

Práctica 1.4: Protocolo IPv6

Objetivos

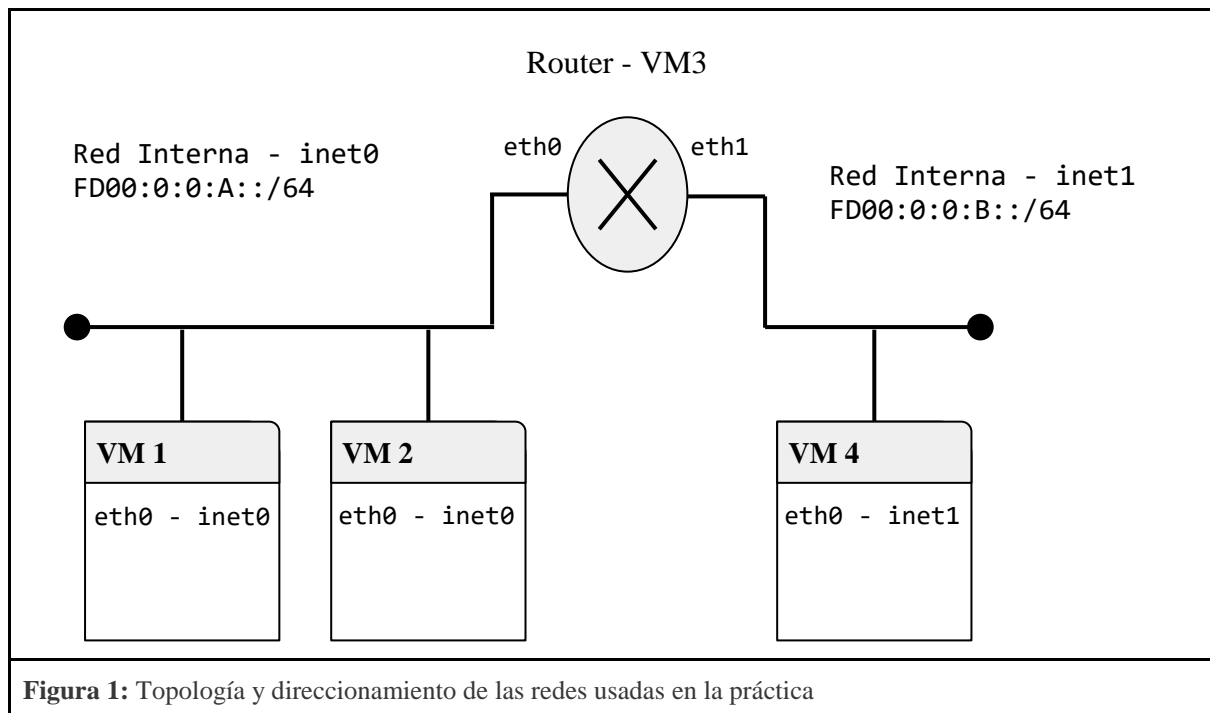
En esta práctica se estudian los aspectos básicos del protocolo IPv6, el manejo de los diferentes tipos de direcciones y mecanismos de configuración. Además se analizarán las características más importantes del protocolo ICMP versión 6.

Contenidos

- Preparación del entorno para la práctica
- Dirección de enlace local
- Dirección ULA
- Encaminamiento estático
- Configuración persistente
- Autoconfiguración. Anuncio de Prefijos
- ICMP versión 6

Preparación del entorno para la práctica

Configuraremos la topología de red que se muestra en la Figura 1.



El siguiente fichero muestra el archivo de configuración de la topología:

```
$ cat pr4.topol
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 y comprobar las direcciones de enlace local que tienen asignadas. La consulta de las direcciones del interfaz debe realizarse con el comando `ip. -I`

Consulta: `ip addr`

MV1	<code>ip link set eth0 up</code>
MV2	<code>ip link set eth0 up</code>

Ejercicio 2 [VM1,VM2]. Comprobar la conectividad entre VM1 y VM2 con la orden `ping6`. Cuando se usan direcciones de enlace local es necesario especificar el interfaz origen, mediante la opción `-I` o añadiendo `%<nombre_interfaz>` a la dirección. Consultar las opciones del comando `ping6` en la página de manual.

`ping6 <direccion ipv6>%eth<N>`

Ejercicio 3 [VM1,VM2]. Arrancar la herramienta `wireshark` y observar el tráfico que genera la orden `ping6`, 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 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`. El comando que se utilizará es `ip addr`. 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.

`ip -6 addr add <direccion ipv6> dev eth<N>`

MV1	<code>ip -6 addr add fd00:0:0:a::1/64 dev eth0</code>
MV2	<code>ip -6 addr add fd00:0:0:a::2/64 dev eth0</code>

Ejercicio 2 [VM1,VM2]. Comprobar la conectividad entre VM1 y VM2 con la orden `ping6` usando la nueva dirección. Observar el contenido de las tramas intercambiadas con `wireshark`.

`ping6 <direccion ipv6>`

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.

