UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

Facultad de Estudios Superiores
"ARAGÓN"

"Actualización del soporte IPv6 en la Red de Internet2 de México (RedCUDI)"

TESIS

QUE PARA OBTENER EL TITULO:

INGENIERO MECÁNICO ELÉCTRICO

PRESENTA:

Mark Ricárdez Zárate

ASESOR DE TESIS:

M. en C. Leobardo Hernández Audelo.

SAN JUAN DE ARAGON, EDO. DE MEXICO **2007**

Agradecimientos

A mis Padres Juidth y Hever
Por encaminarme en todo y darme su apoyo incondicional durante toda mi vida y educar en mí las virtudes necesarias para poder ser lo que soy…

A mis Hermanos Ruth, Einar y Mittzi
Por la compañía y el apoyo que me brindan…

A Lucy
Por su comprensión y apoyo durante los años que le dediqué a este trabajo de Tesis

A mis amigos y compañeros de trabajo
Por la amistad que siempre me brindaron

A Quim. Laura Mata Montiel
Por enseñarme el camino, ya que sin ella no hubiera sido posible este trabajo

A Ing. Ázael Fernandez Alcantara
Por darme su apoyo en la realización de esta trabajo. Y participar en nuestros intereses a fondo de nuestro tema de tesis.

A M. en C. Leobardo Hernández Audelo
Por apoyarme en el desarrollo de esta investigación y ofrecerme sus consejos

A la Dirección General de Servicios de Computo Académico
Por abrirme sus puertas;

Y sobre todo,
A la Universidad Nacional Autónoma de México y
A la Facultad de Estudios Superiores Aragón,
Por hacerme quien soy.

Índice General

Índice de Figuras

Capitulo 1.

Capitulo 2.

Capitulo 3.

Capitulo 4.

Capitulo 5.

Capitulo 6.

Capitulo 7.

Capitulo 8.

Índice de Tablas

Capitulo 2.

Capitulo 3.

Capitulo 4.

Capitulo 5.

Capitulo 6.

Capitulo 7.

Capitulo 8.

Anexos.

Índice de Graficas

Capitulo 6.

Introducción

Hoy en día, Internet es la red mundial más grande de redes de computadoras y ha supuesto una revolución sin precedentes en el mundo de la informática y de las comunicaciones. Internet es a la vez una oportunidad de difusión mundial, un mecanismo de propagación de la información y un medio de colaboración e interacción entre los individuos y sus ordenadores independientemente de su ubicación geográfica.

La demanda de sistemas de comunicaciones basados en la transmisión de información a través de la red mundial Internet, ha tenido un crecimiento sin precedentes. Cada vez se necesita una mayor velocidad y eficiencia en el intercambio de información entre dos puntos cualesquiera del mundo.

El Internet de hoy en día ya no es una red académica, como al principio, sino se ha convertido en una red que involucra principalmente intereses comerciales y particulares, así mismo los ISP´s de Internet, sobre venden el ancho de banda que disponen, haciendo imposible garantizar un servicio mínimo en horas picos de uso de la red, y por otro lado, los enlaces de alta velocidad son aún demasiado costosos para poder realizar su comercialización masiva.

Todo esto hace a Internet, un medio no apto para poder transmitir enormes volúmenes de información, videos, transmisión de conferencias y colaboración en tiempo real o garantizar la comunicación permanente en sincronía; por lo que organismos a nivel mundial, con espíritu de colaboración, se unen para formar la siguiente generación de Internet. Internet2, es una red de cómputo sustentada en tecnologías de vanguardia que permite una alta velocidad de transmisión de contenidos, con el principal objetivo de desarrollar la próxima generación de aplicaciones telemáticas para facilitar las misiones de investigación y educación de las universidades.

Por otro lado, debido a que la transmisión de información vía Internet esta inmersa en un entorno tecnológico de rápida evolución, con tendencia a modificaciones inmediatas en los modos de operación, y demanda cada vez un mayor numero de servicios nuevos, los sistemas y protocolos que se diseñaron en un principio para cubrir las necesidades del momento, se han convertido con el paso del tiempo en insuficientes y obsoletos. Es decir, los sistemas de hoy en día se han quedado al margen del desarrollo tecnológico. En estos momentos el principal problema al que se enfrentan estos sistemas y protocolos, es la insuficiencia de direcciones IP ya que se prevé que en un tiempo no muy largo se agoten por completo.

Por lo anterior, se requiere implementar un nuevo protocolo de manera que no limite el crecimiento de la red y de los servicios, sino al contrario, que pueda incrementar su capacidad sin necesidad de efectuar un rediseño de toda la estructura, el cual debe ser planteado bajo la premisa de que la red continué expandiéndose como hasta ahora lo ha venido haciendo o incluso en una mayor proporción, por lo tanto, deba ser capaz de cubrir la demanda de servicios a futuro.

Es por eso que la IETF (Internet Engineering Task Force) desarrolla la siguiente generación del protocolo IP, el cual ha sido planteado como una solución a la problemática inminente de escasez de direcciones de todas las redes basadas en IP, como lo son Internet e Internet2, debido a las incapacidades de diseño del mismo

protocolo IPv4, y de igual forma, brinda algunas soluciones a las complicaciones que conlleva la migración de las redes IPv4 a IPv6.

Esta tesis se enfoca principalmente en la implementación que se hizo del protocolo de Internet IPv6 y los trabajos desarrollados en la red de Internet2 de México CUDI (Corporación Universitaria para el Desarrollo de Internet), después que el Grupo de trabajo IPv6, obtuvo del RIR (Regional Internet Registries) de la zona "LACNIC" (Latin American and Caribbean Internet Addresses Registry) un bloque de direcciones propio /32 para producción de tipo sTLA (sub Top-Level Aggregation).

Para el desarrollo del trabajo se necesita en primer lugar, conocer muy bien todas las nuevas capacidades y características que trae la nueva versión del protocolo IP, y demás modificaciones y mejoras que se han hecho a los otros protocolos y aplicaciones que involucra el manejo de IP en una red; así como también conocer la topología de la RedCUDI, equipos de backbone, estructura interna de los miembros conectados a la red e información restante concernientes al manejo de la red. Todo esto con el único propósito de obtener la información necesaria para realizar un buen diseño del direccionamiento IPv6, para ser aceptada por el CDR (Comité de desarrollo de la red) y finalmente ser configurada e implementada en la RedCUDI.

Por todo lo anterior, el siguiente trabajo de tesis cuenta con los siguientes temas, que en conjunto brindan una mejor visión del desarrollo de la tesis:

1) Capitulo en el que se plantean los principios básicos de la comunicación de datos entre sistemas computacionales; debido a que es importante tener bien entendido los elementos que intervienen en una comunicación IP, y situar en qué capa del modelo OSI, se lleva el desarrollo del trabajo.

2) Capitulo en el que se explican las características básicas del protocolo de Internet versión 6 (IPv6); brindando con esto una visión amplia del manejo del protocolo IPv6 en las redes de datos y además brinda las herramientas necesarias para llevarlo a la implementación.

3) Capitulo en el que se desarrollan algunos mecanismos de transición utilizados para implementar el protocolo IPv6 sobre infraestructura de enrutamiento IPv4 existente; obteniendo con esto las bases necesarias para ofrecer algunos servicios de conectividad IPv6 a aquellos miembros que aún no cuentan con los suficientes requerimientos en equipo para configurar IPv6 nativo en sus redes.

4) Capitulo en el que se habla del funcionamiento y de los cambios que se le han hecho a los diferentes protocolos estandarizados de enrutamiento tanto IGP y EGP, para proporcionar enrutamiento en las redes IP, que como se conocerá es la parte medular de las redes para que funcionen de manera óptima.

5) Capitulo en el que se habla de los antecedentes de Internet, y el porque en nuestros días no puede ser lo suficientemente óptima para brindar la trasmisión de grandes volúmenes de datos, y de igual forma se discute el surgimiento de Interne2 o NREN´s (National Research and Education Networks); así como la situación de estas redes a nivel mundial y del soporte IPv6 que actualmente presentan.

6) Capitulo en el que se desarrollan los trabajos realizados, para la implementación del bloque de direcciones IPv6 para producción obtenido de LACNIC por el grupo de trabajo IPv6, el 15 de noviembre de 2005, en la RedCUDI, los cuales fueron aceptados por el CDR de CUDI, y algunos mas publicados en Internet como estándar RFCMX de CUDI. Dichos trabajos son presentados con direccionamiento IPv6 de documentación por implicaciones de seguridad.

7) Capitulo en el que se mencionan algunas aplicaciones que actualmente ya cuentan con el soporte IPv6, ya que son parte crucial en las redes de datos para dar algunos servicios a los usuarios de las redes IP, y de igual forma pueden ser retomadas dentro de Internet2 para llevar a cabo las tareas de los investigadores, y así mismo, para brindar servicios y aplicaciones sobre Internet2.

8) Capitulo en el que se desarrollan algunas maquetas de pruebas, para probar la conectividad IPv6 con diferentes marcas de equipos de telecomunicaciones que actualmente tienen implementado el stack de IPv6 dentro sus sistemas operativos; dejando con esto, a toda la comunidad de administradores de redes, y a la comunidad CUDI, un legado de documentos que pueden ser retomados para la implementación del protocolo IPv6 dentro de sus redes, o en su caso para enseñanza.

Por último un capitulo en el que se dan las conclusiones finales obtenidas del trabajo y un anexo, que está enfocado a brindar comandos básicos IPv6 para implementar el protocolo en diferentes equipos que soportan el stack.

Capítulo

1.Conceptos Generales

Resumen

Este capítulo plantea los principios básicos necesarios para la comunicación de datos entre sistemas computacionales. Es importante saber en que parte de un sistema de comunicación se desarrollará la tesis, en éste proyecto en particular se trabajó en la capa 3 del modelo TCP/IP, así como también tener presente que para que exista un intercambio de información es necesario la utilización de estándares que rigen a la comunidad mundial. Dos de los mas importantes estándares conocidos son el modelo de referencia OSI y la pila de protocolos TCP/IP, entenderlos brindará las bases necesarias para conocer el funcionamiento de la comunicación, por lo cual este capitulo es de suma importancia para comprender de forma clara los demás.

1.1. ORGANIZACIONES DE ESTANDARIZACION

El proceso de comunicación es inherente a la vida humana. Desde tiempos antiguos se han utilizado diferentes formas de comunicación para transmitir ideas, pensamientos y sentimientos, por tal razón se han buscado las formas de llevar el intercambio de conocimiento a distancias lejanas a tal punto de compartirlas y distribuirla a nivel mundial.

Existen muchas formas de realizar el proceso de comunicación. Una de ellas y la que más auge ha tenido en los últimos años son las redes de datos. Este sistema consiste en transmitir información entre diferentes dispositivos por medio de interconexiones guiadas y no guiadas.

En la actualidad existen gran variedad de dispositivos y muchos fabricantes propietarios, dando soluciones propias para sus sistemas, provocando incompatibilidad entre ellos. Es por tal motivo que surgió la necesidad de desarrollar guías para los fabricantes, vendedores, agencias de gobierno y otros proveedores de servicios para asegurar la interconectividad entre diferentes sistemas de información a nivel nacional e internacional.

En realidad estas guías antes mencionadas son las que reciben el nombre de **estándar,** por lo que son esenciales para crear y mantener un mercado abierto y competitivo entre los fabricantes de los equipos, garantizando la interoperabilidad nacional e internacional de los datos y la tecnología y los procesos de telecomunicaciones. Por tal motivo, recae gran responsabilidad en las organizaciones reguladoras, ya que de ellos depénde que la innovación y el desarrollo tecnológico siga evolucionando en vez de retrasarla.

Por lo tanto se dice que un **estándar** proporciona un modelo de desarrollo que hace posible que un producto funcione adecuadamente con otros, sin tener en cuenta quien lo ha fabricado.[1]

Los estándares son desarrollados mediante la cooperación entre **comités de creación de estándares**, **foros** y **agencias reguladoras** de los gobiernos.

1.1.1. Comités de creación de estándares

Aunque hay muchas organizaciones que se dedican a la definición y establecimiento de estándares para datos y comunicaciones, en Norteamérica fundamentalmente son los siguientes:

- The International Standards Organization (ISO).
- The International Telecommunications Union-Telecommunication Standards Sector (ITU-T, anteriormente el CCITT).
- The American National Standards Institute (ANSI).
- The Institute of Electrical and Electronics Engineers (IEEE).
- The Electronic Industries Association (EIA).

[1] Behrouz A. Forouzan (2001). Transmisión de datos y redes de comunicaciones. Aravaca, Madrid. McGraw Hill. Segunda Edición. pp. 9.

A continuación se describirán brevemente estas organizaciones:

ISO

La ISO (International Standards Organization) es un organismo multinacional cuyos miembros provienen fundamentalmente de los comités de creación de estándares de varios gobiernos a lo largo del mundo. Creado en 1947, el ISO es una organización totalmente voluntaria dedicada a acuerdos mundiales sobre estándares internacionales, su objetivo es facilitar el intercambio internacional de productos y servicios, proporcionando modelos de compatibilidad, mejoras de calidad, mejoras de productividad y precios más baratos. Los Estados Unidos están representados en el ISO por ANSI.

ITU-T

En los años 70, cierto número de países estaban definiendo estándares. A pesar de esto seguía habiendo incompatibilidad internacional, fue por esta razón que Las Naciones Unidas respondieron a esta problemática, desarrollando como parte de su Unión Internacional de Telecomunicaciones (ITU) un comité denominado Comité Consultivo para la Telefonía y la Telegrafía Internacional (CCITT), el cual se dedico al desarrollo y establecimiento de estándares para telecomunicaciones en general y para la telefonía y los sistemas de datos en particular. El 1 de marzo de 1993, el nombre de este comité se cambio a Unión Internacional de Telecomunicaciones – Sector de Estándares de Telecomunicaciones (ITU-T), este se divide en grupos de estudios los cuales se dedican a aspectos distintos. Los estándares de este comité se publican cada cuatro años.

ANSI

El Instituto Nacional Americano para la Estandarización, es una organización completamente privada sin ánimo de lucro, todas las actividades de ANSI están orientadas hacia el desarrollo de los Estados Unidos. Los miembros de ANSI son sociedades profesionales, asociaciones de la industria, agencias gubernamentales y reguladoras y grupos de consumidores.

IEEE

El Instituto de Ingenieros Eléctricos y Electrónicos, es la mayor sociedad profesional de ingeniería en el mundo. De ámbito internacional, sus objetivos son el desarrollo de la teoría, la creatividad y la calidad de los productos en el campo de la ingeniería eléctrica, la electrónica y la radio, así como otras ramas relacionadas de la ingeniería.

EIA

En la línea de ANSI, la Asociación de Industrias Electrónicas (EIA) es una organización sin ánimo de lucro dedicada a la promoción de aspectos de la fabricación electrónica. Sus objetivos incluyen despertar el interés de la educación pública y hacer esfuerzos para el desarrollo de estándares.

1.1.2. Foros

Los foros trabajan con las universidades y los usuarios para probar, evaluar y estandarizar nuevas tecnologías. Concentrados sus esfuerzos en una tecnología en particular, los foros son capaces de acelerar la aceptación y el uso de esa tecnología en la comunidad de las telecomunicaciones. Los foros presentan sus conclusiones a los organismos estandarizadotes. Algunos de estos foros son los siguientes.

Foro Frame Relay

El Frame Relay fue constituido por DEC, Northern Telecom, Cisco y StrataCom para acelerar la aceptación e implementación de Frame Relay. Todos sus resultados son enviados a ISO.

Foro de ATM y consorcio ATM

El Foro de ATM y el Consorcio de ATM existen para promocionar la aceptación y el uso del Modo de Transferencia Asíncrono (ATM) y sus tecnologías. El Consorcio ATM esta constituido por vendedores de hardware y software que suministran ATM.

Internet Society (ISOC) e Internet Engineering Task Force (IETF)

La Internet Society (ISOC) se concentra en los aspectos de usuario, incluyendo las mejoras al conjunto de protocolos TCP/IP. El IETF es la organización de estándares para Internet en si misma. Revisa tanto el software como el hardware de Internet.

1.1.3. Agencias Reguladoras

Toda la tecnología de comunicaciones esta sujeta a regulación por las agencias del gobierno. A continuación se describe una de estas agencias reguladoras:

FCC

La Comisión Federal de Comunicaciones en los Estados Unidos, tiene autoridad sobre el comercio interestatal e internacional en lo que se refiere a las comunicaciones. Cada elemento de las tecnologías de las telecomunicaciones debe tener una aprobación del FCC antes de que pueda ser vendido

1.2. MODELO OSI

OSI (Open system Interconnection), es un estándar desarrollado por ISO. El modelo OSI no es un protocolo, es un modelo de interconexión de sistemas abiertos que cubre todos los aspectos de las redes de comunicación. Esto es, permite que dos sistemas diferentes se

puedan comunicar independientemente, sin que sea necesario cambiar la lógica del hardware o el software subyacente[2].

El modelo OSI esta formado por siete niveles, ordenados como se muestra en la Figura 1.

Figura 1. El modelo OSI

En la Figura 2 se muestra la comunicación entre dos dispositivos, a medida que el mensaje viaja del nodo A al B a través de una infraestructura de datos, puede pasar por muchos puntos intermedios, estos puntos habitualmente solo tienen los tres niveles del modelo OSI.

La comunicación inicia desde el punto más alto de los 7 niveles del modelo, cada nivel llama a los servicios del nivel inferior, el paso de los datos y la información a cada nivel, es posible porque existe una interfaz entre cada par de niveles adyacentes. Cada interfaz define qué información y servicios debe proporcionar al nivel superior, esta información adquirida de un nivel superior la empaqueta y le añade su propia información, realizando este proceso consecutivamente los niveles restantes hasta llegar al primer nivel de la pila, quién se encarga de convertir toda la información recibida en un formato que sea capaz de transmitirse por un medio de transmisión hasta la maquina receptora, que a la vez tiene que hacer todo el proceso de comunicación descrito pero de forma inversa para extraer los datos originales enviados.

Es importante decir que la comunicación se gobierna mediante una serie de reglas y convenciones acordadas por organismos de estandarización, que se denominan protocolos, la comunicación entre los dispositivos se establece en cada nivel, esto es, el nivel x del emisor se comunica con el nivel x de la maquina receptora, por lo tanto la comunicación de datos sea de igual a igual.

La información que añade cada nivel suele llamarse cabeceras y colas, las cabeceras son añadidas en los niveles 6, 5, 4, 3 y 2, y habitualmente las colas solo en el nivel 2.

[2] Behrouz A. Forouzan (2001). Transmisión de datos y redes de comunicaciones. Aravaca, Madrid. McGraw Hill. Segunda Edición. pp. 41.

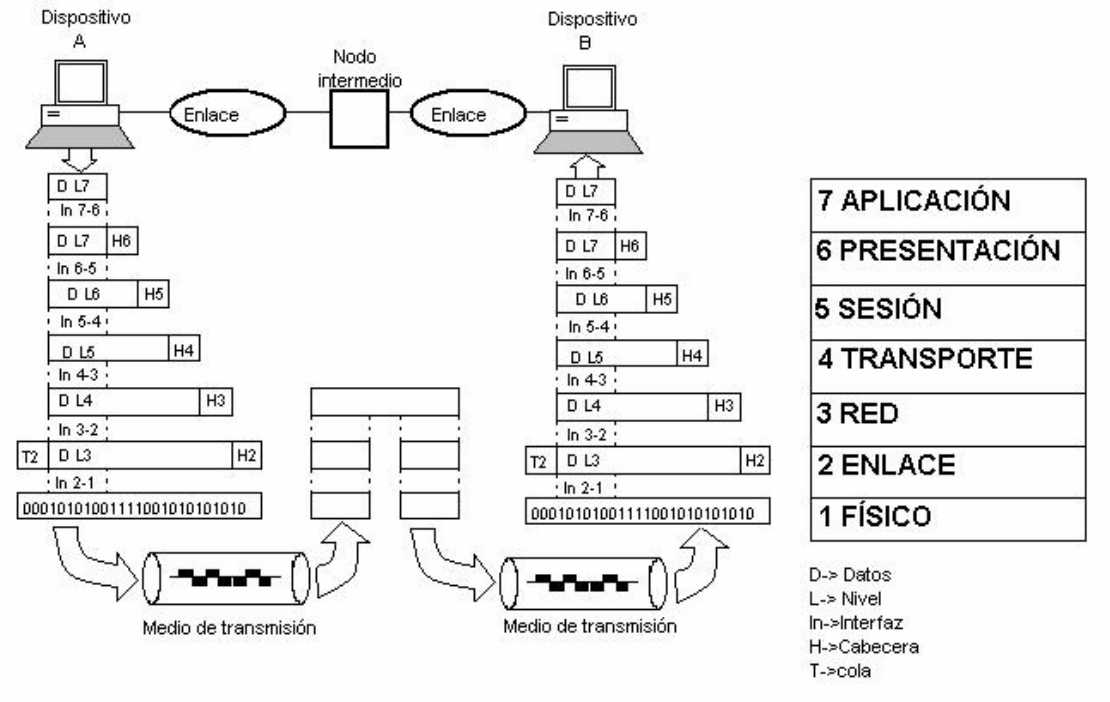


Figura 2. Comunicación utilizando el modelo OSI

1.2.1. Funciones de los niveles

A continuación se describira brevemente la funcion de cada una de las capas del modelo de referencia OSI:

NIVEL FÍSICO

El nivel físico coordina las funciones necesarias para trasmitir el flujo de datos a través de un medio guiado o no guiado. Trata con las especificaciones eléctricas y mecánicas de la interfaz y del medio de transmisión. También define los procedimientos y las funciones necesarias para que las interfaces y los medios logren la transmisión.

NIVEL DE ENLACE DE DATOS

Este nivel hace fiable al enlace y es el responsable de la entrega nodo a nodo, el flujo de datos recibido lo maneja en unidades denominadas tramas, usa las direcciones físicas para llevar a cabo la comunicación, en caso que la trama pase a otra red necesita una dirección lógica que se encuentra en la capa 3, controla la velocidad de transmisión, añade mecanismos para detección de errores. El control de errores se consigue normalmente a través de una cola que se añade al final de la trama y es la responsable del control de acceso al medio mediante un mecanismo para evitar las colisiones en el medio de transmisión.

NIVEL DE RED

El nivel de red es el responsable de la entrega de un paquete desde el origen al destino y a través de múltiples redes (enlaces). Mientras el nivel de enlace supervisa la entrega de datos entre dos dispositivos de la misma red, el nivel de red es el responsable de la transmisión de paquetes a través de múltiples redes.

NIVEL DE TRANSPORTE

El nivel de transporte es el responsable de la entrega de todo el mensaje del origen al destino, mientras que el nivel de red se encarga de la entrega de paquetes sin importar la relación entre éstos, el nivel de transporte asegura que todo el mensaje llegue intacto y en orden. Para mayor seguridad el nivel de transporte puede crear una conexión entre dos puertos finales. Una conexión es un único camino lógico entre el origen y el destino asociado a todos los paquetes del mensaje. La creación de una conexión involucra tres pasos: establecimiento de la conexión, transferencia de los datos y liberación de la conexión. Mediante el confinamiento de la transmisión de todos los paquetes a un único camino, el nivel de transporte tiene más control sobre la secuencia, flujo, detección y corrección de errores.

NIVEL DE SESIÓN

El nivel de sesión coordina la conexión y desconexión de los diálogos entre las aplicaciones de los niveles superiores, esto es dentro de un sistema de programas de aplicación de usuarios deben de ser capaces de comunicarse e intercambiarse archivos y concluir la conexión, establece puntos de sincronización para el intercambio de datos, coordina quién envía, cuándo y de qué forma (full-duplex o semi-duplex), y se asegura que los datos se intercambien de forma completa antes de cerrar la sesión, es por todo esto que se dice que el nivel de transporte puede hacer algo de trabajo mientras que el nivel de sesión debe hacerlo todo o nada.

NIVEL DE PRESENTACIÓN

En el nivel de presentación se realiza la traducción, cifrado/descifrado, autenticación y compresión. Los equipos pueden almacenar cadenas de caracteres utilizando diferentes formatos de código (ASCII o EBCDIC). Para que un equipo pueda entender un código diferente al de su sistema necesita traducirlo.

El cifrado/descifrado consiste en alterar los caracteres o bits de los datos para que no puedan ser interceptados por otras personas malintencionadas y por lo tanto leídas, es por esto que se cifra la información en el transmisor y el único que puede descifrarlo es el receptor.

La autenticación es una técnica que intenta verificar que un mensaje proviene de un emisor auténtico y no de un impostor y por último la compresión hace eficiente el uso del medio de transmisión mediante la reducción del número de bits enviados.

NIVEL DE APLICACIÓN

Es el nivel más alto del modelo OSI que se entiende directamente con el usuario final, al proporcionarle cualquier función requerida por el usuario como: el correo electrónico, el acceso y la transferencia de archivos remotos, la gestión de datos compartidos y otros tipos de servicios para soportar las aplicaciones y administrar las comunicaciones.

1.3. PILA TCP/IP

En 1969, la Agencia de proyectos de investigación avanzada (ARPA), perteneciente al Departamento de Defensa de los EE.UU., financió un proyecto. ARPA estableció una red de comunicación de paquetes de computadoras conectadas mediante líneas punto a punto alquiladas denominada "Red de la agencia de proyectos de investigación avanzada" (ARPANET), que proporcionó la base para las primeras investigaciones en interconexión de redes. Las convenciones desarrolladas por ARPA para especificar la forma en la que computadoras individuales podían comunicarse a través de la red, se convirtió en TCP/IP. El Protocolo de control de transmisión (TCP) fue desarrollado antes que el modelo OSI, por tanto, los niveles del protocolo TCP/IP no coinciden exactamente con los del modelo OSI.[3]

Gracias a TCP/IP es posible que Internet se considere como una sola red que conecta muchas computadores o dispositivos; internamente la red es una interconexión de muchas redes físicas independientes (LAN, MAN) conectadas por dispositivos de interconexión.

TCP/IP consta de cinco niveles: físico, enlace de datos, red, transporte y aplicación. El nivel de aplicación se considera como la combinación de los niveles sesión, presentación y aplicación del modelo OSI. En el nivel de transporte define dos protocolos TCP y UDP; en el nivel de red define principalmente al protocolo IP, es importante hacer notar que la versión actual del protocolo IP es la 4, y por algunas deficiencias que presenta, se ha propuesto el desarrollo de una nueva versión que vendrá a sustituir paulatinamente a la actual versión 4 esta nueva versión es IPv6. En el nivel enlace de datos y físico TCP/IP no define ningún protocolo específico, puede emplear cualquier estándar de propietarios como ATM, Ethernet, Frame Relay, HDLC, PPP, Token Ring, Wi-Fi, SDH, etc.

En la Figura 3 se muestra la relación de TCP/IP con el modelo OSI, y el encapsulado de las unidades de datos creadas en los diferentes niveles de TCP/IP. La unidad de datos creada en el nivel de aplicación se denomina "mensaje", en el nivel de transporte es "segmento" o "datagrama de usuario", en el nivel de red es "datagrama"; para que los datos puedan ser transferidos por Internet, el datagrama debe ser convertido en una "trama" para ser enviado por el medio físicó.

[3] Behrouz A. Forouzan (2001). Transmisión de datos y redes de comunicaciones. Aravaca, Madrid. McGraw Hill. Segunda Edición. pp. 681, 682.

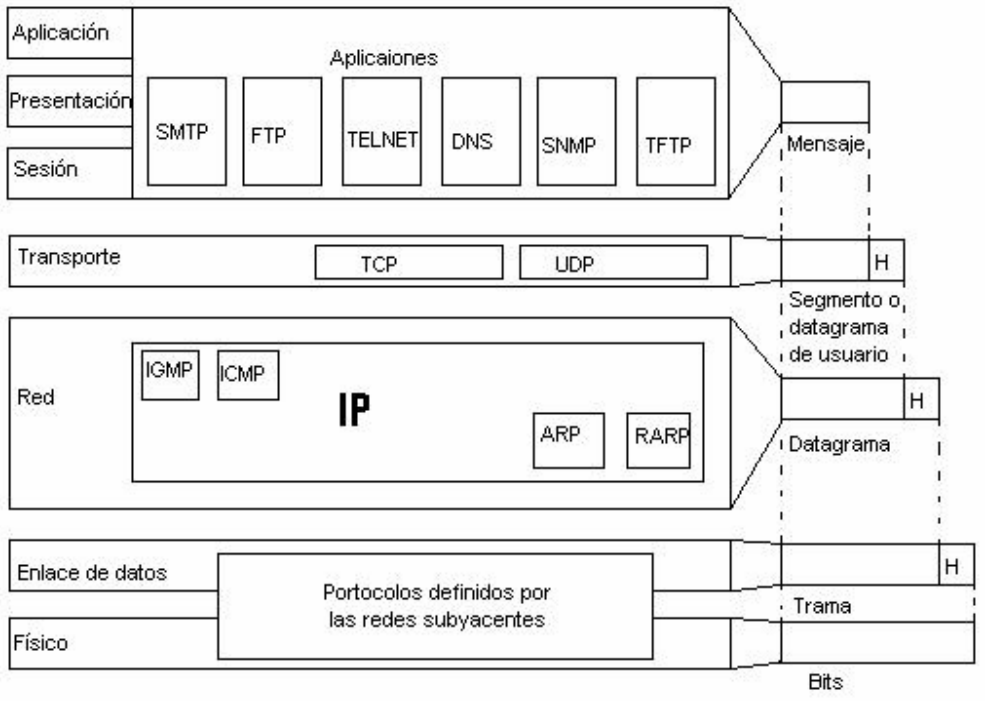


Figura 3. TCP/IP y el modelo OSI

1.3.1. Nivel de red

En el nivel de red, TCP/IP ofrece el protocolo IP (Internet Protocol) en este caso mencionaremos la versión 4, que a su vez, contiene cuatro protocolos: ARP, RARP, ICMP e IGMP.

A continuación se dará una breve explicación del funcionamiento de ellos.

Protocolo IPv4.

Es un protocolo de transmisión basado en datagramas sin conexión y no fiable, no ofrece comprobación ni seguimientos de los paquetes, esto es que cuando un datagrama es enviado a través de una red, IP no se hace responsable de que llegue o no a su destino. IP no se encarga de realizar un seguimiento de los enrutadores ni ofrece facilidades para reordenar los datagramas una vez recibidos. Debido a que es un servicio sin conexión, IP no crea circuitos virtuales para la entrega, es por esta razón que IP necesita de otro protocolo que lo haga fiable, en este caso el protocolo utilizado es TCP.

Datagrama

Un datagrama es un paquete de longitud variable (hasta 65536 bytes) que consta de dos partes: cabecera y datos. La cabecera puede incluir de 20 a 60 bytes y contiene información esencial para el encaminamiento y la entrega. A continuación en la Figura 4 se muestra un datagrama IP y los campos de la cabecera.

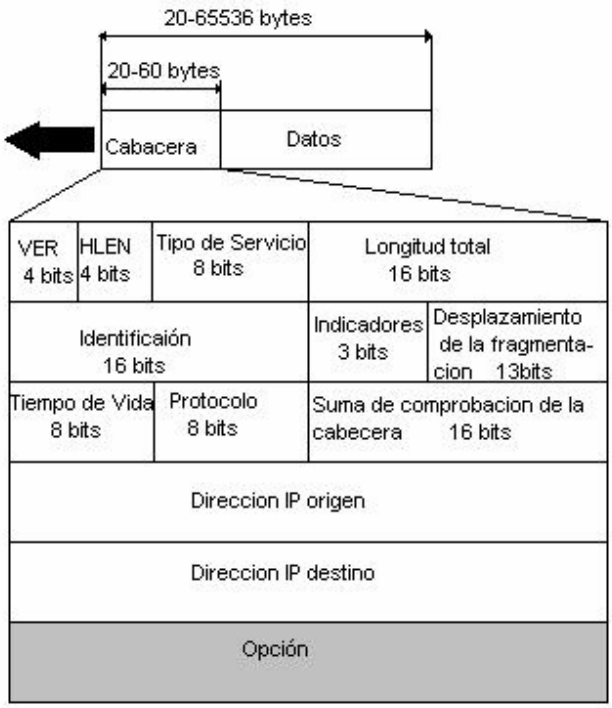


Figura 4. Datagrama y cabecera IPv4

- **Versión**. El primer campo define el número de la versión de IP. La versión actual es la 4, con un valor binario 0100.
- **Longitud de la cabecera.** Este campo define la longitud de la cabecera en múltiplos de cuatro bytes. Cuatro bits pueden representar entre 0 y 15, que cuando se multiplica por 4 da un máximo de 60 bytes.
- **Tipo de servicio.** Este campo define la forma en la que se debería manejar el datagrama, incluye bits que definen la prioridad del datagrama. También contiene bits que especifican el tipo de servicio que el emisor desea como el nivel de prestaciones, fiabilidad y retardo.
- **Longitud total.** El campo con la longitud total define la longitud total del datagrama IP. Es un campo de 2 bytes (16 bits) que puede definir hasta 65 535 bytes.
- **Identificación.** Este campo se utiliza en la fragmentación. Un datagrama, cuando pasa a través de redes diferentes, puede dividirse en fragmentos que coincidan con el tamaño de la trama de red. Cuando esto ocurre, cada fragmento es identificado con un número de secuencia en este campo.
- **Indicadores.** Los bits de este campo están relacionados con la fragmentación (el datagrama puede estar o no fragmentado; puede ser el primero, el ultimo fragmento, etc.).
- **Desplazamiento del fragmento.** El desplazamiento del fragmento es un puntero que muestra el desplazamiento de los datos en el datagrama original (si se fragmenta).
- **Tiempo de vida.** El tiempo de vida define el número de saltos que puede dar el datagrama antes de ser descartado. La estación origen, cuando crea el datagrama,

fija este campo a un valor inicial. A medida que el datagrama viaja por la red, cada enrutador disminuye este valor en 1. si el valor se hace 0 antes de que el datagrama haya alcanzado el destino final, se descarta el datagrama. Esto evita que un datagrama vuelva o viaje de forma indefinida entre enrutadores.

- **Protocolo.** Este campo define el protocolo de nivel superior que se encuentra encapsulado en el datagrama (TCP, UDP, ICMP, etc.).
- **Suma de comprobación de la cabecera.** Este campo de 16 bits se utiliza para comprobar la integridad de la cabecera, no del resto del datagrama.
- **Dirección Origen/Destino.** En este campo se pone una dirección origen o destino respectivamente sea el caso, en nuestro caso por ser la versión 4 del protocolo de Internet la dirección son 4 bytes (32 bits).
- **Opciones.** El campo opciones ofrece mayor funcionalidad al datagrama IP. Puede transportar datos que controlan el encaminamiento, la temporización, la gestión y el alineamiento.

1.3.1.1.Direccionamiento

Además de la dirección física que identifica al dispositivo individual, se necesita una convención que identifique la conexión de una estación a la red. Esta convención es una dirección de 4 bytes (32 bits) separados por puntos, estas direcciones definen tres campos: la clase, el identificador de la red y el identificador de la estación. Estas partes son de longitud variable la cual depende de la clase de la dirección, las clases permiten cubrir las necesidades diferentes de las organizaciones, es por medio de la clase que se puede aumentar o disminuir el número de estaciones conectadas a la red, las clases son A, B, C, D, E.

La clase A permite tener más estaciones (3 bytes para estaciones), la B permite tener menos estaciones que la clase A y más que la clase C (2 bytes para estaciones) y la clase C es la que permite el menor uso de estaciones (1 byte de estaciones), la clase D son para direcciones de tipo multicast y la clase E se ha reservado para uso futuro.

La Figura 5 muestra las 5 clases de direcciones IPv4 existentes.

Figura 5. Clases de direcciones

1.3.1.2.Protocolo de resolución de direcciones (ARP)

El protocolo ARP asocia una dirección lógica con una dirección física. En una red física típica, como una LAN, cada dispositivo conectado a un enlace se encuentra identificado mediante una dirección física normalmente impresa en la NIC. Las direcciones solo tienen jurisdicción local ya que las direcciones pueden ser cambiadas en caso de que ocurra un error con la NIC, por su parte la dirección IP tiene jurisdicción mundial. El funcionamiento del protocolo ARP consiste exclusivamente cuando en una red local se necesita enviar un paquete de un dispositivo A al B, pero el dispositivo A no tiene la dirección física del dispositivo B, por lo tanto el dispositivo A envía un paquete del tipo ARP para preguntar a todos los dispositivos quién contiene la dirección física de la dirección lógica contenida en el paquete enviado, el dispositivo que reconoce la dirección IP envía la contestación con su dirección física.

1.3.1.3.Protocolo de resolución inversa de direcciones (RARP)

El protocolo RARP permite a una estación recuperar su dirección de Internet o lógica a partir de su dirección física. RARP funciona de manera similar al protocolo ARP: una estación que necesita su dirección física envía un paquete RARP, preguntando cuál es su dirección física. Un servidor en la red reconoce el paquete RARP y devuelve la dirección Internet de la estación.

1.3.1.4.Protocolo de mensajes de control de Internet (ICMP)

El protocolo ICMP es utilizado por las estaciones y los enrutadores para enviar mensajes de errores de los datagramas que tienen problemas de llegar a su destino. Como se vio el protocolo IP es inseguro y es por ello que necesita de ICMP para resolver problemas en el envio del datagrama. Un datagrama enviado de un dispositivo viaja a través de diversos enrutadores hasta llegar a su destino final, en el caso que un enrutador antes del receptor del datagrama no logre que el datagrama sea recibido, envía un paquete ICMP indicando al emisor los problemas de la causa, y es responsabilidad del emisor resolver el problema. ICMP utiliza un esquema de prueba/respuesta de eco para probar si un destino es alcanzable y esta respondiendo. También maneja los mensajes de error y de control, pero su única función es informar de problemas, no corregirlos.

1.3.1.5.Protocolo de mensajes de grupos de Internet (IGMP)

El protocolo IP involucra direcciones tipo unicast, broadcast y multicast. Las direcciones unicast son comunicación uno a uno, de un emisor a un receptor; las direcciones broadcast son un tipo de dirección que escuchan todos los elementos involucrados en una segmento; y las de multicast son direcciones de difusión, uno a muchos, esto es, se envía un mismo mensaje a un gran número de receptores simultáneos, este gran número de receptores que escuchan el mensaje forman parte de un grupo. Las direcciones de la clase D (1110) son direcciones de multicast, algunas de éstas se encuentran asignadas permanentemente. El protocolo IGMP se ha diseñado para ayudar a un enrutador con multicast a identificar las estaciones de una LAN que son miembros de un grupo de multicast.

1.3.2. Nivel de trasnporte

El nivel de transporte en TCP/IP está representado por TCP y UDP. UDP es el más simple; ofrece una funcionalidad de transporte que no asegura secuencia, cuando la fiabilidad y la seguridad son menos importantes que el tamaño y la velocidad. La mayoría de las aplicaciones sin embargo, requieren una entrega extremo a extremo fiable y hacen uso de TCP.

Debido que los sistemas operativos son multiusuarios y multiprocesos, es necesario identificar qué proceso es el que recibe un datagrama de muchos; para resolver este problema, TCP/IP define un conjunto de conexiones conceptuales para los procesos individuales denominados puertos del protocolo o sencillamente puertos.

Un puerto es un punto de destino (normalmente buffer) que almacena datos para ser utilizados por un proceso particular. La interfaz entre los procesos y sus puertos correspondientes es ofrecida por el sistema operativo de la estación[4]. En la Figura 6 se muestra el concepto de puerto. Por lo tanto los protocolos de nivel de transporte son protocolos puerto a puerto. Cada puerto es un entero positivo de 16 bits, suficiente para permitir de 0 a 65535 puertos. Los puertos son divididos en tres partes:

- Los bien conocidos son del 0 al 1023
- Los puertos registrados son del 1024 al 49151
- Los dinámicos y/o privados son del 49152 al 65535

Para saber más información acerca de cada puerto, en la página de IANA (Internet Assigned Numbers Authority) se encuentra información actualizada, el link es http://www.iana.org/assignments/port-numbers.

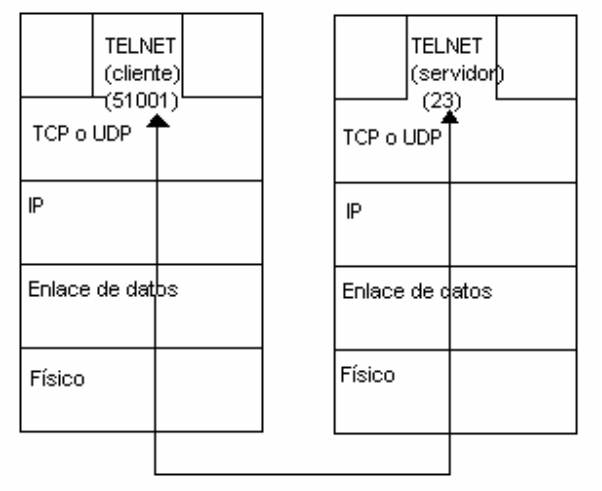


Figura 6. Concepto de puertos.

[4] Behrouz A. Forouzan (2001). Transmisión de datos y redes de comunicaciones. Aravaca, Madrid. McGraw Hill. Segunda Edición. pp. 698.

1.3.2.1.Protocolo de datagramas de usuario (UDP)

UDP proporciona sólo las funciones básicas necesarias para la entrega extremo a extremo de una transmisión. No ofrece funciones de secuenciamiento ni de reordenación y no puede especificar el paquete dañado cuando se informa de un error (por lo que debe usarse con ICMP). UDP puede descubrir que ha ocurrido un error; ICMP puede, a continuación, informar al emisor que un datagrama de usuario se ha dañado o se ha descartado. Tampoco tiene, sin embargo, la capacidad para especificar qué paquete se ha perdido. UDP sólo contiene una suma de comprobación; no contiene un identificador o número de secuencia para un segmento de datos concreto.

A continuación en la Figura 7 se muestra el formato de un datagrama UDP.

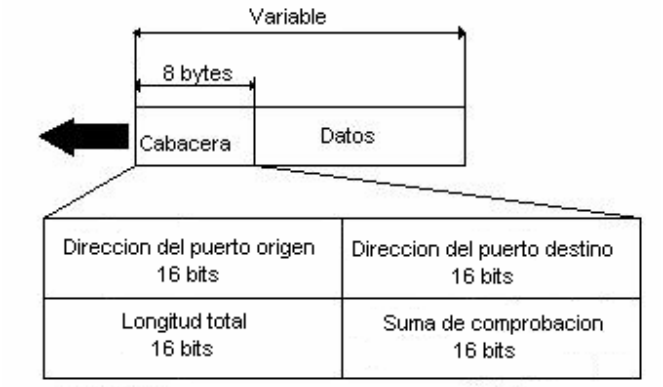


Figura 7. Formato de un datagrama UDP

- **Dirección del puerto origen.** Es la dirección del programa de aplicación que ha creado el mensaje.
- **Dirección del puerto destino.** Es la dirección del programa de aplicación que recibirá el mensaje.
- **Longitud total.** Este campo define la longitud total del datagrama de usuarios en bytes.
- **Suma de comprobación.** Esta suma de comprobación es un campo de 16 bits utilizado para la detección de errores.

1.3.2.2.Protocolo de control de transmisión (TCP)

El protocolo TCP proporciona servicios completos de transporte a las aplicaciones. TCP es un protocolo de transporte puerto a puerto que ofrece un flujo fiable, es orientado a conexión: se establece una conexión entre ambos extremos de la transmisión antes de poder transmitir datos. Al crear esta conexión, TCP genera un circuito virtual entre el emisor y el receptor que se encuentra activo durante la duración de la transmisión. Las conexiones durante la duración de un intercambio entero son diferentes y son manejadas por funciones de sesión en las aplicaciones individuales. TCP inicia cada transmisión informando al receptor que hay datagramas en camino (el establecimiento de la conexión) y finaliza cada transmisión con una terminación de conexión. De esta forma, el receptor conoce la transmisión entera en lugar de un único paquete.

TCP como un servicio orientado a conexión, es responsable de la entrega fiable del flujo entero de bits contenido en el mensaje inicialmente generado por la aplicación emisora. La fiabilidad se asegura mediante la detección de errores y la retransmisión de las tramas con errores; todos los segmentos deben ser recibidos y confirmados antes de que la trasmisión se considere completa y se descarte el circuito virtual.

En el extremo emisor de cada transmisión, TCP divide las transmisiones largas en unidades de datos más pequeñas y empaqueta cada una de ellas en una trama denominada segmento. Cada segmento incluye un número de secuencia para la posterior reordenación de los segmentos en el receptor, junto con un número identificador de confirmación y un campo que indica el tamaño de la ventana deslizante utilizada en las confirmaciones. Los segmentos se transportan por la red dentro de datagramas IP. En el extremo receptor, TCP captura cada datagrama y reordena la transmisión de acuerdo a los números de secuencia.

A continuación en la Figura 8 se muestra el formato del segmento TCP

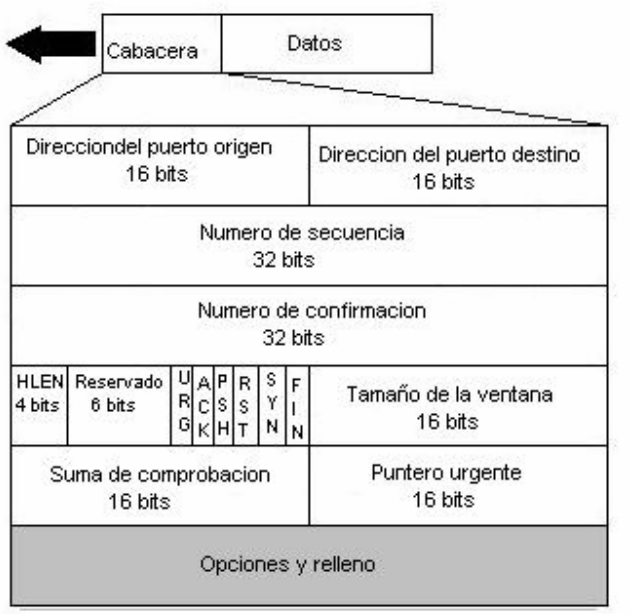


Figura 8. Formato del segmento TCP.

- **Dirección del puerto origen:** esta dirección define el programa de aplicación de la computadora origen.
- **Dirección del puerto destino:** esta dirección define el programa de aplicación de la computadora destino.
- **Número de secuencia:** un flujo de datos del programa de aplicación se puede dividir en dos o más segmentos TCP. El campo con el número de secuencia muestra la posición de los datos en el flujo de datos original.
- **Número de confirmación:** el número de confirmación de 32 bits se utiliza para confirmar la recepción de datos desde el otro dispositivo que participa en la comunicación. Este número es válido sólo si el bit ACK del campo de control

esta activo. En este caso, define el número de secuencia del byte que se espera a continuación.

- **Longitud de la cabecera (LC):** este campo de cuatro bits indica el número de palabras de 32 bits (cuatro bytes) de la cabecera TCP. Los cuatro bits pueden definir hasta 15. Este valor se multiplica por 4 para obtener el número total de bytes de la cabecera. Por lo tanto, el tamaño de la cabecera puede ser hasta de un máximo de 60 bytes (4x15). Puesto que el tamaño mínimo de la cabecera es de 20 bytes, se dispone de 40 bytes disponibles para la sección de opciones.
- **Reservado:** este campo de seis bits se reserva para uso futuro.
- **Control.** Cada bit del campo de control de seis bits funciona de forma individual e independiente. Un bit puede definir el uso de un segmento o servir como una comprobación de la validez de otros campos. El bit urgente, cuando se activa, valida el campo del puntero urgente. Este bit y el puntero indican que los datos del segmento son urgentes. El bit ACK, cuando se activa, valida el campo con el número de confirmación. Ambos se utilizan juntos y tienen funciones diferentes, dependiendo del tipo de segmento. El PSH se utiliza para informar al emisor que se necesita un mayor ancho de banda. Si es posible, los datos deben colocarse en caminos con mayores anchos de banda. El bit RST se utiliza para reiniciar la conexión cuando hay confusión de los números de secuencia. El bit SYN se utiliza para sincronizar los números de secuencia en tres tipos de segmentos: petición de conexión, confirmación de conexión (con el bit ACK activo) y la recepción de confirmación (con el bit ACK activo). El bit FIN se utiliza en la terminación de la conexión en tres tipos de segmentos: petición de terminación, confirmación de terminación (con el bit ACK activo) y confirmación de la confirmación de terminación (con el bit ACK activo).
- **Tamaño de la ventana:** este campo de 16 bits define el tamaño de la ventana deslizante.
- **Suma de comprobación:** este campo de 16 bits se utiliza para la detección de errores.
- **Puntero urgente:** el valor de este campo es válido sólo si esta activo el bit del campo de control URG. En este caso el emisor está avisando al receptor que hay datos urgentes en la porción de datos del segmento. Este puntero define el final de los datos urgentes y el inicio de los datos normales.
- **Opciones y relleno:** este campo se utiliza para enviar información adicional al receptor o para alineamiento.

1.3.3. Nivel de aplicación

Por motivos de el objetivo de este trabajo el ni vel de aplicación será mostrado a grandes rasgos, únicamente nos enfocaremos a tener una noción del funcionamiento de la capa de aplicación del modelo TCP/IP.

El nivel de aplicación de TCP/IP, corresponde a la combinación de los niveles de sesión, presentación y aplicación del modelo OSI; por lo tanto todas las funciones de estos tres niveles son gestionadas en el nivel de aplicación de TCP/IP.

En el nivel de aplicación se encuentran los protocolos de aplicación más comunes, que son utilizados para comunicarse a través de una red con otras aplicaciones y constituyen la interfaz de usuario con la pila de protocolos TCP/IP. La pila TCP/IP incluye algunos protocolos de aplicación, tales como:

- TELNET para el acceso interactivo de una terminal a un host remoto.
- FTP para transferencias de alta velocidad de un disco a otro.
- SMTP es un protocolo sencillo para la transferencia de correo por Internet.
- DNS sistema que sirve para localizar direcciones IP por nombres de Internet o de manera inversa.
- SNMP sirve para gestionar los dispositivos conectados a una red utilizando TCP/IP
- TFTP funciona igual que el protocolo FTP, sólo que es menos robusto, no tiene todas las funcionalidades.

Para poder utilizar los servicios con los que se dispone en Internet es necesario del uso de programas de aplicación ejecutándose en dos computadoras y se comuniquen entre si, es por lo tanto que en Internet los programas de aplicación son los que se comunican entre si, no los usuarios.

El establecimiento de la comunicación entre dos programas de aplicación se logra mediante un concepto denominado cliente-servidor. El modelo a seguir para la comunicación sigue la siguiente estrategia:

- Un programa de aplicación denominado cliente se ejecuta en una maquina local, solicita un servicio a otro programa de aplicación, llamado servidor, que de la misma forma se ejecuta en un maquina remota.
- Un servidor ofrece servicio a cualquier cliente que lo solicite, no sólo a un cliente determinado. La relación cliente-servidor es una relación muchos a uno. Muchos clientes pueden utilizar el servicio de un servidor.
- Un programa cliente, que solicita un servicio, debería ejecutarse sólo cuando es necesario. El programa servidor, que ofrece el servicio, debería estar ejecutándose siempre, debido a que no sabe cuando se va a necesitar el servicio.
- Los servicios utilizados muy frecuentemente por muchos usuarios tienen programas de aplicación cliente-servidor específico. Por lo tanto, se dispone de aplicaciones cliente-servidor para permitir a los usuarios acceder a archivos, enviar correos electrónicos, etc.

Un cliente es un programa que se ejecuta en una maquina local y que solicita un servicio del servidor X. Un programa cliente es finito, lo que significa que es arrancado por un usuario (u otro programa de aplicación) y finaliza cuando el servicio se ha completado.

Un servidor es un programa que se ejecuta en una máquina remota que ofrece un servicio a los clientes. Cuando arranca, abre una puerta para la llegada de las peticiones de los clientes, pero nunca termina hasta que no se le solicite expresamente que lo haga.

Un programa servidor es un programa infinito. Una vez arrancado, se ejecuta indefinidamente a no ser que ocurra un problema. Espera la llegada de peticiones de los clientes. Cuando llega una petición, responde a la misma.

Capítulo

2.Protocolo de Internet versión 6 (IPv6)

Resumen

El capítulo explicará las características básicas del protocolo de Internet versión 6 (IPv6). En la actualidad la versión 4 de este protocolo es el que se encuentra en uso, pero por la necesidad de mejorarlo y tener un mayor espacio de direccionamiento se ha desarrollado la nueva versión 6, actualmente sigue en constante evolución y se empieza a utilizar en diferentes redes como es el caso de las redes avanzadas de Internet a nivel mundial, con el propósito de tener mayor soporte sobre su funcionamiento y de esta forma cuando se tenga que migrar a esta tecnología no sean tomados por sorpresa los administradores de redes.

2.1. HISTORIA DEL PROTOCOLO IPV6

La IETF inició sus esfuerzos para seleccionar un sucesor de IPv4 a finales de 1990, cuando algunas proyecciones indicaban que el uso del espacio de direcciones se incrementaría exponencialmente con lo que se limitaría los recursos. Esfuerzos paralelos empezaron a buscar una forma de resolver la limitación de direcciones y proveer a la vez funciones adicionales. La IETF forma el área llamada IPng a finales de 1993 para investigar cómo proceden varias recomendaciones y propuestas y a la vez hacer recomendaciones para futuros procedimientos.

El área directora de la IETF recomendó la creación de IPv6 en la reunión de Toronto de 1994. Sus recomendaciones están especificadas en el RFC1752, "The Recommendation for the IP Next Generation Protocol". Los directores formaron un grupo de trabajo llamado ALE (Address Lifetime Expectation), su trabajo consistía en determinar si el tiempo de vida de IPv4 podría permitir el desarrollo de un protocolo con nuevas funcionalidades o si el tiempo restante sólo permitiría el desarrollo de una solución para el espacio de direcciones. En 1994, el grupo de trabajo ALE presentó que el agotamiento de direcciones podría ocurrir entre el 2005 y el 2011, basado en las estadísticas disponibles en ese tiempo.

Para febrero de 1992 la comunidad de Internet desarrolló cuatro propuestas separadas para IPng, éstas fueron llamadas CNAT, IP Encaps, Nimrod, y Simple CLNP. Para diciembre de 1992 tres propuestas más le siguieron: PIP (The P Internet Protocol), SIP (The Simple Internet Protocol) y TP/IX. Después en la reunión de San Diego de la IETF en marzo de 1992 "Simple CLNP" introdujo a "TCP y UDP con direccionamiento más grande" (TUBA) y "IP Encaps" introdujo a "Encapsulación de direcciones IP" (IPAE).

Para Noviembre de 1993, IPAE se fusionó con SIP mientras aun mantenía el nombre de SIP. Este grupo entonces se fusionó con PIP y el resultado fue un grupo de trabajo llamado SIPP (Simple Internet Protocol Plus). Al mismo tiempo el grupo de trabajo TP/IX cambió su nombre a CATNIP (Common Architecture for the Internet).

Las propuestas más importantes fueron entonces CATNIP, TUBA y SIPP, de las cuales daremos una breve explicación.

Para más detalle dirigirse a los RFC´s siguientes: para CATNIP en el RFC1707, TUBA en el RFC1347, RFC1526, y el RFC1561, y para SIPP en el RFC1710.

CATNIP

CATNIP fue concebido como un protocolo de convergencia. CATNIP integra CLNP, IP, e IPX. El diseño de CATNIP es provisto para cualquier protocolo de capa de transporte en uso, por ejemplo TP4, CLTP, TCP, UDP, IPX y SPX, corre sobre cualquier formato de los protocolos de la capa de red: CLNP, IP (versión 4), IPX, y CATNIP. Con cierta atención prestada a los detalles, es posible para el protocolo de la capa de transporte (como TCP) operar correctamente con un sistema final usando en la capa de red (por ejemplo IP versión 4) y el otro usando otro protocolo de red, como CLNP.

"El objetivo es proveer compatibilidad entre la Internet, OSI, y protocolos de Novell, así como avanzar con la tecnología de Internet para escalar en el funcionamiento de la siguiente generación de la tecnología de Internet"[1].

CATNIP soporta formato de direcciones NSAP (Network Services Access Point). También usa manejo de caché para proveer rápida identificación del salto siguiente en el funcionamiento de encaminamiento, así como abreviación de la cabecera de red permitiendo que las direcciones sean omitidas cuando un manejo de caché está disponible. La parte fija de la cabecera de la capa de red transporta el manejo de caché.

SIPP

SIPP (Simple Internet Protocol Plus) es una nueva versión de IP el cual se diseña para ser un paso evolutivo para IPv4. Es un incremento natural de IPv4. No fueron metas del diseño tomar medidas radicales lejos de IPv4. Funciones que trabajan en IPv4 fueron mantenidas en SIPP. Funciones que no trabajan fueron removidas. Éste puede ser instalado como actualizaciones normales de software en dispositivos de Internet y operar con el actual IPv4. El desarrollo de la estrategia fue un diseño para no tener días de corte. SIPP esta diseñado para correr sobre funciones altas de red (como ATM) y al mismo tiempo es eficiente para bajos anchos de banda (como wireless). En adición, provee una plataforma para nuevas funcionalidades de Internet que podrían ser requeridas en un futuro no muy lejano.

SIPP incrementa el tamaño de direcciones IP de 32 bits a 64 bits, soporta mas niveles de direccionamiento jerárquico y un gran número de direcciones para nodos. Las direcciones SIPP pueden ser extendidas para un futuro en unidades de 64 bits. SIPP combinó un nuevo tipo de direcciones llamado "cluster addresses" las cuales identifican regiones topológicas más que nodos individuales.

SIPP cambió en la cabecera IP, las opciones en la forma de codificar para permitir eficiente envió, límites menos rigurosos en la longitud de opciones y mejor flexibilidad para introducir nuevas opciones en el futuro. Una nueva capacidad es agregada para habilitar etiquetas de paquetes pertenecientes a un flujo particular para un especial manejo en los envíos de peticiones, como no omitir la calidad de servicio o tiempo real de servicio.

TUBA

Los esfuerzos de TUBA fueron expandir la habilidad de los paquetes en el enrutamiento de Internet usando direcciones con soporte más jerárquico que el actual espacio de direcciones del protocolo de Internet (IP). TUBA especifica el uso continuo de protocolos de Internet de transporte, en particular TCP y UDP, pero especifica su encapsulación en paquetes ISO 8473 (CLNP). Esto permitiría el uso continuo de protocolos de aplicación de Internet como FTP, SMTP, TELNET, etc. TUBA buscó la actualización del sistema actual por una

[1] M. McGovern, R. Ullman (1994). RFC 1707 "CATNIP : common Architecture for the Internet". Informational. Pp 2.

transición del uso de IPv4 a ISO/IEC 8473 (CLNP) y el correspondiente espacio largo de direcciones NSAP.

La oferta de TUBA hacía uso de un simple termino largo de migración, basado sobre una gradual actualización de los Hosts de Internet (que corrieran aplicaciones de Internet sobre CLNP) y servidores DNS (que corrieran direcciones mas largas). Este propósito requería la actualización de enrutadores que soportaran el envío de CLNP (en adición de IP). Sin embargo, este propósito no requería encapsulación o traslación de paquetes o mapeo de direcciones. Direcciones IP y NSAP pueden ser asignadas y usadas independientemente durante el periodo de migración. Encaminamiento y envio de paquetes IP y CLNP pueden ser hechos independientemente.

Después de muchas discusiones en muchos foros y con el consenso de la dirección de IPng, se recomendó que SIPP en su versión de 128 bits, se adoptara como base para IPng, la siguiente generación del protocolo de Internet. El IANA asignó la versión numero 6 a IPng, y el protocolo en si mismo será llamado IPv6. Muchos se preguntan por que le fue asignado de versión el número 6, y qué pasó con el número 5, ya que es la continuación de la versión 4 actual. En la siguiente tabla1 se muestra la asignación de estos números.

