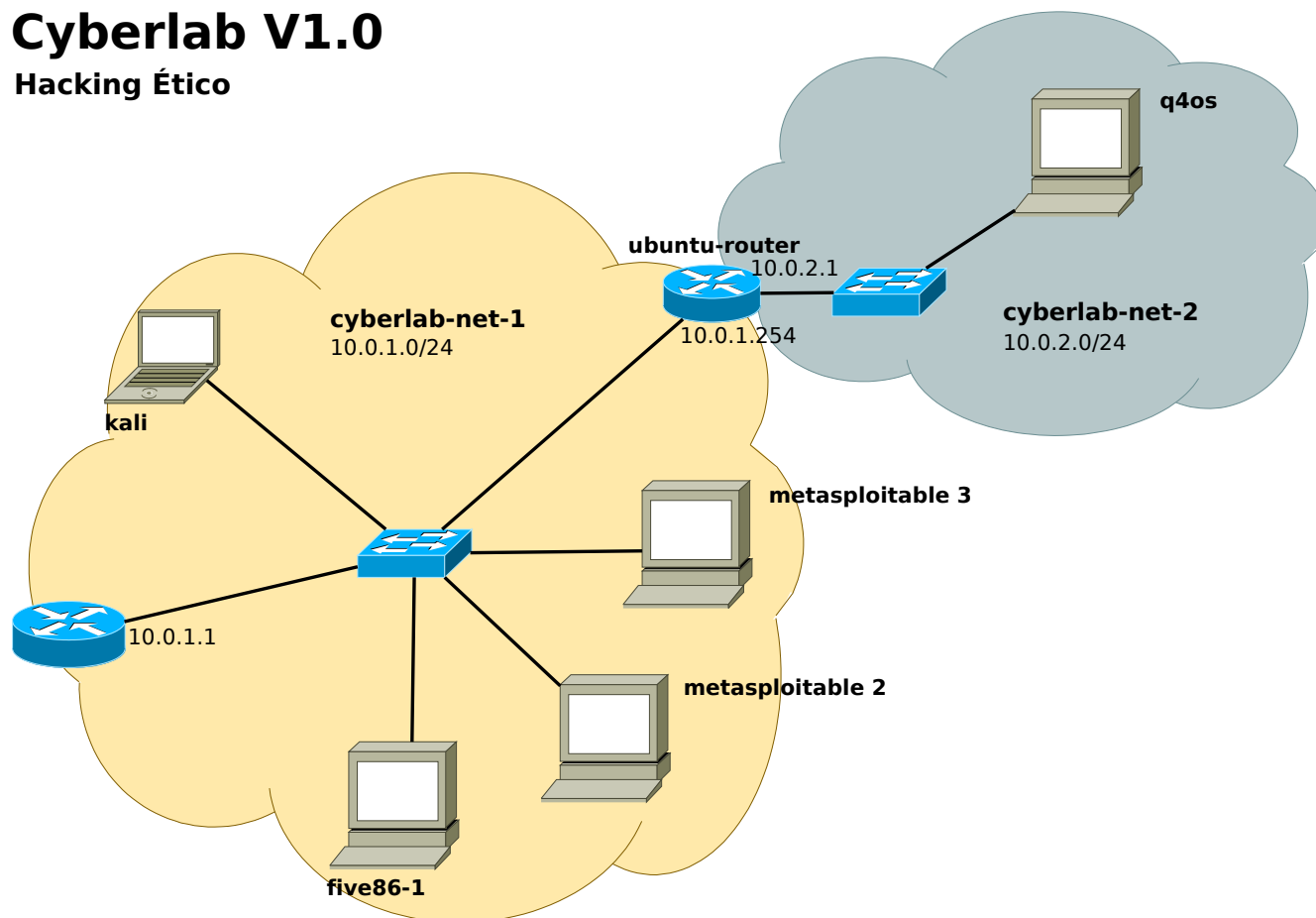


## LABORATORIO DE CIBERSEGURIDAD

### Cyberlab V1.0 Hacking Ético



José Luis Berenguel Gómez – IES Zaidín-Vergeles

## Sumario

Descripción de la tarea.....	3
Configuración de la red.....	4
Subred cyberlab-net-1.....	4
Subred cyberlab-net-2.....	5
Configuración de la máquina Ubuntu-router.....	6

