



GRADO EN INGENIERÍA INFORMÁTICA

SEGURIDAD EN LOS SISTEMAS INFORMÁTICOS

DEPARTAMENTO DE INGENIERÍA INFORMÁTICA

Práctica 4: Escaneo y enumeración de activos

Autor:

Juan Boubeta Puig, Jesús
Lagares Galán y Pedro
José Navas Pérez

Fecha:

6 de noviembre de 2024

Índice

1. Objetivo	3
2. Configuración del laboratorio de <i>hacking</i>	3
2.1. <i>Host</i> objetivo: TryHackMe	3
2.1.1. ¿Qué es TryHackMe?	3
2.1.2. Registro en la plataforma	3
2.1.3. Primeros pasos	4
2.2. <i>Host</i> atacante	4
2.3. Otro <i>host</i> objetivo: Metasploitable (versión 2)	5
2.4. Configuración de Red NAT	6
3. Escaneo y enumeración	6
4. Ping sweepers	8
5. TCP-ping	8
5.1. Ejercicio 1	9
6. Estados de puertos	9
6.1. Nmap	9
7. Técnicas de escaneo	10
7.1. Ejercicio 2	11
7.2. Ejercicio 3	12
7.3. Furious	12
7.4. Ejercicio 4	13
8. Analizadores de vulnerabilidades	14
8.1. Ejercicio 5	15
9. Enumeración de varios protocolos con Netcat	17
9.1. Ejercicio 6	17
9.2. Ejercicio 7	19

Índice de figuras

1.	Botón <i>Start Machine</i>	4
2.	Botón para iniciar máquina atacante.	5
3.	Gráfico de la topología de la Red NAT.	7
4.	Rellenamos los parámetros en el apartado de Red.	7
5.	Ejemplo de uso de la herramienta Furious	13
6.	Formulario de registro	15
7.	Registro correcto	16
8.	Correo de confirmación	16
9.	Opciones de escaneo de Nessus	17
10.	Cambiamos los credenciales SSH para el escaneo	18

1. Objetivo

Hoy en día, la gran mayoría de sistemas informáticos está conectado a una red, ya sean redes locales, internas de organización o Internet. A través de estas redes, miles de datos y archivos son intercambiados. Por ello es importante monitorizar todo lo relacionado con las redes en los sistemas. A menudo estas redes se convierten en puertas de entrada que permiten a los atacantes hacerse con el control de datos, destruir o robar información, entre otras actividades.

El objetivo de esta práctica es conocer los recursos y herramientas que tenemos a nuestra disposición para llevar a cabo la segunda fase del *pentesting*: el escaneo y enumeración (*fingerprinting*).

2. Configuración del laboratorio de *hacking*

En esta práctica usaremos una máquina de **TryHackMe** [13] y una máquina virtual (**Metasploitable versión 2**) con diferentes vulnerabilidades como *hosts* objetivos. Como *hosts* atacantes, utilizaremos *AttackBox*, una máquina virtual en la nube proporcionada por **TryHackMe**, y una máquina virtual con Kali Linux.

2.1. *Host* objetivo: TryHackMe

De ahora en adelante, en las prácticas de la asignatura, para diversas actividades vamos a utilizar la plataforma de gamificación de ciberseguridad **TryHackMe**.

2.1.1. ¿Qué es TryHackMe?

TryHackMe es una plataforma web que nos permitirá levantar diversos entornos virtuales de prueba sin necesidad de desplegar varias máquinas virtuales de manera local, permitiéndonos sacar todo el rendimiento a nuestro equipo.

2.1.2. Registro en la plataforma

Para poder hacer uso de la plataforma, tendremos que registrarnos en la misma. Para ello, nos dirigimos a [13], pulsamos el botón de *Join for Free* e introducimos nuestros datos para registrarnos en la plataforma. Nos pedirá nuestro nivel de experiencia al entrar, seleccionaremos el que más se ajuste a nuestro perfil (en principio, *Beginner*). Más tarde, nos llegará un correo electrónico para que confirmemos nuestros datos. Una vez recibido este correo y validado el registro podremos acceder.

2.1.3. Primeros pasos

El primer paso será aprender a conectarnos a la red de **TryHackMe**. Para ello:

1. Seleccionar una máquina a la que queramos conectarnos. Podemos encontrar un ejemplo en la siguiente máquina utilizada dentro de esta práctica:
<https://tryhackme.com/room/furthernmap>
2. Pulsamos el botón *Start Machine*. Tras unos segundos, la máquina estará disponible para su uso (ver Figura 1). En la Sección 2.2 veremos cómo acceder a ella.

