

# Práctica 1.1. Protocolo IPv4. Servicio DHCP

## Objetivos

En esta práctica se presentan las herramientas que se utilizarán en la asignatura y se repasan brevemente los aspectos básicos del protocolo IPv4. Además, se analizan las características del protocolo DHCP.



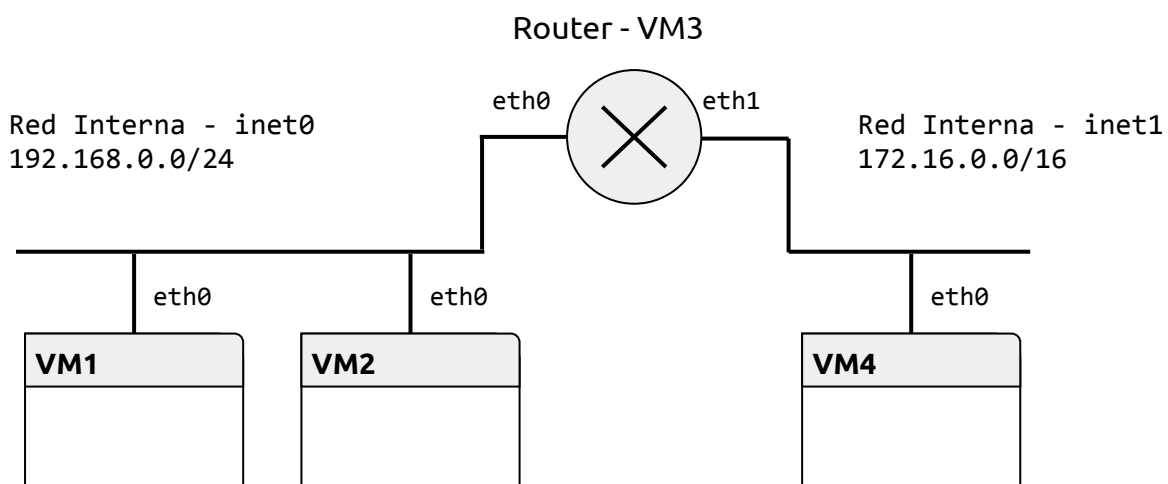
Para cada ejercicio, se tienen que proporcionar los **comandos utilizados con sus correspondientes salidas**, las **capturas de pantalla de Wireshark realizadas**, y la **información requerida de manera específica**.

## Contenidos

- Preparación del entorno para la práctica
- Configuración estática
- Encaminamiento estático
- Configuración dinámica

## Preparación del entorno para la práctica

Configuraremos la topología de red que se muestra en la siguiente figura:



Todos los elementos -el router y las máquinas virtuales VM- son *clones enlazados* de la máquina base ASOR-FE. La configuración de las máquinas se realizará con la utilidad `vtopo1`, que funciona en Linux y Mac (en Windows, la topología ha de crearse directamente con VirtualBox):

1. Definir la máquina base de la asignatura:

```
$ asorregenerate
```

Este comando crea la máquina virtual base (ASOR-FE) en la herramienta VirtualBox.

**Nota:** Este comando solo se debe usar en el laboratorio. En otros equipos, descargar [ASOR-FE.ova](#) e importarlo en VirtualBox.

2. Crear un archivo `pr1.topo1` con la topología de la red, que consta de 4 máquinas y dos redes. El

contenido del fichero es:

```
netprefix inet
machine 1 0 0
machine 2 0 0
machine 3 0 0 1 1
machine 4 0 1
```

La sintaxis es:

```
machine <número de VM> <interfaz0> <red0> <interfaz1> <red1> ...
```

3. Crear la topología de red que arrancará las 4 máquinas virtuales (VM1, VM2, Router y VM4).

```
$ vtopol pr1.topol
```

En VirtualBox se definirán las máquinas virtuales asorfemachine\_1 (VM1), asorfemachine\_2 (VM2), asorfemachine\_3 (Router - VM3) y asorfemachine\_4 (VM4).

