Report W16D4 Benchmark 4

REPORT: Sfruttamento della vulnerabilità **Java RMI** su Metasploitable.

Redatto da: Pandolfi Luciano

Data: 20/12/24

INDICE

1. Introduzione della vulnerabilità

La vulnerabilità **Java RMI** (Remote Method Invocation) è un tipo di vulnerabilità legata al protocollo utilizzato da Java per consentire l'invocazione di metodi su oggetti remoti, cioè oggetti che risiedono su altre macchine.

RMI è utilizzato in ambienti distribuiti per la comunicazione tra applicazioni in esecuzione su macchine differenti.

Se non configurato correttamente, può essere sfruttato da un attaccante per eseguire codice malevolo sulla macchina target. (Questo tipo di attacco è noto come **Remote Code Execution (RCE)**.

Con questo documento andremo ad effettuare e descrivere lo sfruttamento della vulnerabilità **Java RMI** (la vulnerabilità in questione è dovuta ad una configurazione di default errata che permette ad un potenziale attaccante di iniettare codice arbitrario per ottenere un accesso amministrativo alla macchina target).

E' presente su un server Metasploitable ed utilizzeremo il framework Metasploit per sfruttarla.

Infatti, sulla porta **1099 TCP** della nostra Metasploitable è attivo un servizio **Java-RMI**, che ci consentirà di ottenere una sessione remota Meterpreter.

Infine, andremo a mostrare quali misure possono essere adottate per risolvere la vulnerabilità.

1.1 Le condizioni della Vulnerabilità

- Il servizio RMI è in ascolto sulla porta 1099 ed è accessibile dall'esterno senza autenticazione.
- Non sono presenti controlli di sicurezza, infatti il server RMI non implementa meccanismi di autenticazione o autorizzazione robusti, né la crittografia nelle comunicazioni.
- Remote Code Execution (RCE) - l'attaccante può utilizzare il sistema RMI per caricare e far eseguire codice malevolo.

2. Sfruttamento della Vulnerabilità

2.1 Dettagli dell'Ambiente:

- **ATTACCANTE:** Kali Linux (rete interna/IP statico 192.168.11.111);
- **TARGET:** Metasploitable (rete interna/IP statico 192.168.11.112).

2.1.1 Configurazione IP Kali Linux e Metasploitable

Ho configurato modificando i rispettivi IP accedendo al file **/etc/network/interfaces** e verificando la corretta configurazione con il comando **ifconfig**.

