Diseño y Administración de Redes

Práctica 2

Diseño y gestión de escenarios IPv6

Dpto. Ingeniería Electrónica y Comunicaciones
Área de Ingeniería Telemática

Departamento de
Ingeniería Electrónica
y Comunicaciones
Universidad Zaragoza

Autores:
Profesores del área de Ingeniería Telemática

1. Introducción

1.1. Objetivos

Tras la realización de esta práctica, el alumno deberá ser capaz de:

Configurar un escenario de interconexión de redes utilizando el protocolo IPv6 garantizando la conectividad tanto interna como externa así como la interoperabilidad con el protocolo IPv4.

Analizar y evaluar el comportamiento de un escenario IPv6:
Emplear adecuadamente todas las herramientas de verificación necesarias.
Detectar posibles errores de configuración, predecir resultados.

1.2. Contenidos

Los objetivos propuestos en esta práctica pretenden afianzar y complementar los **contenidos teóricos vistos en clase**. Por lo tanto, será necesario un estudio previo de los mismos así como la utilización de los apuntes de clase (y cualquier material adicional que el alumno considere oportuno) como apoyo a la realización práctica.

A modo de orientación se enumeran aquellos aspectos de IPv6 que se consideran más relevantes:

Introducción a IPv6
Direccionamiento
PDU. Cabeceras de extensión
Funciones de control
Autoconfiguración
Encaminamiento
Coexistencia / Transición IPv4-IPv6

Pueden resultar de utilidad los documentos RFC relativos a los distintos aspectos a analizar: http://tools.ietf.org/rfc/index

Del mismo modo se recomienda, en caso necesario, consultar cualquier ayuda online, así como 'man' de Linux, o 'help' de MS-DOS.

Además, en los anexos se encuentra disponible información auxiliar que será necesaria durante la realización práctica:

Anexo I: Manuales de configuración y programas de análisis.
Anexo II: Entregable de evaluación (estudio previo, desarrollo y cuestiones)

1.3. Equipos, tecnologías y herramientas

La práctica propuesta consiste en la configuración, verificación y análisis de un escenario de interconexión de redes IPv6 a través de un túnel IPv6/IPv4. Para ello, se contará con los siguientes equipos y tecnologías de interconexión:

Equipos: máquinas reales con sistema operativo Windows XP en los que se ejecutará una máquina virtual mediante virtualbox. Las máquinas virtuales

tendrán sistema operativo Linux (CentOS 6) y podrán actuar como *host* o como *router*.

Tecnología de conexión: será Ethernet mediante tarjetas internas con una velocidad de 10/100 Mbps. Cada *router* consta de dos tarjetas, identificadas en Linux como eth0 y eth1, correspondientes respectivamente a las tarjetas superior e inferior. En los host se utilizará únicamente la tarjeta superior.

En cuanto a las **herramientas** necesarias para la verificación y el análisis de los escenarios, utilizaremos el software de captura ***tcpdump*** y el analizador de protocolos ***Wireshark***.

1.4. Escenarios

A continuación se muestra la configuración de los escenarios que serán la base de trabajo a lo largo de toda la práctica.

La figura 1 muestra el escenario con el que se trabajará en la práctica. En este caso los host de las redes LAN A y LAN B utilizarán IPv6 nativo. La conexión con el exterior se realizará a través de la propia red pública del laboratorio (155.210.157.0/24).

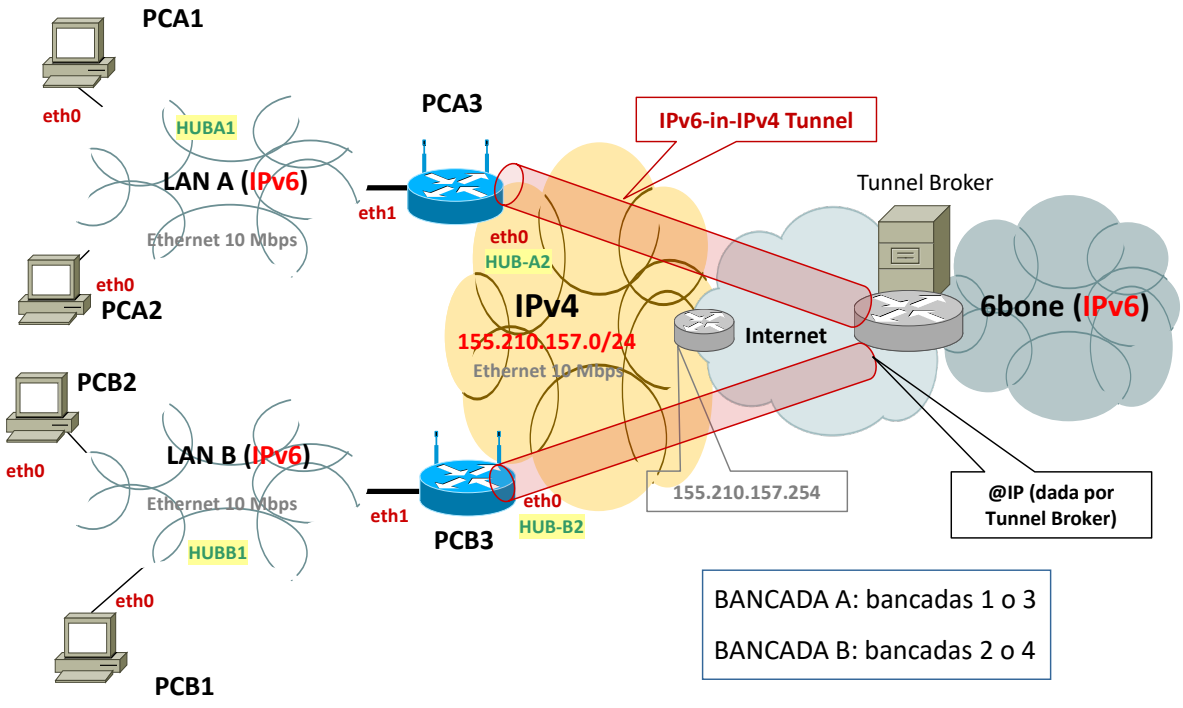


Figura 1: Escenario de interconexión de redes IPv6. Acceso al 6bone a través de IPv4.