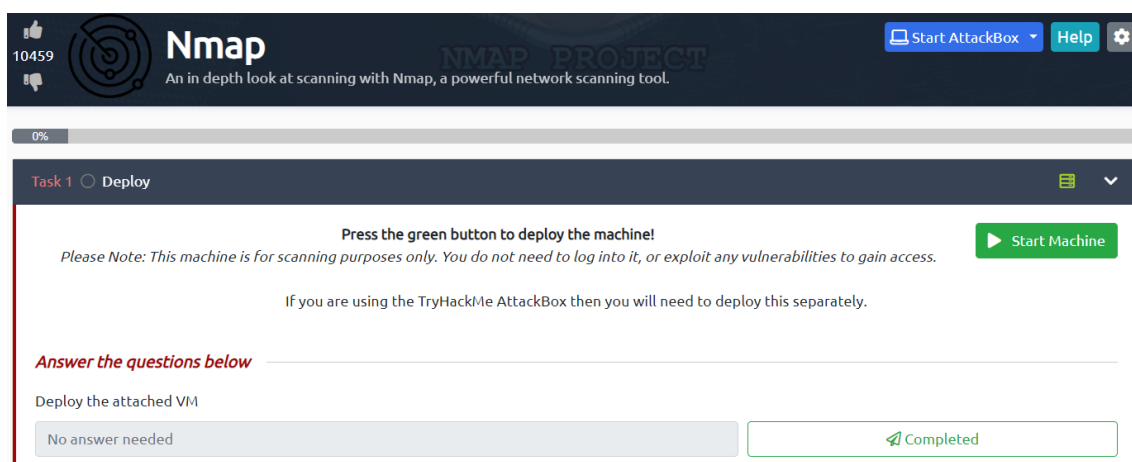


Figura 1: Botón *Start Machine*

Advertencia. Una vez que hemos realizado la conexión a la máquina, comenzará una cuenta regresiva de 1 hora (tiempo que TryHackMe nos permite utilizar la máquina). Si queremos prorrogar este tiempo, habrá que hacer clic en el botón *Add 1 hour* antes de que el tiempo acabe (podemos acceder a este botón en cuanto la máquina sea desplegada, por lo que podremos añadir más tiempo aunque no se haya acabado). Si queremos ampliar información, podemos consultar el tutorial oficial [12].

2.2. Host atacante

Como se ha indicado previamente, para la realización de las diferentes pruebas de *pentesting*, podemos utilizar como máquina base o *host* atacante, una máquina virtual en la nube proporcionada por **TryHackMe**, o la máquina virtual con Kali Linux [9] ya instalada anteriormente en VirtualBox. Para hacer uso de la máquina virtual proporcionada por **TryHackMe**, una vez inicializada la máquina, debemos de pulsar en el botón *Start AttackBox* (véase Figura 1).

Advertencia importante: Al no ser TryHackMe una herramienta totalmente gratuita, solo podremos iniciar una máquina atacante al día.

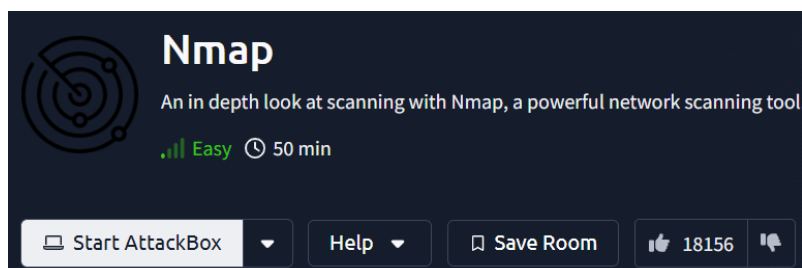


Figura 2: Botón para iniciar máquina atacante.

2.3. Otro *host* objetivo: Metasploitable (versión 2)

Para esta práctica, deberemos instalar y poner a punto otras máquinas virtuales a utilizar. La máquina virtual atacante ya la hemos utilizado en prácticas anteriores (una Kali Linux), y la máquina atacada será *Metasploitable version 2*.

Utilizamos *Metasploitable versión 2*, una distribución basada en Ubuntu, configurada de forma intencionada con fallos en su seguridad y puertos abiertos, perfecta para aprender a realizar ataques informáticos en un entorno seguro.

Procedemos a instalar la máquina virtual que será atacada. Para ello:

1. Descargar e instalar la última versión de *VM Virtual Box*:
<https://www.virtualbox.org/>
2. Descargar la máquina virtual *Metasploitable version 2*:
<https://sourceforge.net/projects/metasploitable/>
3. Extraer la máquina virtual haciendo clic derecho en el *.zip* descargado > Extraer aquí.
4. Ahora debemos crear nuestra nueva máquina virtual en *Virtual Box*. Para ello:
 - a) Abrir *VM Virtual Box* y seleccionar la opción *New*.
 - b) Se nos abrirá una nueva ventana, donde debemos introducir nombre (*Metasploitable*), y tipo de nuestra máquina virtual (Linux, Ubuntu 64bits).
 - c) Pulsamos en *Next*, y dejamos el valor de la memoria RAM *Random Access Memory* por defecto.

- d) Volvemos a pulsar *Next*, y seleccionamos la opción *Create a virtual hard disk file*.
- e) Seleccionamos el tipo de archivo de disco duro que queramos, en este caso, elegiremos la opción *Use an existing Disk Image* y seleccionamos el fichero **Metasploitable.vmdk** que hemos descargado previamente.
- f) *Next*, y definimos la ruta y capacidad para nuestra máquina virtual.
- g) Finalizamos con *Create*. Con esto habremos terminado el proceso de creación de nuestra máquina virtual.

2.4. Configuración de Red NAT

Para realizar ataques desde la máquina con Kali Linux hacia la máquina con Metasploitable, conectaremos ambas a una misma red local.

