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.

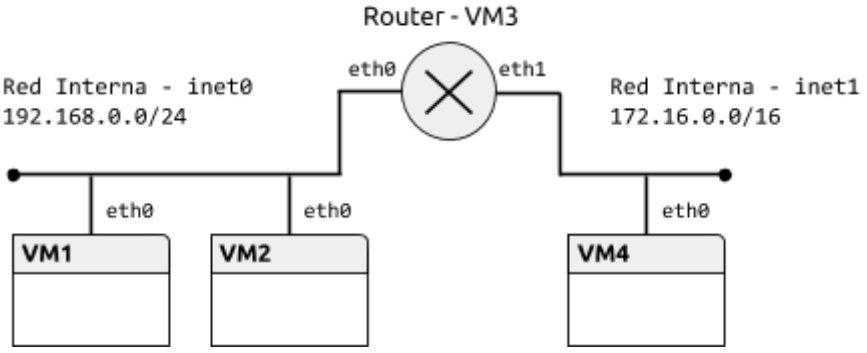
Para cada ejercicio, se tienen que proporcionar los **comandos utilizados con sus correspondientes salidas**, las **capturas de pantalla de Wireshark realizadas**, y la **información requerida de manera específica**.

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 `vtopol`, 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.topol` 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 `cursoredes`, con contraseña `cursoredes`.

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
Comando: ip address

Salida:
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 08:00:27:6c:99:f6 brd ff:ff:ff:ff:ff:ff

Hay una única interfaz (eth0), aún sin direcciones IP, y con dirección MAC: 08:00:27:6c:99:f6

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
Nos convertimos en root mediante: sudo -i

[VM1]
ip link set dev eth0 up
ip address add 192.168.0.1/24 dev eth0

[VM2]
ip link set dev eth0 up
ip address add 192.168.0.2/24 dev eth0

[Router]
ip link set dev eth0 up
ip address add 192.168.0.3/24 dev eth0

ip address show en cada máquina para comprobar que todo está correcto

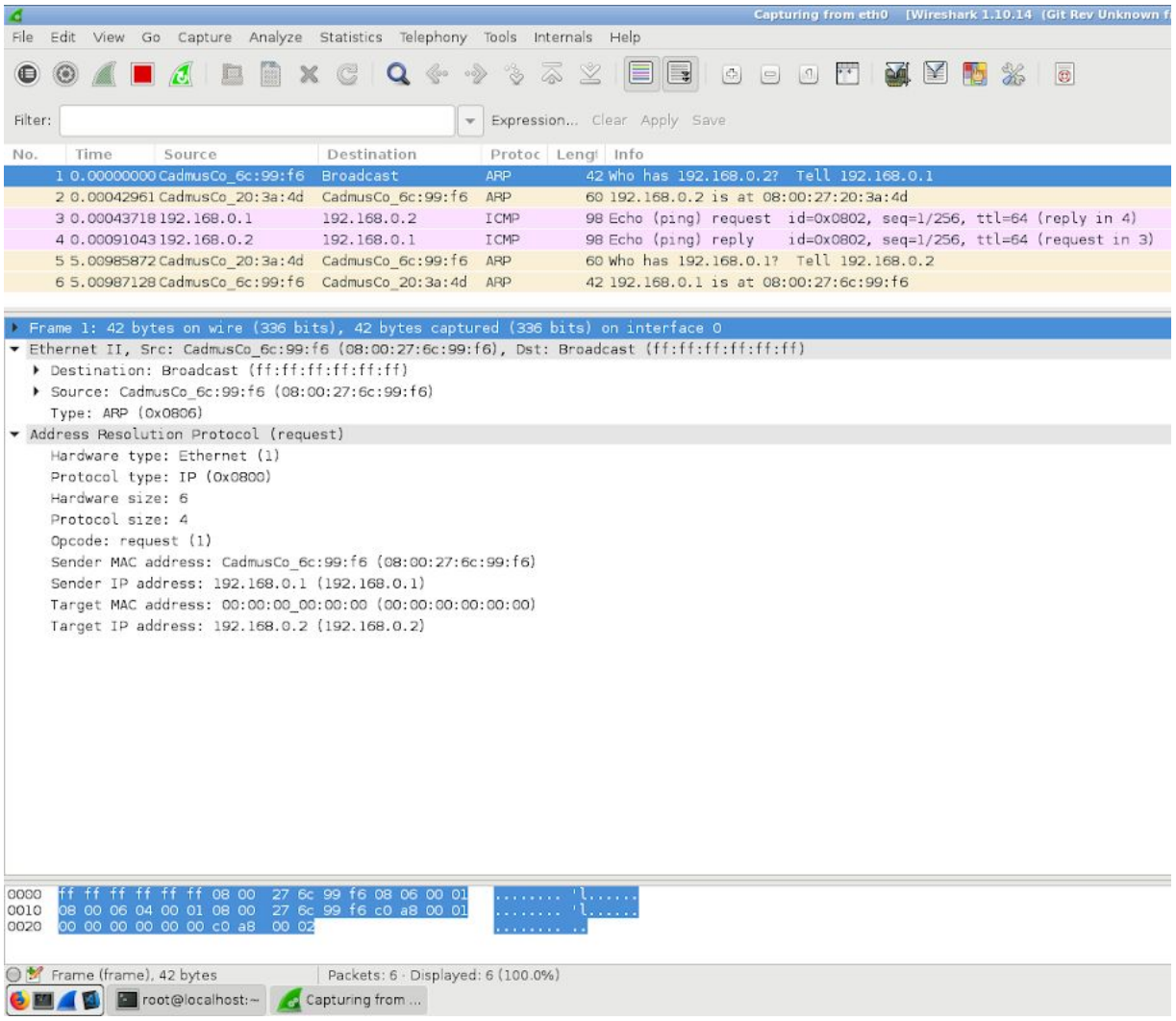
Ejercicio 3 [VM1, VM2]. Abrir la herramienta Wireshark en VM1 e iniciar una captura en el interfaz de red. Comprobar la conectividad entre VM1 y VM2 con la orden `ping`. Observar el tráfico generado, especialmente los protocolos encapsulados en cada datagrama y las direcciones origen y destino. Para 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**
08:00:27:6c:99:f6	ff:ff:ff:ff:ff:ff	ARP	192.168.0.1	192.168.0.2	Pregunta ARP
08:00:27:20:3a:4d	*08:00:27:6c:99:f6*	ARP	192.168.0.2	192.168.0.1	Respuesta ARP
08:00:27:6c:99:f6	*08:00:27:20:3a:4d*	ICMP	192.168.0.1	192.168.0.2	Request ICMP
08:00:27:20:3a:4d	*08:00:27:6c:99:f6*	ICMP	192.168.0.2	192.168.0.1	Reply ICMP

