



Informe:

Creación de laboratorio de prácticas WeSecure Labs

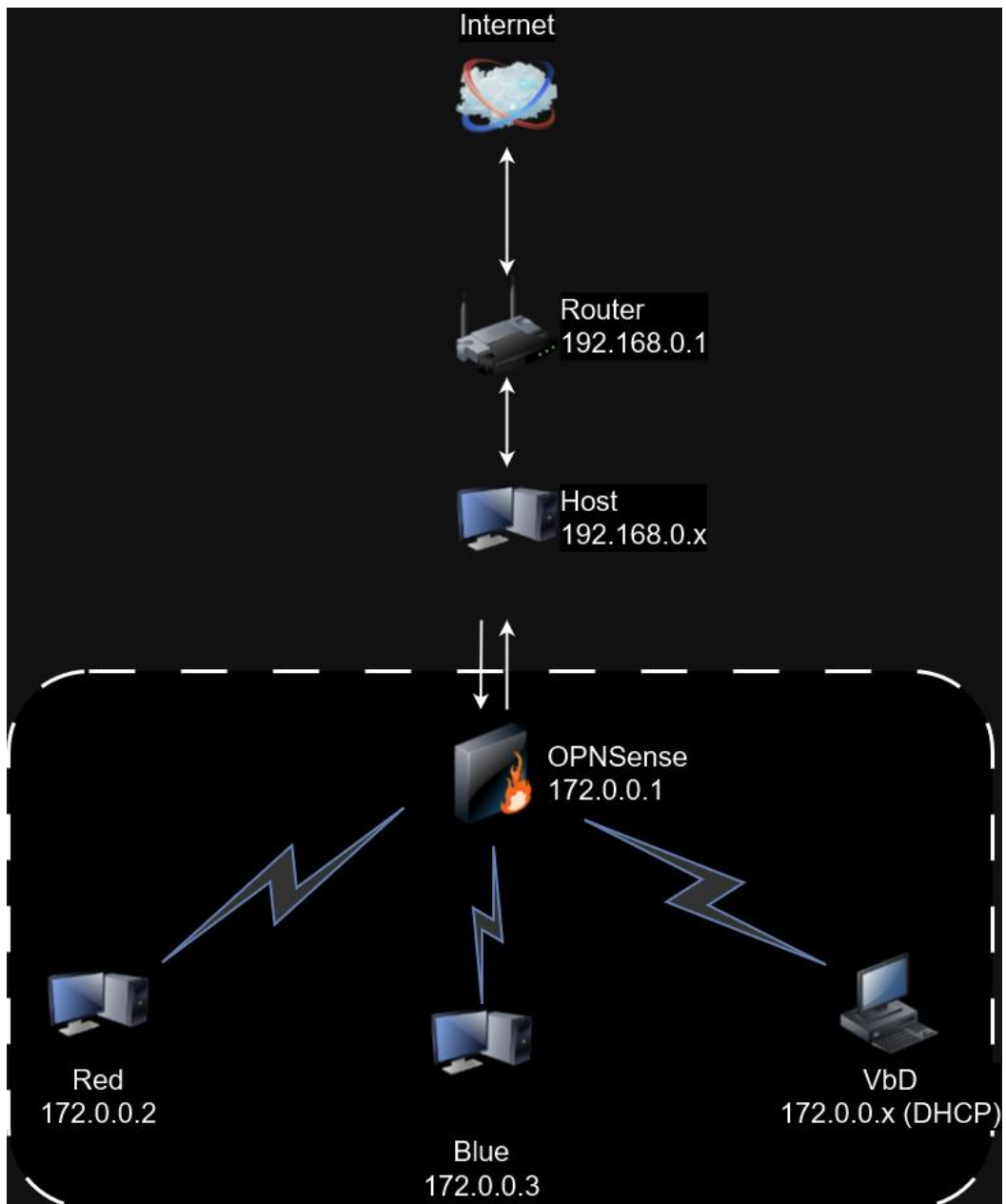
Contenido

1. Configuración de la red aislada	3
2. Configuración IP de las máquinas virtuales	4
2.1 Configuración IP del Firewall	4
2.2 Configuración IP máquina Red	6
2.3 Configuración IP máquina Blue.....	9
3. Prueba de conectividad entre las distintas máquinas de la red privada	10
4. Realización de traceroute desde la máquina Red a un host remoto.....	12
5. Proceso de actualización de las máquinas Red y Blue	13
6. Dificultades encontradas en el proceso	15
7. Aislamiento de la máquina VbD de Internet.....	16
8. Conclusiones	19

1. Configuración de la red aislada

Se requiere crear una red aislada que contenga, por un lado, el firewall y, por otro lado, las máquinas virtuales con las que se realizarán los diversos ejercicios propuestos en los siguientes retos. Así pues, es necesario crear una red privada en la que únicamente el firewall tiene acceso directo a Internet, y el resto de equipos virtuales deberán pasar a través de este para poder conectarse hacia “fuera”.

Así pues, la topología de la red consiste en el firewall que hará las veces de enrutador, ya que brindará conexión a internet e incluso otorgará direcciones IP a través de DHCP a algunos equipos; las máquinas virtuales que estarán en la misma red que el firewall y el equipo anfitrión que se encuentra en la red local física.