2. Realización práctica en el laboratorio

En previsión de la falta de tiempo, este capítulo 2 no se ejecuta tal y como aparece, sino que se harán siguiendo las modificaciones expuestas en el capítulo 3. Sin embargo servirá como base para entender mejor un escenario real.

Como se ha comentado anteriormente, para la realización de esta práctica se utilizarán equipos reales con SO Windows XP y máquinas virtuales con SO Linux. Como cada grupo trabajará con unas máquinas virtuales propias será necesario utilizar la siguiente información: CentOS_DAR_gX, siendo X el número de grupo de prácticas.

**** El entregable sirve de guía para el seguimiento de la práctica. Completar todas las cuestiones implica alcanzar todos los resultados de aprendizaje requeridos ****

**** La evaluación al final de la práctica supondrá la discusión con el profesor de los apartados propuestos y la entrega de cuestiones del entregable ****

**** Las cuestiones marcadas como E se resolverán antes del comienzo de la práctica y se entregarán individualmente ****

**** Comentar con el profesor o contestar en el entregable, según se indique, las preguntas marcadas como P. Se entregarán por grupos ****

**** Habrá cuestiones marcadas como E y P simultáneamente. Esto significa que hay que resolverlas antes de la práctica y también comprobar el grado de acierto en el transcurso de la misma. La E se entrega individualmente y la P por grupos ****

2.1. Utilización de IPv6. Reconfiguración del escenario para acceder al 6bone

2.1.1. Configuración básica

Considerar el **escenario representado en la figura 1**. Este escenario ejemplifica de un modo simplificado un supuesto de dos departamentos configurados con direccionamiento IPv6 (LAN A y LAN B) que pueden tanto conectarse entre sí como conectarse con el *backbone* IPv6 a través de un túnel IPv6/IPv4.

Será necesario verificar el conexionado físico de modo que las redes LAN que aparecen en el escenario sean totalmente independientes (*hub* segmentados adecuadamente y desconectados de los *switch*). Para ello se seguirán las instrucciones del profesor.

En todos los PC debemos quitar las configuraciones de las prácticas anteriores.

Service zebra stop
Service ripd stop
ifdown eth0
ifup eth0

EN PC3 debemos configurar el direccionamiento IPv4 público y el router por defecto indicado en la figura 1.

Para garantizar el funcionamiento del escenario de conectividad propuesto, será necesario configurar adecuadamente todas las interfaces necesarias. Es importante observar la funcionalidad distinta que requieren los equipos terminales (*host*) y los *router*. Para diferenciar correctamente dicho comportamiento es necesario tener en cuenta:

- En los ***host***, las interfaces IPv6 **se autoconfigurarán automáticamente** de acuerdo a los mensajes recibidos de los *router* adecuados.
- En IPv6, un *router* no sólo realiza el encaminamiento sino que es responsable del envío periódico de mensajes de anuncio (*Router Advertisements*) donde se comunican los prefijos de red, el MTU, etc. facilitando así la autoconfiguración de los *host* conectados a su interfaz. No es necesario que todos los *router* de una red anuncien prefijos en la interfaz. Linux proporciona esta funcionalidad mediante la ejecución del *daemon radvd* (configuración de acuerdo al Anexo I). Dicho servicio únicamente debe ejecutarse en los *router* que requieran enviar anuncios para permitir la autoconfiguración de *host*.
- **Los *router* no se autoconfiguran** atendiendo a anuncios de otros *router*[1], por lo que será necesario incluir la información de encaminamiento manualmente. Su función de *router* sólo requiere la utilización de direcciones locales de enlace (FE80::/10). No obstante, si se quiere tener conectividad con ellos desde cualquier punto, se necesitará la configuración manual de direcciones globales (usando el prefijo correspondiente de la red a la que se conecten las interfaces).
- Los *router* constan de diversas interfaces de conexión a las distintas redes. Para posibilitar la funcionalidad de encaminamiento es necesario que estos equipos sean capaces de reenviar la información recibida de una interfaz a aquélla que sea necesario (de acuerdo a su tabla de rutas). **Al igual que en IPv4, es necesario activar el reenvío en IPv6**. La configuración de dicho reenvío (***forwarding***) exige la modificación de parámetros internos a los que podemos tener acceso mediante la función ***sysctl*** (ver Anexo II, apartado 6).

La configuración de los túneles permitirá el acceso al *6bone* al proporcionar un direccionamiento global (enrutable públicamente) para nuestras redes así como el encapsulado de la información en el protocolo IPv4. El acceso al exterior, por lo tanto, se realizará gracias a la conectividad IPv4 de la red del laboratorio (155.210.157.0/24). Los túneles deberán configurarse mediante la herramienta ***Tunnel Broker***, como se explica más adelante, gracias a la cual, un proveedor de servicios externo nos proporcionará los prefijos de red necesarios para realizar nuestra configuración.