## Descripción de la tarea

En esta unidad diseñaremos un laboratorio de ciberseguridad con diversas subredes y máquinas que usaremos en las explicaciones y además nos servirá también para las unidades siguientes. El esquema de red del laboratorio será el siguiente:

### Cyberlab V1.0 Hacking Ético

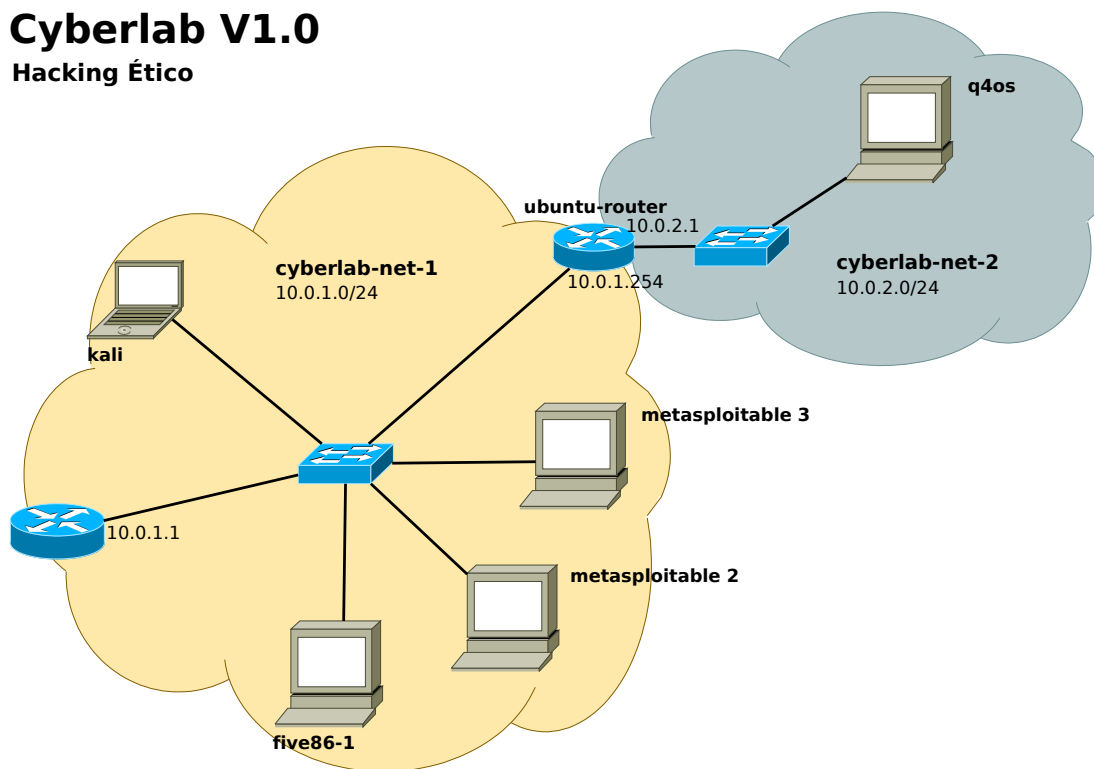


Figura 1: Red del laboratorio V1

Las máquinas para el laboratorio las podremos descargar de los siguientes enlaces:

- Metasploitable 2: <https://drive.google.com/file/d/1mgksDQ8XQEL4Xhv2p4HFpS9E2aoQLsA4/view?usp=sharing>
- Metasploitable 3: <https://drive.google.com/file/d/1suB0W26YB6NDUPzP3NsC8HRsBR3pVeSD/view?usp=sharing>
- Five86-1: <https://www.vulnhub.com/entry/five86-1,417/>
- q4os: <https://drive.google.com/file/d/1uHrDuEzr2eMW29jK6zOXTLZlNlPrqywH/view?usp=sharing>
- Ubuntu-router: [https://drive.google.com/file/d/1tgbsDZ2PJqEW\\_g\\_okP-LWJLERhjm3pYa/view?usp=sharing](https://drive.google.com/file/d/1tgbsDZ2PJqEW_g_okP-LWJLERhjm3pYa/view?usp=sharing)

## Configuración de la red

En los siguientes epígrafes se explica cómo configurar las subredes **cyberlab-net-1** y **cyberlab-net-2** y las máquinas que pertenecen a estas, así como la configuración de la máquina router que conecta ambas subredes.