**Nota:** Este comando está instalado en el laboratorio. En otros equipos, descargar [vtopol](#), dar permisos de ejecución al fichero (con `chmod +x`) y copiarlo, por ejemplo, en `/usr/local/bin`.



**Activar el portapapeles bidireccional en las máquinas** (menú Dispositivos) para copiar la salida de los comandos. Las capturas de pantalla se realizarán usando también Virtualbox (menú Ver)

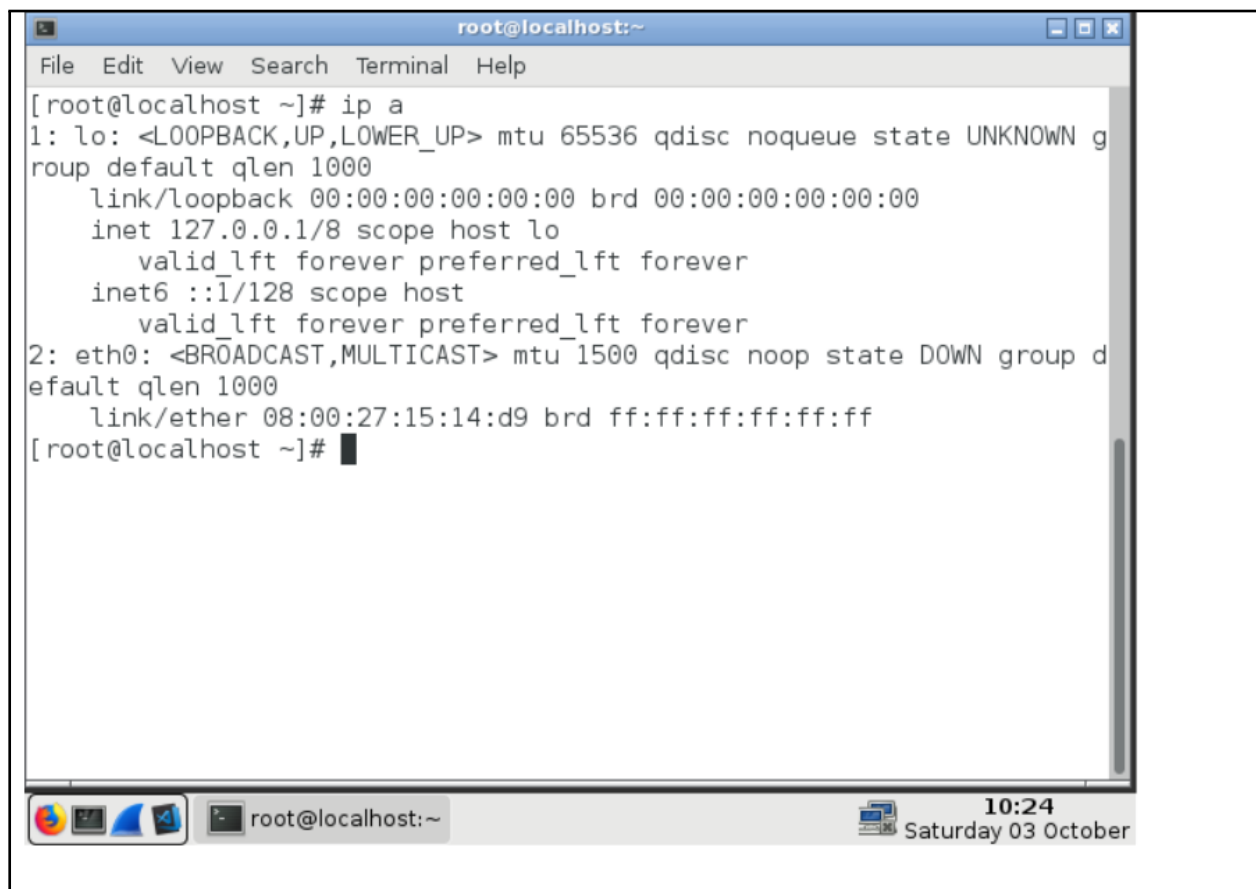
Las **credenciales de la máquina virtual** son: usuario `cursoresdes`, con contraseña `cursoresdes`.