La correcta configuración del escenario se realizará mediante los comandos referidos en el Anexo I, siguiendo los pasos indicados a continuación. Igualmente, se pueden

[1] Cuando un *router* tenga parado el *daemon radvd*, si otro *router* en su red lo tiene en marcha, puede responder como un *host* autoconfigurando una dirección global. Esto no es un problema, simplemente será un efecto que podrá apreciarse en algún paso de la configuración.

consultar todas aquellas páginas de ayuda que se considere oportuno. Posibles aspectos sin especificar se concretarán durante la misma sesión práctica.

2.1.1.1. Conectividad con el *6bone*: *Tunnel Broker*

Teniendo en cuenta que los equipos son duales y la conexión con equipos remotos IPv6 sería técnicamente posible, se propone la utilización de la herramienta *Tunnel Broker* para conseguir la conectividad necesaria con el *6bone* y comprobar el posible acceso a sitios IPv6. Se aprovechará dicha conectividad para comunicar entre sí las redes LAN, que constituyen redes IPv6 aisladas.

Así pues, para configurar cada subred IPv6, se partirá, en cada LAN de un prefijo global de red proporcionado por el ISP al configurar el túnel. La configuración se realizará desde el *router* con conectividad al exterior. A partir del **prefijo de red**, de **64 bits,** los *host* que se autoconfiguren autoasignarán el identificador de interfaz de 64 bits a partir de su dirección MAC (EUI-64)

La herramienta *Tunnel Broker* propuesta es accesible mediante el siguiente enlace:

http://tunnelbroker.net/

Para poder realizar la configuración se hará uso del **perfil de usuario ya creado** en dicho servidor, facilitado por el profesor en la sesión práctica. **Los túneles involucrados en el escenario ya se encuentran creados**, por lo que será necesario identificar cuál de los cuatro túneles existentes es el que corresponde a cada *router* (PC13, PC23, PC33 y PC43).

**** IMPORTANTE ****
**** No hay que crear ningún túnel nuevo, solo identificar los parámetros del túnel correspondiente para configurar el escenario ****

E1. Anotar la siguiente información correspondiente al *router* de nuestro escenario:

- Direcciones IPv4 de los extremos del túnel (local–router y remoto–servidor).
- Direcciones IPv6 de los extremos del túnel (local–router y remoto–servidor).
- Prefijo /64 asignado por el proveedor (*Routed IPv6 Prefixes*), para utilizarlo posteriormente en la red LAN correspondiente.
- Direcciones IPv6 de servidores DNS proporcionadas por el proveedor (*Available DNS resolvers*) para su posterior utilización.

Para luego configurar correctamente nuestro extremo del túnel, ejecutar el ejemplo de configuración proporcionado por el proveedor:

Example IPv6 Tunnel Configurations by OS (Windows, Linux, etc.):

En el desplegable, seleccionar **Linux-route2**

E2. Anotar el ejemplo de configuración

Dicho ejemplo será análogo al que se presenta a continuación (los detalles de los comandos utilizados pueden encontrarse en el Anexo I):

modprobe ipv6

Verifica la instalación de ipv6 (no es necesario en nuestro caso)

ip tunnel add he-ipv6 mode sit remote 216.66.84.42 local 155.210.157.25 ttl 255

Añade una interfaz tipo túnel llamada he-ipv6

Dicho túnel se corresponde con los extremos IPv4:

- local (*client IPv4 address*): 155.210.157.25
- remote (*server IPv4 address*): 216.66.84.42

ip link set he-ipv6 up

Activa la interfaz creada

ip addr add 2001:470:1f12:57d::2/64 dev he-ipv6

Asigna a dicha interfaz la dirección IPv6 dada por el proveedor (*client IPv6 address*). Esta dirección será la que tenga el router para comunicarse con el mundo IPv6.

ip route add ::/0 dev he-ipv6

Configura el túnel como ruta por defecto para los paquetes IPv6

ip -f inet6 addr

Fuerza la utilización de IPv6 para los comandos 'ip' (en nuestro caso no será necesario, porque se utilizará directamente ip -6 [- opciones adicionales])

**** IMPORTANTE ****
**** Comentar con el profesor cualquier duda sobre estos comandos ****

Una vez configurado el túnel, habrá que verificar la conectividad con el exterior. Esta configuración únicamente garantiza nuestro acceso al *6bone* desde los *router* donde hemos configurado los túneles. La configuración de las subredes LAN IPv6 se realizará en el apartado siguiente.

P1. Una vez establecido el túnel, conectarse desde el *router* de salida de la red LAN correspondiente (A o B) a cualquier sitio con conectividad IPv6[2]. Por ejemplo, accede a la siguiente página y **apunta la dirección IPv6** (que usaremos más adelante) de la misma: http://www.consulintel.es
Capturar en la interfaz eth0 del *router* (no en la interfaz túnel creada) y verificar que, efectivamente, el tráfico desde el *router* hacia el exterior es IPv6 sobre IPv4. Muéstralo indicando las direcciones IPv6 e IPv4 que aparecen en los paquetes.

P2. Observa el valor MTU configurado en la interfaz tipo túnel (***ip -6 link show dev [nombre_tunel] / ifconfig [nombre_tunel]***). Justifica dicho valor teniendo en cuenta la información capturada previamente.