Adjuntar una captura de pantalla de Wireshark con los mensajes ICMP y ARP
Orden ping -c 1 192.168.0.2 desde [VM1]



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 de cada entrada

[VM1]
[root@localhost ~]# ip neigh
192.168.0.2 dev eth0 lladdr 08:00:27:20:3a:4d REACHABLE

[VM1]
[root@localhost ~]# ip neigh
192.168.0.1 dev eth0 lladdr 08:00:27:6c:99:f6 REACHABLE

El estado REACHABLE según el manual: "the neighbour entry is valid until the reachability timeout expires". Es decir, que es una entrada válida hasta que el temporizador de alcanzabilidad acabe.

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

[Router]
[root@localhost ~]# ip link set dev eth1 up
[root@localhost ~]# ip address add 172.16.0.3/16 dev eth1

[VM4]
[root@localhost ~]# ip link set dev eth0 up
[root@localhost ~]# ip address add 172.16.0.4/16 dev eth0

[Router]
[root@localhost ~]# ping -c 1 172.16.0.4

Salida de ping:
PING 172.16.0.4 (172.16.0.4) 56(84) bytes of data.
64 bytes from 172.16.0.4: icmp_seq=1 ttl=64 time=0.831 ms

--- 172.16.0.4 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.831/0.831/0.831/0.000 ms

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

Salida: 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

[VM1] y [VM2]
[root@localhost ~]# 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

[VM4]
[root@localhost ~]# ip route add 192.168.0.0/24 via 172.16.0.3

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*.