2. Configuración IP de las máquinas virtuales

2.1 Configuración IP del Firewall

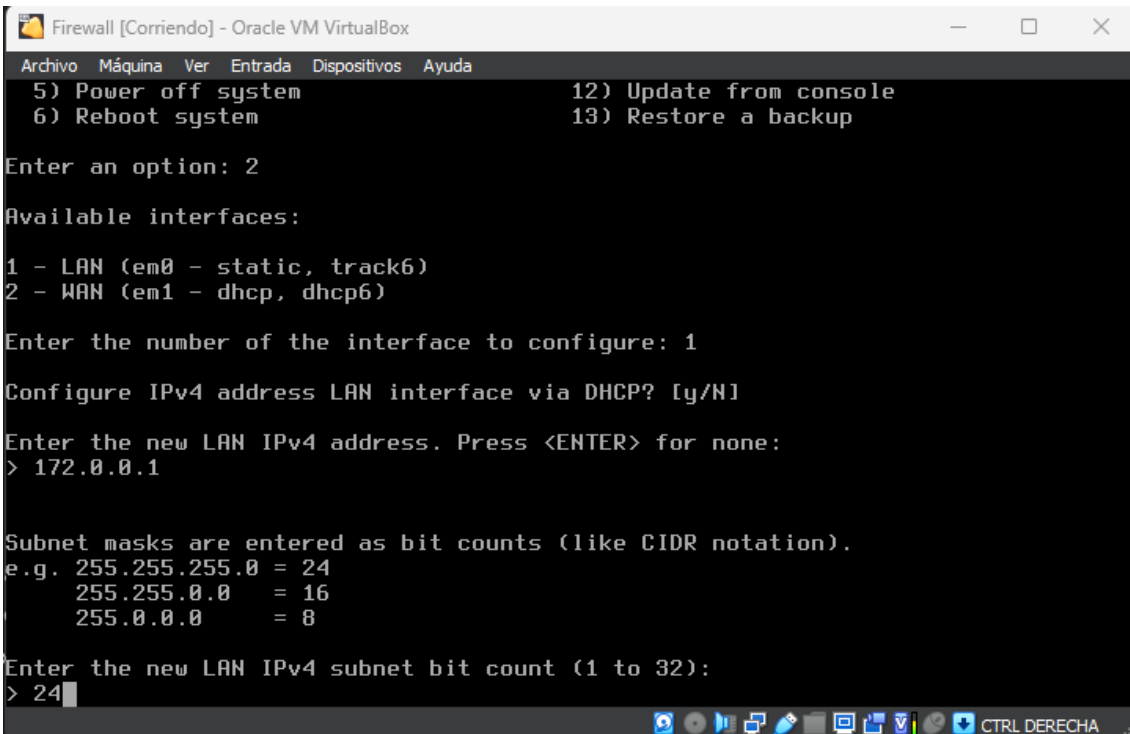
La máquina Firewall consta de dos interfaces de red, una interfaz LAN para la red interna que tiene la dirección IP 172.0.0.1, ya que actuará también como puerta de enlace para el acceso a Internet del resto de máquinas.

La segunda interfaz de red, etiquetada como WAN, está configurada como adaptador puente (*bridged connection*) y es la que le da acceso a internet a través de la red física a la que está conectado el *host*.

En el caso del Firewall, la interfaz que deberemos configurar manualmente será la LAN, ya que deberemos asignarle una IP fija y activar ciertas opciones para que el resto de la red pueda obtener direcciones IP por DHCP si procede.

En el menú de OPNSense, una vez nos hayamos autenticado como *root*, deberemos escoger la opción 2 y seguidamente la opción 1 para configurar la interfaz de red LAN.

En las capturas siguientes se muestra la configuración que se ha hecho tanto de la IP como de la máscara de subred (con notación CIDR), así como la configuración del servidor DHCP:



```
Firewall [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
5) Power off system
6) Reboot system
12) Update from console
13) Restore a backup

Enter an option: 2

Available interfaces:
1 - LAN (em0 - static, track6)
2 - WAN (em1 - dhcp, dhcp6)

Enter the number of the interface to configure: 1

Configure IPv4 address LAN interface via DHCP? [y/N]

Enter the new LAN IPv4 address. Press <ENTER> for none:
> 172.0.0.1

Subnet masks are entered as bit counts (like CIDR notation).
e.g. 255.255.255.0 = 24
     255.255.0.0   = 16
     255.0.0.0     = 8

Enter the new LAN IPv4 subnet bit count (1 to 32):
> 24
```

```
Firewall [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
> 172.0.0.1

Subnet masks are entered as bit counts (like CIDR notation).
e.g. 255.255.255.0 = 24
     255.255.0.0   = 16
     255.0.0.0     = 8

Enter the new LAN IPv4 subnet bit count (1 to 32):
> 24

For a WAN, enter the new LAN IPv4 upstream gateway address.
For a LAN, press <ENTER> for none:
>

Configure IPv6 address LAN interface via WAN tracking? [Y/n] n
Configure IPv6 address LAN interface via DHCP6? [y/N] n

Enter the new LAN IPv6 address. Press <ENTER> for none:
>

Do you want to enable the DHCP server on LAN? [y/N] y

Enter the start address of the IPv4 client address range: 172.0.0.11
Enter the end address of the IPv4 client address range: 172.0.0.254
```

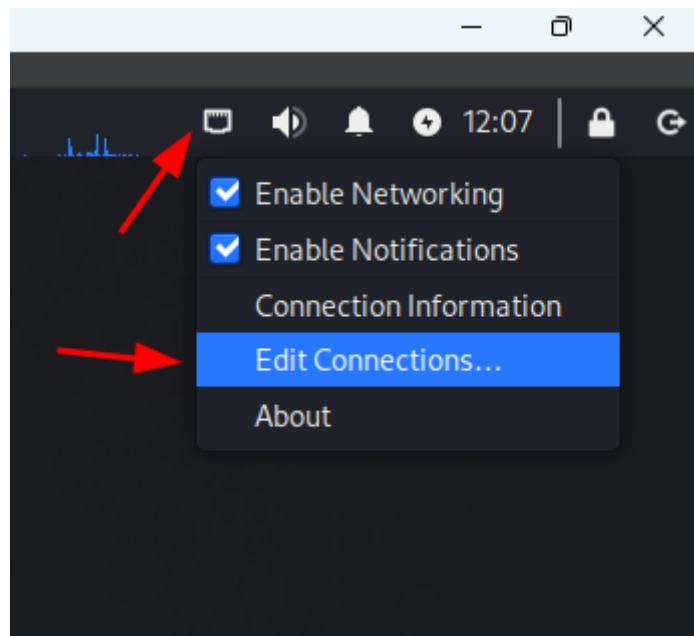
Como se aprecia en las capturas, se ha asignado la IP 172.0.0.1/24 a la interfaz LAN, y se ha activado el servidor DHCP con el rango de direcciones 172.0.0.11 a 172.0.0.254.