### Subred cyberlab-net-1

La subred **cyberlab-net-1** es de tipo 'Red NAT'. Puedes crear una red NAT siguiendo la siguiente imagen:

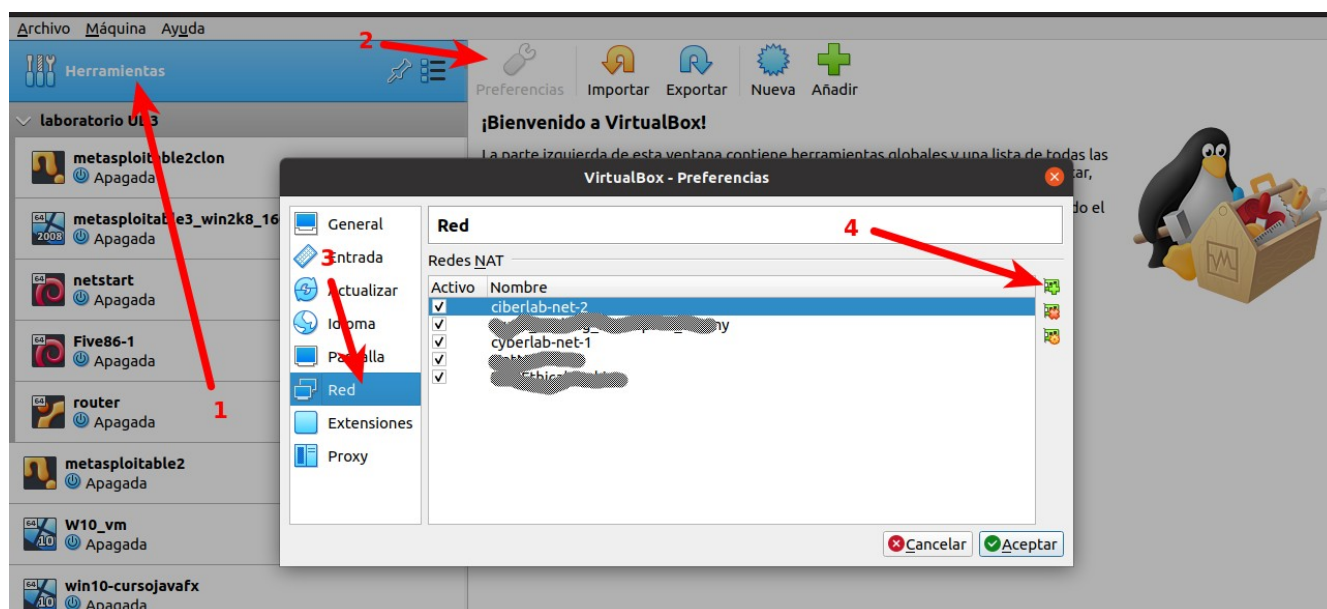


Figura 2: Creación de una red NAT

Es decir: (1) Herramientas → (2) Preferencias → (3) Red → (4) Añadir red  
Establece la siguiente configuración: IP 10.0.1.0/24 con el DHCP activado.



Para las máquinas que pertenecen a la red *cyberlab-net-1* deberán tener configurado su adaptador de red 1 para usar esta red NAT, tal y como muestra la imagen siguiente.

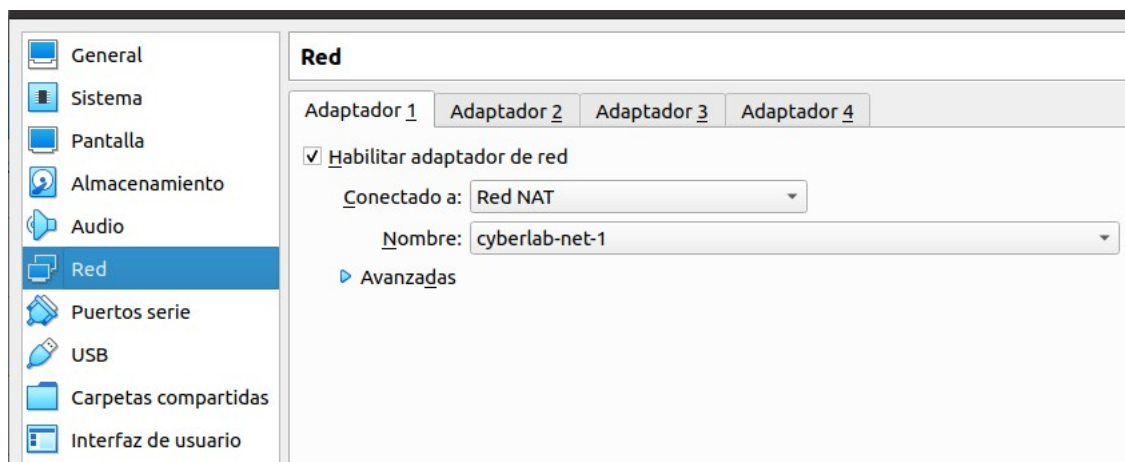


Figura 3: Configuración de red común

En el caso de la máquina router, que tiene 2 adaptadores de red, también se configurará el adaptador de red 1 para pertenecer a la red *cyberlab-net-1*.