Vamos a utilizar un adaptador conectado a **Red NAT**, por tanto, debemos seleccionar el nombre de la red que necesitamos crear donde se verán las distintas máquinas virtuales (Kali, Metasploitable, etc.), que, además, tendrán acceso a Internet. Para ello, deberemos crear una **Red NAT**:

- Vamos a *Herramientas*, y a la pestaña **Redes NAT** y pulsamos en **Propiedades**.
- Creamos una nueva red con los siguientes parámetros:
 - 1. Nombre: **red-NAT-SSI**.
 - 2. Red CIDR **192.168.1.0/24**.
 - 3. E indicamos que soporte DHCP (véase Figura 4).
- De esta forma, podremos asociar a las distintas máquinas virtuales IP en esa red, consiguiendo que tengan conexión a Internet. Podemos ver un gráfico de la topología de la red en la Figura 3.

3. Escaneo y enumeración

Tras obtener el rango de direcciones IP (*Internet Protocol*) de nuestro objetivo durante la primera fase de reconocimiento de un *pentesting*, en esta segunda fase de **escaneo** [1]:

- Identificaremos los *hosts* que estén activos.
- Buscaremos los puertos abiertos en dichas máquinas.

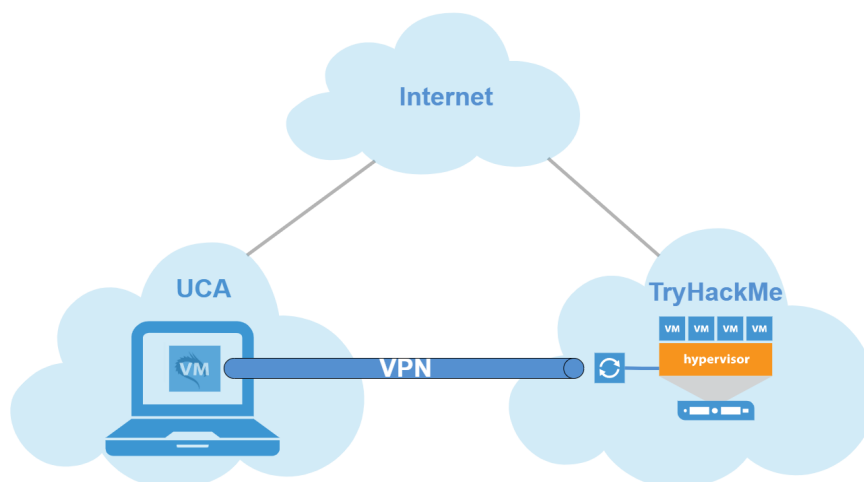


Figura 3: Gráfico de la topología de la Red NAT.

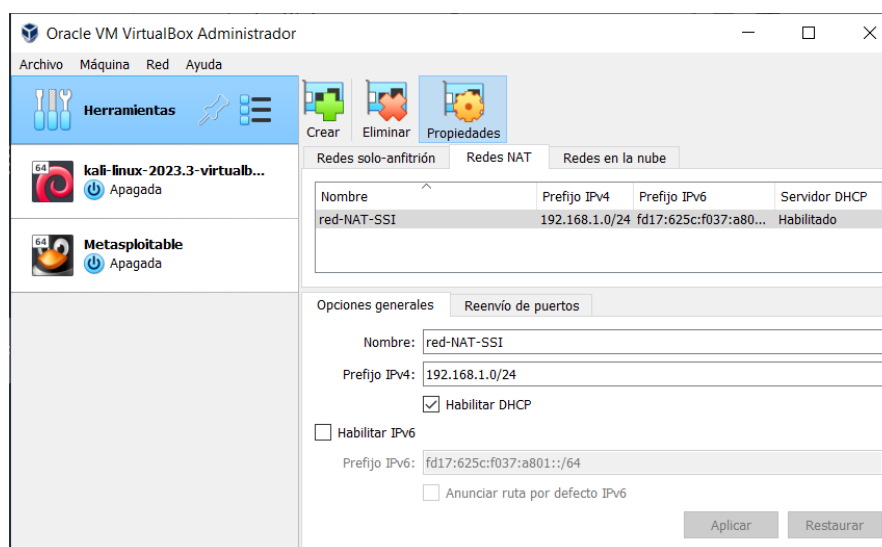


Figura 4: Rellenamos los parámetros en el apartado de Red.

- Gracias a los puertos abiertos, obtendremos información del sistema operativo, aplicaciones, servicios, etc.

Una vez identificados el sistema operativo y servicio, entre otros, podremos decidir si realizar la **enumeración** (escaneo más intenso) para tratar de obtener cuentas de usuario, procesos, etc. Si existen vulnerabilidades, estas podrán ser posteriormente explotadas.

Para conseguir esto, contamos con diferentes técnicas y herramientas como describimos a continuación.

4. Ping sweepers

Son herramientas de barrido de *ping*. Dichas herramientas nos permiten definir un rango de IPs donde enviar solicitudes de respuesta (*echo request*) utilizando el protocolo ICMP (*Internet Control Message Protocol*). Se marcarán como activos aquellos *hosts* que respondan a la solicitud.

Por ello, debemos tener en cuenta las siguientes consideraciones:

- Los *firewalls* suelen bloquear las solicitudes de *ping* realizadas desde el exterior.
- Los sistemas de prevención de intrusos o *Intrusion Detection System* (IDS) podrían detectar el escaneo y enviar órdenes al *firewall* para bloquear ciertas IPs.