Tabla 1. Asignación de las versiones.

Números Asignados	Tecnología	Nota
0-3		No asignados
4	IPv4	Versión actual más extendida de IP
5	ST	Experimental Stream Protocol
6	IPv6	Inicialmente se denominados SIP, SIPP
7	CATNIP	Inicialmente IPv7, TP/IX; caducados.
8	PIP	Caducado
9	TUBA	Caducado
10-15		No asignados

El IESG (Internet Engineering Steering Group) aprobó recomendaciones IPv6 y publicó el draft para propósito de estándar el 17 Noviembre de 1994, y la base del protocolo IPv6 llegó a ser draft estándar el 10 de Agosto de 1998.
Para más detalle sobre el proceso del establecimiento de IPv6 referirse al RFC1752.

2.2. ENCABEZADO DE IPV6

IPv6 es la nueva versión del Protocolo de Internet que vendrá a suceder a IP versión 4. IPv6 introdujo algunos cambios en áreas importantes.

- Expandió el tamaño de las direcciones.
- Simplificó el formato de la cabecera.
- Mejoró las extensiones y opciones.
- Capacidad de etiquetar el flujo.
- Capacidad de autenticación y privacidad.

El expandir el espacio de direcciones IPv6 de 32 bits a 128 bits, significa que IP puede continuar creciendo sin la preocupación del agotamiento de recursos, soporta más niveles

de direccionamiento jerárquico, por lo que ayuda en el mejoramiento de la situación de la eficiencia de encaminamiento.

El simplificar el formato de la cabecera mejora la eficiencia de enrutamiento, porque requiere menos procesamiento los paquetes. Algunos campos del encabezado de IPv4 se hicieron opcionales, lo que reduce el costo de procesamiento y el limite del ancho de banda, sin embargo el mejoramiento en las extensiones y opciones significa la necesidad de poder ser acomodadas sin afectar el funcionamiento de enrutamiento de paquetes normales o de paquetes con especiales necesidades. El etiquetar el flujo agrega un nuevo mecanismo para tratar los paquetes eficientemente, particularmente para aplicaciones de tiempo real. El requerir soporte de IPsec, provee autenticación y privacidad, lo que lo hace un protocolo para uso comercial, ya que emplea especial trato para información sensible.

En IPv6 cinco campos del encabezado IPv4 han sido removidos:

- Longitud de la cabecera.
- Identificación.
- Indicadores.
- Desplazamiento de la fragmentación.
- Suma de comprobación de la cabecera.

La longitud de la cabecera fue removida porque no es necesario con una longitud fija. Los campos de identificación, indicadores y desplazamiento de la fragmentación desaparecieron porque en IPv4 se manejaba la fragmentación de paquetes, en cambio en IPv6 los nodos aprenden el tamaño del MTU (Path Maximum Transmisión Unit) a través de un procedimiento llamado “Path MTU Discovery”. Si un nodo IPv6 quiere fragmentar un paquete, éste hace uso de las cabeceras de extensión, entonces los campos de identificación, indicadores y desplazamiento de la fragmentación serán removidos del encabezado de IPv6 e insertados como una cabecera de extensión.

El campo de suma de comprobación de la cabecera fue removido para mejorar la velocidad de procesamiento, ya que es realizado en el nivel de acceso al medio y también en la capa de transporte (UDP y TCP), por lo tanto es mínimo el riesgo de encontrar errores en el paquete. IP solo hace sus mejores esfuerzos para entregar el paquete, es responsabilidad de los protocolos superiores asegurar la integridad.

El campo de tipo de servicio fue remplazado por el campo de clase de tráfico, es similar al tipo de servicio de IPv4, el campo de TTL fue renombrado por límite de salto (Hop limit) y levemente modificado, y un campo llamado etiqueta de flujo (Flow label) fue agregado para permitir flujo en tiempo real.

El encabezado IPv6 tiene una longitud fija de 40 bytes. Dos campos de las direcciones de origen y destino usan 16 bytes (128 bits) cada uno y 8 bytes para información general del encabezado.
En la figura 9 se muestra el encabezado IPv6 y la comparación con el encabezado IPv4.

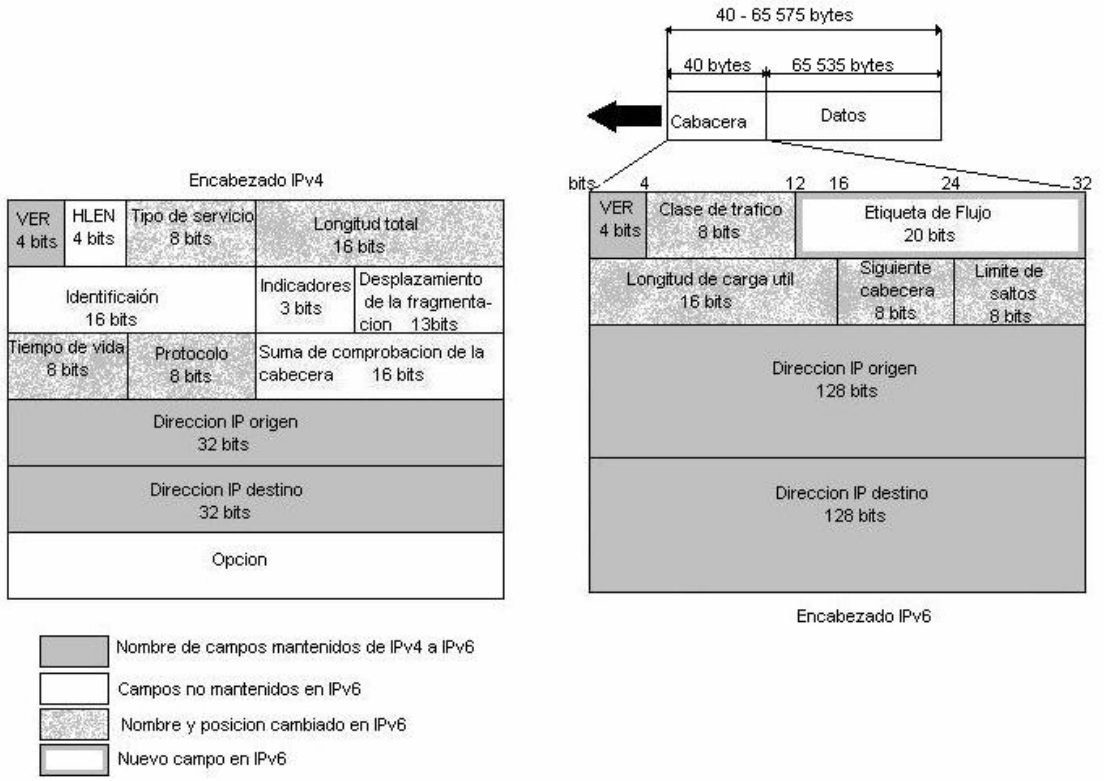


Figura 1. Encabezado IPv6 en comparación con el encabezado IPv4

- **Versión (4 bits):** El primer campo define el número de la versión de IP. La versión es la 6, con un valor binario 0110.
- **Clase de Trafico (8 bits):** este campo facilita el manejo de los datos en tiempo real y otros datos que requieren especial manejo. Este campo puede ser usado para enviar paquetes en nodos y enrutadores, hace la distinción entre diferentes clases y prioridades de paquetes IPv6. El RFC2474 que lleva el nombre de "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers", explica cómo el campo de Clase de Trafico en IPv6 puede ser usado. RFC2474 usa el término DS Field para referirse al campo del encabezado IPv4, así como también al campo de clase de tráfico para el encabezado IPv6.
- **Etiqueta de flujo (20 bits):** este campo distingue paquetes que requieren igual trato, en orden para facilitar el manejo de tráfico en tiempo real. Un nodo emisor puede etiquetar la secuencia de paquetes con un sistema de opciones. Un enrutador puede mantener información de un flujo de datos, para manejar los datos sin necesidad de procesar las cabeceras de los demás paquetes.
- **Longitud de carga útil (16 bits):** este campo especifica la carga útil, es decir los datos transportados después del encabezado IPv6. El cálculo del tamaño en IPv6 es diferente que en IPv4, el campo de longitud en IPv4 incluye la longitud del encabezado IPv4; mientras que en el campo de longitud de carga útil en IPv6 sólo contiene el tamaño de los datos después del encabezado. El hecho de que el campo de la longitud de carga útil sea de 16 bits, limita el tamaño máximo a 65 535 octetos de longitud. IPv6 contiene una cabecera de extensión llamado Jumbograma, el cual soporta tamaño de paquetes grandes entre 65 536 y 4 294 967 295 octetos de

longitud, si es necesario. La opción de carga útil jumbo es relevante sólo para nodos IPv6 que requieran unir enlaces con un MTU mayor a 65 575 octetos (esto es 65 535 + 40, donde 40 octetos es el tamaño del encabezado IPv6). La opción de carga útil Jumbo no necesita ser implementada o entendida por nodos IPv6 que no soporten unir enlaces con MTU mayores que 65 575[2].

- **Siguiente cabecera (8 bits):** en IPv4, este campo es llamado tipo de protocolo, fue renombrado en IPv6 para reflejar la nueva organización de paquetes IP. Si la siguiente cabecera es UDP o TCP, este campo contendría el mismo número de protocolo como en IPv4. Pero si cabeceras de extensión son usados con IPv6, este campo contendrá el tipo de cabecera de extensión siguiente. Estas cabeceras están localizadas entre la cabecera IP y la cabecera TCP o UDP. En la tabla 2 se enlista algunos posibles valores que podría tener el campo de la siguiente cabecera. La lista completa se encuentra en la página del IANA en el link http://www.iana.org/assignments/protocol-numbers.

Tabla 2. Numero de Protocolos de Internet Asignados

Numero	Protocolo	Referencia
0	Option IPv6 Hop by Hop	RFC1883
1	Internet Control Message	RFC792
2	Internet Group Management	RFC1112
4	IP in IP (encapsulación)	RFC2003
6	TCP Transmission Control	RFC793
8	Exterior Gateway Protocol (EGP)	RFC888,DLM1
9	IGP (usado por CISCO para sus IGRP)	IANA
17	UDP User Datagram	RFC768,JBP
41	IPv6	Deering
43	Routing Header para IPv6	Deering
44	Fragment Header para IPv6	Deering
45	Interdomain Routing Protocol (IDRP)	Sue Hares
46	Reservation Protocol (RSVP)	Bob Bramen
50	Encap Security Payload (ESP)	RFC2406
51	Authentication Header	RFC2402
58	ICMPv6	RFC1883
59	No Next Header para IPv6	RFC1883
60	Destination Options para IPv6	RFC1883
88	EIGRP	CISCO,GXS
89	OSPIGP	RFC1583,JTM4
108	IP Payload Compression Protocol	RFC2393
115	Layer Two Tunneling Protocol (L2TP)	Aboba
121	SMP	Ekblab
132	Stream Control Transmission Protocol (SCTP)	Stewart
135	Mobility Header	RFC3775
137	MPLS in IP	RFC4023
138-252	No asignado	IANA
253 y 254	Usado para experimentación y prueba	RFC3692
255	Reservado	IANA

- **Límite de saltos (8 bits):** este campo es similar al campo tiempo de vida en IPv4. el campo tiempo de vida contiene un número de segundos, indicando cuánto tiempo

[2] D. Borman, S. Deering, R. Hinden (1999). RFC2675 “IPv6 Jumbograms”. Standards Track. Pp. 0

un paquete puede permanecer en la red antes de ser destruido. La mayoría de los enrutadores simplemente decrementa este valor en una unidad en cada salto. Este campo fue renombrado a límite de saltos en IPv6. el valor en este campo ahora expresa un número de saltos y no un número de segundos. Cada envio de un nodo decrementa el número en una unidad. El paquete es desechado si el campo de límite de salto es decrementado a cero.

- **Dirección IP origen (128 bits):** este campo contiene la dirección IPv6 del nodo que origina el paquete.
- **Dirección IP destino (128 bits):** este campo contiene la dirección IPv6 del receptor previsto del paquete. Esta dirección puede ser unicast, multicast o anycast. Con IPv4, este campo siempre contenía la dirección del último destino del paquete. Con IPv6, este campo podría no contener la dirección IPv6 del ultimo destino si la cabecera de enrutamiento (Routing Header) esta presente.

2.2.1. Encabezados de extensión

Los encabezados IPv4 pueden ser extendidos en orden de 20 a 60 bytes para opciones específicas. Esta capacidad raramente ha sido usada porque causa alto desempeño. IPv6 tiene una nueva forma de repartir el proceso con opciones mejorando el procesamiento, esto se logra mediante el manejo de opciones dentro de cabeceras adicionales llamadas cabeceras de extensión.

La actual especificación de IPv6 (RFC2460), define seis cabeceras de extensión:

- Encabezado de opción salto por salto (Hop by Hop Options Header)
- Encabezado de Enrutamiento (Routing Header)
- Encabezado de Fragmentación (Fragment header)
- Encabezado de opción de Destino (Destination Options header)
- Encabezado de Autenticación (Authentication header)
- Encabezado de Encapsulado y Seguridad de la Carga Útil (Encapsulating Security Payload header)

Pueden ser una o más cabeceras de extensión entre el encabezado IPv6 y el encabezado del protocolo de capa superior. Cada cabecera de extensión es identificada por el campo de siguiente cabecera en la cabecera precedente. Las cabeceras de extensión son examinadas o procesadas sólo por el nodo identificado en el campo de dirección IP destino del encabezado IPv6. Si la dirección en el campo de dirección IP destino es una dirección multicast, las cabeceras de extensión son examinadas y procesadas por todos los nodos pertenecientes a el grupo multicast. Las cabeceras de extensión deben ser estrictamente procesadas en el orden que aparecen en el encabezado del paquete.

Hay una excepción en la regla: sólo el nodo destino procesará la cabecera de extensión. Si la cabecera de extensión es una cabecera opción salto por salto, la información transportada debe ser examinada y procesada por cada nodo a lo largo del camino del paquete. La cabecera opción salto por salto, si esta presente, debe ser seguida inmediatamente del encabezado IPv6, esta es indicada por el valor cero en el campo de siguiente cabecera del

encabezado IPv6 (este valor se muestra en la Tabla 2). Las primeras cuatro cabeceras de extensión son descritas en el RFC2460, la cabecera de Autenticación se describe en el RFC2402 y la cabecera de encriptación y seguridad de la carga útil en el RFC2406.

La Figura 10 muestra como las cabeceras de extensión son usadas.

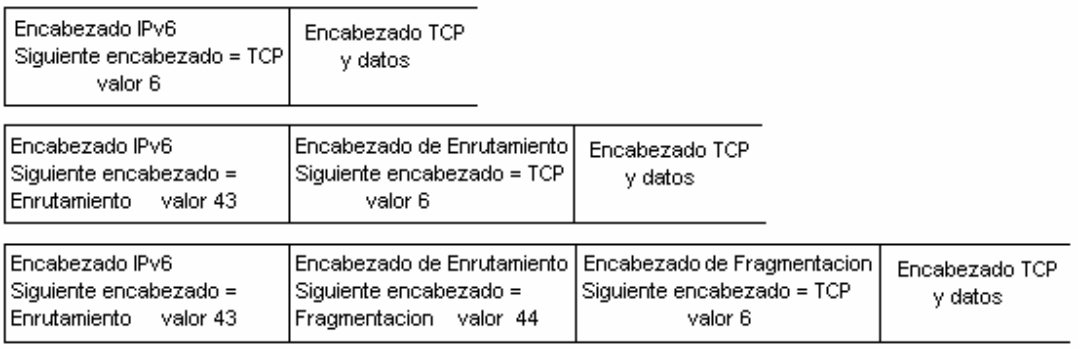


Figura 2. Uso de los encabezados de extensión

Cada cabecera de extensión es un múltiplo de 8 octetos de largo. De esta forma, subsecuentes cabeceras pueden siempre ser alineadas. Si un nodo requiere procesar la siguiente cabecera, pero no identifica el valor del campo de la siguiente cabecera, entonces se descarta el paquete y se envía un mensaje hacia el origen del paquete con un mensaje de problema de parámetro ICMPv6.

Cuando más de una cabecera de extensión es usada en el mismo paquete, se recomienda que las cabeceras aparezcan en el orden siguiente:

1. Encabezado IPv6.
2. Encabezado de opción salto por salto.
3. Encabezado de opción de destino (nota a)
4. Encabezado de enrutamiento.
5. Encabezado de fragmentación.
6. Encabezado de autenticación
7. Encabezado de encapsulado y seguridad de la carga útil
8. Encabezado de opción de destino (nota b)
9. Encabezado de capa superior.

Notas:

a) Para opciones a ser procesadas por el primer destino que aparece en el campo de dirección IP destino más los subsecuentes destinos listados en el encabezado de enrutamiento.
b) Para opciones a ser procesadas sólo por el destino final del paquete.

Cada encabezado de extensión debe ocurrir sólo una vez, excepto para el encabezado de opción de destino el cual debe de ocurrir a lo mucho dos veces (una ves antes del encabezado de enrutamiento y una vez antes del encabezado de capa superior).

En el caso de que IPv6 sea encapsulado en IPv4, el encabezado superior puede ser otro encabezado IPv6 y puede contener encabezados de extensión con las mismas reglas.

El valor de 59 en el campo de siguiente encabezado de un encabezado IPv6 o cualquier encabezado de extensión, indica que ya no hay más encabezados que sigan a este encabezado. Si el campo de longitud de carga útil del encabezado IPv6 indica la presencia de bytes más allá del final de un encabezado en el cual su campo de encabezado siguiente contenga el valor de 59, estos bytes deben ser ignorados, y pasar sin cambios si el paquete se vuelve a enviar.

2.2.1.1.Opciones

Dos de los encabezados de extensión (encabezado de opción salto por salto y el encabezado de opción de destino), transportan un número variable TLV (tipo-longitud-valor), con el formato mostrado en la Figura 11.

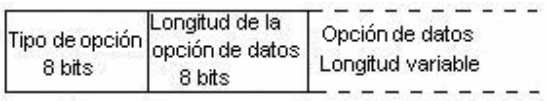


Figura 3. Formato de opciones

- **Tipo de opción (8 bits):** el identificador de tipo de opción, internamente se codifica de tal forma que los dos bits de mayor orden especifican la acción que debe ser tomada si el procesamiento del nodo IPv6 no reconoce el tipo de opción:
  - 00 – salta esta opción y continúa procesando el encabezado.
  - 01 – descarta el paquete.
  - 10 – descarta el paquete, y sin importar si la dirección de destino es o no una dirección de multicast, envía un mensaje ICMP (problema de parámetro) con código 2 a la dirección fuente del paquete, señalando que el tipo de opción es desconocido.
  - 11 – descarta el paquete y, sólo si la dirección destino del paquete no es una dirección multicast, envía un mensaje ICMP (problema de parámetro) con código 2 a la dirección fuente del paquete, señalando que el tipo de opción es desconocido.

  El tercer bit de el campo tipo de opción especifica si la opción de información puede cambiar en-route (1), al paquete final de destino, o no puede cambiar en-route (0).

- **Longitud de la opción de datos (8 bits):** entero sin signo. Longitud del campo opción de datos de esta opción, en octetos.

- **Opción de datos (Longitud variable):** Opción-tipo-dato específico.

La secuencia de opciones dentro de un encabezado debe ser procesada estrictamente en el orden que aparece en el encabezado; un receptor no debe explorar a través del encabezado

buscando un tipo de opción particular y procesar esta opción antes de procesar todas las opciones precedentes.

2.2.1.2.Encabezado de opción salto por salto

El encabezado de opción salto por salto es usado para transportar información que debe ser examinada por cada nodo a lo largo de la trayectoria de la entrega de un paquete. El encabezado de opción salto por salto es identificado por un valor en el campo de cabecera siguiente de cero "0" en el encabezado de IPv6.

Este encabezado siempre aparece inmediatamente después del encabezado IPv6 y contiene datos opcionales que cada nodo a lo largo del camino, debe examinar. Hasta ahora dos tipos de opción salto por salto han sido especificadas: la opción de carga útil jumbo y la opción de alerta de enrutador.

La opción de carga útil jumbo identifica la carga útil de los paquetes que son más grandes a 65,535 octetos (incluyendo el encabezado de opción salto por salto). Si un enrutador esta deshabilitado para manejar jumbogramas, éste regresa un mensaje de error ICMPv6.

La otra opción de salto por salto es la opción de alerta de enrutador. Éste es usado para notificar a los enrutadores que la información dentro de los datagramas IPv6 están previsto para ser vistos y procesados por un enrutador intermedio. El formato del encabezado de opción salto por salto se muestra en la Figura 12.

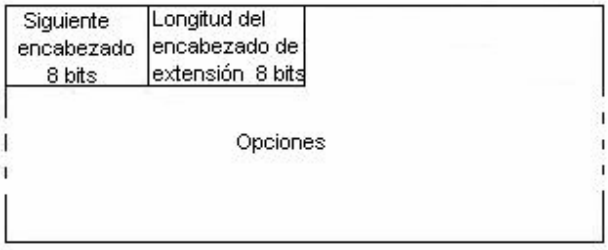


Figura 4. Encabezado de opción salto por salto

- **Siguiente encabezado (8 bits):** identifica el tipo de encabezado inmediato, que sigue al encabezado de opción salto por salto. Usa los mismos valores que el campo de protocolo en el encabezado IPv4 mencionados en la Tabla 2.
- **Longitud del encabezado de extensión (8 bits):** identifica la longitud del encabezado de opción salto por salto en unidades de 8 octetos, no incluye los primeros 8 octetos.
- **Opciones (Longitud variable):** la longitud de la opción es variable y se determina en el campo de longitud del encabezado de extensión. Contiene una o mas opciones TLV-codificadas, como se describe en la sección 2.3.1.1.

2.2.1.3.Encabezado de enrutamiento

El encabezado de enrutamiento es usado por un nodo origen IPv6 para listar uno o más nodos intermedios a ser visitados sobre el camino de un paquete hacia su destino. Esta función es muy similar en IPv4 a "Loose Source" y "Record Route option". El encabezado de enrutamiento es identificado por el número 43 en el campo de encabezado siguiente del encabezado precedente inmediato, y tiene el formato como se muestra en la Figura 13.

Figura 5. Formato del encabezado de enrutamiento

- **Siguiente encabezado (8 bits):** identifica el tipo de encabezado inmediato, que sigue al encabezado de enrutamiento. Usa los mismos valores que el campo de protocolo en el encabezado IPv4 mencionados en la Tabla 2.
- **Longitud del encabezado de extensión (8 bits):** identifica la longitud del encabezado de enrutamiento en unidades de 8 octetos, no incluye los primeros 8 octetos.
- **Tipo de enrutamiento (8 bits):** identificador de un encabezado particular de enrutamiento variante.
- **Segmentos izquierdos (8 bits):** entero sin signo. Número de segmentos de enrutador restantes, por ejemplo, número explícitamente listado de nodos intermedios que todavía faltan ser visitados antes de alcanzar el destino final.
- **Tipo específico de datos (Longitud variable):** la longitud de este campo depende del tipo de enrutamiento. La longitud siempre será el encabezado completo en múltiplos de 8 octetos.

Si, mientras se procesa un paquete recibido, un nodo encuentra un encabezado de enrutamiento con un desconocido valor de tipo de enrutamiento, el comportamiento requerido del nodo depende del valor del campo de segmentos izquierdos, como sigue:

- Si el segmento izquierdo es cero, el nodo debe ignorar el encabezado de enrutamiento y proceder a procesar la siguiente cabecera en el paquete, el cual es identificado por el campo de siguiente cabecera en el encabezado de enrutamiento.
- Si el segmento izquierdo no es cero, el nodo debe descartar el paquete y enviar un mensaje ICMP (problema de parámetro) con código 0 a la dirección fuente del paquete, señalando que el tipo de enrutamiento es desconocido.

Si después de procesar un encabezado de enrutamiento de un paquete recibido, un nodo intermedio determina que el paquete será enviado sobre un enlace en el cual el tamaño del

MTU del enlace es menor que el tamaño del paquete, el nodo debe descartar el paquete y enviar un mensaje ICMP (paquete demasiado grande) a la dirección origen del paquete.

El encabezado de enrutamiento tipo 0 tiene el formato que se muestra en la Figura 14.

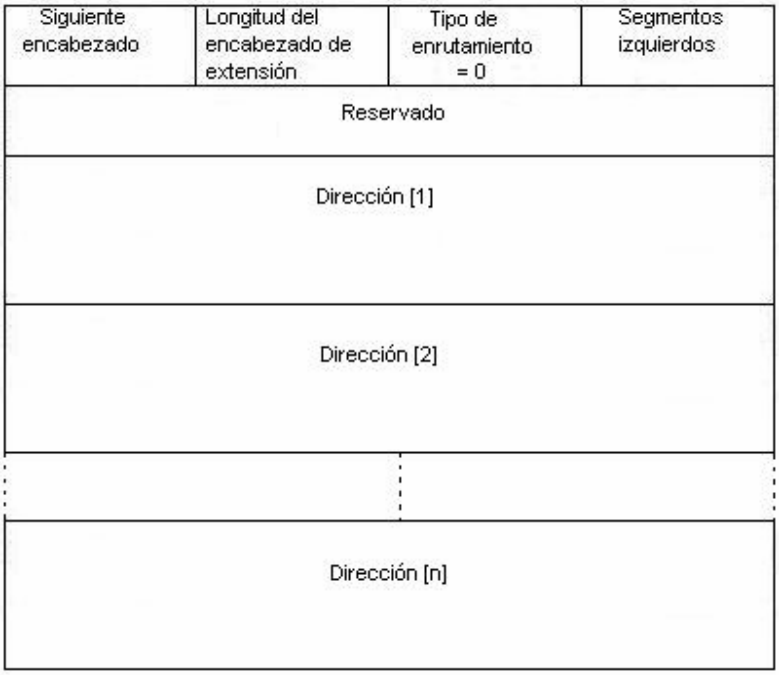


Figura 6. Formato del encabezado de enrutamiento tipo 0.

El único tipo de enrutamiento descrito en el RFC2460 es un tipo cero "0". El primer nodo que procesa el encabezado de enrutamiento es el nodo direccionado por el campo de dirección IP destino en el encabezado IPv6. Este nodo decrementa el campo de segmentos izquierdos en uno e inserta el campo de la siguiente dirección proveniente dentro del encabezado de enrutamiento en el campo de dirección IP destino del encabezado IPv6. Entonces el paquete es enviado al siguiente salto que de nuevo realizará el proceso del encabezado de enrutamiento como se describe, hasta que el destino final sea alcanzado. El destino final es la última dirección en el campo de datos del encabezado de enrutamiento.

Direcciones multicast no deben aparecer en los encabezados de enrutamiento tipo cero "0", o en el campo de dirección IP destino de un paquete transportando un encabezado de enrutamiento de tipo cero.

2.2.1.4.Encabezado de fragmentación

Un nodo IPv6 que quiere enviar un paquete a un destino IPv6 usa descubriendo el MTU de la trayectoria (Path MTU discovery) para determinar el máximo tamaño del paquete que puede ser usado sobre el camino al destino. Si el paquete a ser enviado es más grande que el soportado por el MTU, entonces el host origen fragmenta el paquete. A diferencia de IPv4, con IPv6, un paquete no se fragmenta por un enrutador a lo largo del camino. La fragmentación sólo ocurre en el host origen que envía el paquete. El host destino maneja el

reensamblado. El encabezado de fragmentación es identificado por el campo de siguiente encabezado con un valor de 44 del encabezado precedente.
El formato del encabezado de fragmentación es mostrado en la Figura 15.

Figura 7. Formato del encabezado de fragmentación.

- **Siguiente encabezado (8 bits):** identifica el tipo inicial de encabezado de la parte fragmentada del paquete original. Usa los mismos valores que el campo de protocolo en el encabezado IPv4 mencionados en la Tabla 2.
- **Reservado (8 bits):** se inicializa en cero para la transmisión; se ignora en la recepción.
- **Compensación del fragmento (15 bits):** la compensación, en unidades de 8 octetos, de los datos que siguen este encabezado, relativo al comienzo de la parte fragmentable del paquete original.
- **Res (2 bits):** no usado, se inicializa en cero para la transmisión; se ignora en la recepción.
- **Bandera M (1 bit):** indica 1 = mas fragmentos; 0 = último fragmento.
- **Identificación (32 bits):** generado por el host origen para identificar todos los paquetes pertenecientes al paquete original, este identificador debe ser diferente que cualquier otro paquete fragmentado enviado recientemente con la misma dirección origen y destino. Este campo es usualmente implementado como un contador, se incrementa en uno por cada paquete que necesita ser fragmentado por el host origen.

El paquete grande inicial no fragmentado se refiere como el "paquete original", y éste se considera que consiste en dos partes, como se muestra en la Figura 16.

Figura 8. Paquete original.

La parte no fragmentada consiste en el encabezado IPv6 más cualquier encabezado de extensión que debe ser procesado por nodos en-route al destino, esto es, todos los encabezados, incluso el encabezado de enrutamiento si está presente, el encabezado de opciones salto por salto si está presente, o si no hay encabezados de extensión.

La parte fragmentable del paquete original es dividido en fragmentos. Los fragmentados son transmitidos es separados "paquetes fragmentados", como se ilustra en la Figura 17.

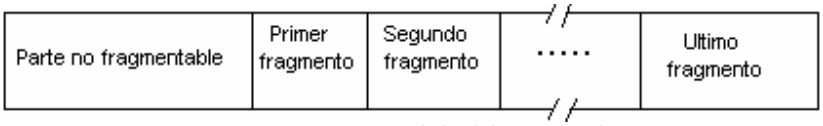


Figura 9. Paquete original fragmentado

Cada paquete fragmentado es compuesto como se muestra en la Figura 18.

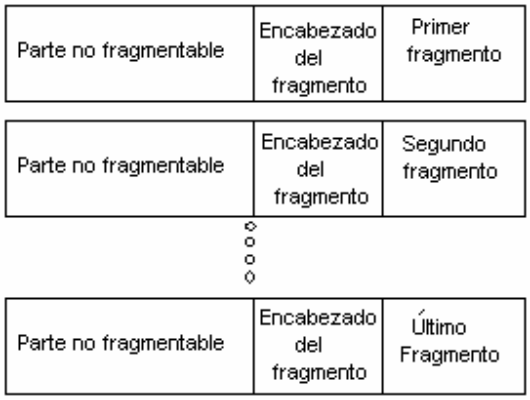


Figura 10. Paquetes fragmentados.

La parte no fragmentada del paquete original aparece en cada paquete fragmentado, seguido por un encabezado de fragmentación, y después datos fragmentados. El encabezado IPv6 del paquete original ha sido modificado levemente. El campo de longitud refleja la longitud del fragmento (excluyendo el encabezado IPv6) y no la longitud del paquete original.

El nodo destino recolecta todos los fragmentos y los reensambla. El fragmento debe tener idénticas direcciones de origen y destino, así como el mismo valor de identificación en orden para ser reensamblado. Si todos los fragmentos no llegan al destino en 60 segundos después del primer fragmento, el destino descartará todos los paquetes. Si el destino ha recibido el primer fragmento (compensación = 0), éste debe enviar un mensaje ICMPv6 (Fragment Reassembly Time Exceeded) al origen.

2.2.1.5.Encabezado de opción de destino

Un encabezado de opción de destino transporta información opcional que es examinada sólo por el nodo destino. El valor de siguiente cabecera identifica este tipo de encabezado con el valor 60. La Figura 19 muestra el formato del encabezado de opción de destino.

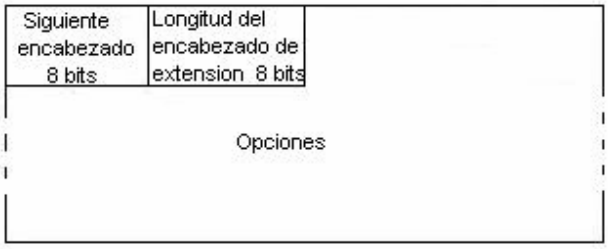


Figura 11. Formato del encabezado de opción de destino.

- **Siguiente encabezado (8 bits):** identifica el tipo de encabezado inmediato, que sigue al encabezado de opción de destino. Usa los mismos valores que el campo de protocolo en el encabezado IPv4 mencionados en la Tabla 2.

- **Longitud del encabezado de extensión (8 bits):** identifica la longitud del encabezado de opción de destino en unidades de 8 octetos, no incluye los primeros 8 octetos.
- **Opciones (Longitud variable):** la longitud de la opción es variable y se determina en el campo de longitud del encabezado de extensión. Contiene una o más opciones TLV-codificadas, como se describe en la sección 2.3.1.1.

Nota que hay dos posibles formas de codificar información opcional de destino en un paquete IPv6: como una opción en el encabezado de opción de destino, o como un encabezado de extensión separado. El encabezado de fragmentación y el encabezado de autenticación son ejemplos de últimos acercamientos. Tales acercamientos se pueden usar dependiendo de qué acción se desee de un nodo destino, que no entiende la información opcional[3].

- Si la acción es para que el nodo destino deseche el paquete (solamente si la dirección del paquete destino no es una dirección multicast), y que envié un paquete ICMP (tipo de mensaje no reconocido) a la dirección origen, entonces la información puede ser codificada como un encabezado separado o como una opción en el encabezado de opción de destino, la cual tiene el tipo de opción de valor 11 en sus dos bits de mayor orden. La opción puede depender de los factores tales como tomar pocos octetos, o que rinda una alineación mejor a un análisis más eficiente.
- Si se decide cualquier otra acción, la información debe ser codificada como una opción en el encabezado de opción de destino, la cual tiene un valor de tipo 00, 01, o 10 en los dos bits de mayor orden, especificando la acción deseada, como se describe en la sección 2.3.1.1.

2.3. ARQUITECTURA DE DIRECCIONAMIENTO

Una de las diferencias obvias entre IPv4 e IPv6 es el espacio de direcciones, IPv4 sólo maneja una longitud de 32 bits, mientras que IPv6 usa 128 bits de longitud; además hay más formas diferentes de usar estas direcciones; para usuarios normales muchas de las aplicaciones que trae IPv6 le son invisible, pero lo que haría notar la diferencia de una versión a otra es su espacio de direcciones mayor.

2.3.1. Tipos de Direcciones

Los tipos de direcciones reconocidos por IPv4 son unicast, broadcast, y multicast. Con IPv6, el broadcast ha desaparecido, en lugar de este tipo de dirección quedó multicast y anycast. Esta es una buena noticia ya que el broadcast ha sido un problema en muchas

[3] S. Deering, R. Hinden (1998). RFC2460 “Internet Protocol, Version 6 (IPv6) Specification”. Standards Track. Pp. 22.

redes. El nuevo tipo de direcciones usado por IPv6 es anycast, ésta fue introducida por el RFC1546.

Las direcciones IPv6 son identificadores de 128 bits para interfaces y sistemas de interfaces y se clasifican en tres tipos:

- **Unicast:** un identificador para una sola interfaz de un nodo IPv6. Un paquete enviado a una dirección unicast es liberado por la interfaz identificada por esta dirección.
- **Anycast:** un identificador para un sistema de interfaces (típicamente pertenecen a nodos diferentes). Un paquete enviado a una dirección anycast es liberado por una de las interfaces identificada por esta dirección (la interfaz más cercana, de acuerdo a las medidas de distancia de los protocolos de enrutamiento).
- **Multicast:** un identificador para un sistema de interfaces (típicamente pertenecen a nodos diferentes). Un paquete enviado a una dirección multicast es liberado por todas las interfaces identificadas por esta dirección[4].

2.3.2. Algunas reglas generales

Direcciones IPv6 de todos los tipos son asignadas a interfaces, como en IPv4, no a nodos, como en OSI. Una dirección unicast IPv6 se refiere a una sola interfaz; debido a que cada interfaz pertenece a un solo nodo, cualquiera de las direcciones unicast de las interfaces de ese nodo, se puede utilizar como identificador del nodo.

Todas las interfaces requieren tener por lo menos una dirección unicast de enlace-local (link-local). Una sola interfaz puede tener asignada múltiples direcciones IPv6 de cualquier tipo (unicast, anycast, y multicast) o ámbito. Es también posible asignar una dirección unicast a múltiples interfaces por razones de compartir-carga (load-sharing), pero si se hace esto, se debe estar seguro que el hardware y los controladores lo soportan. Con IPv6 son valores legales tener en los campos puros ceros o unos, excluyendo algunas combinaciones especiales que dependen de tipos de direcciones (por ejemplo, RFC2373 “Subnet Router anycast address”).

Actualmente IPv6 continúa usando el modelo de IPv4, en el cual asocia prefijos de subred con un enlace. Múltiples prefijos de subred pueden ser asignadas al mismo enlace.

2.3.3. Representación de las direcciones IPv6

En IPv6 existen tres formas convencionales de representar las direcciones como secuencias de texto:

1. el método preferido para representarlas es mediante una secuencia de ocho valores de 16 bits, separada por dos puntos, en notación hexadecimal.

[4] R. Hinden, S. Deering (1998). RFC2373 “IP Version 6 Addressing Architecture”. Standards Track. Pp. 1.

xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx

Nótese que una "x" representa un valor hexadecimal de 4 bits, a continuación se muestra dos ejemplos de direcciones IPv6:

FEDC:BA98:7654:3210:FEDC:BA98:7654:3210

1080:0:0:0:8:800:200C:417A

En la dirección IPv6 del segundo ejemplo, se muestra una de las características importantes de las direcciones, ya que no es necesario escribir una cadena de ceros en un campo individual, únicamente con poner un sólo cero se sobreentiende que el campo contiene una cadena de ceros.

2. debido a algunos métodos de asignar ciertos estilos de direcciones IPv6, es común para las direcciones contener largas secuencias de bits puestas en cero. Se ha establecido una sintaxis especial que haga más fácil la escritura de los ceros, esto es mediante la compresión de grupos de ceros. El uso de "::" indica uno o mas grupos de 16 bits puestos en cero. El "::" puede solo aparecer una vez en una dirección, la tabla 3 nos muestra ejemplos de tipos de direcciones en su representación estándar y comprimida.

Tabla 3. Representación de direcciones IPv6 estándar y comprimidas.

Tipo de dirección	**Representación estándar**	**Comprimida**
Dirección unicast	1080:0:0:0:8:800:200C:417A	1080::8:800:200C:417A
Dirección Multicast	FF01:0:0:0:0:0:0:101	FF01::101
Dirección Loopback	0:0:0:0:0:0:0:1	::1
Dirección no especificada	0:0:0:0:0:0:0:0	::

3. en ambientes en donde nodos IPv4 e IPv6 son mezclados, una alternativa más conveniente es poner los valores de una dirección IPv4 dentro de los cuatro bytes de menor orden de la dirección, y los seis bytes de mayor orden son valores hexadecimal. En la tabla 4 se muestra un ejemplo de esta representación. El formato es el siguiente:

xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:d.d.d.d

Tabla 4. Ejemplo de mezclar direcciones IPv4 con IPv6

Dirección IPv4	192.168.0.2
Dirección mezclada	0:0:0:0:0:0:192.168.0.2
Comprimida	::192.168.0.2
Notación hexadecimal	::C0A8:2

2.3.4. Notación de prefijos.

Un prefijo son los bits de mayor orden de una dirección IP, usados para identificar la subred o un tipo de dirección específica, también es llamado como prefijo global de enrutamiento (global routing prefix). La notación de los prefijos es muy similar a la forma

de direcciones IPv4 en CIDR (Classless Interdomain Routing), y también es comúnmente usado para subnetear direcciones IPv4. Un prefijo de dirección IPv6 es representado por la notación:

Dirección IPv6/Longitud de prefijo

La dirección IPv6 puede estar en cualquiera de sus representaciones mostradas en la sección 2.4.3. La longitud de prefijo especifica cúantos bits a la izquierda específica el prefijo. Esta es otra forma de notación de una mascara de subred. Hay que recordar, una mascara de subred especifica los bits de las dirección IPv4 que pertenecen al ID de red (Network ID). Los prefijos son usados para identificar la subred a la que una interfaz pertenece y también es usado por los enrutadores para el enrutamiento. Para entender esto, el siguiente ejemplo explica cómo interpretar los prefijos.

Por ejemplo considere la siguiente notación IPv6 del prefijo 2E78:DA53:1200::/40, para entender esto hay que convertir la notación hexadecimal a binario, esta conversión se realiza en la tabla 5.

Tabla 5. Conversión de notación hexadecimal a binario.

Notación hexadecimal	Notación Binaria	Numero de Bits
2E78	0010 1110 0111 1000	16 bits
DA53	1101 1010 0101 0011	16 bits
12	0001 0010	8 bits, total de 40 bits

El resultado nos indica que la longitud del prefijo es de 40 bits, en el siguiente ejemplo se presenta el prefijo de subred con la combinación de la dirección de un nodo:

Dirección del nodo	12AB:0:0:CD30:123:4567:89AB:CDEF
Subred	12AB:0:0:CD30::/60
Abreviado como	12AB:0:0:CD30:123:4567:89AB:CDEF/60

2.3.5. Identificación de los tipos de direcciones

Los tipos de direcciones son identificados por los bits de mayor orden de la dirección para identificar direcciones especiales, como direcciones de enlace-local o multicast, a continuación se muestra la tabla 6 que identifica los tipos de direcciones IPv6 reconocidos por el RFC3513.

Tabla 6 . Tipo de direcciones IPv6 (RFC3513) [5]

Tipo de dirección	Prefijo en binario	Notación IPv6
No especificada	00…0 (128 bits)	::/128
Loopback	00…1 (128 bits)	::1/128
Multicast	11111111	FF00::/8
Enlace-Local Unicast	1111111010	FE80::/10
Sitio-Local Unicast	1111111011	FEC0::/10 (Desaprobada por la IETF)

[5] R. Hinden, S. Deering (2003). RFC3513 “Internet Protocol Version 6 (IPv6) Addressing Architecture”. Standards Track. Pp. 5.

Única Local Unicast	1111110	FC00::/7 (RFC 4193)
Global Unicast	001	2000::/3

Direcciones Anycast son tomadas del espacio de direcciones unicast (de cualquier ámbito) y su sintaxis no es distinguible de direcciones unicast.

2.3.5.1.Direcciones Unicast

Existen varios tipos de direcciones unicast en IPv6, éstas son en particular la global unicast, sitio-local unicast (site-local), y enlace-local unicast (link-local). Hay también algunas de propósito especial que son subtipos de la global unicast, tal como direccionamiento IPv6 incrustada en direcciones IPv4 (IPv6 embedded IPv4), o codificada en direcciones NSAP. Tipos de direcciones adicionales o subtipos pueden ser definidas en el futuro.

Nodos IPv6 pueden tener poco o considerable conocimiento de la estructura interna de una dirección IPv6, dependiendo del rol que el nodo juegue (nodo simple o enrutador). Mínimamente un nodo debe considerar que la dirección unicast (incluyendo la suya) no tiene estructura interna, lo que significa que debe verse como se muestra en la figura 20.

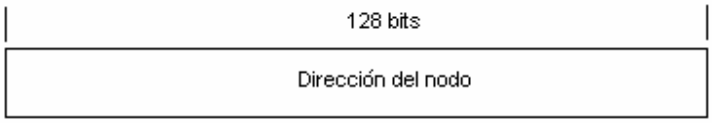


Figura 12. Dirección unicast IPv6 reconocida por un nodo.

Un host levemente sofisticado (pero aún algo simple) puede estar enterado del prefijo de subred para el enlace, donde las direcciones pueden tener diferentes valores n del tamaño de los bits en el prefijo de subred, como se muestra en la figura 21.

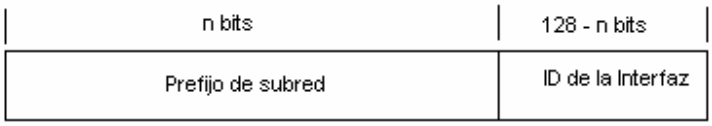


Figura 13. Dirección unicast IPv6 reconocida por un nodo un poco más sofisticado.

Aunque un enrutador muy simple no pueda tener conocimiento de la estructura interna de una dirección unicast IPv6, los enrutadores tienen más conocimiento general de uno o más límites de jerarquía para la operación de los protocolos de enrutamiento. Los límites conocidos diferirán de enrutador a enrutador, dependiendo de la posición que mantengan en la jerarquía de enrutamiento.

2.3.5.1.1. Identificadores de Interfaz (ID interfaz)

Los identificadores de interfaz en las direcciones unicast IPv6 son usados para identificar interfaces en un enlace. Se requiere que sean únicos dentro de un prefijo de subred. Se recomienda que el mismo identificador de interfaz no sea asignado a diferentes nodos en un enlace. Pueden también ser únicos en un ámbito más amplio. En algunos casos un identificador de interfaz será derivado directamente de la interfaz de la dirección de la capa

de enlace. El mismo identificador de interfaz puede ser usado en múltiples interfaz de un solo nodo, mientras se unen a diferentes subredes.

Observe que la unicidad de los identificadores de interfaz es independiente de la unicidad de las direcciones IPv6. Por ejemplo, una dirección global unicast puede ser creada con un identificador de interfaz de ámbito no global, y una dirección de sitio-local puede ser creada con un identificador de interfaz de ámbito global.

Para todas las direcciones unicast, excepto aquellas que empiecen con el valor binario 000, se requiere que los identificadores de interfaz sean de 64 bits de largo y sean construidos con el formato modificado EUI-64[6].

Identificadores de interfaz basados en el formato modificado EUI-64 pueden tener ámbito global cuando son derivados de un símbolo global (por ejemplo, IEEE 802 con 48-bits de la MAC o identificadores IEEE EUI-64) o pueden tener ámbito local cuando el símbolo global no está habilitado (por ejemplo, enlaces seriales, túnel de punto final (end-points), etc), o cuando el símbolo global es indeseable, por ejemplo con símbolos temporales para privacidad.

Los identificadores de interfaz EUI-64 son asignados por las interfaces de red de la capa de enlace, usualmente por fabricantes quienes se les asigna OUI`s por la IEEE. Cada OUI es un valor de 24 bits que se liga a una y sólo una entidad (usualmente a un fabricante de equipos de red). Fabricantes de interfaces Ethernet (y de otras capas de enlace de red) graban dentro de cada interfaz un identificador único global usualmente conocido como dirección de Control de Acceso al Medio (Media Access Control “MAC”). Los 24 bits de mayor orden son el OUI, y el resto de la dirección (los otros 24 bits para Ethernet) son asignados por la entidad OUI. El resultado es que todas las tarjetas Ethernet tienen una dirección única global MAC y tiene una longitud de 48 bits (MAC-48).

La IEEE define a los identificadores de interfaz EUI-64 como un valor OUI de 24 bits de mayor orden seguidos por 40 bits de menor orden tal que la secuencia entera identifica un caso de la implementación (por ejemplo, una interfaz de tarjeta de red) global única. Como sea, hay millones de tarjetas Ethernet en uso que sólo tienen direcciones MAC-48. El RFC3513 especifica métodos para soportar direcciones de interfaz IPv6 con identificadores de interfaz EUI-64 modificado, para interfaces con los siguientes tipos de identificadores de interfaz.

2.3.5.1.1.1.Identificadores de interfaz EUI-64

Si un identificador EUI-64 existe, puede ser usado como un identificador de interfaz IPv6 simplemente invirtiendo el bit “u” (universal/local) del valor OUI (el bit identificado aquí como “X”). Los bits “c” representan el OUI, el bit “g” representa individual/grupo y los bits “m” representan la parte única del vendedor de la interfaz ID. La figura 22 muestra las características del identificador de interfaz EUI-64.

[6] R. Hinden, S. Deering (2003). RFC3513 “Internet Protocol Version 6 (IPv6) Addressing Architecture”. Standards Track. Pp. 7.

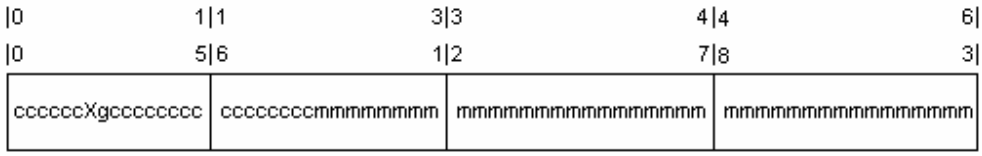


Figura 14. Identificador de Interfaz EUI-64.

2.3.5.1.1.2.Identificador de interfaz MAC-48

El identificador MAC-48 puede ser transformado de su formato estándar, que se muestra en la figura 23.

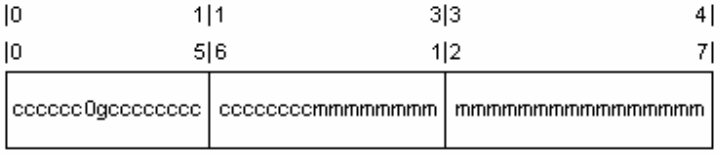


Figura 15. Formato estándar del identificador de interfaz MAC-48

A el formato que se muestra en la figura 24, convirtiéndose en un identificador EUI-64.

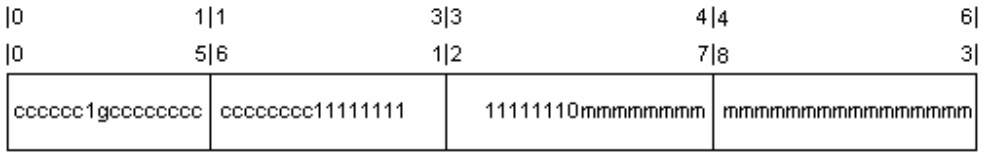


Figura 16. Formato del identificador de interfaz EUI-64 modificado.

El formato del identificador de interfaz modificado EUI-64 es formado invirtiendo el bit "u" (bit universal/local en terminología EUI-64). En el resultado del formato modificado EUI-64 el bit "u" puesto en uno "1" indica ámbito global, y puesto en cero "0" indica ámbito local. El bit "u" representa el bit universal/local, g es el bit individual/grupo, "c" representa el OUI, y los bits "m" representan la parte única del vendedor de la interfaz ID.

2.3.5.1.1.3.Identificadores de interfaz Nonglobal

Algunos protocolos de capa de enlace, tales como ARCnet y Apple`s LocalTalk, no usan direcciones únicas globales pero permiten a cada enlace asignar direcciones a quien quiera. Un formato EUI-64 puede ser generado tomando la dirección de enlace (única sólo en ese enlace) y prefijado con ceros de modo que sea 64 bits de largo. Por ejemplo, un identificador de un nodo de 8 bit, cuyo valor hexadecimal es 0x4F viene a ser el mostrado en la figura 25.

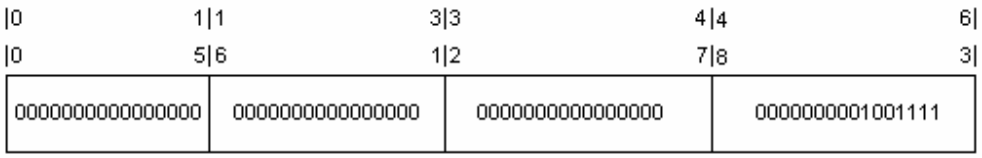


Figura 17. Ejemplo del identificador de interfaz Nonglobal

Hay que notar que el bit universal/local (el séptimo bit de izquierda a derecha) siempre será puesto en cero en estos tipos de identificadores, resultando claro que para otros nodos IPv6 que la dirección del interfaz de este nodo es única sólo en su propio enlace.

2.3.5.1.1.4. Identificador de seguridad

El RFC3041 "Privacy Extensions for Stateless Address Auntoconfiguration in IPv6", trata las preocupaciones de seguridad y privacidad del uso permanente del mismo identificador de interfaz, tal como es hecho con direcciones IPv6 basadas en direcciones IEEE MAC-48, pueden ser mapeadas en detalle por un usuario de Internet activo. Técnicas de recolección de información pueden producir precisos perfiles de usuarios, de su nombre, dirección, tarjetas de crédito y otra información personal; éstas deducciones se hacen mediante los recursos de Internet que el usuario habitualmente visita. En el RFC3041, un mecanismo es definido para generar direcciones aleatorias, que se ajustan al formato modificado EUI-64. Hosts que usan este mecanismo pueden cambiar su identificador de interfaz cada vez que ellos se conecten a la red o a intervalos más frecuentes.

2.3.5.1.1.5. No Identificadores de interfaz

Enlaces punto a punto y cualquier otra configuración de enlaces de túnel, no necesitan identificadores de interfaz. El único requerimiento de los identificadores de interfaz en estos casos es que ellos deben ser únicos en el enlace. El RFC3513 sugiere que los identificadores pueden ser configurados manualmente, creados con un número aleatorio generador, basándose en algunos otros sistemas de identificador (números seriales) o algunos otros métodos. Los autores del RFC3513 recomiendan fuertemente para evitar la duplicación del identificador de interfaz sobre un enlace, un algoritmo de detección de colisión, éste debe ser implementado para que no se use otro identificador de interfaz ya utilizado, ésto se realiza mediante descubrimiento de vecinos "neighbor discovery".

La tabla 7 muestra algunos RFC´s que especifican cómo generar identificadores de interfaz para cualquier protocolo de la capa de enlace.

Tabla 7. Especificaciones para transmitir paquetes IPv6 sobre varios protocolos de capa de enlace

RFC #	Título
2590	Transmission of IPv6 Packets over Frame Relay Networks Specification
2497	Transmission of IPv6 Packets over ARCnet Networks
2492	IPv6 over ATM Networks
2491	IPv6 over Non-Broadcast Multiple Access (NBMA) Networks
2472	IP versión 6 over PPP
2470	Transmission of IPv6 Packets over Token Ring Networks

2467	Transmission of IPv6 Packets over FDDI Networks
2464	Transmission of IPv6 Packets over Ethernet Networks

2.3.5.1.2. Dirección no especificada "Unspecified"

La dirección 0:0:0:0:0:0:0:0 es llamada la dirección no especificada. No debe ser asignada a cualquier nodo. Indica la ausencia de una dirección. Un ejemplo del uso de esta dirección está en el campo de dirección origen de cualquier paquete IPv6 enviado por un host inicializado antes de que tenga su propia dirección.

La dirección no especificada no debe ser usada como la dirección destino de paquetes IPv6 o en encabezados de enrutamiento IPv6. Un paquete IPv6 con dirección de origen no especificada no debe ser enviado por un enrutador IPv6.

2.3.5.1.3. Dirección Loopback

La dirección unicast 0:0:0:0:0:0:0:1 es llamada la dirección Loopback. Puede ser usada por un nodo IPv6 que envía un paquete a si mismo. Puede nunca ser asignada a cualquier interfaz física. Es tratada como tener ámbito de enlace-local, y puede ser pensado como la dirección unicast de enlace-local de una interfaz virtual (típicamente llamada "interfaz loopback") o como un enlace imaginario que va a ninguna parte.

Las direcciones Loopback no deben ser usadas como la dirección origen de un paquete IPv6 que se envía fuera de un solo nodo. Un paquete IPv6 con una dirección de destino Loopback nunca debe ser enviada fuera de un nodo y nunca debe ser enviada por un enrutador IPv6. Un paquete recibido en una interfaz con una dirección Loopback de destino debe ser desechado.

2.3.5.1.4. Dirección Global Unicast

El formato general de una dirección global unicast IPv6 se muestra en la figura 26.

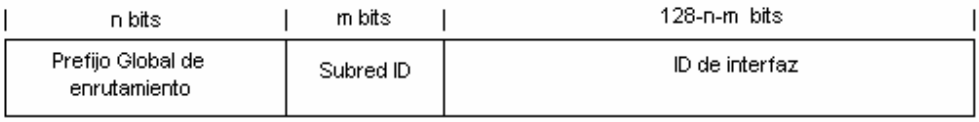


Figura 18. Formato general de una dirección global unicast IPv6

Donde el prefijo global de enrutamiento es un (típicamente estructura jerárquica) valor asignado para un sitio (una ramificación de subred/enlace), la subred ID es un identificador de un enlace dentro de un sitio, y el ID de interfaz es definido según la sección 2.4.6.1.

Esta estructura tiene una semejanza superficial con el uso de subredes en las direcciones IPv4, en la cual una sola red IPv4 es subdividida en subredes. Como sea, la arquitectura de direcciones IPv6 fue prevista para ser una arquitectura agregable.

Cuando IPv4 fue creado, Internet era pequeño, y el modelo para asignación de bloques de direcciones fue basado en un coordinador central (IANA). Quien deseara bloques de

direcciones iría directo con la autoridad central. Como el Internet creció, este modelo llegó a ser impractico. Hoy el esquema de direcciones de clases IPv4 permite longitudes variables de networks ID´s y asignación jerárquica de bloques de direcciones. Grandes ISP´s obtienen largos bloques de la autoridad central y la subdividen para hacer asignaciones a sus clientes; esto es manejado por los proveedores de Internet de hoy en día, pero no hay nada en el espacio de direcciones que ayude a manejar el proceso de asignación. En lugar de esto, cada organización tiene la capacidad de subdividir su dirección asignada para satisfacer sus requisitos internos.

Los diseñadores de IPv6 con esta experiencia tuvieron la ventaja de diseñar una estructura de dirección unicast que reflejara la topología total de Internet. Esto incluye:

- Fácil asignación de bloques de direcciones a varios niveles de la topología jerárquica de Internet.
- Direcciones de red IP que automáticamente reflejen la jerarquía por la cual los enrutadores mueven información a través de Internet, permitiendo que las rutas sean agregadas fácilmente para un enrutamiento más eficiente.
- Flexibilidad para las organizaciones como ISP´s para subdividir sus bloques de direcciones para los clientes.
- Flexibilidad para organizaciones de usuarios final para subdividir su bloque de direcciones para que coincida con redes internas, como se hizo con el subneteo en IPv4.
- Mayor significado a las direcciones IP. En vez de ser sólo una cadena de 128 bits sin estructura, con sólo mirar una dirección que sea posible saber ciertas cosas de él.

La forma más genérica de dividir el espacio de dirección de 128 bits de una dirección unicast es en tres secciones, como se muestra en la figura 26. El prefijo global de enrutamiento y el identificador de subred representan los dos niveles básicos que las direcciones necesitan para ser construidas jerárquicamente. El prefijo de enrutamiento consiste de un número de bits que puede ser subdividida en el futuro de acuerdo a las necesidades del Registro de Internet (Internet Registry) y del proveedor de servicios de Internet, para reflejar la topografía de Internet en su totalidad. La subred ID da un número de bits para que el administrador del sitio pueda crear su propia estructura interna de red.

En teoría, cualquier tamaño para “n” y “m” puede ser usado. La puesta en práctica elegida para IPv6, asigna 48 bits para el prefijo de enrutamiento, de la cual se toman los primeros 3 (001) para identificar qué es una dirección unicast y 16 para el identificador de subred. Esto significa que quedan 64 bits para el identificador de interfaz, que es construido mediante el formato modificado del IEEE EUI-64 explicado en la sección 2.4.6.1.

Debido a esta estructura, la mayoría de los sitios finales (compañías regulares y organizaciones, en comparación de los proveedores de servicios de Internet) serán asignados redes IPv6 con prefijos de 48 bits.

Los 16 bits de la subred ID permite considerable flexibilidad en crear subredes que refleje la estructura de la red del sitio. Por ejemplo:

- Una pequeña organización puede poner todos los bits de la subred ID en cero y tener una estructura interna plana.
- Una organización de tamaño mediana podría utilizar todos los bits de subred ID para realizar lo equivalente al subneteo en IPv4, asignando una subred ID diferente para cada subred, hay 16 bits, lo que nos permite tener 65, 536 subredes.
- Una organización muy grande podría usar los bits para crear múltiples niveles jerárquicos de subredes, exactamente como VLSM (Variable Length Subnet Masking) en IPv4. Por ejemplo la compañía puede usar dos bits para crear cuatro subredes. Podría tomar los siguientes tres bits para crear ocho sub-subredes en algunas o en todo de las cuatro subredes. Restan aún 11 bits para seguir creando sub-sub-subredes y así sucesivamente.