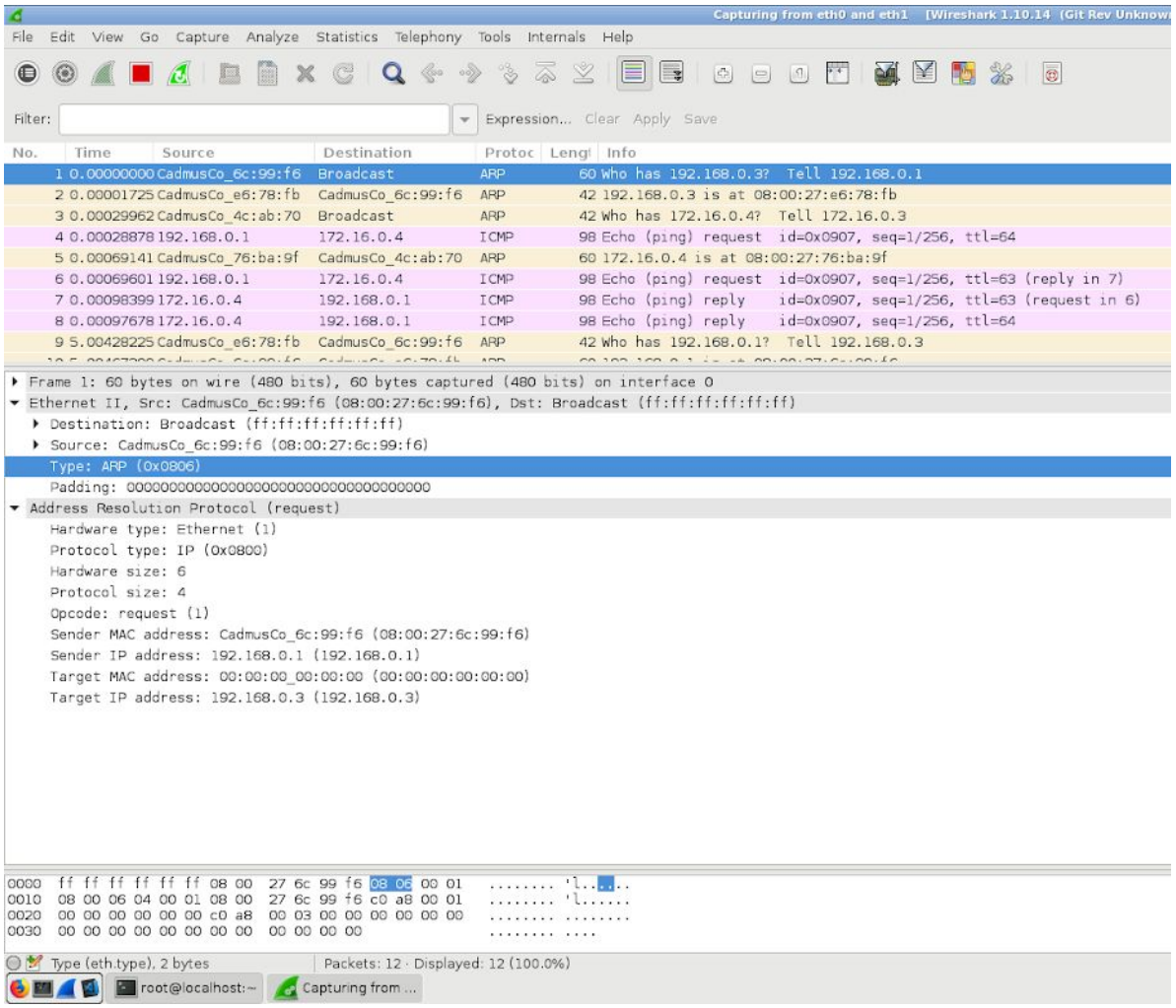
Red 192.168.0.0/24 - Router (eth0)

MAC origen	MAC destino	Protocolo	IP origen	IP destino	Tipo de mensaje
08:00:27:6c:99:f6	ff:ff:ff:ff:ff:ff	ARP	192.168.0.1	192.168.0.3	Pregunta ARP
08:00:27:e6:78:fb	*08:00:27:6c:99:f6*	ARP	192.168.0.3	192.168.0.1	Respuesta ARP
08:00:27:6c:99:f6	08:00:27:e6:78:fb	ICMP	192.168.0.1	172.16.0.4	Request ICMP
08:00:27:e6:78:fb	*08:00:27:6c:99:f6*	ICMP	172.16.0.4	192.168.0.1	Reply ICMP

Red 172.16.0.0/16 - Router (eth1)

MAC origen	MAC destino	Protocolo	IP origen	IP destino	Tipo de mensaje
08:00:27:4c:ab:70	ff:ff:ff:ff:ff:ff	ARP	172.16.0.3	172.16.0.4	Pregunta ARP
08:00:27:76:ba:9f	08:00:27:4c:ab:70	ARP	172.16.0.4	172.16.0.3	Respuesta ARP
08:00:27:4c:ab:70	08:00:27:76:ba:9f	ICMP	192.168.0.1	172.16.0.4	Request ICMP
08:00:27:76:ba:9f	08:00:27:4c:ab:70	ICMP	172.16.0.4	192.168.0.1	Reply ICMP