5. TCP-ping

Permite comprobar si un *host* está activo utilizando el protocolo TCP (*Transmission Control Protocol*), en vez de ICMP. Hay casos en los que el *host* ha bloqueado el tráfico ICMP, por lo que no podemos hacerle ping de la forma tradicional. A través de TCP-ping, podemos comprobar su presencia en la red, ya que estaríamos enviando paquetes a un puerto conocido, o directamente tratando de establecer conexión TCP.

Cuando intentamos realizar conexión TCP con el *host*, este aceptará o rechazará la conexión, lo cual es información más que suficiente para saber si existe o no en la red. Por lo tanto, y a través de este método, podemos saber si el *host* está en la red, independientemente de que haya bloqueado la posibilidad de hacerle ping.

En resumen, tenemos dos métodos para conocer si el *host* está presente en la red:

1. Estableciendo conexión a los puertos más conocidos, como aquellos que usan HTTP, FTP, etc.
2. Obteniendo un error del tipo *conexión rechazada* al tratar de conectarnos.

Una herramienta conocida para realizar TCP-ping es *tcpping* [4]. Podemos instalarla en la máquina de Kali Linux (o cualquier otra distribución) ejecutando las siguientes órdenes en la terminal:

- `sudo apt-get install tcptraceroute`

- `cd /usr/bin`
- `sudo wget http://www.vdberg.org/~richard/tcpping`
- `sudo chmod 755 tcpping`

Podemos ahora probar que funciona haciendo TCP-ping a una página web cualquiera como, por ejemplo, Google:

```
tcpping www.google.es
```

5.1. Ejercicio 1

Ejecute la siguiente orden:

```
tcpping www.diariodecadiz.com
```

- Describa la salida que ha obtenido.
- ¿En qué se diferencia de un ping tradicional?

Ahora seleccione una dirección IP de su red local, puede ser un ordenador portátil, un teléfono móvil, o cualquier otro que tenga a mano.

- ¿Qué ocurre si hacemos *tcpping* a esa dirección? Justifique su respuesta.
- ¿Qué información útil le proporcionaría *tcpping* desde el punto de vista de la seguridad?

6. Estados de puertos

A la hora de realizar nuestro escaneo, será necesario conocer el estado de los puertos que nuestro objetivo tiene expuestos. Con esta información podremos saber qué puertos nos permiten ejecutar alguna acción, o cuáles están cerrados, con todo lo que ello significa. Para conocer esta información la herramienta más conocida es **Nmap**.

6.1. Nmap

Nmap [8] es la herramienta de escaneo de puertos más conocida. Puede instalarse a través de la orden `sudo apt-get install nmap`. Esta herramienta define los siguientes estados de puerto:

- **Abierto:** está disponible y escuchando. Por ejemplo: UDP/53 (DNS), TCP/80 (HTTP) y TCP/443 (HTTPS).

- **Cerrado:** está accesible pero no tiene un servicio/aplicación que responda a solicitudes de conexión.
- **Filtrado:** no está accesible. Un router con ACL (*Acces Control List*) implementada o *firewall* impide saber si el puerto está abierto o cerrado.
- **No-filtrado:** accesible, pero no se sabe si está abierto o cerrado.
- **Abierto/Filtrado:** el escáner no sabe si está abierto o filtrado.
- **Cerrado/Filtrado:** el escáner no sabe si está cerrado o filtrado.

7. Técnicas de escaneo

A continuación, se detallan las técnicas de escaneo [1]:

- **Escaneo SYN o *half-open*:**
 - Identifica puertos con servicios que usan TCP como protocolo de transporte.
 - En primer lugar, se envía una solicitud de sincronización (SYN) a la víctima.
 - Si se recibe como respuesta de la víctima:
 1. La sincronización junto con un acuse de recibo (SYN + ACK) → puerto abierto.
 2. Un *reset* (RST) → puerto cerrado.
 3. Sin respuesta → puerto filtrado.
- **Escaneo *full* o *connect-scan*:**
 - También utiliza el protocolo TCP.
 - En esta técnica, sí se finaliza la conexión con el envío del acuse de recibo final (ACK) a la máquina objetivo.
 - Requiere más tiempo para ejecutarse.
 - Podría quedar un registro de la conexión en los logs de eventos de la máquina objetivo.
- **Escaneo UDP (*User Datagram Protocol*):**
 - Utiliza el protocolo de transporte UDP.

- Envía un paquete UDP a los puertos de los *hosts* remotos y espera respuesta.
- Si la respuesta es:
 1. Un segmento UDP → puerto abierto.
 2. Un mensaje ICMP *port-unreacheable* → puerto cerrado.
 3. Otro tipo de error ICMP → puerto filtrado.
- **Escaneo ACK:**
 - Permite comprobar si existe un *firewall*.
 - Envía un segmento con la bandera ACK encendida al puerto destino de la víctima.
 - Si la respuesta es:
 1. RST → puerto no filtrado (*unfiltered*), accesible (el puerto puede estar abierto o cerrado).
 2. Sin respuesta o mensaje de error ICMP → puerto filtrado.