### Subred cyberlab-net-2

Esta subred es de tipo '**Red Interna**'. En Virtualbox este tipo de red no tiene un servidor DHCP habilitado por defecto por lo que hay que configurarlo en línea de comandos o añadir direcciones IP estáticas a las máquinas que pertenezcan a esta subred, haremos lo segundo.

La máquina q4os pertenece a la red interna *ciberlab-net-2*, por lo que su adaptador de red 1 deberá tener la siguiente configuración:

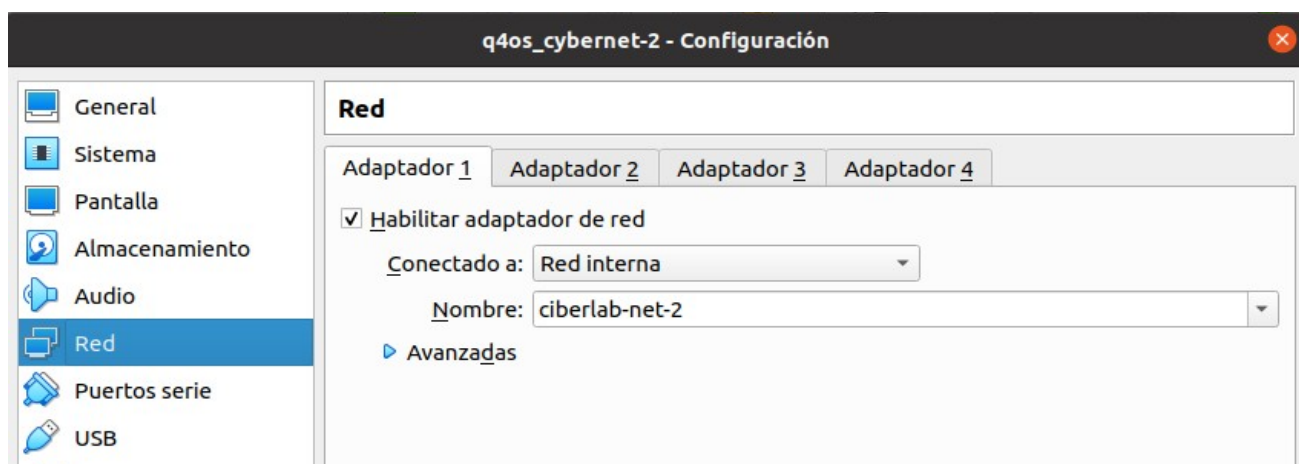
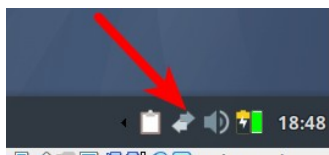


Figura 4: Configuración de red de la máquina q4os

Inicia la máquina (*cyberlab/cyberlab*), y cambia su configuración de red de la siguiente manera:



A continuación edita la conexión con estos datos:

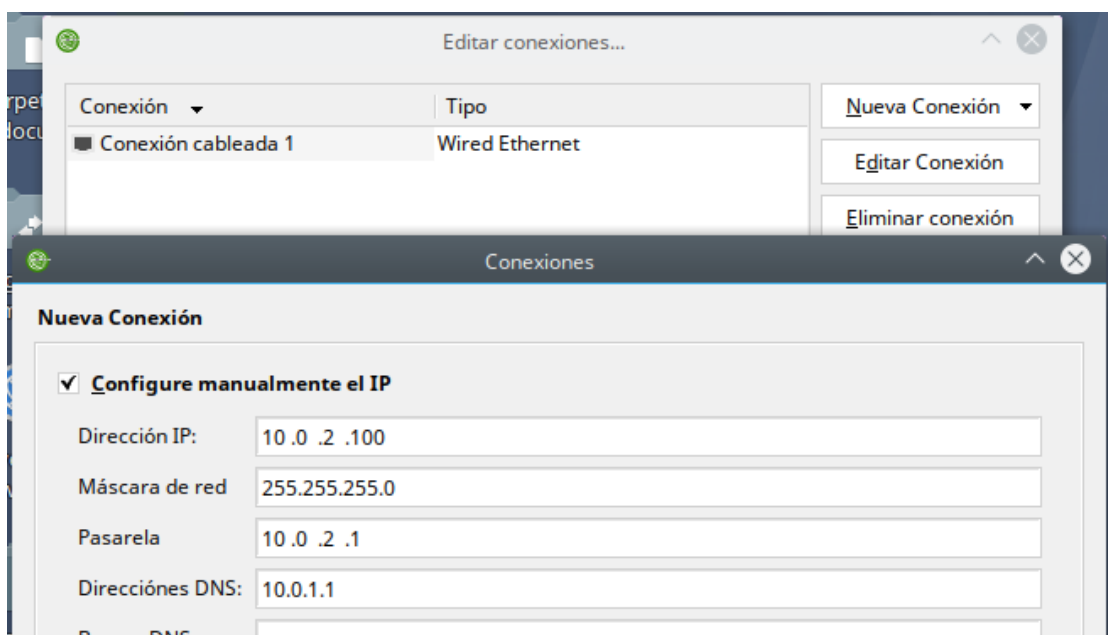
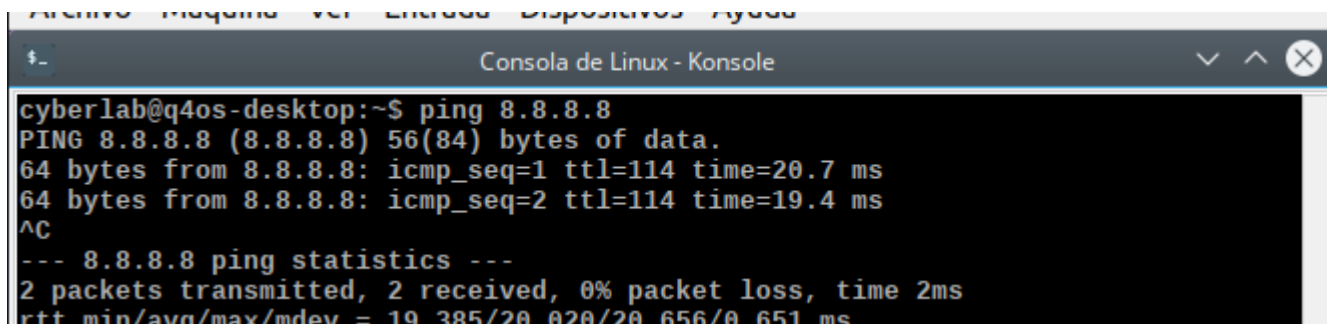


Figura 5: configuración de red de q4os

Podrás comprobar si tienes acceso a Internet una vez finalices la configuración del router que se explica en la siguiente sección.



### Configuración de la máquina Ubuntu-router

La máquina router, es un Ubuntu Server con 2 interfaces de red y un script en *iptables* que hace forwarding entre ellas.

Las credenciales son *cyberlab/cyberlab*

Los pasos son los siguientes:

1. crea en /etc el script **script\_iptables.sh**:

```
$ sudo nano /etc/script_iptables.sh
```

2. Escribe (o mejor pega si puedes hacer ssh desde kali linux :), lo siguiente:

```
#!/bin/bash

# activar forwarding
sysctl -w net.ipv4.ip_forward=1

# Habilitar NAT
iptables -t nat -F
iptables -t nat -A POSTROUTING -s 10.0.2.1/24 -j MASQUERADE
iptables -t nat -A PREROUTING -p tcp --dport 2022 -j DNAT --to-destination 10.0.2.100:22
iptables -t nat -A PREROUTING -p tcp --dport 8888 -j DNAT --to-destination 10.0.2.100:80
```

3. Dale al fichero permisos de ejecución:

```
$ sudo chmod +x /etc/script_iptables.sh
```

4. Comprueba que funciona:

```
$ sudo /etc/script_iptables.sh
```

Vamos a convertirlo en un servicio para que se ejecute automáticamente al arrancar. Para ello:

1. Crea el archivo **/lib/systemd/system/mi-firewall.service**

```
$ sudo nano /lib/systemd/system/mi-firewall.service
```

