RELAZIONE
RITISH BHANTOOA

INDICE

1. INTRODUZIONE DELL'ESERCIZIO
2. SPIEGAZIONE DEI CONCETTI APPRESI DURANTE LA SETTIMANA E APPLICATI NELL'ESERCIZIO
3. CONFIGURAZIONE DELLE MACCHINE VIRTUALI
4. AVVIO DI METASPLOIT ED ESECUZIONE DELL'EXPLOIT
5. SESSIONE METERPRETER, CONFIGURAZIONE DI RETE E TABELLA DI ROUTING DELLA MACCHINA TARGET
6. CONCLUSIONI

1. INTRODUZIONE DELL'ESERCIZIO

L'esercizio di oggi richiede di sfruttare la vulnerabilità della **porta 1099** della nostra macchina virtuale Metasploitable. Ci viene chiesto di sfruttare questa vulnerabilità tramite l'**exploit java RMI** su **Metasploit**, per poi ottenere una sessione **Meterpreter** sulla macchina remota. Una volta che siamo riusciti ad ottenere questa sessione, bisognerà raccogliere informazioni su:

- **Configurazione di rete** della macchina target;
- Informazioni sulla **tabella di routing** della macchina target;

2. SPIEGAZIONE DEI CONCETTI APPRESI DURANTE LA SETTIMANA E APPLICATI NELL'ESERCIZIO

- **Metasploit**: è un framework di **penetration testing**, utilizzato per scoprire, sfruttare e testare le vulnerabilità nei sistemi informatici. È molto utilizzato da esperti informatici per simulare attacchi e migliorare le prestazioni di sistema. Metasploit presenta delle componenti principali, ovvero:
  -**Exploit**: è il codice che sfrutta le vulnerabilità per ottenere l'accesso;
  -**Payload**: è il codice eseguito sulla macchina target;
  -**Auxiliary**: strumenti di scansione e attacco senza sfruttare vulnerabilità;
  -**Post-exploitation**: strumenti per raccogliere informazioni per mantenere l'accesso;
  L'exploit, il payload e gli auxiliary, fanno parte dei suddetti **moduli**, ovvero componenti che eseguono delle funzioni specifiche, ad esempio, gli exploit come abbiamo detto in precedenza sfrutta le vulnerabilità per accedere, il payload definisce cosa fare una volta che un exploit ha avuto successo mentre l'auxiliarycontiene strumenti di scansione, brute force e molto altro. Metasploit sarà già presente nella nostra macchina kali e per avviarla basterà usare il comando **msfconsole** da terminale.
- **Meterpreter**: è un payload di metasploit progettato per il post-exploitation, una volta sfruttata la vulnerabilità, meterpreter

fornisce un ambiente interattivo per controllare il sistema senza scrivere file sul disco, riducendo così il rischio di essere rilevati da antivirus. Tra i vantaggi troviamo:
-La possibilità di eseguire comandi di sistema, scaricare file, registrare tastiere, catturare webcam e molto altro;
-Supporta moduli aggiuntivi per privilege escalation, pivoting e molto altro;
-Può usare connessioni **reverse TCP**, **HTTP**, **HTTPS** per eludere firewall e IDS.
La sessione di meterpreter ci verrà creata sul terminale della macchina virtuale una volta che l'exploit avrà avuto successo.

- **Exploit**: entriamo un po' più nel dettaglio per quanto riguarda gli exploit, questi come detto prima sono dei codici che sfruttano le vulnerabilità di una macchina per accedere, dal punto di vista pratico i passaggi sarebbero questi:
1. L'attaccante identifica tutte le vulnerabilità presenti nella macchina target;
2.Attraverso Metasploit, viene cercato l'exploit dato che, una volta avviato quest'ultimo ci verrà visualizzata la voce "**msf6>"**, la quale ci permetterà, attraverso una parola chiave, di cercare tutti gli exploit presenti tramite questa parola;
3.Successivamente, quando avremo deciso quale exploit usare, adopereremo il comando **use** con accanto il numero della posizione in cui si trova l'exploit;
4.Una volta fatto ciò attraverso il comando **options** andiamo a vedere le configurazioni richieste che sono necessarie per avviare l'exploit;
5.Una volta fatto ciò mandiamo in esecuzione l'exploit tramite il comando **run** o **exploit**.