7.1. Ejercicio 2

Despliegue la máquina de la *room* <https://tryhackme.com/room/furthernmap> y realice los siguientes pasos (véase la Sección 2 para más información sobre cómo desplegar esta máquina):

- Acceda a la documentación de Nmap: <http://www.nmap.org>.
- Describa la sintaxis completa con las opciones del comando **nmap**.
- Describa y ejecute el comando **nmap** junto con las opciones necesarias para realizar:
 - Un escaneo en modo *half-scan* de la máquina objetivo.
 - Un escaneo tipo *connect* que permita detectar el sistema operativo de la máquina objetivo.
 - Un escaneo en modo *half-scan* que permita detectar el sistema operativo, la versión de los servicios y la obtención de *banners* de la máquina objetivo.

7.2. Ejercicio 3

Describe y explique los resultados obtenidos tras ejecutar el comando `nmap` junto con las opciones necesarias para realizar:

- Un escaneo en modo *half-scan* del servidor `scanme.nmap.org`.
- Un escaneo tipo `connect` que permita detectar el sistema operativo del servidor `scanme.nmap.org`.
- Un escaneo en modo *half-scan* que permita detectar el sistema operativo, la versión de los servicios y la obtención de *banners* del servidor `scanme.nmap.org`.

7.3. Furious

Si bien Nmap es una navaja suiza que nos permite ejecutar diversas tareas de reconocimiento en red, existen alternativas que realizan mejor ciertas tareas, este es el caso de **Furious**.

Furious [5] es una herramienta que permite realizar escaneos de tipo SYN a una velocidad mucho mayor de lo que lo hace Nmap. Esto lo hace ideal para tareas de enumeración de puertos.

Su sintaxis de uso es bastante sencilla, como se indica a continuación:

```
furious -s connect $IP
```

Un ejemplo de uso sería:

```
furious -s connect github.com
```

Aunque para examinar un simple *host* bastaría con:

```
furious $IP
```

Por ejemplo,

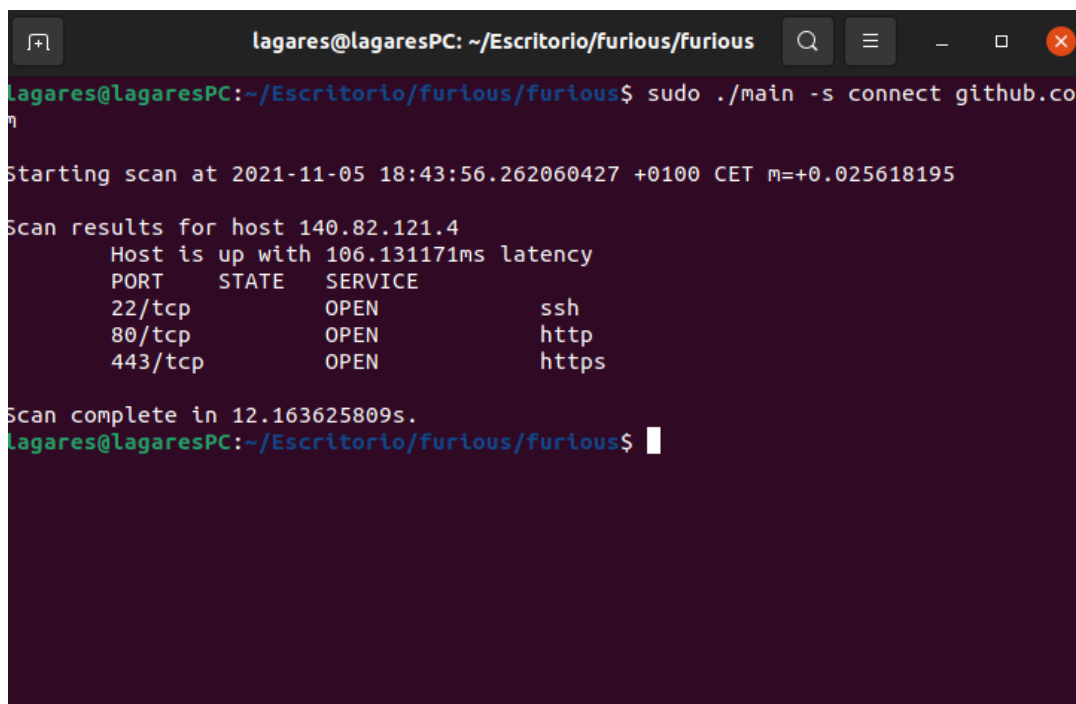
```
furious 192.168.1.4
```

Para la instalación de esta herramienta se nos requerirá la biblioteca **libcap** [2]:

1. Instalamos **libcap** y **Go**:

```
sudo apt-get install libpcap-dev golang-go
```
2. Ahora clonamos el repositorio oficial en nuestra máquina:

```
git clone https://github.com/liamg/furious
```



```
lagares@lagaresPC: ~/Escritorio/furious/furious
lagares@lagaresPC:~/Escritorio/furious/furious$ sudo ./main -s connect github.co
Starting scan at 2021-11-05 18:43:56.262060427 +0100 CET m=+0.025618195

Scan results for host 140.82.121.4
Host is up with 106.131171ms latency
PORT      STATE SERVICE
22/tcp    OPEN  ssh
80/tcp    OPEN  http
443/tcp   OPEN  https

Scan complete in 12.163625809s.
lagares@lagaresPC:~/Escritorio/furious/furious$
```

Figura 5: Ejemplo de uso de la herramienta Furious

3. Accedemos dentro del directorio con `cd furious` y contruimos el programa con la orden:
`go build main.go`
4. Ahora para ejecutar Furious tan solo debemos hacer uso del ejecutable que acabamos de construir. Por ejemplo:
`sudo ./main -s connect github.com` (véase Figura 5).