2. Pega dentro el siguiente contenido:

```
[Unit]
Description=Script iptables reenrutado
After=syslog.target

[Service]
Type=forking
ExecStart=/etc/script_iptables.sh

[Install]
WantedBy=multi-user.target
```

### 3. Lo configuramos para que arranque al inicio del sistema operativo:

```
$ sudo systemctl enable mi-firewall
```

### 4. Lo iniciamos para ver si funciona:

```
$ sudo systemctl start mi-firewall
```

### 5. Comprobamos su estado para ver que ha ido bien:

```
$ sudo systemctl status mi-firewall
```

### 6. Debería salirte esto:

```
● mi-firewall.service - Script iptables reenrutado
   Loaded: loaded (/lib/systemd/system/mi-firewall.service; enabled; vendor preset:
   enabled)
   Active: inactive (dead) since Thu 2021-01-28 17:35:33 UTC; 3s ago
     Process: 4253 ExecStart=/etc/script_iptables.sh (code=exited, status=0/SUCCESS)

Jan 28 17:35:33 ubuntuuser systemd[1]: Starting Script iptables reenrutado...
Jan 28 17:35:33 ubuntuuser script_iptables.sh[4254]: net.ipv4.ip_forward = 1
Jan 28 17:35:33 ubuntuuser systemd[1]: mi-firewall.service: Succeeded.
Jan 28 17:35:33 ubuntuuser systemd[1]: Started Script iptables reenrutado.
```

El **adaptador 2 de red** de esta máquina, es de tipo Red Interna, y está en una red de nombre **ciberlab-net-2**:

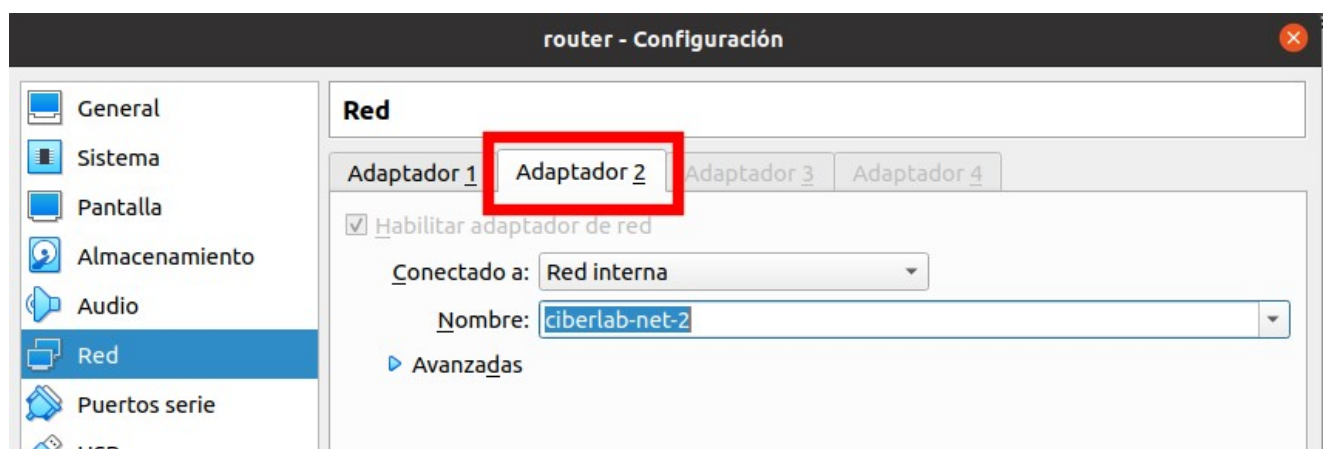


Figura 6: Adaptador de red 2 como red interna

También hay que poner las IPs estáticas en el sistema operativo, para ello:

#### 1. Edita el fichero:



```
$ sudo nano /etc/netplan/00-installer-config.yaml
```

## 2. El fichero ha de tener el siguiente contenido:

```
network:
  ethernets:
    enp0s3:
      addresses: [10.0.1.254/24]
      gateway4: 10.0.1.1
      dhcp4: no
      nameservers:
        addresses: [10.0.1.1]

    enp0s8:
      addresses: [10.0.2.1/24]
  version: 2
```

## 3. Aplica la configuración:

```
$ sudo netplan apply
```

Prueba que las máquinas de la red *cyberlab-net-2* tengan acceso a Internet.