```
ping6 fe80:ff:fe00:100%eth0
```

```
ping6 fe80:ff:fe00:400%eth1
```

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 (fd00:0:0:b::/64). Igual que en el apartado anterior, elegir la parte de identificador de interfaz de forma que no coincida dentro de la misma red.

```
ip -6 addr add <direccion ipv6> dev eth<N>
```

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.

```
ping6 fd00:0:0:a::1
```

```
ping6 fd00:0:0:b::1
```

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 virtuales.

Ejercicio 1 [VM1, Router]. Consultar las tablas de rutas en VM1 y Router con los comandos `route` e `ip route`. Consultar las páginas de manual de ambos comandos para seleccionar las rutas IPv6.

Ejercicio 2 [Router]. Para que Router actúe efectivamente como encaminador, hay que activar el reenvío de tráfico IP (*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 mediante el comando `ip route` la dirección correspondiente de Router como ruta por defecto. Comprobar la conectividad entre VM1 y VM4 usando el comando `ping6`.

```
ip -6 route add default via <direccionIPGlobalRouter>
```

Consulta: `route -6` ó `ip -6 route`

Ejercicio 4. Usar la orden `ping6` entre VM1 y VM4. Con ayuda de la herramienta `wireshark`, completar la siguiente tabla:

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

MAC Origen	MAC Destino	IPv6 Origen	IPv6 Destino	ICMPv6 Tipo
02:00:00:00:01:00	IPv6mcast_ff:00:00:03	fd00:00:0:a::1	ff02::1:ff00:3	Neighbor solicitation
02:00:00:00:03:00	02:00:00:00:01:00	fd00:00:0:a::3	fd00:00:0:a::1	Neighbor advertisement
02:00:00:00:01:00	02:00:00:00:03:00	fd00:00:0:a::1	fd00:00:0:b::1	Echo request
02:00:00:00:03:00	02:00:00:00:01:00	fd00:00:b:a::1	fd00:00:0:a::1	Echo reply

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

MAC Origen	MAC Destino	IPv6 Origen	IPv6 Destino	ICMPv6 Tipo
02:00:00:00:04:00	02:00:00:00:03:01	fd00:0:0:b::1	fd00:0:0:a::1	Echo request
02:00:00:00:03:01	02:00:00:00:04:00	fd00:0:0:a::1	fd00:0:0:b::1	Echo reply
02:00:00:00:03:01	02:00:00:00:04:00	fe80:ff:fe00:301	fe80:ff:fe00:400	Neighbour solicitation
02:00:00:00:04:00	02:00:00:00:03:01	fe80:ff:fe00:400	fe80:ff:fe00:301	Neighbour advertisement

Configuración Persistente

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

Ejercicio 1 [Router]. Añadir al fichero `/etc/network/interfaces` dos entradas de tipo *static* (iface eth0 inet6 static) con la configuración de los dos interfaces usando las opciones `address` y `netmask`. Consultar la página de manual (`man interfaces`).

```
GNU nano 2.2.4      Fichero: interfaces      Modificado
# This file describes the network interfaces available on your syst$
# and how to activate them. For more information, see interfaces(5).

# The loopback network interface
auto lo
iface lo inet loopback
iface eth0 inet6 static
address fd00:0:0:a::3 netmask 64
iface eth1 inet6 static
address fd00:0:0:b::3 netmask 64

^G Ver ayud^O Guardar ^R Leer Fic^Y Pág Ant ^K CortarTx^C Pos actual
^X Salir   ^J Justific^W Buscar ^V Pág Sig ^U PegarTxt^T Ortografía
```

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

En caso de que no funcione (ni con `up` ni con `down`) hay que dar de baja manualmente las interfaces (`ip link set down dev eth<N>`), poner `netmask` en otra línea abajo.

`ifup <ethN>`

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 (`ip addr del`) y desactivarlos (ej. `ip link set eth0 down`).

`Ip -6 addr del <dir ipv6global> dev eth<N>`

Ejercicio 2 [Router]. Configurar el demonio zebra para que que el encaminador anuncie prefijos:

- Activar el demonio zebra (`zebra=yes`) en `/etc/quagga/daemons`.
- Incluir la información de los prefijos para las dos redes en el archivo `/etc/quagga/zebra.conf`. Cada entrada será de la forma:

Listado 1: Archivo de configuración `/etc/quagga/zebra.conf`, para anuncio de prefijos

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



```
GNU nano 2.2.4          Fichero: zebra.conf          Modificado

interface eth0
  no ipv6 nd suppress-ra
  ipv6 nd prefix fd00:0:0:a::/64
interface eth1
  no ipv6 nd suppress-ra
  ipv6 nd prefix fd00:0:0:b::/64
```

Finalmente arrancar el demonio con el comando `service quagga start`.

Nota: En `/usr/share/doc/quagga/examples` hay archivos de ejemplo para la configuración de quagga. Puede usarse como referencia `zebra.conf.sample`.

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

Ejercicio 4 [VM1 y VM2]. Estudiar los mensajes correspondientes al 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 del ID de grupo multicast en 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 “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 el 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, que consisten en 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`.

ICMP versión 6

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.

(No hacer)

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).