2.3.5.1.4.1.División original del prefijo global de enrutamiento: "Aggregators"

Dado que uno de los problemas que IPv6 resuelve es la mejor organización jerárquica del enrutamiento en las redes públicas (globales), es indispensable el concepto de direccionamiento "agregable" (aggregators).

En la actualidad ya se emplea este tipo de direcciones, basadas en la agregación por parte de los proveedores del troncal Internet, y los mecanismos adoptados para IPv6, permiten su continuidad. Pero además, se incorpora un mecanismo de agregación basado en "intercambios". La combinación permite eficiente agregación de enrutamiento para sitios que se conectan directamente a proveedores y para sitios que se conectan a intercambiadores. Los sitios tendrán que escoger cualquier tipo de entidad de agregación.

Direcciones agregables se organiza entres niveles jerárquicos:

- Topología Pública. – conjunto de proveedores e "intercambiadores" que proporcionan servicios públicos de tránsito de Internet.
- Topología de Sitio.- redes de organizaciones que no proporcionan servicios públicos de tránsito a nodos fuera de su propio sitio.
- Identificador de Interfaz. – identifican interfaces de enlaces.

Como se muestra en la figura 27, el formato de direcciones agregables se diseño para soportar proveedores de larga distancia (identificados como P1, P2, P3, y P4), intercambiadores (identificados como X1 y X2), proveedores de niveles inferiores (podrían ser ISP´s, identificados como P5 y P6), y clientes o subscritores (identificados como S.X).

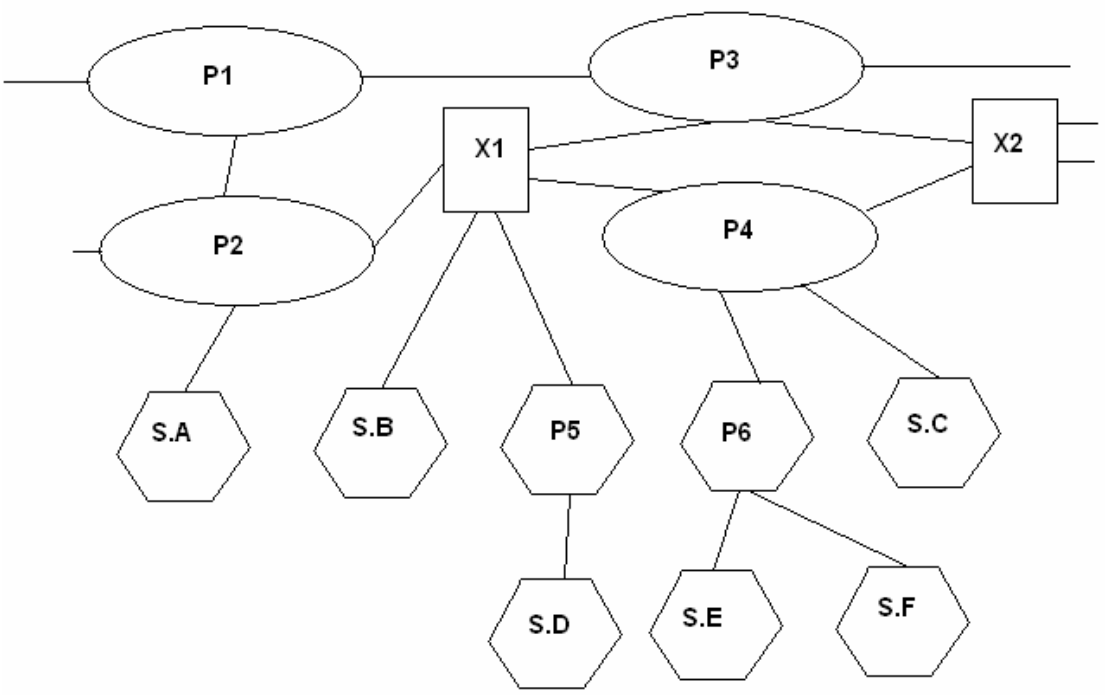


Figura 19. Topología de direcciones agregables.

A diferencia de lo que ocurre actualmente, los intercambiadores también proporcionan direcciones públicas IPv6. Las organizaciones conectadas a dichos intercambiadores también recibirán servicios de conectividad directos, indirectamente a través del intercambiador, de uno o varios proveedores de larga distancia. De esta forma, su direccionamiento es independiente de los proveedores de tráfico de larga distancia, y pueden, por tanto, cambiar de proveedor sin necesidad de renumerar su organización. Este es uno de los objetivos de IPv6. Además una organización puede estar suscrita a múltiples proveedores (multi-homing), a través de un intercambiador, sin necesidad de tener prefijos de direcciones de cada uno de los proveedores.

El formato de las direcciones unicast globales agregables se muestra en la figura 28.

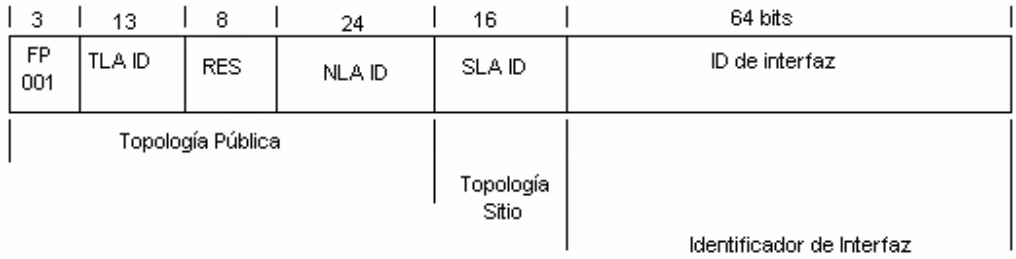


Figura 20. Formato de direcciones unicast globales agregables

Donde:

- **FP** .- Prefijo de Formato (001) “Format Prefix”. Tabla 6.
- **TLA ID**.- Identificador de Agregación de Nivel Superior “Top-Level Aggregation Identifier”

- **Res**.- Reservado para uso futuro.
- **NLA ID**.- Identificador de Agregación de Siguiente Nivel "Next-Level Aggregation Identifier"
- **SLA ID**.- Identificador de Agregación de Nivel de Sitio "Site-Level Aggregation Identifier"
- **ID de interfaz**.- identificador de interfaz

Como se puede observar el prefijo global de enrutamiento es dividido en forma jerárquica. Existen 45 bits disponibles (48 bits menos los primeros tres que están fijados en "001"). Cuando la estructura de las direcciones unicast fue primeramente detallado en el RFC2374, el documento describía una división especifica de los 45 bits basados en dos niveles de topología jerárquica de Internet Registry y proveedores.

TLA ID .-El identificador de agregación de nivel superior (TLA) contiene el nivel más alto de información de enrutamiento de la direcciones. Los enrutadores situados en este nivel, en la tabla de encaminado, deben de tener una entrada para cada TLA ID activo, y probablemente entradas adicionales relativas al propio TLA ID donde están físicamente situados. Esta estructura de direccionamiento permite 8.192 (2^{13}) identificadores de TLA. Se prevé su crecimiento haciendo que este campo crezca hacia la derecha en el espacio reservado para el futuro, o usando este mismo formato/estructura para prefijos de formato (FP) adicionales. En especificaciones anteriores, el TLA fue de identificadores basada por proveedores. Fueron asignadas a ARIN (American Registry for Internet Numbers) en Norte América, RIPE (Réseau IP Européens) centro de coordinación de red en Europa, y APNIC (Asia Pacific Network Information Center).

Res.- El campo reservado es para uso futuro y debe ser puesto en cero. Este campo permite un crecimiento futuro de TLA y NLA como sea apropiado.

NLA ID.- Son usadas por organizaciones (proveedores e intercambiadores) asignadas a un TLA ID para crear un direccionamiento jerárquico para identificar sitios u organizaciones que de ella dependen. Pueden reservar los bits superiores para la diferenciación de la estructura de su red, en función de sus propias necesidades. Puede utilizar el resto de los bits del campo para identificar sitios que desea servir, esto se muestra en la figura 29. Dado que cada organización que recibe un TLA dispone de 24 bits de espacio NLA, permite proporcionar servicio aproximadamente al número total de direcciones IPv4 soportadas actualmente.

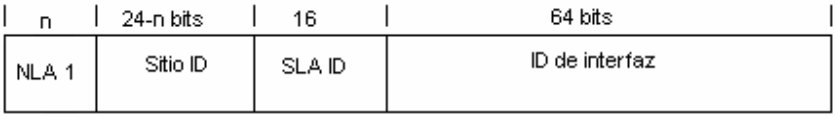


Figura 21. Formato de la estructura NLA

Las organizaciones que reciben un TLA pueden soportar varios NLA en su propio espacio de direccionamiento (Sitio ID). Esto permite que sirvan tanto a clientes directos (suscriptores) como a otras organizaciones proveedoras de servicios públicos de transito. Y así sucesivamente, como se muestra en la figura 30.

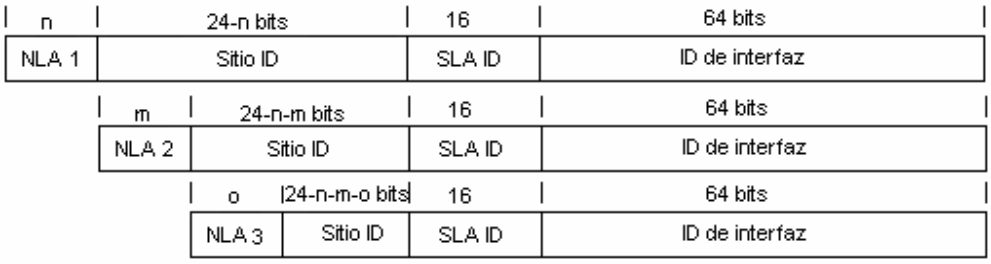


Figura 22. Formato con varios NLA´s.

El diseño del espacio NLA de cada organización es libre para cada TLA asignado, y así sucesivamente con los niveles inferiores. Sin embargo, se recomienda seguir el procedimiento que se describe en el RFC2050 "Internet Registry IP Allocation Guidelines".

En cualquier caso es fundamental apreciar el balance entre eficacia de encaminado agregable y flexibilidad. Las estructuras más jerárquicas permiten una mejor agregación, y por tanto reducen las tablas de encaminado. Por el contrario, asignaciones más planas del espacio NLA proporcionan mejor flexibilidad en la conexión (crecimientos no previstos en un determinado espacio), resultando en tablas de encaminamiento mayores, y por tanto menos eficaces.

SLA ID.- El identificador de agregación de nivel de sitio es usado por organizaciones finales para crear su propia estructura jerárquica de direcciones e identificar sus subredes. Es equivalente al concepto de subred en IPv4, con la diferencia de que cada corporación tiene un mayor número de subredes (16 bits proporcionan capacidad para 65 535).

Del mismo modo que en el caso del NLA, se puede escoger entre estructura plana, o crear varios niveles, según lo muestra la figura 31.

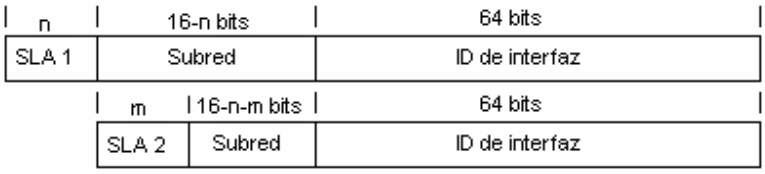


Figura 23. Formato SLA

Una gran compañía podría necesitar varios identificadores SLA. Como es lógico, cada caso dependerá de cómo esten conectadas sus diversas delegaciones.

ID de Interfaz.- el identificador de interfaz es usado para identificar interfaces en un enlace. Requiere ser única en el enlace. Pueden ser también únicas en un ámbito. En muchos casos un identificador de interfaz es igual o basado en la dirección de capa de enlace. Identificadores de interfaz usada en el formato de direcciones agregables globales unicast se requiere que sean de 64 bits de largo y ser construidas con el formato EUI-64. Para mayor información acerca de los identificadores de interfaz dirigirse a la sección 2.4.6.1.

2.3.5.1.5. Direcciones IPv6 con direcciones IPv4 acopladas "Embedded"

Porque la transición para IPv6 será gradual, se han definidos dos mecanismos de transición que incluyen una técnica para hosts y enrutadores para túneles dinámicos de paquetes IPv6 sobre infraestructura de enrutamiento IPv4. Ambas se describen en el RFC3513 como sigue.

Direcciones IPv6 "IPv4-Compatible"

Este tipo de direcciones son usadas para túneles de paquetes IPv6 dinámicos sobre una infraestructura de enrutamiento IPv4. El formato de una dirección IPv6 "IPv4-compatible" es 0:0:0:0:0:0:A.B.C.D o ::A.B.C.D. Nodos IPv6 que usan esta técnica se asignan una dirección especial IPv6 unicast que transporta una dirección IPv4 en los 32 bits de menor orden y en los 96 bits de mayor orden se ponen en cero. Los 128 bits enteros de la Dirección IPv6 "IPv4-compatible" son usados como la dirección IPv6 de un nodo; mientras la dirección IPv4 acoplada en los 32 bits de menor orden, son usadas como la dirección IPv4 del nodo. Direcciones IPv6 "IPv4-compatible" son asignadas a nodos que soportan ambas pilas de protocolo IPv4 e IPv6 y se usan en túneles automáticos. La figura 32 muestra la estructura de este tipo de direcciones.

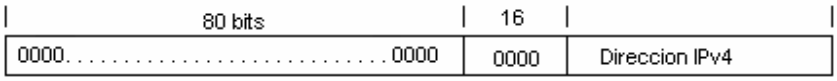


Figura 24. Dirección IPv6 "IPv4-Compatible"

Nota: las direcciones IPv4 usadas en la direcciones IPv6 "IPv4-compatible" deben ser únicas direcciones globales unicast IPv4[7].

Direcciones IPv6 "IPv4-Mapeada".

Este tipo de direcciones son utilizadas para representar las direcciones de nodos solamente-IPv4 (IPv4-only). Estas direcciones pueden ser usadas por un nodo IPv6 para enviar un paquete a un nodo solamente-IPv4. La dirección también transporta la dirección IPv4 en los 32 bits de menor orden.

La figura 33 muestra la estructura de este tipo de direcciones.

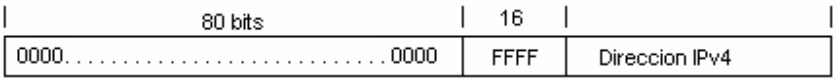


Figura 25. Dirección IPv6 "IPv4-mapeada"

[7]R. Hinden, S. Deering (2003). RFC3513 "Internet Protocol Version 6 (IPv6) Addressing Architecture". Standards Track. Pp. 9.

2.3.5.1.6. Direcciones IPv6 Unicast para uso local.

Con IPv4, las organizaciones solían usar direcciones IP del rango privado, como se define en el RFC1918. Las direcciones reservadas para uso privado nunca deben ser enviadas hacia los enrutadores de Internet, deben ser limitadas a la red de la organización. Para salir a Internet se necesitaba de técnicas como NAT.

IPv6 permite dos espacios de direcciones separados para enlace-local "link-local" y sitio-local "site-local" (desprobada por la IETF), llamada ahora direccion unica-local.

Direcciones de enlace-local

Las direcciones de enlace-local son para uso en un solo enlace y nunca debe ser enrutadas. Estas direcciones tienen un ámbito más pequeño que las direcciones de sitio-local; se refieren sólo a un particular enlace físico. Pueden ser usadas para mecanismos de autoconfiguración, para descubrimiento de vecinos (neighbor discovery), y en redes sin enrutadores. Los enrutadores nunca deben enviar paquetes con direcciones origen o destino de enlace-local a otros enlaces. La figura 34 muestra el formato de este tipo de dirección. Se identifican porque empiezan con el valor hexadecimal FE80.

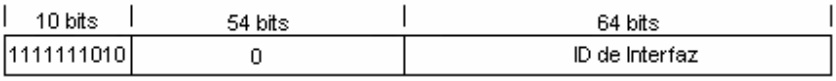


Figura 26. Formato de dirección de enlace-local

Direcciones de sitio-local (Desaprobada por la IETF)

Estas direcciones tienen el ámbito de un sitio entero u organización. Pueden ser enrutadas dentro de un sitio, los enrutadores no pueden enviar paquetes fuera del sitio. Estas direcciones permiten el direccionamiento dentro de una organización sin la necesidad de usar un prefijo público. En la figura 35 se muestran los formatos de las direcciones de sitio local, según el RFC2373 y el RFC3513. Cabe señalas que la diferencia existente entre estos dos formatos es el tamaño del campo del ID de subred, en el RFC2373 se especifica de 16 bits, y en el RFC3513 se especifica de 54 bits. Los 54 bits pueden ser utilizados para el diseño de las subredes, pero se recomienda que se haga con el formato de 16 bits, para que cuando la organización adquiera un prefijo global, se pueda conectar sin la necesidad de hacer modificaciones del ID de subredes. Se identifican porque empiezan con el valor hexadecimal FEC0.

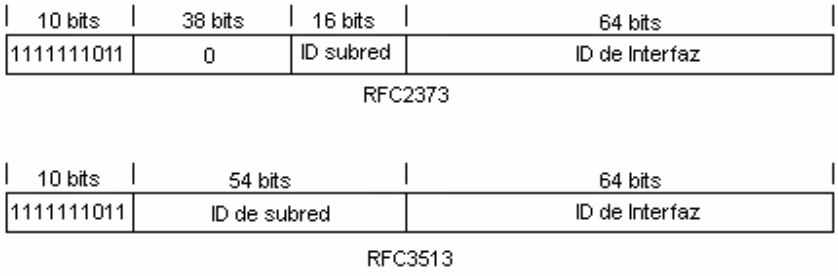


Figura 27. Formato de dirección de sitio-local

Direcciones Única Local

Este tipo de direcciones son propuestas en el RFC4193, haciendo obsoletas a las direcciones de Sitio-Local. El formato de las direcciones IPv6 Única Local tiene el objetivo de crear una dirección que es globalmente única para comunicaciones locales, usualmente dentro de un sitio, y no son enrutables al Internet Global. La ventaja de esta dirección, es la capacidad que presentan para ser enrutables entre un sistema de sitios que utilizan el mismo formato de dirección internamente, sin crear conflictos de direccionamiento o requerir renumeración de las interfaces, ya que utiliza un algoritmo pseudos-aleatorio para formar al prefijo de red, garantizando unicidad entre las redes.

La figura 36 muestra el formato de la dirección Única Local, según el RFC 4193; donde L es la política de asignación, si es puesto a 1 el prefijo es asignado localmente y 0 aun no ha sido definido, el ID Global es el identificador pseudos-aleatorio generado para garantizar la unicidad de la dirección entre sitios, y es formada por un algoritmo descrito en el RFC4193, el ID de subred y el ID de Interfaz tienen el mismo uso de la dirección de sitio local.

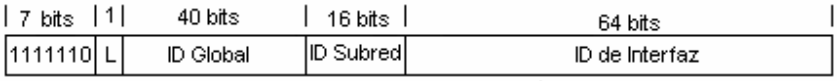


Figura 28. Formato de dirección Única Local

2.3.5.2.Direcciones Anycast

Una de las características más sorprendentes de IPv6 fue la introducción de un nuevo modelo de comunicación, anycast, que une a los modelos existentes de unicast y multicast. Donde la comunicación unicast permite la transmisión de paquetes a un nodo especifico, y multicast permite la transmisión del mismo paquete a uno o más nodos; anycast agrega la capacidad de enviar paquetes a cualquier nodo (solamente uno) de un grupo.

Una especificación experimental para anycasting IPv4 fue definido en el RFC1546, “Host Anycast Service” en 1993. Este documento sugiere introducir otra clase de direcciones fuera del espacio de direcciones IPv4 para direcciones anycast. De está forma, los nodos pueden identificar paquetes anycast simplemente con mirar sus direcciones de destino y tratarlos apropiadamente.

La motivación del experimental RFC1546 fue para proveer nodos con una forma simple de alcanzar cualquier grupo de servidores de aplicaciones intercambiables. Por ejemplo, algo tan requerido por usuarios, el escoger un particular servidor FTP de una lista, con anycast podría permitir a los usuarios simplemente especificar el grupo de servidores FTP, y cualquiera podría ser satisfactorio.

Una dirección anycast IPv6 es una dirección asignada a más de una interfaz (típicamente pertenecen a nodos diferentes), con la propiedad que un paquete enviado a una dirección anycast es encaminado a la interfaz “más cerca” que tiene esa dirección, de acuerdo a las métricas de los protocolos de enrutamiento.

Las direcciones anycast utilizan el espacio de direcciones de las direcciones unicast, usando cualquier formato de las direcciones unicast definido. Así, las direcciones anycast son sintácticamente indistinguibles de la direcciones unicast. Cuando una dirección unicast es asignada a más de una interfaz, ésta se convierte en una dirección anycast, a los nodos que se les asigna estas direcciones deben ser explícitamente configurados para saber que éstas son direcciones anycast.

Todas las interfaces configuradas con direcciones anycast compartirán algún prefijo de red en común. Por ejemplo, considere el caso de un grupo de nodos, cada uno configurado con las misma dirección anycast, dentro de una típica red IPv6 /48. Si estos nodos tienen interfaces configuradas con las direcciones anycast localizadas en todas las subredes de la red, el prefijo compartido será la red completa /48 de las direcciones de la red. Este prefijo (/48) identifica el área de red, dentro de la cual la dirección anycast debe ser anunciada, con un anuncio separado de encaminamiento por cada interfaz de red a través de la red entera /48[8].

En el supuesto de que el prefijo compartido sea nulo (significa que no hay un prefijo de enrutamiento identificable en común) la dirección anycast podría ser anunciada a través de todos los enrutadores en el Internet global IPv6. En general, direcciones globales anycast podrían ser una tensión severa en la infraestructura de enrutamiento de Internet, debido a la dificultad de escalamiento. Por lo tanto, se espera esté disponible solamente sobre una base muy estricta, o inaccesible.

Un uso esperado de las direcciones anycast es para identificar el sistema de enrutadores que pertenecen a una organización que provee servicios de Internet. Tales direcciones podrían utilizarse como direcciones intermediarias en un encabezado de enrutamiento IPv6, para causar que un paquete sea entregado por medio de un particular proveedor de servicios o secuencia de proveedores de servicios.

Otro posible uso, es identificar el sistema de enrutadores unidos a una particular subred o al sistema de enrutadores que provee la entrada dentro de un particular dominio de enrutamiento.

Hay poca experiencia en el uso de direcciones anycast de Internet, y se conoce algunas complicaciones y peligros en general. Las siguientes restricciones son impuestas en direcciones anycast IPv6:

- Una dirección anycast no debe ser usada como la dirección origen de un paquete IPv6.
- Una dirección anycast no debe ser asignada a un host IPv6, esto es, deben ser asignadas a enrutadores IPv6 solamente.

[8] PETE LOSHIN (2004). IPv6: Theory, Protocol, and practice. San Francisco CA. Morgan Kaufmann Publisher e impreso por Elsevier. Segunda Edición. Pp. 200.

2.3.5.2.1. Dirección requerida Anycast.

Existe una dirección anycast, requerida para cada subred, se denomina "dirección anycast del enrutador de la subred" (subset-router anycast address). La figura 37 muestra el formato de este tipo de dirección. El "prefijo de subred" en una dirección anycast es el prefijo que identifica un enlace especifico. Esta dirección anycast es sintácticamente igual a una dirección unicast para una interfaz de un enlace, con el identificador de interfaz puesto en ceros.

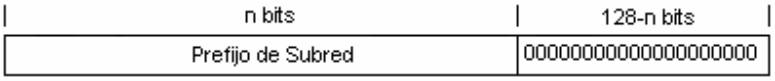


Figura 29. Formato de dirección anycast del enrutador de la subred.

Todos los enrutadores han de soportar esta dirección para las subredes a las que están conectadas. Los paquetes enviados a la "dirección anycast del enrutador de la subred", serán enviados a un enrutador de la subred.

Una aplicación evidente de esta característica, además de la tolerancia a fallos, es la movilidad. Imaginemos nodos que necesitan comunicarse con un enrutador entre el conjunto de los disponibles en su subred.

2.3.5.2.2. Formato de direcciones reservadas de subred anycast.

El RFC2526 define un sistema adicional de direcciones anycast reservadas dentro cada prefijo de subred, y lista la asignación inicial de estas direcciones de subred anycast reservadas.

Dentro de cada subred, los 128 valores superiores de los identificadores de interfaz están reservados para su asignación como direcciones anycast de la subred.

La construcción de una dirección reservada de subred anycast, depende del tipo de dirección IPv6 usada dentro de la subred, según lo indicado por el formato del prefijo en la dirección.

En particular, para los tipos de direcciones IPv6 que requieren tener 64 bits en el identificador de interfaz en formato EUI-64, el bit universal/local debe estar puesto en cero (local), para indicar que el identificador de interfaz en la dirección no es globalmente único. Direcciones IPv6 de este tipo son actualmente especificadas en tener el formato de prefijo entre 001 y 111, excepto para direcciones multicast (1111 1111). En este caso, las direcciones reservadas anycast de subred se construyen de la forma mostrada en la figura 38.

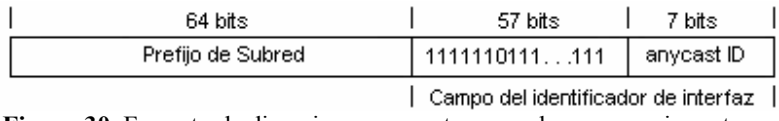


Figura 30. Formato de direcciones anycast reservadas que requieren tener 64 bits en el identificador de interfaz con formato EUI-64.

En el resto de los casos, el identificador de interfaz puede tener una longitud diferente de 64 bits, por lo que la construcción se realiza según como lo muestra la figura 39.

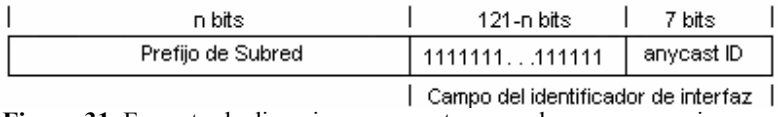


Figura 31. Formato de direcciones anycast reservadas que no requieren formato EUI-64.

El identificador anycast (anycast ID) identifica una dirección particular anycast reservada dentro del prefijo de subred, del sistema de direcciones reservadas de subred anycast.

2.3.5.2.2.1.Lista de direcciones reservadas de subred anycast.

Actualmente los siguientes identificadores anycast para estas direcciones reservadas de subred anycast, que se han definido, se muestran en la tabla 8. Identificadores anycast adicionales se espera sean definidas en el futuro.

Tabla 8. Identificadores Anycast reservados.

Decimal	Hexadecimal	Descripción
127	7F	Reservado
126	7E	Mobile IPv6 Home-Agents anycast
0-125	00-7D	Reservado

2.3.5.3.Direcciones Multicast

Una dirección IPv6 multicast es un identificador para un grupo de interfaces (típicamente en nodos diferentes), identificado por el byte de mayor orden FF, o en notación binaria 1111 1111 (Tabla 6). Una interfaz puede pertenecer a cualquier número de grupos multicast. Multicast ya existía en IPv4, pero se ha redefinido y mejorado para IPv6. La figura 40 muestra el formato general de las direcciones Multicast.

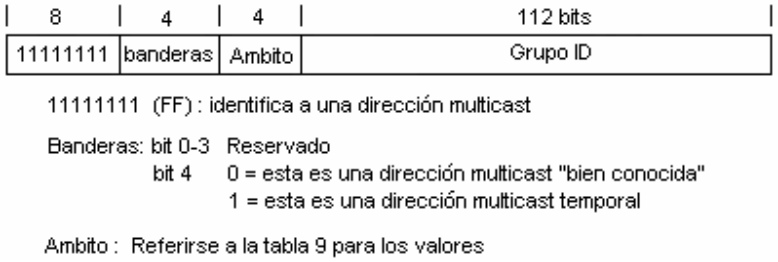


Figura 32. Formato de direcciones multicast.

Los 11111111 bits al inicio identifican que es una dirección multicast. Los siguientes 4 bits es un campo de banderas que se define a continuación: los primeros 3 bits del campo de banderas son reservados para uso futuro y deben estar puesto a cero; el último bit del campo bandera indica si la dirección es permanentemente asignada (puesto a cero “bien conocidas”, asignadas por el IANA) o es una dirección temporal (puesto a uno). El campo

de ámbito es utilizado para limitar el ámbito de un grupo multicast, la tabla 9 muestra los valores según el RFC3513.

Tabla 9. Valores del campo Ámbito.

Valor	Descripción
0	Reservado
1	Ámbito Interfaz-local (interface-local scope)
2	Ámbito Enlace-local (link-local scope)
3	Reservado
4	Ámbito administración-local (admin.-local scope)
5	Ámbito Sitio-local (site-local scope)
6, 7	No asignadas
8	Ámbito organización-local (organization-local scope)
9, A, B, C, D	No asignadas
E	Ámbito global (global scope)
F	Reservado

Ámbito de interfaz-local unicamente se propaga en una sola interfaz de un nodo, y es usado sólo para transmisiones loopback de multicast.

Ámbito multicast de enlace-local y sitio-local se propaga en las mismas regiones topológicas como los correspondientes ámbitos unicast.

Ámbito admin-local es el ámbito más pequeño que debe ser administrativamente configurado.

Ámbito de organización-local es prevista para propagarse a múltiples sitios pertenecientes a una sola organización.

Ámbitos etiquetados con "no asignados" son disponibles para los administradores, para definir adicionales regiones multicast.

Grupo ID identifica el grupo multicast, permanente o transitorio, dentro del ámbito dado.

El significado de una dirección multicast permanentemente-asignada es independiente del valor del ámbito. Por ejemplo, si el "grupo de servidores NTP" es asignado a una dirección multicast permanente con un grupo ID de 101 (hex), entonces[9]:

FF01:0:0:0:0:0:0:101 significa que todos los servidores NTP en la misma interfaz (el mismo nodo) que el del remitente.

FF02:0:0:0:0:0:0:101 significa que todos los servidores NTP en el mismo enlace que el del remitente.

FF05:0:0:0:0:0:0:101 significa que todos los servidores NTP en el mismo sitio que el del remitente.

[9] R. Hinden, S. Deering (2003). RFC3513 "Internet Protocol Version 6 (IPv6) Addressing Architecture". Standards Track. Pp. 14.

FF0E:0:0:0:0:0:0:101 significa que todos los servidores NTP en Internet.

Direcciones multicast no-permanentemente-asignadas son significativas sólo dentro de un ámbito dado. Por ejemplo, un grupo identificado por la no-permanente, dirección multicast de sitio-local FF15:0:0:0:0:0:0:101 en un sitio no lleva ninguna relación con un grupo usando la misma dirección en un sitio diferente, ni a un grupo no-permanente usando la misma identificación de un grupo con diferente ámbito, ni a un grupo permanente con el mismo grupo ID.

Direcciones multicast no deben ser usadas como direcciones origen en paquetes IPv6, o aparecer en cualquier encabezado de enrutamiento.

Los enrutadores no deben enviar cualquier paquete multicast más allá del ámbito indicado por el campo de ámbito en la dirección de destino multicast.

Los nodos no deben originar un paquete para una dirección multicast en el cual su campo de ámbito contenga el valor reservado de 0; si se recibe tal paquete, debe ser eliminado silenciosamente. Los nodos no deben originar un paquete para una dirección multicast en la cual su campo de ámbito contenga el valor reservado de F; si tal paquete es enviado o recibido, éste debe ser tratado igual que un paquete destinado a una dirección multicast global (ámbito E).

2.3.5.3.1. Direcciones Multicast predefinidas

Los últimos 112 bits de la dirección transporte el grupo ID multicast. El RFC3513 define la inicial asignación de direcciones multicast IPv6 que son permanentemente asignadas. Algunas asignaciones son hechas para ámbitos fijos, y algunas asignaciones son válidas sobre todos los rangos de ámbitos. A continuación la tabla 10 muestra las asignaciones de direcciones multicast permanentes registradas por el IANA. Para la asignación permanente de las direcciones multicast de ámbito variable, puede ser consultado en la liga http://www.iana.org/assignments/ipv6-multicast-addresses.

Tabla 10. Asignación de direcciones multicast permanentes de ámbito fijo.

Dirección	Descripción
Ámbito de interfaz-local o nodo-local	
FF01:0:0:0:0:0:0:1	All Nodes Address
FF01:0:0:0:0:0:0:2	All Routers Address
FF01:0:0:0:0:0:0:FB	mDNSv6
Ámbito de enlace-local	
FF02:0:0:0:0:0:0:1	All Nodes Address
FF02:0:0:0:0:0:0:2	All Routers Address
FF02:0:0:0:0:0:0:3	Unassigned (No asignada)
FF02:0:0:0:0:0:0:4	DVMRP Routers
FF02:0:0:0:0:0:0:5	OSPFIGP
FF02:0:0:0:0:0:0:6	OSPFIGP Designated Routers
FF02:0:0:0:0:0:0:7	ST Routers
FF02:0:0:0:0:0:0:8	ST Hosts
FF02:0:0:0:0:0:0:9	RIP Routers

FF02:0:0:0:0:0:0:A	EIGRP Routers
FF02:0:0:0:0:0:0:B	Mobile-Agents
FF02:0:0:0:0:0:0:C	SSDP
FF02:0:0:0:0:0:0:D	All PIM Routers
FF02:0:0:0:0:0:0:E	RSVP-ENCAPSULATION
FF02:0:0:0:0:0:0:16	All MLDv2-capable routers
FF02:0:0:0:0:0:0:6A	All-Snoopers
FF02:0:0:0:0:0:0:FB	mDNSv6
FF02:0:0:0:0:0:1:1	Link Name
FF02:0:0:0:0:0:1:2	All-dhcp-agents
FF02:0:0:0:0:0:1:3	Link-local Multicast Name Resolution
FF02:0:0:0:0:0:1:4	DTCP Announcement
FF02:0:0:0:0:1:FFXX:XXXX	Solicited-Node Address
Ámbito de sitio-local	
FF05:0:0:0:0:0:0:2	All Routers Address
FF05:0:0:0:0:0:0:FB	mDNSv6
FF05:0:0:0:0:0:1:3	All-dhcp-servers
FF05:0:0:0:0:0:1:4	Desaprobada (2003-03-12)
FF0X:0:0:0:0:0:1:1000 - FF0X:0:0:0:0:0:1:13FF	Service Location, Version 2

2.3.5.3.2. Dirección multicast de nodo-solicitado (Solicited-Node)

La dirección multicast de nodo-solicitado es una dirección multicast que cada nodo debe acoplar para cada dirección unicast y anycast asignada. Lo anterior se utiliza en el proceso DAD (Duplicate Address Detection).

Una dirección multicast de nodo-solicitado es formada tomando los 24 bits de menor orden (unicast o anycast) y añadiendo esos bits al prefijo FF02:0:0:0:0:1:FF00::/104 resultando una dirección multicast en el rango FF02:0:0:0:0:1:FF00:0000 a FF02:0:0:0:0:1:FFFF:FFFF.

Cualquier nodo procura configurarse una dirección IPv6, esto requiere enviar una solicitud de descubrimiento de vecino en la dirección multicast de nodo-solicitado para esta dirección IPv6. De esta forma, si la dirección ya ha sido usada, el nodo que la contenga responderá la solicitud, y la petición del nodo puede evitar la colisión de direcciones con el nodo existente.

La dirección multicast de nodo-solicitado es diseñada para satisfacer esta función en la forma más eficiente posible.

- Primero, para usar los 24 bits de menor orden de cada dirección IPv6, cada nodo tendrá probablemente que admitir solamente una dirección de multicast de nodo-solicitado por interfaz IPv6. La misma dirección puede servir para cualquier o todas las direcciones anycast/unicast basadas en el formato EUI-64, sin importar su ámbito.

- Segundo, aunque la dirección multicast de nodo-solicitado por interfaz pertenezca a dos nodos diferentes en el mismo enlace, es posible que sea minimizado por el uso del espacio de los 24 bits de la dirección. Con más de 16 millones de direcciones únicas, la probabilidad de que dos interfaces IPv6 colisionen en estos bits en el mismo enlace es mínima.

Las direcciones multicast de nodo-solicitado es también utilizado para otros propósitos de descubrimiento de vecinos, permitiendo a los nodos mapear rápidamente una dirección IPv6 para una dirección de capa de enlace de red.

Por ejemplo, la dirección multicast de nodo-solicitado correspondiente a la dirección IPv6 4037::01:800:200E:8C6C es FF02::1:FF0E:8C6C. Direcciones IPv6 difieren solamente en los bits de mayor orden.

2.3.5.4.Direcciones requeridas por un nodo

El estándar especifica que un host debe asignar las siguientes direcciones como identificadores:

- Se requiere direcciones de enlace-local por cada interfaz.
- Direcciones adicionales unicast o anycast que sean configuradas para las interfaces de los nodos (manual o automáticas)
- Dirección de loopback
- Direcciones multicast de All-Nodes
- Direcciones multicast de nodo-solicitado para cada una de sus direcciones unicast y anycast.
- Direcciones multicast de todos los grupos en donde el nodo pertenece.

Un enrutador debe reconocer todas las direcciones que un host requiere, mencionadas anteriormente, más las siguientes direcciones como identificadores:

- La dirección de subred anycast para todas las interfaces para la cual es configurado para actuar como enrutador.
- Todas las otras direcciones anycast con la cual el enrutador ha sido configurado.
- La dirección multicast de All-Routers.

2.4. ICMPv6

El protocolo de Internet versión 6 usa ICMP (Internet Control Message Protocol) como se define para IPv4 en el RFC792, con un número de cambios significativos. El protocolo resultante es el llamado ICMPv6, y lleva en el encabezado de IPv6 en el campo de siguiente cabecera el valor de 58.

ICMPv6 es usado por nodos IPv6, para reportar errores encontrados en el procesamiento de paquetes, y para realizar otras funciones de la capa de Internet, como un diagnostico (ICMPv6 “ping”). ICMPv6 es una parte integral de IPv6, y base del protocolo, por lo que se deben implementar completamente por cada nodo IPv6.

ICMPv6 es mucho más poderoso que IPv4 y contiene nuevas funcionalidades. Sea el caso de las funciones de IGMP (Internet Group Management Protocol) que maneja grupos de miembros multicast con IPv4 ha sido incorporado dentro de ICMPv6. De la misma forma para las funciones de ARP/RARP (Address Resolution Protocol/Reverse Address Resolution Protocol) que son usadas en IPv4 para mapear las direcciones de capa 2 para las direcciones IP (y viceversa). ND (Neighbor discovery) es introducido; usa mensajes ICMPv6 en orden para determinar direcciones de la capa de enlace de vecinos que sean iguales en el mismo enlace, para encontrar routers, para mantener actualizado qué vecinos son alcanzables, y para detectar cambios en las direcciones de capa de enlace. ICMPv6 también soporta Mobile IPv6.

2.4.1. Formato General de los Mensajes

Todos los mensajes ICMPv6 tienen la misma estructura general del encabezado, como se muestra en la figura 41. Cabe destacar que los primeros tres campos de Tipo, Código (code), y la suma de comprobación de la cabecera (checksum), no han cambiado de ICMPv4. Un encabezado IPv6 y cero o más encabezados de extensión se anteponen de cada mensaje ICMPv6. El encabezado que se antepone del encabezado ICMP tiene un valor de siguiente cabecera de 58, este valor es diferente en IPv4.

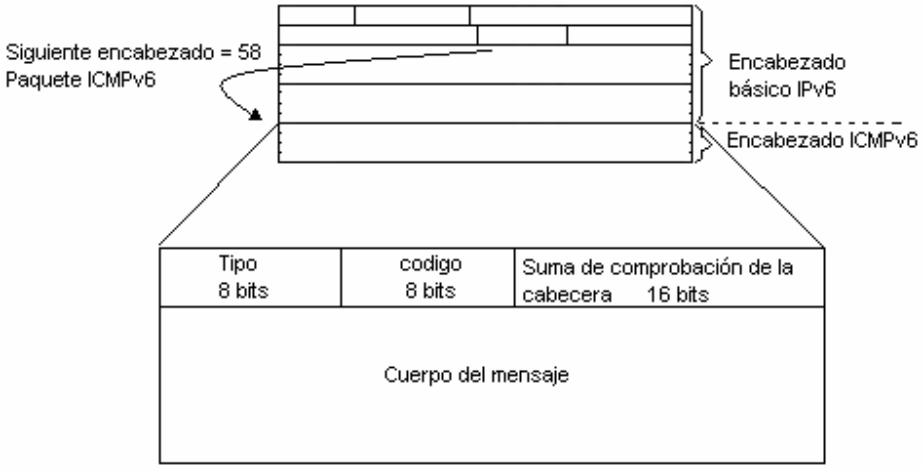


Figura 33. Formato general de los mensajes ICMPv6

- El campo **Tipo** indica el tipo de mensaje. Este valor determina el formato de los datos restantes.
- El campo **Código** depende del tipo de mensaje. Esto es usado para crear un nivel adicional de mensajes.
- El campo de **Suma de comprobación de la cabecera** es usado para detectar datos corruptos en los mensajes ICMPv6 y también en el encabezado de IPv6.
- Mensajes ICMPv6 son agrupados en dos clases:

    - Mensajes de error ICMP: los mensajes de errores son identificados por un cero en el bit de mayor orden en su campo de Tipo del mensaje, así estos mensajes de error tienen tipos de mensajes en el rango de 0 a 127.

- Mensajes informacionales ICMP: los mensajes informacionales son identificados por un uno en el bit de mayor orden en su campo de Tipo del mensaje, así estos mensajes informacional tienen tipos de mensajes en el rango de 128 a 255.

El RFC4443 define los siguientes tipos de mensajes ICMPv6, con su información adicional de código, de la cual depende cada mensaje. A continuación se muestran los mensajes más relevantes de cada tipo; los mensajes de error ICMPv6 se muestran en la tabla 11 y los mensajes informacionales ICMPv6 se muestra en la tabla 12. La lista completa de mensajes se encuentra en el siguiente enlace de la página del IANA: http://www.iana.org/assignments/icmpv6-parameters.

Tabla 11. Mensajes de error ICMPv6 y tipos de códigos

Número de Mensaje	Tipo de Mensaje	Campo de código
1	Destination Unreachable	0 = no route to destination 1 = communication with destination administratively prohibited 2 = beyond scope of source address 3 = address unreachable 4 = port unreachable
2	Packet Too Big	0 = puesto a cero por el emisor e ignorado por el receptor
3	Time Exceeded	0 = hop limit exceeded in transit 1 = fragment reassembly time exceeded
4	Parameter Problem	0 = erroneous header field encountered 1 = unrecognized next header type encountered 2 = unrecognized IPv6 option encountered El campo del puntero identifica el octeto compensado dentro del paquete invocado donde el error fue detectado. El puntero señalara más allá del final del paquete ICMPv6 si el campo del error esta más allá, que pueda caber en el máximo tamaño de un mensaje ICMPv6 de error.
100	Private experimentation	
101	Private experimentation	
127	Reserved for expansion of ICMPv6 error messages.	

Hay que notar que los números de mensajes y tipos están substancialmente cambiados comparados con ICMPv4. ICMP para IPv6 es un protocolo diferente, y las dos versiones de ICMP no son compatibles.

Tabla 12. Mensajes informacionales ICMPv6.

Número de Mensaje	Tipo de mensaje	Descripción
127	Reserved for expansion of ICMPv6 error messages	RFC-ietf-ipngwg-icmp-v3-07.txt
128	Echo Request	RFC4443. Ambos usados para el

129	Echo Reply	comando de ping.
130 131 132	Multicast Listener Query Multicast Listener Report Multicast Listener Done	RFC2710. usado para el manejo de grupos multicast (IPv4 usa IGMP para esta funcionalidad)
133 134 135 136 137 138 139 140	Router Solicitation Router Advertisement Neighbor Solicitation Neighbor Advertisement Redirect Message Router Renumbering ICMP Node Information Query ICMP Node Information Response	RFC2461. usado para descubrimiento de vecinos y autoconfiguración. RFC2461 RFC2461 RFC2461 RFC2894. Usado para renumerar routers. Crawford Crawford
141 142	Inverse ND Solicitation Inverse ND Adv Message	RFC3122 RFC3122
150 151 152 153	ICMP messages utilized by experimental mobility protocols such as Seamoby Multicast Router Advertisement. Multicast Router Solicitation. Multicast Router Termination	RFC4065 RFC-ietf-magma-mrdisc-07.txt RFC-ietf-magma-mrdisc-07.txt RFC-ietf-magma-mrdisc-07.txt
200 201 255	Private experimentation Private experimentation Reserved for expansion of ICMPv6 informational messages.	RFC-ietf-ipngwg-icmp-v3-07.txt RFC-ietf-ipngwg-icmp-v3-07.txt RFC-ietf-ipngwg-icmp-v3-07.txt

2.4.2. Mensajes de error ICMP

Cada mensaje ICMP puede tener un encabezado levemente diferente dependiendo del tipo de error reportado o información transportada.

2.4.2.1.Mensajes de destino Inalcanzable (Unreachable)

Un mensaje de destino inalcanzable debe ser generado por un enrutador, o por la capa IPv6 en el nodo que se origina, en respuesta a un paquete que no puede ser entregado a su dirección de destino por otras razones que no sean congestión. Un mensaje ICMPv6 no debe ser generado si un paquete se ha perdido por congestión. El formato del mensaje de destino inalcanzable se muestra en la figura 42.

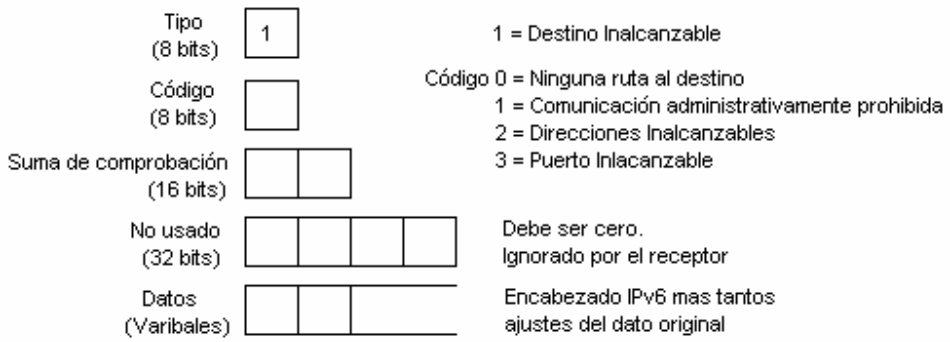


Figura 34. Formato del mensaje de destino inalcanzable.

El campo Código proporcionara mayor información acerca de las razones por las cuales el datagrama no fue entregado. Los códigos posibles son listados en la tabla 13. La porción de datos del mensaje ICMPv6 contiene parte del mensaje original – tanto como los ajustes dentro del mensaje ICMP.

Tabla 13. Código de valores de mensajes de destino inalcanzables (Tipo 1)

Código	Descripción
0	Ninguna ruta al destino Este mensaje es generado si un enrutador no puede enviar un paquete porque no tiene una ruta en su tabla para la red destino. Esto sólo puede pasar si el enrutador no tiene una entrada para un camino por default.
1	Comunicación con destino administrativamente prohibida Este tipo de mensaje, por ejemplo, puede ser enviado por un firewall que no puede enviar un paquete para un host dentro del firewall, porque es un paquete filtrado. También podría ser enviado si un nodo es configurado para no aceptar petición de eco (echo request) no autentificados.
2	Ámbito más allá de la dirección origen Este código es usado cuando el ámbito multicast de la dirección origen es más pequeño que el ámbito de la dirección destino.
3	Dirección Inalcanzable Este código es usado si una dirección destino no puede ser resuelta dentro de una dirección de red correspondiente o si hay un problema en la capa de enlace de datos proviniendo del nodo alcanzable de la red destino.
4	Puerto Inalcanzable Este código es usado por el protocolo de transporte (por ejemplo UDP), no tiene oyente o si no hay otro significado para enviar al emisor. Por ejemplo, si una petición DNS es enviado por un host y el servidor DNS no esta corriendo, este tipo de mensajes es generado.

Un nodo que recibe mensajes ICMPv6 de destino inalcanzable debe notificar a los procesos de capa superiores si el proceso identificado es relevante.

2.4.2.2.Paquete demasiado grande

Si un enrutador no puede enviar un paquete porque es mucho más grande que el MTU del enlace de salida, éste generará un mensaje de paquete demasiado grande que se muestra en la figura 43. Este tipo de mensaje ICMPv6 es usado como parte del proceso de descubrimiento del MTU del camino “Path MTU discovery”. El mensaje ICMP es enviado a la dirección destino del paquete que lo invoca.

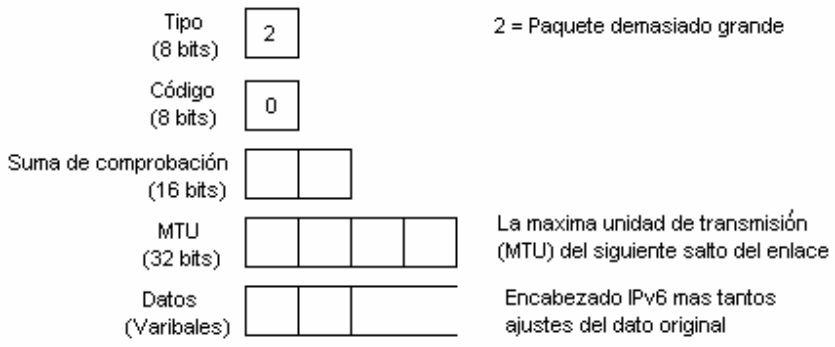


Figura 35. Formato del mensaje de paquete demasiado grande.

El campo Tipo el valor de 2, el cual identifica el paquete demasiado grande. En este caso, el campo de código no es usado y es puesto a cero. La información importante de este tipo de mensajes es el campo de MTU, el cual contiene el tamaño del MTU del siguiente salto del enlace.

El RFC4443 establece que un mensaje ICMPv6 no debe ser generado como una respuesta para un paquete con una dirección destino IPv6 multicast, una dirección multicast de capa de enlace, o una dirección broadcast en la capa de enlace. El mensaje de paquete demasiado grande es una excepción para esta regla. Porque el mensaje ICMP contiene el soporte MTU del siguiente salto del enlace, el host origen puede determinar el MTU que debe usar para futuras comunicaciones. Un host que recibe un mensaje de paquete demasiado grande deber informar al proceso de capa superior.

2.4.2.3.Tiempo excedido

Cuando un enrutador envía un paquete, éste siempre decrementa el límite de saltos a uno. Si un enrutador recibe un paquete con un límite de salto de uno y lo decrementa al límite cero, éste descarta el paquete, genera un mensaje de tiempo excedido con un código de valor cero, y envía este mensaje de regreso al host origen. Este error puede indicar un loop de encaminamiento o el hecho que el límite de saltos del emisor inicial sea demasiado bajo. Esta función también es usada para la utilidad del traceroute.

El campo Tipo transporta el valor de 3, especificando el mensaje de tiempo excedido. El campo de código puede ser cero, el cual significa límites de saltos excedido en el tránsito, ó 1, el cual significa que el tiempo de reensamblado de la fragmentación es excedido. La porción de datos del mensaje ICMP contiene parte del mensaje original - tanto como los ajustes dentro del mensaje ICMP, dependiendo del MTU usado. La figura 44 muestra el formato del mensaje de tiempo excedido.

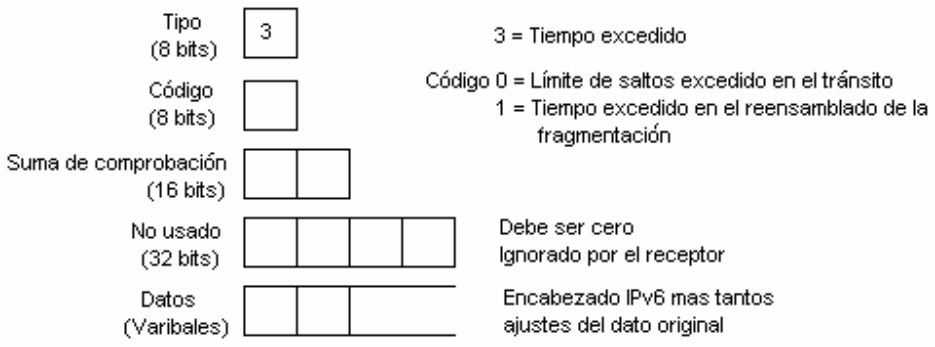


Figura 36. Formato del mensaje de tiempo excedido.

Un mensaje entrante de tiempo excedido debe ser pasado a los procesos de capas superiores. La tabla 14 muestra los valores del campo código para el mensaje de tiempo excedido.

Tabla 14. Valores de código para el mensaje de tiempo excedido (Tipo 3)

Código	Descripción
0	Límite de saltos excedido en el tránsito Posibles causas: el valor inicial del campo límite de saltos es demasiado bajo, o hay loops de encaminamiento.
1	Tiempo excedido en el reensamblado de la fragmentación Si un paquete fragmentado es enviado usando el encabezado de fragmentación, y el receptor no puede reensamblar todos los paquetes a un determinado tiempo, éste notifica al emisor utilizando este tipo de mensaje ICMP.

2.4.2.4.Problema del parámetro

Si un nodo IPv6 al procesar un paquete encuentra un problema en un campo en el encabezado IPv6 o en los encabezados de extensión, tal que no puede completar el procesamiento del paquete, se debe descartar el paquete y se debe originar un mensaje ICMPv6 de Problema del Parámetro para el paquete origen, indicando el tipo y la localización del problema. Este tipo de mensajes son utilizados frecuentemente cuando un error no se encuentra dentro de cualquiera de las otras categorías. El formato de este mensaje ICMP se muestra en la figura 45.

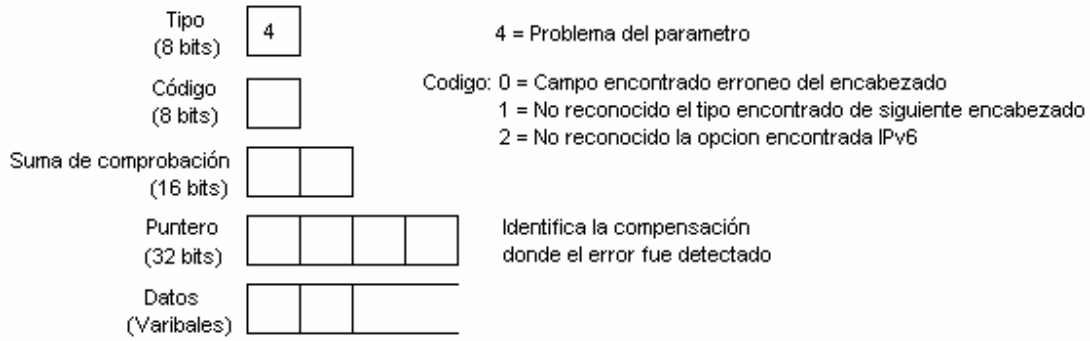


Figura 37. Formato del mensaje Problema del parámetro.

El campo tipo contiene el valor de 4, el cual especifica el mensaje de Problema del parámetro. El campo código puede contener cualquier de los tres valores descritos en la tabla 15. El campo de puntero identifica en qué byte del paquete original fue detectado el error. Los mensajes ICMP incluyen tanto datos originales, como ajustes mínimos del MTU IPv6.

Por ejemplo, un mensaje ICMPv6 con campo Tipo de 4, campo de Código de 1, y campo de puntero de 40 podría indicar que el encabezado de extensión seguido del encabezado IPv6 del paquete original, contiene un valor no reconocible del siguiente encabezado.

Tabla 15. Valores de los Códigos de Problema del parámetro (tipo 4)

Código	Descripción
0	Campo encontrado erróneo del encabezado
1	No reconocido el tipo encontrado de siguiente encabezado
2	No reconocido la opción encontrada IPv6

2.4.3. Mensajes Informacionales ICMP

En el RFC4443, se definen dos tipos de mensajes informacionales: Petición del Eco (Echo Request) y Contestación del Eco (Echo Reply). Otros mensajes informacionales ICMP son usados para el descubrimiento del MTU del trayecto (Path MTU Discovery) y descubrimiento de vecinos (neighbor discovery), estos mensajes se definen en los RFC1981 y RFC2461 respectivamente.

Los mensajes Petición de Eco y Contestación del Eco son usados para una de las más comunes utilidades TCP/IP: Ping (Packet Internet Groper). Ping es usado para determinar si un host específico está disponible en una red y si ya tiene comunicación. El host destino usa un mensaje Petición de Eco para el destino específico. El host destino, si está disponible, responde con un mensaje de Contestación del Eco.

2.4.3.1.Mensaje Petición de Eco.

El formato de los mensajes de Petición de Eco se muestra en la figura 46.

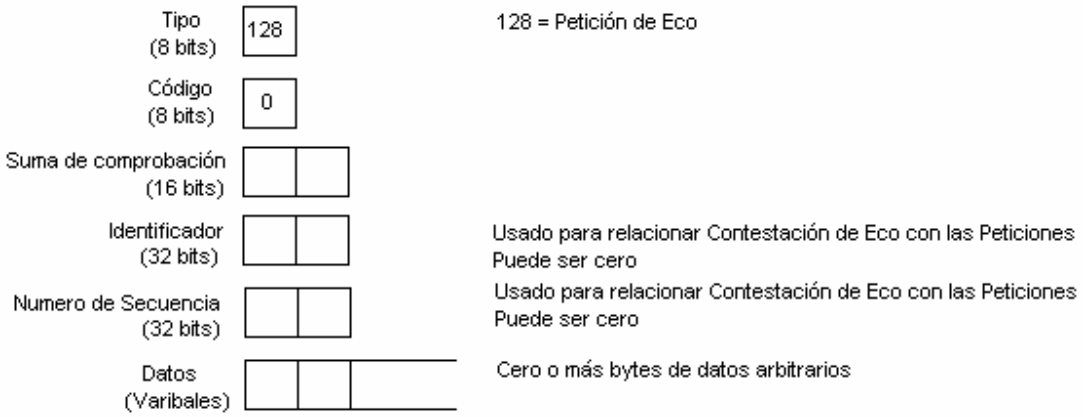


Figura 38. Formato de mensajes de Petición de Eco.

El campo Tipo es 128, el valor para Petición de Eco. El campo código no es usado para este mensaje y por lo tanto es cero. El campo identificador y Número de Secuencia es usado para relacionar las peticiones con las Contestaciones. La Contestación siempre debe contener el mismo número que las peticiones. Si un número de identificador y Número de Secuencia se están usando, el tipo de datos arbitrarios incluido en la Petición de Eco, depende de la Pila TCP/IP que se esté usando.

2.4.3.2.Mensaje de Contestación de Eco

El formato del mensaje de contestación de Eco es muy similar al de Petición de Eco, como se muestra en la figura 47.