- Screen:

```
┌──(kali㉿kali)-[~]
└─$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.11.111  netmask 255.255.255.0  broadcast 192.168.11.255
        inet6 fe80::a00:27ff:fe1b:7a46  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:1b:7a:46  txqueuelen 1000  (Ethernet)
        RX packets 189  bytes 23987 (23.4 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 270  bytes 260667 (254.5 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 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 16  bytes 960 (960.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 16  bytes 960 (960.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

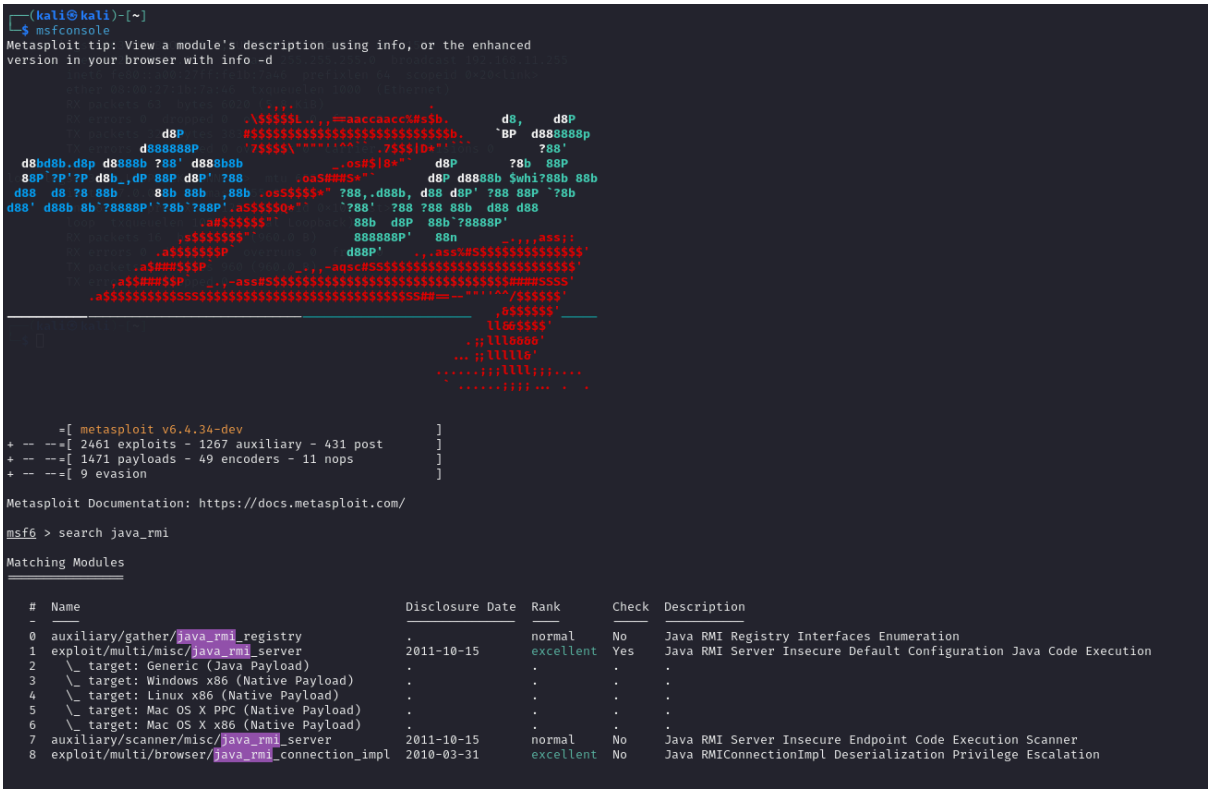
```
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:a9:ae:31
          inet addr:192.168.11.112  Bcast:192.168.11.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fea9:ae31/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:247 errors:0 dropped:0 overruns:0 frame:0
          TX packets:195 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:258741 (252.6 KB)  TX bytes:24383 (23.8 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:167 errors:0 dropped:0 overruns:0 frame:0
          TX packets:167 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:56461 (55.1 KB)  TX bytes:56461 (55.1 KB)
```

2.2 Avvio MSFCONSOLE ed utilizzo del modulo SEARCH/USE/SHOW OPTIONS