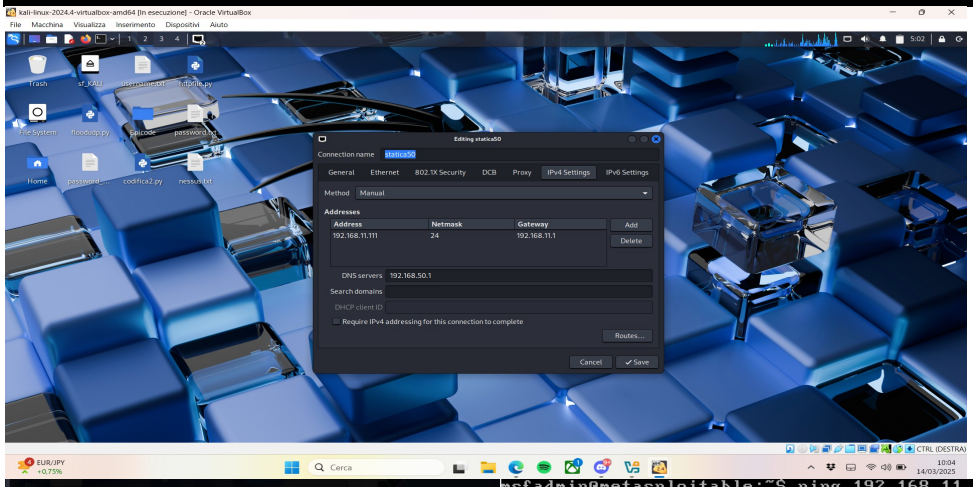
3. CONFIGUAZIONE DELLE MACCHINE VIRTUALI

Come prima cosa andiamo a configurare le nostre macchine virtuali, kali e metasploitable, sulla stessa rete affinchè possano comunicare e collegarsi tra di loro.
Alla macchina Metasploitable diamo l'indirizzo IP di 192.168.1.112 e a kali diamo l'IP DI 192.168.1.111

```
# 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 192.168.11.112
netmask 255.255.255.0
gateway 192.168.11.101
```



```
┌──(kali㉿kali)-[~]
└─$ ping 192.168.11.111
PING 192.168.11.111 (192.168.11.111) 56(84) bytes of data.
64 bytes from 192.168.11.111: icmp_seq=1 ttl=64 time=0.070 ms
64 bytes from 192.168.11.111: icmp_seq=2 ttl=64 time=0.103 ms
64 bytes from 192.168.11.111: icmp_seq=3 ttl=64 time=0.127 ms
64 bytes from 192.168.11.111: icmp_seq=4 ttl=64 time=0.167 ms
^C
— 192.168.11.111 ping statistics —
4 packets transmitted, 4 received, 0% packet loss, time 3081ms
rtt min/avg/max/mdev = 0.070/0.116/0.167/0.035 ms

┌──(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=11.0 ms
64 bytes from 192.168.11.112: icmp_seq=2 ttl=64 time=14.4 ms
64 bytes from 192.168.11.112: icmp_seq=3 ttl=64 time=3.39 ms
64 bytes from 192.168.11.112: icmp_seq=4 ttl=64 time=1.83 ms
64 bytes from 192.168.11.112: icmp_seq=5 ttl=64 time=18.5 ms
64 bytes from 192.168.11.112: icmp_seq=6 ttl=64 time=5.72 ms
^C
— 192.168.11.112 ping statistics —
6 packets transmitted, 6 received, 0% packet loss, time 5016ms
rtt min/avg/max/mdev = 1.833/9.146/18.493/6.014 ms

┌──(kali㉿kali)-[~]
└─$
```

```
msfadmin@metasploitable:~$ 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=0.060 ms
64 bytes from 192.168.11.112: icmp_seq=2 ttl=64 time=0.290 ms
64 bytes from 192.168.11.112: icmp_seq=3 ttl=64 time=0.175 ms
64 bytes from 192.168.11.112: icmp_seq=4 ttl=64 time=25.8 ms
64 bytes from 192.168.11.112: icmp_seq=5 ttl=64 time=0.161 ms

--- 192.168.11.112 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4084ms
rtt min/avg/max/mdev = 0.060/5.305/25.840/10.267 ms
msfadmin@metasploitable:~$ ping 192.168.11.111
PING 192.168.11.111 (192.168.11.111) 56(84) bytes of data.
64 bytes from 192.168.11.111: icmp_seq=1 ttl=64 time=1.48 ms
64 bytes from 192.168.11.111: icmp_seq=2 ttl=64 time=2.01 ms
64 bytes from 192.168.11.111: icmp_seq=3 ttl=64 time=1.55 ms
64 bytes from 192.168.11.111: icmp_seq=4 ttl=64 time=1.72 ms
64 bytes from 192.168.11.111: icmp_seq=5 ttl=64 time=2.14 ms

--- 192.168.11.111 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4008ms
rtt min/avg/max/mdev = 1.489/1.786/2.149/0.258 ms
msfadmin@metasploitable:~$
```