[2] Se pueden buscar en Internet, *IPv6 sites*

2.1.1.2. Configuración particular de la conectividad IPv6.

Mediante los comandos de configuración apropiados, como se indica en el Anexo I, realizar en cada red LAN la configuración de los distintos equipos:

1) Configuración del *router*

- **Habilitar el reenvío** en el *router* **para IPv6**.
  - Comando ***sysctl*** (Anexo II, apartado 6)
- **Deshabilitar el firewall IPv6**
  - **Ip6tables -F**
- Establecer manualmente la dirección IPv6 del interfaz interno.
- Establecer manualmente las **rutas estáticas** que garantizan el encaminamiento necesario (Comando ***ip -6 route add***… (Anexo I, apartado 3))
  - Ruta directa hacia la propia LAN IPv6 (recordar que el *router* no se autoconfigura solo). Se realiza directamente a través de una interfaz (*dev*).
  - Encaminamiento indirecto: rutas hacia redes remotas. Se realiza a través de una interfaz específica (*dev*) y un *router* como siguiente salto (*gateway*):
    - Observar en la figura 1.2 que únicamente se requiere una ruta por defecto para acceder a cualquier destino IPv6 (6bone o la red LAN remota). Dicha entrada (destino ::/0) ya se habrá configurado a través del túnel cuando se ha creado éste.
- Configurar y activar el *daemon* de anuncios de *router* en los *router* apropiados – PCA3 o PCB3 – (Anexo II, apartado 7.1.):
  - Configurar el archivo **/etc/radvd.conf**: anuncio de los prefijos en las redes LAN para permitir que los *host* se configuren. Seguir el patrón dado en el archivo ya existente en el equipo y las indicaciones del proveedor.
  - Activar el servicio ***radvd***.

2) Configuración del *host*

Para emular un escenario IPv6 nativo:

- Borrar todas las direcciones IPv4 para emular un escenario IPv6 nativo en las redes LAN. Utilizar los siguientes comandos:
  - **ip -4 addr show**: visualizar todas las direcciones IPv4 configuradas

  Con la información de redes/prefijos observada, borrar todas:
  - **ip -4 addr del <ipv4address>/<prefixlength> dev <interface>**
- Borrar todas las rutas IPv4 para emular un escenario IPv6 nativo en las redes LAN.
  - **route del -net <@red> netmask <mask> gw <@IPgw>**

Para configurar el direccionamiento/encaminamiento IPv6:

- Los *host* "aprenden" tanto su dirección IPv6 como su *router* por defecto de la información recibida de los *router*.

2.1.2. Análisis

2.1.2.1. Cuestiones preliminares

E3/P3. Una vez configurado el escenario completo, se comprobará la conectividad del mismo verificando la comunicación entre los equipos de las redes LAN A y LAN B mediante **ping6** (consultar las páginas de ayuda necesarias – Anexo II, apartado 8).

E4/P4. Sin modificar el fichero resolv.conf ¿Se puede acceder a sitios IPv6 desde los *host* de las redes LAN, por ejemplo www.consulintel.com? Prueba a realizar nuevamente la conexión, utilizando directamente la dirección IPv6 de www.consulintel.com (http://[<dir.IPv6>]:80). ¿Se puede ahora acceder? Da una explicación de por qué funciona en el *router* y no lo hace en el *host*.

Si da problemas el navegador, ejecutar dig + trace o ejecutar wget por comandos.

a) Los *router* de salida no tenían problemas para acceder. Recuerda que los *host* son IPv6 nativos mientras que los *router* duales tienen conexión IPv4 e IPv6. Observa el contenido del fichero /etc/resolv.conf y explica lo que ha sucedido.
b) Teniendo en cuenta la información proporcionada por el proveedor del túnel (apuntada previamente), modifica el contenido del fichero de resolv.conf para que funcione correctamente el acceso web.

P5. Observa las direcciones IPv6 existentes en las interfaces de los PCs y pon un ejemplo de dirección local de enlace y dirección global. Muestra, en una de ellas, cómo se ha obtenido el identificador de interfaz de 64 bits (EUI-64).

2.1.2.2. Autoconfiguración

A continuación se va a analizar la **facilidad** de autoconfiguración del protocolo IPv6. Para ello, se necesita capturar la información generada entre el *router* y el PC en el momento en el que éste activa la interfaz. Para poder visualizar este evento será necesario:

1. Capturar en la interfaz eth1 de PCA3/PCB3 (*router*). Aplicar un filtro para eliminar la información UDP que aparezca facilitando la observación sólo de los paquetes IPv6 relativos al proceso de autoconfiguración.
2. Desactivar la interfaz de uno de los dos PCs de la red LAN correspondiente (**ip -6 link set eth0 down**)
3. Reactivar dicha interfaz (**ip -6 link set eth0 up**)
4. Observar lo capturado.

Durante el proceso de autoconfiguración, se realiza el mecanismo de detección de direcciones duplicadas (DAD, procedimiento equivalente a ARP gratuito en IPv4). Su finalidad es verificar que la dirección elegida es única (tanto la local de enlace como la global).

P6. A partir de la captura indica los dos casos de procedimiento DAD descritos especificando en ambos: dirección *multicast* (*Solicited Node Address*) a la que se dirige el mensaje de ND (*Neighbor Discovery*, ICMPv6) y dirección *unicast* por la que se pregunta (*target*)

Verifica la correspondencia entre las direcciones *multicast* y *unicast* identificadas.

Muestra los paquetes de petición y respuesta (por parte del equipo y el *router*) de los parámetros necesarios para la autoconfiguración. Resalta dichos parámetros en el mensaje correspondiente.

2.1.2.3. Resolución de direcciones

En IPv6, el protocolo *Neighbor Discovery* (ND) de ICMPv6, entre otras funciones, realiza la resolución de direcciones equivalente a ARP en IPv4. La tabla correspondiente (tabla de vecinos) puede visualizarse con el comando **ip -6 neigh show** (detalles, Anexo I, apartado 4).

