

# Práctica 1.4: Protocolo IPv6

## Objetivos

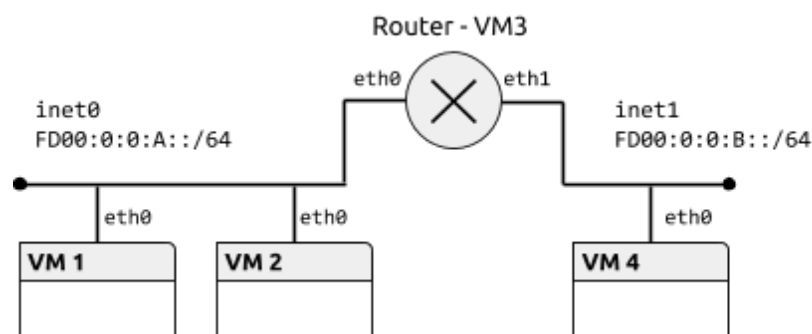
En esta práctica se estudian los aspectos básicos del protocolo IPv6, el manejo de los diferentes tipos de direcciones y mecanismos de configuración. Además se analizarán las características más importantes del protocolo ICMP versión 6.

## Contenidos

- Preparación del entorno para la práctica
- Dirección de enlace local
- Dirección ULA
- Encaminamiento estático
- Configuración persistente
- Autoconfiguración. Anuncio de prefijos
- ICMPv6

## Preparación del entorno para la práctica

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



El fichero de configuración de la topología tendría el siguiente contenido:

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

## Dirección de enlace local

Una dirección de enlace local es únicamente válida en la subred que está definida. Ningún encaminador dará salida a un datagrama con una dirección de enlace local como destino. El prefijo de formato para estas direcciones es fe80::/10.

**Ejercicio 1 [VM1,VM2].** Activar el interfaz eth0 en VM1 y VM2. Comprobar las direcciones de enlace local que tienen asignadas con el comando ip.

**Ejercicio 2 [VM1,VM2].** Comprobar la conectividad entre VM1 y VM2 con la orden ping6. Cuando se usan direcciones de enlace local, y **sólo en ese caso**, es necesario especificar el interfaz origen, añadiendo %<nombre\_interfaz> a la dirección. Consultar las opciones del comando ping6 en la

página de manual. Observar el tráfico que generado con la herramienta wireshark , especialmente los protocolos encapsulados en cada datagrama y los parámetros del protocolo IPv6.

**Para saber más...** En el protocolo IPv4 también se reserva un bloque de direcciones (169.254.1.0-169.254.254.255) para direccionamiento de enlace local, cuando no es posible la configuración de los interfaces por otras vías. Los detalles se describen en el RFC 3927.

## Dirección ULA

Una dirección ULA (*Unique Local Address*) puede usarse dentro de una organización, de forma que los encaminadores internos del sitio deben encaminar los datagramas con una dirección ULA como destino. El prefijo de formato para estas direcciones es fc00::/7.

**Ejercicio 1 [VM1,VM2].** Configurar VM1 y VM2 para que tengan una dirección ULA en la red fd00:0:0:a::/64 con el comando ip. La parte de identificador de interfaz puede elegirse libremente, siempre que no coincida para ambas máquinas. **Nota:** Incluir la longitud del prefijo al fijar las direcciones.

**Ejercicio 2 [VM1,VM2].** Comprobar la conectividad entre VM1 y VM2 con la orden ping6 usando la nueva dirección. Observar los mensajes intercambiados con wireshark.

**Ejercicio 3 [Router,VM4].** Activar el interfaz eth0 de VM4 y los dos interfaces de Router. Comprobar la conectividad entre Router y VM1, y entre Router y VM4 usando la dirección de enlace local. **Nota:** Recordar la necesidad en este caso de especificar el interfaz origen.

**Ejercicio 4 [Router,VM4].** Configurar direcciones ULA en los dos interfaces de Router (redes fd00:0:0:a::/64 y fd00:0:0:b::/64) y en el de VM4 (red fd00:0:0:b::/64). Elegir el identificador de interfaz de forma que no coincida dentro de la misma red.

**Ejercicio 5.** Comprobar la conectividad entre Router y VM1, y entre Router y VM4 usando direcciones ULA. Comprobar además que VM1 no puede alcanzar a VM4.

## Encaminamiento estático

Según la topología que hemos configurado en esta práctica, Router debe encaminar el tráfico entre las redes fd00:0:0:a::/64 y fd00:0:0:b::/64. En esta sección vamos a configurar un encaminamiento estático basado en las rutas que fijaremos manualmente en todas las máquinas.

**Ejercicio 1 [VM1, Router].** Consultar las tablas de rutas en VM1 y Router con el comando ip route. Consultar la página de manual del comando para seleccionar las rutas IPv6.

**Ejercicio 2 [Router].** Para que Router actúe efectivamente como encaminador, hay que activar el reenvío de paquetes (*packet forwarding*). De forma temporal, se puede activar con el comando sysctl -w net.ipv6.conf.all.forwarding=1.

**Ejercicio 3 [VM1,VM2,VM4].** Finalmente, hay que configurar la tabla de rutas en las máquinas virtuales. Añadir la dirección correspondiente de Router como ruta por defecto con el comando ip route. Comprobar la conectividad entre VM1 y VM4 usando el comando ping6.