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;
  }
```

```
Se ha hecho:  [root@localhost ~]# nano /etc/dhcp/dhcpd.conf

Para poner:

subnet 192.168.0.0 netmask 255.255.255.0 {
   range 192.168.0.50 192.168.0.100;
   option routers 192.168.0.3;
   option broadcast-address 192.168.0.255;
}

subnet 172.16.0.0 netmask 255.255.0.0 {
   range 172.16.0.50 172.16.0.100;
   option routers 172.16.0.3;
   option broadcast-address 172.16.255.255;
}
```

- 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:

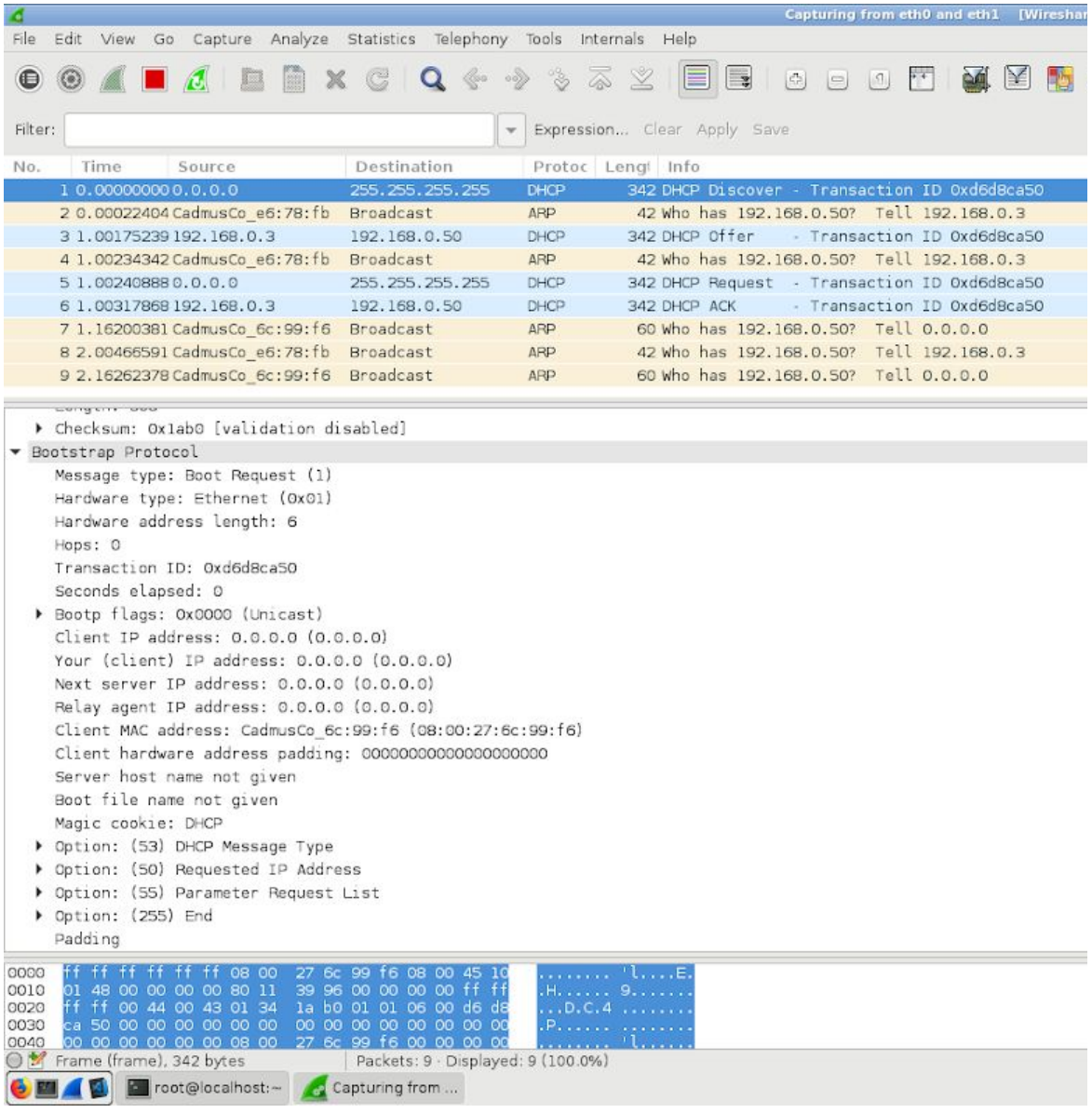
IP Origen	**IP Destino**	**Mensaje DHCP**	**Opciones DHCP**
0.0.0.0	*255.255.255.255*	*DHCP Discover*	*DHCP Message Type: 1 (Discover)* *Requested IP Address (10.0.2.15)* *Parameter Request List* *End*
192.168.0.3	*192.168.0.50*	*DHCP Offer*	*DHCP Message Type: 2 (Offer)* *DHCP Server Identifier (192.168.0.3)* *IP Address Lease Time (12 hours)* *SubnetMask (255.255.255.0)* *BroadcastAddress(192.168.0.255)* *Router (192.168.0.3)* *End*
0.0.0.0	*255.255.255.255*	*DHCP Request*	*DHCP Message Type: 3 (Request)* *Requested IP Address (192.168.0.50)* *Parameter Request List* *End*
192.168.0.3	*192.168.0.50*	*DHC ACK*	*DHCP Type: 5 (ACK)* *DHCP Server Identifier (192.168.0.3)* *IP Address Lease Time (12 hours)* *SubnetMask (255.255.255.0)* *BroadcastAddress(192.168.0.255)* *Router (192.168.0.3)* *End*

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

***Salida del comando**:*

Internet Systems Consortium DHCP Client 4.2.5
Copyright 2004-2013 Internet Systems Consortium.
All rights reserved.
For info, please visit https://www.isc.org/software/dhcp/

Listening on LPF/eth0/08:00:27:6c:99:f6
Sending on LPF/eth0/08:00:27:6c:99:f6
Sending on Socket/fallback
DHCPDISCOVER on eth0 to 255.255.255.255 port 67 interval 6 (xid=0x50cad8d6)
DHCPREQUEST on eth0 to 255.255.255.255 port 67 (xid=0x50cad8d6)
DHCPOFFER from 192.168.0.3
DHCPACK from 192.168.0.3 (xid=0x50cad8d6)
bound to 192.168.0.50 -- renewal in 16592 seconds.



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:

```
Consulta: [root@localhost ~]# cat /etc/sysconfig/network-scripts/ifcfg-eth0

Resultado:
[root@localhost ~]# cat /etc/sysconfig/network-scripts/ifcfg-eth0
TYPE=Ethernet
BOOTPROTO=dhcp
DEFROUTE=yes
NAME=eth0
DEVICE=eth0
ONBOOT=no

Con configuración estática sería:
TYPE=Ethernet
BOOTPROTO=none
IPADDR=<dirección IP estática en formato CIDR>
GATEWAY=<dirección IP estática del encaminador por defecto (si existe)>
DEVICE=eth0
```

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

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

```
Hacemos ifup eth0:

Consultamos con ip address:

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 08:00:27:76:ba:9f brd ff:ff:ff:ff:ff:ff
```

inet 172.16.0.50/16 brd 172.16.255.255 scope global dynamic eth0

valid_lft 43196sec preferred_lft 43196sec

inet6 fe80::a00:27ff:fe76:ba9f/64 scope link

valid_lft forever preferred_lft forever

Nos ha asignado la dirección IP 172.16.0.50/16 a VM4.

Comprobamos la conectividad de todas las máquinas. Por ejemplo, si hacemos ping -c 1 192.168.0.50 desde VM4 para enviar un mensaje a VM1 obtenemos:

PING 192.168.0.50 (192.168.0.50) 56(84) bytes of data.

64 bytes from 192.168.0.50: icmp_seq=1 ttl=63 time=1.42 ms

--- 192.168.0.50 ping statistics ---

1 packets transmitted, 1 received, 0% packet loss, time 0ms

rtt min/avg/max/mdev = 1.423/1.423/1.423/0.000 ms

[root@localhost ~]# ping -c 1 192.168.0.50

PING 192.168.0.50 (192.168.0.50) 56(84) bytes of data.

64 bytes from 192.168.0.50: icmp_seq=1 ttl=63 time=0.693 ms

--- 192.168.0.50 ping statistics ---

1 packets transmitted, 1 received, 0% packet loss, time 0ms

rtt min/avg/max/mdev = 0.693/0.693/0.693/0.000 ms

Con ifdown eth0 desactivamos la interfaz.