Práctica 1.1. IPv4 y 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.

**Contenidos**

[Preparación del entorno para la práctica](#_14a3ftqman5y)

[Configuración estática](#_7799bdv34ent)

[Encaminamiento estático](#_w2ynjijgd1kw)

[Configuración dinámica](#_nj2ry1q77ael)

# Preparación del entorno para la práctica

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

**vm\_3router**

**vm\_2**

enp0s3

**vm\_1**

net\_0

192.168.0.0/24

enp0s3

**vm\_4**

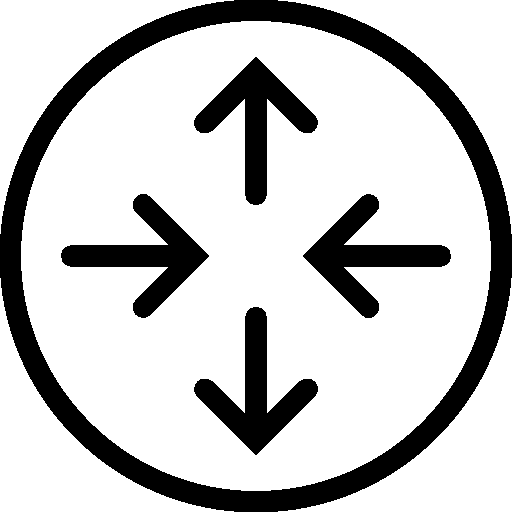
enp0s3

net\_1

172.16.0.0/16

enp0s3

enp0s8



Todas las máquinas virtuales -el encaminador y los nodos- son *clones enlazados* de la máquina base ASOR-FE. La topología se creará con la utilidad vtopol instalada en los equipos del laboratorio (en otros equipos Linux y Mac, descargar el fichero [vtopol](https://drive.google.com/file/d/0B3JrkzcA788mZTAtX0VQWVlrN2c/view?usp=sharing&resourcekey=0-kx5XvWNjkxOFuJ4RxFTd7A), dar permisos de ejecución con chmod +x vtopol y copiar en /usr/local/bin; en otros equipos Windows, la topología ha de crearse manualmente con VirtualBox siguiendo esta [guía](https://docs.google.com/document/u/0/d/1TNfzyejVvSSRP9u9j6JbZFxJI8mnhmmzI_518Db7F24/edit)):

1. Borrar las máquinas virtuales existentes ejecutando el siguiente comando en la consola:

|  |
| --- |
| $ **rm -rf $HOME/VirtualBox\ VMs/** |

1. En los equipos del laboratorio, ir al directorio /mnt/DiscoVMs/ASOR usando el gestor de ficheros (hay un acceso directo en el escritorio) y localizar el fichero ASOR-FE.ova (en otros equipos, descargar el fichero [ASOR-FE.ova](https://drive.google.com/file/d/1AFYebQ04-33i_hsUCc0b7G6GHe20yLy1/view?usp=drive_link)). Abrir el fichero haciendo doble-clic y pulsar “Terminar” para importar la máquina virtual base ASOR-FE en VirtualBox. Alternativamente, se puede usar la opción “Importar servicio virtualizado” desde VirtualBox.
2. Crear un fichero pr1.topol con la topología de la red, que consta de 4 máquinas virtuales y dos redes internas. El contenido del fichero debe ser:

|  |
| --- |
| prefix vm  netprefix net machine 1 0 0 machine 2 0 0  machine 3router 0 0 1 1  machine 4 0 1 |

La sintaxis es:

|  |
| --- |
| machine <VM> <interfaz0> <red0> <interfaz1> <red1> ... |

1. Para crear la topología de red que arrancará las 4 máquinas virtuales (vm\_1, vm\_2, vm\_3router y vm\_4), ejecutar:

|  |
| --- |
| $ **vtopol pr1.topol** |

# Configuración estática

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

***Ejercicio 1 [vm\_1].*** Determinar las interfaces de red que tiene la máquina y las direcciones IP y MAC que tienen asignadas. Utilizar los comandos ip address e ip link.

***Interfaz Lo -- DIreccion ip-127.0.0.1/8***

***enp0s3***

***enpos8***

***VM1,2 y 4 ----> lo y enpos3***

***VM3---->lo(loopback),enpos3 y enpos8***

***Ejercicio 2 [vm\_1, vm\_2, vm\_3router].*** Activar las interfaces enp0s3 en vm\_1, vm\_2 y vm\_3router, y asignar una dirección IP adecuada. Utilizar los comandos ip address e ip link.