## Configuración estática

En primer lugar, configuraremos cada red de forma estática asignando a cada máquina una dirección IP adecuada.

**Ejercicio 1 [VM1].** Determinar los interfaces de red que tiene la máquina y las direcciones IP y MAC que tienen asignadas. Utilizar el comando `ip`.

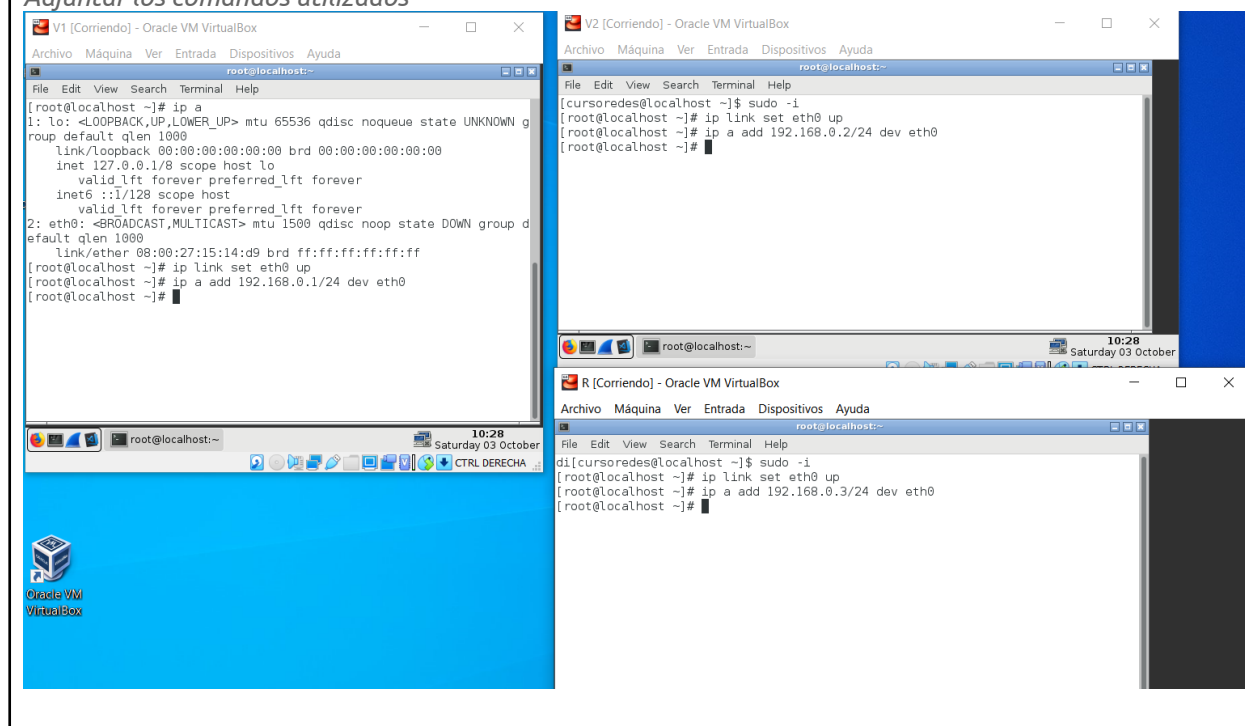
*Adjuntar el comando utilizado y su salida*



```
root@localhost:~  
File Edit View Search Terminal Help  
[root@localhost ~]# ip a  
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> mtu 1500 qdisc noop state DOWN group default qlen 1000  
    link/ether 08:00:27:15:14:d9 brd ff:ff:ff:ff:ff:ff  
[root@localhost ~]#
```

**Ejercicio 2 [VM1, VM2, Router].** Activar los interfaces eth0 en VM1, VM2 y Router, y asignar una dirección IP adecuada. La configuración debe realizarse con la utilidad ip, en particular los comandos ip address e ip link.