Successivamente verificiamo se le due macchine tra di loro pingano e se sono collegate alla rete.
Come vediamo entrambe sono connesse alla rete e riescono a pingare.

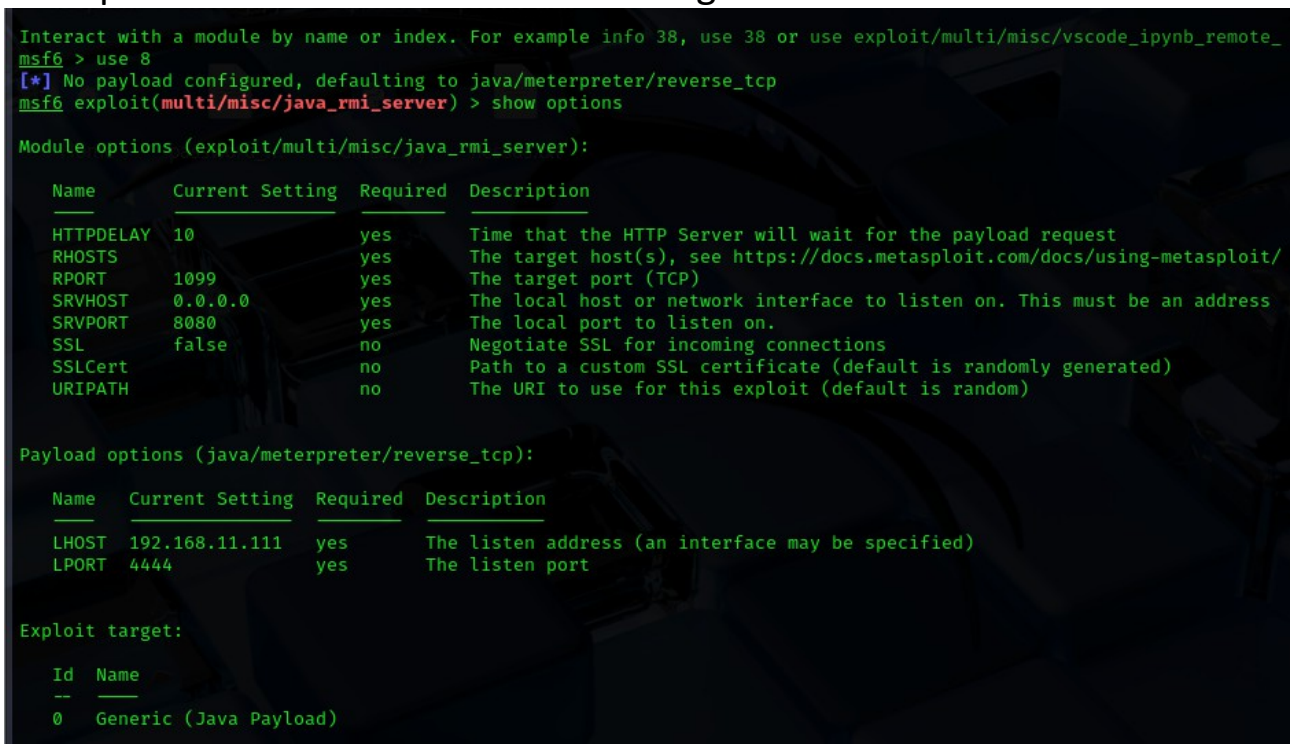
4. AVVIO DI METASPLOIT ED ESECUZIONE DELL'EXPLOIT

Una volta aver cambiato gli indirizzi IP delle macchine, avviamo Metasploit sul terminale di kali tramite il comando **msfconsole**

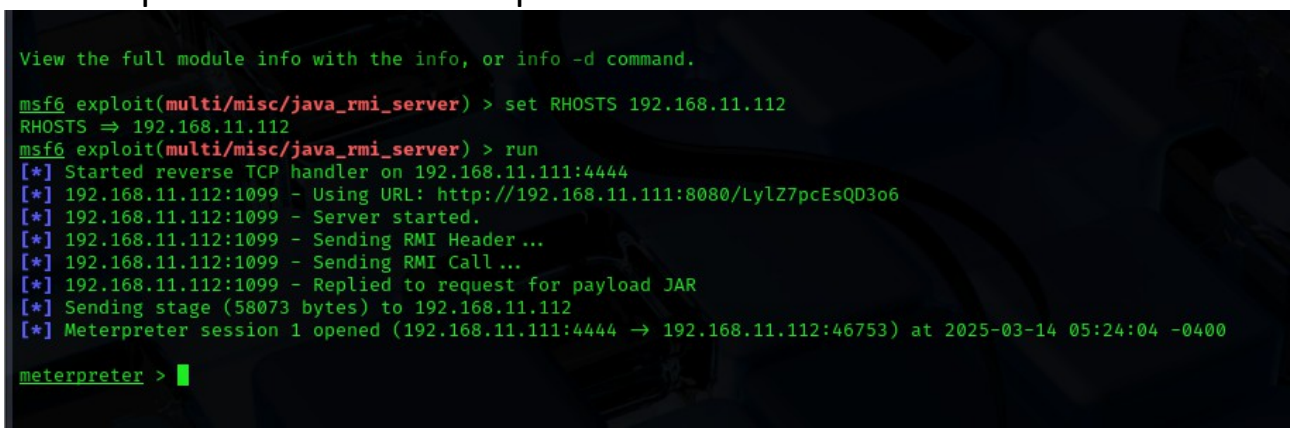
Come vediamo Metasploit inizia ad avviarsi, mostrando delle figure random ogni volta che si accede, notiamo che ci vengono segnati quanti exploit , auxiliary, post, payloads, encoders, nops e evasion vi sono presenti. Ritornando al punto in cui parlavamo accuratamente di metasploit dopo l'avvio, notiamo in baso la voce **msf6** che ci permetterà di ricercare exploit tramite delle parole chiave,

proseguendo con l'esercizio qui andiamo a ricercare l'exploit che ci serve per sfruttare le vulnerabilità di Metasploitable.

Come vediamo in questo caso la parola chiave che abbiamo utlizzato è **java RMI** la quale è preceduta dalla parola **search**, l'exploit che andremo ad usare sarà quello evidenziato ovvero, **multi/misc/java_rmi_server**.
Adesso tamite il comando **show options** andiamo a vedere i parametri che devono essere configurati.



