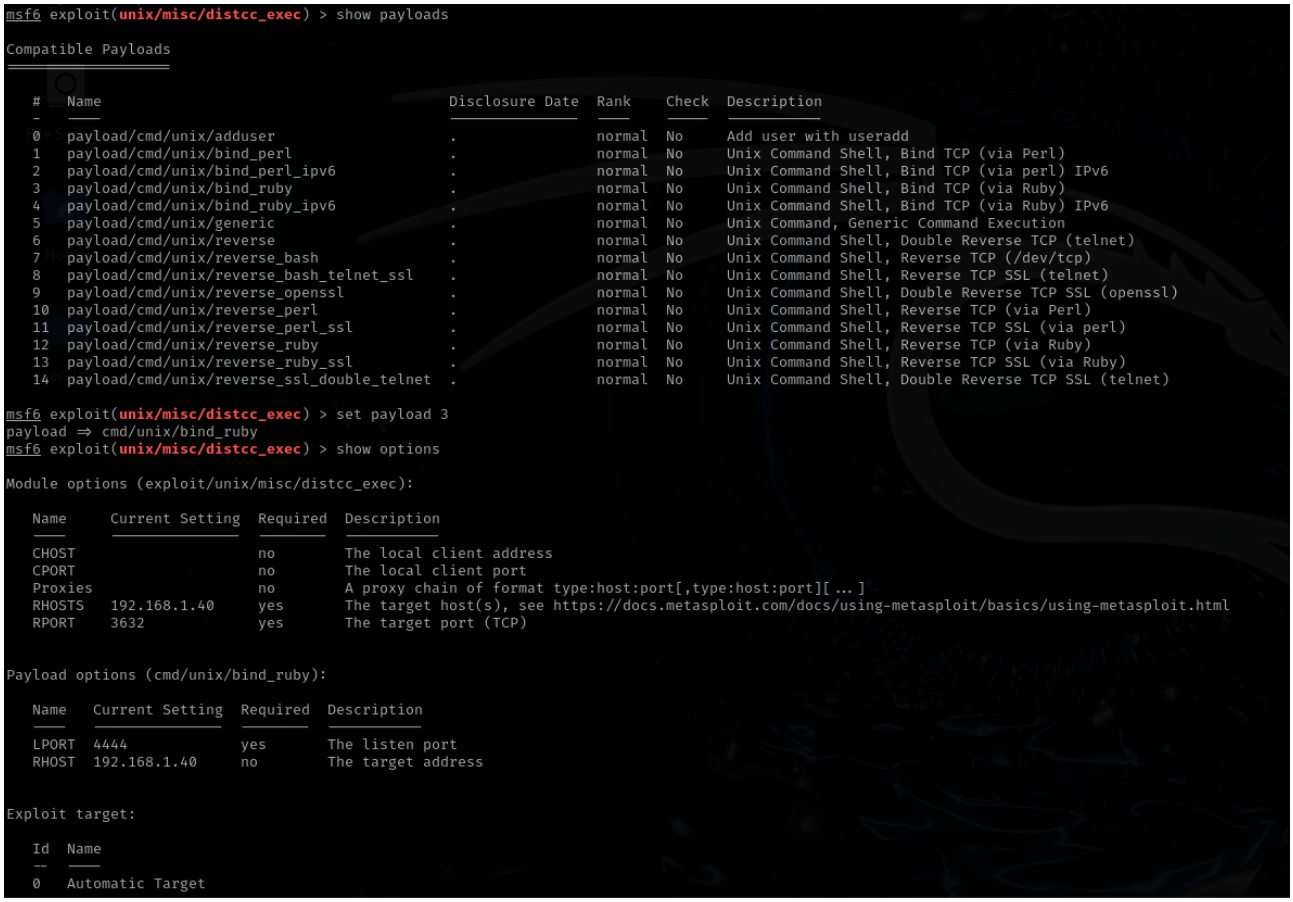
Controlliamo le varie opzioni ed inseriamo gli indirizzi IP come fatto in precedenza, quindi facciamo un check per verificare la buona riuscita delle modifiche:



Eseguiamo quindi l'exploit:



Non essendo andato a buon fine andiamo a modificare il payload da utilizzare scegliendo il numero 3:



Rilanciamo adesso l'exploit che in questo caso andrà a buon fine. Possiamo muoverci all'interno della shell e controllare informazioni sul target come configurazioni di rete e privilegi. In questo caso siamo **daemon.**