#### Adjuntar los comandos utilizados



```
V1 [Corriendo] - Oracle VM VirtualBox  
root@localhost:~  
File Edit View Search Terminal Help  
[root@localhost ~]# ip a  
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> mtu 1500 qdisc noop state DOWN group default qlen 1000  
    link/ether 08:00:27:15:14:d9 brd ff:ff:ff:ff:ff:ff  
[root@localhost ~]# ip link set eth0 up  
[root@localhost ~]# ip a add 192.168.0.1/24 dev eth0  
[root@localhost ~]#  
10:28 Saturday 03 October  
CTRL DERECHA  
Oracle VM VirtualBox  
V2 [Corriendo] - Oracle VM VirtualBox  
root@localhost:~  
File Edit View Search Terminal Help  
[cursoredes@localhost ~]$ sudo -i  
[root@localhost ~]# ip link set eth0 up  
[root@localhost ~]# ip a add 192.168.0.2/24 dev eth0  
[root@localhost ~]#  
10:28 Saturday 03 October  
R [Corriendo] - Oracle VM VirtualBox  
root@localhost:~  
File Edit View Search Terminal Help  
di[cursoredes@localhost ~]$ sudo -i  
[root@localhost ~]# ip link set eth0 up  
[root@localhost ~]# ip a add 192.168.0.3/24 dev eth0  
[root@localhost ~]#
```

**Ejercicio 3 [VM1, VM2].** Abrir la herramienta Wireshark en VM1 e iniciar una captura en el interfaz de red. Comprobar la conectividad entre VM1 y VM2 con la orden ping. Observar el tráfico generado, especialmente los protocolos encapsulados en cada datagrama y las direcciones origen y destino. Para ver correctamente el tráfico ARP, puede ser necesario eliminar la tabla ARP en VM1 con la orden `ip neigh flush dev eth0`.

Completar la siguiente tabla para todos los mensajes intercambiados hasta la recepción del primer mensaje ICMP Echo Reply:

Anotar las direcciones MAC e IP de los mensajes.

Para cada protocolo, anotar las características importantes (p. ej. pregunta/respuesta ARP o tipo ICMP) en el campo "Tipo de mensaje".

Comparar los datos observados durante la captura con el formato de los mensajes estudiados en clase.

MAC origen	MAC destino	Protocolo	IP origen	IP destino	Tipo de mensaje
MAC V1	broadcast	ARP	IP V1	IP V2	REQUEST
MAC V2	MAC V1	ARP	IP V2	IP V1	REPLY
MAC V1	MAC V2	ICMP	IP V1	IPV2	REQUEST
MAC V2	MAC V1	ICMP	IPV2	IPV1	REPLY

*Adjuntar una captura de pantalla de Wireshark con los mensajes ICMP y ARP*

The screenshot shows the Wireshark interface with the following details:

- Filter:** Expression... Clear Apply Save
- Packet List:**
  - 1 0.000000000 CadmusCo\_15:14:d9 Broadcast ARP 42 Who has 192.168.0.2? Tell 192.168.0.1
  - 2 0.000396654 CadmusCo\_15:14:d9 CadmusCo\_15:14:d9 ARP 60 192.168.0.2 is at 08:00:27:15:14:d9
  - 3 0.000407746 192.168.0.1 192.168.0.2 ICMP 98 Echo (ping) request id=0x089c, seq=1/256, ttl=64 (reply in 4)
  - 4 0.000565420 192.168.0.2 192.168.0.1 ICMP 98 Echo (ping) reply id=0x089c, seq=1/256, ttl=64 (request in 3)
  - 5 1.009351926 192.168.0.1 192.168.0.2 ICMP 98 Echo (ping) request id=0x089c, seq=2/512, ttl=64
  - 6 1.010803465 192.168.0.2 192.168.0.1 ICMP 98 Echo (ping) reply id=0x089c, seq=2/512, ttl=64 (request in 5)
  - 7 5.018658714 CadmusCo\_15:14:d9 CadmusCo\_15:14:d9 ARP 60 Who has 192.168.0.1? Tell 192.168.0.2
  - 8 5.018705330 CadmusCo\_15:14:d9 CadmusCo\_15:14:d9 ARP 42 192.168.0.1 is at 08:00:27:15:14:d9
- Packet Details:**
  - Frame 1: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface 0
  - Ethernet II, Src: CadmusCo\_15:14:d9 (08:00:27:15:14:d9), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
  - Address Resolution Protocol (request)
- Packet Bytes:**