7.4. Ejercicio 4

Una vez instalada la herramienta Furious, realice un escáner de tipo *connect* a la web oficial de la universidad:

`uca.es`

Una vez realizado, responda a las siguientes preguntas:

- ¿Cuál es la dirección del *host*?
- ¿Tiene algún puerto abierto? En caso afirmativo, ¿cuál?
- ¿Qué comando ha utilizado para realizar un escaneo de este tipo?

- ¿Y si quisiéramos hacer un escaneo de tipo SYN, qué comando podríamos utilizar? No es necesario indicar la salida del comando, tan solo el comando a utilizar.

8. Analizadores de vulnerabilidades

Los analizadores de vulnerabilidades permiten ejecutar escaneos y enumeraciones sobre el objetivo. Las vulnerabilidades identificadas del objetivo pueden ser clasificadas según el riesgo [1]:

- **Alto:** el equipo objetivo tiene una o más vulnerabilidades críticas explotables fácilmente por un atacante. Se requiere una acción correctiva inmediata.
- **Medio:** tiene una o más vulnerabilidades severas que requieren una mayor complejidad para ser explotadas. Se requiere atención a corto plazo.
- **Bajo:** tiene una o más vulnerabilidades moderadas que podrían ofrecer información al atacante. No requiere atención urgente.

Algunos de los analizadores más conocidos son OpenVas [6], Nexpose [7] y Nessus [11].

Para esta práctica, vamos a realizar un análisis de vulnerabilidades con **Nessus**. Para ello:

1. Accedemos al siguiente enlace:
`https://es-la.tenable.com/tenable-for-education/nessus-essentials?edu=true`.
2. Rellenamos el formulario con nuestros datos, como vemos en la Figura 6:
 - a) Debemos registrarnos con un correo `@alum.uca.es` válido.
 - b) Debemos introducir Universidad de Cádiz en el nombre de la empresa.Si lo hemos realizado correctamente, se nos dirigirá a una pantalla de confirmación como la que vemos en la Figura 7.
3. Nos llegará un correo similar al que vemos en la Figura 8 con los datos de activación de la licencia; debemos guardar el código para más tarde.
4. Accedemos a la web de descarga oficial [10] y elegimos la descarga correspondiente a la distribución que vayamos a usar. Para instalar esta herramienta dentro de nuestra máquina Kali:

nessus
essentials

To register to use Nessus Essentials for education, please complete the following form. There is no cost for students and instructors.

Instructors: Share this page with your students to provide them with access to Nessus Essentials. Each student will need to complete the registration to get their own individual license.

Tenable provides Nessus Essentials for educators and students to use for educational purposes. Each individual can download their own Nessus Essentials license at no cost. Tenable does not support or endorse any program or course.

If you have any questions, please contact education@tenable.com.

Looking for additional help to get started? Check out our [Instructor/Student Guide](#).

Register for an Activation Code

Nombre Apellido

Correo electrónico laboral 1

Nombre de la empresa 2

☐ Marque para recibir actualizaciones de Tenable

Tenable solo procesará sus datos personales como se describe en su Política de privacidad.

Empezar

Figura 6: Formulario de registro

- a) Instalamos el paquete oficial de Nessus con el comando:

```
dpkg -i Nessus-X.Y.Z-debian6_amd64.deb
```

Advertencia: Siendo X, Y, Z la última versión del paquete disponible en el momento de su instalación.

- b) Durante la instalación de la herramienta se requerirá la licencia para activar la herramienta.

8.1. Ejercicio 5

Despliegue la máquina virtual metasploitable y realice los siguientes ejercicios (véase la Sección 2 para más información sobre cómo desplegar esta máquina):

1. Configure y lance un escaneo con Nessus sobre la máquina objetivo. Podemos configurar las distintas opciones de escaneo como podemos observar en la Figura 9 y en la Figura 10.
2. Una vez que se haya completado el escaneo revise las vulnerabilidades encontradas e indique:
 - ¿Cuáles son las vulnerabilidades de riesgo crítico que ha detectado el escáner?
 - El nombre de NETBIOS de la máquina objetivo.