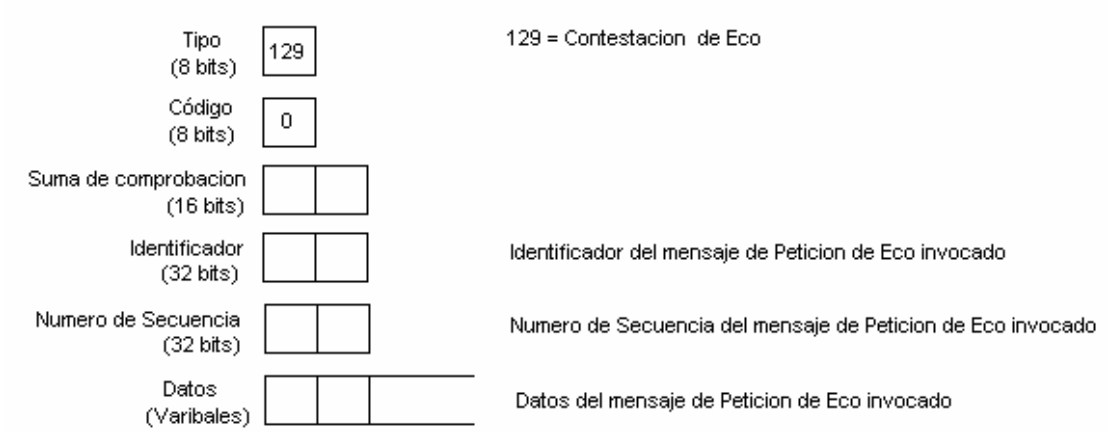


Figura 39. Formato de mensajes de Contestación de Eco

El campo Tipo contiene el valor de 129 para Contestación de Eco. El campo código no es usado y es puesto en cero. Los campos del identificador y número de secuencia deben relacionarse con los campos de Petición. El dato del mensaje de Petición de Eco debe ser copiado enteramente y no modificado dentro del campo de dato de Contestación. Si un proceso de capa superior inicia la Petición de Eco, la contestación debe ser pasada al proceso. Si el mensaje de Petición de Eco fue enviado para una dirección unicast, la dirección origen del mensaje de contestación de Eco debe ser la misma como la dirección destino del mensaje de Petición de Eco. Si la Péticion de Eco fue enviada por una dirección IPv6 Multicast, la dirección origen de la Contestación de Eco debe ser una dirección unicast de la interfaz en la cual la Petición de Eco Multicast fue recibida.
Mensajes ICMPv6 Petición de Eco y contestación de Eco pueden ser autentificados, usando un encabezado de autenticación. Esto significa que un nodo puede ser configurado para ignorar ping no autentificados ICMPv6 y proveer protección a diferentes tipos de ataques ICMPv6.

2.4.4. Reglas de procesamiento de los mensajes

Existen algunas reglas que gobiernan el procesamiento de los paquetes ICMP. Estas reglas se enlistan a continuación:

- Si un nodo recibe un mensaje de error ICMPv6 de tipo no conocido, éste debe pasarlo a la capa superior.
- Si un nodo recibe un mensaje informacional ICMPv6 de tipo no conocido, éste debe silenciosamente descartarlo.
- Como en ICMPv4, como sea posible, el paquete que causó el mensaje de error ICMP será incluido en el cuerpo del mensaje ICMP. El paquete ICMP no debe exceder el mínimo MTU IPv6.
- Si el mensaje de error ha sido pasado al protocolo de capa superior, el tipo de protocolo es determinado extrayéndolo del paquete original (presente en el cuerpo del mensaje de error ICMPv6). En caso de que el tipo de protocolo no sea encontrado en el cuerpo del mensaje ICMPv6 (porque existieron muchos encabezados de extensión en el paquete original, y la parte del encabezado que contenía el tipo de protocolo de capa superior, fue truncado), el mensaje ICMPv6 se descartará silenciosamente.

Un mensaje ICMPv6 no debe ser enviado en los siguientes casos:

- Como un resultado de un mensaje de error ICMPv6.
- Como un resultado de un mensaje de redirigir ICMPv6.
- Como un resultado de un paquete enviado para una dirección Multicast IPv6. Hay dos excepciones para esta regla: el mensaje de paquete demasiado grande que es usado para el descubrimiento del MTU del trayecto y el mensaje de problema del parámetro, código 2 para una opción IPv6 no reconocida.
- Como un resultado de un paquete enviado como un multicast de capa de enlace (con las excepciones que se mencionaron anteriormente).
- Como un resultado de un paquete enviado como un broadcast de capa de enlace (con las excepciones que se mencionaron anteriormente).
- Como un resultado de un paquete cuya dirección origen no identifica únicamente a un solo nodo. Por ejemplo, la dirección no especificada IPv6, una dirección multicast IPv6, o una dirección anycast IPv6 conocida por el mensaje ICMP origen.

Cada nodo IPv6 debe implementar una función de limitación de tasa (rate-limiting) que limita la tasa de mensajes ICMPv6 enviados. Los límites configurables pueden ser basados en contadores o en el ancho de banda. Si esta función es implementada propiamente, éste protege de los ataques mediante la denegación de servicios.

2.4.5. Descubrimiento de vecinos (Neighbor Discovery).

Descubrimiento de Vecinos (Neighbor discovery “ND”) se especifica en el RFC2461. Las especificaciones en este RFC mencionan los diferentes protocolos y procesos conocidos por IPv4 que han sido modificados y mejorados. Nuevas funcionalidades también han sido agregadas. Combina ARP (Address Resolution Protocol), descubrimientos de enrutadores ICMP (ICMP router Discovery) y Redirigir (Redirect). Con IPv4 no se tenía cómo detectar si un vecino era alcanzable o no. Con el protocolo de descubrimiento de vecinos, un mecanismo de detección de un vecino no alcanzable, es definido. Detección de direcciones IP duplicadas se ha implementado también. Nodos IPv6 usan descubrimientos de vecinos para los siguientes propósitos:

- Para determinar la dirección de capa 2 de nodos en el mismo enlace.
- Para encontrar enrutadores vecinos que pueden enviar sus paquetes.
- Para mantener un pista de cuáles vecinos son alcanzables y cuáles no, y detectar cambios en la direcciones de capa de enlace.

Las siguientes mejoras en el sistema de protocolos IPv4 puede ser notado:

- Descubrimiento de enrutadores es ahora parte de la base del sistema de protocolos. Con IPv4, el mecanismo necesitaba obtener la información de la tabla de enrutamiento.
- Los paquetes de anuncio de enrutadores contienen direcciones de capa de enlace para los enrutadores. No se necesita que el nodo reciba un anuncio de enrutador para enviar un petición adicional ARP (como un nodo IPv4 tendría que hacerlo) para obtener la dirección de capa de enlace para la interfaz del enrutador. También utilizado para mensajes de redirigir ICMPv6, éstos contienen las direcciones de capa de enlace de la interfaz del enrutador del siguiente salto nuevo.
- Los paquetes de anuncio de enrutadores contienen los prefijos para un enlace (información de Subnet). Ya no es necesario configurar máscaras de subred. Éstas pueden ser aprendidas de los mensajes de anuncio de enrutadores.
- Descubrimiento de vecinos provee mecanismos para renumerar redes fácilmente. Prefijos nuevos y direcciones pueden ser introducidos, y los viejos pueden ser desaprobados y removidos.
- Anuncio de enrutadores permite autoconfiguración de direcciones “stateless”y puede notificar a los hosts cuando el uso de configuración de direcciones “stateless” esté presente (por ejemplo DHCP).
- Los enrutadores pueden anunciar un MTU para ser usado en un enlace.
- Múltiples prefijos pueden ser asignados a un enlace. Por default, los hosts aprenden todos los prefijos del enrutador, pero el enrutador puede ser configurado para no anunciar algunos o todos los prefijos. En tal caso, los host asumen que un prefijo no anunciado, su destino es remoto, y envían los paquetes al enrutador. El enrutador puede entonces usar mensajes de redirigir ICMP como sea necesario.
- Detección de vecinos no alcanzables es parte de los protocolos base. Substancialmente mejora la liberación de paquetes en caso de fallas en los enrutadores o en interfaces de enlace que cambian su dirección de capa de enlace. Esto resuelve el uso del caché ARP. ND detecta fallas en la conectividad y el tráfico que no se envía a vecinos que no son alcanzables. Los vecinos no alcanzables también detectan falla en los enrutadores y switches.
- Anuncio de enrutadores y redirigir ICMP usa direcciones de enlace local para identificar enrutadores. Esto permite a los hosts mantener sus asociaciones de enrutadores en el caso de renumeración o uso de un nuevo prefijo global.
- Mensajes de descubrimientos de vecinos tienen un valor de límite de salto de 255, y peticiones con bajo límite de salto no son contestadas. Esto hace a descubrimiento de vecinos, inmune a hosts remotos que intentan husmear dentro del enlace; ya que sus paquetes se han decrementado límite de saltos y por lo tanto son ignorados.
- El protocolo de descubrimiento de vecinos es usado para detectar direcciones IP duplicadas en el enlace.

- Autenticación estándares IP y mecanismos de seguridad, pueden ser aplicadas al descubrimiento de vecinos.

El protocolo de descubrimiento de vecino consiste de cinco mensajes ICMP: un par de mensajes de Solicitud de enrutador/Anuncio de enrutador (Router Solicitation/Router Advertisement), un par de mensajes Solicitud de vecino/Anuncio de vecino (Neighbor Solicitation/Neighbor Advertisement), y un mensaje de Redirigir ICMP (ICMP Redirect). Estos tipos de mensajes ICMP se muestran en la tabla 12.

2.4.5.1.Solicitud de enrutador y Anuncio de enrutador

Los enrutadores envían mensajes de Anuncio de enrutador en intervalos regulares. Los Hosts pueden hacer peticiones de Anuncio de enrutador usando el mensaje Solicitud de enrutador. Esto accionará que los enrutadores utilicen inmediatamente Anuncio de enrutador, fuera de los intervalos regulares. El formato de los mensajes de Solicitud de enrutador se muestra en la figura 48.

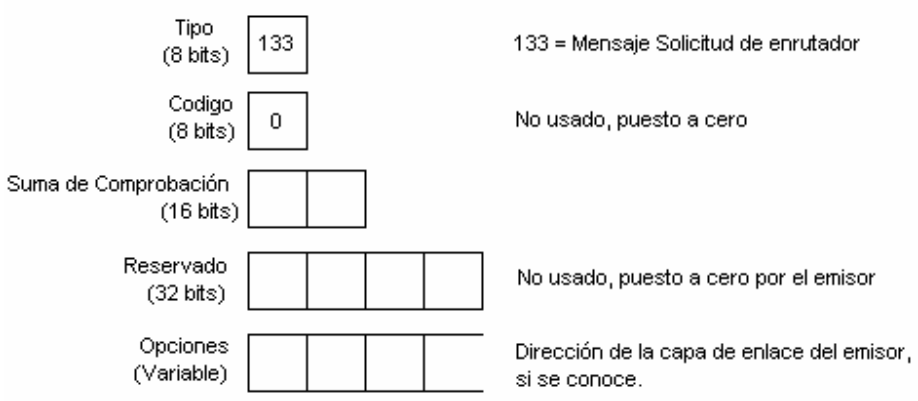


Figura 40. Mensaje de Solicitud de enrutador.

En el encabezado IP de un mensaje de Solicitud de enrutador, normalmente se utilizará una dirección multicast de todos los enrutadores FF02::2 como dirección destino. El límite de salto es de 25. Y el campo Tipo de ICMP es de 133, el cual es el valor para los mensajes de Solicitud de enrutador. El campo Código no es usado y tiene el valor de 0. Los siguientes dos bytes son utilizados para la suma de comprobación. Los siguientes cuatro bytes no son usados y están reservados para uso futuro, el emisor pone en cero estos campos y el receptor los ignora. Para un mensaje de Solicitud de enrutador, las direcciones de capa de enlace del host emisor son validas opciones, si se conocen. Si la dirección origen en la capa IP es la dirección no especificada (todo cero), este campo no es usado.

Los enrutadores que reciben este mensaje de Solicitud de enrutadores, contestan con un mensaje de Anuncio de enrutador. Los enrutadores también usan estos mensajes periódicamente. El formato de los mensajes de Anuncio de enrutadores se muestra en la figura 49.

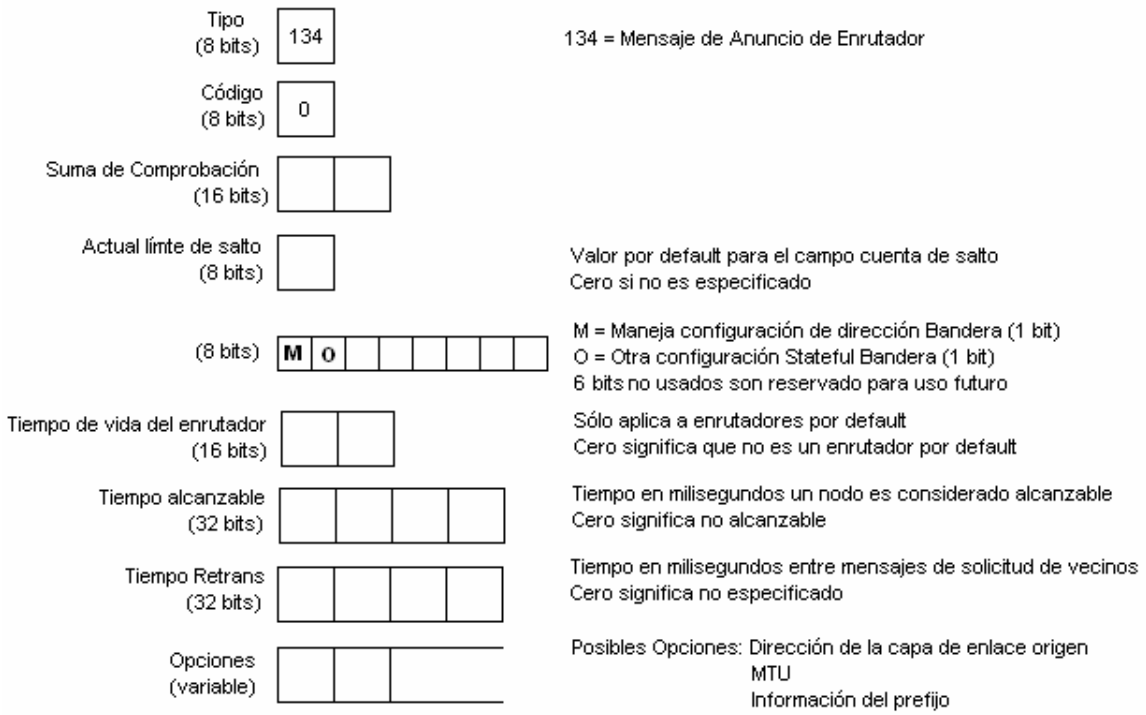


Figura 41. Mensaje de Anuncio de enrutador.

Por medio del encabezado IP de los mensajes de Anuncio de enrutador, se puede determinar si un mensaje de Anuncio de enrutador es periódico o fue enviado como contestación de un mensaje de Solicitud. La dirección destinos de mensajes de Anuncio de enrutador, sería una dirección multicast para todos los nodos de la forma FF02::1 si fueran mensajes periódicos. La dirección destino de un mensaje de Anuncio de enrutadores que fue Solicitada, sería la dirección de la interfaz que originó el mensaje de Solicitud de enrutadores. El límite de salto estaría en 255.

El campo Tipo de ICMP es puesto en 134, el valor para los mensajes de Anuncio de enrutador; el campo Código no es usado y es puesto en cero. El campo límite de salto actual puede ser usado para configurar todos los nodos en un enlace para un default límite de salto. El valor puesto en este campo será usado como un valor de límite de salto por default, en paquetes salientes por todos los nodos en el enlace. Un valor de cero en este campo significa que esta opción no está especificada por este enrutador, en tal caso, el valor de límite de salto por default del host origen será usado.

El siguiente campo de 1 bit, la bandera M, especifica si la configuración por Stateful se está usando. Configuración con Stateful se refiere a la configuración por DHCP en IPv4. Si este bit es 0, el nodo en este enlace utiliza autoconfiguración, si el bit está puesto a 1, especifica que es configuración por Stateful. La bandera O, configura si los nodos en este enlace utilizan configuración Stateful para otra información con excepción de direcciones IP. Un valor de 1 en este campo, significa que los nodos en este enlace utilizan configuración Stateful para información no relacionada con las direcciones. Los restantes 6 bits de este byte, son reservados para uso futuro y son puestos en 0.

El campo tiempo de vida del enrutador es importante sólo si el enrutador es usado como un enrutador por default por los nodos en este enlace. Un valor de 0 indica que este enrutador no es un enrutador por default y por consiguiente no aparecerá en la lista de enrutador por default de los nodos. Cualquier otro valor en este campo especifica el tiempo de vida, en segundos, asociado con el enrutador por default. El valor máximo es de 18.2 horas.

El campo de Tiempo alcanzable es el tiempo en el cual un host asume que un vecino es alcanzable, después de haber recibido una confirmación de alcance. Un valor de 0 significa no especificado. El algoritmo de detección de vecino no alcanzable usa este campo.

El campo Tiempo Retrans, es usado por la resolución de direcciones y el algoritmo de detección de vecinos no alcanzables. Es el tiempo en milisegundo entre mensajes de solicitud de vecinos retransmitidos.

Para el campo Opciones, actualmente hay tres posibles valores: dirección de la capa de enlace del origen, tamaño del MTU usado en enlaces con variables tamaños de MTU (por ejemplo Token Ring), e información del prefijo. Este campo es importante para la autoconfiguración. El enrutador inserta todos los prefijos para el enlace que los nodos en el enlace necesiten conocer.

2.4.5.2.Solicitud de vecino y Anuncio de vecino

Este par de mensajes lleva a cabo dos funciones: la resolución de direcciones de la capa de enlace, que es manejado por ARP en IPv4, y el mecanismo de detección de vecino no alcanzable. Si la dirección destino es una dirección multicast, el origen está resolviendo una dirección de capa de enlace. Si el origen está verificando el alcance de un vecino, la dirección destino es una dirección unicast. Este tipo de mensajes son también usados para la detección de direcciones IP duplicadas (Duplicate IP address Detection "DAD"). El formato del mensaje de solicitud de vecino se muestra en la figura 50.

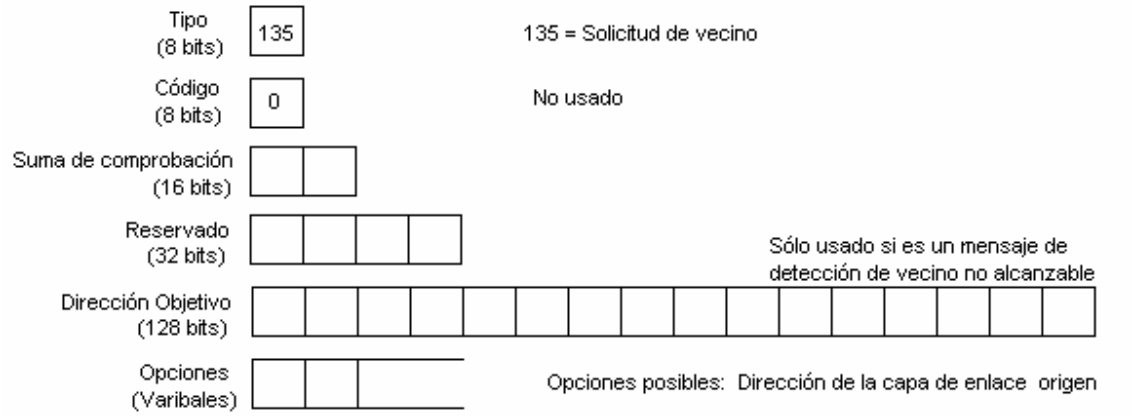


Figura 42. Formato del mensaje de solicitud de vecino.

En el encabezado IP de este tipo de mensaje, la dirección origen puede ser la dirección de la interfaz del hosts que lo origina, o en el caso de DAD la dirección no especificada (todo cero). El Límite de Salto es de 255. El campo Tipo en el encabezado ICMP es de 135, y el campo Código no es usado y puesto en cero. Después los dos bytes de suma de

comprobación y, cuatro bytes no usados que son reservados y deben estar en cero. La dirección objetivo es sólo usado en mensajes que se utilizan en la detección de vecino no alcanzable y DAD. No debe ser una dirección multicast.

El campo Opción puede contener la dirección de la capa de enlace del origen. Contiene sólo la dirección de capa de enlace, unicamente si no es un mensaje DAD. En un mensaje DAD, que usa la dirección no especificada como dirección origen, el campo opción es puesto a cero. La Opción de dirección de capa de enlace origen debe ser usada en solicitaciones multicast (descubrimiento de vecinos y funciones ARP) y debe ser usado en solicitaciones unicast (detección de vecinos no alcanzables).

Mensajes de anuncio de vecinos, son enviados como contestación de los mensajes de solicitud de vecinos o para propagar información rápidamente. El formato de los mensajes de anuncio de vecinos se muestra en la figura 51.

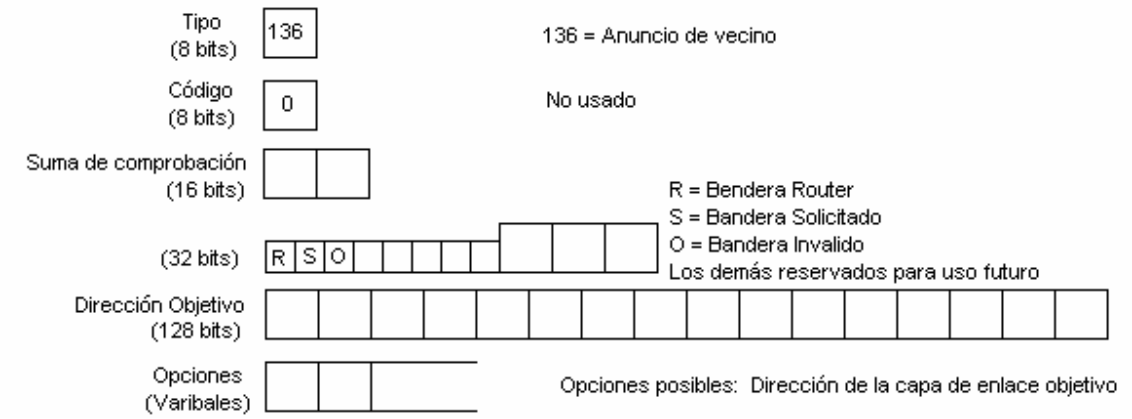


Figura 43. Formato del mensaje de anuncio de vecino

El tipo de dirección en el encabezado IP indica si el mensaje es la contestación a una solicitud o si el mensaje no es una contestación. En caso de un mensaje de anuncio solicitado, la dirección IP destino es la dirección origen de la interfaz que envió la solicitud. Si el mensaje es la contestación para un mensaje DAD que fue originado de una dirección origen no especificada, la contestación será a la dirección multicast de todos los nodos FF02::1. De la misma forma será también para todos los mensajes periódicos de anuncios no solicitados.

El campo Tipo en el encabezado ICMP tiene el valor de 136, que identifica a los mensajes de anuncio de vecinos. El campo Código no es usado y es puesto en 0.

Cuando la bandera R es puesta, indica que el emisor es un enrutador. Esta bandera es usada por el mecanismo de detección de vecinos no alcanzables, para detectar a un enrutador que cambia para un host.

Cuando la bandera Solicitado es puesta, el mensaje se envía en respuesta a una solicitud de vecino. En el caso de que un host confirme que fue alcanzado en respuesta a un mensaje de

detección de vecinos no alcanzables, el bit S es puesto. El bit S no es puesto en anuncios multicast o en anuncios no solicitados unicast.

La bandera Inválido (Override) indica que la información en el mensaje de anuncio debe invalidar una entrada existente en la caché de vecinos y actualizar las direcciones de capa de enlace en la caché. Si el bit O no es puesto, los anuncios no actualizarán las direcciones de capa de enlace del caché, pero actualizará una entrada existente del vecino en la caché para la dirección de capa de enlace que no exista. El bit O no debe ser puesto en un anuncio para una dirección anycast. Los restantes 29 bits son reservados para uso futuro y son puestos en 0.

En anuncios solicitados, la dirección objetivo (Target Address) contiene la dirección de la interfaz que envió la solicitud. En anuncios no solicitados, este campo contiene la dirección de la interfaz en la cual su dirección de capa de enlace ha cambiado. Esta dirección no debe ser una dirección multicast. Una posible opción para el campo Opción, es la dirección de capa de enlace objetivo (Target Link-Layer Address), que es la dirección de capa de enlace para el objetivo, por ejemplo, el emisor del anuncio.

2.4.5.3.Mensaje ICMP de Redirigir (Redirect)

Los enrutadores envían Redirigir paquetes para informar a un nodo de un mejor primer salto en el camino hacia el destino. Un mensaje de Redirigir puede también informar a un nodo que el destino usado es de un vecino en el mismo enlace y no un nodo de una subred remota. El formato del mensaje de Redirigir ICMPv6 se muestra en la figura 52.

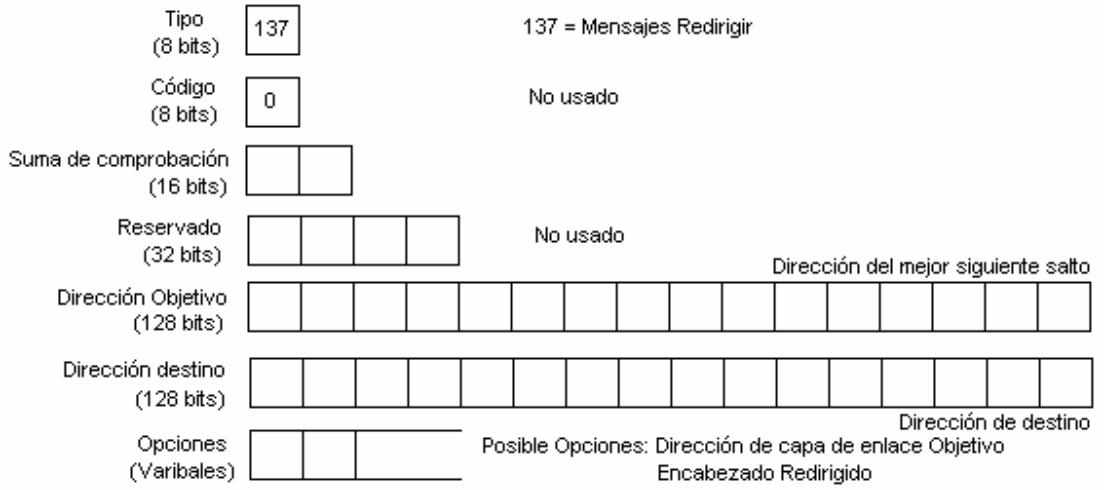


Figura 44. Formato de mensajes de Redirigir.

La dirección origen en el encabezado IP, debe ser la dirección de enlace local de la interfaz de donde el mensaje es enviado. La dirección destino en el encabezado IP es la dirección origen del paquete que activó al mensaje Redirigir. El límite de salto es puesto a 255.

El campo de la Dirección Objetivo (Target Address) contiene la dirección de enlace local de la interfaz, que es un mejor siguiente salto a usar para la dirección de destino dada. El campo de la dirección destino contiene la dirección del destino, que es redirigida, cuál se

debe utilizar para obtener a la dirección destino. Si la dirección en el campo de Dirección Objetivo es la misma que en el campo de dirección destino, el destino es un vecino y no un nodo remoto. El campo Opción contiene la dirección de capa de enlace de la Dirección Objetivo (el enrutador para el mejor siguiente salto). Esto es una mejora a la versión IPv4, en la cual el host necesitaba usar por separado una petición ARP para determinar la dirección de capa de enlace del enrutador para el siguiente salto.

2.4.5.4.Caches de Vecinos y Destinos

Los nodos IPv6 necesitan mantener diferentes tablas de información para cada interfaz. Entre estas tablas encontramos la de caché de vecinos y caché de destinos que son particularmente importantes. A continuación se muestra una descripción de estas dos listas:

- Caché de Vecinos: El caché de vecinos mantiene una lista de vecinos a la cual se les ha mandado tráfico recientemente. Ellos son listados por su dirección IP unicast, cada entrada contiene información acerca de las direcciones de capa de enlace y una bandera indicando si el vecino es un enrutador o host. Esto puede ser comparado con el caché ARP en un nodo IPv4. Las entradas también contienen información acerca de la existencia de paquetes en cola para enviar información acerca de vecinos alcanzables, y el tiempo programado para el siguiente evento de detección de vecino no alcanzable que se llevará acabo.
- Caché de Destinos: Esta tabla contiene información acerca de destinos a los que se ha enviado tráfico recientemente, incluyendo destinos locales y remotos. El caché de destinos puede ser visto como un subsistema de información caché de destinos. En caso de destinos remotos, las entradas listan las direcciones de capa de enlace del siguiente enrutador a saltar. El caché destino es actualizado con información recibida de mensajes ICMP de Redirigir. Puede también contener información adicional como tamaño del MTU y contadores “round trip” mantenido por protocolos de transporte.

Los caches de vecinos y destinos se relacionan con la bandera de inválido (O) que puede ser puesta en el mensaje de anuncio de vecino. Si la bandera O es puesta, la información en el mensaje de anuncio de vecino debe invalidar entradas existentes en el caché de vecinos, y actualizar las direcciones de capa de enlace en el caché de los hosts que reciben los anuncios. Si el bit O no esta puesto, el anuncio no actualizará la dirección de capa de enlace en los cachés, pero actualizaran una entrada existente en el caché de vecinos para la dirección de capa de enlace no existente.

Una entrada en el caché de vecinos puede ser uno de cinco estados, según lo establece el RFC2461. Los cinco estados son mencionados en la tabla 16.

Tabla 16. Estado en las entradas en el caché de vecinos.

Estado	Descripción
Incomplete	La resolución de dirección está en progreso y la dirección de capa de enlace del vecino aún no ha sido determinada.
Reachable	El vecino es actualmente alcanzable, lo que significa que fue recibida la confirmación recientemente, desde hace 10 segundos.
Stale	El vecino no es tan conocido para ser alcanzable pero hasta que el tráfico se

	envié al vecino, no se debe hacer ninguna verificación tentativa de su alcance.
Delay	El tiempo del vecino alcanzable ha expirado, y un paquete fue enviado dentro del último "DelayFirstProbe Time Seconds". Si no se recibe confirmación dentro del "DelayFirstProbe Time Seconds", entonces se envía una solicitud de vecino y cambia el estado del vecino a estado Probe.
Probe	El vecino no es tan conocido para ser alcanzable, y solicitud de vecinos unicast se esta enviando para verificar su alcance.

2.4.5.5.Autoconfiguración

La autoconfiguración es el conjunto de pasos por los cuales un host decide cómo autoconfigurar sus interfaces en IPv6. Este mecanismo es el que nos permite decir que IPv6 es "Plug & Play". El proceso de autoconfiguración incluye la creación de una dirección de enlace local, verificar que no este duplicada en el enlace, y determinar qué información debe ser autoconfigurada (direcciones, otra información, o ambas). En el caso de direcciones, si debe ser obtenida mediante el mecanismo de stateless, stateful, o ambas.

IPv6 define dos mecanismos de configuración de direcciones stateful y stateless autoconfiguración, que se describen a continuación:

- **Stateful.**-La autoconfiguración stateful es lo que en IPv4 llamamos DHCP, para IPv6 es un protocolo UDP cliente/servidor, diseñado para reducir el coste de gestión de nodos IPv6 en entornos donde los administradores precisan un control sobre la asignación de los recursos de la red, superior al facilitado por el mecanismo de configuración "stateless".

  En el modelo de autoconfiguración stateful, los hosts obtienen las direcciones de sus interfaces, parámetros, y/o información de configuración de un servidor. El servidor mantiene una base de datos de qué direcciones ha asignado a qué hosts. El protocolo de autoconfiguración stateful permite a los hosts obtener direcciones, otra información de configuración o ambos del servidor. Stateful y stateless se complementan.

  Si se tiene una red IPv6, no se necesita DHCP para configurar los hosts con información de direcciones. Con el mecanismo de autoconfiguración stateless se configuran las direcciones IPv6 de los hosts, sin la necesidad de tener un servidor DHCP. Todo lo que se necesita es habilitar al enrutador IPv6 con la información del prefijo para el enlace. Pero aún se puede escoger tener servidores DHCP para algunos casos. La configuración de host usando DHCP con IPv6 es llamado configuración stateful. Quizás se tenga un esquema de direccionamiento IPv6 específico, se necesite asignación dinámica de servidores DNS, o no tener direcciones MAC como parte de las direcciones IPv6. En estos casos se puede usar DHCP para la configuración de direcciones o tener hosts IPv6 sobre un enlace sin enrutadores IPv6, en tal caso se podría necesitar usar DHCP para distribuir los prefijos para configurar los hosts IPv6. Se puede combinar stateful y stateless para usar la autoconfiguración de direcciones IPv6 y servidores DHCP para proveer información adicional de configuración.

- **Stateless**.-Lo nuevo en IPv6 es que los hosts pueden autoconfigurar sus direcciones IPv6 sin ninguna configuración manual del host. El mecanismo stateless permite a un host generar su propia dirección, usando una combinación de información local disponible como su dirección MAC e información anunciada por los enrutadores. El enrutador anuncia el prefijo que identifica a la subred asociada con el enlace, mientras el host genera un "identificador de interfaz" que únicamente identifica su interfaz en la subred. La dirección es formada por la combinación de los dos datos. En ausencia de enrutador, un host puede generar únicamente la dirección de enlace local FE80, esta dirección es suficiente para permitir la comunicación entre nodos en el mismo enlace. Por ejemplo si en un sitio se cambia de ISP y, el nuevo ISP asigna un nuevo prefijo IPv6, se pueden configurar los enrutadores para que anuncien el nuevo prefijo, y que se mantenga el SLA que se usaba con el viejo prefijo. Todos los host entonces generan sus nuevas direcciones del nuevo prefijo, manteniendo el SLA del viejo prefijo.

  Stateless se usa cuando un sitio no necesita asignar direcciones estrictamente exactas a un hosts para que use, sólo que sean únicas y apropiadas para enrutar, en cambio stateful se usa cuando un sitio requiere estricto control de las direcciones asignadas a los hosts. Ambos mecanismos pueden ser usados simultáneamente. El administrador del sitio especifica qué mecanismo de autoconfiguración usará, con sólo ponerlo en el campo apropiado de los mensajes de anuncio de enrutador.

Una dirección IPv6 es asignada a una interfaz por un cierto tiempo de vida. Cuando el tiempo de vida expira, la dirección se convierte en invalida (invalid) y ésta puede ser reasignada a otra interfaz. Las direcciones pasan por diferentes estados para asignarlas a una interfaz, estos estados son los siguientes:

- **Dirección Tentativa**.- La dirección tentativa (tentative) aún no ha sido asignada. Es el estado antes de ser asignada, es cuando su unicidad está siendo verificada. Una interfaz descarta paquetes recibidos para la dirección tentativa, pero acepta paquetes de descubrimiento de vecinos, relacionado con la detección de direcciones duplicadas.

- **Dirección Preferida**.- La dirección preferida (preferred) ha sido asignada a una interfaz y puede ser usada sin ninguna restricción. Estas direcciones pueden utilizarse como dirección origen o destino para paquetes enviados de o para la interfaz.

- **Dirección Desaprobada**.- La dirección desaprobada (deprecated) es una dirección asignada a una interfaz que su uso es desaconsejado, pero no prohibido. Una dirección desaprobada no debe ser usada como una dirección origen en nuevas comunicaciones, pero paquetes enviados de o para direcciones desaprobadas son liberadas, como se espera. Una dirección desaprobada podría continuar usándose como dirección origen en comunicaciones en donde el cambiar a direcciones preferida, causa mal funcionamiento a actividades específicas de capas superiores, por ejemplo una conexión existente TCP.

Cuando un nodo es autoconfigurado, se realizan los siguientes pasos:

1. una dirección de enlace local es generada usando el prefijo de enlace local FE80 y añadiendo el identificador de interfaz. Esta dirección es una dirección tentativa.
2. el nodo acopla los siguientes grupos multicast: el grupo multicast de todos los nodos (FF02::1) y el grupo multicast de nodo-solicitado para la dirección tentativa.
3. un mensaje de solicitud de vecino es enviado con la dirección tentativa como dirección objetivo. La dirección IP origen de este mensaje es la dirección no especificada (todo los campo en cero); la dirección IP destino es la dirección multicast de nodo-solicitado. Esto detecta si otro nodo en el enlace ya usa esta dirección; esto es DAD. Si existe un nodo con esta dirección, éste contesta con un mensaje de anuncio de vecino y el mecanismo de autoconfiguración se suspende. En esta situación, la configuración manual del host es requerida. Si no es contestado el mensaje de solicitud de vecino, la dirección es asignada a la interfaz y su estado cambia a "preferida". La conectividad IP es ahora establecida. Hasta este punto, el proceso es el mismo par los hosts y enrutadores. Sólo los hosts realizan los siguientes pasos.
4. para determinar qué enrutadores hay y qué prefijos, el host envía un mensaje de solicitud de vecino para el grupo multicast de todos los enrutadores FF02::2.
5. todos los enrutadores en el enlace contestarán con un anuncio de enrutador. Por cada prefijo en el anuncio de enrutador con la bandera de autónomo puesta, una dirección es generada, combinando el prefijo con el identificador de interfaz. Estas direcciones son agregadas a la lista de direcciones asignadas a la interfaz. Cabe destacar que esto sucede cuando existe enrutador en el enlace, pero si no existiera enrutador, entonces se continuaría con la configuración stateful.

Para estar seguro que todas las direcciones asignadas a un enlace dado son únicas, los nodos corren un algoritmo de detección de direcciones duplicadas (duplicate address detection "DAD") antes de asignarlas a las interfaces. El algoritmo de detección de direcciones duplicadas es aplicado a las direcciones, independientemente si han sido obtenidas por autoconfiguración stateless o stateful. Si la dirección de enlace local fue generada por el mecanismo de autoconfiguración, usando el identificador de interfaz, la unicidad que ha sido verificado en el paso 3, podría no se repetido para direcciones adicionales que usen el identificador de interfaz. Todas las direcciones configuradas manualmente o por stateful necesitarán ser verificadas individualmente.

2.4.5.6.Descubrimiento del MTU del trayecto

Con IPv4, cada enrutador puede fragmentar paquetes, si es necesario. Si un enrutador no puede enviar un paquete porque el MTU del siguiente enlace es más pequeño que el del paquete a enviar, entonces el enrutador fragmenta el paquete. El paquete original es dividido en pequeños partes, de tal forma que se ajusten al tamaño del MTU, entonces el paquete ahora es enviado como un sistema de fragmentos. En el destino se recibe cada fragmento del paquete y se reensamblan. Dependiendo del diseño de la red, un paquete IPv4 puede ser fragmentado más de una vez durante su traslado en la red.

Con IPv6 los enrutadores ya no tienen que fragmentar paquetes, el emisor toma sus precauciones para esto. Descubrimiento del MTU del trayecto se asegura que el paquete es enviado usando el tamaño más grande posible que es soportado en una cierta ruta. El MTU del trayecto usado es el más pequeño MTU de todos los enlaces del origen al destino.

El proceso de descubrimiento se realiza de la siguiente forma. Primero un host asume que el MTU del trayecto es el mismo MTU que el del primer salto en el enlace, y usa el tamaño de éste. Si el paquete es demasiado grande para algún enrutador a lo largo del trayecto, entonces el enrutador descarta el paquete y envía de regreso un mensaje ICMPv6 de paquete demasiado grande. Este tipo de mensaje incluye el tamaño del MTU del siguiente salto del enlace. El host ahora usa este MTU para enviar paquetes futuros al mismo destino. El host no reducirá el tamaño del MTU por debajo del valor mínimo del MTU a 1280 bytes. El proceso de recibir un paquete demasiado grande y reducirlo puede repetirse más de una vez, antes que el paquete alcance su destino. El proceso de descubrimiento finaliza cuando el paquete llega a su destino final.

El trayecto de un origen dado hacia el destino dado puede cambiar, y también el MTU del trayecto. Tamaños de MTU más pequeños son descubiertos por mensajes de paquete demasiado grande. Un host IPv6 intentará incrementar el tamaño del MTU en el orden que sea capaz de detectar un MTU del trayecto más grande. Descubrimiento del MTU del trayecto también soporta destinos multicast. Si el destino es multicast, entonces habrán muchos trayectos que el paquete podría cruzar, y cada trayecto pueden tener diferentes MTU del trayecto. Mensajes de paquete demasiado grande se generaran sólo con un destino unicast, y el tamaño del paquete usado por el emisor será el más pequeño MTU del trayecto del conjunto de destinos.

2.4.5.7.Multicast Group Management

El grupo de direcciones Multicast son usadas como un identificador para un grupo de nodos. Son identificadas por el byte de mayor orden FF. Un protocolo es requerido para manejar el enrutamiento eficiente de paquetes con direcciones de grupo multicast como un destino.

El manejo de grupos multicast en IPv4 es realizado por el protocolo IGMP (Internet Group Management Protocol). La versión 2 de este protocolo se define en el RFC2236. IPv6 usa mensajes ICMPv6 para la misma función; ésta fue desarrollada con base en las especificaciones IGMPv2. Pero ahora es llamada MLD (Multicast Listener Discovery).

Todos los mensajes MLD son enviados con una dirección origen de enlace local IPv6 y un límite de salto de uno para asegurarse que permanezcan en la red local. Si el paquete tiene un encabezado de opción de salto por salto, se tienen que poner la bandera de “Router Alert”. Así, los enrutadores no ignoraran los paquetes, aunque ellos no escuchen la dirección de grupo multicast en cuestión.

Los tres tipos de mensajes tienen el mismo formato, el cual se muestra en la figura 53.

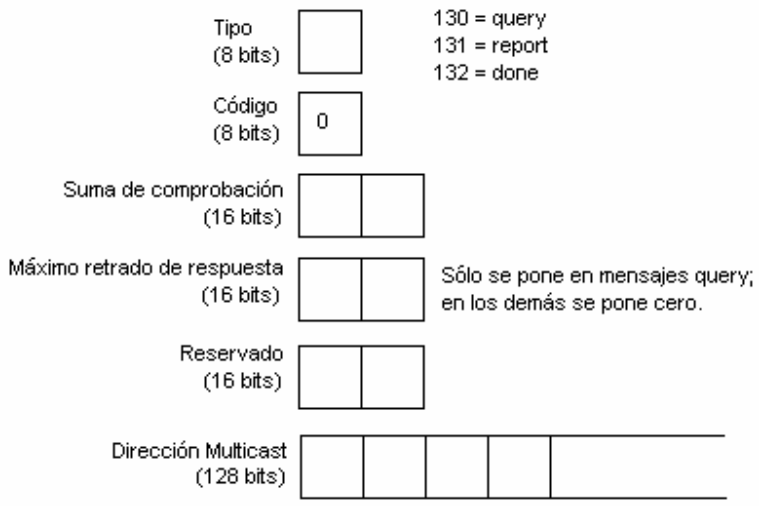


Figura 45. Formato de los mensajes MLD.

El campo Tipo es 130 para mensajes "Multicast Listener Queries", 131 para "Multicast Listener Reports", y 132 para "Mulicast Listener Done". Hay dos tipos de mensajes query. Uno es un query general, utilizado para determinar cual grupo de direcciones multicast están escuchando en el enlace. La otra es un query de dirección-especifica, utilizado para determinar si está escuchando una dirección específica en el enlace. El campo de máximo retardo de respuesta, es usado sólo en mensajes query. En todos los demás mensajes, este campo es puesto en 0. El campo de dirección multicast es puesto en cero en un query general. En un query de dirección-especifica, éste contiene la dirección del grupo multicast que se requiere. En mensajes report y done, éste campo contiene el grupo multicast, en el que el miembro escucha o sale del grupo.

Enrutadores usan MLD para descubrir qué direcciones multicast están escuchando en cada uno de sus enlaces. Para cada enlace, el enrutador mantiene una lista de las direcciones que se encuentran escuchando.

Queries generales enviados a direcciones multicast con ámbito de enlace local para todos los nodos FF02::1. Cualquier nodo que quiere enviar un reporte en respuesta a un query, inicia un contador cuando se recibe el query y se espera un tiempo aleatorio de retardo antes de enviar el reporte. El máximo retardo es el especificado en el campo máximo retardo de respuesta en el query. Si dentro del retardo, la estación ve a otra estación enviado en reporte, se detiene el proceso. Así, múltiples reportes para la misma dirección es permitido. Miembros del grupo reúnen los reportes y terminaciones son enviados a las direcciones en cuestión.

La dirección de ámbito de enlace local de todos los nodos (FF02::1) es una dirección especial. Ésta nunca debe ser enviada como un mensaje de report o un done. Si una dirección tiene en ámbito de 1 (nodo-local), nunca debe enviarse mensajes MLD. La tabla 17 describe tipos de mensajes y sus direcciones destino.

Tabla 17. Tipos de mensajes y sus destinos.

Tipo de Mensajes	**Direcciones destinos IPv6**
General query	Ámbito enlace local de todos los nodos (FF02::1)
Query de dirección-multicast-especifica	La dirección multicast que es del queried
Reporte	La dirección multicast que es del reported
Done	Ámbito enlace local de todos los enrutadores (FF02::2)

Capítulo

3.Mecanismos de transición de IPv4 a IPv6

Resumen

El capítulo desarrollará algunos mecanismos utilizados para implementar el protocolo IPv6 sobre infraestructura de enrutamiento IPv4 existente. En la actualidad el protocolo de Internet versión 4, se encuentra desplegado en la totalidad de las redes existentes. Para que se logre una exitosa migración a IPv6, es necesario que ambos protocolos convivan por largo tiempo, y así obtener experiencia de su funcionamiento, como también seguirlo desarrollando; para que en un futuro se tenga el soporte necesario y la migración sea exitosa. Al finalizar este capítulo se conocerán las bases necesarias para implementar una red IPv6 que esté interoperando con una infraestructura de red IPv4.

3.1. INTRODUCCIÓN

El desarrollo de IPv6 depende en gran parte, de la coexistencia con IPv4. Para que se logre una exitosa transición a la nueva versión del protocolo IPv6, primero tendrá que coexistir con las extensas redes ya existentes basadas en IPv4. Es por tal razón que se ha desarrollado un sistema de mecanismos que pueden ser implementados para que host y enrutadores IPv6 puedan ser compatibles con host y enrutadores IPv4 en las actuales infraestructuras de enrutamiento IPv4. Cada mecanismo de transición puede ser clasificado como también pertenecer a una o más de las siguientes categorías:

- Capa dual o Dual Stack
- Encapsulación (túnel)
- Traducción[14]

Cada categoría describe la metodología básica de un mecanismo. Un mecanismo de transición podría pertenecer a más de una categoría y frecuentemente trabajar en conjunto. Estos mecanismos pueden ser usados en combinación, una con otro como sea apropiado. La migración a IPv6 puede ser hecho paso por paso, empezando por un solo host o subred, se puede migrar la red completa o parte de ella.

3.2. CAPA DUAL O DUAL STACK

Un nodo Dual Stack tiene soporte completo para ambas versiones del protocolo IP. Frecuentemente se hace referencia a este tipo de nodos como un nodo IPv6/IPv4. En la comunicación con un nodo IPv6, estos nodos se comportan como un nodo solamente-IPv6 (IPv6-only), y en comunicación con un nodo IPv4, se comportan como un nodo solamente-IPv4 (IPv4-only). En implementaciones probablemente tiene funciones de switch para que habilite o deshabilite una de las pilas. Así estos nodos tienen tres modos de operación: cuando la pila IPv4 esta habilitada y la pila IPv6 deshabilitada, el nodo se comporta como un nodo solamente-IPv4; cuando la pila IPv6 esta habilitada y la IPv4 esta deshabilitada, éste se comporta como un nodo solamente-IPv6; y finalmente cuando ambas pilas de IPv4 e IPv6 están habilitadas, el nodo puede usar ambos protocolos.

Un nodo IPv6/IPv4 tiene al menos una dirección por cada versión del protocolo. Usa mecanismos IPv4 para configurar su dirección IPv4 (configuración estática o DHCP) y usa mecanismos IPv6 para configurar su dirección IPv6 (configuración estática o autoconfiguración). Dual Stack permite que los host continúen utilizando recursos IPv4, mientras a la vez agregan funcionalidades IPv6. Aplicaciones corriendo en los nodos son actualizados haciendo uso de la pila del protocolo IPv6. Aplicaciones que no son actualizados –sólo soportan la pila del protocolo IPv4- pueden coexistir con aplicaciones actualizadas en el mismo nodo. La figura 54 muestra a un host con Dual Stack.

[14] http://www.ipv6style.jp/en/building/20030820/index.shtml

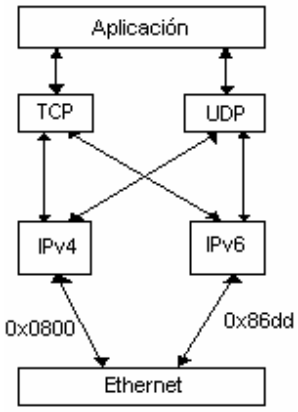


Figura 54. Nodo Dual Stack

El DNS (Domain Name System) es usado con ambas versiones del protocolo para resolver nombres y direcciones IP. Un nodo IPv6/IPv4 necesita un "DNS resolver" que sea capaz de resolver ambos tipos de direcciones de registro DNS. El registro DNS A es usado para resolver direcciones IPv4 y el DNS AAAA o registro A6 es usado para resolver direcciones IPv6.

En algunos casos, el DNS regresa sólo una dirección IPv4 o IPv6, si el host que resolvió es Dual Stack, el DNS podría regresar ambos tipos de direcciones. Esperanzadamente, para este caso, el "DNS resolver" en el cliente y una aplicación usando DNS, tendría que configurar opciones que nos permita especificar o filtrar la forma de usar la dirección. Generalmente, aplicaciones que están hechas para correr en nodos Dual Stack necesitan un mecanismo para determinar si ésta es una comunicación con un IPv6 peer o un IPv4 peer.

Una red Dual Stack es una infraestructura en la cual IPv4 e IPv6 son habilitados en los enrutadores. La desventaja de esta técnica es que se deben actualizar las características de software de la red completa para correr las dos pilas del protocolo. Esto significa que todas las tablas (por ejemplo tablas de enrutamiento) son mantenidas simultáneamente, protocolos de enrutamiento serán configurados para ambos protocolos. Para administradores de red se tiene diferentes comandos para cada protocolo, esto significa que toma más memoria y procesamiento de CPU.

En la figura 55, una aplicación que soporta ambos protocolos IPv4 e IPv6, solicita todas las direcciones disponibles para el host destino con nombre www.a.com al servidor DNS. El servidor DNS contesta con todas las direcciones disponibles (ambas direcciones IPv4 e IPv6) de www.a.com. La aplicación escoge una dirección –en la mayoría de los casos, la dirección IPv6 es la elegida por default- y conecta el nodo origen al destino usando la pila del protocolo IPv6.

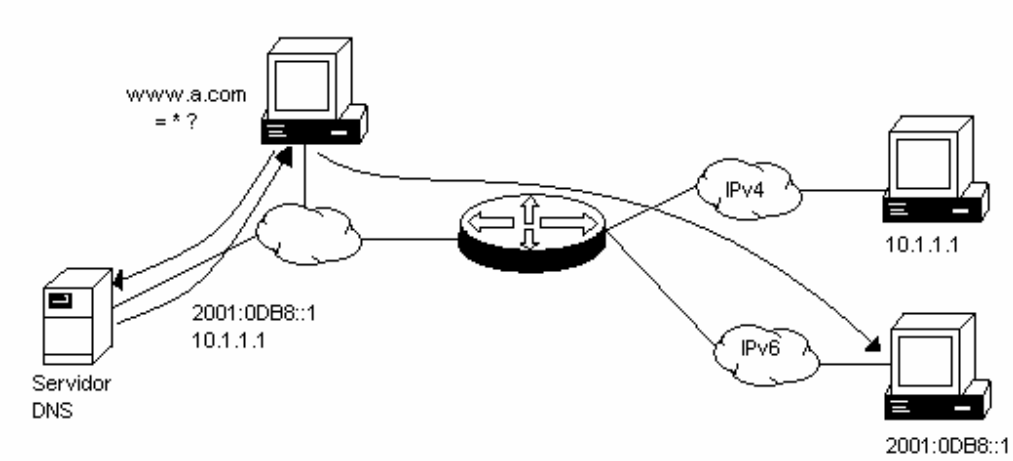


Figura 55. Aplicación con doble pila de protocolo IPv4 e IPv6

3.3. ENCAPSULACIÓN O TÚNEL

Este método consiste en encapsular y transportar dentro de la carga útil un paquete IPv6, el resultado de este paquete es llamado un "paquete túnel IPv6", la trayectoria entre el origen y el destino del paquete túnel es llamado un túnel IPv6, por lo tanto la técnica es llamada túnel IPv6.

El mecanismo de túnel puede ser usado para desarrollar una infraestructura IPv6, mientras la infraestructura IPv4 total es aún la básica. Túneles pueden ser usados para transportar tráfico IPv6 encapsulados dentro de paquetes IPv4 y enviarlos sobre la infraestructura de IPv4. Por ejemplo, si tu proveedor tiene aún una infraestructura solamente-IPv4, los túneles te permiten tener una corporación con red IPv6 y hacer túneles através de la red IPv4 de tu ISP para alcanzar otras redes o host IPv6.

Existen dos tipos de túneles especificados, los cuales son:

- Túnel Configurado: es una técnica donde las direcciones IPv4 del punto final del túnel, son determinadas por la información de configuración del nodo que encapsula el paquete. El túnel puede ser unidireccional o bidireccional. Un túnel IPv6 es un mecanismo unidireccional, esto es, el flujo de los paquetes van en una sola dirección, entre el nodo que funciona como punto de entrada del túnel IPv6 y el nodo que sirve como punto de salida. Para que el túnel sea bidereccional se tendrá que configurar dos túneles, cada uno en dirección opuesta del otro, el nodo de punto de entrada de un túnel, es el nodo de punto de salida del otro túnel.
- Túnel Automático: es una técnica donde la dirección IPv4 del punto final del túnel, es obtenida de la dirección IPv4 incrustada en la dirección destino compatible-IPv4 del paquete IPv6 a ser tuneleada; este tipo de direcciones son asignadas exclusivamente a nodos que usan túneles automáticos.

En muchos casos la combinación de ambas técnicas de túneles es usada para host IPv6/IPv4 que están conectados a segmentos en donde el enrutador no soporta IPv6. Así los host pueden tener dos entradas de enrutamiento por túnel. Si un host enviando un paquete tiene

ambas direcciones IPv6 IP4-compatible y dirección IPv6 global nativa, el host debe usar la dirección IPv4-compatible como dirección origen para paquetes con destinos IPv6 IPv4-compatible, y usar la dirección IPv6 nativa como dirección origen para paquetes con destino IPv6 nativa.

3.3.1. Cómo funcionan los túneles

El funcionamiento de los túneles a ser explicado se aplica a cualquier clase de túneles. La figura 56 muestra dos redes conectadas através de una red solamente-IPv4.

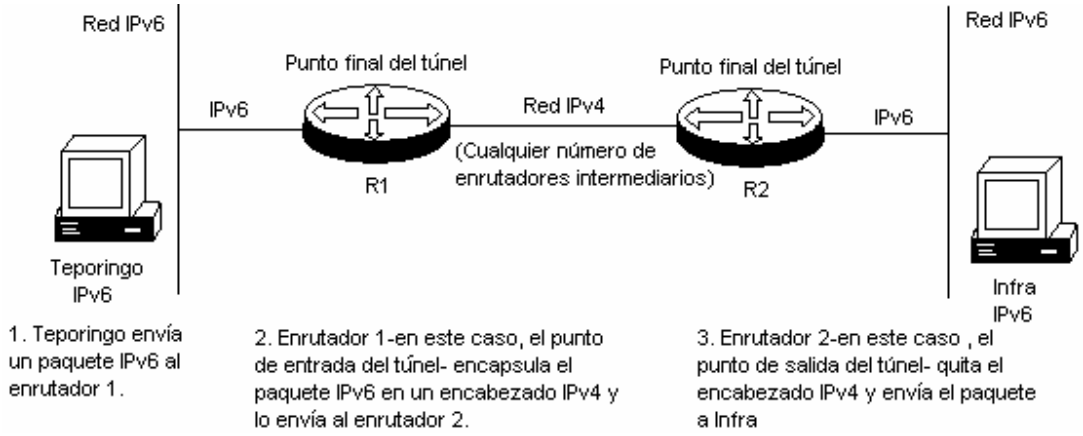


Figura 56. Encapsulación y túnel.

El Host Teporingo está en una red IPv6 y quiere enviar un paquete IPv6 al Host Infra que se encuentra en otra red IPv6. La red entre el enrutador R1 y R2 es una red solamente-IPv4. El enrutador R1 es el punto de entrada del túnel "tunnel entry point". Teporingo envía el paquete IPv6 al enrutador R1 (Paso 1 en la figura 56). Cuando el enrutador R1 recibe en paquete diseccionado hacia Infra, éste encapsula el paquete en un encabezado IPv4 y lo envía hacia el enrutador R2 (Paso 2 en la figura 56), el cual es el punto de salida del túnel "tunnel exit point". El enrutador R2 desencapsula el paquete y lo envía a su destino final (Paso 3 en la figura 56). Entre los enrutadores R1 y R2, puede haber cualquier número de enrutadores.

Un túnel tiene dos puntos finales: uno es el punto de entrada del túnel, y el otro, el punto de salida del túnel. En el ejemplo mostrado, los puntos finales del túnel son los dos enrutadores. Los túneles pueden ser implementados en formas diferentes, éstas pueden ser de enrutador-a-enrutador (router-to-router), de host-a-enrutador (host-to-router), de host-a-hots (host-to-host) y de enrutador-a-host (router-to-host). Dependiendo del escenario usado, la entrada del túnel y el punto de salida, puede ser cualquiera, un host o un enrutador. Si el punto de salida es un host, la dirección destino IPv6 del paquete original es idéntica al punto de salida del túnel y puede ser tomada del encabezado IPv6 del paquete original. Si el punto de salida del túnel es un enrutador, la dirección IPv6 destino del paquete original no es idéntica a la dirección del punto de salida del túnel. En este caso, el punto de entrada del túnel debe proveer la información de la dirección del punto de salida del túnel.

Los pasos para la Encapsulación del paquete IPv6 son los siguientes:

1. el punto de entrada del túnel decrementa en el encabezado IPv6 el campo de límite de salto en uno, encapsula el paquete IPv6 en un encabezado IPv4, y transmite el paquete encapsulado através del túnel. Si es necesario el paquete IPv4, es fragmentado.
2. el punto de salida del túnel recibe el paquete encapsulado. Si el paquete fue fragmentado, el punto de salida lo reensambla. Entonces el punto de salida remueve el encabezado IPv4 y procesa el paquete IPv6 a su original destino.

La figura 57 muestra la Encapsulación de un paquete IPv6 dentro de un paquete IPv4.

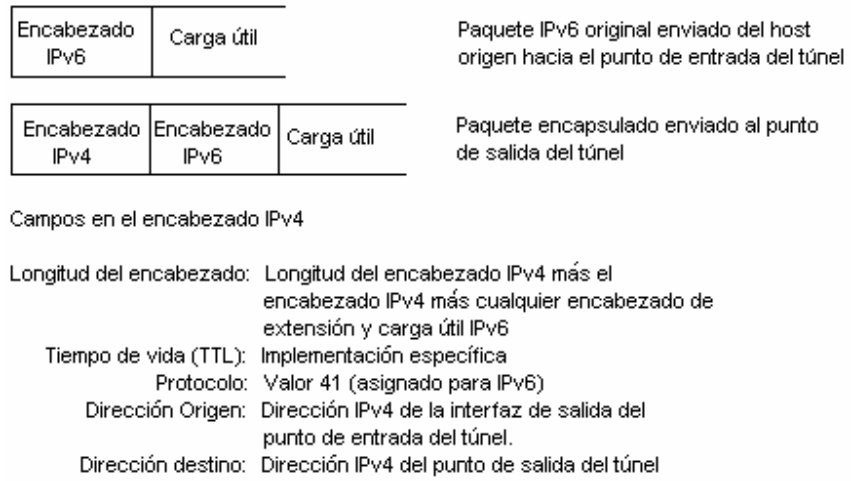


Figura 57. Encapsulación

Los siguientes campos en el encabezado IPv4 es interesante mencionarlos: el campo de longitud del encabezado contiene la longitud del encabezado IPv4, más la longitud del encabezado IPv6, más cualquier encabezado de extensión y la longitud de la carga útil IPv6. Si el paquete encapsulado ha sido fragmentado, habrá valores correspondientes en el campo de Indicadores "Flags" y en el campo de Desplazamiento de la fragmentación "Fragment offset". El valor en el campo de tiempo de vida (TTL) depende de la implementación usada. El número del protocolo es puesto a 41, es el valor asignado para IPv6. La dirección IPv4 origen es la dirección de la interfaz de salida del punto de entrada del túnel y la dirección destino IPv4, es la dirección IPv4 del punto de salida del túnel. El túnel de IPv6-sobre-IPv4 "IPv6-over-IPv4" es considerado un sólo salto. Esto oculta la existencia de un túnel para el usuario final y no es detectable por herramientas comúnes tales como traceroute.