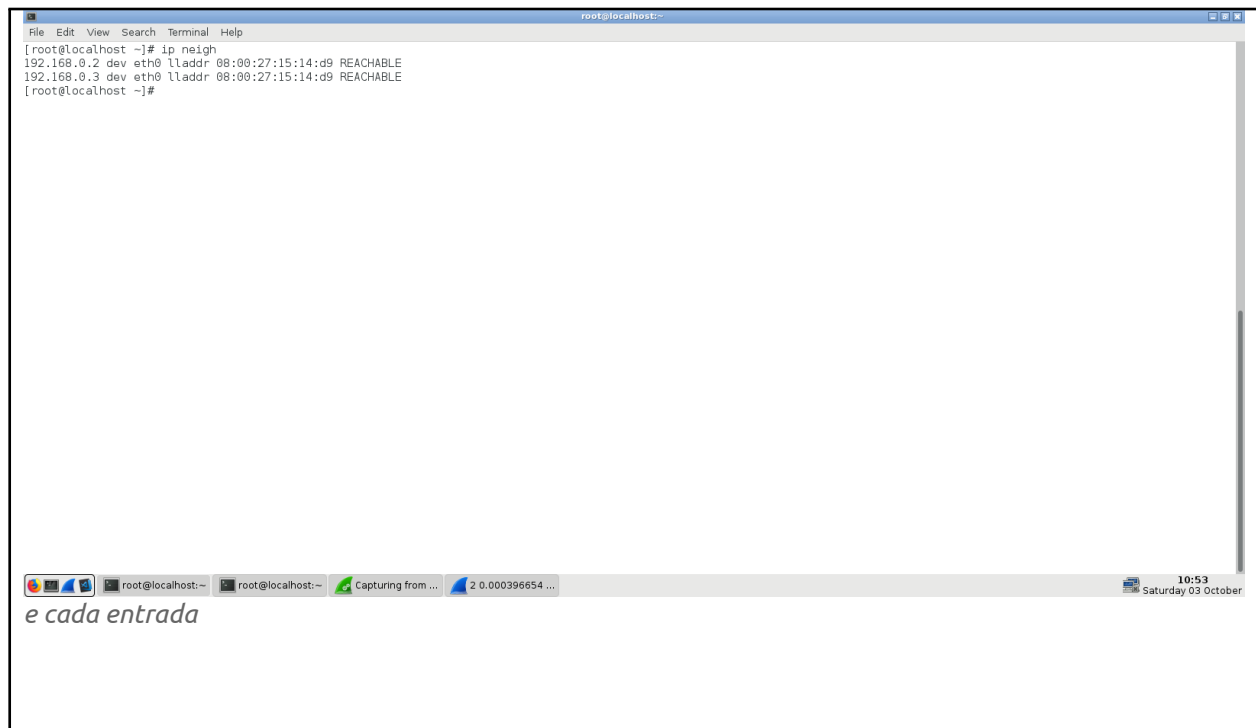
```

0000  ff ff ff ff ff ff 08 00 27 15 14 d9 08 06 00 01 .....
0010  08 00 06 04 00 01 08 00 27 15 14 d9 c0 a8 00 01 .....
0020  00 00 00 00 00 00 c0 a8 00 02 .....

```
- Status Bar:** eth0: <live capture in progress> File: ft Packets: 8 · Displayed: 8 (100.0%) Profile: Default 10:33 Saturday 03 October