2.2 Configuración IP máquina Red

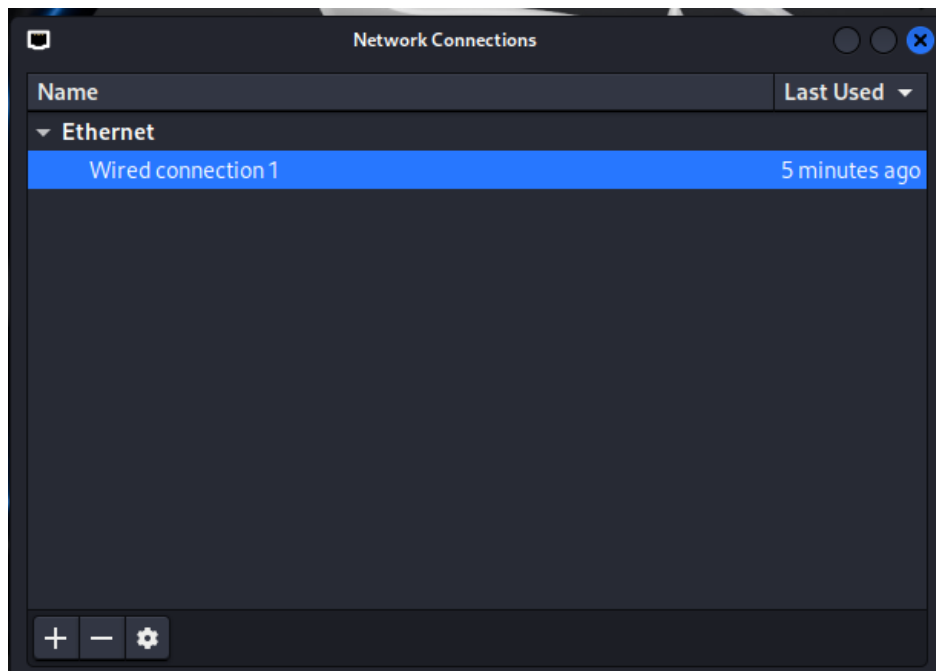
Para la máquina Red, configuraremos la IP 172.0.0.2. Esto puede hacerse bien desde la terminal o desde la interfaz gráfica. En este caso, asignamos la dirección IP, la máscara de subred que será 255.255.255.0 y, de manera opcional, un servidor DNS para ser utilizado por nuestro sistema para la resolución de nombres, que en este caso será el 1.1.1.1.

Los pasos para realizar la configuración mediante interfaz gráfica son los siguientes:

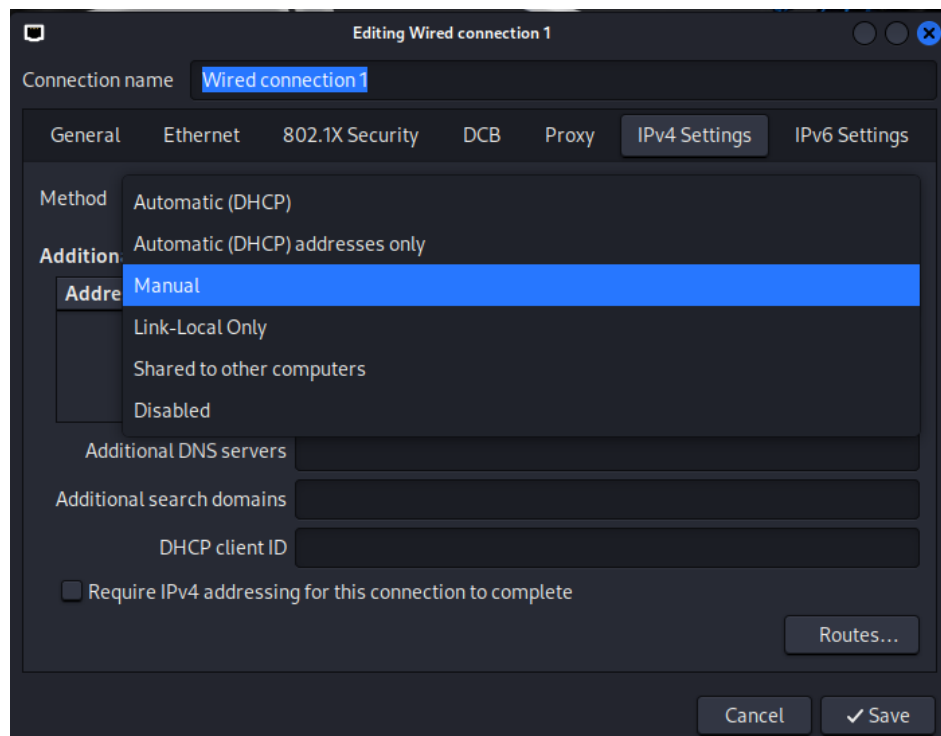
- En primer lugar hacemos click derecho sobre el icono del widget de red que se encuentra en la parte derecha de la barra de tareas, y dentro del menú click en “Edit Connections”:



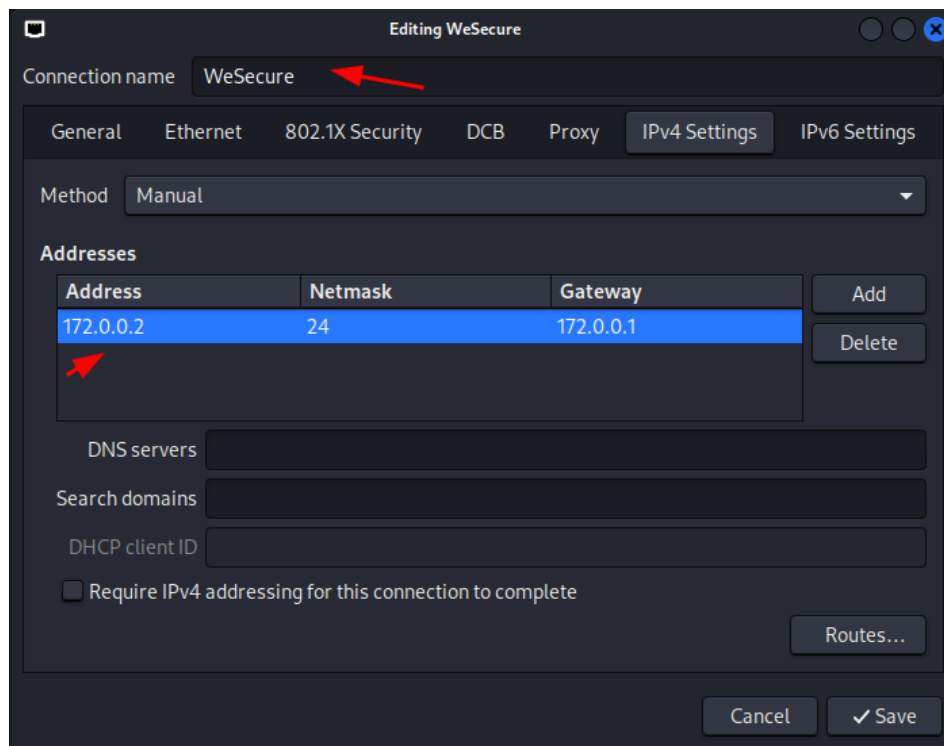
- A continuación, hacemos doble click en la conexión de red:



- Después, nos desplazamos hasta la pestaña IPv4 Settings y, en el desplegable "Method", seleccionamos la opción "Manual":



- Por último, haremos click en el botón “Add” e introduciremos los datos de dirección IP, máscara de subred y puerta de enlace predeterminada:



- Finalizaremos la configuración haciendo click en el botón “Save”, reiniciaremos la interfaz de red y comprobaremos que está bien configurada ejecutando `ifconfig` en una terminal:

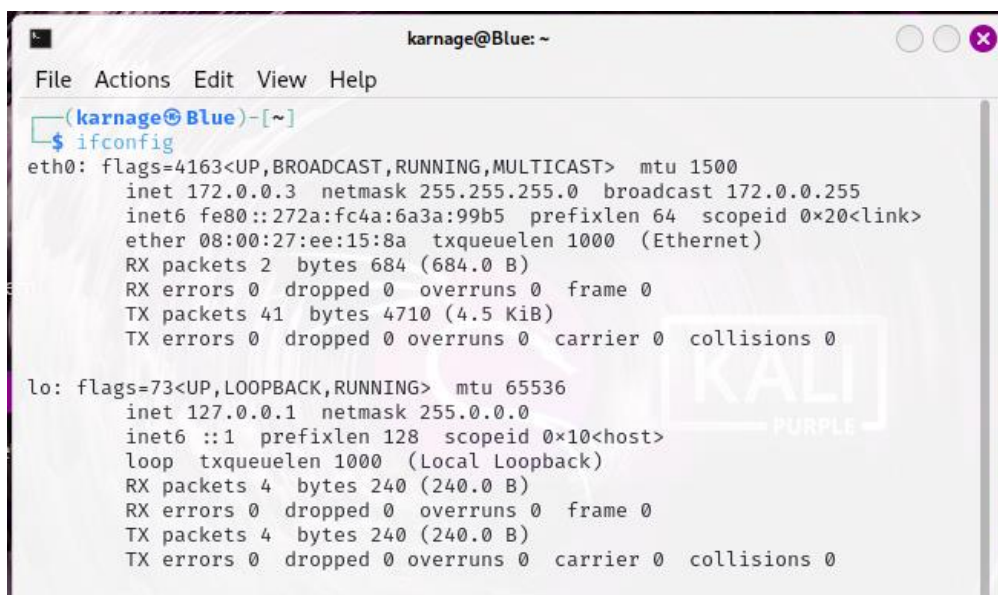
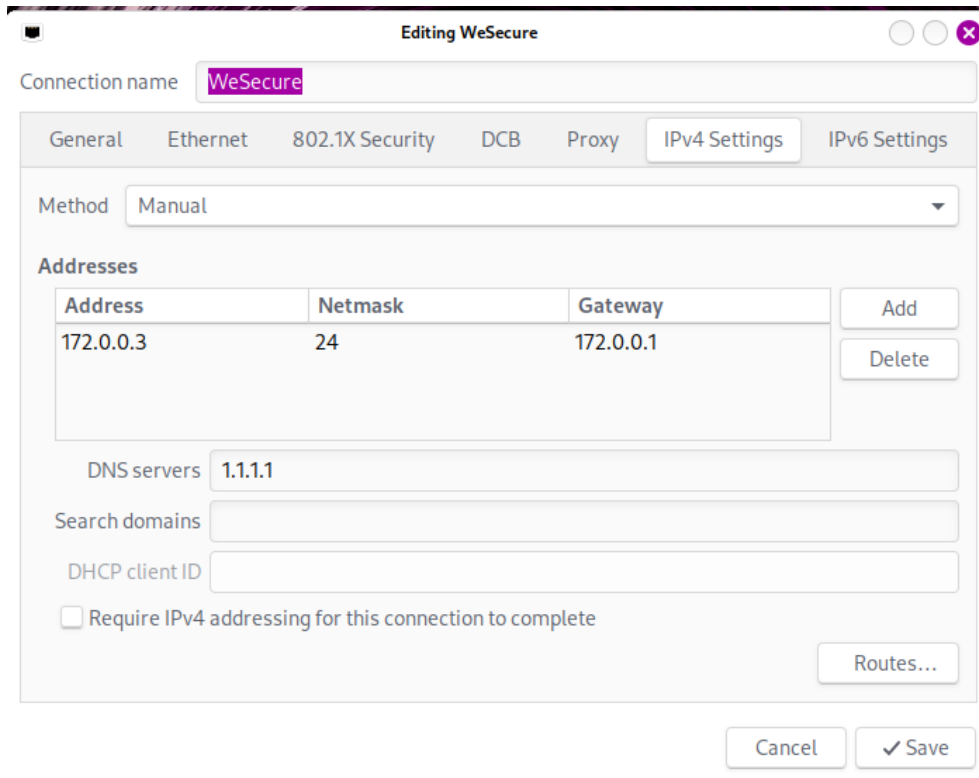
```
karnage@Red: ~
File Actions Edit View Help
(karnage@Red)-[~]
$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 172.0.0.2 netmask 255.255.255.0 broadcast 172.0.0.255
    inet6 fe80::c7f1:717a:6df6:facb prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:61:b0:eb txqueuelen 1000 (Ethernet)
    RX packets 17861 bytes 19968323 (19.0 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 8217 bytes 908225 (886.9 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 508 bytes 40032 (39.0 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 508 bytes 40032 (39.0 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```


2.3 Configuración IP máquina Blue

En el caso de la máquina Blue, la IP que le asignaremos será la 172.0.0.3. Los pasos para la configuración de la IP de esta máquina son idénticos a los de la máquina Red, así que no los repetiremos.

A continuación, se adjunta una captura de pantalla de la configuración IP de la máquina Blue, así como el resultado de ejecutar `ifconfig` para comprobar que dicha configuración se ha realizado correctamente:



3. Prueba de conectividad entre las distintas máquinas de la red privada

Para comprobar que existe conexión entre las distintas máquinas de la red privada, tanto entre Red y Blue como de éstas hacia la máquina VbD, utilizaremos la herramienta Ping.

La herramienta Ping permite el envío de paquetes ICMP a un host (local o remoto), y la recepción de una respuesta, tanto si existe conectividad como si no existe (aunque en ciertos casos no se recibirá respuesta, lo cual también querrá decir que no existe conexión entre los *hosts*).

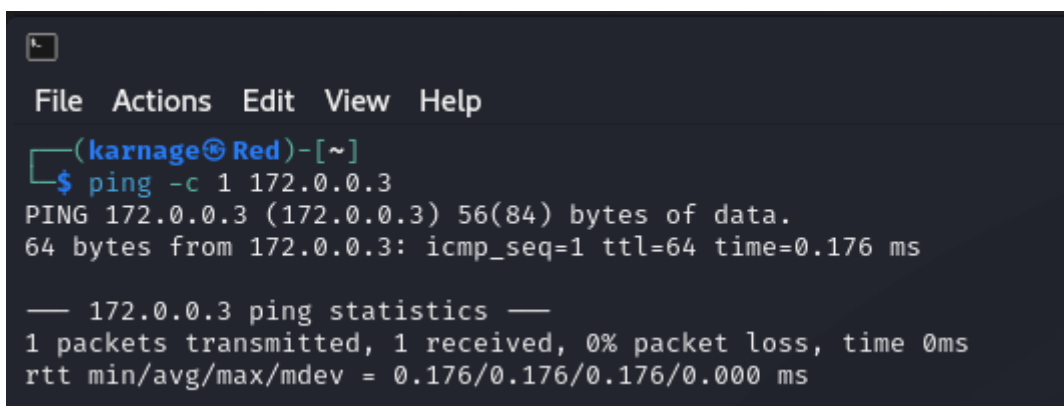
A continuación, se muestran capturas de pantalla de los distintos Ping realizados de la máquina Red a la Blue, de la Blue a la Red y, por último, desde la máquina Red a la máquina VbD:

El comando utilizado para la realización del ejercicio es el siguiente:

```
ping -c 1 172.0.0.x
```

El parámetro -c nos permite especificar la cantidad de paquetes que serán enviados, en este caso enviamos un solo paquete que será suficiente para comprobar que existe conectividad entre los *hosts*.

Para la realización de los pings desde las distintas máquinas, sustituiremos la “x” de la dirección IP por la que corresponda en cada caso, siendo 2 para Red, 3 para Blue y 11 (que recordemos era la primera IP del rango de asignación de DHCP) para VbD.



```
File Actions Edit View Help
(karnage@Red)-[~]
$ ping -c 1 172.0.0.3
PING 172.0.0.3 (172.0.0.3) 56(84) bytes of data.
64 bytes from 172.0.0.3: icmp_seq=1 ttl=64 time=0.176 ms

— 172.0.0.3 ping statistics —
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.176/0.176/0.176/0.000 ms
```

```
karnage@Blue: ~  
File Actions Edit View Help  
(karnage@Blue)-[~]  
$ ping -c 1 172.0.0.2  
PING 172.0.0.2 (172.0.0.2) 56(84) bytes of data.  
64 bytes from 172.0.0.2: icmp_seq=1 ttl=64 time=0.167 ms  
  
— 172.0.0.2 ping statistics —  
1 packets transmitted, 1 received, 0% packet loss, time 0ms  
rtt min/avg/max/mdev = 0.167/0.167/0.167/0.000 ms
```

```
(karnage@Red)-[~]  
$ ping -c 1 172.0.0.11  
PING 172.0.0.11 (172.0.0.11) 56(84) bytes of data.  
64 bytes from 172.0.0.11: icmp_seq=1 ttl=64 time=0.151 ms  
  
— 172.0.0.11 ping statistics —  
1 packets transmitted, 1 received, 0% packet loss, time 0ms  
rtt min/avg/max/mdev = 0.151/0.151/0.151/0.000 ms
```

Como vemos en las capturas, existe conectividad entre todas las máquinas, así que podemos afirmar que la configuración de la red privada ha sido exitosa, con lo que procedemos con los siguientes pasos de la configuración.

4. Realización de traceroute desde la máquina Red a un host remoto

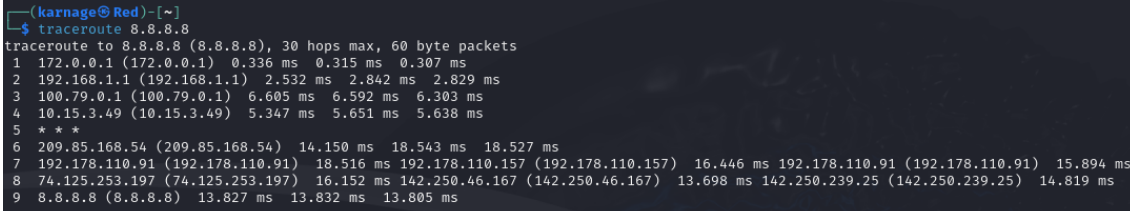
La herramienta traceroute permite visualizar los “saltos” que da un paquete desde que sale de nuestra máquina hasta que llega al host remoto hacia el que va dirigido.

En este caso, se ha realizado un traceroute hacia la dirección IP 8.8.8.8, que corresponde con uno de los servidores de DNS de Google.

El comando a ejecutar es muy sencillo, ya que sólo hay que introducir en un emulador de terminal “traceroute” y la IP destino hacia la que nos queremos dirigir, por ejemplo:

```
traceroute 8.8.8.8
```

El resultado de la ejecución del comando es el siguiente:



```
(karnage@Red) ~]$ traceroute 8.8.8.8
traceroute to 8.8.8.8 (8.8.8.8), 30 hops max, 60 byte packets
 1  172.0.0.1 (172.0.0.1)  0.336 ms  0.315 ms  0.307 ms
 2  192.168.1.1 (192.168.1.1)  2.532 ms  2.842 ms  2.829 ms
 3  100.79.0.1 (100.79.0.1)  6.605 ms  6.592 ms  6.303 ms
 4  10.15.3.49 (10.15.3.49)  5.347 ms  5.651 ms  5.638 ms
 5  * * *
 6  209.85.168.54 (209.85.168.54)  14.150 ms  18.543 ms  18.527 ms
 7  192.178.110.91 (192.178.110.91)  18.516 ms  192.178.110.157 (192.178.110.157)  16.446 ms  192.178.110.91 (192.178.110.91)  15.894 ms
 8  74.125.253.197 (74.125.253.197)  16.152 ms  142.250.46.167 (142.250.46.167)  13.698 ms  142.250.239.25 (142.250.239.25)  14.819 ms
 9  8.8.8.8 (8.8.8.8)  13.827 ms  13.832 ms  13.805 ms
```

En la captura se observan un total de nueve saltos entre nuestra máquina y la IP de destino.

- En el primer salto, se pasa por la IP 172.0.0.1 que corresponde a la máquina Firewall, que recordemos que actúa también como la puerta de enlace predeterminada de nuestra red privada.
- El segundo salto sale hacia la red local del *host* en el que están instaladas las máquinas virtuales, a la IP 192.168.1.1 que corresponde al router, que a su vez es la puerta de enlace predeterminada de dicha red.
- A continuación, se muestran varios saltos más que corresponderán a los servidores del ISP que nos proporciona internet y otros servidores que se encuentran en distintos puntos geográficos, hasta llegar en el salto 9 a la IP destino correspondiente al servidor DNS de Google.

5. Proceso de actualización de las máquinas Red y Blue

Dado que las máquinas Red y Blue tienen un sistema operativo Kali Linux, y éste se basa en la distribución Debian, disponen de `apt` como gestor de paquetes.

Así pues, la actualización de dichas máquinas se realizará mediante `apt`, ejecutando los comandos siguientes:

```
sudo apt update
```

```
sudo apt upgrade
```

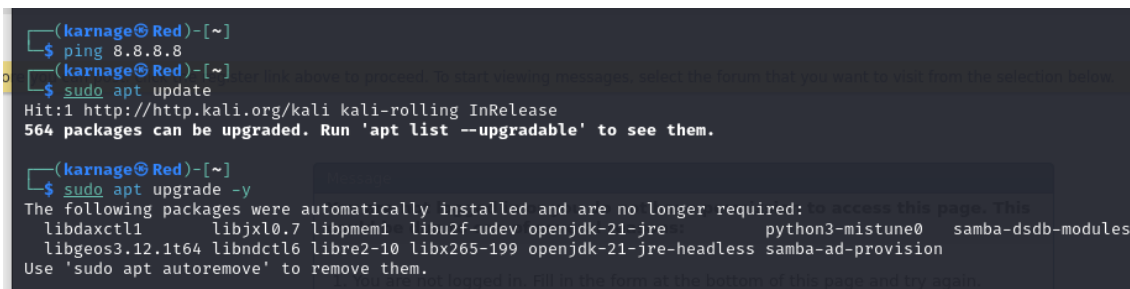
En caso de que queramos hacer una actualización completa del sistema operativo, podemos ejecutar también:

```
sudo apt dist-upgrade
```

Lo que actualizará la distribución a la última versión disponible del *kernel* o núcleo y aplicará todas las actualizaciones disponibles.

Existen algunos parámetros que pueden incluirse en el comando anterior, como por ejemplo el `-y` que hará que no se nos pregunte si estamos seguros de querer actualizar los paquetes. La sintaxis quedaría entonces así:

```
sudo apt upgrade -y
```



```
(karnage@Red)-[~]
$ ping 8.8.8.8
(karnage@Red)-[~]
$ sudo apt update
Hit:1 http://http.kali.org/kali kali-rolling InRelease
564 packages can be upgraded. Run 'apt list --upgradable' to see them.

(karnage@Red)-[~]
$ sudo apt upgrade -y
The following packages were automatically installed and are no longer required: to access this page. This
libdaxctl1 libjxl0.7 libpmem1 libu2f-udev openjdk-21-jre python3-mistune0 samba-dsdb-modules
libgeos3.12.1t64 libndctl6 libre2-10 libx265-199 openjdk-21-jre-headless samba-ad-provision
Use 'sudo apt autoremove' to remove them.
```



```
karnage@Blue: ~  
File Actions Edit View Help  
Get:376 http://kali.download/kali kali-rolling/main amd64 pipewire-pulse amd64 1.0.7-1 [22.8 kB]  
Get:379 http://kali.download/kali kali-rolling/main amd64 pipewire-bin amd64 1.0.7-1 [364 kB]  
Get:382 http://kali.download/kali kali-rolling/main amd64 libpipewire-0.3-common all 1.0.7-1 [77.9 kB]  
Get:385 http://http.kali.org/kali kali-rolling/main amd64 libradare2-dev amd64 5.9.2+dfsg-1 [238 kB]  
Get:389 http://http.kali.org/kali kali-rolling/main amd64 libsdl2-2.0-0 amd64 2.30.4+dfsg-1 [657 kB]  
Get:390 http://kali.download/kali kali-rolling/main amd64 libsigscan1 amd64 2.0240219-0kali1 [358 kB]  
Get:391 http://http.kali.org/kali kali-rolling/main amd64 libsynctex2 amd64 2.024.20240313.70630+ds-2 [62.2 kB]  
Get:411 http://http.kali.org/kali kali-rolling/non-free amd64 nmap-common all 7.94+git20230807.3be01efb1+dfsg-2+kali3 [4241 kB]  
Get:393 http://kali.download/kali kali-rolling/main amd64 wireplumber amd64 0.5.3-1 [102 kB]  
Get:394 http://kali.download/kali kali-rolling/main amd64 libwireplumber-0.5-0 amd64 0.5.3-1 [271 kB]  
Get:395 http://http.kali.org/kali kali-rolling/main amd64 libwxbase3.2-1t64 amd64 3.2.5+dfsg-1 [1015 kB]  
Get:400 http://kali.download/kali kali-rolling/main amd64 lightdm amd64 1.32.0-6 [163 kB]  
Get:402 http://kali.download/kali kali-rolling/main amd64 linux-image-6.8.11-amd64 amd64 6.8.11-1kali2 [99.7 MB]  
50% [276 libgl1-mesa-dri 2819 kB/8284 kB 34%] [409 mesa-vulkan-drivers 5618
```