3.4. TÚNELES IPv6 IN IPv4 (6in4)

Es importante hacer notar la diferencia entre las expresiones "6in4" y "6over4" que han venido usándose, como consecuencia documentos de vendedores, manuales de productos, publicaciones, papeles, libros, tutoriales y muchos otros utilizan estas expresiones. Como sea no todos estos documentos, incluyendo la IETF, se refieren actualmente a la misma

encapsulación IPv6/IPv4 o mecanismo de transición IPv4/IPv6. El resultado de esta confusión de terminología es un número de errores entre vendedores, operadores, ingenieros y usuarios cuando diseñan y documentan productos.

Los RFC´s 1853, 1933, 2893, y 2473 definen el mecanismo básico de transición IPv4/IPv6, incluyendo la encapsulación de paquetes IPv6 en paquetes IPv4 lo que significa el valor 41 en el campo del protocolo del encabezado IPv4, estos mecanismos de encapsulación son los que usan la terminología "in", el cual es utilizado en algunos mecanismos de transición.

El RFC2529, especifica un mecanismo de transición, el cual usa "6in4" (encapsulación de paquetes IPv6 en IPv4), creando un virtual enlace IPv6 sobre "over" una infraestructura multicast IPv4. Este mecanismo de transición es llamado como "6over4".

Claramente, "6in4" y "6over4" son cosas absolutamente diferentes, actualmente "6over4" es un mecanismo de transición el cual usa "6in4" como el procedimiento para encapsular paquetes IPv6 en infraestructuras multicast IPv4. El hecho de que son cosas diferentes y el requisito especial de infraestructura multicast IPv4 para "6over4", es lo que hace la diferencia entre ambos términos.[15]

A veces es difícil desarrollar una combinación completa de una red IPv6/IPv4, por ejemplo, porque los viejos enrutadores no tienen suficiente memoria para correr la ultima versión de su sistema operativo el cual incluye IPv6. En este contexto, parte de la red puede ser IPv6/IPv4 y otra parte puede ser solamente-IPv4.

3.4.1. Encapsulación Host a Enrutador.

La figura 58 muestra un ejemplo donde el nodo A es dual-stack y quiere enviar un datagrama IPv6 a un nodo B solamente-IPv6. El nodo A es configurado manualmente para enviar todos los datagramas IPv6 dentro un túnel IPv6-in-IPv4 al enrutador R1.

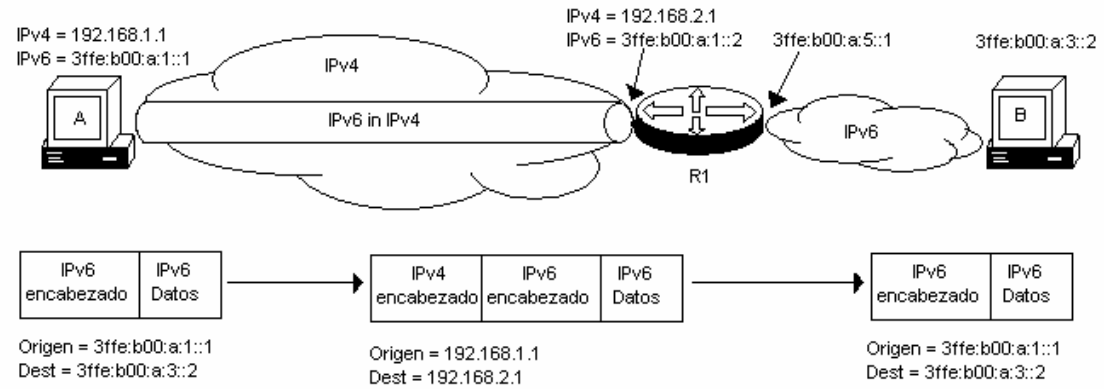


Figura 58. Túnel IPv6 in IPv4 de un host a un enrutador

El nodo origen A internamente hace un datagrama IPv6 para el nodo destino B, este datagrama esta aún dentro de la pila IP de A. La dirección origen del datagrama IPv6 es la

[15] J. Palet (2005). "6in4 versus 6over4 terminology". Internet-Draft. Pp. 3.

dirección de A (3ffe:b00:a:1::1) y la dirección destino es la dirección de B (3ffe:b00:a:3::2). Este datagrama IPv6 aún no es enviado, pero en lugar de esto es encapsulado dentro de la carga útil de un datagrama IPv4 en A, el nodo A ha sido cónfigurado para enviar todo el trafico IPv6 encapsulado al enrutador R1 el cual es el punto final del túnel.

El origen del datagrama IPv4 es la dirección IPv4 de A (192.168.1.1) y la dirección destino es la dirección IPv4 de R1 (192.168.2.1). La carga útil del datagrama IPv4 contiene el datagrama IPv6 como se ilustra en la figura 57.

El datagrama IPv4 viaja sobre la red IPv4, el enrutador R1 recibe el datagrama IPv4, remueve el encabezado IPv4 y obtiene el datagrama IPv6. R1 envía el datagrama IPv6 sobre la red IPv6 hacia el nodo destino B, la dirección destino del datagrama IPv6 desencapsulado es la dirección de B (3ffe:b00:a:3::2), y como último paso el nodo destino B recibe el datagrama IPv6, el nodo B no se da cuenta que el datagrama hubiera sido encapsulado en cualquier parte del camino.

3.4.2. Encapsulación Enrutador a Enrutador

La figura 59 muestra un ejemplo donde los nodos A y B son nodos IPv6 y el túnel es hecho por los enrutadores de frontera.

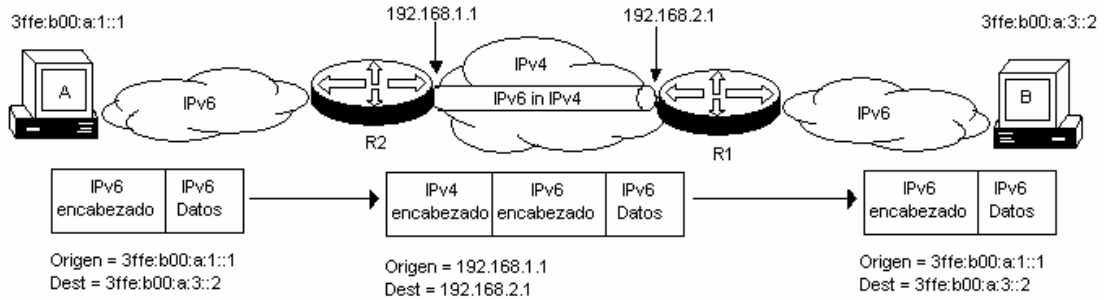


Figura 59. Túnel IPv6 in IPv4 entre dos enrutadores.

En este caso, el nodo origen A y el destino B se comportan como nodos normales IPv6 y no tiene configurado ningún túnel. El enrutador R2 es dual-stack y tiene configurado un túnel con el enrutador R1.

El nodo origen A hace un datagrama IPv6 con destino al nodo B, el nodo A envía el datagrama IPv6 sobre su red IPv6, la dirección origen es la dirección de A (3ffe:b00:a:1::1) y la dirección destino es la dirección de B (3ffe:b00:a:3::2). El datagrama es enviado al enrutador R2, el cual tiene una ruta hacia la red destino a través del túnel hacia R1. R2 encapsula el datagrama IPv6 en un datagrama IPv4 con la dirección origen IPv4 que pertenece al enrutador R2 (192.168.1.1) y la dirección destino IPv4 pertenece al otro punto del túnel que es el enrutador R1 (192.168.2.1), el datagrama IPv4 viaja sobre la red IPv4, el enrutador R1 recibe el datagrama IPv4, remueve el encabezado IPv4 y envía el paquete original IPv6 a la red IPv6 a la que pertenece el nodo destino B, por último el nodo B recibe el paquete IPv6. A y B no se dan cuenta que el datagrama hubiera sido encapsulado en cualquier parte del camino.

3.4.3. Túnel Estático

Configuración manual de la dirección destino y origen de IPv4 e IPv6 en ambos puntos finales del túnel es requerido para hacer que trabaje el túnel. Ambos puntos finales tienen que ser dual-stack.

La tabla 18 muestra los datos necesarios para la configuración de un túnel entre dos puntos finales, en este caso son el enrutador R2 y R1 de la figura 60.

Tabla 18. Configuración de los puntos finales de un túnel.

Parámetro de configuración	Enrutador R2	Enrutador R1
Dirección Origen IPv6	3ffe:b00:1:1::1	3ffe:b00:1:1::2
Dirección Destino IPv6	3ffe:b00:1:1::2	3ffe:b00:1:1::1
Dirección Origen IPv4	192.0.2.1	192.0.3.1
Dirección Destino IPv4	192.0.3.1	192.0.2.1

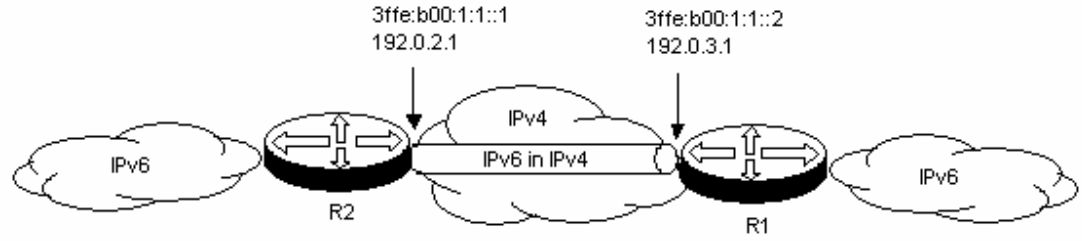


Figura 60. Túnel estático entre dos enrutadores.

Algunas implementaciones no necesitan la dirección destino IPv6 en la configuración, como los túneles son enlaces punto-a-punto no es necesario conocer la otra dirección del punto final. Lo que si es importante y esencial, es la dirección destino IPv4, ya que es la primera información requerida para enviar el datagrama encapsulado.

Lo que se requiere conocer para la configuración de los enrutadores es: el enrutador R1 y R2 deben ser dual-stack, el enrutador R1 tiene que alcanzar la dirección IPv4 del enrutador R2, y viceversa, después de la configuración de los dos puntos finales del túnel, rutas estáticas o dinámicas han de ser configuradas para que el tráfico IPv6 sea enviado a través del túnel. Usando como ejemplo la figura 60, si R1 esta conectado al Internet IPv6, entonces R2 tendrá una ruta por default estática señalando a la interfaz del túnel.

Las limitantes de los túneles manuales son la configuración tediosa para cada punto final del túnel, si muchos túneles tienen que ser hechos, y si las direcciones IP cambian, los túneles estáticos no pasan IPv4 NAT, y los paquetes de IPv6 através de IPv4; usando el protocolo 41 puede ser filtrado por seguridad de las puertas de enlace, en estos casos los túneles no trabajan. Frecuentemente los túneles configurados manualmente son utilizados cuando pocos túneles serán utilizados y si no hay presente IPv4 NAT.

3.5. TÚNELES IPv6 OVER IPv4 (6over4)

El mecanismo de 6over4 permite a hosts IPv6 aislados, localizados en un enlace físico que no esta conectado directamente a un enrutador IPv6, llegar a ser hosts IPv6 completamente funcionales, realizando tareas como Descubrimiento de Vecinos y auto configuración de direcciones Stateless, usando un dominio IPv4 multicast como su enlace local virtual. Así, al menos un enrutador IPv6 conectado nativamente por una de sus interfaces a una red externa IPv6, usando el mismo método (6over4) puede ser conectado al dominio IPv4 mediante su otra interfaz. Hosts IPv6 conectados usando este método no requieren direcciones IPv4-compatibles o túneles configurados. En esta forma IPv6 gana considerable independencia de los subyacentes enlaces y puede pasar sobre muchos saltos de subredes IPv4, es por tal razón que se llega a nombrar "virtual Ethernet". La figura 61 muestra un esquema de una red con 6over4.

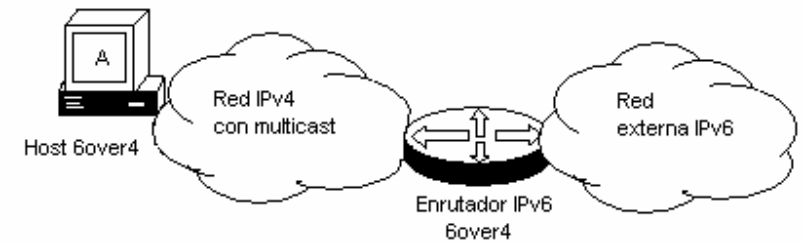


Figura 61. Red 6over4

Los paquetes IPv6 son transmitidos en paquetes IPv4 con un Tipo de protocolo IPv4 de 41, de la misma forma como ha sido asignado para paquetes IPv6 que son tuneleados dentro de tramas IPv4 (6in4). El encabezado IPv4 contiene la dirección destino y origen IPv4. El cuerpo del paquete IPv4 contiene el encabezado IPv6 seguido inmediatamente después por la carga útil. El campo de tiempo de vida debe ser puesto en un valor bajo, para prevenir que tales paquetes accidentalmente se escapen del dominio IPv4, se recomienda un valor de 8.

3.5.1. Concepto básico de 6over4

La idea básica del mecanismo de 6over4 se centra principalmente en el concepto de Ethernet, esto es, simular la capa de enlace Ethernet usando el ámbito de las direcciones IPv4 multicast. Las redes Ethernet se basan en conectar los hosts mediante un mismo enlace compartido, de la misma forma lo realiza 6over4, pero este mecanismo en vez de compartir el mismo enlace, usa el ámbito de las direcciones multicast, esto es, los hosts comparten el mismo ámbito de dominio de las direcciones multicast, de esta forma se logra que el host IPv6 aislado logre comunicarse con el enrutador IPv6 que soporta 6over4, y realice las funciones comunes de un hosts IPv6 que se encuentra conectado directamente con un enrutador IPv6.

El mecanismo de Neighbor Discovery que se usa entre los hosts IPv6 y el enrutador IPv6 que utilizan 6over4, se realiza de la misma manera que en una red Ethernet normal con IPv6 nativo. Es decir, cuando se tiene un paquete de Neighbor Discovery IPv6 que será enviado a un host específico, el paquete se encapsula en una trama Ethernet, con sus

direcciones MAC destino y origen correspondientes, de esta manera se envía al host destino; de la misma forma sucede con IPv6 con el mecanismo de 6over4, sólo que en vez de encapsularlo en una trama Ethernet con sus direcciones MAC correspondiente, el paquete es encapsulado en un paquete IPv4 con direcciones multicast.

La figura 62 muestra la comunicación entre un host IPv6 con 6over4 y un enrutador IPv6, utilizando Neighbor Discovery, unidos mediante una red IPv4 multicast, la cual se comporta como un enlace virtual Ethernet.

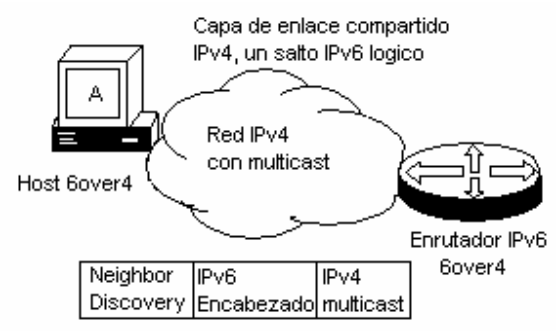


Figura 62. Neighbor Discovery en 6over4.

Como ya se ha mencionado la red entera IPv4 multicast es usada como un medio compartido de capa de enlace, y para que se soporte Neighbor Discovery y auto configuración de direcciones IPv6 Stateless, debe ser usado un sistema específico de direcciones IPv4 multicast, las cuales se muestran a continuación:

- Dirección Multicast Todos-los-nodos (All-nodes): **239.X.0.1** usada para alcanzar cada nodo en el dominio IPv4 soportando este mecanismo. Equivalente a una dirección broadcast Ethernet.
- Dirección Multicast Todos-los-enrutadores (All-routers): **239.X.0.2** usada para alcanzar cada enrutador en el dominio IPv4 soportando este mecanismo.
- Dirección Multicast de nodo-solicitado (solicited-node): **239.X.C.D** computada como una función de la dirección objetivo solicitada (solicited-target`s ardes) ver 2.4.5.3.2. Siendo C y D los bits de menor orden de la dirección IPv4.

Donde X es el identificador de ámbito local, normalmente se usa 192 según el RFC2365. Además de este tipo de direcciones también se establece el formato para las direcciones IPv6, para el identificador de interfaz se necesita que el tamaño sea de 64 bits, como el identificador de la interfaz IPv4 es de 32 bits, se rellena con ceros hasta alcanzar los 64 bits, para la dirección de enlace local se sigue manejando el prefijo FE80::/64, como se muestra en la figura 63, y para la auto configuración se agrega el identificador de interfaz antes formado.

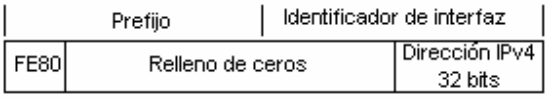


Figura 63. Dirección de enlace-local 6over4.

3.5.2. Operación 6over4

Para poder implementar una red con soporte IPv6 es necesario algunos requerimientos como: un enrutador con IPv6 habilitado y con soporte de 6over4, del cual una de sus interfaces debe ser conectado a una red externa IPv6, y otra de sus interfaces debe estar conectada al dominio de red IPv4 multicast, la cual se ubicará entre el enrutador y hosts, que servirá de enlace entre el host y el enrutador.
Los hosts con soporte 6over4 ubicados en el dominio IPv4 podrán comunicarse con el enrutador y con la red IPv6, mediante la construcción de túneles dinámicos.

Como primer paso para la puesta en operación de un host IPv6 6over4, éste obtiene su configuración usando el protocolo de Neighbor Discovery sobre paquetes IPv4 multicast, esta configuración que se obtiene es, la dirección de enlace-local, el prefijo, como se menciona en la sección 2.5.5.5, y la dirección IPv4 del enrutador con IPv6 habilitado. El enrutador IPv6 anuncia dos prefijos, uno para la LAN nativa y otro para el dominio de 6over4, como con cualquier prefijo IPv6 asignado a una subred IPv6, deben ser únicas dentro de su ámbito. Los prefijos de las direcciones de enlace-local deben ser /128. La figura 64 muestra la puesta en operación por primera vez de un host 6over4.

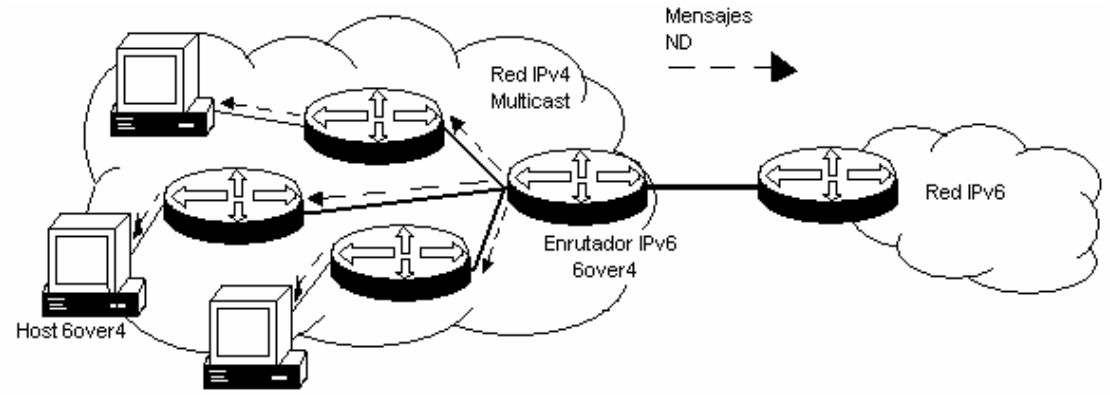


Figura 64. Inicio de un host en una red 6over4.

Una vez que se ha configurado el host con los anuncios de Neighbor Discovery y se ha descubierto la puerta de enlace (gateway), los hosts envían sus datagramas IPv6 encapsulados en paquetes IPv4 con el valor de 41 en el campo de Tipo de protocolo, de la misma forma que se haría con túneles automáticos 6to4, los hosts tienen la responsabilidad de crear los túneles automáticos através del dominio de red IPv4, como se muestra en la figura 65.

Las ventajas de este mecanismo son: los nodos pueden descubrir otros hosts IPv6 automáticamente, los hosts IPv6 no requieren direcciones IPv4 compatibles o túneles configurados, en este mecanismo se utilizan los túneles automáticos, elimina cualquier restricción de la capa de enlace, de esta forma los hosts pueden migar a otra parte de la red, y pueden ser esparcidos en toda la amplitud del dominio e incluso puede ser localizado a algunos saltos del enrutador IPv6.

Las desventajas que presenta son: el número de túneles soportado por los enrutadores 6over4 es limitado, el dominio IPv4 debe soportar multicast, cada red IPv6 necesita tener un enrutador de frontera IPv6.

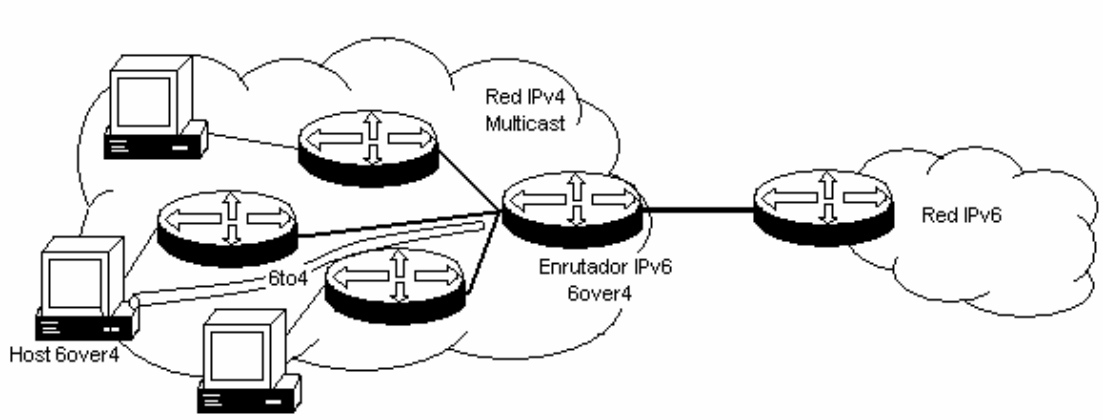


Figura 65. Túnel creado por un host sobre 6over4

6to4, ISATAP, TSP tunnel broker y Teredo son mecanismos automáticos para la creación de túneles.

3.6. 6to4

El mecanismo 6to4 construye túneles 6in4 sobre demanda y asigna un espacio de direcciones IPv6. El espacio de direcciones IPv6 para el mecanismo 6to4 es obtenido del prefijo reservado 2002::/16, seguido por 32 bits de la dirección IPv4 externa del enrutador de frontera del sitio, dándole al sitio un prefijo /48, como se muestra en la figura 66.

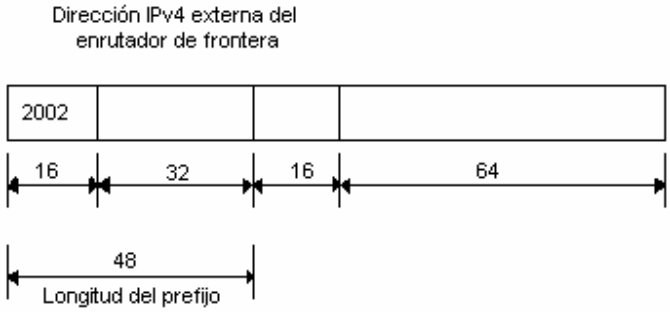


Figura 66. Estructura de las direcciones 6to4.

Este mecanismo es aplicado a sitios, más que ha host individuales, y se necesita por lo menos una dirección IPv4 global única, permite a sitios IPv6 aislados, unidos a una amplia área de red que no tiene soporte IPv6 nativo, comunicarse con otro dominio IPv6 con mínima configuración manual. Sitios IPv6 o hosts conectados usando este método, no requieren direcciones IPv6 IPv4-compatibles o túneles configurados.

La figura 67 muestra un ejemplo de un sitio IPv6 usando el mecanismo 6to4. El enrutador de frontera tiene una dirección IPv4 externa (192.0.2.1). El sitio IPv6 detrás del enrutador

de frontera usa la dirección IPv6 2002:c000:0201::/48 para numerar a toda su red. El espacio de direcciones es tomado de 2002:<dirección IPv4 externa en hexadecimal>::/48, donde la dirección IPv4 es la dirección IPv4 externa del enrutador de frontera (192.0.2.1), representado en hexadecimal como c000:0201. El mecanismo de 6to4 necesita solamente ser implementado en el enrutador de frontera. Los hosts dentro del sitio no necesitan soportar 6to4.

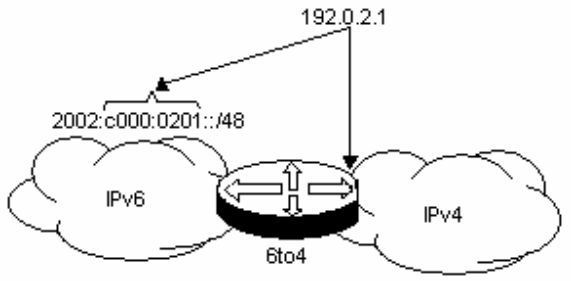


Figura 67. Espacio de direcciones 6to4 de un sitio basados en la dirección IPv4 externa del enrutador de frontera.

3.6.1. Túnel de enrutador a enrutador

La figura 68 muestra un ejemplo del proceso de un túnel automático 6to4. El host A (2002:c000:201:1::1) envía un paquete al host B (2002:c000:301:2::2).

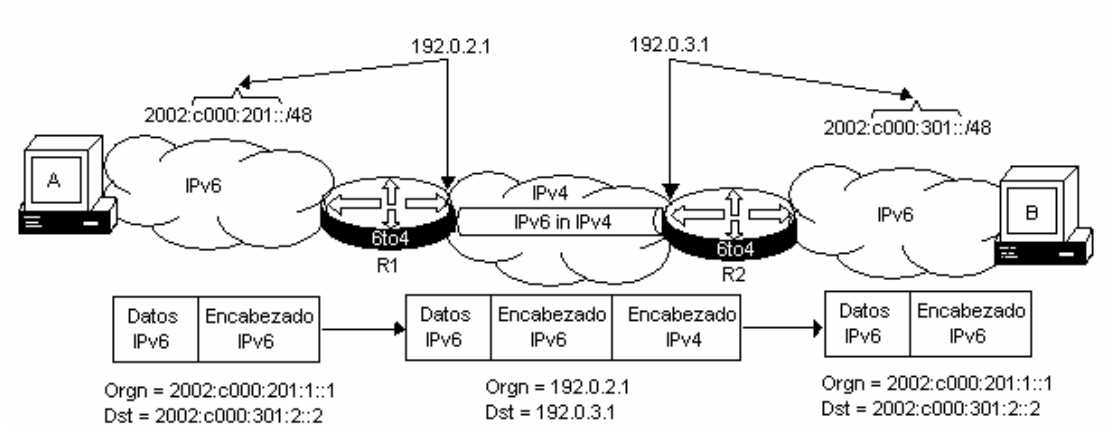


Figura 68. Túnel 6to4 de enrutador a enrutador

El Host A envía un datagrama IPv6 al host B que tienen la dirección 2002:c000:301:2::2. Cuando el enrutador 6to4 frontera R1recibe el datagrama IPv6 con la dirección de destino 2002:c000:301:2::2, la cual empieza con 2002, el enrutador extrae la dirección IPv4 en los 32 bits seguidos después de 2002 en la dirección destino (c000:0301 => 192.0.3.1), entonces el enrutador envía el paquete IPv6 encapsulado en un paquete IPv4 hacia el enrutador R2, quien tiene la dirección IPv4 192.0.3.1. La dirección origen del paquete encapsulado es la dirección del enrutador R1 (192.0.2.1). De esta forma el enrutador R2 recibe el paquete IPv6 encapsulado en un paquete IPv4, éste lo desencapsula y lo envía al host B. B para contestarle al host A, hace la misma operación, pero en sentido contrario del trayecto.

3.6.2. Túnel 6to4 de host a enrutador

Similarmente, un sólo host puede usar su dirección IPv4 y crear el prefijo 6to4 /48 para su propio uso. La figura 69 muestra el mismo proceso explicado anteriormente, pero aquí el host está habilitado 6to4 y puede encapsular el tráfico IPv6 dentro de paquetes IPv4 y enviarlos hacia el enrutador.

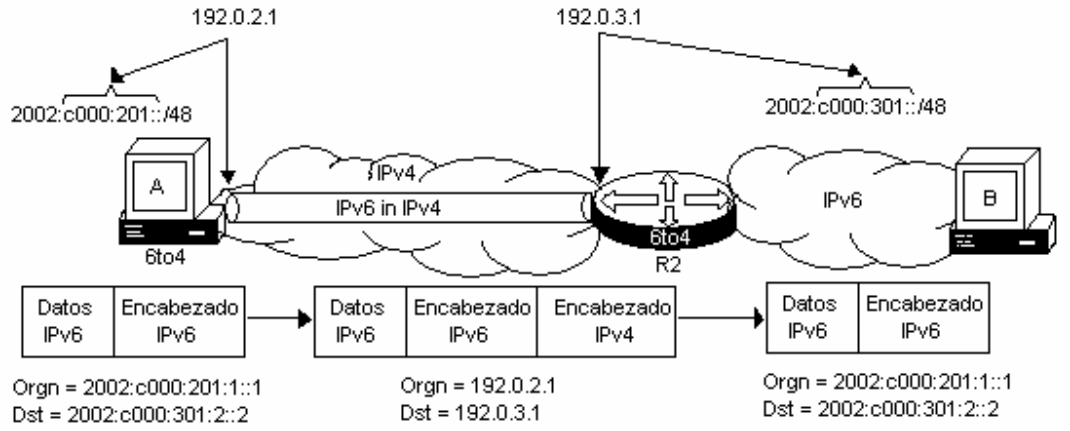


Figura 69. Túnel 6to4 de host a enrutador

El host A envía un datagrama IPv6 hacia el host B quien tienen la dirección 2002:c000:301:2::2. El host A tiene habilitado 6to4. Cuando la pila interna procesa el datagrama IPv6 con la dirección destino 2002:c000:301:2::2, la cual inicia con 2002, el host extrae la dirección IPv4 en los 32 bits siguientes de 2002 de la dirección destino (c000:0301 => 192.0.3.1), entonces éste envía el paquete IPv6 encapsulado en un paquete IPv4 hacia el enrutador R2 quien tienen la dirección IPv4 192.0.3.1. La dirección IPv4 origen del paquete encapsulado es la dirección IPv4 del host A (192.0.2.1). Así el enrutador R2 recibe el paquete encapsulado, lo desencapsula y envía esta hacia B. B para contestarle al host A hace la misma operación pero en sentido contrario del trayecto.

3.6.3. Túnel 6to4 de host a host

Un escenario de un host a un host trabaja de la misma forma, donde los dos host tienen 6to4 habilitado, como se muestra en la figura 70. Ambos host encapsulan y desencapsulan los paquetes antes de enviarlos a la red.

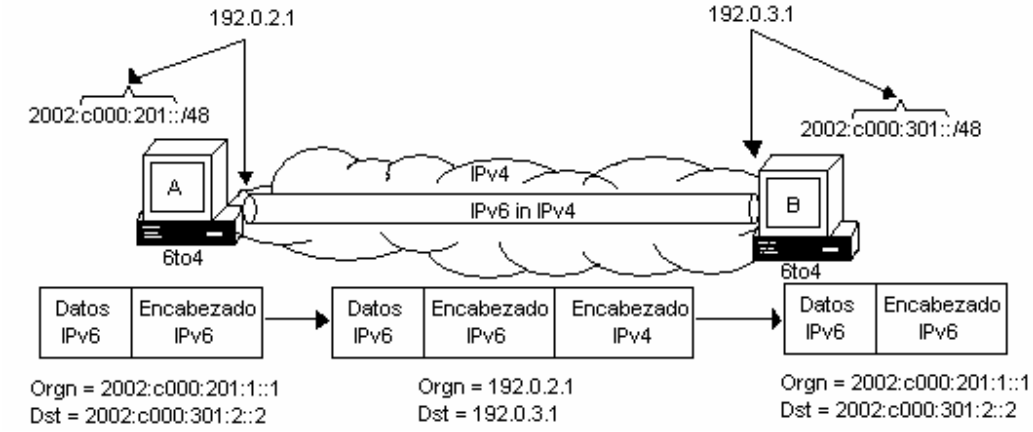


Figura 70. Túnel 6to4 de host a host

3.6.4. Relay 6to4

La figura 71 muestra la comunicación entre dos sitios 6to4.

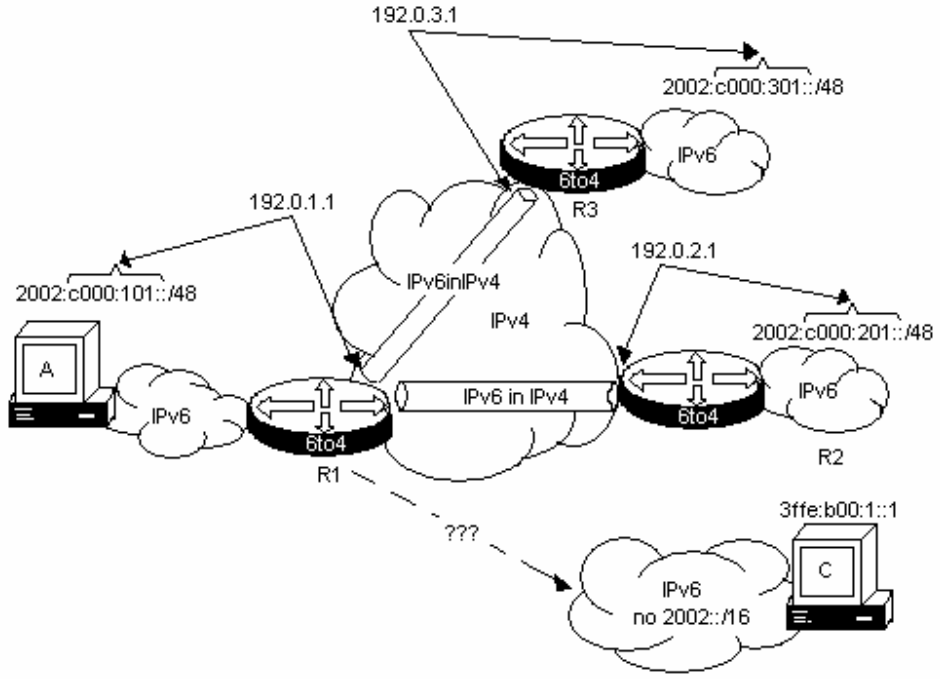


Figura 71. Sitio 6to4 sin relay

Si un host A envía un paquete a C con direcciones que no sean 6to4 tales como 3ffe:b00:1::1, el enrutador R1, enrutador de frontera del sitio A, no conoce la ruta del paquete para la dirección destino ya que no es una dirección 6to4. El enrutador R1 necesita entonces un relay 6to4 para el Internet IPv6 con direcciones que no sean 6to4.

Un relay 6to4 es un enrutador de frontera 6to4 que tiene conectividad al resto de la red IPv6. Es usado como tránsito para otros sitios 6to4 para alcanzar redes IPv6 con direcciones que no sean 6to4.

En la figura 72, A envía un paquete a C, el enrutador R3 es el relay 6to4 para el enrutador R1 y esta conectado al Internet IPv6 con direcciones que no son 6to4. Cuando R1 quiere enviar un paquete con una dirección destino que no es 6to4 (3ffe:b00:1::1), éste no puede construir un túnel automático a algún otro enrutador 6to4, ya que no puede extraer la dirección IPv4 de la dirección IPv6 destino. Si el enrutador R1 tiene como enrutador por default al enrutador R3 vía el mecanismo 6to4., entonces R1 encapsula el paquete IPv6 hacia R3 y R3 desencapsula el paquete y lo envía a la red IPv6. R3 y el enlace entre el enrutador R3 y el Internet IPv6 que no usa 6to4, son usados como una red de tránsito. De esta forma, los sitios 6to4 deben usar a R3 como tránsito. Para habilitar el relay 6to4, es simplemente un enrutador 6to4 con un enrutador por default al Internet IPv6. Un sitio 6to4 que esta usando un relay 6to4 instala en el enrutador de frontera 6to4, una ruta por default IPv6 señalando la dirección 6to4 del relay. Los enrutadores relay 6to4 no requieren características específicas para actuar como un relay 6to4, sólo una entrada de ruta estática.

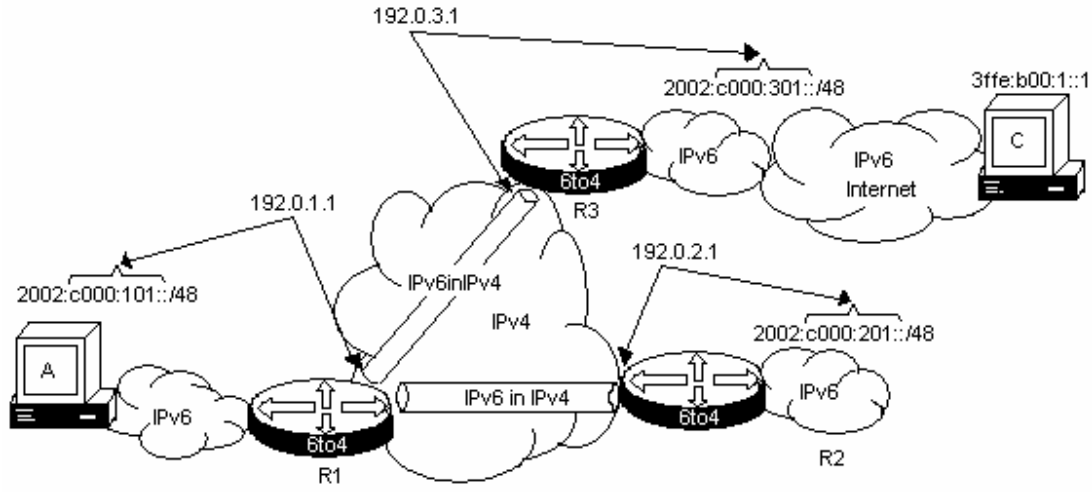


Figura 72. Relay 6to4

3.6.5. Requerimientos y limitaciones del mecanismo 6to4

Los requerimientos para desarrollar una red con el mecanismo de 6to4 son:

- Los enrutadores de frontera (R1, R2) del sitio deben ser dual-satck
- Los enrutadores de frontera (R1, R2) del sitio deben soportar 6to4
- Todos los sitios 6to4 son alcanzables através de la dirección IPv4 del enrutador de frontera.
- Todos los sitios 6to4 tienen un relay 6to4, configurado estáticamente en el enrutador de frontera del sitio, para transitar tráfico IPv6 a direcciones que no sean 6to4.
- Los Hosts del sitio no necesitan soportar o conocer acerca de 6to4. Sólo el enrutador de frontera debe conocer 6to4.

El mecanismo 6to4 incrusta las direcciones IPv4 dentro del prefijo IPv6 del sitio. Así todos los nodos del sitio 6to4 tienen la dirección IPv4 del enrutador de frontera del sitio incrustada en su dirección IPv6, en la parte del prefijo.

El incrustar direcciones IPv4 dentro de la dirección IPv6 introduce algunas limitaciones como:

La dirección IPv4 externa del enrutador de frontera es usado para definir el prefijo /48 del sitio, en el supuesto de que el enrutador de frontera cambie su dirección IPv4, entonces todo el sitio tendrá que renumerarse; esto es, todos los nodos dentro del sitio tendrán que cambiar su dirección IPv6, hacer cambios en el DNS, y además cambios en la red que afecten la renumeración. Cambiar la dirección IP de todos los nodos dentro de un sitio es bastante laborioso. Aún si tienes el protocolo de renumeración IPv6, esto es inconveniente y costoso.

La dirección IPv4 externa del enrutador de frontera es usada dentro de la dirección IPv6. Todo el tráfico proveniente del sitio 6to4, debe pasar através del enrutador que tiene la dirección IPv4 incrustada en la dirección IPv6. Aún si el sitio tiene múltiples conexiones al Internet IPv4, todo el tráfico encapsulado IPv6 siempre pasará a través del mismo enrutador de frontera, entonces si el enrutador se muere por alguna circunstancia, todo el sitio muere también, de aquí que no sea posible la redundancia, ya que la dirección IPv6 tiene a la dirección IPv4 externa.

Si todos los usuarios de la actual Internet IPv4 usan 6to4 para tener conectividad, entonces el espacio completo de direcciones IPv4 (2^{32} direcciones) se insertarán dentro de la tabla de enrutamiento del sitio IPv6.

Cuando un nodo dual stack combinado con una red IPv4/IPv6 tienen soporte IPv6, y quiere enviar un paquete con una dirección destino 6to4, ¿qué es lo que pasará?, ¿éste enviará el paquete al enrutador 6to4?, o ¿éste mismo hará un túnel automático 6to4?. Por default, el nodo intentará hacer el túnel automático 6to4, lo cual resultará dentro del sitio, que este paquete será ruteado através de enrutamiento IPv4 en vez de enrutamiento IPv6.

Finalmente, el mecanismo 6to4 no puede atravesar NAT´s.

3.7. ISATAP

ISATAP (Intra-Site Automatic Tunnel Addressing Protocol) es un mecanismo para automatizar la creación de túneles de nodos a enrutadores y de nodos a nodos dentro de un sitio. Nodos dual Stack IPv6/IPv4 usan ISATAP para crear túneles automáticos de paquetes IPv6 en IPv4, los nodos ISATAP ven a la red IPv4 como la capa de enlace para IPV6 y ven a otros nodos en la red, como potenciales host o enrutadores IPv6 según sea el caso. Este mecanismo incrusta la dirección IPv4 del nodo en los últimos 32 bits en la parte del identificador de interfaz, sin importar que la dirección sea global o privada. La figura 73 muestra el formato de una dirección ISATAP. Los primeros 32 bits del identificador de interfaz son “00:00:5E:EF”, reservado por el IANA para ISATAP, y es el que define el identificador de interfaz ISATAP.

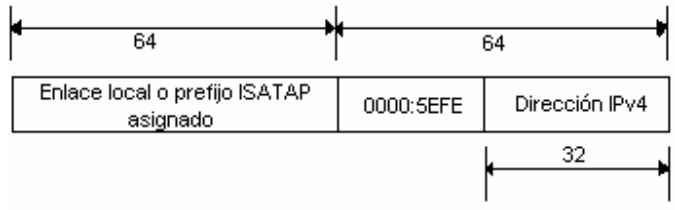


Figura 73. Formato de dirección ISATAP

3.7.1. Túnel ISATAP

ISATAP crea un enlace virtual sobre un sitio completo IPv4. Un prefijo /64 del prefijo /48 del sitio es dedicado para el enlace ISATAP. Éste será usado por los nodos como su prefijo /64, como se muestra en la figura 73. La dirección de enlace local (fe80::/64) es también usada sobre el enlace virtual ISATAP.

La figura 74 muestra un ejemplo de una red ISATAP. El host A, B y el enrutador R1 son dual Stack y tienen habilitado la implementación de ISATAP. El host C es IPv6 sin ISATAP. El administrador de la red usa el prefijo 3ffe:b00:1::/48 para su sitio. Éste asigna el prefijo 3ffe:b00:1:2::/64 para el enlace virtual ISATAP sobre la red IPv4.

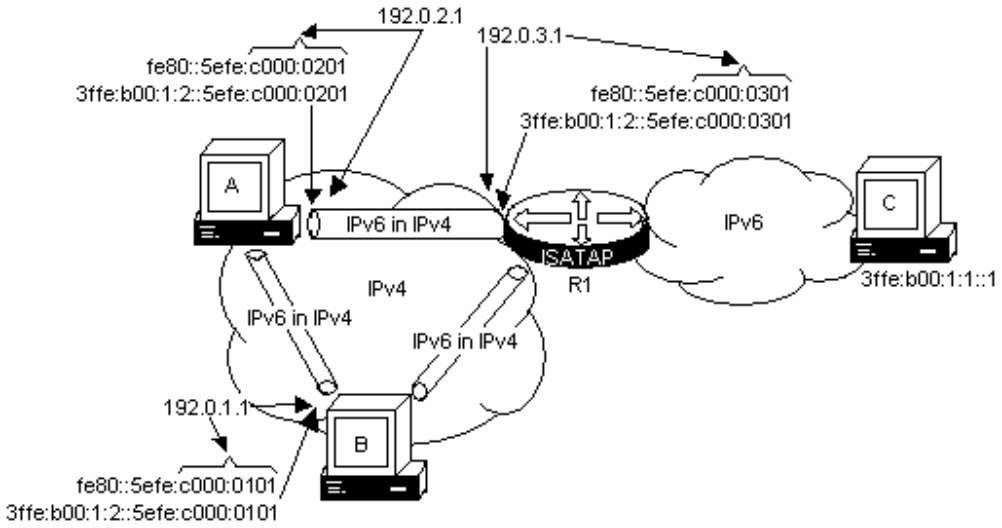


Figura 74. Red sobre ISATAP

El host A tiene la dirección IPv4 192.0.2.1 y crea su dirección de enlace local basándose en el formato ISATAP: fe80::5efe:c000:0201, usando los 32 bits ISATAP del identificador de interfaz (0000:5efe) y la representación hexadecimal (c000:0201) de la dirección IPv4 del host (192.0.2.1). El host B y el enrutador R1 hacen lo mismo respectivamente, cada uno construye su dirección de enlace local como lo muestra la figura 75.

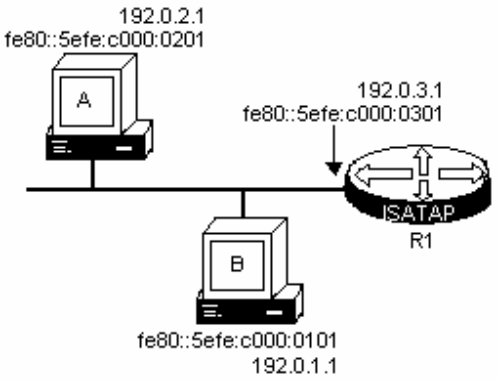


Figura 75. Enlace virtual ISATAP con direcciones de enlace local.

El host A envía un mensaje de solicitud de vecino para la dirección ISATAP al enrutador ISATAP (R1) configurado estáticamente en el nodo. Recibe entonces un anuncio de enrutador del enrutador ISATAP R1 sobre el enlace virtual ISATAP, el host A entonces configura su dirección global como 3ffe:b00:1:2::5efe:c000:0201, del prefijo recibido 3ffe:b00:1:2::/64 del enrutador R1 del enlace virtual. Los anuncios de enrutador también incluyen que R1 es identificado así mismo como el enrutador por default, anunciando su dirección ISATAP como el siguiente salto. El host B hace lo mismo respectivamente, levanta el enlace virtual con la dirección global y un enrutador por default, para alcanzar la red IPv6 dentro del sitio, como se muestra en la figura 76.

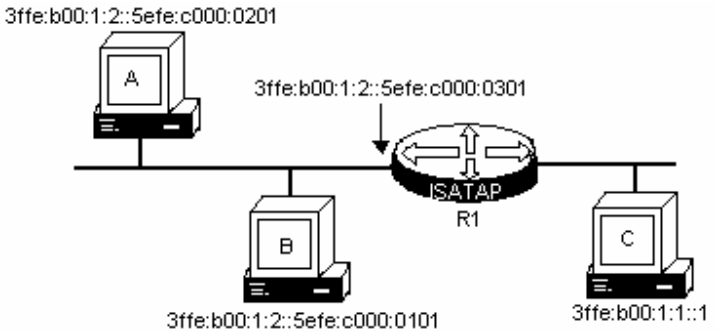


Figura 76. Enlace virtual ISATAP con direcciones globales.

Cuando A envía un paquete a B, éste ve a B en el mismo enlace, entonces A envía un mensaje de solicitud de vecinos para la dirección B. B responde con un mensaje de anuncio de vecino. Estos mensajes IPv6 son encapsulados dentro de paquetes IPv4 unicast.

¿Qué pasa cuando el nodo A envía su primer paquete IPv6 al nodo B ISATAP?, usaremos como referencia la figura 74 para ejemplificar este proceso. A quiere enviar un paquete hacia B. B tiene la dirección IPv6 (3ffe:b00:1:2::5efe:c000:0201) que es el mismo prefijo del enlace (3ffe:b00:1:2::/64) a la cual está conectado A (3ffe:b00:1:2::5efe:c000:101) por medio de su interfaz virtual ISATAP. Así A envía un mensaje de solicitud de vecino sobre la interfaz virtual ISATAP hacia B. La tabla 19 describe el proceso.

Tabla 19. Comunicación ISATAP de nodo a nodo

De → Hacia	Tipo de paquete	Dirección origen y destino	Descripción
1 A → B	Solicitud de vecino IPv6 sobre IPv4	IPv4 Orig: 192.0.2.1 IPv4 Dst: 192.0.1.1 IPv6 Orig: 3ffe:b00:1:2::5efe:c000:201 IPv6 Dst: ff02::1:ff00:101	El mensaje de solicitud de vecino es encapsulado dentro de un paquete IPv4. La dirección destino IPv4 es tomada de la dirección IPv6 destino. La dirección IPv6 destino es la dirección multicast de nodo-solicitado.
2 B → A	Anuncio de vecino IPv6 sobre IPv4	IPv4 Orig: 192.0.1.1 IPv4 Dst: 192.0.2.1 IPv6 Orig: 3ffe:b00:1:2::5efe:c000:101 IPv6 Dst: 3ffe:b00:1:2::5efe:c000:201	El mensaje de anuncio de vecino es encapsulado en IPv4. Regresa la confirmación de 3ffe:b000:1:2::5efe:c000:101 con la dirección de capa de enlace de 192.0.1.1
3 A → B	Paquete IPv6 sobre IPv4	IPv4 Orig: 192.0.2.1 IPv4 Dst: 192.0.1.1 IPv6 Orig: 3ffe:b00:1:2::5efe:c000:201 IPv6 Dst: 3ffe:b00:1:2::5efe:c000:101	Envio de un paquete IPv6 encapsulado.
4 B → A	Paquete IPv6 sobre IPv4	IPv4 Orig: 192.0.1.1 IPv4 Dst: 192.0.2.1 IPv6 Orig: 3ffe:b00:1:2::5efe:c000:101 IPv6 Dst: 3ffe:b00:1:2::5efe:c000:201	Contestación del paquete IPv6.

Cuando el nodo A ISATAP envía por primera vez un paquete al nodo C sin ISATAP, ¿qué es lo que sucede?, usaremos como referencia la figura 73 para ejemplificar este proceso. A quiere enviar un paquete a C. La dirección IPv6 de C (3ffe:b00:1:1::1) no está en el mismo enlace del prefijo (3ffe:b00:1:2::/64) de A (3ffe:b00:1:2::5efe:c000:101) sobre la interfaz virtual ISATAP de A. De esta manera, A necesita enviar el paquete hacia C através del enrutador por default R1 (previamente configurado estáticamente en A). Así A envía un mensaje de solicitud de vecino sobre la interfaz virtual ISATAP para el enrutador R1. La tabla 20 describe el proceso.

Tabla 20. Comunicación de un nodo ISATAP a un nodo no ISATAP.

De → Hacia	Tipo de paquete	Dirección origen y destino	Descripción
1 A → R1	Solicitud de vecino IPv6 sobre IPv4	IPv4 Orig: 192.0.2.1 IPv4 Dst: 192.0.3.1 IPv6 Orig: 3ffe:b00:1:2::5efe:c000:201 IPv6 Dst: ff02::1:ff00:301	La solicitud de vecino es encapsulado en IPv4. La dirección destino IPv4 es la del enrutador R1, quien es el enrutador por default ISATAP, su dirección fue previamente configurada estáticamente en A.
2 R1 → A	Anuncio de vecino IPv6 sobre IPv4	IPv4 Orig: 192.0.3.1 IPv4 Dst: 192.0.2.1 IPv6 Orig:	Los anuncios de vecinos son encapsulados en IPv4.

		3ffe:b00:1:2::5efe:c000:301 IPv6 Dst: 3ffe:b00:1:2::5efe:c000:201	Regresa confirmación de fe80::5efe:c000:0301 con la dirección de capa de enlace 192.0.3.1
3 A → C Através de R1	Paquete IPv6 sobre IPv4	IPv4 Orig: 192.0.2.1 IPv4 Dst: 192.0.3.1 IPv6 Orig: 3ffe:b00:1:2::5efe:c000:201 IPv6 Dst: 3ffe:b00:1:1::1	Paquetes IPv6 se envían encapsulados en IPv4 hacia R1, mientras C es el destino final IPv6.
4 R1 → C	Paquete IPv6 nativo	IPv6 Orig: 3ffe:b00:1:2::5efe:c000:201 IPv6 Dst: 3ffe:b00:1:1::1	R1 recibe los paquetes encapsulados IPv6 en IPv4, los desencapsula y los envía nativamente a C.
4 C → A	Contestación en paquete IPv6	IPv6 Orig: 3ffe:b00:1:1::1 IPv6 Dst: 3ffe:b00:1:2::5efe:c000:201	

3.7.2. Requerimientos y limitaciones para el mecanismo ISATAP

Para implementar ISATAP en una red, se requiere:

- Nodos IPv6 en el sitio que sean dual Stack
- Nodos IPv6 que tengan implementado ISATAP
- Un enrutador dual Stack con ISATAP y que esté disponible sobre la red IPv4 para alcanzar el resto de IPv6 en el sitio
- El enrutador por default ISATAP es configurado estáticamente en todos los nodos ISATAP
- ISATAP es usado dentro de un sitio, o dentro de un dominio administrativo

El mecanismo ISATAP crea un enlace virtual sobre una red potencialmente amplia IPv4. Cualquier paquete como Broadcast, tal como enviar ff01::1, en el enlace creará un gran número de paquetes en la red. Y otros de los factores que afecta al mecanismo ISATAP es que no puede atravesar NAT´s, porque utiliza Encapsulación IPv6 in IPv4, usando en el campo de protocolo del encabezado IPv4 el número 41.

3.8. ENCAPSULACIÓN IPv6 EN UDP IPv4.

Un túnel IPv6 en IPv4 requiere que ambos puntos finales del túnel sean alcanzables por su dirección IPv4. Si uno o muchos NAT´s se encuentran en el camino entre los dos puntos finales, entonces el túnel no trabaja. Esto es porque la dirección IP del punto final es trasladada como se muestra la figura 77.

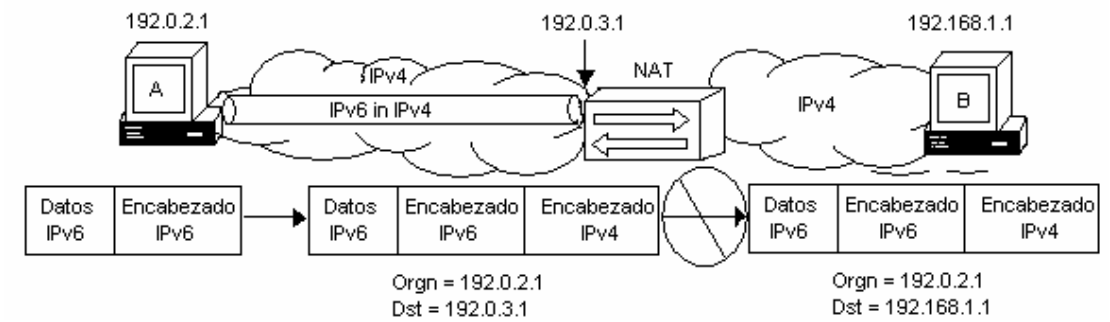


Figura 77. Túnel con un NAT intermedio

Una pasarela NAT cambia la dirección origen en cada paquete de salida y, dependiendo del método, también el puerto origen para que sea único. Estas traducciones de dirección se almacenan en una tabla, para recordar qué dirección y puerto le corresponde a cada dispositivo cliente y así saber donde deben regresar los paquetes de respuesta. Si un paquete que intenta ingresar a la red interna no existe en la tabla de traducciones, entonces es descartado. Esto se muestra en la figura 77, en donde el nodo A conoce al nodo B por la dirección externa del NAT (192.0.3.1), cuando el nodo A envía un paquete encapsulado hacia B, este paquete alcanza al NAT pero éste no es enviado hacia B, porque la dirección destino no corresponde al del nodo B.

Es posible evitar esto y habilitar un túnel cuando sólo un NAT se encuentra en el trayecto, sólo cuando este NAT se encuentra bajo el control administrativo del punto final del túnel. La primera forma sería configurar el NAT estáticamente para enviar cualquier paquete IPv4 recibido con el valor de 41 en el campo de protocolo de su encabezado, hacia la dirección interna IPv4 del punto final del túnel, como se muestra en la figura 78.

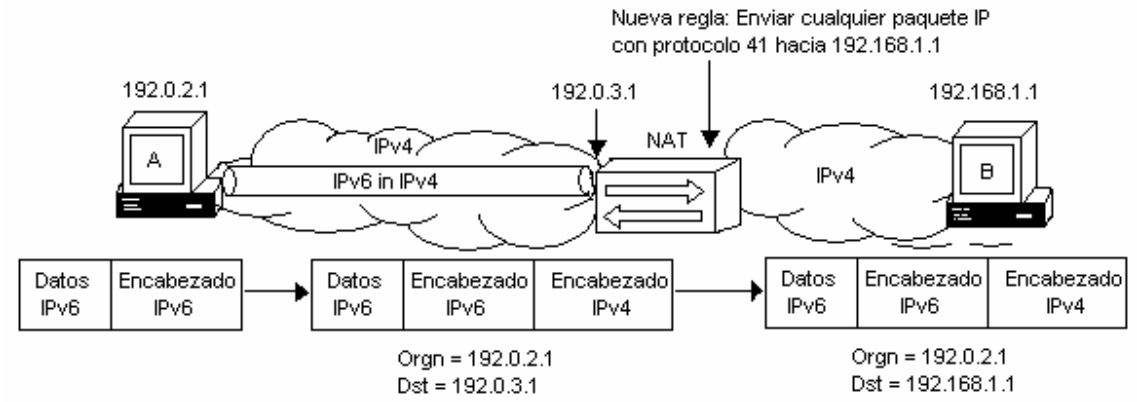


Figura 78. NAT enviando paquetes IPv4 con protocolo 41.

Como se puede observar en la figura 78 se agregó una regla al NAT para permitir el paso de cualquier paquete IPv4 con protocolo 41 (contiene paquete IPv6) y que sea enviado hacia 192.168.1.1. Sólo un nodo puede recibir detrás del NAT el paquete, limitando esta técnica a un nodo por NAT. Si múltiples NAT se encuentran en el trayecto esto es difícil de hacer ya que hay que agregar regla por NAT.