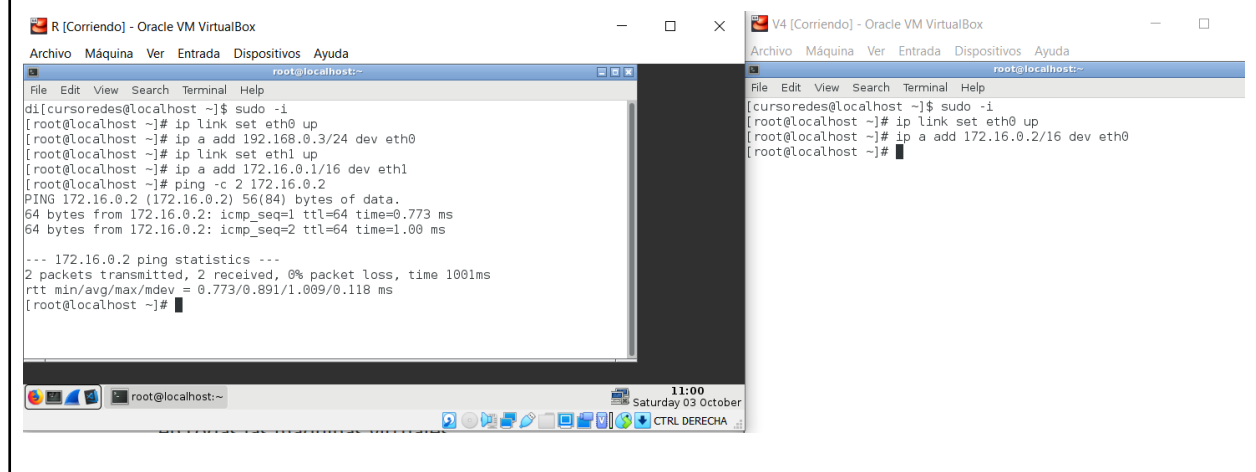
**Ejercicio 4 [VM1, VM2].** Ejecutar de nuevo la orden ping entre VM1 y VM2 y, a continuación, comprobar el estado de la tabla ARP en VM1 y VM2 usando el comando `ip neigh`. El significado del estado de cada entrada de la tabla se puede consultar en la página de manual del comando.

*Adjuntar la salida del comando `ip neigh` y describir el estado d*



**Ejercicio 5 [Router, VM4].** Configurar Router y VM4 y comprobar su conectividad con el comando ping.

Adjuntar los comandos utilizados y la salida del comando ping



## Encaminamiento estático

Según la topología de esta práctica, Router puede encaminar el tráfico entre ambas redes. En esta sección, vamos a configurar el encaminamiento estático, basado en rutas que fijaremos manualmente en todas las máquinas virtuales.

**Ejercicio 6 [Router].** Activar el reenvío de paquetes (*forwarding*) en Router para que efectivamente pueda funcionar como encaminador entre las redes. Ejecutar el siguiente comando:

```
$ sudo sysctl net.ipv4.ip_forward=1
```

**Ejercicio 7 [VM1, VM2].** Añadir Router como encaminador por defecto para VM1 y VM2. Usar el comando `ip route`.

*Adjuntar el comando utilizado*

`ip route add default via 192.168.0.3`

**Ejercicio 8 [VM4].** Aunque la configuración adecuada para la tabla de rutas en redes como las consideradas en esta práctica consiste en añadir una ruta por defecto, es posible incluir rutas para redes concretas. Añadir en VM4 una ruta a la red 192.168.0.0/24 vía Router. Usar el comando `ip route`.

*Adjuntar el comando utilizado*

`ip route add 192.168.0.0/24 via 172.16.0.2`

**Ejercicio 9 [VM1, VM4, Router].** Abrir la herramienta Wireshark en Router e iniciar una captura en sus dos interfaces de red. Eliminar la tabla ARP en VM1 y Router. Usar la orden `ping` entre VM1 y VM4. Completar la siguiente tabla para todos los paquetes intercambiados hasta la recepción del primer *Echo Reply*.