```
karnage@Blue: ~  
File Actions Edit View Help  
Unpacking plaso (20240409-0kali1) over (20211229-0kali4) ...  
Setting up libvsapm1:amd64 (20240226-0kali1) ...  
Setting up firmware-linux-free (20240610-1) ...  
Setting up python3-zstd (1.5.5.1-1) ...  
Setting up python3-acstore (20240407-0kali1) ...  
Setting up python3-flor (1.1.3-2) ...  
Setting up libphdi1:amd64 (20240307-0kali1) ...  
Setting up libfsfat1:amd64 (20240220-0kali1) ...  
Setting up python3-pyfsfat (20240220-0kali1) ...  
Setting up python3-xattr (0.10.1-1) ...  
Setting up libcaes1:amd64 (20240114-0kali1) ...  
Setting up python3-pyvsapm (20240226-0kali1) ...  
Setting up libfcrypto1:amd64 (20240414-0kali1) ...  
Setting up python3-pyfcrypto (20240414-0kali1) ...  
Setting up python3-pyphdi (20240307-0kali1) ...  
Setting up python3-pycaes (20240114-0kali1) ...  
Setting up python3-dfvfs (20240505-0kali1) ...  
Setting up python3-plaso (20240409-0kali1) ...  
Setting up plaso (20240409-0kali1) ...  
Processing triggers for man-db (2.12.1-2) ...  
Processing triggers for kali-menu (2023.4.7) ...  
Processing triggers for libc-bin (2.38-13) ...  
Processing triggers for initramfs-tools (0.142) ...  
update-initramfs: Generating /boot/initrd.img-6.8.11-amd64  
  
(karnage@Blue)-[~]  
$
```

6. Dificultades encontradas en el proceso

Se han hallado diversas dificultades a la hora de poder configurar el laboratorio por completo, la mayoría de las cuales vienen dadas por el entorno de virtualización, que tiende a realizar acciones "por su cuenta".

- En primer lugar, se han encontrado dificultades en cuanto a la configuración del Firewall, ya que pese a realizar una configuración correcta las reglas no se aplicaban correctamente al tráfico de las distintas máquinas.
- Ha habido problemas también con la asignación de direcciones IP por DHCP de las máquinas, ya que VirtualBox asignaba unas direcciones antes que el Firewall, y en consecuencia la red no quedaba configurada correctamente.
- Por último, la configuración del Firewall puede resultar un tanto confusa si no se está acostumbrado a trabajar en entornos similares, así que ha habido un tiempo de adaptación, tanto para familiarizarse con la interfaz del firewall así como para encontrar la forma de realizar algunas acciones (en concreto, limitar la conectividad a Internet de la máquina VbD, de lo cual hablaremos en un apartado más adelante).

Como forma de mitigación de las dificultades encontradas, en primer lugar recomendaría instalar en primer lugar el Firewall, y configurar correctamente las interfaces de red del mismo. Esto nos permitirá que ya exista una red cuando instalemos el resto de máquinas, lo cual hará que los errores en la configuración de red de estas últimas sean menores o, incluso, inexistentes. Esto es especialmente importante a la hora de la asignación de direcciones IP por DHCP, que es uno de los problemas que se han encontrado durante la creación del laboratorio.

En segundo lugar, aconsejo intentar familiarizarse con el proceso de instalación de sistemas operativos Linux, ya que esto nos aportará las herramientas necesarias para poder lidiar con cualquier incidencia que pueda ocurrir durante la instalación de las máquinas virtuales.

Ya por último, también es una buena práctica el levantar las máquinas por orden, es decir, ya que la red la gestiona el Firewall, primero levantaremos este y después el resto de máquinas del laboratorio. Esto ayudará a mitigar posibles errores que puedan suceder por no encontrarse activo el servidor DHCP o el Firewall de red al iniciar las distintas máquinas virtuales.

7. Aislamiento de la máquina VbD de Internet

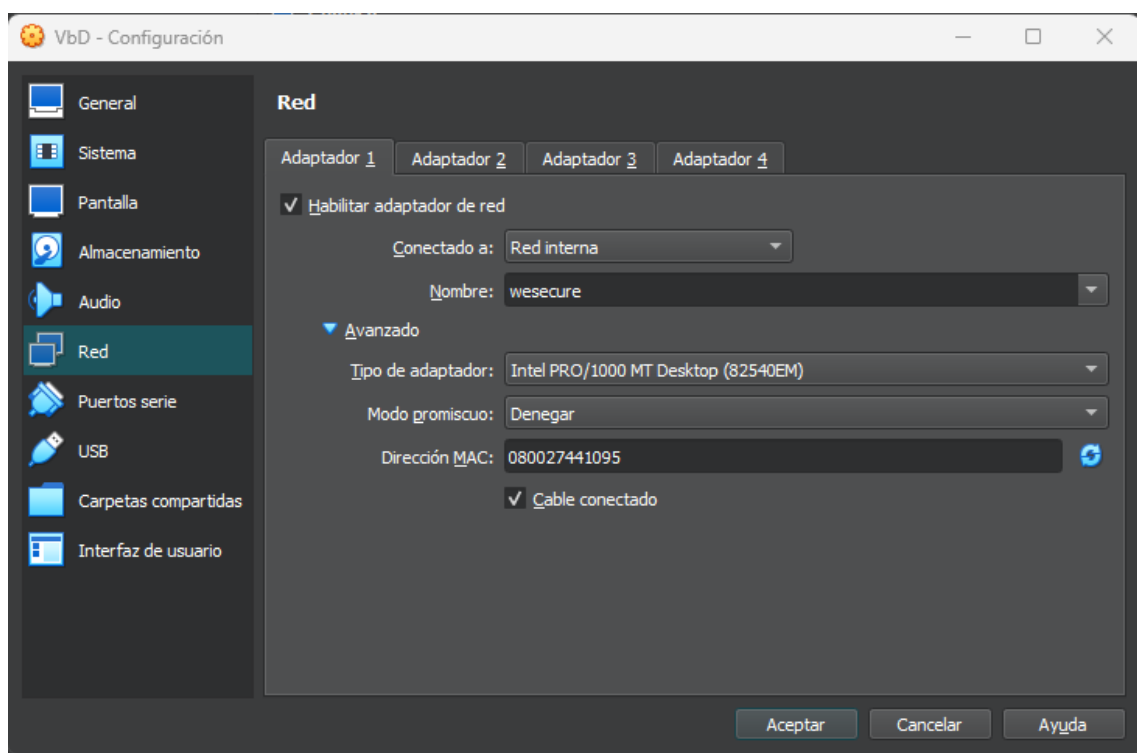
Una de las partes más importantes de la creación de este primer laboratorio es el aislamiento de la máquina Metasploitable 2, etiquetada como VbD, del acceso a Internet. Esto es primordial ya que no queremos exponer una máquina que tiene decenas o cientos de vulnerabilidades a la red, para no correr el riesgo de que un agente malicioso tenga acceso a nuestra red local a través de la misma.

El proceso para aislar la máquina VbD de Internet se basa en realizar algunas configuraciones en el Firewall que tengan como objetivo bloquear los intentos de conexión “hacia fuera” de la máquina.

Esto lo conseguiremos a través de la configuración de un “Alias” en nuestro Firewall, en el que especificaremos que la interfaz con una dirección MAC determinada (la de la máquina VbD) vea sus intentos de conexión bloqueados.

Los pasos a seguir para conseguir este efecto son los siguientes:

1. En primer lugar, deberemos conocer la dirección MAC de la máquina VbD, esto se puede hacer tanto desde VirtualBox en la sección de “Configuración -> Red”, o dentro de la propia máquina ejecutando `ifconfig`, lo cual nos mostrará información sobre todas las interfaces de red instaladas en la misma.



- Una vez anotada la dirección MAC (que, en este caso, es la 08:00:27:44:10:95), nos vamos a la interfaz gráfica del Firewall (recordemos, accediendo desde el navegador de la máquina Red a la dirección 172.0.0.1) y, en el menú lateral, accedemos a “Firewall -> Aliases” y creamos un nuevo alias, en el que deberemos especificar MAC Address en el desplegable “Type” y escribir la dirección MAC en la casilla “Content”. Además de esto, deberemos asignarle un nombre para poder seleccionarlo a la hora de crear una nueva regla en el Firewall:

Edit Alias

full help

Enabled ☒ Enable this alias

Name VbD Denegar Internet
The name must start with a letter or single underscore, be less than 32 characters and only consist of alphanumeric characters or underscores. Aliases can be nested using this name.

Type MAC address

Categories
For grouping purposes you may select multiple groups here to organize items.

Content 08:00:27:44:10:95
Clear All Copy

Statistics ☐ Maintain a set of counters for each table entry

Description
You may enter a description here for your reference (not parsed).

Cancel Save

- Ahora vamos a la sección “Rules” dentro de “Firewall” y, en el apartado “LAN” crearemos una nueva regla para bloquear o rechazar (el efecto final es el mismo, aunque internamente el funcionamiento de “Block” o “Reject” sea algo distinto) el tráfico hacia internet de nuestra máquina VbD, definida en el alias que hemos creado anteriormente:

Firewall: Rules: LAN

Edit Firewall rule

Action Block

Disabled ☐ Disable this rule

Quick ☒ Apply the action immediately on match.

Interface LAN

Direction in

TCP/IP Version IPv4

Protocol any

Source / Invert ☐ Use this option to invert the sense of the match.

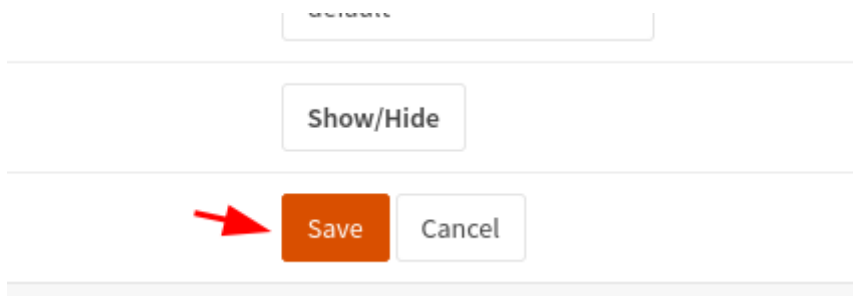
Source VbD_no_internet

Source Advanced

Destination / Invert ☐ Use this option to invert the sense of the match.

Destination any

4. Por último, guardaremos la configuración:



5. Una vez realizados los cambios, deberemos asegurarnos de que la regla que hemos creado quede por encima del resto de reglas presentes en el Firewall. Esto es así ya que el Firewall aplica las reglas en orden descendente, y si la dejamos en última posición, la máquina VbD tendrá acceso a internet ya que las reglas anteriores así lo permiten, y son aplicadas antes que el bloqueo:

Firewall: Rules: LAN

	Protocol	Source	Port	Destination	Port	Gateway	Schedule	Description
<input type="checkbox"/>								Automatically generated rules
<input type="checkbox"/>	IPv4 *	VSO_vso_internet	*	*	*	*	*	Bloqueo a internet de la máquina VSO
<input type="checkbox"/>	IPv4 *	LAN net	*	*	*	*	*	Default allow LAN to any rule
<input type="checkbox"/>	IPv6 *	LAN net	*	*	*	*	*	Default allow LAN IPv6 to any rule

6. Sólo resta comprobar que, en efecto, la máquina VbD no tiene acceso a internet, lo que haremos intentando hacer un Ping a un host remoto:

```
msfadmin@metasploitable:~$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
From 172.0.0.1 icmp_seq=1 Destination Host Unreachable
From 172.0.0.1 icmp_seq=2 Destination Host Unreachable
From 172.0.0.1 icmp_seq=3 Destination Host Unreachable

--- 8.8.8.8 ping statistics ---
3 packets transmitted, 0 received, +3 errors, 100% packet loss, time 2012ms
msfadmin@metasploitable:~$ _
```

8. Conclusiones

La creación de este laboratorio ha supuesto un gran aprendizaje, por distintos motivos:

- En primer lugar, me ha permitido familiarizarme con la interfaz y la configuración de un Firewall de red distinto de los que incluyen las distribuciones Linux o los sistemas operativos Windows, más completo y con muchas más opciones, lo que lo hace también mucho más potente.
- En segundo lugar, me ha permitido mejorar mis conocimientos sobre creación y administración de redes, y aplicar los conocimientos que ya tenía anteriormente a la creación de esta red con todas sus particularidades.
- Además, me ha permitido también ampliar mis conocimientos sobre VirtualBox y su funcionamiento, para poder gestionar mejor las posibles incidencias que pueda encontrar cuando realice nuevos laboratorios o agregue equipos al laboratorio existente.

Estoy muy contento de haber podido concluir satisfactoriamente todas las tareas propuestas en este reto, ya que en algunos casos ha sido un desafío poder averiguar el porqué ciertas cosas no funcionaban y cuál ha sido el procedimiento a seguir para corregir las distintas incidencias.