Facciamo partire Metasploit da console con il comando **MSFConsole**, e cerchiamo utilizzando la keyword **«search»** un exploit che possa fare al nostro caso.

Nel nostro caso, utilizziamo il comando **«search java_rmi»**.

- Screen:



2.2.1 Selezione dell'Exploit

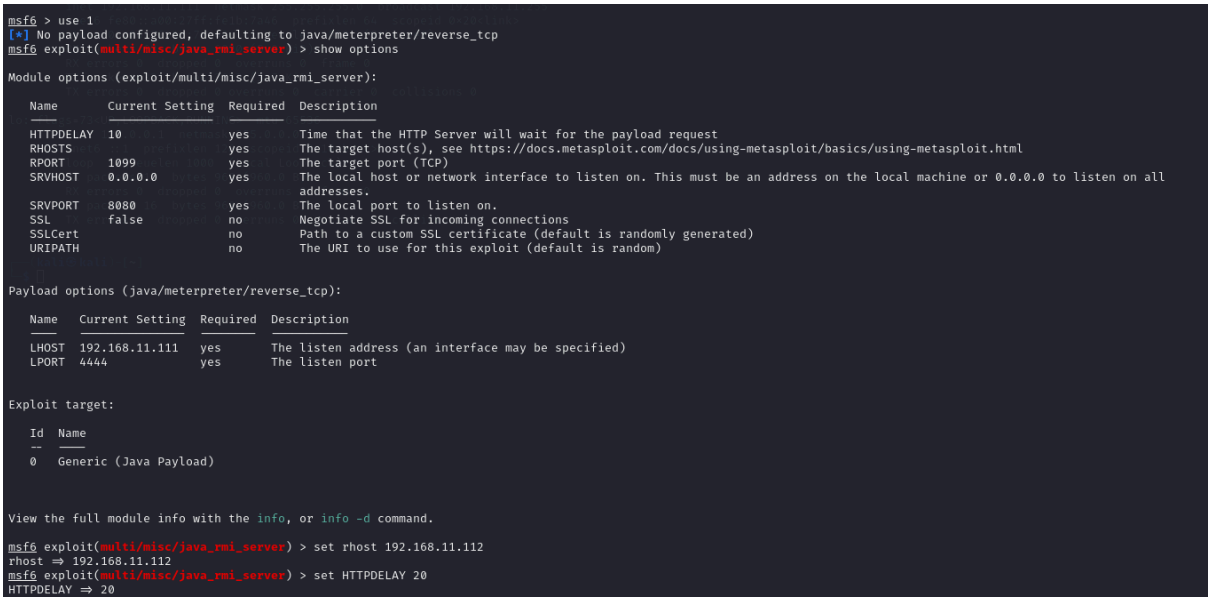
Selezioniamo l'exploit di interesse (Java RMI Server Insecure Default Configuration Java Code Execution) con il seguente comando:

" use 1 " oppure **" use exploit/multi/misc/java_rmi_server ";**

2.2.2 Configurazione dell'Exploit

Visto che alcuni parametri come PAYLOAD, RPORT, LPORT ed LHOST sono già stati inseriti correttamnete di default, modifichiamo in questo caso solamente l'RHOSTS con l'IP della macchina Target (192.168.11.112) e l'HTTPDELAY aumentando i tempi di attesa del server per la richiesta effettuata dal payload (da 10 a 20 secondi).

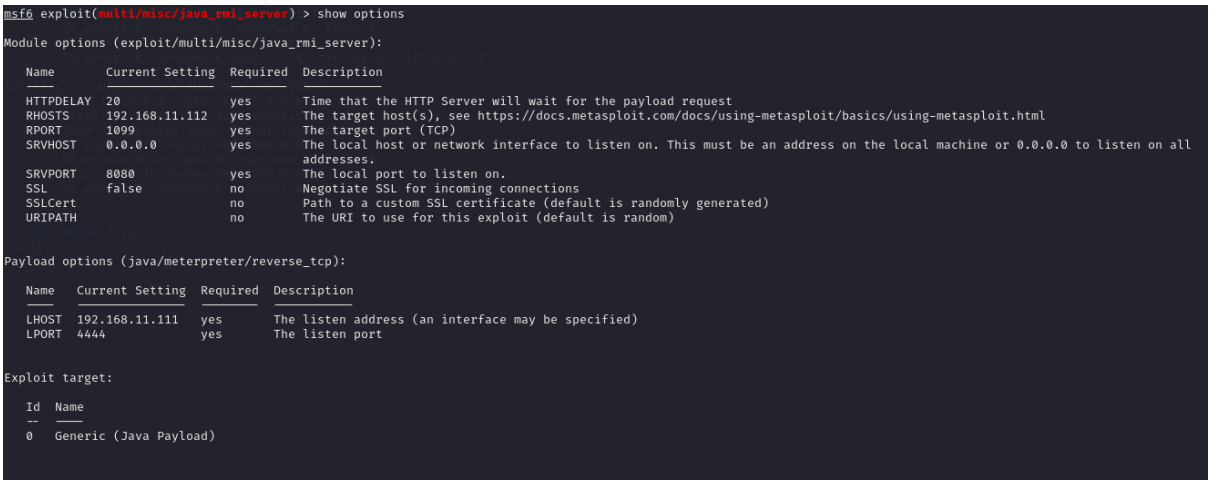
- Sceen:



2.2.3 Verifica delle Configurazioni

Digitiamo nuovamente il comando **SHOW OPTIONS** per essere certi di aver effettuato la corretta configurazione.

- Screen:



2.2.4 Lancio dell'Exploit ed Accesso alla Sessione Meterpreter

Una volta che abbiamo configurato tutte le impostazioni ed i parametri, possiamo lanciare l'attacco con il comando **«exploit»**.

In base al payload che abbiamo utilizzato ci aspettiamo di ricevere, se l'attacco fa a buon fine, una shell di Meterpreter.