**Ejercicio 4 [VM1,Router,VM4].** Borrar la *cache* de vecinos en VM1 y Router (con ip neigh flush dev eth0). Con ayuda de la herramienta wireshark, completar la siguiente tabla al usar la orden ping6 entre VM1 y VM4 con todos los mensajes hasta el primer ICMP Echo Reply:

#### Red fd00:0:0:a::/64 - VM1

MAC Origen	MAC Destino	IPv6 Origen	IPv6 Destino	ICMPv6 Tipo

#### Red fd00:0:0:b::/64 - VM4

MAC Origen	MAC Destino	IPv6 Origen	IPv6 Destino	ICMPv6 Tipo

## Configuración persistente

Las configuraciones realizadas en los apartados anteriores son volátiles y desaparecen cuando se reinician las máquinas. Durante el arranque del sistema se pueden configurar automáticamente determinados interfaces según la información almacenada en el disco.

**Ejercicio 1 [Router].** Crear los ficheros ifcfg-eth0 e ifcfg-eth1 en el directorio /etc/sysconfig/network-scripts/ con la configuración de cada interfaz. Usar las siguientes opciones:

Para IPv6:	Para IPv4:
Type=Ethernet BOOTPROTO=static IPv6ADDR=<dirección IP en formato CIDR> IPv6_DEFAULTGW=<dirección IP del encaminador por defecto (si existe)> DEVICE=<nombre del interfaz (eth0...)>	Type=Ethernet BOOTPROTO=static IPADDR=<dirección IP en formato CIDR> GATEWAY=<dirección IP del encaminador por defecto (si existe)> DEVICE=<nombre del interfaz (eth0...)>

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

**Ejercicio 2 [Router].** Comprobar la configuración automática con las órdenes ifup e ifdown.

## Autoconfiguración. Anuncio de prefijos

El protocolo de descubrimiento de vecinos se usa también para la autoconfiguración de los interfaces de red. Cuando se activa un interfaz, se envía un mensaje de descubrimiento de encaminadores. Los encaminadores presentes responden con un anuncio que contiene, entre otros, el prefijo de la red.

**Ejercicio 1 [VM1, VM2, VM4].** Eliminar las direcciones ULA de los interfaces (con `ip addr del`) y desactivarlos (con `ip link set eth0 down`).

**Ejercicio 2 [ Router].** Configurar el servicio zebra para que el encaminador anuncie prefijos. Para ello, crear el archivo `/etc/quagga/zebra.conf` e incluir la información de los prefijos para las dos redes. Cada entrada será de la forma:

```
interface eth0
no ipv6 nd suppress-ra
ipv6 nd prefix fd00:0:0:a::/64
```

Finalmente, arrancar el servicio con el comando `service zebra start`.

**Nota:** En `/usr/share/doc/quagga-0.99.22.4` hay archivos de configuración de ejemplo.

**Ejercicio 3 [VM4].** Comprobar la autoconfiguración del interfaz de red en VM4, activando el interfaz y consultando la dirección asignada.

**Ejercicio 4 [VM1, VM2].** Estudiar los mensajes del protocolo de descubrimiento de vecinos:

- Activar el interfaz en VM2, comprobar que está configurado correctamente e iniciar una captura de tráfico con `wireshark`.
- Activar el interfaz en VM1 y estudiar los mensajes ICMP de tipo Router Solicitation y Router Advertisement.
- Comprobar las direcciones destino y origen de los datagramas, así como las direcciones destino y origen de la trama Ethernet. Especialmente la relación entre las direcciones IP y MAC. Estudiar la salida del comando `ip maddr`.

**Para saber más...** En el proceso de autoconfiguración se genera también el identificador de interfaz según el "extended unique identifier" (EUI-64) que se describe en el RFC 4193. La configuración del protocolo de anuncio de encaminadores tiene múltiples opciones que se pueden consultar en la documentación de zebra (ej. intervalo entre anuncios no solicitados). Cuando sólo se necesita un servicio que implemente el anuncio de prefijos, y no algoritmos de encaminamiento para el router, se puede usar el proyecto de código libre "*Router Advertisement Daemon*", `radvd`.

**Ejercicio 5 [VM1].** La generación del identificador de interfaz mediante EUI-64 supone un problema de privacidad para las máquinas clientes, que pueden ser rastreadas por su dirección MAC. En estos casos, es conveniente activar las extensiones de privacidad para generar un identificador de interfaz pseudoaleatorio temporal para las direcciones globales. Activar las extensiones de privacidad en VM1 con `sysctl -w net.ipv6.conf.eth0.use_tempaddr=2`.

## ICMPv6

El protocolo ICMPv6 permite el intercambio de mensajes para el control de la red, tanto para la detección de errores como para la consulta de la configuración de ésta. Durante el desarrollo de la práctica hemos visto los más importantes.

**Ejercicio 1.** Generar mensajes de los siguientes tipos en la red y estudiarlos con ayuda de la herramienta `wireshark`:

- Solicitud y respuesta de eco.
- Solicitud y anuncio de encaminador.
- Solicitud y anuncio de vecino.
- Destino inalcanzable - Sin ruta al destino (Code: 0).