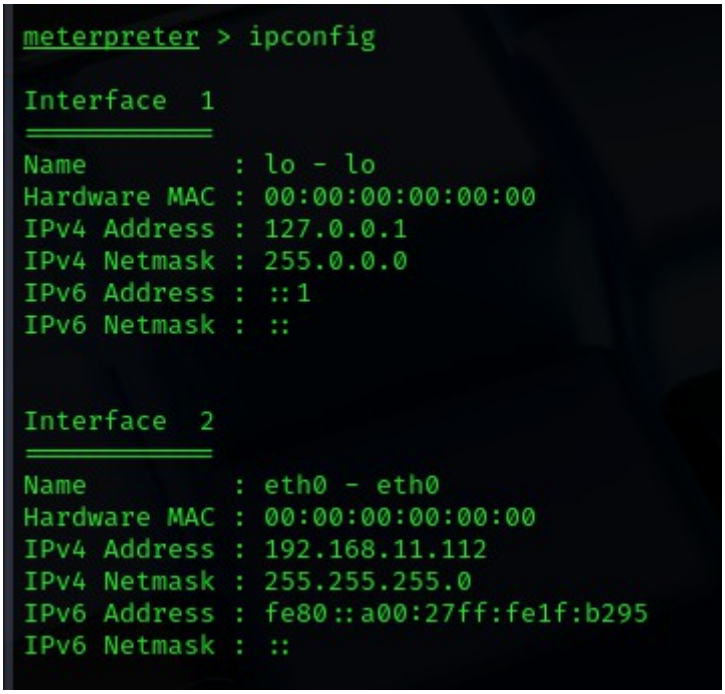
Una volta inserito questo comando ci verrà visualizzata questa schermata, con tutte le informazioni inerenti il local host e il server remoto, nel nostro caso l'unico parametro che dovremo configurare è **RHOSTS** ovvero l'IP della macchina target quindi, 192.168.11.112. Tra gli altri paraametri vediamo anche RPORT, ovvero la porta, di Metasploitable, che verrà sfruttata per l'esecuzione dell'exploit, più in baso invece vediamo LHOST e LPORT, rispettivamente l'IP e la porta della macchina dell'attaccante.



Come abbiamo detto quindi, configuriamo RHOSTS e, tramite il comando **run**, mandiamo in esecuzione l'exploit.
Una volta che l'exploit avrà avuto successo, più in baso vedremo che si è aperta una sessione **Meterpreter**, da qui andremo a raccogliere informazioni riguardanti la configurazione della rete e della tabella di routing della macchina Metasploitable.

5. SESSIONE METERPRETER, CONFIGURAZIONE DI RETE E TABELLA DI ROUTING

Una volta che ci troviamo nella sessione di Meterpreter, mandiamo in esecuzione il comando **ipconfig** che ci mostrerà l'indirizzo IP, l'indirizzo MAC, in poche parole ci mostrerà l'interfaccia di rete.



Adesso invece grazie al comando **route** visualizzeremo quella che è la tabella di routing con, l'indirizzo IP di destinazione o la rete, il gateway, la subnet mask, il metric, l'interfaccia, ci mosterà quindi come viene indirizzato il traffico tra le reti.

6. CONCLUSIONI

In conclusione possiamo dire che, l'esercizio è andato a buon fine, siamo riusciti a sfruttare la vulnerabilità del servizio Java RMI, usufruendo della porta 1099 della macchina Metasploitable usando Metasploit. Tra gli obiettivi raggiunti abbiamo:

- Sfruttamento della vulnerabilità RMI, utilizzando l'exploit “exploit/multi/misc/java_rmi_server, che ci ha permesso di eseguire il codice sulla macchina remota;
- Accesso alla macchina target tramite Meterpreter, che siamo riusciti ad ottenere una volta che l'exploit ha avuto successo;
- Siamo riusciti a raccogliere le informazioni che ci erano state rischieste ovvero la configurazione di rete e la tabella di routing;