Otra forma de poder habilitar un túnel através de un NAT, es asignar una dirección secundaria a la interfaz externa del NAT para que ésta sea mapeada con la dirección interna del punto final del túnel. Cualquier tráfico que sea recibido a la dirección externa secundaria, será enviado a la dirección interna del punto final del túnel como se muestra en la figura 79. Como sea el caso, direcciones secundarias frecuentemente no son disponibles.

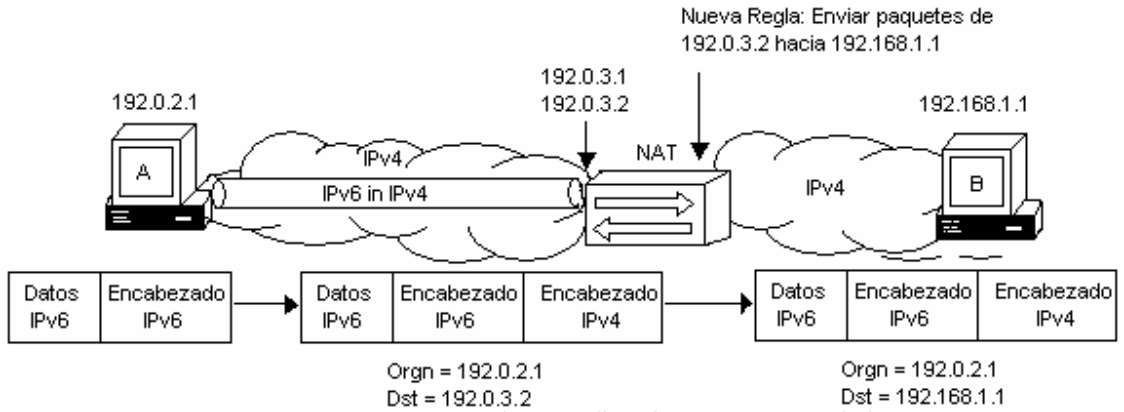


Figura 79. NAT mapeando a una dirección externa secundaria.

Estas dos técnicas obviamente no son factibles, ya que sólo un túnel es posible por dirección externa IPv4 del NAT, se requiere control administrativo de los NAT´s, configuración estática por cada uno y soporte para esta capacidad de envio. Para cruzar NAT´s, muchas soluciones usan STUN[16] para aplicaciones IPv4, ésta se basa en la encapsulación de paquetes IP en UDP. Esta misma técnica se usa para túneles 6in4 con la presencia de NAT en el trayecto. Esto consiste en mover hacia arriba de la pila el paquete IPv6 y así encapsularla en paquetes UDP de transporte, sobre paquetes IPv4, como se muestra en la figura 80.

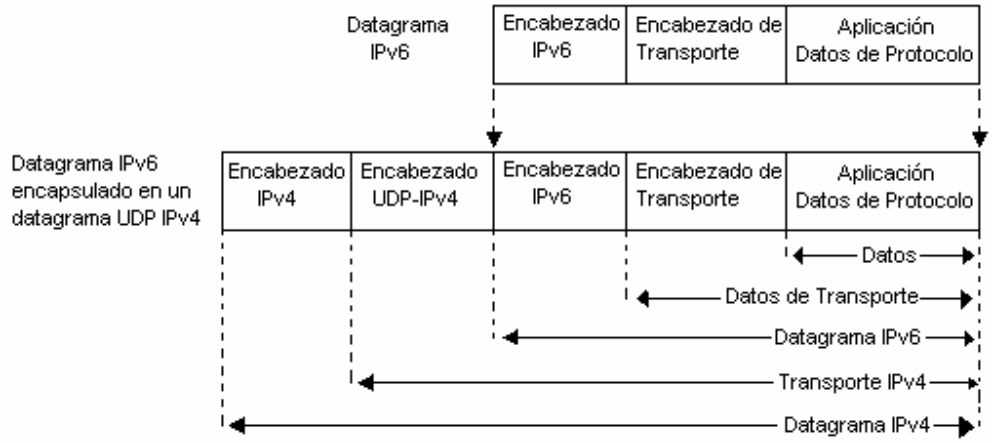


Figura 80. Encapsulación de un datagrama IPv6 en un datagrama UDP-IPv4.

El protocolo IP es no orientado a conexión como UDP, el cual hace a UDP el mejor candidato para transportar la encapsulación.

[16] J. Rosenberg, J. Weinberger, C. Huitema, R. Mahy (2003). RFC3489 "STUN - Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs). Pp. 1

3.9. TUNNEL SETUP PROTOCOL (TSP) TUNNEL BROKER

El Internet IPv6 global hoy en día usa muchos túneles sobre la existente infraestructura IPv4. Es difícil configurar y mantenerlos en un gran ambiente escalable. Se puede hacer, pero el proceso es muy complejo para usuarios finales, quienes ya tienen una conexión IPv4 y les gustaría entrar al mundo IPv6. TSP (Tunnel Setup Protocol) es diseñado para automatizar el proceso de establecer túneles IPv6 in IPv4 para proveer acceso transparente a IPv6 en todas partes. Este es un protocolo de señalización entre el cliente y el broker, donde el cliente hace la petición de un túnel y el broker envía la información relacionada con el túnel asignado. Ambas partes configuran su respectivo punto final del túnel y éste queda establecido. El broker puede ser visto como un virtual ISP, que provee conectividad IPv6 a usuarios que ya están conectados al Internet IPv4. TSP es un versión mejorada del modelo de tunnel broker (RFC3053). TSP usa mensajes básicos XML sobre TCP o UDP. El uso de XML brinda extensibilidad y fácil opción de procesamiento.

3.9.1. Arquitectura básica

La figura 81 muestra los componentes básicos de la arquitectura TSP tunnel broker.
Un nodo N1 tiene el software cliente TSP y se conecta al Tunnel Broker TB usando TCP o UDP sobre el bien conocido puerto TSP (3653). Después de la autenticación, el cliente hace la petición de un túnel, entonces el broker asigna y configura el nuevo túnel en el servidor de túnel TS1. El broker contesta la petición del túnel de N1, enviándole información acerca del túnel. El cliente TSP en el nodo N1 configura su punto final del túnel como un túnel estático, pero manejado por TSP. El túnel es establecido y los paquetes IPv6 enviados del nodo N1, son encapsulados en IPv4 hacia el servidor de túnel TS1.

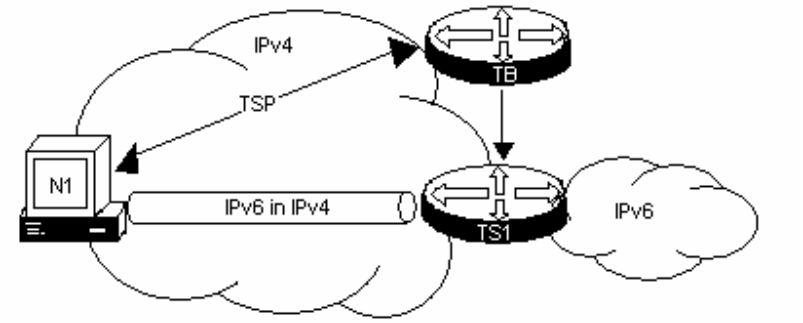


Figura 81. Tunnel Broker TSP

El Tunnel Broker (TB) y el servidor (TS) pueden ser combinados en el mismo hardware, como se muestra en la figura 82. Pero nosotros la manejaremos de forma separada.

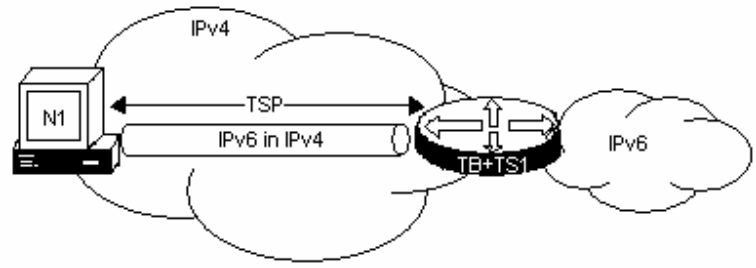


Figura 82. Tunnel Broker y servidor combinado

Con base en una implementación, se puede decidir combinar los dos en un miso hardware. Como sea, para soportar la característica de atravesar NAT, para todos los tipos de NAT, el tunnel broker y el servidor de túnel deben estar en la misma interfaz, compartiendo la misma dirección IPv4.

El Tunnel Broker es el lugar donde los usuarios se conectan para registrarse y activar túneles. El TB maneja la creación, modificación y eliminación de túneles por los usuarios. Por razones de escalabilidad, el TB puede compartir la carga de los puntos finales de los túneles entre algunos servidores de túneles. El TB envía las configuraciones de los túneles al servidor de túnel en donde necesite ser configurado, modificado o eliminado.

Un TS es un enrutador dual-stack conectado el Internet global. Éste recibe la orden del TB para configurar, crear, modificar y eliminar del lado del servidor de cada túnel.

3.9.2. Señalización TSP

El túnel es levantado entre dos puntos finales con el intercambio TSP. Cuando un túnel cliente quiere establecer un túnel con un tunnel broker, el cliente TSP se conecta al servidor TSP sobre TCP en el puerto 3653 sobre el tunnel broker, se autentifica con el broker, pide el túnel y obtiene la información del túnel en respuesta del broker. Ambos finalizan la conexión TSP y entonces configuran en su lado el punto final del túnel. El túnel es entonces establecido. La figura 83 muestra el proceso de un TSP cliente haciendo la petición de un túnel.

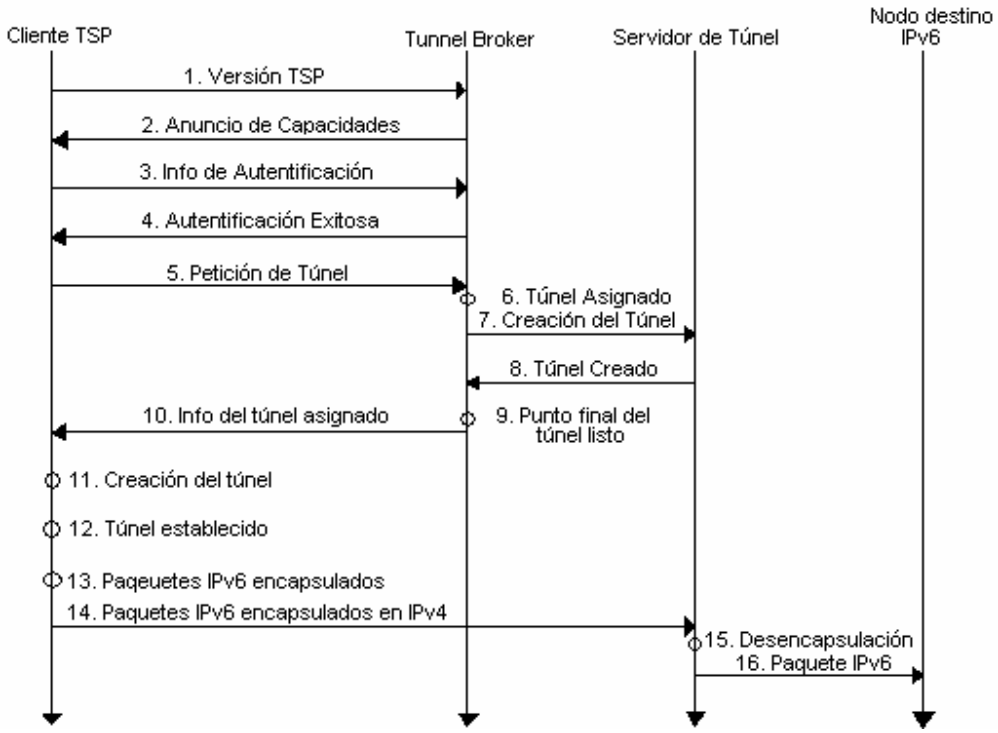


Figura 83. Flujo TSP

Tabla 21. Detalle de la señalización TSP

Paso	Descripción
1	El cliente TSP se conecta al servidor TSP por el puerto 3653 usando TSP sobre IPv4 del tunnel broker. La dirección IPv4 del tunnel broker es configurado en el cliente TSP. Cuando la conexión TCP es establecida, el cliente envía la versión del protocolo TSP el cual es usado por el broker. La declaración de la versión
2	El túnel broker verifica si soporta la version recibida del protocolo TSP y entonces envía una lista de sus capacidades, basándose en su configuración. Por ejemplo, el broker anuncia al cliente el mecanismo de autenticación soportado modos de Encapsulación de túneles, delegación de prefijo y modos delegación de DNS.
3	Basado en las capacidades recibidas y la configuración del cliente TSP, el cliente TSP envía su información de autenticación, por ejemplo envía su username y password.
4	El broker verifica la información de autenticación y envía un mensaje de status de autenticación exitosa hacia el cliente.
5	El cliente envía una petición de túnel al servidor, en un objeto en formato XML, como se muestra en la sección 3.10.4. La petición contiene la dirección IPv4 del cliente, y podría contener la petición para un prefijo o delegación DNS.
6	Para un nuevo túnel, el broker asigna direcciones IPv6 de un pool de direcciones IPv6 y puede asignar un prefijo IPv6 si es pedido.
7	El broker configura el túnel asignado en su servidor de túneles.
8	El servidor de túneles avisa al broker que el túnel es creado.
9	El punto final del túnel del lado broker/servidor es creado y la información del túnel es permanentemente guardada con información del usuario.
10	La información del túnel, formado en un objeto XML, es enviada al cliente TSP. La información contiene las direcciones IPv4 e IPv6 de ambos puntos finales del túnel y puede contener al prefijo asignado IPv6 e información DNS. Éste también contiene el tiempo de vida (lifetime) del túnel, el cual indica al cliente cuándo debe renovar el túnel antes de que éste expire en el broker/servidor.
11	El cliente TSP crea su punto final del túnel basándose en la información del túnel

	recibido del broker y de los pasos previos.
12	El túnel es establecido entre los dos puntos finales
13	El cliente entonces puede empezare a enviar paquetes IPv6 sobre el túnel. De esta forma, un paquete IPv6 es encapsulado dentro de un paquete IPv4 con la dirección destino del servidor de túneles.
14	Los paquetes IPv6 in IPv4 son enviados hacia el servidor de túneles
15	El servidor de túneles recibe el paquete, lo desencapsula y lo envía nativamente sobre la red IPv6 al nodo destino IPv6.
16	El nodo destino IPv6 recibe el paquete.

La tabla 21 describe cada paso de la figura 83 en detalle, de esta forma se describe en 12 pasos, por lo cual el levantamiento se lleva a cabo con el intercambio de unos cuantos paquetes.

3.9.3. Capacidades del Tunnel Broker TSP

En el paso 2 de la figura 83, el broker envía todas sus capacidades. La tabla 22 muestra la lista de algunas capacidades definidas. Un Broker puede anunciar algunas o todas de estas capacidades, dependiendo de su configuración.

La capacidad de encapsulación de túnel del tunnel broker TSP, tal como IPv6 in IPv4, IPv6 in UDP-IPv4 y IPv4 in IPv6, habilita a un cliente para establecer conectividad en todos estos contextos, haciendo la solución versátil. Por ejemplo, el cliente podría no conocer si se encuentra más allá de un NAT. Con las capacidades del protocolo TSP y los subsecuentes procesos que se mencionarán, el broker está habilitado para encontrar si el cliente se encuentra mas allá de un NAT y propondrá un túnel IPv6 in UDP-IPv4 para el cliente. Sin embargo, en casos donde el cliente está sobre una red solamente-IPv6 (IPv6-only) y necesita conectividad IPv4, la negociación TSP maneja esta situación automáticamente. La negociación TSP permite la adaptación del contexto para establecer la conectividad con el otro protocolo IP, con el cual el cliente no es aun conectado. Cuando el cliente TSP es un nodo móvil, el servicio permite al cliente conectarse con ambas redes IP, si su punto actual de conexión coincide con uno o con otro protocolo disponible.

Tabla 22. Capacidades del túnel broker TSP.

Palabra clave de la capacidad	Descripción
TUNNEL = V6V4	Este broker ofrece túneles IPv6 in IPv4
TUNNEL = V6UDPV4	Este broker ofrece tuneles IPv6 in UDP in IPv4
TUNNEL = V4V6	Este broker ofrece túneles IPv4 in IPv6
AUTH = ANONYMOUS	Este broker soporta autenticación anónima
AUTH = PLAIN	Este broker soporta autenticación de texto plano, donde el username y el password son transferidos limpiamente, sin ninguna encriptación
AUTH = DIGEST-MD5	Este broker soporta el mecanismo de autenticación digest-md5 donde el password no es enviado limpiamente

3.9.4. Mensajes XML para petición de túnel TSP

En el paso 5 de la figura 83, el cliente envía una petición de túnel en formato XML. Una petición para un túnel IPv6 in IPv4 es formado como se muestra a continuación:

```
<tunnel action=”create” type=”v6v4” >
 <client>
   <address type=”ipv4”>192.0.2.1</address>
 </client>
</tunnel>
```

El cliente pide un túnel (<tunnel>) a ser creado (action=”create”) usando encapsulación IPv6 in IPv4 (type=”v6v4”), anunciando su dirección IPv4 (<address type=”ipv4”>192.0.2.1</address>).
El broker responde con la información del túnel (paso 10 de la figura 83), confirmando la creación del túnel, formado en XML como sigue:

```
<tunnel action=”info” type=”v6v4” lifetime=”1440”>
 <server>
  <address type=”ipv4”>192.0.1.1</address>
  <address type=”ipv6”>3ffe:b00:0:1::</address>
 </server>
 <client>
  <address type=”ipv4”>192.0.2.1</address>
  <address type=”ipv6”>3ffe:b00:0:1::1</address>
 </client>
</tunnel>
```

Las direcciones IPv6 de ambos puntos finales y la dirección IPv4 del punto final del servidor de túneles (192.0.1.1) son enviadas al cliente. El cliente tiene toda la información necesaria para configurar su propio punto final del túnel.

3.9.5. Tiempo de vida del túnel (lifetime)

En el ejemplo anterior, el broker incluye en su respuesta el tiempo de vida del túnel (1440 segundos), el cual le dice al cliente cuándo expira el túnel. Es responsabilidad del cliente reconectarse al broker antes de que el tiempo de vida expire para mantener el túnel arriba. Una implementación típica de un cliente TSP usualmente duerme en el periodo de tiempo de vida y se levanta antes de que el tiempo de vida expire para reconectarse con el broker. El proceso de renovación del túnel es el mismo procedimiento explicado anteriormente, si el túnel aún está arriba en ambos lados del túnel, entonces el broker y el cliente no modifican nada en sus respectivos lados.

3.9.6. Cliente TSP detrás de un NAT

Como se ha mencionado anteriormente, túneles IPv6 in IPv4 no trabajan si un NAT se encuentra en el camino entre los dos puntos finales del túnel. Para atravesar NAT´s, el cliente TSP y el broker manejan la sesión TSP sobre UDP IPv4, como se muestra en la figura 84.

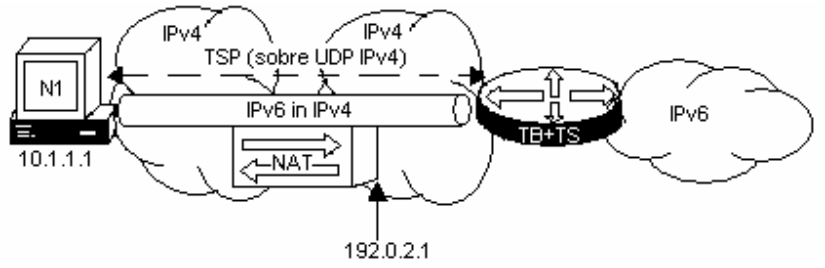


Figura 84. Sesión TSP sobre UDP IPv4

Cuando la sesión TSP es terminada, el túnel es establecido sobre el mismo canal UDP, asegurándose que el mapeo del puerto en el NAT es preservado. El tráfico IPv6 es encapsulado en UDP IPv4 y cruza el NAT a través del mismo puerto mapeado.
El broker TSP descubre cuando un cliente se encuentra atrás de un NAT, porque la dirección IPv4 que viene dentro de la petición de túnel TSP XML, es diferente de la dirección IPv4 origen del paquete TSP, como se muestra en la figura 85.

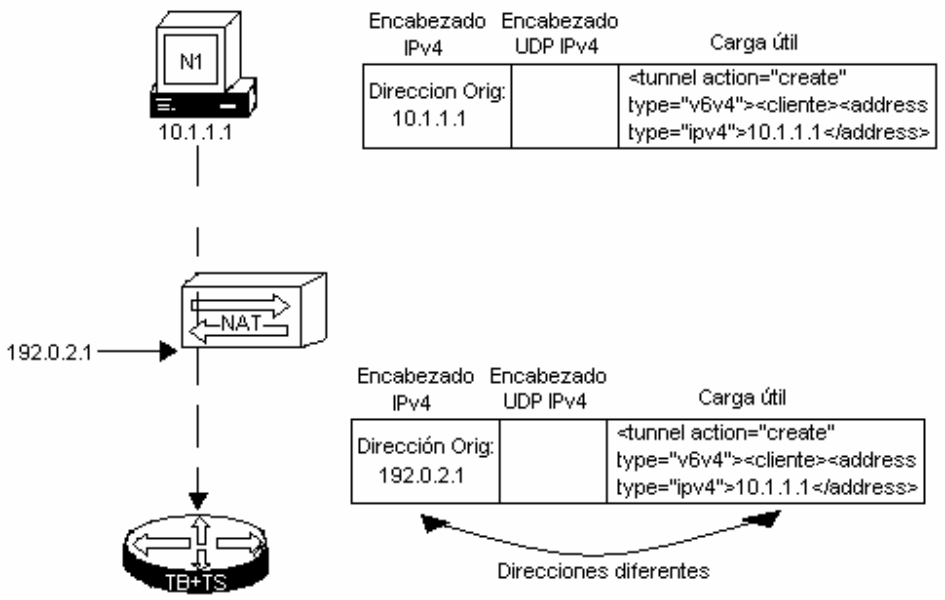


Figura 85. Detección de un NAT por TSP.

Cuando el TSP del cliente N1 ensambla en su paquete TSP su dirección IPv4 10.1.1.1, como la dirección origen del paquete en la petición de túnel TSP XML. Entonces el NAT, al recibir el paquete IPv4, cambia la dirección origen del host por la dirección del NAT 192.0.2.1, pero éste no cambia la dirección origen en la petición de túnel TSP XML.
Cuando un NAT es detectado, el broker ofrece al cliente un túnel IPv6 in UDP IPv4, usando la palabra clave “v6udpv4”, como se muestra:

```
<tunnel action=”create” type=”v6v4” >
 <client>
   <address type=”ipv4”>10.1.1.1</address>
 </client>
</tunnel>
```

El cliente pide un túnel (<tunnel>) a ser creado (action=”create”) usando encapsulación IPv6 in IPv4 (type=”v6v4”), anunciando su dirección IPv4 (<address type=”ipv4”>10.1.1.1</address>). El broker detecta que el cliente está atrás de un NAT, y éste mejor ofrece encapsulación IPv6 in UDP IPv4.

```
<tunnel action=”info” type=”v6udpv4” lifetime=”1440”>
 <server>
  <address type=”ipv4”>192.0.1.1</address>
  <address type=”ipv6”>3ffe:b00:0:1::</address>
 </server>
 <client>
  <address type=”ipv4”>10.1.1.1</address>
  <address type=”ipv6”>3ffe:b00:0:1::1</address>
 </client>
</tunnel>
```

Este túnel IPv6 in UDP IPv4 tomará lugar sobre el mismo canal UDP por el cual fue establecido la conexión TSP.

3.9.7. Registro del punto final del túnel del cliente TSP en un DNS

El cliente TSP podría tener registrada su dirección de punto final del túnel en el DNS por el broker. El broker confirma el registro, usando (<address type=”dn”> en la declaración del informe de la contestación, como se muestra:

```
<tunnel action=”info” type=”v6udpv4” lifetime=”1440”>
 <server>
  <address type=”ipv4”>192.0.1.1</address>
  <address type=”ipv6”>3ffe:b00:0:1::</address>
 </server>
 <client>
  <address type=”ipv4”>10.1.1.1</address>
  <address type=”ipv6”>3ffe:b00:0:1::1</address>
  < address type=”dn”>client1.freenet6.net</address>
 </client>
</tunnel>
```

En el ejemplo el punto final del túnel 3ffe:b00:0:1::1 es conocido en el DNS por client1.freenet6.net usando un registro AAAA.

3.9.8. El cliente TSP es un enrutador y pide un prefijo

Cuando el cliente TSP es un enrutador con una red detrás, éste necesitará un prefijo IPv6 para usar en su red IPv6, la figura 86 muestra tal situación.

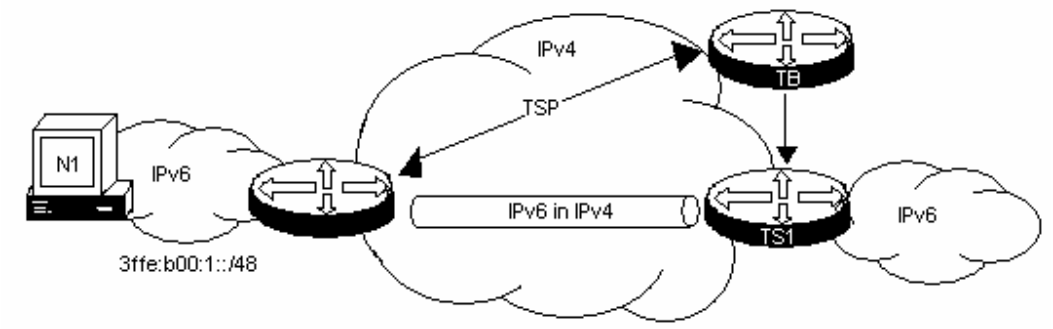


Figura 86. Cliente TSP como enrutador.

En este caso, el cliente TSP incluye en su petición del túnel el (<router>), declarando también la longitud del prefijo solicitado (<prefix>). El siguiente ejemplo muestra la petición de un túnel con un prefijo IPv6 /48.

```
<tunnel action=”create” type=”v6v4” >
 <client>
  <address type=”ipv4”>192.0.2.1</address>
  <router>
   <prefix lenght=”48”/>
  </router>
 </client>
</tunnel>
```

El túnel broker entonces asigna un prefijo de su pool de prefijos, y responde con la información del túnel y el prefijo. A continuación se muestra la contestación de la petición de un túnel y de un prefijo descrito arriba, donde el prefijo asignado es el 3ffe:b00:1::/48.

```
<tunnel action=”info” type=”v6v4” lifetime=”1440”>
 <server>
  <address type=”ipv4”>192.0.1.1</address>
  <address type=”ipv6”>3ffe:b00:0:1::</address>
 </server>
 <client>
  <address type=”ipv4”>192.0.2.1</address>
  <address type=”ipv6”>3ffe:b00:0:1::1</address>
  <router>
   <prefix lenght=”48”>3ffe:b00:1::</prefix>
  </router>
</client>
</tunnel>
```

El enrutador cliente TSP entonces usa el prefijo /48 para numerar su red.

3.9.9. Requerimientos y limitaciones

Los requerimientos para desarrollar la solución del tunnel broker TSP son:

- El nodo es dual-stack
- El nodo implementa el protocolo TSP como un cliente TSP y soporta túneles configurados.
- El túnel broker TSP implementa el protocolo TSP
- El túnel broker TSP tiene acceso a un servidor de túneles
- El servidor de túnel es dual-satck, soporta túneles configurados y está conectado a la red IPv6. Un servidor de túneles puede ser implementado dentro del mismo software del túnel broker.
- Hosts que se encuentran atrás de un enrutador TSP no necesitan soportar o conocer TSP, sólo tener habilitado IPv6

Si la distancia de la red entre el cliente TSP y el servidor de túneles es muy larga, entonces el tiempo del viaje de ida y vuelta del paquete puede ser alta. Esta limitación, también aplica a 6to4 relay. El servidor de túneles más cercano puede ser desarrollado usando una dirección IPv4 anycast como con 6to4 relay.
La técnica usada para atravesar NAT en la solución del túnel broker TSP, asegura que para todos los tipos de NAT´s que el túnel sea establecido. Para cruzar el NAT, el túnel broker y el servidor de túneles deben ser localizados, y el canal TSP abierto sobre UDP para el broker será usado por ambos, para TSP y el tráfico IPv6. Ésta localización hace más difícil repartir la carga entre múltiples servidores.

3.10. TEREDO

Teredo es una tecnología de túneles automáticos y de asignaciones de direcciones que provee conectividad IPv6 a hosts que se encuentran atrás de un NAT en una red IPv4. Un nodo implementa el cliente teredo, recibe una dirección IPv6 del servidor teredo y atraviesa el NAT IPv4 usando encapsulacion IPv6 sobre UDP-IPv4. La infraestructura teredo consiste de los siguientes componentes, como se muestra en la figura 87.

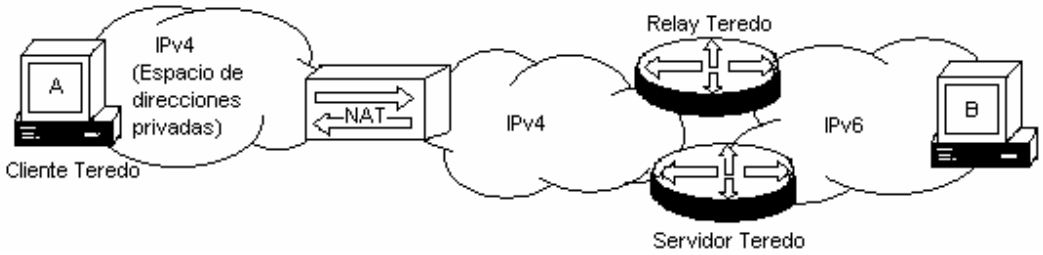


Figura 87. Componentes Teredo

- Cliente Teredo: es un nodo IPv6/IPv4 que soporta teredo. Un teredo cliente se comunica con un servidor teredo para obtener una dirección IPv6 teredo, para comunicarse con otros clientes teredos o hosts en el Internet IPv6.
- Servidor Teredo: Un servidor teredo es un nodo IPv6/IPv4 que está conectado a ambos Internet IPv4 e Ipv6. El rol general del servidor teredo es asistir en la configuración de la dirección del cliente teredo y facilitar la comunicación inicial entre clientes teredo o entre clientes teredo y hosts IPv6-only. El servidor teredo escucha tráfico sobre el puerto UDP 3544.

- Relay Teredo: Un teredo relay es un enrutador IPv6/IPv4 que puede enviar paquetes entre clientes teredo sobre el Internet IPv4 (usando una interface de túnel teredo) y hosts IPv6-only. En algunos casos, el teredo relay interactúa con el servidor teredo para ayudar y facilitar la comunicación inicial entre clientes teredos y hosts IPv6-only. El teredo relay escucha tráfico sobre el puerto UDP 3544.

3.10.1. Dirección Teredo

Como con 6to4, Teredo usa un prefijo especial para proveer direcciones IPv6 a los nodos. El IANA ha asignado el prefijo 2001:0000::/32 para este mecanismo. El formato de una dirección Teredo es mostrado en la figura 88, con los componentes que la conforman.

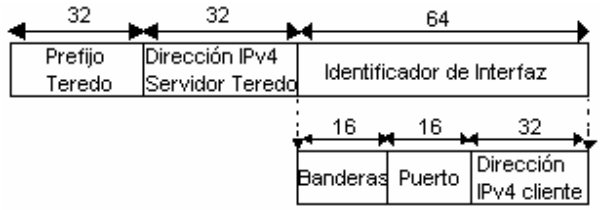


Figura 88. Formato de la dirección Teredo.

Tabla 23. Bit Cone del campo banderas de la dirección Teredo

Valor	Descripción
0x0000	El tipo de NAT no es un Cone
0x8000	El tipo de NAT es un Cone

- Prefijo asignado por el IANA para Teredo (32 bits) 2001:0000::/32
- Dirección IPv4 del servidor Teredo (32 bits)
- Banderas (16 bits) descritas en la tabla 23
- Número del puerto externo del NAT que mapea al cliente (16 bits)
- Dirección IPv4 externa del NAT que mapea al cliente (32 bits)

Cada componente de una dirección Teredo es usada para establecer el túnel. La dirección IPv4 del servidor Teredo contiene la dirección IPv4 pública del servidor Teredo, que ayuda al host para configurar su dirección IPv6 Teredo. El campo de banderas es usado para identificar el tipo de NAT, el bit de mayor orden es el único definido y es conocido como la bandera cone (flag cone), la tabla 23 muestra los dos posibles valores que puede tomar este campo.

Los campos número de puerto, y dirección IPv4 externos del NAT que mapean al cliente, tienen un formato diferente, ambas son enturbecidas (obfuscated), esto es invirtiendo cada bit en la dirección y puerto aplicándoles una operación OR exclusiva de 16 bits para el número de puerto con el valor hexadecimal 0xFFFF, y de 32 bits para la dirección IPv4 con el valor hexadecimal 0xFFFFFFFF. Esto se realiza porque algunas implementaciones NAT miran dentro de los paquetes IPv4 para encontrar direcciones IPv4 y cambiarlas a la dirección externa mapeada, de esta forma se evita.

Por ejemplo la versión turbia del puerto externo 5000 en formato hexadecimal es EC77 (5000 = 0x1388, 0x1388 XOR 0xFFFF = 0xEC77), y la versión turbia de la dirección IPv4 pública 131.107.0.1 en formato hexadecimal con dos punto es 7C94:FFFE (131.107.0.1 = 0x836B0001, 0x836B0001 XOR 0xFFFFFFFF = 0x7C94FFFE).

3.10.2. Mecanismo del Túnel

La base de Teredo es realizar la optimización del camino cuando ciertos tipos de NAT se encuentran en el camino. El primer paso para que un cliente establezca un túnel, es encontrar el tipo de NAT y obtener una dirección IPv6.

3.10.3. Encontrando el tipo de NAT y obteniendo una dirección.

Existen varios tipos de NAT y cada uno se comporta de manera diferente, los tipos son los siguientes:

- Cone NAT: Todos los paquetes de la misma dirección y mismo puerto internos son mapeadas a la misma dirección y mismo puerto externo. Cualquier host externo puede mandar un paquete al host interno mandándolo a la dirección y el puerto externo que ha sido mapeado. Se conoce también como "one-to-one NAT". (NAT uno a uno).
- Restricted NAT: Todos los paquetes de la misma dirección y mismo puerto internos son mapeadas a la misma dirección y mismo puerto externo. En este caso, en contraposición con cone NAT, un host externo (con IP x.x.x.x) sólo puede mandar un paquete al host interno si previamente el host interno le había enviado un paquete a la dirección IP x.x.x.x.
- Symmetric NAT: es NAT donde todas las peticiones de la misma IP y puerto interno con destino a otra IP y su correspondiente puerto, son mapeadas en el enrutador con la misma IP y puerto. Si el mismo host interno manda un paquete con la misma dirección interna y puerto a un destino diferente, se usará un mapeo diferente. Sólo el host externo que recibe un paquete puede mandar un paquete UDP de vuelta al host interno.

Para distinguir entre los tipos de NAT, un mapeo debe ser provocado por un paquete de salida hacia una dirección objetivo y una respuesta usando otra dirección debe ser regresada. Si la respuesta alcanza al nodo interno, entonces el NAT es un cone. Si la respuesta de la otra dirección es descartada por el NAT, entonces este es un restrictivo, puerto restrictivo o Symetric NAT. Cuando un NAT cone es presente, el cliente Teredo puede hacer algunas optimizaciones de trayecto.

Para descubrir el tipo de NAT, el cliente Teredo primero realiza una serie de pruebas con el Servidor Teredo. El cliente Teredo debe ser preconfigurado con las dos direcciones IPv4 del servidor Teredo. El protocolo Teredo usa versiones modificadas de mensajes IPv6 de Solicitud de Enrutador (RS) y Anuncio de Enrutador (RA), llamados Teredo RS (TRS) y Teredo RA (TRA), sobre UDP-IPv4.

Através del intercambio de múltiples paquetes, este proceso logra dos metas para el cliente Teredo: identificar el tipo de NAT y obtener una dirección IPv6. Los mensajes RA enviados por el servidor Teredo contiene el prefijo IPv6 para el cliente Teredo. El prefijo IPv6 /64 es formado con el prefijo Teredo 2001:0000::/32 (32bits), y la dirección IPv4 del servidor Teredo (32 bits), como es mostrado en la figura 88. El cliente Teredo autoconfigura por si mismo la parte del identificado de interfaz usando los siguientes componentes: banderas (16 bits), número del puerto externo del NAT que mapea al cliente (16 bits), y la dirección IPv4 externa del NAT que mapea al cliente (32 bits), los dos últimos serán recibidos dentro del TRA como un campo especial llamado origen (origin), en el paquete TRA.

3.10.4. Enviando paquetes IPv6 usando Teredo

La figura 89 muestra un ejemplo con los componentes Teredo, usaremos esta figura para la descripción del flujo de los paquetes:

- Cliente Teredo (C1)
- NAT (N1)
- Servidor Teredo (S1)
- Relay Teredo (R1)
- Nodo B IPv6 NO-Teredo

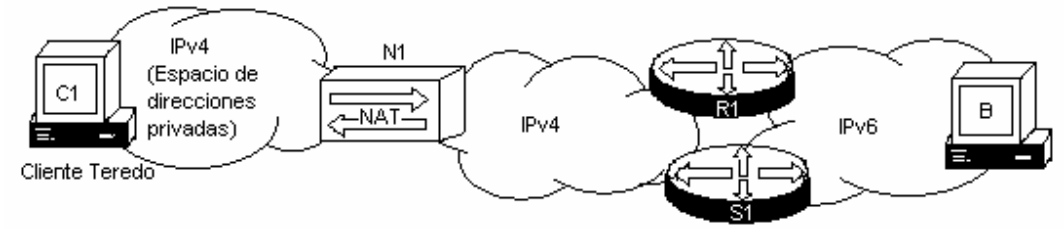


Figura 89. Ejemplo de una red con componentes Teredo.

Enviando un paquete IPv6 de un cliente Teredo hacia un nodo IPv6 NO-Teredo

Cuando un cliente Teredo envía un paquete hacia un destino IPv6 NO-Teredo, muchos paquetes son intercambiados antes de enviar el paquete.
La tabla 24 describe los pasos de la figura 90. El paso 0 es mantener el paquete IPv6 a ser enviado hacia el destino IPv6: el primer paquete hacia el nuevo destino es puesto en buffer hasta que los siguientes pasos sean completados. Los pasos 1 hasta el 18 son hechos para abrir un nuevo hueco en el NAT del cliente Teredo para alcanzar directamente el Relay Teredo usado por el destino IPv6. Sin embargo, el Relay Teredo no es conocido por el cliente Teredo, y solamente será conocido cuando el destino IPv6 envié un paquete IPv6 hacia el cliente Teredo. La ventaja de esto es que el Relay Teredo puede estar ubicado en diferentes partes de la red, por lo cual se puede asignar el Relay Teredo que optimice mejor el trayecto del cliente Teredo hacia el NAT, hacia el Internet IPv4, hacia el Relay Teredo, hacia el Internet IPv6, hacia el destino IPv6.

Los pasos 1 a 5 son un ping enviado del cliente Teredo C1 hacia el destino IPv6 NO-Teredo nodo B. Para contestar el ping (paso 6), el destino IPv6 alcanza su Relay Teredo más cercano. Los pasos 7 al 15 perforan el nuevo hueco en el NAT para el Relay. Los pasos 16 al 18 son la contestación del paquete ping, el cual le dice al cliente Teredo que el hueco esta perforado. Los pasos del 19 al 23 son los paquetes enviados del cliente Teredo hacia el destino IPv6, usando directamente el mapeo del Relay Teredo. Así como la mayoría de los nodos en el Internet IPv6 no son Teredo, cada nueva conexión hacia un nuevo destino, iniciada por el cliente Teredo, comienza de nuevo con todo el proceso.

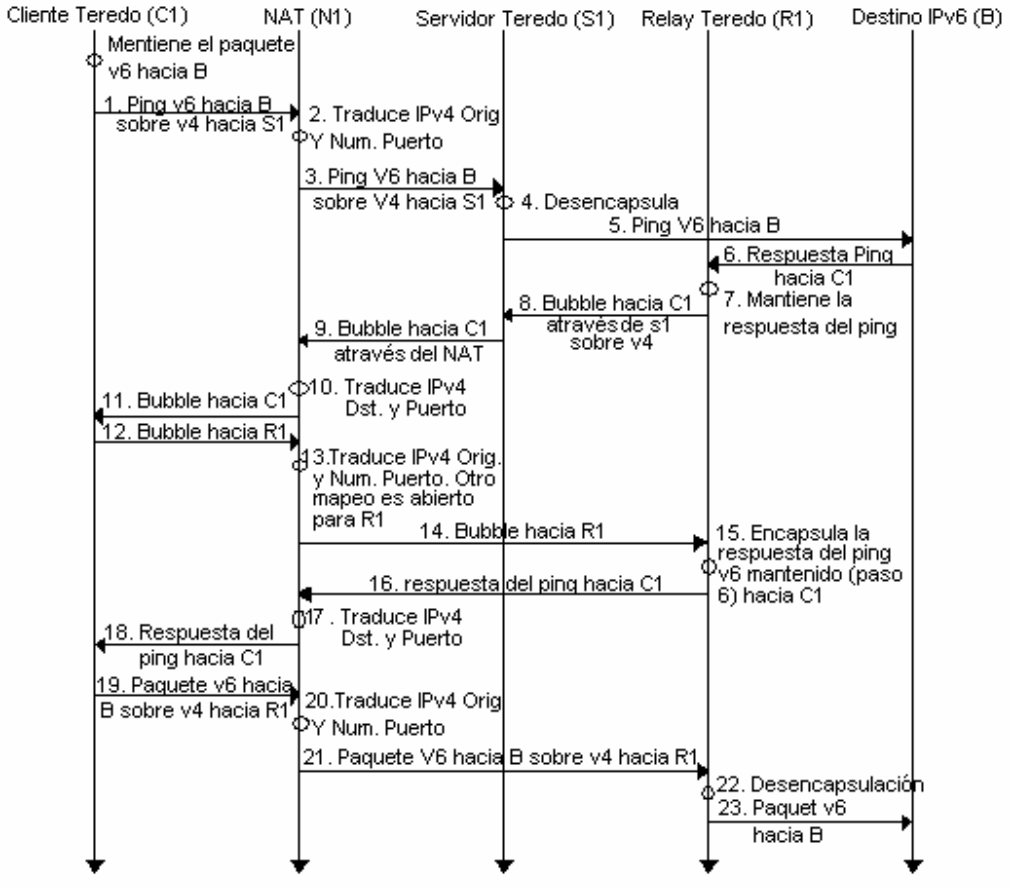


Figura 90. Flujo de un paquete del cliente Teredo hacia un NO-Teredo

Tabla 24. Paquete del cliente Teredo hacia un NO-Teredo

	Descripción
0	El cliente Teredo quiere enviar un paquete IPv6 hacia un destino IPv6, el cual no tiene dirección IPv6 Teredo. El cliente Teredo mantiene el paquete IPv6 en buffer antes de enviarlo.
1	El cliente Teredo envía un ping IPv6 hacia el destino, encapsulado en UDP-IPv4, con la dirección destino IPv4 de su servidor Teredo (S1).
2	La dirección IPv4 origen y el número del puerto son mapeados por el NAT (N1).
3	El paquete del ping IPv6 encapsulado en UDP-IPv4 es recibido por el servidor Teredo (N1).
4	El servidor Teredo desencapsula el paquete del ping IPv6.
5	El servidor Teredo S1 envía el paquete del ping IPv6 directamente hacia B sobre la red IPv6.
6	B envía la contestación del ping IPv6 hacia la dirección C1.

7	Basado en el routing IPv6, el Relay Teredo más cercano anuncia el prefijo Teredo hacia el destino C1 de la contestación del ping IPv6. El Relay mantiene en un buffer el paquete del ping de la contestación IPv6.
8	El Relay Teredo R1 envía un paquete IPv6 en UDP-IPv4 tipo bubble hacia C1, con la dirección destino IPv4 de S1.
9	El servidor Teredo S1 recibe el paquete tipo bubble y lo envía hacia C1.
10	El NAT N1 recibe el bubble y los envía hacia C1 remapeando la dirección destino IPv4 y el puerto.
11	El cliente Teredo C1 recibe el bubble.
12	El cliente Teredo C1 contesta con un bubble IPv6 en UDP-IPv4 hacia el Relay Teredo R1, sin tener que pasar através del servidor Teredo S1.
13	El NAT N1 crea un nuevo mapeo para la conexión hacia el Relay Teredo R1.
14	El bubble es enviado hacia el Relay Teredo R1. El Relay ahora conoce el mapeo específico sobre el NAT (N1) del cliente Teredo, esto es realizado con el propósito de enviar directamente paquetes del Relay hacia el cliente sin tener que pasar através del servidor Teredo (S1).
15	La contestación del paquete del ping IPv6 mantenido en el paso 7 es ahora encapsulado sobre UDP-IPv4.
16	Paquetes encapsulados son enviados por el Relay hacia C1 a trabes del NAT N1.
17	EL NAT N1 envía el paquete de constelación del ping IPv6 en UDP-IPv4 hacia C1, usando el nuevo mapeo establecido en el paso 13.
18	La contestación del ping es recibido por el cliente Teredo C1. Ahora el cliente conoce que un nuevo camino es establecido entre el cliente y el Relay através de un mapeo específico hecho en el NAT.
19	El paquete IPv6 retenido en el paso 0 es ahora enviado sobre UDP-IPv4, con la dirección destino IPv4 del Relay Teredo R1.
20	La dirección origen IPv4 y el número de puerto son mapeados por el NAT (N1).
21	El paquete encapsulado es recibido por el Teredo Relay R1.
22	El Relay Teredo R1 desencapsula el paquete IPv6.
23	El Relay Teredo R1 envía el paquete IPv6 nativo sobre la red IPv6 hacia el destino IPv6 B.

Enviando un paquete IPv6 de un nodo NO-Teredo hacia un cliente Teredo

Cuando un nodo IPv6 NO-Teredo envía un paquete hacia un nodo Teredo, el nodo origen IPv6 no hace algún proceso especial. El paquete IPv6 alcanzará al Relay Teredo más cercano que esté anunciando el prefijo Teredo sobre el Internet IPv6. Dado que el Relay no ha abierto con anterioridad un mapeo con el NAT que se encuentra frente al nodo Teredo, un viaje redondo de un paquete tipo bubble es usado para abrir un nuevo mapeo del cliente Teredo hacia el Relay. Entonces el paquete IPv6 original es enviado hacia el cliente Teredo, através del NAT.

Enviando un paquete IPv6 entre dos clientes Teredo

Cuando un cliente Teredo envía un paquete IPv6, la dirección destino IPv6 es verificada contra el prefijo Teredo. Si el destino es otro cliente Teredo, entonces ambos clientes inician mensajes tipo bubble para abrir sus respectivos NAT para establecer comunicación directa entre los dos NAT sin necesidad de ir através de Relays Teredo o Servidores
Si los dos clientes Teredo están en el mismo enlace, Teredo tiene un mecanismo de descubrimiento, usando una dirección IPv4 multicast, para enviar mensajes tipo bubble entre los dos clientes Teredo.

3.10.5. Requerimientos y limitaciones

Los requerimientos para desarrollar Teredo son:

- Los nodos IPv6 deben ser dual-stack.
- Los nodos IPv6 implementan Teredo.
- Un servidor Teredo conectado hacia el Internet IPv4 y el Internet IPv6 es disponible para los clientes Teredo.
- Muchos Relay Teredo son disponibles y dispersos en el Internet IPv4 y el Internet IPv6.
- Symetric NAT no existen entre el Teredo cliente y el Teredo servidor.
- La dirección IPv4 del Teredo servidor debe ser configurada estáticamente en todo los clientes Teredo.

La dirección Teredo incrusta la dirección IPv4 del servidor Teredo, el cliente Teredo mapea la dirección IPv4 y el puerto, esta dependencia en la dirección IPv4 hace el uso de Teredo limitada para los nodos que no necesitan una permanente o estable dirección IPv6. Desde que el tiempo de vida de la dirección Teredo puede ser tan corto como 30 segundos, un cliente Teredo entonces no puede usar su dirección para cualquier servicio público. Los clientes Teredos son nodos realmente mudos.

Comunicación inicial entre clientes Teredo y nodos IPv6 u otros clientes Teredo se inicia utilizando el buffer, mientras se perfora un hueco Teredo, y se descubre el Relay. Combinado con posibles retardos en esta fase y posibles tiempos fueras de todos los pasos, lo que significa para el usuario bajo servicio para nuevas comunicaciones. Cuando una pagina Web es cargada con 50 referencias de diferentes sitios Web IPv6, cada referencia podría generar muchos huecos, y viajes redondos de paquetes tipo bubble. El buffer inicial de paquetes IPv6 en Relay Teredo, servidores y clientes significa adicional memoria, todo esto afecta la característica de la plataforma Teredo.

Teredo se basa principalmente en el comportamiento de NAT. Esto basado en resultados experimentales de diferentes distribuidores hasta ahora, por lo tanto esto no es una garantía, ya que el comportamiento de los NAT probablemente no sea siempre el mismo. Así que podrían los NAT comportarse de diferente manera en nuevas versiones.

Teredo no trabaja si un NAT symmetric se encuentra en el camino entre el cliente Teredo y el servidor Teredo. Un cliente Teredo en esta situación no tiene conectividad IPv6. NAT symmetric son conocidos por ser usados en la mayoría de las empresas, porque su implementación es la más segura para trasladar direcciones y puertos.

El desarrollo de Teredo sería poco eficiente si unos pocos Relays Teredo son disponibles. Poco tiempo de respuesta y largos caminos sería el resultado. También Teredo asume que el Relay Teredo siempre será el mejor camino entre los dos puntos finales, esto no siempre es el caso y la ingeniería de tráfico no puede ser realizado. Finalmente, clientes Teredo atrás del mismo NAT, pero que no estén en el mismo enlace, puede que no están

habilitados para comunicarse. En este caso, el cliente Teredo no tiene forma de comunicarse.

3.11. NAT-PT

NAT-PT (Network Address Traslator-Protocolo Translator), es un traductor de direcciones por medio de un protocolo traductor SIIT (Stateless IP ICMP Translation). NAT-PT provee la conectividad entre hosts en una red IPv6 y una red IPv4 basándose en la traducción de direcciones modificando los encabezados del protocolo.

Como se muestra en la figura 91, un NAT-PT es requerido cuando un host IPv6 y un IPv4 necesitan comunicarse, entre ellos mismos o con cualquier otro.

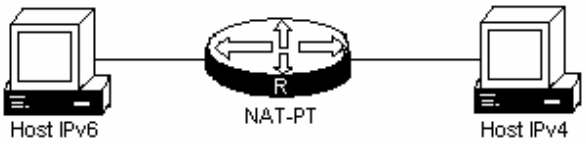


Figura 91. Red NAT-PT

Antes de que se esparza ampliamente la adopción de IPv6, se espera un periodo largo de transición durante el cual, nodos IPv4 e IPv6 necesitaran comunicarse. La comunicación requerirá la traducción de direcciones de versión 4 a 6 y viceversa. El NAT-PT que reside dentro de un enrutador de frontera de la red IPv4 o IPv6, provee esta traducción.

3.11.1. El mecanismo NAT-PT

Tres mecanismos NAT-PT son disponibles para traducir una dirección IPv4 hacia una dirección IPv6 y viceversa.

1) Mapeo NAT-PT estático.

NAT-PT Estático usa reglas de traducción estáticas para mapear una dirección IPv6 hacia una dirección IPv4. Nodos de la red IPv6 se comunica con nodos de la red IPv4 usando un mapeo IPv6 de las direcciones IPv4 configuradas en el enrutador NAT-PT.

La figura 92 muestra cómo un nodo IPv6-only llamado A puede comunicarse con el nodo IPv4-only llamado C usando NAT-PT. El dispositivo NAT-PT está configurado para mapear la dirección IPv6 origen del nodo A 2001:0db8:bbbb:1::1 a la dirección IPv4 192.168.99.2. El NAT-PT es también configurado para mapear la dirección origen IPv4 del nodo C 192.168.30.1 a la dirección IPv6 2001:0db8::a. Cuando paquetes con una dirección origen IPv6 del nodo A son recibidos en el enrutador NAT-PT, él tiene que traducirla para que se tenga una dirección destino que coincida con el nodo C en la red IPv4-only. El NAT-PT puede también ser configurado para que coincida la dirección IPv4 y que traduzca el paquete hacia una dirección destino IPv6, de esta forma permitiendo que los hosts IPv4-only comunicarse con los hosts IPv6-only.
Si se tiene múltiples hosts IPv6-only o IPv4-only que necesiten comunicarse, tendrías que configurar muchos mapeos estáticos NAT-PT. NAT-PT Estático es usado cuando

aplicaciones o servidores requieren acceso a una dirección IPv4 estable. El acceso a un DNS externo es un ejemplo donde NAT-PT Estático puede ser usado.

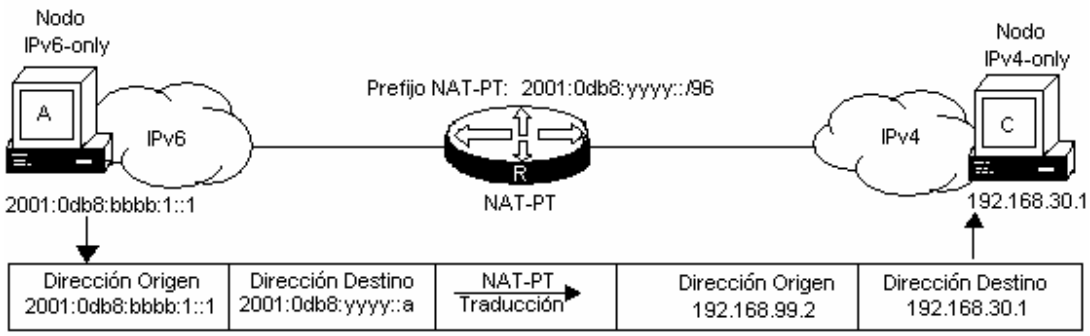


Figura 92. NAT-PT Estático.

2) Mapeo NAT-PT dinámico.

El NAT-PT dinámico permite múltiples mapeo NAT-PT para la asignación de direcciones de un pool. El NAT-PT es configurado con un pool de direcciones IPv6 y/o IPv4. Al inicio de una sesión NAT-PT una dirección temporal es dinámicamente asignada de un pool. El número de direcciones disponibles en el pool de direcciones, determina el número máximo de sesiones concurrentes. El dispositivo NAT-PT registra cada mapeo entre las direcciones, en una tabla de estado dinámico.

La figura 93 muestra cómo opera NAT-PT dinámico. El nodo B IPv6-only puede comunicarse con el nodo D IPv4-only usando NAT-PT dinámico. El dispositivo NAT-PT dinámico es configurado con una lista de acceso, lista de prefijos, o mapa de rutas para determinar qué paquetes serán traducidos por el NAT-PT. Un pool de direcciones IPv4 (10.21.8.1 a 10.21.8.10 en la figura 93) son también configuradas. Cuando un paquete IPv6 es identificado para ser traducido, el NAT-PT usa la regla configurada de mapeo y asigna una dirección IPv4 temporal del pool de direcciones IPv4 configurado.

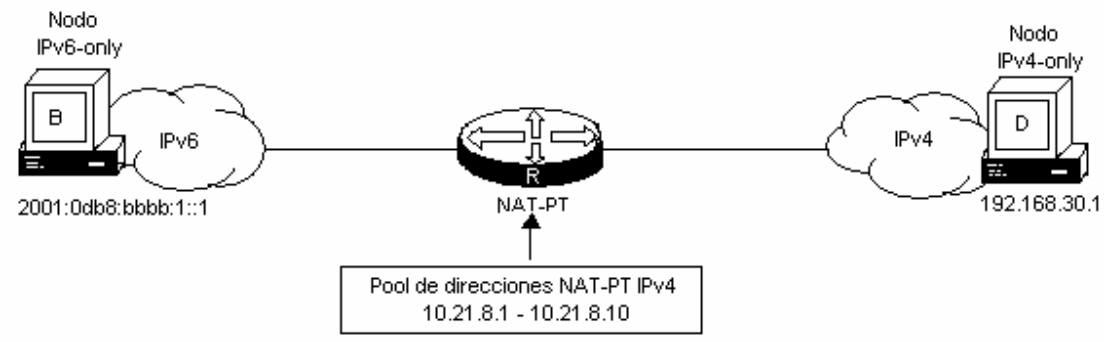


Figura 93. NAT-PT Dinámico.

La operación de traducción NAT-PT dinámico requiere al menos un mapeo estático para el servidor DNS IPv4. Después que la conexión de IPv6 hacia IPv4 es establecida, la contestación de los paquetes de IPv4 hacia IPv6 toma ventaja del mapeo dinámico

previamente establecido para traducir de IPv4 a IPv6. Si la conexión es iniciada por un hosts IPv4-only entonces se repite todo el proceso.

a. Traducción de direcciones de puerto (PAT) o sobrecarga.

PAT (Port Address Translation), mejor conocido como sobrecarga, permite a una sóla dirección IPv4 ser usada entre múltiples sesiones, multiplexando el número del puerto para asociar algunos usuarios IPv6 con una sóla dirección IPv4. La traducción de direcciones de puerto, puede ser realizada através de una interfaz específica o através de un pool de direcciones. La figura 94 muestra múltiples direcciones IPv6 de la red IPv6 enlazadas a una sola interfaz IPv4 dentro de la red IPv4.

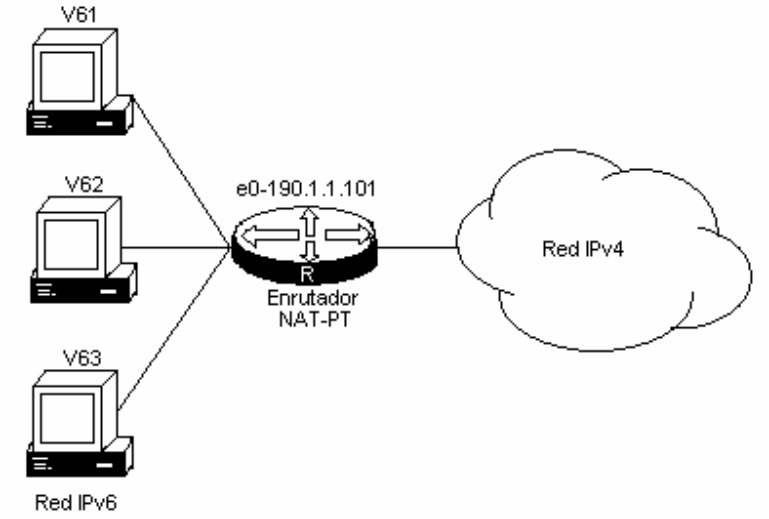


Figura 94. Traducción de dirección de puerto (PAT).

3) NAT-PT con DNS-ALG

NAT-PT-DNS-ALG traduce registros AAAA de las peticiones DNS y responde con paquetes tipo A y vice-versa.

DNS-ALG soporta los siguientes tipos de registros DNS:

- Peticiones A del host IPv4.
- Peticiones AAAA del host IPv6.
- Respuesta A del servidor DNS IPv4.
- Respuesta AAAA del servidor DNS IPv6.
- Peticiones PTR y CNAME de hosts IPv4 e IPv6.
- Respuesta PTR y CNAME del servidor DNS IPv4 e IPV6.