P7. Realiza las siguientes conexiones, identificando la correspondencia entre dirección MAC destino (dirección de nivel de enlace – *link layer* – lladdr) y dirección IPv6 destino del paquete capturado en el equipo origen del ping:

- ping6 desde PCA1/PCB1 a PCA2/PCB2
- ping6 desde PCA1/PCB1 a PCB1/PCA1

P8. Comprueba las conexiones (mediante varios ping, paso a paso) entre PCA1/PCB1 y PCB1/PCA1, es decir comprobando que funciona el ping a cada una de las direcciones intermedias del camino entre los extremos. Justifica los tiempos que tarda cada uno de los ping en función de la posición del destino.

2.1.2.4. Encaminamiento, fragmentación y reensamblado

En este apartado se requiere la colaboración de las dos bancadas (A y B) para modificar el MTU en diferentes interfaces y ver el efecto sobre el escenario completo.

E9/P9. Modificar el **MTU del interfaz eth1 de PCB3 a un valor de 1300 y el del interfaz eth0 de PCA3 a un valor de 1350**. Observa qué sucede si se realiza un **ping6 desde PCA1 hacia PCB1** con un tamaño de **1400 bytes.** ¿Quién realiza la fragmentación? ¿Qué tramas se intercambian entre los quipos? ¿Qué diferencia habría si la red fuera totalmente IPv4?

Nota: tener en cuenta que en IPv6 se utiliza una caché de destinos que almacena la información de encaminamiento hacia los destinos recientes. Una vez hecha la conexión con un destino, no se consulta la tabla de encaminamiento, sino la tabla de destinos. Las entradas tienen un tiempo de vida (expirado este, se repite el proceso y se crea nuevamente con información actualizada, que puede haber variado) [la caché puede visualizarse mediante el comando ***ip -6 route show table all*** → identificar *cache*]

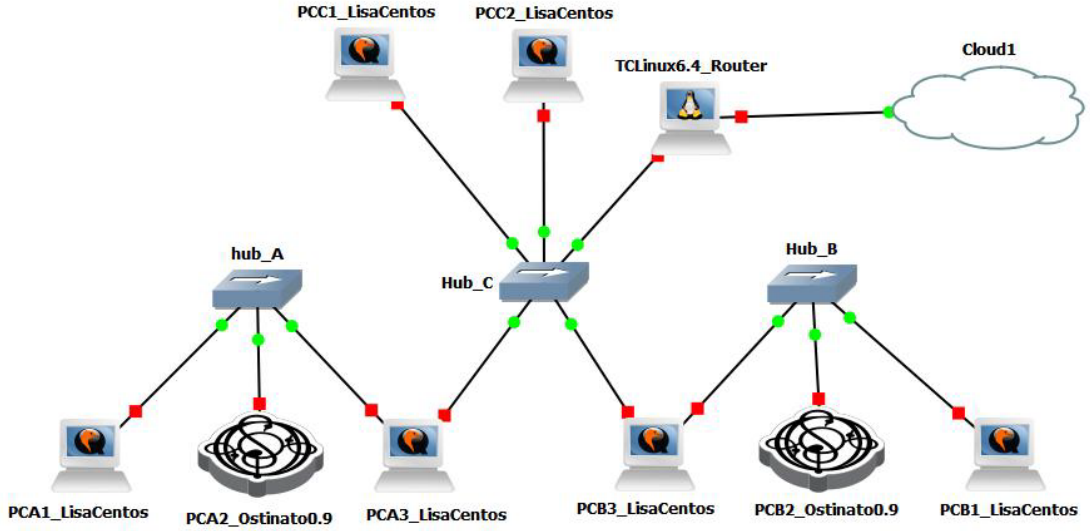
3. Realización práctica en GNS3

Si no podemos acceder presencialmente al laboratorio procedemos a realizar la siguiente práctica en GNS3, con las modificaciones que vienen indicadas a continuación.

**** La evaluación de la práctica requiere la entrega del escenario GNS3 con la configuración adecuada, las capturas correspondientes que avalen su correcto funcionamiento y un documento con la explicación oportuna. ****

**** Para facilitar la elaboración del documento explicativo aparecen una serie de cuestiones a lo largo del enunciado de la práctica. ****

Es este apartado se propone la creación de un escenario similar al expuesto en el apartado 2. El escenario que crearemos será similar al de la siguiente figura



Los PC serán máquinas LisaCentos, (Las mismas utilizadas para las prácticas 1.1 y 1.2 que no tiene entorno gráfico, para que sea escalable).
El túnel IPv6 sobre IPV4 estará configurado entre las máquinas **PCA3 y PCB3** que dispondrán de *dual-stack* IPv4 e IPv6 para la conexión con el *hub_C*. Mientras que las máquinas PCA1, PCA2, PCB1 y PCB2 serán nativas IPv6. Las máquinas PCA3 y PCB3 actuarán como *router* y, por lo tanto, deberán tener también configurado el *demonio radvd* como viene indicado en el anexo I.
Se deben ir contestando las cuestiones que aparecen en el capítulo 2 adaptándolas al nuevo escenario (la comunicación de P1 la sustituimos por un ping6, P4 no hay que responderla, en P8 no hay que responder a la cuestión de justificar los tiempos, el resto es igual) que nos permitirá elaborar el documento explicativo.
En el caso de que necesitemos conexión con el exterior, por ejemplo, para instalar *radvd* (yum *install radvd para Centos) lo* haremos de la misma forma que en las prácticas anteriores, mediante un tiny-core-linux que usaremos como *router* de nuestro escenario y nos permitirá la conexión de la red LAN (intranet) con Internet, mediante NAT.