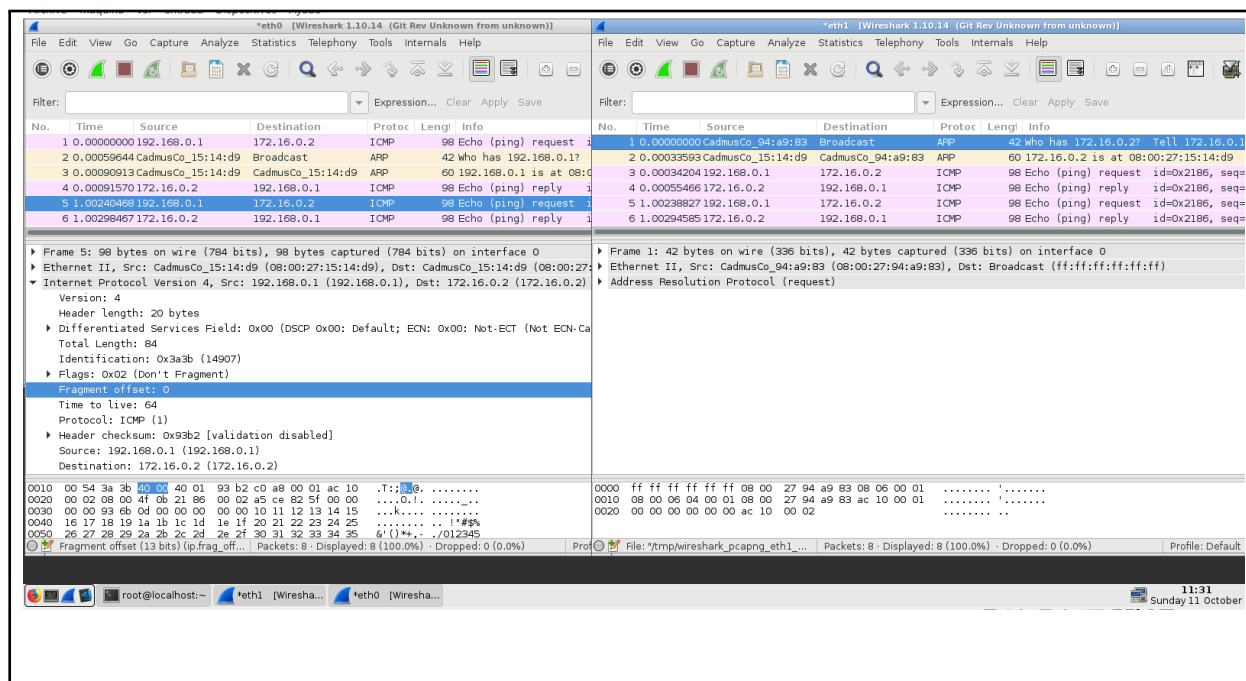
**Red 192.168.0.0/24 - Router (eth0)**

MAC origen	MAC destino	Protocolo	IP origen	IP destino	Tipo de mensaje
Router	Broadcast	ARP	Router	V1	REQUEST
V1	Router	ARP	V1	ROUTER	REPLY
V1	V4	ICMP	V1	V4	REQUEST
V4	V1	ICMP	V4	V1	REPLY

**Red 172.16.0.0/16 - Router (eth1)**

MAC origen	MAC destino	Protocolo	IP origen	IP destino	Tipo de mensaje
Router	BROADCAST	ARP	ROUTER	V4	REQUEST
V4	ROUTER	ARP	V4	ROUTER	REPLY
V1	V4	ICMP	V1	V4	REQUEST
V4	V1	ICMP	V4	V1	REPLY

*Adjuntar una captura de pantalla de Wireshark con los mensajes ICMP y ARP.*



## Configuración dinámica

El protocolo DHCP permite configurar dinámicamente los parámetros de red de una máquina. En esta sección configuraremos Router como servidor DHCP para las dos redes. Aunque DHCP puede incluir muchos parámetros de configuración, en esta práctica sólo fijaremos el encaminador por defecto.

**Ejercicio 10 [VM1, VM2, VM4].** Eliminar las direcciones IP de los interfaces (`ip addr del`) de todas las máquinas salvo Router.