Práctica 4: Escaneo y enumeración de activos

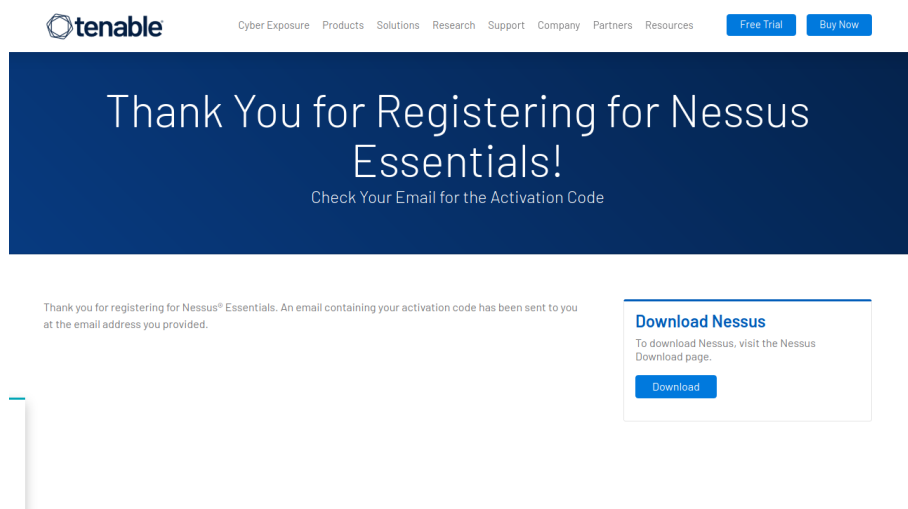


Figura 7: Registro correcto

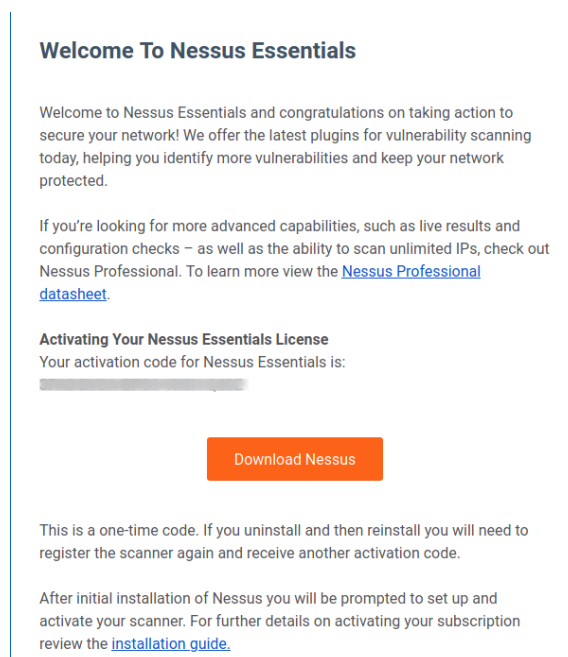


Figura 8: Correo de confirmación

Práctica 4: Escaneo y enumeración de activos

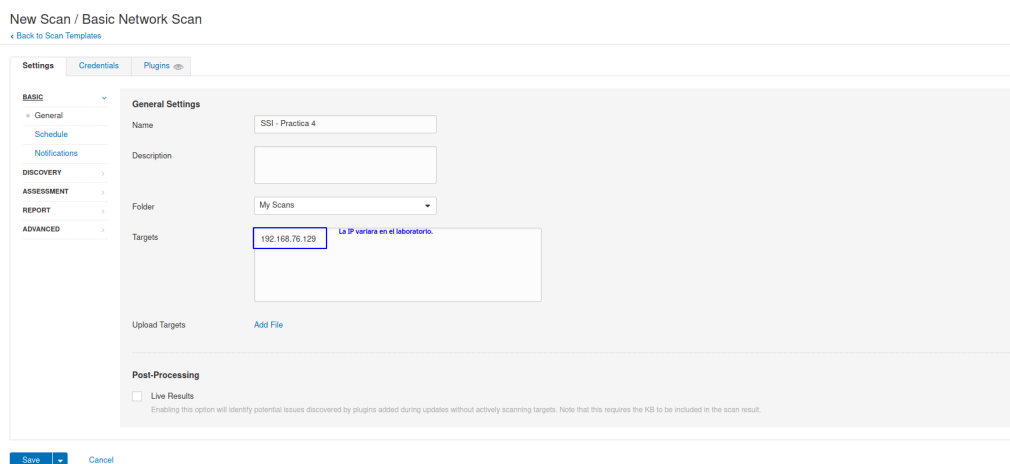


Figura 9: Opciones de escaneo de Nessus

- Las tres vulnerabilidades a la que más urgencia da Nessus de parchear. ¿Por qué Nessus da tanta importancia a las mismas?
- Elija una de esas tres vulnerabilidades y explique brevemente en qué consiste, si hay parche disponible, si hay prueba de concepto pública, etc.

9. Enumeración de varios protocolos con Netcat

Netcat [3] es una herramienta de código abierto que permite:

- Escanear puertos.
- Abrir puertos de escucha en un equipo y realizar conexiones remotas.
- Transferir ficheros.

9.1. Ejercicio 6

A partir de la información sobre enumeración de protocolos mediante la herramienta Netcat, realice las siguientes actividades. Para esta práctica desplegaremos la misma máquina objetivo que usamos en el ejercicio 2.

- Abra la terminal en Kali, ejecute el siguiente comando y describa la sintaxis de netcat:
 - `nc -h.`

Práctica 4: Escaneo y enumeración de activos

SSH

Authentication method: password

Username: ssi

Password (unsafe):

Elevate privileges with: sudo

sudo user: ssi

sudo password:

Location of sudo (directory): /usr/bin

Custom password prompt: password:

Global Credential Settings

known_hosts file: [Add File](#)