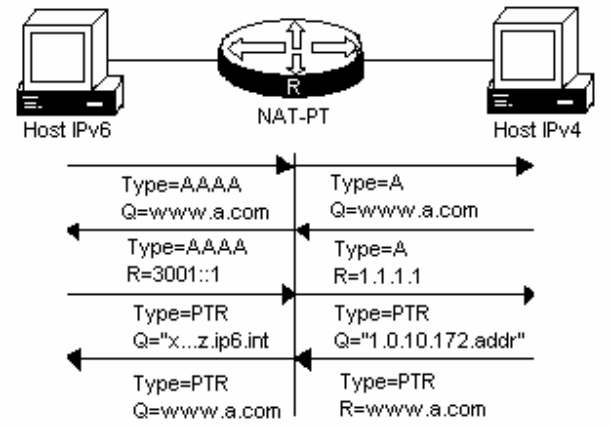


Figura 95. NAT-PT con DNS-ALG

Si un host IPv6 intenta comunicarse con el nodo IPv4 usando un nombre de host en vez de la dirección IP, el nombre del host primero debe ser trasladado por el DNS-ALG en el NAT-PT. El procedimiento de comunicación es el siguiente:

I. Un servidor IPv4 en la red IPv4 mantiene el mapeo del dominio "www.a.com" para las direcciones IPv4. El host IPv6 envía la petición AAAA con el nombre "www.a.com" para resolver. La dirección destino de este paquete es una dirección IPv6 con la longitud del prefijo de 96. El prefijo es puesto especialmente para el envio de tráfico hacia la red IPv4. La dirección destino de esta petición DNS (la dirección del servidor DNS) es configurada estáticamente en el host IPv6.
II. El enrutador NAT-PT traduce la petición AAAA dentro de una petición A con la misma cadena de petición "www.a.com". La dirección IPv4 del servidor DNS es mapeada estáticamente. La dirección IPv6 origen del paquete (la dirección IPv6 del host IPv6) puede ser estáticamente o dinámicamente traducida dentro de una dirección IPv4.
III. El servidor DNS IPv4 contesta la petición que incluye una dirección IPv4 y el nombre"www.a.com". El contenido del paquete de la respuesta DNS, puede ser visto como la traducción dinámica que fue creada en el enrutador NAT-PT (si mapeo estático no existe) entre la dirección del paquete de respuesta DNS IPv4 y la dirección IPv6 contenida en la petición con el prefijo /96. El enrutador NAT-PT puede también usar este mapeo después, para traducir la dirección IPv6 del paquete enviado hacia el host B, dentro la dirección IPv4 del host B.
IV. El enrutador NAT-PT traduce la repuesta A del servidor DNS en un respuesta AAAA, y cambia la dirección IPv4 del paquete de respuesta DNS en una dirección IPv6 de acuerdo al mapeo mencionado en el paso 3. Como resultado, el host A puede enviar un paquete hacia el host B usando la dirección traducida IPv6. Cuando el paquete IPv6 alcanza al enrutador, el NAT-PT traduce la dirección IPv6 dentro de una dirección IPv4 del host B basándose de la previa traducción dinámica creada.
V. Por razones de seguridad, el servidor FTP verifica si el registro PTR existe para la dirección IP de la conexión entrante. Basándose en esta verificación, el enrutador NAT-PT realiza la traducción del registro PTR.
   En este caso, la dirección IPv6 mapeada de "www.a.com" es conocida. Para verificar si un registro PTR existe para esta dirección, el host A crea una petición IPv6 PTR

para “2010::45”. Este paquete es enviado hacia la misma dirección IPv6 como en el paso 1.

VI. En el enrutador NAT-PT, la traducción IPv4 hacia IPv6 del host B (ver paso 3) traduce la petición del PTR IPv6 dentro una petición PTR IPv4. Si esta traducción aún no ha sido creada, podría ser imposible traducir la petición de traducción PTR IPv6 dentro una petición PTR IPv4 porque el enrutador NAT-PT podría no conocer la dirección IPv4 destino. La dirección IPv6 del servidor DNS es traducida dentro la dirección IPv4 como en el paso 2.

VII. El DNS IPv4 contesta la petición PTR conteniendo el nombre “www.a.com”.

VIII. El enrutador NAT-PT traduce la respuesta PTR IPv4 dentro una respuesta IPv6 PTR. La dirección IP del encabezado IPv4 puede ser traducido siguiendo el mapeo dinámico creado en el paso3.

3.11.2. Ventajas y desventajas del NAT-PT

La ventaja del NAT-PT es la siguiente:

- Acceso de host IPv4 hacia host IPv6 en una red externa puede ser implementado sin tener que cambiar la infraestructura IPv4 existente.

Las desventajas del NAT-PT son las siguientes:

- Las peticiones y repuestas pertenecientes a la misma sesión deben ser enrutadas através del mismo enrutador NAT-PT.
- Campo de opciones en el encabezado IPv4 no pueden ser traducidas.
- Carencia de seguridad end-to-end.

Capítulo

4.Protocolos de Enrutamiento para IPv6

Resumen

Los enrutadores (routers) son dispositivos de red que raramente se encuentran aislados entre sí. Al contrario, suelen estar interconectados, formando una especie de "telaraña" que hace posible el tráfico de datos entre redes separadas físicamente. La principal función de los enrutadores es la de enviar paquetes de datos de un host origen a uno destino através de la red, escogiendo la mejor ruta a su destino y el menor número de saltos posibles. Para poder realizar esta tarea, los enrutadores se comunican constantemente entre sí, informándose de las rutas que conocen para mantener actualizada su tabla de enrutamiento y evitar tener registros inválidos. Esto se consigue por medio de una serie de protocolos de enrutamiento, responsables de que los diferentes enrutadores mantengan sus tablas de enrutamiento acordes, obteniéndose una red convergente.

4.1. INTRODUCCIÓN AL ENRUTAMIENTO IP

La primera diferencia entre un host ordinario y un enrutador, es que el enrutador es configurado para aceptar paquetes esperados para otro destino y enviar estos paquetes hacia el que el enrutador determine sea el mejor siguiente salto. El enrutador usualmente también soporta por lo menos un protocolo de enrutamiento através del cual éste adquiere la información actual acerca de las rutas de la red.

El más simple de los enrutadores son aquellos que sirven a una sola red con dos interfaces: una para la red local y la otra para enviar todo el tráfico restante. Estos enrutadores funcionan como puerta de enlace (gateway) para la red local. Hosts locales reconocen dos tipos de destinos: aquellos hosts que están sobre la subred local lógica IP "LIS" (logical IP subnett) y que pueden ser alcanzados directamente sobre el enlace local, y aquellos hosts que no son locales (se encuentran en cualquier otro lugar). Hosts en esta red son configurados para entregar paquetes locales directamente, entre ellos mismos, sobre la capa de enlace, y todos los demás paquetes son enviados hacia la puerta de enlace IP (el enrutador local), el cual los envía entre sus otras interfaces. Los enrutadores locales (o gateway) típicamente son configurados para aceptar paquetes de entrada destinados para la red local y para enviar cualquier paquete recibido de la red local hacia su propio enrutador upstream. Si el gateway está sobre un enlace punto a punto, éste no hace nada más que pasar los paquetes de la red local hacia el otro punto final del enlace del sistema, como lo muestra la figura 96.

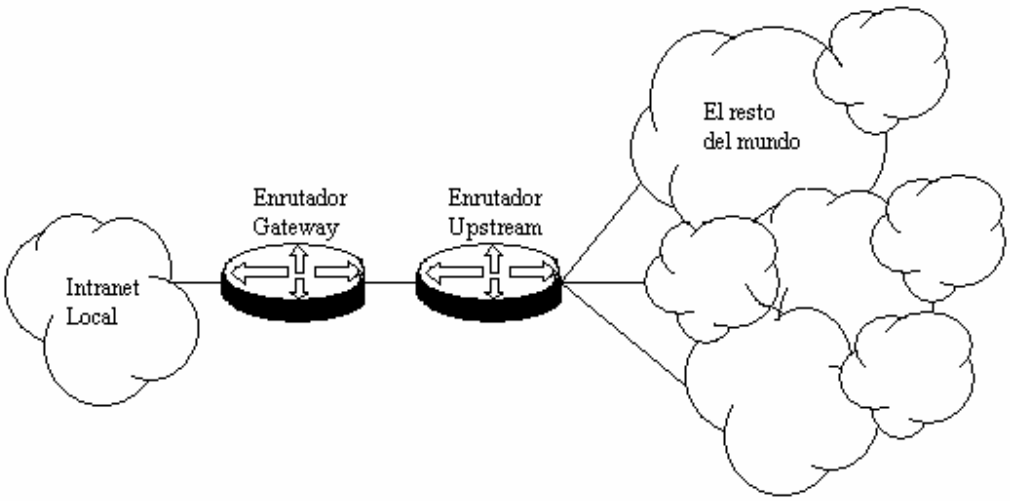


Figura 96. Arquitectura de un enrutador simple local (gateway).

4.1.1. Redes enrutadas

Como las intranets llegan a ser más complejas con más de una LIS interna, esparciéndose a LAN´s, MAN´s, o WAN´s, entonces el enrutamiento interno llega a ser necesario. Estos enrutadores proveen conectividad a los hosts dentro de la Intranet así como también hacia el resto del Internet global, si es el caso. El número y tipo de enrutadores, así como el número de redes de cada enlace del enrutador, dependen del diseño de la red, metas de organización y requerimientos para tal red. La figura 97 muestra cómo en una simple

Intranet multi-rutas, los enrutadores internos deben decidir cómo será el mejor envio de los paquetes para la red local.

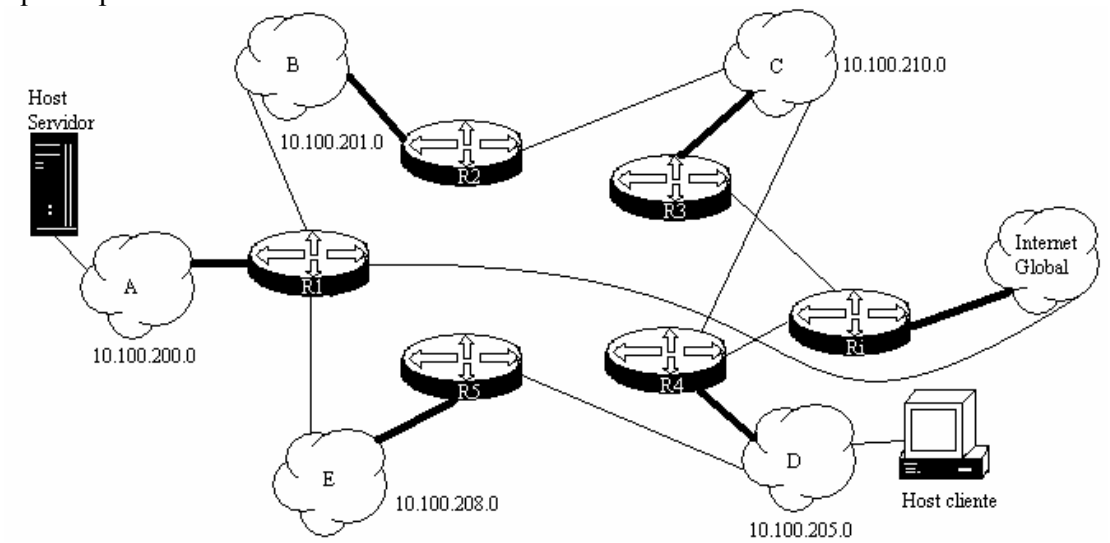


Figura 97. Dominio de enrutamiento Intranet complicado.

Usando el ejemplo de la figura 97, se puede apreciar fácilmente que los paquetes enviados de un host de la red A y destinados para un host de la red D, tendrían que ser enviados por el enrutador R1, el cual tendría que enviarlo al enrutador R5 sobre la red E; de aquí los paquetes son enviados hacia el enrutador R4 sobre la red D. El enrutador R4 entonces enviará el paquete directamente hacia el host destino.

Cuando el mismo paquete tiene que volver a entregarse, pero el enrutador R5 no está disponible por alguna razón, el enrutador R1 tendrá que enviar el paquete hacia otro enrutador que sea capaz, como última instancia, de entregar el paquete hacia la red D. Otra opción abierta, es el caso que sean enviados los paquetes hacia el Internet global (no aceptable) o hacia el enrutador R2 sobre la red B. El enrutador R2 envía el paquete hacia el enrutador R3, el cual envía el paquete hacia el enrutador R4, mismo que entrega el paquete hacia su destino. El enrutador Ri es el gateway de Internet, envía paquetes hacia y del Internet global, el enrutador R1 es un respaldo del gateway de Internet.

Con cinco LIS´s internos, más el Internet global, hay seis diferentes LIS´s contenidas con completa ínter conectividad entre ellas, así todas las redes son alcanzables por cualquier otra, en un salto, requiriendo 15 enlaces.

Sin embargo, los enrutadores necesitan información (cuál de sus propios enlaces se encuentran en estado “arriba” (up) o “abajo” (down), qué porción de la red de otros enrutadores pueden alcanzar). El intercambio y actualización de esta información son las primeras metas de los protocolos de ruteo, y para ayudar estas metas es necesario evitar la propagación de información falsa, mientras al mismo tiempo la optimización se realiza (minimizando el número de saltos tomados de un origen hacia un destino).

En la figura 97 los enrutadores son enrutadores interiores porque ellos enrutan paquetes dentro un AS (Sistema Autónomo) u otro dominio de ruteo; dentro del dominio, los LIS´s

son dominio administrativos (ADs) separados. Un AD es comparable a un AS, excepto que es a menor escala. Así, el ruteo exterior, el cual ocurre entre enlaces de enrutadores con diferente AS através del backbone, requiere una diferente característica para el intercambio de información y determinación de las rutas óptimas.

4.1.2. Rutas estáticas y dinámicas

Las rutas estáticas pueden ser fácilmente configuradas en un sistema, ellas necesitan bajos requerimientos del sistema. Son aplicables a simple y pequeñas redes escalables. Las rutas estáticas no pueden ser adaptadas automáticamente a los cambios de la topología de la red. De esta forma ellas deben ser configuradas manualmente.

Con sus propios algoritmos de ruteo; protocolos de ruteo dinámico pueden ser adaptados automáticamente a los cambios de la topología de la red. Así ellos son aplicables a redes equipadas con cierto número de dispositivos capa 3. De esta forma., los enrutadores dinámicos son complicados y difíciles de configurar. También necesitan altos requerimientos del sistema.

Los protocolos de ruteo pueden ser también clasificados por las siguientes condiciones:

I. De acuerdo al rango de funciones
   - Interior Gateway Protocol (IGP): corren dentro un AS. Se pueden mencionar RIP, OSPF e IS-IS
   - Exterior Gateway Protocol (EGP): corren entre diferentes AS. Se puede mencionar a BGP

II. De acuerdo al Algoritmo
   - Protocolo de ruteo Distance-Vector: incluye RIP y BGP (BGP también es llamado Path-Vector)
   - Protocolo de ruteo Link-State: incluye OSPF e IS-IS

III. De acuerdo al tipo de dirección destino
   a. Protocolo de ruteo Unicast: incluye RIP, OSPF, BGP e IS-IS.
   b. Protocolo de ruteo Multicast: incluye DVMRP, PIM-SM y PIM-DM

Los protocolos de ruteo IPv6 aún usan la coincidencia del prefijo más largo como algoritmo de ruteo para la selección de rutas, como su equivalente IPv4. De esa forma, los protocolos IPv6 son definidos como una nueva familia de protocolos, la tabla de ruteo IPv6 es manejada separadamente de la tabla de ruteo IPv4, cuando ambos protocolos son habilitados simultáneamente en el enrutador.

4.1.3. Protocolos de ruteo Interior y Exterior

Las dos tareas básicas del ruteo son, primero, asegurarse que todas las redes dentro de Internet, rutean tráfico apropiadamente entre ellas mismas (ruteo interior); y segundo, asegurar que todas las Internetworks conectados hacia un gran Internet (tal como el Internet global) sean capaz de rutear confiablemente entre cada una de las demás (ruteo exterior).

Simples estrategias de ruteo como default gateways y anuncio de rutas ICMP serán suficientes para mover tráfico de red dentro de la mayoría de las intranets.

Sin embargo los protocolos de ruteo no definen el proceso de ruteo, ellos sólo definen el proceso por el cual los enrutadores intercambian información acerca de la red. Información de la tabla de ruteo debe ser mantenida actualizada, y los enrutadores constantemente se comunican entre ellos para anunciar su propia conectividad.

Típicamente los hosts adquieren cualquier información de ruteo como parte de su configuración estática o através de DHCP (Dynamic Host Configuration Protocol). El host usa ARP para adquirir una dirección física para todo el tráfico local; y para todo el demás trafico, es pasado hacia el enrutador gateway por default. En redes más pequeñas, el enrutador conectado directamente hacia el enrutador del ISP, es conectado alternadamente hacia un Backbone de Internet, una red enlazando más de un AS. Los enrutadores sobre la red de backbone deben mantener tablas de ruteo menos comprensibles porque ellos deben rutear entre y cualquier redes. Ellos usualmente no tienen un gateway por default especificado, por lo tanto a veces son referidos como enrutadores nondefault.

Protocolos Exterior o routing de backbone deben permitir la comunicación entre enrutadores para reportar cambios frecuentes en condiciones y conectividad, rápida y eficientemente. Un protocolo interior de ruteo permite a los enrutadores dentro de pequeñas redes, reportar sus propias condiciones y conectividad, aunque generalmente soportando arquitecturas de ruteo menos complicadas. El protocolo interior de ruteo soportado por un enrutador es frecuentemente referido como Interior Gateway Protocol (IGP), donde “gateway” es usado como sinónimo de enrutador; y protocolo de ruteo exterior es ampliamente conocido como Exterior Gateway Protocol (EGP).

4.1.4. Algoritmos de Ruteo

La formulación más simple de una estrategia de ruteo es la de optar por el camino más corto de la ruta donde sea que exista una opción. Como determinar cual es el camino corto presente. Hay dos estrategias dominantes para determinar el camino más corto para ruteo interior, cada uno es implementado en su propio protocolo. El algoritmo de de ruteo vector-distancia (distance-vector), también conocido como el algoritmo Bellman-Ford, es descrito en el RFC1058, “Routing Information Protocol”, el cual también define el protocolo de ruteo RIP para redes IP. Otra estrategia para ruteo interior es el llamado algoritmo Dijkstra, y este también es conocido como estado de enlace (link-state) o algoritmo “open shortest path first” (OSPF). El OSPF es también el nombre del protocolo de ruteo interior definido en el RFC2328, “OSPF versión 2”.

4.1.5. Distancia Administrativa

Diferentes protocolos de ruteo (así como configuración estática) pueden aprender diferentes rutas hacia un mismo destino, pero no todas estas rutas son óptimas. En un cierto momento, sólo un protocolo de ruteo determinará la mejor ruta hacia un destino específico. Cada uno de estos protocolos de ruteo (incluyendo la configuración estática) tiene un valor numérico de preferencia, a la cual se la da el nombre de distancia administrativa. El valor de la

distancia administrativa más bajo, es la que tiene mayor prioridad para el enrutador. Las distancias administrativas manejadas por los protocolos de ruteo para IPv6 son las mismas que se usan para IPv4. La tabla 25 muestra las distancias administrativas soportadas por cada protocolo de ruteo IPv6.

Tabla 25. Distancias administrativas de protocolos de ruteo IPv6[17]

Protocolo de ruteo	Distancia Administrativa (Default)
Interfaz Conectada	0
Ruta estática (hacia la interfaz)	0
Ruta estática (hacia el siguiente salto)	1
External BGP (eBGP)	20
OSPF	110
IS-IS	115
RIP	120
Internal BGP (iBGP)	200

4.1.6. Métricas de ruteo

Tablas de ruteo contienen cierta información utilizada para realizar el switching para seleccionar la mejor ruta. Los algoritmos de ruteo utilizan muchas métricas diferentes para determinar la mejor ruta. Sofisticados algoritmos de ruteo pueden basarse de múltiples métricas para la selección de una ruta, combinándolas en una sola métrica, resultando una métrica híbrida. A continuación se describen las métricas utilizadas por diferentes algoritmos de ruteo.

- **Longitud de la Trayectoria (Path Lenght):** es la métrica más común. Algunos protocolos de ruteo permiten a los administradores de red asignar costos arbitrarios a cada enlace de red. En este caso, la longitud del trayecto es la suma de los costos asociados con cada enlace cruzado. Otros protocolos de ruteo definen la cuenta de saltos (hop), una métrica que especifica el número de pasos através de dispositivos de Internetworking, tal como enrutadores, que un paquete debe seguir para poder llegar de un origen hacia un destino.
- **Confiabilidad (Reliability):** en el contexto de algoritmos de ruteo, se refiere a la formalidad (usualmente descrito en términos de la taza del bit-error) de cada enlace de red. Algunos enlaces de red pueden caerse frecuentemente más que otros. Después de una falla en la red, ciertos enlaces de red pueden ser reparados más fácilmente o más rápidamente que otros enlaces. Cualquier factor de confiabilidad puede ser tomado dentro de una cuenta asignada de taza de confiabilidad, las cuales son valores numéricos arbitrarios usualmente asignados a los enlaces de la red por los administradores de red.
- **Retardo de ruteo (Routing delay):** se refiere a al tiempo requerido para mover un paquete del origen hacia su destino através del Interntework. El retardo depende de muchos factores, incluyendo el ancho de banda (bandwidth) del enlace intermedio de red, la cola de los puertos en cada enrutador a lo largo del camino, congestión de la red sobre todos los enlaces de red intermedios, y la distancia física a ser cruzada.

[17] Desmeules Régis. (May 2003). "Cisco Self-Study: Implementing Cisco IPv6 Networks (IPv6)". Edit. Cisco Press. Primera Edición. Pp. 159.

Porque el retardo es una conglomeración de varias variables importantes, es una métrica común muy utilizada.

- **Ancho de Banda (Bandwidth):** se refiere a la capacidad de tráfico disponible de un enlace. De esto que el ancho de banda sea una taza del máximo ancho de banda efectivo (troughput) sobre el enlace. Enrutadores através de enlaces con grandes anchos de banda no necesariamente proveen mejores rutas que enrutadores con enlaces más lentos. Por ejemplo, si un enlace muy rápido esta ocupado, el tiempo actual requerido par enviar un paquete hacia su destino podría ser mayor.
- **Carga (load):** se refiere al grado que los recursos de la red, tal como un enrutador, estén ocupados. La carga puede ser calculado en una variedad de formas, incluyendo utilización de CPU y procesamiento de paquetes por segundos.
- **Costo de Comunicación (Communication cost):** es otra métrica importante, especialmente porque algunas compañías no ponen tanta atención acerca del funcionamiento como en los costos de operación. Aunque las líneas sean más largas y retrasen, ellos enviaran los paquetes sobre sus propias líneas que por líneas públicas que cuestan dinero por el tiempo de uso.

4.2. RIPng

"Routing Information Protocol Next Generation" (RIPng) es una extensión del protocolo RIPv2 para redes IPv4. La mayoría de los conceptos RIP son aplicables para RIPng. Es un protocolo IGP, por lo tanto es principalmente usado en pequeñas y simples estructuras de red como redes de campus o redes regionales.

Para aplicaciones IPv6, RIPng tiene las siguientes diferencias del existente RIP:

- Número de puerto UDP: envía y recibe información de ruteo usando el número de puerto UDP 521.
- Dirección Multicast: usa FF02::9 como la dirección multicast del enrutador RIPng en el ámbito local del enlace.
- Longitud del Prefijo: La dirección destino usa un prefijo de 128 bits (la longitud de la mascara).
- Dirección de siguiente salto: usa una dirección IPv6 de 128 bits.
- Dirección origen: usa la dirección de enlace local FE80::/10 como la dirección origen para enviar los paquetes de actualización de la información de ruteo RIPng.

4.2.1. Principio de operación de RIPng

Es un protocolo basado en el algoritmo Vector Distancia e intercambia información de ruteo através de paquetes UDP (número de puerto 521). Emplea la cuenta de saltos para medir la distancia hacia el host destino, el cual es llamado "routing cost". En RIPng el número de saltos de un enrutador a su red conectada directamente es 0, y aquella red que puede ser alcanzada por otro enrutador es 1, y así sucesivamente. El número de saltos que es igual o excede el número 16, es definida como infinita, lo que nos dice que la dirección destino del host es inalcanzable. Por esta razón, RIPng no es aplicada a grandes redes.

RIPng envía un paquete de refresco de ruteo cada 30 segundos. Si no se es recibido el paquete de refresco de ruteo de un vecino de red en 180 segundos, entonces RIPng etiquetara todas las rutas del vecino de red que son inalcanzables. Si no se es recibido el paquete de refresco de ruteo de un vecino de red en 300 segundos, entonces RIPng finalmente removerá las rutas del vecino de red de la tabla de ruteo.

Para mejorar el funcionamiento y evitar loops de ruoting, RIPng soporta Split Horizon y Poison Reverse. Por otro lado, RIPng puede importar rutas de otros protocolos de ruteo.

Split Horizon es un algoritmo para evitar problemas causados por incluir rutas en actualizaciones enviadas hacia el gateway por donde fueron aprendidas. El algoritmo básico de split horizon omite rutas aprendidas por un vecino en actualizaciones enviadas hacia el vecino. Split Horizon con Poisoned Reverse (más simple, Poison Reverse) incluye las rutas mencionadas anteriormente en las actualizaciones, pero se pone las métricas a infinito. En efecto, anuncios hechos por tales rutas son inalcanzables.

Cada enrutador que correo RIPng maneja una base de datos de routing, la cual contiene entradas de ruteo de todos los destinos alcanzables en la red. Estas entradas de ruteo contienen la siguiente información:

- El prefijo IPv6 del destino.
- Una métrica, la cual representa el costo total de enviar un datagrama del enrutador hacia el destino. Esta métrica es la suma de los costos asociados con las redes que cruzará un paquete para alcanzar el destino.
- La dirección IPv6 del siguiente enrutador a lo largo del camino hacia el destino (por ejemplo el siguiente salto). Si el destino está en una de las redes conectadas directamente, este artículo no es necesitado.
- Una bandera para indicar qué información acerca de una ruta ha cambiado recientemente. Esta puede ser referida como la “route change flan”.
- Varios contadores asociados con la ruta, en particular relacionado cuando deben enviarse anuncios y para cuando las rutas deban estar en tiempo fuera.[18]

4.2.2. Formato de los paquetes RIPng

I. Formato básico RIPng

Un paquete RIPng consiste de un encabezado y múltiples entradas de tabla de ruteo (route table entries RTE). En un paquete RIPng, el número máximo de RTE depende del MTU en la interfaz. La figura 98 muestra el formato básico de un paquete RIPng.

[18] G. Malkin, R. Minnear (1997). RFC2080 " RIPng for IPv6”. Standards Track. Pp 3.

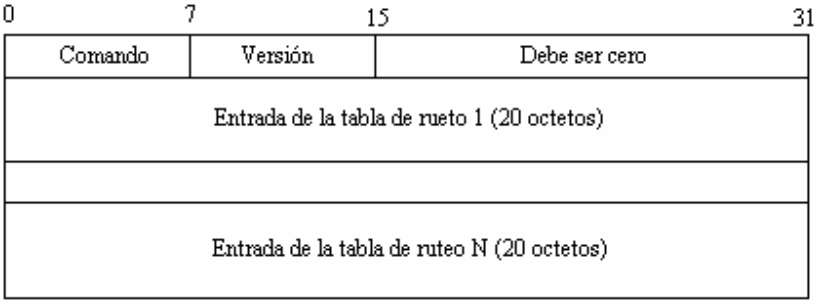


Figura 98. Formato básico de un paquete RIPng

Los principales son descritos como sigue:

- Comando (command): define el tipo de paquete. El valor 0x01 representa un paquete de petición (request), y el valor 0x02 representa un paquete de respuesta (response).
- Versión: versión de RIPng. Esta puede ser actualmente solo 0x01.
- Entrada de la tabla de ruteo (Route Table Entry): cada entrada es de 20 bytes de largo.

II. Formato de RTE

En RIPng, existen dos tipos de RTE´s:

- Siguiente Salto RTE (Next Hop RTE): otra importante característica de RIPng es la capacidad para especificar una dirección IPv6 como de siguiente salto para cualquier entrada de la tabla de ruteo. Esta es una característica incorporada en RIPv2 en la forma de un campo de siguiente salto en el RTE, el incluir una dirección de siguiente salto en la entrada de la tabla de ruteo RIPng podría incrementar el tamaño del RTE de 20 bytes a 36 bytes y así reducir el número de RTE´s que un enrutador pueda enviar hacia cualquier enlace dado. La solución fue definir una RTE de siguiente salto especial, mostrado en la figura 99. La dirección de siguiente salto RTE indica que todos los subsecuentes RTE´s hasta el final del mensaje RIPng (o hasta otra dirección de siguiente salto RTE es encontrada) usar la dirección de siguiente salto especificada.
- Prefijo IPv6 RTE (IPv6 Prefix RTE): entrada de la tabla de ruteo (RTE) RIPng toma el formato mostrada en la figura 100. Esta entrada transporta información de ruteo, especialmente la etiqueta de la ruta, longitud del prefijo y métrica de ruteo por cada ruta de prefijo IPv6.

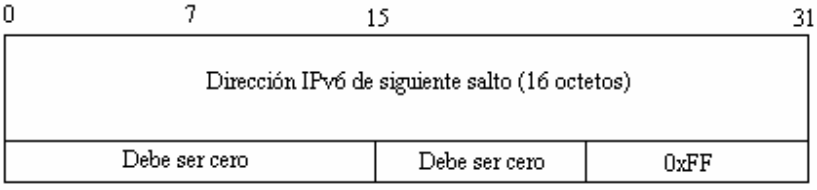


Figura 99. Formato de la dirección de siguiente salto RTE.

La dirección de siguiente salto IPv6 en el paquete representa la dirección IPv6 del siguiente salto.

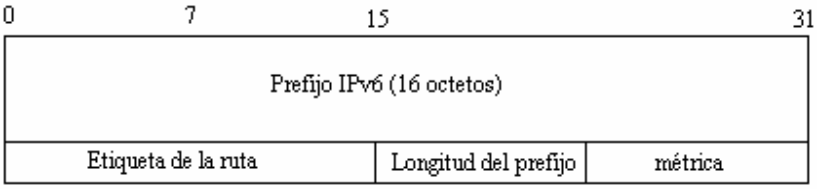


Figura 100. Formato de la entrada de la tabla de ruteo RTE.

Los campos son explicados como sigue:

- Prefijo IPv6: indica el prefijo de la dirección destino IPv6.
- Etiqueta de ruta (route tag): es usada para diferenciar entre rutas que son internas del dominio de ruteo RIPng y rutas que son importadas de otro dominio de ruteo, incluyendo otro dominio de ruteo interno, así como protocolos de ruteo externo tal como BGP.
- Longitud del prefijo: indica la longitud del prefijo de la dirección IPv6
- Métrica: indica el costo de ruteo.

4.2.3. Procesamiento de paquetes RIPng

- Paquetes de petición

Una petición es usada para pedir una respuesta que contenga todo o parte de la tabla de ruteo de enrutadores. Cuando un enrutador RIPng inicia o necesita algunas de sus entradas, éste envía un paquete de petición a sus vecinos, solicitándoles información de ruteo, de sólo una entrada o de la tabla completa según sea el caso. Usualmente, el paquete de petición es enviado en modo multicast, por el puerto RIPng.
El enrutador RIPng recibe el paquete de petición y procesa sus RTE´s. Si el paquete de petición sólo tiene un RTE, y son 0 el prefijo IPv6 y su longitud del prefijo, y el costo es infinito (16), este enrutador envía toda la información de ruteo de su tabla actual de enrutamiento hacia el enrutador que hizo la petición, por medio de un paquete de respuesta. Si no, el enrutador examina y procesa entrada por entrada de la lista RTE de la petición, actualiza el costo de cada ruta y finalmente regresa la información hacia el enrutador que hizo la petición.

- Paquetes de respuesta

Un paquete de respuesta contiene la información de la tabla de enrutamiento local. Este es recibido en ciertos casos:

- Respuesta a una petición específica
- Actualizaciones regulares (respuesta no solicitada)
- Actualización generada causada por el cambio de una ruta.

Los enrutadores reciben los paquetes de respuesta para actualizar sus propias tablas de ruteo. Como los paquetes de respuesta modifican las tablas de ruteo, se debe verificar

cuidadosamente el paquete de respuesta. Por ejemplo si el datagrama es de un vecino válido, si la dirección IPv6 origen es una dirección de enlace local, si el número de puerto es correcto. Los paquetes que no pasan la validación, son ignorados por el enrutador.
Una vez que los datagramas han sido validados, los RTE´s en el paquete de respuesta son procesaos uno por uno, para volver a ser validados, esta validación ahora es en cuanto a la métrica que sea correcta, entre 1 y 16, que el prefijo presente en el RTE no sea una dirección multicast o una dirección de enlace local, que la longitud del prefijo sea entre 0 y 128. Los paquetes que no pasan la validación, son ignorados por el enrutador.
Una vez que la entrada ha sido validada, actualiza la métrica, agregando el costo de la red sobre la cual el mensaje llegó. Si el resultado es más grande que infinito, usa infinito.
Ahora, verifica si ya hay una ruta para el prefijo RTE, si no la hay, entonces agrega esta ruta a la tabla de ruteo, al menos que la métrica no sea infinita.
Si no hay una ruta existente, compara la dirección de siguiente salto, con la que viene en el RTE, si esta ruta es la misma que la existente, entonces se reinicializa un contador. Después compara la métrica y realiza las actualizaciones correspondientes, si así fuese necesario. Si la métrica es infinita, el proceso de borrado de la ruta se inicia, hay que mencionar que este proceso sólo se da cuando la ruta ya era infinita con anterioridad.

4.3. OSPF PARA IPv6

OSPF para IPv6 modifica al existente OSPF para IPv4 para soportar IPv6. Los fundamentos de OSPF para IPv4 permanecen sin cambios. Algunos cambios han sido necesarios para acomodar el incremento del tamaño de las direcciones de IPv6 y la semántica del protocolo entre IPv4 e IPv6. OSPF para IPv6 es definido en el RFC2740.
OSPF para IPv4 es estandarizado en el RFC2328. En adición a este RFC, algunas extensiones para OSPF han sido definidas. El RFC1584 describe la extensión de OSPF para multicast IPv4. EL RFC1587 agrega "not-so-stubby areas" (NSSAs) para OSPF. EL RFC2740 modifica OSPF para soportar el intercambio de información para IPv6. OSPF para IPv6 tiene un nuevo número de versión: versión 3. Como se puede observar existe mucho sobre OSPF, y como la tesis no esta enfocada a este tema en particular, daremos sólo una pequeña explicación, para mayor referencia sobre el tema dirigirse a los RFC antes mencionados.

OSPF es clasificado como un IGP, el cual es usado dentro de sistemas autónomos. Éste fue diseñado para superar algunas de las limitaciones introducidas por RIP, tales como el diámetro pequeño, largo tiempo de convergencia, y una métrica que no refleja las características de la red. En adición OSPF maneja muchas tablas de ruteo extensas, para acomodar un gran número de rutas.

4.3.1. Diferencias entre OSPF para IPv4 y OSPF para IPv6

La mayoría de los conceptos de OSPF para IPv4 han sido mantenidos; los siguientes son unos cuantos cambios que se hicieron:

- El protocolo procesa por enlace, no por subred: IPv6 conecta interfaces a enlaces. Múltiples subredes de IP pueden ser asignadas hacia un sólo enlace y dos nodos pueden hablar directamente sobre un sólo enlace, aun si ellos no comporten una

subred IP (prefijo IPv6) en común. OSPF para IPv6 corre por-enlace en lugar de por-subred. Los términos "red" y "subred" usados en OSPF para IPv4 deben ser remplazados por el termino "enlace"; por ejemplo, una interfaz OSPF ahora es conectada hacia un enlace en lugar hacia una subred IP.

- Remueve la semántica del direccionamiento: Direcciones IPv6 no están presentes en encabezados de paquetes OSPF. Ellos son sólo permitidos como información de carga útil. Router-LSA y Network-LSA (aun existen) no contienen direcciones IPv6. Los Router ID, area ID, y Link State ID de OSPF permanecen a 32 bits, así ellos no tomaran el valor de una dirección IPv6. Enrutadores designados (Designated Routers "DR") y enrutadores designados de respaldo (Backup Designated Routers "BDR") ahora son identificados por su Router ID y ya no mas por su dirección IP.
- Ámbito de Inundación (Scope Flooding): Cada tipo LSA contiene un código explicito para especificar su ámbito de inundación. Este código es incrustado en el campo tipo de LS (LS type). Tres ámbitos de inundación han sido introducidos: link-local, área, y AS.
- Soporte explicito para múltiples instancias por-enlace: Múltiples instancias del protocolo OSPF ahora pueden correr sobre un sólo enlace. Esto permite que sistemas autónomos separados, cada uno corriendo OSPF, usar un enlace común. Otro uso para esta característica es la de tener un sólo enlace que pertenezca a múltiples áreas.
- Uso de direcciones de enlace-local: OSPF asume que cada interfaz ha sido asignado a una dirección de enlace-local unicast. Todos los paquetes OSPF usan la dirección de enlace-local como la dirección origen. Los enrutadores aprenden las direcciones de enlace-local de todos sus vecinos y usan estas direcciones como la dirección de siguiente salto. Paquetes enviados sobre enlaces virtuales, deben usar la dirección IP global o de sitio-local como el origen de los paquetes OSPF.
- Autenticación: La autenticación ha sido removida de OSPF para IPv6 porque éste confía de la autenticación de IPv6.
- Cambio en el formato de los paquetes:
  - El número de versión ha sido incrementado de 2 a 3.
  - El campo de opciones en el paquete Hello y el paquete de descripción de la base de datos (Database) ha sido expandido a 24 bits.
  - La autenticación y el campo "AuType" ha sido removido del encabezado del paquete OSPF.
  - El paquete "Hello" ahora no contiene ninguna información de todas las direcciones, e incluye un identificador de interfaz (Interface ID) del enrutador que lo origina y que ha sido asignado como único identificador (dentro sus interfaces propias) de sus interfaces hacia el enlace.
  - Se ha agregado dos bits opción, el "R-bit" y el "V6-bit" en el campo opciones para procesar Router-LSA durante el cálculo del SPF.
  - El paquete de encabezado OSPF ahora incluye un "instance ID" el cual permite múltiples instancias del protocolo OSPF correr sobre un sólo enlace.
- Cambio en el formato LSA:
  - Tipo 3 (resumen del enlace) ha sido renombrado "Inter-Area-Prefix-LSA".
  - Tipo 4 (AS resumen del enlace) ha sido renombrado "Inter-Area-Router-LSA".

- Dos nuevos LSA transportan información del prefijo IPv6 en su carga útil. Link-LSA (Tipo 8) transporta la información de las direcciones IPv6 de los enlaces locales, e Inter-Area-Prefix-LSA (tipo 9) transporta el prefijo IPv6 del enrutador y enlace de red.

- Manejo de tipos LSA desconocidos: En lugar de simplemente descartarlos, OSPF para IPv6 introduce una forma más flexible de manejar los tipos LSA desconocidos. Un nuevo bit manejador de LSA ha sido agregado en el campo "LS Type" para permitir la inundación de tipos de LSA desconocidos.
- Soporte de áreas "stub": el concepto de áreas "stub" ha sido retenido en OSPF para IPv6. Una regla adicional especifica la inundación de LSA desconocidos dentro del área "stub".

4.3.2. Principio de funcionamiento de OSPF

Cada enrutador mantiene una base de datos (database) describiendo el estado de los enlaces dentro del sistema autónomo. Esta base de datos es construida por el intercambio de anuncios de estado de enlace (Link State Advertisements "LSA") entre enrutadores vecinos. Dependiendo de su contenido un LSA es inundado hacia todos los enrutadores en el sistema autónomo (inundación de ámbito de AS), todos los enrutadores dentro de la misma área (inundación de ámbito de área), o simplemente hacia sus vecinos. La inundación siempre ocurre a lo largo de un trayecto de enrutadores vecinos, así una relación estable de vecinos es extremadamente importante para que OSPF trabaje apropiadamente. La relación de vecinos es llamada adyacencia.

Cada enrutador origina anuncios de enrutador LSA del estado local de sus interfaces hacia todos los enrutadores dentro de la misma área. Adicionales LSA son originados para identificar enlaces con múltiples enrutadores (redes multiacceso), rutas IPv6 de otras áreas, o rutas IPv6 externas hacia el sistema autónomo OSPF. Cada enrutador pone el LSA recibido dentro su base de dato LSA, llamado la base de datos del estado de enlace (Link State Database "LSDB"). El LSDB es la componente más importante de OSPF.

Usando el LSDB como la entrada, cada enrutador corre el mismo algoritmo para construir un árbol de los trayectos de menor costo (shortest-path-first tree [SPF tree]) para cada ruta. El LSDB es como tener un mapa de la red usado para trazar el camino más corto hacia cada destino. El costo es descrito por una sola métrica adimensional, la cual es configurable sobre cada interfaz del enrutador. La métrica asignada a la interfaz es usualmente inversamente proporcional a su velocidad de la línea, por ejemplo, gran ancho de banda significa bajo costo. Una formula común, de acuerdo al RFC, es la de dividir 10^8 por la velocidad de la línea en bits por segundo. Tu puedes, así escoger tu propia formula de acuerdo a tu estándar corporativo. OSPF puede poner múltiples trayectos de igual costo hacia la misma ruta dentro de la tabla de ruteo. El algoritmo para distribuir tráfico entre estos trayectos hace la discreción, normalmente basado sobre la dirección IPv6 origen y destino.

4.3.3. Paquetes OSPF

Los enrutadores usan paquetes OSPF para intercambiar información LSA y para establecer y mantener relación de vecindad. Paquetes OSPF son directamente encapsulados en IPv6, especificado por el número de protocolo número 89. Este número debe ser insertado en el campo de siguiente cabecera del encabezado IPv6.
Mensajes OSPF normalmente usan la dirección de enlace-local IPv6 de la interfaz de salida o como su dirección de origen en el encabezado. La excepción son mensajes enviados sobre un enlace virtual. Ellos usan la dirección de enlace-local o global unicast del enlace virtual como su origen. Dependiendo de la situación, mensajes OSPF pueden ser enviados como unicast hacia un vecino especificado o como una multicast hacia múltiples vecinos. Las siguientes dos direcciones multicast son usadas, cada una para un propósito diferente:

ALLSPRRouters(FF02::5): Todos los enrutadores OSPF deben escuchar esta dirección multicast. Paquetes Hello son siempre enviados para esta dirección con excepción de redes non-broadcast. Esta dirección también es usada por algunos paquetes durante la inundación de LSA.
ALLDRouters(FF02::6): El DR y el BDR sobre un medio multi-acceso ambos deben escuchar este tipo de direcciones multicast. Este tipo de direcciones multicast es usado por algunos paquetes durante la inundación de LSA.

Hay cinco diferentes tipos de paquetes usados por OSPF. Todos los paquetes OSPF comienzan con el encabezado estándar OSPF de 16 bytes, como lo muestra la figura 101.

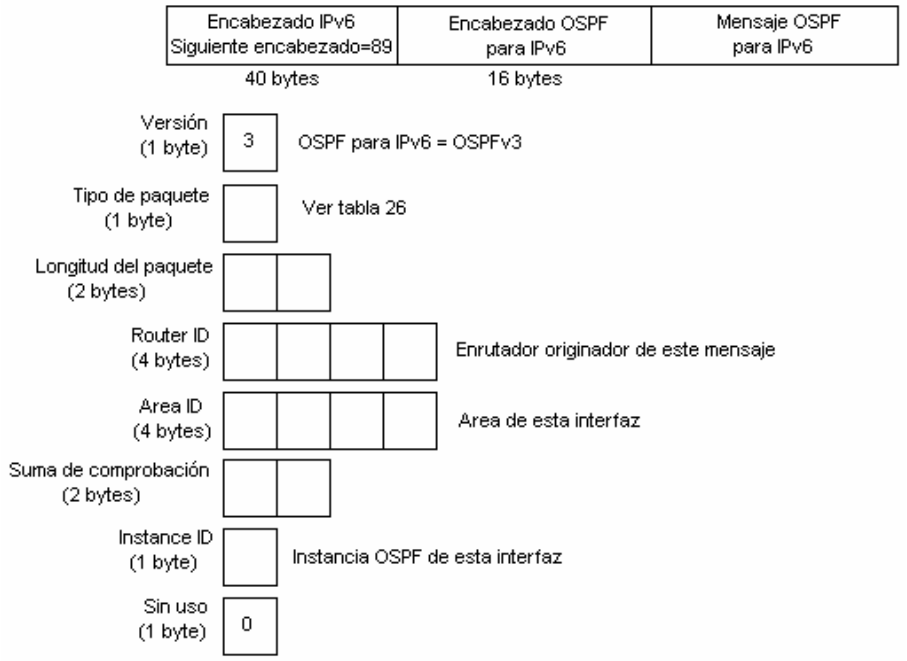


Figura 101. Formato del encabezado OSPF para IPv6

A continuación se describe cada uno de los campos del encabezado OPSF para IPv6:

- Versión (1 byte): OSPF para IPv6 usa el número de versión 3.
- Tipo de paquete (1 byte): Este campo representa el tipo del mensaje OSPF para este enrutador. La tabla 26 lista los posibles tipos.

Tabla 26. Tipos de paquetes OSPF para IPv6

Tipo de paquete	Nombre	Descripción
1	Hola (Hello)	El protocolo Hello es el responsable para inicializar adyacencias, así como la de elegir un DR y BDR. Éste asegura que la comunicación entre dos enrutadores sea bidereccional. Paquete Hello son enviados por cada interfaz en intervalos regulares.
2	Descripción de la base de datos (Database Description)	Estos paquetes son intercambiados cuando una adyacencia ha sido inicializada. Para iniciar el intercambio envían un paquete inicial DD sin datos. Ellos establecen una relación maestro-esclavo para alcanzar un intercambio ordenado. Cada enrutador se declara maestro en el paquete inicial DD. La única información relevante dentro del paquete inicial DD es el número de secuencia del DD usado por cada lado. El enrutador con el mayor Router ID permanece como maestro durante la fase intercambio DD. Los enrutadores ahora se encuentran en un estado de intercambio. De esta forma los enrutadores inician el intercambio de paquetes describiendo el contenido de sus LSDB. Tan pronto ambos enrutadores no tienen más que enviar, los enrutadores entran en un estado de carga.
3	Petición del estado de enlace (Link State Request)	Después del intercambio de paquetes DD con enrutadores vecinos, un enrutador puede encontrar que partes de su LSDB están fuera de fecha (out-of-date). Los paquetes LSR son usados para pedir las piezas de los LSDB vecinos que están mas actualizados (up-of-date). Muchos LSR podrían ser necesitados.
4	Actualización del estado de enlace (Link State Update)	Intercambio de LSA respondiendo a peticiones o cuando forman adyacencia o durante la inundación LSA.
5	Acuse de recibo del estado de enlace (Link State Aknowledgment)	Acuse de recibo de la recepción de un LSA. Cada LSA debe enviar un acuse recibo que fue recibido.

- Longitud del paquete (2 bytes): Este es la longitud del paquete del protocolo OSPF en bytes, incluyendo el encabezado OSPF.
- Router ID (4 bytes): Este es el identificador del enrutador que origina este paquete. Cada enrutador debe tener un único Router ID, un número de 32 bit normalmente representado en representación decimal separados por punto, como una dirección IPv4. El Router ID deber ser único dentro el AS entero.
- Área ID (4 bytes): Este es el identificador de área donde este paquete OSPF fue originado. Éste identifica el área en donde este paquete pertenece. Todos los paquetes OSPF son asociados con una sóla área. El Área ID es un entero de 32 bits, normalmente representado en notación decimal. El área 0 representa el área de backbone.

- Suma de comprobación (checksum (2 bytes): OSPF usa la suma de comprobación estándar para aplicaciones IPv6.
- Instancia ID (1 byte): Este identifica la instancia OSPF de la cual el paquete OSPF pertenece. La instancia ID es un número de 8 bits asignado a cada interfaz del enrutador. El valor por default es 0. La instancia ID habilita a múltiples instancias del protocolo OSPF para correr sobre un sólo enlace. Si el receptor no reconoce la instancia ID, ese descarta el paquete.

4.3.3.1. Tipos de anuncios de estado de enlace (Link State)

Los anuncios de estado de enlace (Link State Advertisements LSA) son principalmente los que originan, calculan y mantienen la información de ruteo. El RFC2740 define siete tipos de LSA para OSPFv3 como se listan a continuación:

- Router-LSAs: originado por cada enrutador. La colección completa de router-LSA originado por el enrutador describe el estado y costo de las interfaces del enrutador hacia el área, y es solamente transmitido en el área donde reside el enrutador.
- Network-LSAs: originados por los DR (Designated Router) en las redes Broadcast y NBMA. Cada Network-LSA describe todos los enrutadores unidos al enlace, incluyendo al mismo DR, y es solamente transmitido en el área donde reside el DR.
- Inter-Area-Prefix LSA: similar al LSA tipo 3 de OSPFv2, son originados por los enrutadores de frontera de área (Area Border Router ABR) y son transmitidos al área que pertenecen. Cada inter-area-prefix LSA describe un prefijo externo al área, siendo aún internos al sistema autónomo (AS).
- Inter-Area-Router LSA: similar al LSA tipo 4 de OSPFv2, son originados por los ABR. Cada inter-area-router-LSA describe un trayecto hacia un enrutador OSPF destino (un AS boundary routers ASBR) que es externo al área, pero aún interno al sistema autónomo.
- AS-external-LSA: originado por ASBR. Cada AS-external-LSA describe un trayecto hacia un prefijo externo del sistema autónomo y es transmitido através de AS enteros (excluyendo áreas stub). Los enrutadores por default de un AS puede también ser descrito por un AS-external-LSA. Enrutador por default son usados cuando no existe ruta específica hacia el destino.
- Link LSA: los enrutadores originan un Link-LSA por cada enlace y los Link-LSA son transmitidos en el ámbito de enlace-local. Cada Link-LSA describe el prefijo de la dirección IPv6 asociado con este enlace, incluyendo la dirección de enlace local.
- Intra-Area-Prefix LSA: un enrutador usa Intra-Area-Prefix-LSA para anunciar uno o más prefijos de direcciones IPv6 que están asociados con A) el mismo enrutador, B) un segmento de red “stub” unido o C) un segmento de red de tránsito unido. Como OSPF para IPv6 removió toda la semántica del direccionamiento de Router-LSA y Network-LSA, el Intra-Area-Prefix-LSA provee toda esta información.

A continuación la tabla 27 muestra más características de los tipos de LSA.

Tabla 27. Tipos de Estado de enlaces (Link State)

Tipo LS	Nombre	Ámbito de inundación (flooding)	Anunciado por
0x2001	Router-LSA	Área	Cada enrutador
0x2002	Network-LSA	Área	DR
0x2003	Inter-Area-Prefix-LSA	Área	ABR
0x2004	Inter-Area-Router-LSA	Área	ABR
0x4005	AS-External-LSA	AS	ASBR
0x0008	Link-LSA	Enlace	Cada enrutador por cada enlace
0x2009	Intra-Area-Prefix-LSA	Área	Cada enrutador

4.3.4. Vecinos y Adyacencias

En OSPF, los conceptos de vecinos y adyacencias son completamente diferentes.

Después de que un enrutador OSPF inicia, envía paquetes tipo “Hello” hacia afuera de sus interfaces. El enrutador OSPF que recibe estos paquetes, verifica algunos parámetros definidos en los paquetes. Si los parámetros de ambos enrutadores son consistentes, la relación de vecino puede ser establecida.
La relación de vecino entre los “peers” (pares) no es equivalente a la relación de adyacencia. El orden para intercambiar LSAs, el enrutador debe crear canales de confianza hacia sus vecinos, llamado adyacencia. Estos canales permiten a los enrutadores sincronizar los LSDB durante la inicialización e inundar de LSA en caso de cambios.

Los vecinos necesitan primero ser descubiertos. Esto es hecho realizando el paquete “Hello”. Cada interfaz sobre un enrutador OSPF es asignado a uno de cuatro tipos de enlaces: punto-a-punto, tránsito, stub, o virtual. En un punto-a-punto o enlace virtual, sólo un vecino puede ser descubierto. Sobre redes multi-acceso múltiples vecinos pueden ser descubiertos. OSPF llama a estas redes enlaces de tránsito. Formar adyacencias con todos los enrutadores sobre los enlaces de tránsito no es necesario. Cada enlace de tránsito elige un DR (Designated Router) para formar adyacencias con todos los enrutadores sobre el enlace de tránsito. Esto garantiza que todos los enrutadores sobre este enlace tienen sincronizado su LSDB. Para asegurar ininterrumpida operación, un BDR (Backup Designated Router) es elegido también; éste forma adyacencia con todos los enrutadores sobre el enlace de tránsito, también. La figura 102 muestra adyacencia sobre enlaces punto-a-punto y enlaces de tránsito. Si un vecino no es descubierto sobre un enlace dado, el enlace es declarado como un enlace “stub”, y obviamente la adyacencia no es formada sobre tal enlace.

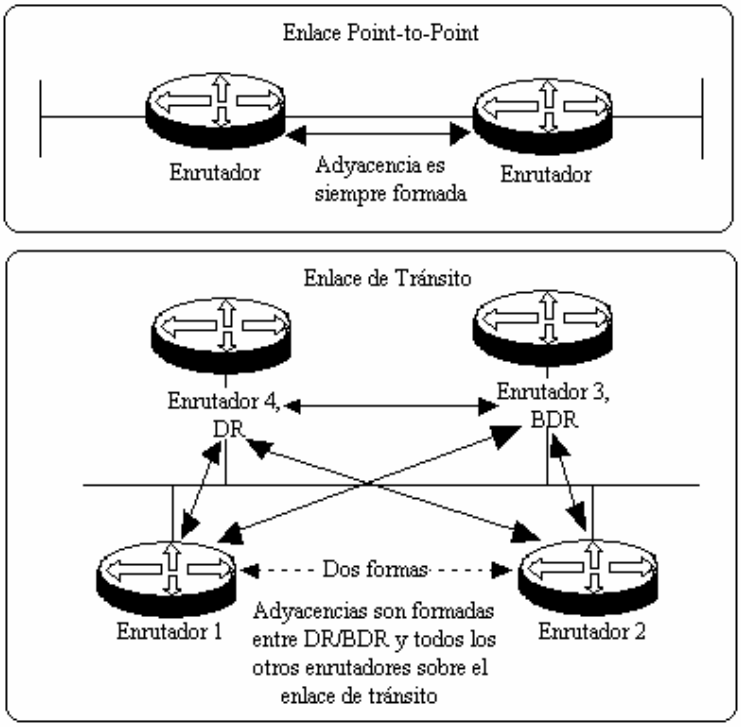


Figura 102. Adyacencias sobres enlaces punto-a-punto y enlace de tránsito.

4.3.5. Áreas OSPF y Rutas Externas

Dentro un sistema autónomo, los enrutadores pueden ser agrupados para formar áreas. Cada área es asignada a un único Área ID, un entero de 32 bits, típicamente en notación decimal separado por puntos (el backbone es el área 0). Este no significa otra cosa más que el identificador de área. Un LSA con ámbito de inundación de área, nunca inundará fuera de esta área. Juntos, ellos forman la estructura de datos del área, también conocido como área LSDB. El Router-LSA y Network-LSA pertenecen a esta categoría. Enrutadores y redes de un área, son ocultos para otras áreas. Esto es como fracturar el mapa de la red dentro de múltiples mapas, cada uno de los cuales representa una topología de un área. Cada enrutador dentro de un área, calcula el árbol SPF (SPF Tree) para todas las rutas dentro de la misma área. Estas rutas son llamadas "intra-area-routes". Enrutadores con todas las interfaces perteneciendo a una sola área son llamados enrutadores internos (internal Routers). Para encontrar caminos hacia rutas fuera del área,"puntos de salida" (exit point) son proveídos en la forma de enrutadores de frontera de área (area border routers ABR). Cada área debe ser siempre unida a una única área en común llamada área de backbone. Esta es alcanzada por el ABR, teniendo al menos una interfaz hacia el área de backbone y una interfaz hacia el área local. El ABR anuncia todas las rutas del área local hacia el área de backbone. Este de vuelta, anuncia todas las rutas del area de backbone hacia el àrea local. Estos aseguran que todas las rutas son distribuidas dentro del AS.

El ruteo dentro del AS toma lugar en dos niveles. Si la dirección IP origen y destino de un paquete pertenecen a la misma àrea, el paquete es enviado solamente con la información obtenida del area LSDB. Esto es llamado "intra-area-routing". Si la dirección destino està

fuera del àrea, el paquete tendrá que ser enviado hacia el ABR del àrea local. EL ABR conoce todos los destinos y envía el paquete, cruzando el backbone hacia el ABR del àrea destino o hacia el àrea de Backbone. Esto es llamado "inter-area-routing".

La ventaja de tener áreas, se nota en la reducción de procesamiento. Porque la topología de cada àrea es más pequeña que el AS entero, el cálculo del árbol SPF toma menos tiempo. En adición, cambios en la topología permanecen locales, y solamente los enrutadores en el àrea local necesitaran recalcular el árbol SPF. Enrutadores en otras áreas no son afectados porque su topología de su àrea no fue cambiada. El mayor beneficio en los enrutadores internos es por la fractura del AS en áreas, porque sus LSDB son mucho más pequeñas.

4.3.5.1.El àrea de Backbone

El àrea de backbone es una àrea especial usando Àrea ID 0.0.0.0 (area 0). El àrea de backbone contienen todos los ABR del sistema autónomo. Si el AS no es dividido en áreas, el àrea de backbone es usualmente la única area configurada. Si el sistema autónomo es dividido en áreas, el area de backbone es la colección de todos los enrutadores de todas las áreas que no son del backbone (non-backbone). El àrea de backbone debe ser contiguo: cada enrutador dentro del mismo àrea tiene al menos un enlace directo hacia otro enrutador en el mismo àrea, y tal enlace pertenece al àrea. Sin embargo, con el concepto de enlaces virtuales, un àrea de backbone no tiene que ser físicamente contigua. Un àrea de trànsito puede ser usada para crear un túnel (un enlace virtual) que pertenezca al àrea de backbone.

4.3.5.2.Áreas que no son de Backbone (Non-Backbone)

Áreas que no son de backbone reciben un único Àrea ID diferente a 0.0.0.0. Ellas deben ser físicamente contiguas. Cada àrea que no es de backbone debe tener un ABR conectado hacia el backbone, usando un enlace físico o un enlace virtual. Un ABR anuncia todas las rutas del àrea que no es de backbone dentro del àrea de backbone. De forma contraria, un ABR anuncia todas las rutas conocidas en el àrea de backbone hacia el àrea que no es de backbone. Normalmente, el ABR usa un LSA (llamado Inter-Area-Prefix-LSA) por cada ruta anunciada. El ABR puede ser configurado para resumir (summarize) rutas usando un prefijo IPv6 màs corto, el cual representa algunas o todas las rutas a ser anunciadas. Esto reduce el número de anuncios, memoria y procesos requerido. Esto es muy importante para planear la asignación de prefijos IPv6 dentro del àrea, para alcanzar mayores beneficios del resumen (summary). Un àrea que no es de backbone puede tener múltiples ABR. La figura 102 muestra anuncios Inter-Area-Prefix-LSA de ABR.