Anexo I. Configuración de redes en un entorno IPv6:

Así como para configurar los parámetros de red en IPv4 se han utilizado comandos como ifconfig o route (anexo I, Práctica 3(I)), a continuación, se muestra cómo realizar la configuración mediante el **comando ip**, especialmente adecuado para la configuración en IPv6. Dicho comando puede utilizarse igualmente para la configuración de IPv4 (mediante el comando "ip -4" en lugar de "ip -6").

1. Manejo de las interfaces:

Se utilizan interfaces tanto físicas (como eth0, eth1, (Ethernet), ath0 (*wireless*)) como virtuales (como ppp0, tun0, sit0).

Las interfaces tipo túnel (como por ejemplo los túneles necesarios para encapsular IPv6 sobre IPv4, y enviar el tráfico utilizando alguna de las interfaces físicas) tienen un tratamiento especial. Estas interfaces se nombran normalmente mediante **sitx** (sit = *Simple Internet Transition*), donde 'x' puede ser cualquier valor distinto de 0 (reservado)

ip -6 addr show dev <interface>

Muetra los datos de la interfaz. Cabe destacar la identificación del ámbito de las direcciones configuradas (una interfaz, en IPv6, puede tener configuradas varias direcciones).

Ejemplo de configuración estática:

 # ip -6 addr show dev eth0

2: eth0: <BROADCAST,MULTICAST,UP> mtu 1500 qdisc pfifo_ fast qlen 100
inet6 fe80::210:a4ff:fee3:9566/10 **scope link**
inet6 2001:0db8:0:f101::1/64 **scope global**
inet6 fec0:0:0:f101::1/64 **scope site**

ip link set dev <interface> up/down

Activa/desactiva la interfaz

ip link set <interface> mtu <value>

Modificación del tamaño MTU de la interfaz. Se pueden cambiar diversos parámetros característicos de una interfaz física. Para ver los diversos parámetros configurables, consultar el manual de ayuda de Linux.

2. Direccionamiento (configuración estática o manual)

ip -6 addr add <ipv6address>/<prefixlength> dev <interface>
ip -6 addr del <ipv6address>/<prefixlength> dev <interface>

Añadir o borrar una dirección IPv6 de forma manual, en una interfaz (sin proceso de autoconfiguración)

Ejemplo: ip -6 addr add 2001:0db8:0:f101::1/64 dev eth0
ip -6 addr del 2001:0db8:0:f101::1/64 dev eth0

3. Encaminamiento.

ip -6 route show [dev <device>] [table <id/all>]

Visualiza las rutas existentes. Se pueden especificar en la visualización, las tablas internas que maneja el equipo (como la caché de destinos). Para ello:

ip -6 route show [dev <device>] table all

Ejemplo:

#ip -6 route show dev eth0

2001:0db8:0:f101::/64 proto kernel metric 256 mtu 1500 advmss 1440
fe80::/10 proto kernel metric 256 mtu 1500 advmss 1440
ff00::/8 proto kernel metric 256 mtu 1500 advmss 1440
default proto kernel metric 256 mtu 1500 advmss 1440

ip -6 route add <ipv6network>/<prefixlength> dev <device> metric 1

Añade/borra una entrada estática a la tabla de rutas (encaminamiento directo a través de una interfaz).

Ejemplo:

ip -6 route add 2000::/3 dev eth0 metric 1

Metric "1" is used here to be compatible with the metric used by route, because the default metric on using "ip" is "1024".

ip -6 route add <ipv6network>/<prefixlength> via <ipv6address> [dev <device>]
ip -6 route del <ipv6network>/<prefixlength> via <ipv6address> [dev <device>]

Añade/borra una entrada estática a la tabla de rutas (encaminamiento indirecto a través de un gateway).

Ejemplo:

ip -6 route add 2000::/3 via 2001:0db8:0:f101::1
ip -6 route del 2000::/3 via 2001:0db8:0:f101::1

Nota: lo habitual es identificar el Gateway (siguiente salto) con su dirección IPv6 local de enlace (FE80::/64)

Ruta por defecto: En el caso de la ruta por defecto, hay que distinguir la funcionalidad de *host* y *router:*

- Host: pueden establecer una ruta por defecto del tipo "::/0", o aprenderla mediante autoconfiguración de un router en el que esté activo radvd.
- *Router* **(forwarding activado)**: no se autoconfiguran, por lo que es necesario establecer la ruta manualmente.

Nota: **Kernels de Linux antiguos** (<= 2.4.17) no soportan la configuración de rutas por defecto. Se pueden establecer pero generan error a la hora de realizar el reenvío necesario de paquetes. Para soportar en dicha situación la ruta por defecto, ésta debe **configurarse con el prefijo global "2000::/3".**

4. Protocolo *Neighbor Discovery*:

Neighbor discovery es el sucesor, en IPv6, del protocolo de resolución de direcciones ARP de IPv4. Se puede obtener la información de los vecinos actuales así como crear/borrar entradas en la tabla de vecinos que el Kernel mantiene (como ARP).

ip -6 neigh show [dev <device>]
Mostrar los vecinos:

Ejemplo: (muestra un vecino alcanzable que es un router)
ip -6 neigh show
fe80::201:23ff:fe45:6789 dev eth0 lladdr 00:01:23:45:67:89 router nud reachable

ip -6 neigh add <IPv6 address> lladdr <link-layer address> dev <device>
ip -6 neigh del <IPv6 address> lladdr <link-layer address> dev <device>
Manipular las entradas a la tabla manualmente:

Ejemplo:
ip -6 neigh add fec0::1 lladdr 02:01:02:03:04:05 dev eth0
ip -6 neigh del fec0::1 lladdr 02:01:02:03:04:05 dev eth0

Para una configuración más avanzada, consultar el manual de "ip"

5. Túneles punto a punto IPv6-in-Ipv4:

ip -6 tunnel show [<device>]
Visualizar los túneles existentes:

Ejemplo:
ip -6 tunnel show
sit0: ipv6/ip remote any local any ttl 64 nopmtudisc
sit1: ipv6/ip remote 195.226.187.50 local any ttl 64

<device>: nombre asociado a la nueva interfaz virtual creada (sitx)

ip tunnel add <device> mode sit ttl <ttldefault>
remote <pv4addressofforeigntunnel> local <ipv4addresslocal>
ip tunnel del <device>
Añadir/eliminar un túnel punto a punto

Ejemplo:
ip tunnel add sit1 mode sit ttl 64 remote 155.210.157.254 local 155.210.157.18
// se crea la **interfaz sit1**, que identifica el túnel de extremos 155.210.157.254 y 155.210.157.18
ip link set dev sit1 up
// se activa la interfaz
ip -6 route add 2001::/3 dev sit1 metric 1
// se añade una ruta por defecto hacia cualquier sitio IPv6, que use la interfaz sit1

El borrado del túnel, por lo tanto, se realiza del modo siguiente, a la inversa:
p -6 route del 2001::/3> dev sit1
ip link set sit1 down
ip tunnel del sit1

6. Acceso a las variables del Kernel (/proc-filesystem) mediante "sysctl"

Existen una serie de ficheros que podemos denominar tablas de configuración en el directorio /proc/net/ con los datos de configuración (ver Anexo I, Práctica 4). Se pueden modificar cambiando su valor directamente (echo 0 > /proc/sys/net/…) o mediante la intefaz sysctl. Por ejemplo, en IPv6:

sysctl net.ipv6.conf.all.forwarding
net.ipv6.conf.all.forwarding = 0
Consultar el valor de la variable "capacidad de reenvío":

sysctl -w net.ipv6.conf.all.forwarding=1
net.ipv6.conf.all.forwarding = 1
Establecer el valor de una variable (si ésta es modificable)

Algunas variables de interés:

Forwarding: configura el comportamiento específico de Host/Router
Tipo: BOOLEAN
Por defecto: FALSE

FALSE: Por defecto, se asume comportamiento de Host, lo que significa:
1. El flag "IsRouter" no se establece en los mensajes *Neighbour Advertisements*.
2. *Router Solicitations* se envían cuando es necesario.
3. Si "accept_ra" es TRUE (por defecto), se aceptan mensajes *Router Advertisements* (y por tanto, se realiza autoconfiguración).
4. Si "accept_redirects" es TRUE (por defecto), se aceptan *ICMPv6 Redirects*.

TRUE: Si se habilita el reenvío, se asume comportamiento de Router, lo que significa lo contrario que lo descrito anteriormente:

1. El flag "IsRouter" se establece en los mensajes *Neighbour Advertisements*.
2. No se envían *Router Solicitations*.
3. Se ignoran los *Router Advertisements*.
4. Se ignoran los *ICMPv6 Redirects*.

Mtu: establece el tamaño máximo del paquete IP (unidad de datos máxima según la interfaz física)
Tipo: INTEGER
Por defecto: 1280 (Mínimo requerido por IPv6)

Nota: No usar espacios alrededor de "=" al establecer valores. Además, al establecer múltiples valores por línea, no olvidar las comillas "", como por ejemplo:
# sysctl -w net.ipv4.ip_local_port_range="32768 61000"
net.ipv4.ip_local_port_range = 32768 61000

7. Autoconfiguración *stateless*:

Soportada por defecto y visualizada en la dirección autoasignada local de enlace (*link-local*) tras activar una interfaz IPv6.

Ejemplo:
ip -6 addr show dev eth0 scope link
2: eth0: <BROADCAST,MULTICAST,UP> mtu 1500 qlen1000
inet6 fe80::211:d8ff:fe6b:f0f5/64 scope link
valid_lft forever preferred_lft forever

Requiere la activación del *daemon* de anuncios de *router* en el *router* del mismo enlace:

7.1. *Router Advertisement Daemon* (radvd)

Antes de activar el servicio es necesario configurar adecuadamente los prefijos y rutas a anunciar en los mensajes *Router Advertisement*. Dicha configuración se realiza mediante el fichero **/etc/radvd.conf** (aunque también se pude utilizar un fichero alternativo)
Los parámetros comúnmente configurados son:

- Prefijos (necesario)
- Tiempo de vida de los prefijos
- Frecuencia de envío de los anuncios (opcional)

Un ejemplo sencillo de configuración (en el fichero antes mencionado):

```
interface eth0   ← anuncio en la interfaz eth0
{
        AdvSendAdvert on;   ← habilito el envío de anuncio
        MinRtrAdvInterval 3;
        MaxRtrAdvInterval 10;   ← intervalo cada cuánto se anuncia (entre valor mínimo y valor máximo)
        prefix 2001:0db8:0100:f101::/64   ← prefijo a anunciar
        {
                AdvOnLink on;
                AdvAutonomous on;
                AdvRouterAddr off;
        };
};
```

Valores por defecto (no modificar). Información detallada en man radvd.conf

Lo que finalmente resulta en un *host* conectado en dicho enlace

```
# ip -6 addr show eth0
3: eth0: <BROADCAST,MULTICAST,UP> mtu 1500 qdisc pfifo_fast qlen 100
inet6 2001:0db8:100:f101:2e0:12ff:fe34:1234/64 scope global dynamic
valid_lft 2591992sec preferred_lft 604792sec
inet6 fe80::2e0:12ff:fe34:1234/10 scope link
```

Como no se definió un tiempo de vida, se ha usado un valor muy alto.