**sudo ip link set enp0s3 up**

**sudo ip address add 192.168.1.10/24 dev enp0s3**

|  |
| --- |
| **Para saber más …** |
| Tradicionalmente el nombre de las interfaces ha seguido el esquema *“eth0”, “eth1”, “wlan0”, etc.* La forma en la que el kernel asignaba estos nombres se basaba en el orden en el que los distintos drivers informaban al kernel de su existencia. Aunque esto producía nombres de interfaces fáciles de recordar, se podría dar el caso de que distintas interfaces recibieran distintos nombres cada vez que se reiniciaba la máquina.  Para evitar este problema, desde la versión v197 de systemd/udev, los nombres de cada interfaz se asignan a partir de información invariante entre reinicios (proveniente del firmware, BIOS, PCIe, conector físico usado, dirección MAC, etc.). En función de qué información se utilice para dar el nombre a la interfaz, ésta puede tener distintos formatos: *eno1, ens1, enp2s0, enx78e7d1ea46da, eth0, etc.*   Para más información, consultar la página de manual systemd.net-naming-scheme(7) y <https://systemd.io/PREDICTABLE_INTERFACE_NAMES/> |

***Ejercicio 3 [vm\_1, vm\_2].*** Abrir la herramienta Wireshark en vm\_1 e iniciar una captura en la interfaz de red. Desde vm\_1, comprobar la conectividad con vm\_2 usando 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 vm\_1 con la orden ip neigh flush dev enp0s3.

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

* 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.

**sudo wireshark**

**sudo ip neigh flush dev enp0s3 --borrar tablas arp**

**sudo ping 192.168.0.15**

| **MAC origen** | **MAC destino** | **Protocolo** | **IP origen** | **IP destino** | **Tipo de mensaje** |
| --- | --- | --- | --- | --- | --- |
| 08:00:27:12:32:77 | Ff:ff:ff:ff:ff | ARP | 192.168.0.10 | 192.168.0.15 | Arp request |
| 08:00:27:62:ff:ee | 08:00:27:12:32:77 | ARP | 192.168.0.15 | 192.168.0.10 | Arp reply |
| 08:00:27:12:32:77 | 08:00:27:62:ff:ee | ICMP | 192.168.0.10 | 192.168.0.15 | Echo request |
| 08:00:27:62:ff:ee | 08:00:27:12:32:77 | Icmp | 192.168.0.15 | 192.168.0.10 | Echo reply |

***Ejercicio 4 [vm\_1, vm\_2].*** Ejecutar de nuevo la orden ping entre vm\_1 y vm\_2 y, a continuación, comprobar el estado de la tabla ARP en vm\_1 y vm\_2 usando el comando ip neigh. El significado del estado de cada entrada de la tabla se puede consultar en la página de manual ip-neighbour(8).

**192.168.0.15 dev enp0s3 lladdr 08:00:27:62:ff:ee STALE**

**Ip,red,mac, estado inactivo**

***Ejercicio 5 [vm\_3router, vm\_4].***Configurar vm\_3router y vm\_4 y comprobar su conectividad con el comando ping.

**sudo ip address add 172.16.0.15/16 dev enp0s3**

**sudo ip address add 172.16.0.10/16 dev enp0s8 (/en routerI**

**sudo ping 172.16.0.10**

# Encaminamiento estático

Según la topología de esta práctica, vm\_3router 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 [vm\_3router].*** Activar el reenvío de paquetes (*forwarding*) en vm\_3router para que efectivamente pueda funcionar como encaminador entre las redes. Ejecutar el siguiente comando:

|  |
| --- |
| $ **sudo sysctl net.ipv4.ip\_forward=1** |

***Ejercicio 7 [vm\_1, vm\_2].*** Establecer vm\_3router como encaminador por defecto para vm\_1 y vm\_2. Usar el comando ip route.ç

**sudo ip route add default via 192.168.0.18**

***Ejercicio 8 [vm\_4].*** 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 vm\_4 una ruta a la red 192.168.0.0/24 a través de vm\_3router. Usar el comando ip route

**sudo ip route add 192.168.0.0/24 via 172.16.0.10**

***Ejercicio 9 [vm\_1, vm\_3router, vm\_4].***Abrir la herramienta Wireshark en vm\_3router e iniciar dos capturas, una en cada interfaz de red. Eliminar la tabla ARP en vm\_1 y vm\_3router. Desde vm\_1, comprobar la conectividad con vm\_4 usando la orden ping. Completar la siguiente tabla para todos los paquetes intercambiados hasta la recepción del primer *Echo Reply*.