**Ejercicio 11 [Router].** Configurar el servidor DHCP para las dos redes:

- Editar el fichero `/etc/dhcp/dhcpd.conf` y añadir dos secciones subnet, una para cada red, que definan los rangos de direcciones, `192.168.0.50-192.168.0.100` y `172.16.0.50-172.16.0.100`, respectivamente. Además, incluir la opción `routers` con la dirección IP de Router en cada red. Ejemplo:

```
subnet 192.168.0.0 netmask 255.255.255.0 {
    range 192.168.0.11 192.168.0.50;
    option routers 192.168.0.3;
    option broadcast-address 192.168.0.255;
}
```

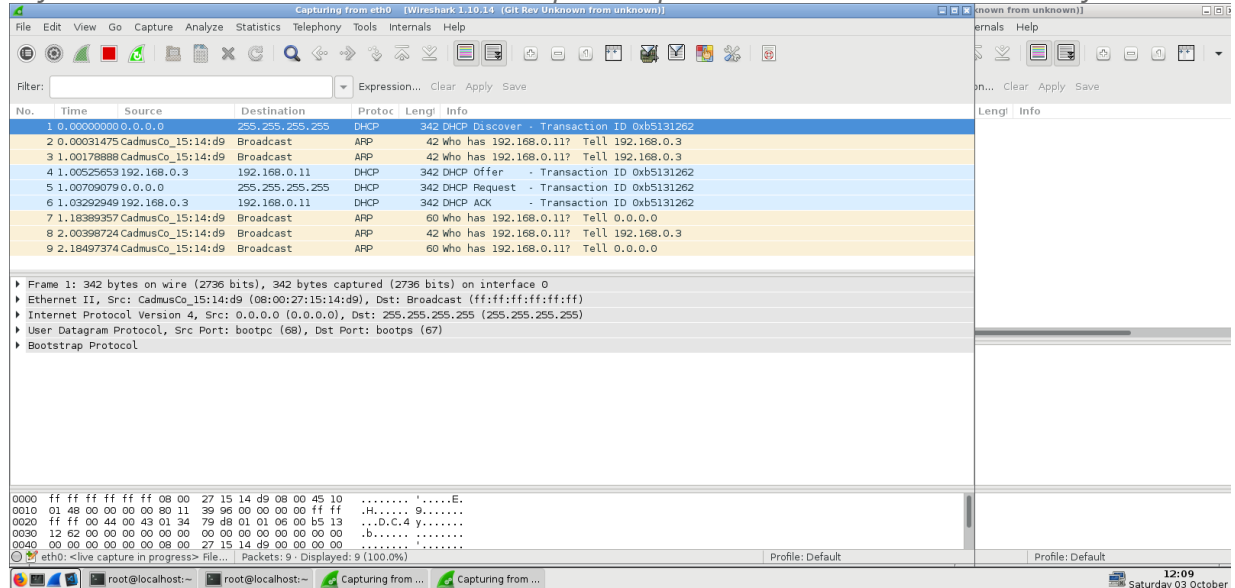
- Arrancar el servicio con el comando `service dhcpd start`.

**Ejercicio 12 [Router, VM1].** Iniciar una captura de paquetes en Router. Arrancar el cliente DHCP en VM1 con `dhclient -d eth0` y observar el proceso de configuración. Completar la siguiente tabla:

IP Origen	IP Destino	Mensaje DHCP	Opciones DHCP
0.0.0.0	BROADCAST	DISCOVER	
192.168.0.3	192.168.0.11	REQUEST	

0.0.0.0	broadcast	OFFER	
192.168.0.3	192.168.0.11	ACK	

Adjuntar la salida del comando `dhclient` una captura de pantalla de Wireshark con los mensajes DHCP.



**Ejercicio 13 [VM4].** Durante el arranque del sistema se pueden configurar automáticamente interfaces según la información almacenada en el disco del servidor (configuración persistente). Consultar el fichero `/etc/sysconfig/network-scripts/ifcfg-eth0` de VM4, que configura automáticamente `eth0` usando DHCP. Para configuración estática, se usarían las siguientes opciones:

```
TYPE=Ethernet
BOOTPROTO=none
IPADDR=172.16.0.16
GATEWAY=172.16.0.1
DEVICE=eth0
```

**Nota:** Estas opciones se describen en detalle en `/usr/share/doc/initscripts-*/sysconfig.txt`.

**Ejercicio 14 [VM2, VM4].** Comprobar la configuración persistente con las órdenes `ifup` e `ifdown`. Verificar la conectividad entre todas las máquinas de las dos redes.