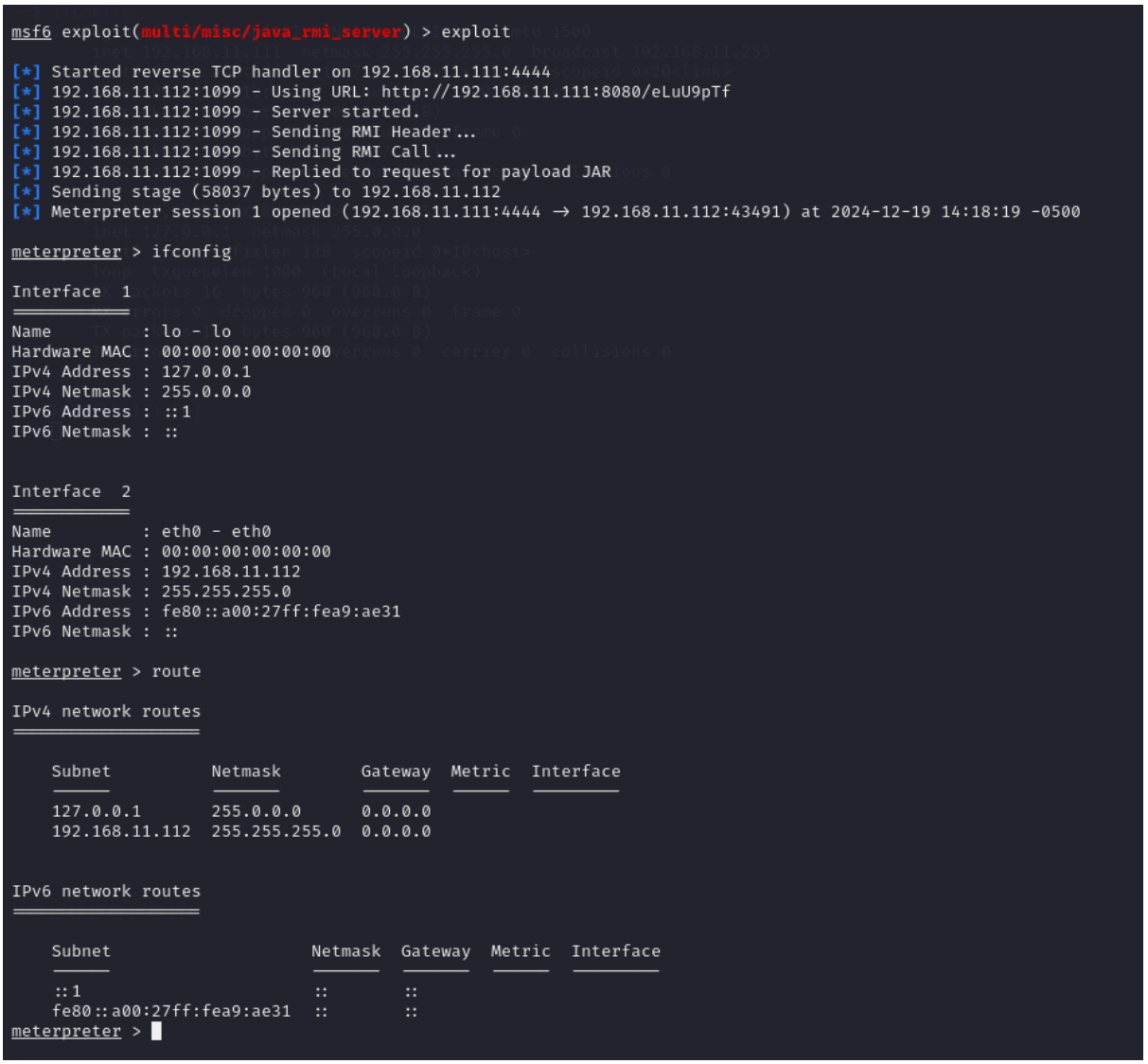
Con successo, abbiamo ottenuto una sessione Meterpreter sulla macchina Target.

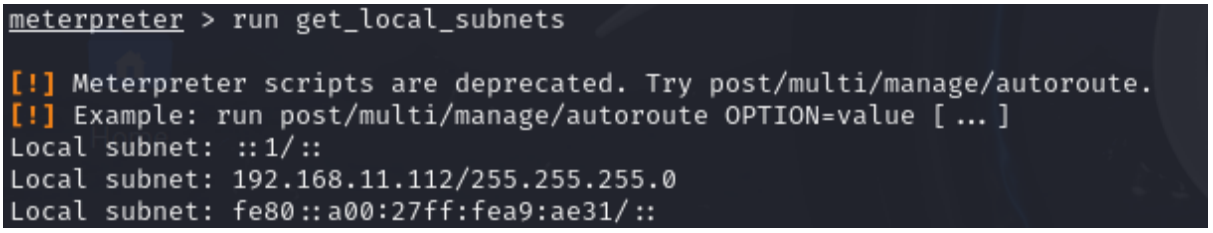
2.2.5 Test di Conferma dell'Accesso alla Macchina Target con Raccolta Informazioni

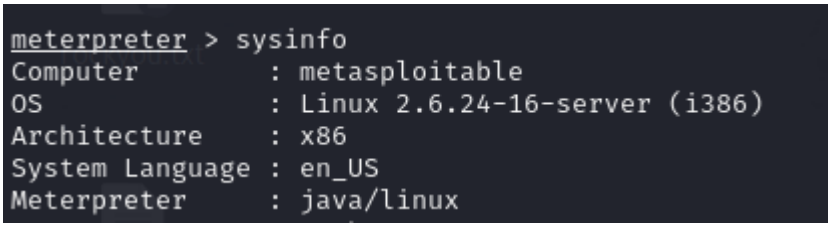
Una volta ottenuto l'accesso alla macchina Target facciamo un paio di test per verificare, che siamo all'interno, andando ad effettuare dei comandi volti a fornire informazioni:

- sulla Configurazione di Rete (ifconfig e run get_local_subnets);
- sulla Tabella di Routing (route);
- sul Sistema Operativo (sysinfo).

- Screen:







2.3 Azioni di remediation su Metasploitable (Vulnerabilità RMI/porta 1099)

Se utilizzi una macchina **Metasploitable** e desideri proteggere la porta **1099** o mitigare i rischi associati alla vulnerabilità RMI, potresti intraprendere alcune misure preventive:

2.3.1 Disabilitare RMI o limitare l'accesso:

- Se non hai bisogno del servizio RMI, la prima cosa da fare è **disabilitarlo**. Puoi sospendere il servizio RMI sulla macchina di destinazione.
- Se hai bisogno del servizio, **limita l'accesso** alla porta 1099 solo a macchine o utenti specifici, utilizzando policy di un firewall o un altri sistemi di controllo degli accessi.

2.3.2 Configura correttamente l'autenticazione e la crittografia:

- Utilizzare **SSL/TLS** per cifrare le comunicazioni tra client e server RMI, riducendo il rischio di intercettazioni. (es. Attacchi di man-in-the-middle)