Un vez correctamente configurado, el servicio se activa ejecutando el comando:
service radvd start

El servicio se detiene con el comando:
service radvd stop

La posible configuración de todos los parámetros disponibles se encuentra detallada en el manual de ayuda correspondiente.

8. Enlaces de ayuda

Configuración IPv6 en Linux:

http://tldp.org/HOWTO/Linux+IPv6-HOWTO/

La consulta de las páginas de ayuda de Linux (man) puede ser de utilidad a lo largo de la práctica, tanto para los comandos de configuración como para las funciones de análisis.

Del mismo modo, se pueden consultar vía web:

http://linux.die.net/man/8/ip
http://linux.die.net/man/8/radvd
http://linux.die.net/man/8/ping
http://linux.die.net/man/8/ping6
http://linux.die.net/man/8/tracepath
http://linux.die.net/man/8/tracepath6
http://linux.die.net/man/8/traceroute
http://linux.die.net/man/8/traceroute6
http://linux.die.net/man/8/netstat

Anexo II. Entregable de evaluación

E1. Anotar la siguiente información:

Direcciones IPv4 de los extremos del túnel (local – cliente – y remoto – servidor). Direcciones IPv6 de los extremos del túnel (local – cliente – y remoto – servidor). Prefijo /64 asignado por el proveedor (*Routed IPv6 Prefixes*), para utilizarlo posteriormente en la red LAN correspondiente. Direcciones IPv6 de servidores DNS proporcionadas por el proveedor (*Available DNS resolvers*) para su posterior utilización.

E2. Anotar el ejemplo de configuración

P1. Una vez establecido el túnel, conectarse desde el *router* de salida de la red LAN correspondiente (A o B) a cualquier sitio con conectividad IPv6[3]. Por ejemplo, accede a la siguiente página y **apunta la dirección IPv6** de la misma: http://www.consulintel.es Capturar en la interfaz eth0 del *router* (no en la interfaz túnel creada) y verificar que, efectivamente, el tráfico desde el *router* hacia el exterior es IPv6 sobre IPv4. Muéstralo indicando las direcciones IPv6 e IPv4 que aparecen en los paquetes.

	Dirección fuente	Dirección destino
Direcciones IPv6		
Direcciones IPv4		

[3] Se pueden buscar en Internet, *IPv6 sites*

P2. Observa el valor MTU configurado en la interfaz tipo túnel (***ip -6 link show dev [nombre_tunel] / ifconfig [nombre_tunel]***). Justifica dicho valor teniendo en cuenta la información capturada previamente.

E3/P3. Una vez configurado el escenario completo, se comprobará la conectividad del mismo verificando la comunicación entre los equipos de las redes LAN A y LAN B mediante comandos como **ping6, o traceroute6** (consultar las páginas de ayuda necesarias – Anexo II, apartado 8).

E4/P4. Sin modificar el fichero resolv.conf ¿Se puede acceder a sitios IPv6 desde los *host* de las redes LAN, por ejemplo www.consulintel.com? Prueba a realizar nuevamente la conexión, utilizando directamente la dirección IPv6 de www.consulintel.com (http://[<dir.IPv6>]:80). ¿Se puede ahora acceder? Da una explicación de por qué funciona en el *router* y no lo hace en el *host*.

COMENTAR CON EL PROFESOR

P5. Observa las direcciones IPv6 existentes en las interfaces de los PCs y pon un ejemplo de dirección local de enlace y dirección global. Muestra, en una de ellas, cómo se ha obtenido el identificador de interfaz de 64 bits (EUI-64).

COMENTAR CON EL PROFESOR

P6. A partir de la captura indica los dos casos de procedimiento DAD descritos especificando en ambos: dirección *multicast* (*Solicited Node Address*) a la que se dirige el mensaje de ND (*Neighbor Discovery*, ICMPv6) y dirección *unicast* por la que se pregunta (*target*)

Verifica la correspondencia entre las direcciones *multicast* y *unicast* identificadas.

Muestra los paquetes en los que se pide y responde (por parte del equipo y el *router*) los parámetros necesarios para la autoconfiguración. Resalta dichos parámetros en el mensaje correspondiente.

P7. Realiza las siguientes conexiones, identificando la correspondencia entre dirección MAC destino (dirección de nivel de enlace – *link layer* – lladdr) y dirección IPv6 destino del paquete capturado en el equipo origen del ping:

COMENTAR CON EL PROFESOR

P8. Comprueba las conexiones (mediante varios ping, paso a paso) entre PCA1/PCB1 y PCB1/PCA1, es decir comprobando que funciona el ping a cada una de las direcciones intermedias del camino entre los extremos. Justifica los tiempos que tarda cada uno de los ping en función de la posición del destino.

E9/P9. Modificar el **MTU del interfaz eth1 de PCB3 a un valor de 1300 y el del interfaz eth0 de PCA3 a un valor de 1350**. Observa qué sucede si se realiza un **ping6 desde PCA1 hacia PCB1** con un tamaño de **1400 bytes.** ¿Quién realiza la fragmentación? ¿Qué tramas se intercambian entre los quipos? ¿Qué diferencia habría si la red fuera totalmente IPv4?