**Red 192.168.0.0/24 - vm\_3router (enp0s3)**

| **MAC origen** | **MAC destino** | **Protocolo** | **IP origen** | **IP destino** | **Tipo de mensaje** |
| --- | --- | --- | --- | --- | --- |
| 08:00:27:12:32:77 | Ff:ff:ff:ff:ff:ff | ARP | 192.168.0.10 | 192.168.0.18 | Pregunta ARP |
| 08 00 27 ff b8 88 | 08:00:27:12:32:77 | ARP | 192.168.0.18 | 192.168.0.10 | Respuesta ARP |
| 08:00:27:12:32:77 | 08 00 27 ff b8 88 | ICMP | 192.168.0.10 | 172.16.0.15 | Echo request |
| 08 00 27 ff b8 88 | 08:00:27:12:32:77 | ICMP | 172.16.0.15 | 192.168.0.10 | Echo reply |

**Red 172.16.0.0/16 - vm\_3router (enp0s8)**

| **MAC origen** | **MAC destino** | **Protocolo** | **IP origen** | **IP destino** | **Tipo de mensaje** |
| --- | --- | --- | --- | --- | --- |
| 08 00 27 5a be 09 | Ff ff ff ff ff ff | Arp | 172.16.0.10 | 172.16.0.15 | Pregunta arp |
| 08 00 27 dd 56 67 | 08 00 27 5a be 09 | Arp | 172.16.0.15 | 172.16.0.10 | Respuesta arp |
| 08 00 27 5a be 09 | 08 00 27 dd 56 67 | Icmp | 192.168.0.10 | 172.16.0.15 | Icmp request |
| 08 00 27 dd 56 67 | 08 00 27 5a be 09 | icmp | 172.16.0.15 | 192.168.0.10 | Icmp reply |

# Configuración dinámica

**0.0.0**

El protocolo DHCP permite configurar dinámicamente los parámetros de red de una máquina. En esta sección configuraremos vm\_3router 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 [vm\_1, vm\_2, vm\_4].***Eliminar las direcciones IP de las interfaces (con ip addr del) de todas las máquinas salvo vm\_3router. Observa que al eliminar la dirección IP se elimina automáticamente la entrada asociada en la tabla de rutas.

**sudo ip addr del 192.168.0.10/24 dev enp0s3**

***Ejercicio 11 [vm\_3router].***Configurar el servidor DHCP en vm\_3router para las dos redes:

* Editar el fichero /etc/dhcp/dhcpd.conf y añadir dos secciones subnet, una para cada red, que definan, respectivamente, los rangos de direcciones192.168.0.50-192.168.0.100 y 172.16.0.50-172.16.0.100. Además, incluir la opción routers con la dirección IP de vm\_3router en cada red. Ejemplo:

**sudo nano /etc/dhcp/dhcpd.conf**

**subnet 192.168.0.0 netmask 255.255.255.0 {**

**range 192.168.0.50 192.168.0.100;**

**option routers 192.168.0.18;**

**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 172.16.0.10;**

**option broadcast-address 172.16.255.255;**

**sudo systemctl start isc-dhcp-server**

**}**

|  |
| --- |
| 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; } |

* Arrancar el servicio con el comando sudo systemctl start isc-dhcp-server
* Se puede consultar el estado del servicio con los comandos systemctl status isc-dhcp-server y journalctl -eu isc-dhcp-server

***Ejercicio 12 [vm\_1, vm\_3router].*** Iniciar una captura de paquetes en vm\_3router. Arrancar el cliente DHCP en vm\_1 con dhclient -d enp0s3 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* | *discover* | *Dhcp message type*  *Requested ip address*  *Parameter request list* |
| *192.168.0.18* | *192.168.0.50* | *offer* | *Dhcp message type*  *Dhcp server identifier*  *Ip address lease time*  *Subnet mask*  *Broadcast address*  *router* |
| *0.0.0.0* | *255.255.255.255* | *Request* | *Dhcp message type*  *Dhcp server identifier*  *Request ip address*  *Parameter request list* |
| *192.168.0.18* | *192.168.0.50* | *ack* | *Dhcp message type*  *Dhcp server identifier*  *Ip address lease time*  *Subnet mask*  *Broadcast address*  *router* |

| **Para saber más …** |
| --- |
| Durante el arranque del sistema se pueden configurar automáticamente interfaces según la información almacenada en el disco del servidor (configuración persistente). De esta forma, no es necesario tener que configurar las interfaces cada vez que se reinicie la máquina. Los ficheros en la ruta /etc/systemd/network/\* indican cómo se configuran estas interfaces.  Por ejemplo, si se quiere configurar una interfaz de forma automática usando DHCP, se puede crear el fichero /etc/systemd/network/enp0s3.network con el siguiente contenido:   |  | | --- | | [Match] Name=enp0s3  [Network] DHCP=yes |   Por el contrario, si se quiere asignar una configuración estática, se pueden utilizar las siguientes opciones:   |  | | --- | | [Match] Name=enp0s3  [Network] Address=*<dirección IP estática en formato CIDR>* Gateway=*<dirección IP estática del encaminador por defecto (si existe)>* DNS=*<dirección IP estática del servidor DNS (si existe)>* |   Estas opciones se describen en detalle en la página del manual systemd.network(5). |