2.3.3 Patch di sicurezza:

- Assicurati che tutte le vulnerabilità note di Java siano **correttamente patchate**.
- Molte vulnerabilità RMI derivano da bug nei JDK di Java, quindi mantenere la versione aggiornata di Java è fondamentale.

2.3.4 Monitoraggio e Logging:

- **Monitorare** e **Loggare** ogni accesso e interazione con il servizio RMI, per poter intervenire rapidamente in caso di compromissione.

2.3.5 Testing delle vulnerabilità:

- Utilizza strumenti di scansione delle vulnerabilità, come **Nmap**, per identificare eventuali configurazioni vulnerabili su RMI.
- Usare come in questo caso **Metasploit** per testare se la macchina è vulnerabile a exploit conosciuti per RMI.

3. Conclusioni

L’esercitazione ha dimostrato come una vulnerabilità, nel servizio **Java RMI,** possa essere sfruttata ottenendo il controllo remoto di una macchina Target (nel nostro caso Metasploitable).

Utilizzando Metasploit, siamo riusciti ad ottenere una sessione Meterpreter, potendo così raccogliere informazioni sensibili sulla macchina Target.

Per ridurre in futuro il rischio di un attacco tramite **Java RMI** sulla porta 1099 di Metasploitable, dovresti concentrarti sul disabilitare il servizio se non necessario, applicare le giuste misure di sicurezza (crittografia, autenticazione) e monitorare l'accesso al servizio stesso.