Preferred port: 22

Client version: OpenSSH_5.0

sssi2021

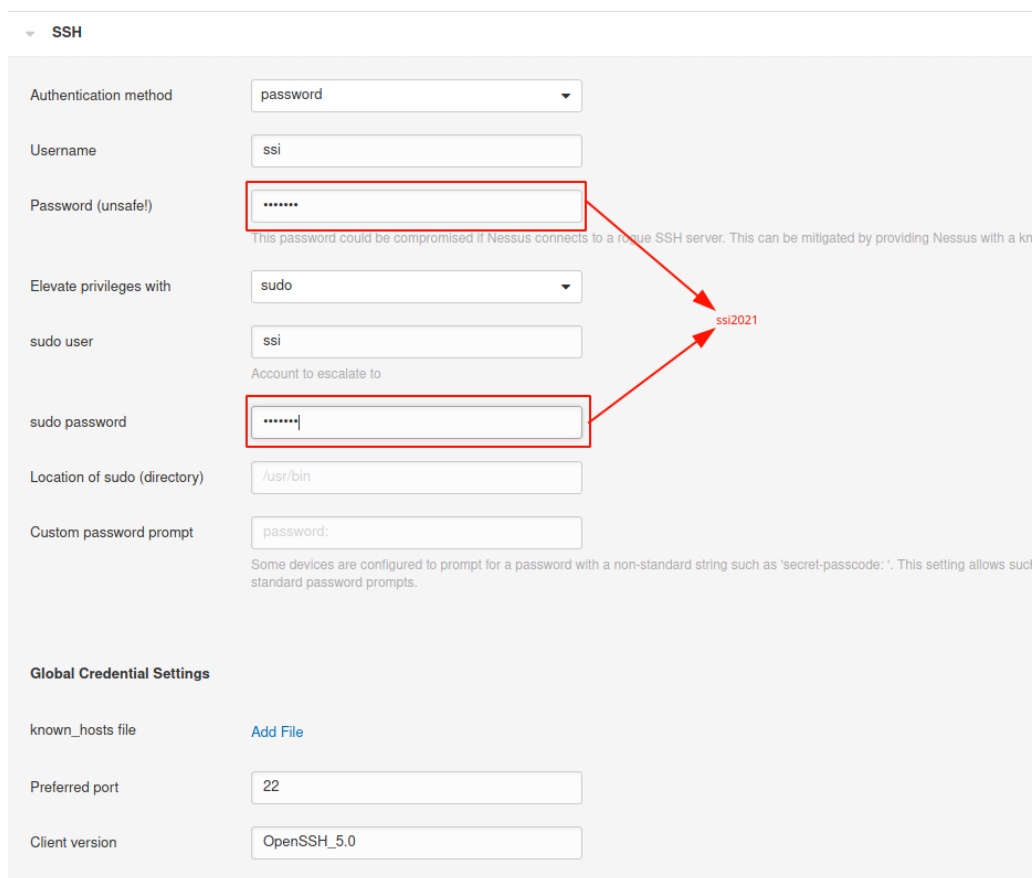


Figura 10: Cambiamos los credenciales SSH para el escaneo

- Utilizando el comando `nc`, realiza un escaneo de la máquina objetivo con las siguientes opciones:
 - Información detallada (*verbose*) de la conexión por consola.
 - Tiempo de espera por conexión de 3 segundos.
 - Modo de escaneo de puerto (zero I/O).
 - Puertos del 15 al 1000.
 - ¿Qué información se obtiene?

9.2. Ejercicio 7

Repita el Ejercicio 6 pero en esta ocasión escaneando todos los puertos (1-65535).

- ¿Qué opción permite escanear los puertos UDP?
- Use Netcat para conectarse desde Kali al puerto 80 de la máquina objetivo. Una vez conectado, ejecute el siguiente comando:

`GET / HTTP /`

¿Qué información se muestra en la consola de Kali?

- Busque por Internet más información sobre Netcat. ¿Qué otro tipo de información/operación podríamos obtener/realizar haciendo uso de este comando?

Referencias

- [1] K. Astudillo B: *Hacking Ético: Cómo Convertirse en Hacker Ético en 21 Días o Menos*. Ra-Ma, 3ª edición, 2018, ISBN 978-84-9964-767-8.
- [2] Free software directory: *Libcap Overview*. <https://directory.fsf.org/wiki/Libcap>, visitado el 03/11/2023.
- [3] EcuRed: *Web informativa Netcat*. <https://www.ecured.cu/Netcat>, visitado el 03/11/2023.
- [4] Eli Fulkerson: *tcping.exe - ping over a TCP connection*. <https://elifulkerson.com/projects/tcping.php>, visitado el 03/11/2023.
- [5] Liam Galvin: *Repositorio oficial de Furious*. <https://github.com/liamg/furious>, visitado el 03/11/2023.
- [6] Greenbone Networks: *OpenVAS web oficial*. <https://www.openvas.org/>, visitado el 03/11/2023.
- [7] Rapid7: *Nexpose web oficial*. <https://www.rapid7.com/products/nexpose/>, visitado el 03/11/2023.
- [8] Rapid7: *Nmap web oficial*. <https://nmap.org/>, visitado el 03/11/2023.
- [9] Offensive Security: *Kali Linux web oficial*. <https://www.kali.org/>, visitado el 03/11/2023.
- [10] Tenable: *Web de descarga oficial de la herramienta Nessus*. <https://www.tenable.com/downloads/nessus>, visitado el 03/11/2023.
- [11] Inc. Tenable: *Nessus web oficial*. <https://www.cs.cmu.edu/~dwendlan/personal/nessus.html>, visitado el 03/11/2023.
- [12] TryHackMe: *TryHackMe Tutorial Oficial*. <https://tryhackme.com/room/tutorial1>, visitado el 03/11/2023.
- [13] TryHackMe: *TryHackMe Web Oficial*. <https://tryhackme.com/>, visitado el 03/11/2023.