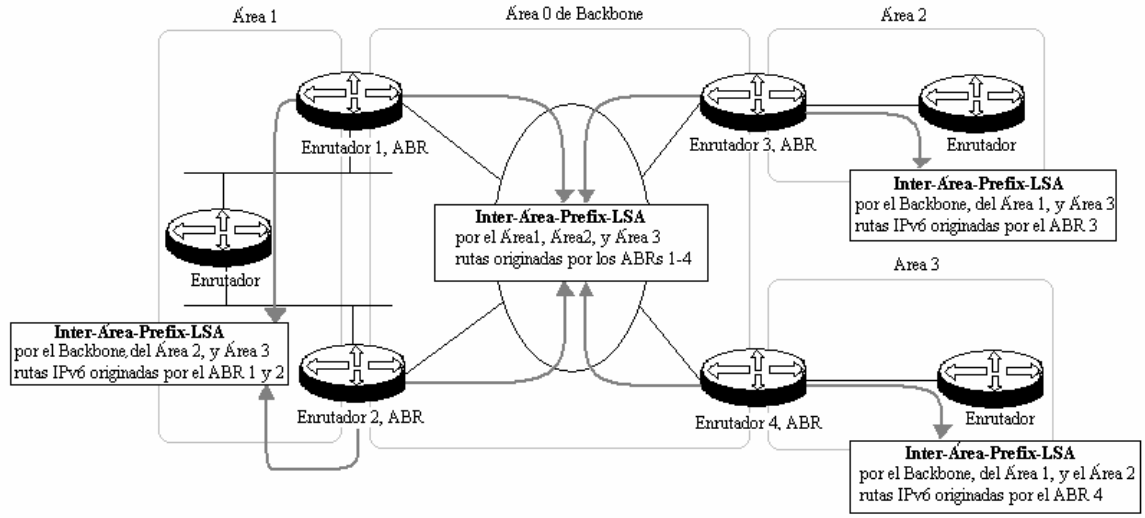


Figura 103. Áreas OSPF y sus actualizaciones de ruteo.

4.3.5.3.Enlaces Virtuales (Virtual Links)

Un enlace virtual, es un enlace lógico que tunelea el tráfico de backbone a través de un àrea que no es de backbone. Este puede ser configurado entre dos ABR, en un àrea común que no es de backbone, llamado àrea de tránsito. El àrea de tránsito no debe ser un àrea "stub". Un àrea remota sin una interfaz física hacia el àrea de backbone puede ser conectado al backbone utilizando enlaces virtuales. Enlaces virtuales pueden también ser usados para crear conexiones redundantes hacia el backbone. OSPF considera un enlace virtual a un enlace punto-a-punto. El camino màs corto entre los ABR a través del àrea de tránsito se determina por la dirección actual del punto final del túnel. Estas direcciones deben ser direcciones unicast IPv6 globales o de sitio-local. La figura 104 muestra en ejemplo de enlace virtual.

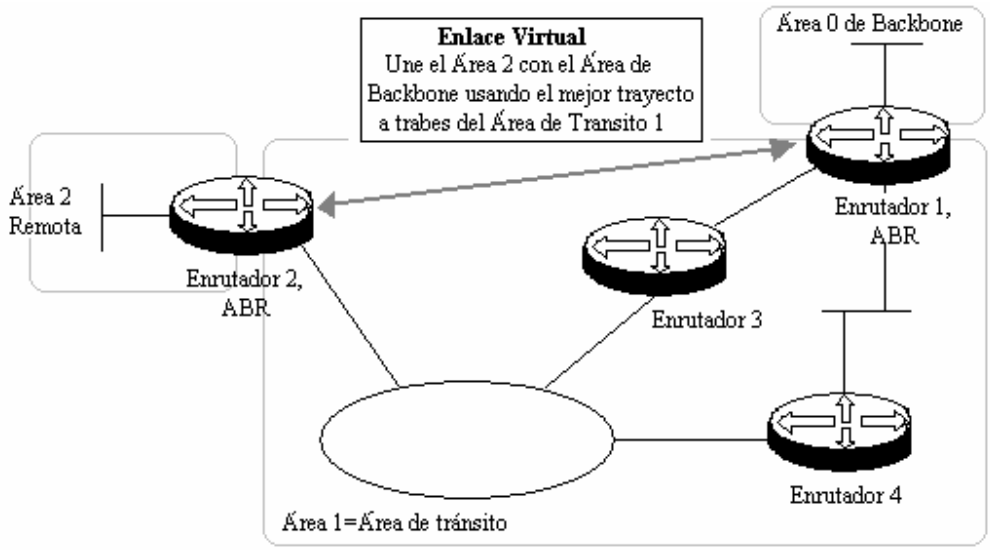


Figura 104. Conexión de enlace virtual de un area remota.

4.3.5.4.Rutas Externas

Un enrutador puede aprender rutas IPv6 por medio de diferentes orígenes, tal como RIP, entradas estáticas, BGP, ISIS, etc. Cada ruta que no es de origen OSPF, es considerada como una ruta externa OSPF y puede ser importada dentro de OSPF. Para importar rutas externas dentro de OSPF, un enrutador debe tener por lo menos una interfaz configurada con OSPF y conocer por lo menos una red que no sea OSPF. Este enrutador es llamado enrutador de frontera de sistema autónomo (Autonomous System Border Router ASBR). Rutas externas son importadas usando un sòlo AS-external-LSA por cada ruta externa. Dependiendo de la implementación, un ASBR puede resumir (summarize) un rango de rutas externas en un sòlo LSA externo. La figura 105 muestra còmo rutas externas son importadas dentro de OSPF.

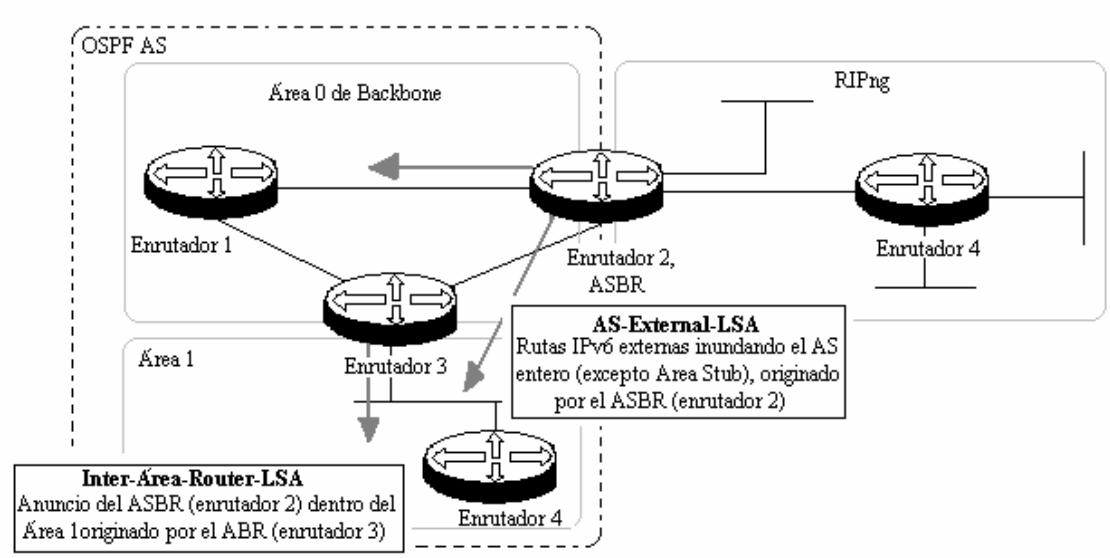


Figura 105. Rutas externas importadas dentro de OSPF.

AS-external-LSA deber ser inundado a través del sistema autónomo. Cualquier enrutador dentro del AS enviarà paquetes hacia redes externas por el ASBR o por una dirección opcional especificada por el ASBR.

Métricas de rutas externas no son compatibles con las métricas de OSPF. ASBR anuncia rutas externas usando una de dos tipos, rutas external-1 y external-2. Rutas external-1 son consideradas que están cerca del ASBR. Enrutadores dentro del AS agrega el costo OSPF para alcanzar el ASBR. Rutas external-2 se asumen que están más lejos del ASBR. La métrica màs larga en cuanto a costo de cualquier trayecto intra-AS, será agregada a la métrica de la ruta external-2.

4.3.5.5.Áreas Stub.

En OSPF para IPv4, áreas stub fueron diseñadas para minimizar el LSDB y el tamaño de las tablas de ruteo para los enrutadores internos de una àrea.

Un àrea stub es una zona libre de AS-External-LSA. Estas LSA normalmente podrían inundar a través del AS entero, lo cual podría resultar en un completo gran LSDB consistiendo de muchos anuncios externos. Para reducir el tamaño del LSDB, un ABR puede bloquear AS-external-LSA dentro del àrea local. Como el ABR priva al àrea de conocer las rutas externas, èste debe compensarlo anunciando una ruta de reemplazo en la forma de una ruta por default. Èsta usa los Inter-Area-Prefix-LSA, anunciando el prefijo 0:0:0:0:0:0:0:0 con una longitud de prefijo de 0. La métrica asociada con la ruta por default es llamada la métrica stub. Si hay múltiples ABRs por esa àrea, cada ABR bloquea los AS-external-LSA y las remplaza con una ruta por default. Enrutadores internos del àrea calculan el mejor camino de la ruta por default, agregando la métrica para obtener el ABR (de acuerdo al SPF) por la métrica stub. Hay algunas restricciones para las áreas stub. Ellas nunca deben ser configuradas como áreas de trànsito para enlaces virtuales. En adición, ASBR no pueden ser puestos en un àrea stub, porque los enrutadores en un àrea stub no pueden importar información externa. El àrea de backbone nunca puede ser una àrea stub. La figura 106 da un ejemplo de un àrea stub.

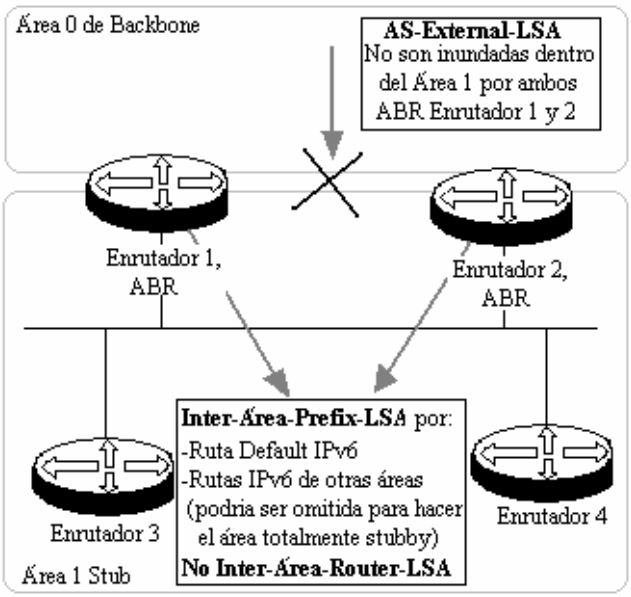


Figura 106. Área Stub

4.3.5.6.Áreas No-so-stubby

Hay casos cuando áreas stub necesitan conectarse a enrutadores que no tienen rutas OSPF. El implementarlo no quiere decir que se quiere invertir el àrea hacia un àrea normal para permitir esas rutas externas a ser importadas. Porque los AS-external-LSA no son permitidos en un àrea stub, por lo tanto se diseñò un nuevo tipo de LSA llamado Type-7-LSA . El Type-7-LSA es exactamente como As-external-LSA, pero estos pueden existir en áreas stub. Áreas Stub en donde existen Type-7-LSA son referidas como Not-So-Stubby Áreas, o NSSA. Type-7-LSA son inundados sòlo dentro del NSSA. Un NSSA ASBR usa un Type-7-LSA por ruta externa. ABR del NSSA pueden trasladar Type-7-LSA dentro de AS-external-LSA para anunciarlos al resto del AS. En adición, estos ABR pueden

comportarse como ABR en un àrea stub y anunciar una ruta por default usando un Type-7-LSA en este caso.

4.3.6. Tipos de redes OSPF

OSPF divide las redes dentro de cuatro tipos por su protocolo de capa de enlace:

- Broadcast: Si el protocolo de capa de enlace es Ethernet o FDD, el tipo de red OSPF por default es broadcast. Los paquetes del protocolo en esta red son transmitidos en modo multicast (224.0.0.5 y 224.0.0.6).
- Non-Broadcast Multi-access (NBMA): si el protocolo de la capa de enlace es Frame Relay, ATM o X25, el tipo de red OSPF por default es NBMA. Los paquetes de protocolo son transmitidos en modo unicast.
- Point-to-Multipoint (P2MP): No importa què protocolo de capa de enlace es. Una red P2MP deber ser cambiada a otro tipo de red obligatoriamente. Los paquetes del protocolo en esta red son transmitidos en modo multicast (224.0.0.5).
- Point-to-Point (P2P): si el protocolo de capa de enlace es PPP, HDLC o LAPB, el tipo de red OSPF por default es P2P. Los paquetes del protocolo en esta red son transmitidos en modo multicast (224.0.0.5).

4.3.6.1. Principios de las redes NBMA

La red NBMA se refiere a non-broadcast y red multi-acceso. Redes ATM y Frame Relay son típicamente redes NBMA. Se necesita hacer algunas configuraciones especiales para las redes NBMA. La adyacencia de sus enrutadores no puede ser descubiertos por paquetes Hello broadcasting. Entonces para esta interfaz, se debe configurar manualmente la dirección IP de sus enrutadores adyacentes. La red NBMA deber ser completamente conectada, y dos enrutadores de la red deben ser alcanzables directamente. Si todos los enrutadores no son directamente accesibles en una red NBMA, se puede configurar la interfaz tipo P2MP. Si el enrutador tiene solamente un peer en la red NBMA, se puede cambiar el tipo de interfaz a P2P.

La diferencia entre redes NBMA y P2MP son las siguientes:

- En OSPF, una red NBMA se refiere a una red que es completamente conectada, non-broadcast y multi-accesible; mientras una red P2MP no necesariamente tiene que estar completamente conectada.
- Una red NBMA necesita elegir DR y BDR; mientras una red P2MP no tiene DR y BDR.
- NBMA es un tipo de red por default. Una red P2MP deber ser cambiada obligatoriamente a otro tipo de red. Lo común en la práctica es cambiar una red no completamente conectada NBMA en una red P2MP.
- NBMA transmite los paquetes por unicast y los vecinos necesitan ser configurados manualmente; mientras P2MP transmite paquetes por multicast.

4.3.6.2.DR y BDR

En redes broadcast y NBMA, información de ruteo es transmitida entre dos enrutadores cualquiera. Si hay n enrutadores en la red, n X (n-1)/2 adyacencias necesitan ser establecidas. En este caso, el enrutador cambia resultados en múltiples transmisiones cuando son innecesarias. Esto crea basura en nuestros recursos de ancho de banda. Designated Router (DR) es definido por OSPF para resolver estos problemas. Todos los enrutadores envían información sòlo al DR por broadcasting del estado de los enlaces de la red.

Si el enrutador llega ser invàlido por algùn problema, este debe ser reelegido y sincronizado. Esto lleva mucho tiempo y mientras tanto el cálculo de rutas llega a ser incorrecto. Para evitar este procesamiento, OSPF maneja el concepto de Backup Designated Router (BDR).
El BDR es un respaldo para DR. Tan pronto como IPv6 sobre una interfaz OSPF es operacional, el enlace es levantado, y el procesamiento de paquetes Hello es iniciado, un enlace de tràsito espera el estado para descubrir el DR/BDR. Las adyacencias son también establecidas entre el BDR y todos los enrutadores en el segmento. Ellos también intercambian información de ruteo. Cuando el DR llega a ser invàlido, el BDR se convierte instantáneamente en DR. Como la reelección no es necesitada y las adyacencias ya han sido establecidas, el proceso es muy rápido. La elección de un nuevo BDR toma un largo tiempo pero esto no ejerce ninguna influencia en el càlculo de rutas.
Adyacencias no son establecidas entre otros dos enrutadores que no sean DR o BDR (estos son llamados como DR others), DR others no intercambian información de ruteo. Así, esto reduce el número de adyacencias entre dos enrutadores cualquiera en redes broadcast o NBMA.
En la figura 107, las líneas normales representan las conexiones físicas Ethernet, y las líneas punteadas representan las adyacencias establecidas. Ùnicamente siete adyacencias son necesarias entre cinco enrutadores con el mecanismo de DR/BDR.

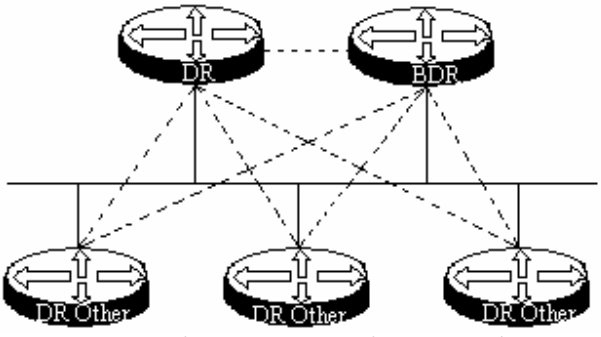


Figura 107. Diagrama esquemático para DR/BDR

4.3.6.3.Proceso de elección para DR/BDR

El DR no es designado manualmente. Èste es elegido por todos los enrutadores en el segmento de red. La prioridad de la interfaz de un DR determina su calificación para ser elegido DR/BDR. En el segmento de red, enrutadores con prioridad mayor de 0 son los elegidos “candidatos”.

Votos son los paquetes Hello. Cada enrutador pone el DR elegido en el paquete y lo envía a todos los demás enrutadores en el segmento. Entre todos los enrutadores se proclaman ser el DR, el enrutador con la mayor prioridad será el elegido. Si dos enrutadores tienen la misma prioridad, el que tenga más grande su Router ID será el elegido como el DR. Un enrutador cuya prioridad sea 0, no puede ser elegido como un DR o BDR. El BDR es elegido de la misma forma. Enrutadores que no fueron elegidos como DR/BDR son llamados otros DR (DR-other).

Note lo siguiente:

- Solamente cuando una interfaz es de tipo broadcast o NBMA, es necesario elegir DR. Una interfaz P2MP o P2P no necesitan elegir DR.
- Un enrutador actúa como un DR sobre un cierto segmento de red, en el sentido de su interfaz de red. Quizás un enrutador es un DR en una interfaz, pero puede ser un BDR o DR-Other sobre la otra interfaz.
- Si un nuevo enrutador es agregado después de la elección del DR y BDR, es imposible para el enrutador llegar a ser DR aun si su prioridad es la mayor.
- El DR en la red no necesariamente tiene que ser el enrutador con la mayor prioridad. Asimismo, el BDR no necesariamente tiene que ser el enrutador con segunda prioridad más alta.

4.4. IS-IS

El RFC1195 especifica un protocolo de ruteo integrado, basándose del protocolo de ruteo OSI Intra-Domain IS-IS, el cual puede ser usado como un protocolo IGP para soportar TCP/IP así como a OSI. Esto permite que un sòlo protocolo de ruteo pueda ser usado para soportar puros ambientes IP, puros ambientes OSI, y ambientes duales. Este tipo de IS-IS es llamado IS-IS integrado o Dual IS-IS.

Como un protocolo IGP, IS-IS es usado dentro de sistemas autónomos (AS). IS-IS es un protocolo de estado de enlace (link-state). Èste usa el algoritmo SPF para el cálculo de las rutas. Este protocolo tiene una gran semejanza con el protocolo OSPF.

4.4.1. Conceptos Básicos

4.4.1.1. Términos del protocolo de ruteo IS-IS

- Sistema Intermedio (Intermediate System “IS”): Igual a un enrutador TCP/IP. Èste es la unidad bàsica en el protocolo IS-IS. Èste es usado para transmitir información de ruteo y generar rutas. En lo siguiente, “IS” tiene el mismo significado como el de “enrutador”.
- Sistema Final (End System “ES”): Igual que un host en un sistema TCP/IP. Estos no están envueltos en el proceso IS-IS. ISO ha dedicado el protocolo ES-IS para definir la comunicación entre un ES y un IS.
- Dominio de Ruteo (Routing Domain “RD”): Un grupo de IS intercambiando información de ruteo a travès del mismo protocolo de ruteo en un dominio de ruteo.
- Àrea: Es la división unidad en el dominio de ruteo.

- Base de Datos de Estado de Enlace (Link State Database “LSDB”): Todos los estados de enlace (Link-State) en la red forman el LSDB. En un IS, al menos un LSDB es disponible. EL IS usa el algoritmo SPF y el LSDB para generar sus propias rutas.
- Unidad de Datos del Protocolo de Estado de Enlace (Link State Protocol Data Unit “LSP”): En IS-IS, cada IS genera un LSP el cual contiene toda la información del estado de enlace del IS. Cada IS recolectà todos los LSP del àrea local para generar sus propios LSDB.
- Unidad de Datos del Protocolo de Red (Network Protocol Data Unit “NPDU”): Estos son los paquetes de la capa de red de ISO e iguales al paquete IP de TCP/IP.
- IS Designado (Designated IS “DIS”): Un enrutador elegido en una red broadcast. Èste puede ser también un DR.
- Punto de Acceso del Servicio de Red (Network Service Access Point “NSAP”): Una dirección de capa de red de ISO. Èste define un punto de acceso de servicio de red abstracto y describe la estructura de la dirección de red del modelo OSI.

4.4.1.2.Estructura de la dirección del protocolo IS-IS.

1. NSAP

ISO adopta la estructura de la dirección NSAP como se muestra en la figura 108. NSAP consiste de la parte inicial del dominio (Inicial Domain Part “IDP”) y de la parte específica del dominio (Domain Specific Part “DSP”).
IDP es equivalente al número principal de la dirección IP, y DSP el número de subred y dirección de host de la dirección IP.
Estipulado en ISO, el IDP consiste de la Autoridad y formato identificador (Authority and Format Identifier “AFI”), y el identificador inicial de dominio (Initial Domain Identifier “IDI”). El AFI especifica el mecanismo de asignación de direcciones y el formato de dirección. El IDI es usado para identificar un dominio.
El DSP consiste del DSP de mayor orden (High Order DSP “HODSP”), identificador del sistema (System ID) y SEL. HODSP es para partición de áreas, System ID identifica un host y SEL identifica el tipo de servicio.
La longitud del IDP y DSP son variables. Su máxima longitud total es de 20 bytes y mínimo de 8 bytes.

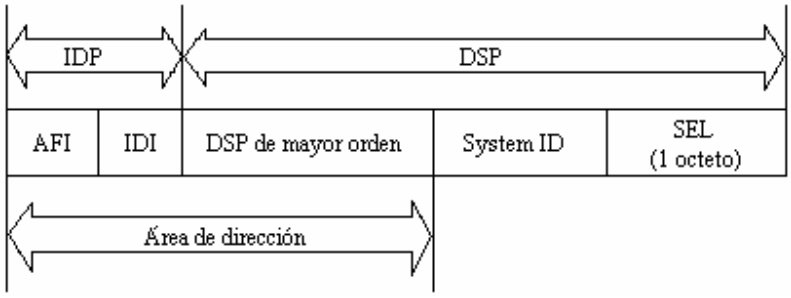


Figura 108. Diagrama esquematizo de la estructura de la dirección de IS-IS.

2. Dirección de àrea

IDP, con HODSP de DSP, pueden identificar un dominio de ruteo y áreas en un dominio de ruteo. La combinación de IDP y HODSP son llamados como dirección de àrea. Esto es equivalente al àrea ID de OSPF. Las direcciones de àreas varían con las áreas.
Generalmente, un enrutador necesita ser configurado con una dirección de àrea solamente. Por otra parte, todos los nodos en la mismà àrea deben tener la misma dirección de àrea. Para soportar combinación, división y transformación de áreas, un enrutador puede ser configurado con un máximo de tres direcciones de áreas.

3. Identificador del sistema (System ID)

Un identificador de sistema es usado únicamente para identificar un host o un enrutador en un àrea. Su longitud invariable es de 48 bits (6 bytes).
Normalmente, un Router ID es adoptado para corresponder con una System ID. Supongamos a un enrutador que toma la dirección IP 168.10.1.1 de la interfaz de Loopbaack0 como su Router ID. Entonces sus System ID usado en IS-IS puede ser transformado con el siguiente método:

- Extender cada parte de la dirección IP 168.10.1.1 a tres dígitos. Agregar 0 a la izquierda a cada parte que tenga menos de tres bits.
- Dividir la dirección extendida 168.010.001.001 en tres partes, con cada parte consistiendo de cuatro dígitos decimales.
- La reconstruida 1680.1000.1001 es el System ID.

Actualmente, hay muchas formas de designar un System ID, de forma que estos solamente puedan identificar un ES o un enrutador.

4. SEL

El rol de un SEL (NSAP Selector, o N-SEL como es llamado a veces) es similar el identificador de protocolo en IPv4 o de siguiente cabecera en IPv6. Diferentes protocolos de transporte corresponden a diferentes SEL. El SEL sobre IP esta dentro de “00”.

5. Ruteo (Routing)

Como este tipo de estructura de direcciones definitivamente define un àrea, enrutadores nivel-1 (Level-1) pueden fácilmente identificar paquetes enviados hacia un àrea externa. Estos paquetes serán enviados hacia enrutadores nivel-2 (Level-2).

Enrutadores nivel-1 usan System ID para implementar ruteo intra-àrea. Una vez que un enrutador nivel-1 encuentra que la dirección destino del paquete està fuera de su propia àrea, este lo enviarà al enrutador nivel-1-2 màs cercano.
Basándose en sus direcciones IDP, HODSP, enrutadores nivel-2 realizan ruteo inter-àrea.

4.4.1.3.Titulo de la Entidad de la Red (Network Entity Title)

Titulo de la Entidad de la Red (NET) indica la información de capa de red de un IS. Èste excluye información de la capa de transporte (SEL=0). Èsta puede ser vista como una especial NSAP. Por lo tanto, la longitud de NET es la misma como la de NSAP. Èsta puede ser 20 bytes de largo a lo máximo y como mínimo 8 bytes. Cuando configuramos IS-IS en un enrutador, el usuario solamente puede considerar NET, en vez de NSAP.
Generalmente, un enrutador es solamente configurado con un NET. Cuando es necesario reconstruir un àrea, múltiples NET deben ser configurados en un enrutador. Por ejemplo, para combinar muchas áreas o para dividir un àrea en áreas separadas, se necesitara configurar múltiples NET.
Se puede configurar màximo tres àreas en un proceso IS-IS de un enrutador, tres NET pueden ser configurados a lo máximo. Cuando configuramos múltiples NET, debemos estar seguros que sus System ID son idénticos.

4.4.2. Áreas IS-IS

1. Estructura dos-niveles

Para soportar gran escala en redes de ruteo, IS-IS adopto una estructura de dos niveles en un dominio de ruteo. Un gran dominio de ruteo es dividido en una o más áreas. Ruteo Intra-àrea es manejado por enrutadores de nivel-1 (Level-1), mientras ruteo Inter-àrea es manejado por enrutadores de nivel-2 (Level-2).

2. Nivel-1 y Nivel-2 (Level-1 y Level-2)

   - **Enrutador Nivel-1:** Un enrutador nivel-1 maneja el ruteo intra-àrea. Èste establece la relación de vecindad solamente con enrutadores Nivel-1 y Nivel-1-2 en la misma área. Èste mantiene un LSDB Nivel-1. El LSDB contiene la información de ruteo del área local. Un paquete hacia un destino fuera de esta área, èste es enviado al enrutador más cercano Nivel-1-2.
   - **Enrutador Nivel-2:** Un enrutador nivel-2 maneja el ruteo Inter-àrea. Estos pueden formar la relación de vecindad con enrutadores de Nivel-2 y enrutadores Nivel-1-2 de otras áreas. Estos mantienen un LSDB Nivel-2. El LSDB contiene información de ruteo entre las áreas. Todos los enrutadores de Nivel-2 forman la red de Backbone de las diferentes áreas. Los enrutadores de Nivel-2 en el dominio de ruteo deben estar en sucesión para asegurar la continuidad de la red de backbone. Solamente enrutadores de Nivel-2 pueden intercambiar los paquetes de datos o la información de ruteo con enrutadores fuera del dominio de ruteo.
   - **Enrutadores Nivel-1-2:** Un enrutador el cual pertenece a ambas áreas Nivel-1 y Nivel-2, son llamados enrutadores de Nivel-1-2. Estos pueden formar relación de vecindad Nivel-1 con enrutadores Nivel-1 y enrutadores Nivel-1-2 en la misma área. En adición, estos pueden formar relación de vecindad Nivel-2 con enrutadores Nivel-2 y enrutadores Nivel-1-2 en otras áreas. Un enrutador Nivel-1 debe ser conectado a otras áreas a travès de un

enrutador Nivel-1-2. Un enrutador Nivel-1-2 mantiene dos LSDB. El LSDB Nivel-1 es usado para ruteo Intra-àrea y el LSDB Nivel-2 para ruteo Inter-àrea.

La figura 109 ilustra una red con IS-IS habilitado, similar a una topología OSPF con múltiples áreas. El Àrea 1 es el àrea de backbone. Todos los enrutadores en esta àrea son enrutadores Nivel-2. Las otras cuatro áreas son áreas de non-backbone. Ellos son conectados al àrea 1 a través de los enrutadores Nivel-1-2.

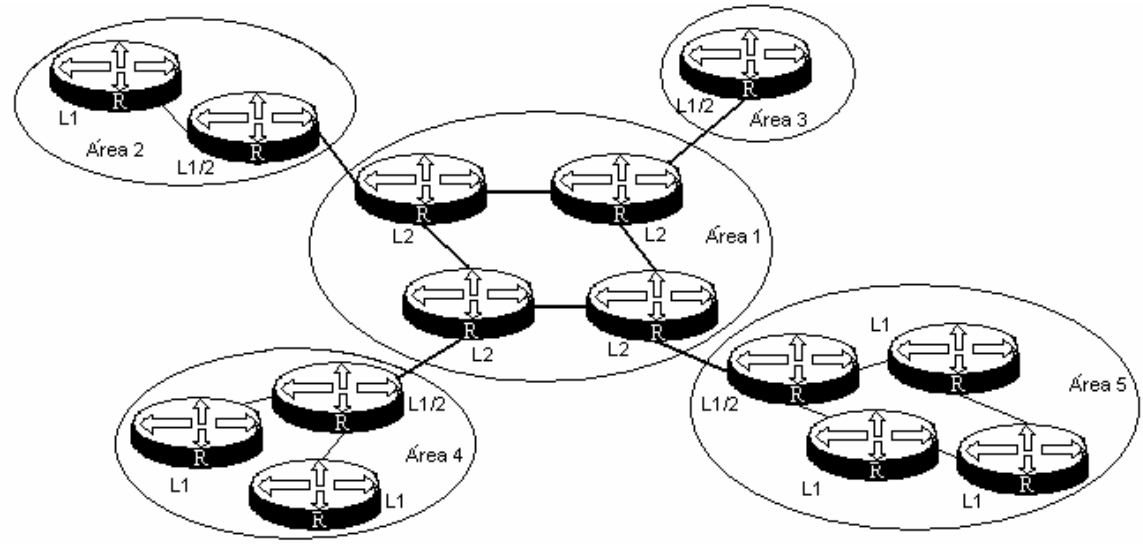


Figura 109. Topología I IS-IS.

La figura 110 muestra otro tipo de topología IS-IS. Los enrutadores Nivel-1-2 no son solamente usados para conectar enrutadores Nivel-1 y Nivel-2, forman también la red de backbone con otros dos enrutadores Nivel-2. En esta topología, las áreas no son definidas como áreas de backbone. La red de backbone contiene todos los enrutadores Nivel-2. Ellos pueden pertenecer a diferentes áreas, pero deben ser sucesivos. La red de backbone IS-IS no indica un àrea específica.

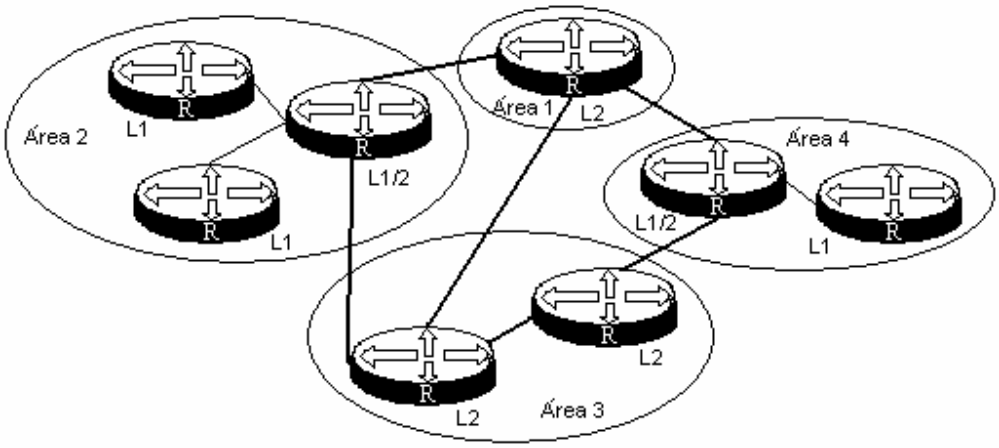


Figura 110. Topología II IS-IS.

Este tipo de esquema de redes muestra la diferencia entre IS-IS y OSPF. Para OSPF, las rutas inter-àrea son enviadas por el àrea de backbone, y el algoritmo SPF es solamente usado en la misma àrea. Para IS-IS, enrutadores de ambos niveles, Nivel-1 y Nivel-2 usan el algoritmo SPF para generar el árbol del camino más corto (Shortest Path Tree SPT) respectivamente.

3. Tipos de Interfaces

Los enrutadores de Nivel-1-2 podrían necesitar establecer relación de vecindad de Nivel-1 con solamente un cierto "peer" y establecer una relación de vecindad de Nivel-2 con otro "peer". Se puede limitar la relación de vecindad que puede ser establecida en la interfaz, configurando el tipo de interfaz. Por ejemplo, se puede especificar que las interfaces de Nivel-1 solamente establezcan relación de vecindad Nivel-1, e interfaces de Nivel-2 solamente establezcan relación de vecindad Nivel-2.
Para un enrutador Nivel-1-2, se puede configurar algunas interfaces hacia el Nivel-2. Esto prevendría paquetes Hello Nivel-1 ser enviados hacia la red de backbone Nivel-2. Esto ayuda a salvar el ancho de banda.

4. Escape de Rutas (Route Leaking)

Usualmente, un àrea IS-IS llamada àrea de Nivel-1 maneja las rutas de intra-àrea por enrutadores de Nivel-1. Todos los enrutadores de Nivel-2 forman un area de Nivel-2. Sin embargo, un dominio de ruteo puede contener múltiples áreas Nivel-1 pero solamente un àrea Nivel-2.
Las áreas de Nivel-1 solamente pueden ser conectadas hacia el àrea de Nivel-2. Ellas no pueden ser conectadas mutuamente.
Todos las àreas Nivel-1 notifican su información de ruteo hacia el àrea de Nivel-2 a travès de enrutadores Nivel-1-2. Así, los enrutadores en el àrea de Nivel-2 pueden obtener información de ruteo del dominio entero IS-IS.
Enrutadores de Nivel-2, por default, no anuncian la información de ruteo de áreas de Nivel-1 y áreas de Nivel-2 conocidas, hacia cualquier area de Nivel-1. Sin embargo, enrutadores en áreas de Nivel-1 no pueden obtener información de ruteo fuera del àrea. Esto hace el ruteo óptimo hacia las direcciones destino fuera del àrea, no disponibles.
Para tratar el problema, rutas IS-IS son permitidas escaparse del Nivel-2 hacia el Nivel-1. Esto habilita enrutadores en el àrea especìfica Nivel-1 aprender información de ruteo fuera del àrea.

4.4.3. Tipos de redes IS-IS.

1. Tipos de redes.

IS-IS solamente soporta dos tipos de redes, las cuales pueden ser clasificadas basándose de su enlace físico:

- Enlaces Broadcast: por ejemplo Ethernet y Token Ring.
- Enlaces Punto-a-Punto: por ejemplo, PPP y HDLC.

Para redes Non-Broadcast Multi-Access (NBMA) tal como ATM, se puede configurar sub-interfaces para esto y configurar las sub-interfaces como punto-a-punto o redes broadcast. IS-IS no puede correr sobre redes Point to MultiPoint (P2MP).

2. DIS y pseudos-nodos

En redes broadcast, IS-IS necesita elegir un Sistema Intermedio Designado (Designated Intermediate System "DIS") de todos los enrutadores.
El DIS de Nivel-1 y Nivel-2 son elegidos respectivamente, y se pueden configurar diferentes prioridades para ellos. El de prioridad más alta es el que tiene mayores posibilidades de ser elegido como el enrutador DIS. Si hay dos o más enrutadores con la misma prioridad en la red broadcast, será elegido el que tenga la dirección MAC más grande. Los DIS de diferentes niveles pueden ser el mismo enrutador o diferentes enrutadores.
En las redes broadcast, los enrutadores del mismo nivel en el mismo segmento de red, pueden formar adyacencias. Todos los enrutadores non-DIS pueden también formar adyacencias. Esto es diferente de OSPF, como se muestra en la figura 111.

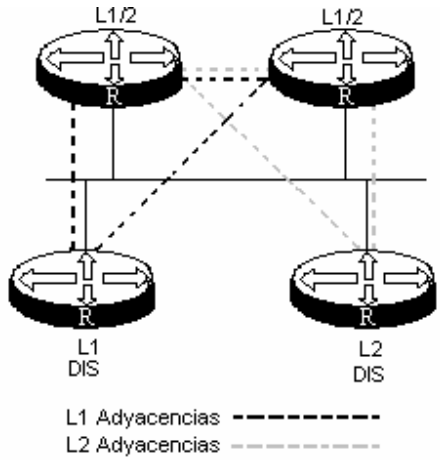


Figura 111. DIS y adyacencias en redes broadcast IS-IS.

DIS es usado para crear y actualizar pseudo-nodos. Èste es también responsable para generar LSP de los pseudo-nodos, los cuales describen los enrutadores disponibles en la red.
Los pseudo-nodos simulan un nodo virtual en la red broadcast y no son enrutadores reales. En IS-IS, los pseudo-nodos son identificados por el System ID de DIS y el 1-byte de Circuit ID.
Con los pseudo-nodos, la topología de la red es más simple y los LSP son más cortos. Cuando la red cambia, el enrutador genera menos LSP y así SPF consume menos recursos. En las redes broadcast IS-IS, aunque todos los enrutadores forman adyacencias con cada otro, la sincronización aùn es asegurada por el DIS.

4.4.4. Tipos de PDU

Los paquetes IS-IS son directamente encapsulados en tramas de enlace de datos. Un PDU consiste del encabezado y campos de longitud variables. El encabezado consiste de un encabezado común y un encabezado especìfico. El encabezado común es el mismo para todos los PDU, pero el encabezado especìfico varìa dependiendo del tipo de PDU, como se muestra la figura 112.

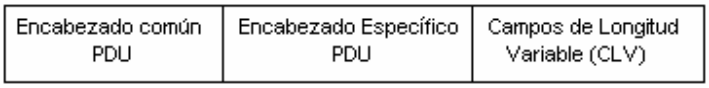


Figura 112. Formato de un PDU.

Todos los PDU tienen un encabezado común, como se muestra en la figura 113.

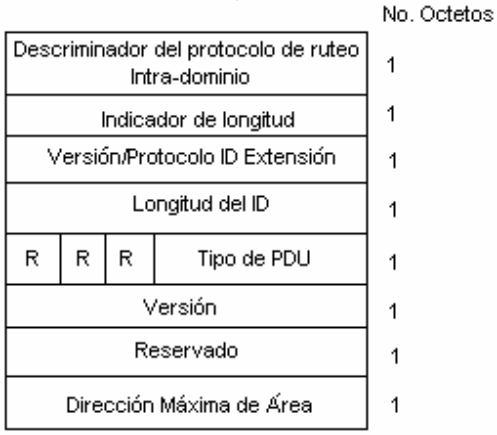


Figura 113. Formato del encabezado común del PDU.

Los campos son:

- Discriminador del protocolo de ruteo Intra-dominio (Intradomain Routing Protocol Discriminator): Este es puesto a 0x83
- Indicador de longitud (Lenght Indicador): indica la longitud del encabezado del PDU (incluyendo el encabezado común y el encabezado específico) en bytes.
- Versiòn/Protocol ID Extensión: es puesto a 1(0x01).
- Longitud del ID (ID Lenght): indica la longitud de la dirección NSAP y NET.
- R (Reservado): es puesto a 0.
- Tipos de PDU (PDU Type): Indica el tipo de PDU a ser transportado. Los diferentes tipos de PDU que pueden ser transportados son los que se encuentran en la tabla 28, estos PDU son los que definen el encabezado específico.
- Versiòn: es puesto a 1 (0x01).
- Dirección Máxima de Área (Maximum Area Address): indica el número máximo de áreas soportadas.

Tabla 28. Tipos de PDU

Valor del Tipo	Tipo de PDU	Acronimo
15	Nivel-1 LAN IS-IS Hello PDU	L1 LAN IIH
16	Nivel-2 LAN IS-IS Hello PDU	L2 LAN IIH
17	Point-to-Point IS-IS Hello PDU	P2P IIH
18	Nivel-1 Link State PDU	L1 LSP
20	Nivel-2 Link State PDU	L2 LSP
24	Nivel-1 Complete Sequence Number PDU	L1 CSNP
25	Nivel-2 Complete Sequence Number PDU	L2 CSNP
26	Nivel-1 Partial Sequence Number PDU	L1 PSNP
27	Nivel-2 Partial Sequence Number PDU	L2 PSNP

1. Paquetes Hello: Los paquetes Hello, también son llamados IS-to-IS Hello PDU (IIH), son usados para iniciar y mantener adyacencias entre enrutadores vecinos. Entre ellos, el Nivel-1 LAN IIH aplica a enrutadores Nivel-1 sobre LAN broadcast, Nivel-2 LAN IIH, aplica a enrutadores Nivel-2 sobre LAN broadcast y P2P IIH aplica sobre redes non-broadcast. Los paquetes en redes diferentes tienen diferentes formatos.
2. Link State PDU (LSP): son usados para intercambiar información del estado de los enlaces. Hay dos tipos de LSP, los cuales son Nivel-1 LSP y Nivel-2 LSP. Enrutadores Nivel-2 transmiten LSP Nivel-2. Enrutadores Nivel-1 transmiten LSP Nivel-1. Enrutadores Nivel-1-2 pueden transmitir ambos tipos de LSP.
3. SNP (Sequence Number PDU): El número de secuencia de PDU describe LSPs en toda o parte de la base de datos para sincronizar y mantener LSDB. Son usados para asegurarse que los enrutadores vecinos tengan la misma noción de cuàl es el más reciente LSP de cada enrutador. El número de secuencias PDU sirve como una función similar para reconocer los paquetes, pero es más eficiente en la operación. SNP incluye CSNP (Complete SNP) y PSNP (Partial SNP). Son divididos en Nivel-1 CSNP, Nivel-2 CSNP, Nivel-1 PSNP y Nivel-2 PSNP. CSNP contiene todo el resumen de la información LSP en el LSDB. Èste mantiene la sincronización entre los enrutadores vecinos. Sobre una red broadcast, DIS transmiten periódicamente los CSNP. El periodo por default de transmisión es de 10 segundos. Sobre un enlace Point-to-Point, CSNP es solamente transmitido cuando la relación de vecindad es inicializada y levantada. Los PSNP solamente lista uno o más nùmero de secuencias LSP recibidos últimamente. Puede conocer múltiples LSP al mismo tiempo. Una vez que el LSDB es encontrado asíncrono, PSNP es adoptado para hacer peticiones a vecinos para que envíen nuevos LSP.
4. CLV (Code-Lenght-Values): El campo de longitud variable son múltiples Code-Lenght-Values (CLVs). El formato que tienen se muestra en la figura 114.

Figura 114. Formato del CLV

Los CLVs varían con el tipo de PDU, los tipos de CLV se muestran en la tabla 29.

Tabla 29. Tipos de PDU incluyendo el nombre del CLV

CLV Code	Nombre	Tipo de PDU aplicado
1	Area Addresses	IIH, LSP
2	IS Neighbors (LSP)	LSP
4	Partition Designated Level2 IS	L2 LSP
6	IS Neighbors (MAC Address)	LAN IIH
7	IS Neighbors (SNPA Address)	LAN IIH
8	Padding	IIH
9	LSP Entries	SNP
10	Authentication Information	IIH, LSP, SNP
128	IP Internal Reachability Information	LSP
129	Protocols Supported	IIH, LSP
130	IP External Reachability Information	L2 LSP
131	Inter-Domain Routing Protocol Information	L2 LSP
132	IP Interface Address	IIH, LSP

4.4.5. Soporte IPv6 para IS-IS

Como IS-IS es un protocolo de ruteo integrado, usa los recursos eficientemente y es más estable. Esta es la idea de fondo para integrar IS-IS, en la cual el protocolo ha sido adaptado para transportar información de ruteo presente para protocolos de red OSI y para IPv4. Esto es alcanzado introduciendo un campo de datos de longitud variable en la forma <Type,Lenght,Value> (TLV). Cada protocolo de capa de red puede usar los TLV de acuerdo a su sintaxis de direccionamiento. Cada protocolo de capa de red soportado es especificado por su identificador del Protocolo de Capa de Red (Network Layer Protocol Identifier "NLPID"), asignado por OSI.
Integrar IS-IS provee la inserción de TLV en todos los paquetes (Hello, LSP y SNP). Información de direccionamiento relevante hace uso de este campo. Paquetes Hello y LSP transportan un campo especificando el protocolo de capa de enlace. Cada protocolo de capa de red soportado es especificado por su NLPID, asignado por el ISO. El valor del NLPID de IPv6 es 142 (ox8E).

El Internet draft-ietf-isis-ipv6-06.txt propone dos nuevos TLV para IPv6. Ellos son descritos en la siguiente lista.

- IPv6 Reachability TLV (Type 236): Define los anuncios de prefijos IPv6 dentro de L1-LSP y L2-LSP. Dentro un L2-LSP, estos pueden ser usados para anunciar un prefijo IPv6 externo al dominio de ruteo, poniendo el bit external en el campo de control. Los siguientes campos componen este TLV: Longitud del Prefijo, Pefijo IPv6, Métrica (4 bytes), y el campo de control.
- IPv6 Interface Address TLV (Type 232): Define la dirección IPv6 de una o más interfaces del enrutador. Èste es anunciado en paquetes Hello, L1-LSP, y L2-LSP. Para paquetes Hello, èste debe contener la dirección de enlace local IPv6 asignada

a la interfaz que esta enviando el paquete Hello. En LSP, èste debe contener la dirección global/sitio-local asignado al enrutador.

4.5. EXTENSIONES BGP PARA IPv6

No es actual IPv6 para BGP. El soporte IPv6 deriva de la capacidad de BGP-4 para intercambiar información con otros protocolos de capa de red que no sean IPv4. Estas extensiones Multi-protocolo de BGP-4 son definidas en el RFC2858, el cual hace obsoleto al RFC2283. El RFC2283 es mencionado aquí porque es la base del RFC2545, el cual define las extensiones IPv6 de BGP-4. Es importante conocer BGP-4 antes de revisar sus extensiones multi-protocolos.

4.5.1. Descripción de BGP-4

Cada AS corre su protocolo de ruteo interior (RIP, OSPF, etc) para distribuir toda la información de ruteo dentro del AS. BGP es un protocolo de ruteo exterior, su función primaria es intercambiar información acerca de la accesibilidad de redes entre AS. Cada AS recibe un único número de AS asignado por la autoridad de numeración. La figura 115 muestra los diferentes tipos de AS que pueden ser interconectados usando BGP-4.

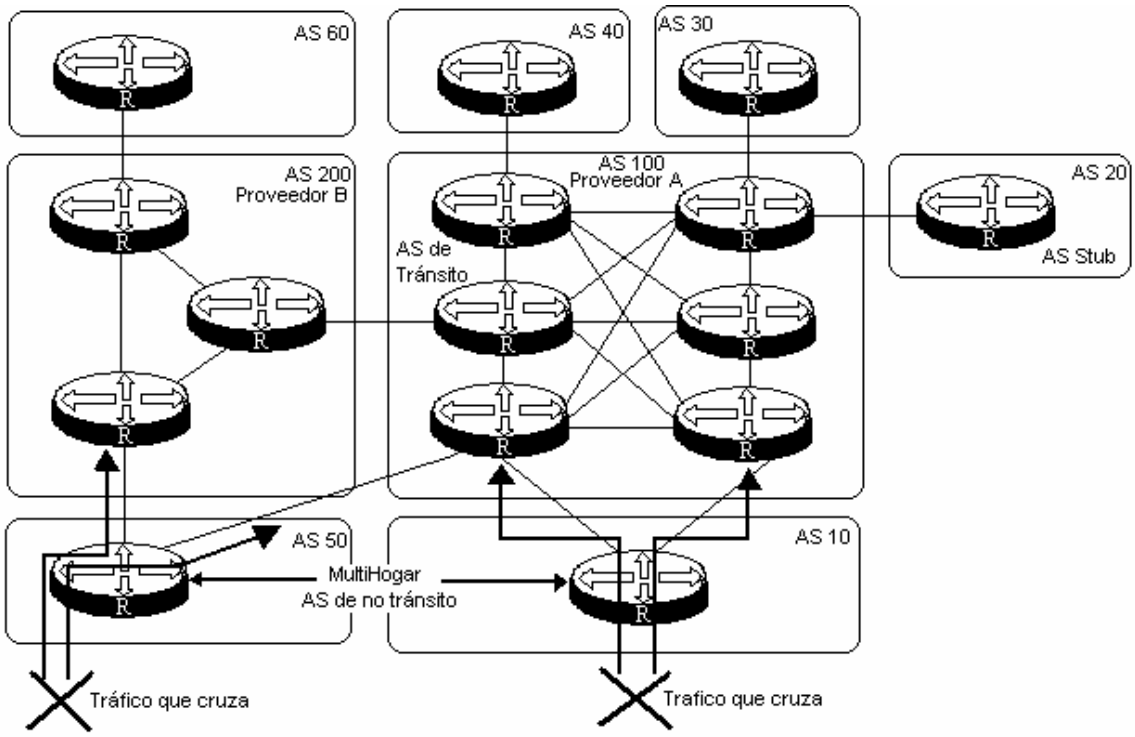


Figura 115. Tráfico BGP-4 y tipos de AS.

Los tipos de AS se mencionan en la siguiente lista[19]:

- AS de Tràsito (Transit AS): Un AS de tràsito tiene múltiples conexiones hacia otros AS. Actualizaciones de ruteo que lleguen de cualquier AS al AS de tràsito pueden ser pasadas a travès del AS y distribuidas hacia otros AS vecinos. Un AS de tràsito puede enviar tràfico hacia cualquier otro AS basándose de la información de ruteo recibida. Los AS de grandes ISP son usualmente de este tipo.
- AS Stub: Un AS Stub tiene una sola conexión hacia otro AS. Todo el tràfico del o hacia el AS Stub pasa a travès de este enlace. Pequeños ISP y redes de campus o corporaciones usan este tipo de AS.
- AS multi-hogar de no-tràsito (Multihomed nontransit AS): Un AS multi-hogar de no-tràsito tiene múltiples conexiones hacia uno o mas AS. Èste no pasa actualizaciones de ruteo a travès de el. Tràfico no perteneciente a este AS nunca será enviada. Un AS multi-hogar de no-tràsito permite múltiples puntos de entrada/salida a ser usadas para compartir carga de tràfico de entrada y salida.

Dos enrutadores intercambiando información de ruteo con BGP son llamados BGP peers o BGP speakers. Ellos establecen primero, una sesión TCP, porque TCP garantiza una conexión confiable. Los peers entonces abren una conexión BGP para intercambiar mensajes BGP. Los mensajes màs importantes de BGP son los mensajes de Actualización (UPDATE), los cuales contienen las rutas para ser intercambiadas. Una ruta BGP es definida como una unidad de información consistiendo de la información de accesibilidad de la capa de red (Network Layer Reachability Information "NLRI") y un sistema de atributos de trayecto. El NLRI es básicamente un prefijo IPv4 y su longitud de prefijo. Cualquier concepto de información de clases IPv4 ha sido eliminado. El NLRI permite representar una sola red o lo màs común, un resumen (summary) de un rango de direcciones. Cada NLRI es acompañado por un sistema de atributos del trayecto que agregan información adicional de rutas BGP, por ejemplo, la dirección de siguiente salto, una secuencia de ASs por la cual las rutas tienen que pasar durante su actualización, o su origen. Decisiones de ruteo y manejo de tràfico frecuentemente son basados en estos atributos de trayectos. Un atributo debe ser enfatizado aquí, porque este juega un rol importante en la detección de loops: esto es llamado AS_PATH, y èste transporta una secuencia de números de AS por la cual la ruta tiene que pasar. Si el peer receptor recibe y reconoce su propio número de AS dentro del AS_PATH, èste rechaza la ruta correspondiente.

Actualizaciones de ruteo BGP son intercambiados entre dos peers. Ellos son gobernados por un sistema de reglas llamadas políticas. Políticas de salida especifica cuàles NLRI son anunciadas hacia un peer particular. Un enrutador puede anunciar solamente el NLRI que usa el mismo. Políticas de entrada especifica cuàles NLRI son aceptados por un peer en particular. Políticas permiten también ser usados para modificar un NLRI y sus atributos para cambiar las características de una ruta.

[19] HAGEN SILVIA (2002). Ipv6 Essentials. Gravenstein Highway North, Sebastopol, CA. O`relly Media, Inc. Primera Edicion. Pp 194.

4.5.1.1. Estableciendo una conexión BGP

El orden para intercambiar actualizaciones de ruteo, dos peers primero tiene que establecer una conexión BGP. La Figura 116 ilustra los pasos necesarios para establecer una conexión BGP, incluyendo los diferentes mensajes intercambiados y los estados de los peer. Los estados enteros de màquina se encuentran explicados en el RFC1771. Cada mensaje y sus campos son explicados en la sección de "Encabezado de mensajes BGP".

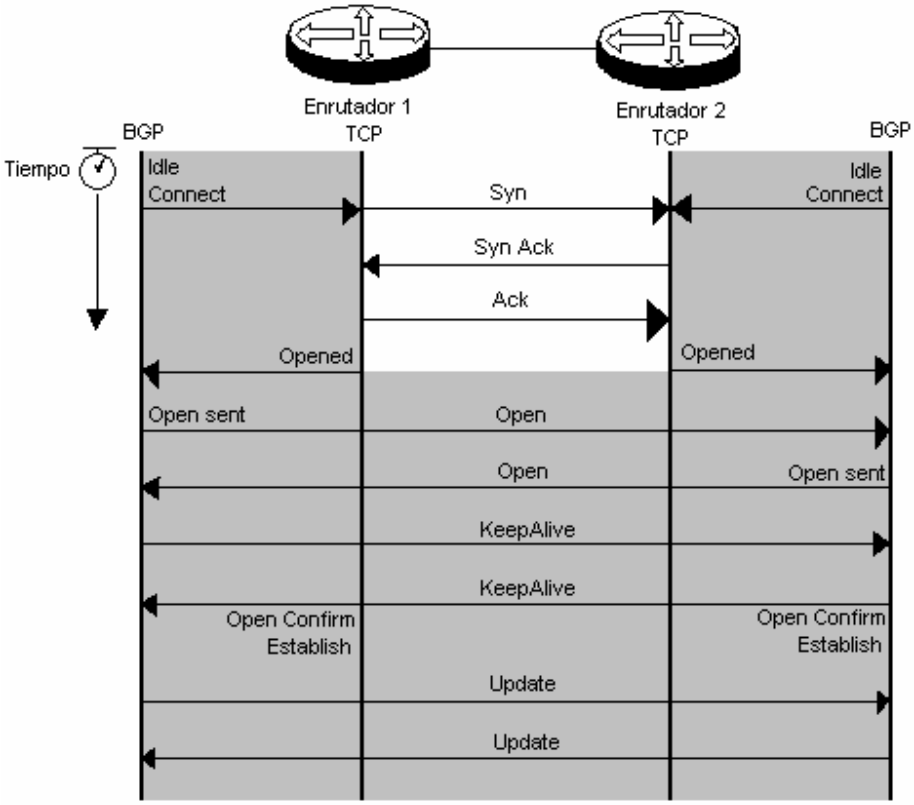


Figura 116. Estableciendo una conexión BGP.

Si ambos enrutadores simultáneamente intentan establecer una conexión BGP por cada uno, dos conexiones paralelas podrían ser formadas. Para evitar esta colisión de conexión, un enrutador tiene que ceder. La conexión iniciada por el enrutador con el mayor identificador BGP es el que prevalece. El identificador BGP es únicamente asignado a cada enrutador BGP y es intercambiado durante el mensaje de abertura (OPEN). Una vez que la abertura es confirmada, los enrutadores intercambian la tabla entera de ruteo basándose en sus políticas. Cambios solamente en la tabla de ruteo son intercambiados de ahora en adelante. Mensajes "mantener con vida" la conexión (KEEPALIVE) previene que la conexión quede fuera de tiempo (timing out). La sesión TCP garantiza la entrega confiable de cada paquete.

BGP distingue entre las siguientes conexiones de peer:

- Conexión IBGP: Los peers están en el mismo AS y son llamados peers internos. Rutas BGP aprendidas de peers internos no deben ser enviados hacia atrás hacia otro peers internos; ellos solamente pueden ser enviados hacia peers externos. Cada

peer interno debe tener una conexión hacia todos los otros peers internos. Peers internos son completamente "mesh", esto no significa que todos los dispositivos deban ser conectados con cada otro, aunque todos deben ser alcanzables a través de capa 3. Los atributos AS_PATH y NEXT_HOP no deben ser modificados cuando pasan actualizaciones hacia peers internos.

- Conexión EBGP: Los peers están en diferentes AS y son llamados peers externos. Rutas BGP aprendidas de peers externos pueden ser actualizadas hacia todos los otros peers. Cuando se envía una actualización hacia un peer externo, los atributos AS_PATH y NEXT_HOP son modificados. El enrutador emisor agrega el número local de AS en AS_PATH y pone en el campo de NEXT_HOP su dirección IPv4 local.

Mensajes de notificación (NOTIFICATION) informa a los peers de cualquier error durante el proceso de abertura (OPEN) y actualización (UPDATE). La conexión puede ser dada de baja usando un mensaje de notificación de finalizar.

4.5.1.2.Almacenamiento de rutas y políticas

Las rutas BGP son almacenadas en una Base de Información de Ruteo (Routing Information Base "RIB"), la figura 117 muestra los tres diferentes RIB y su interacción.

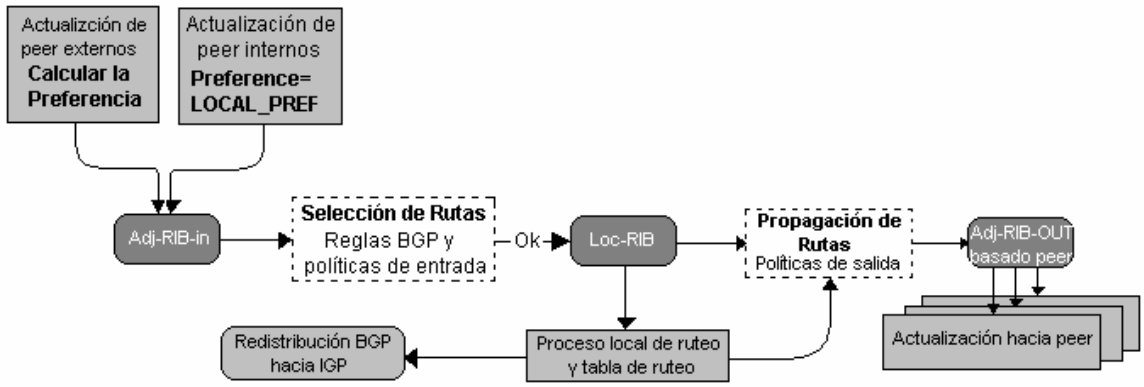


Figura 117. BGP RIB y sus interacciones.

Mensajes entrantes pueden contener nuevas rutas factibles (feasibles), reemplazos de rutas de tempranas actualizaciones, o rutas que han sido retiradas (withdrawn) por los anuncios de los peers. Todas estas rutas son puestas dentro de Adj-RIB-In. Por cada nueva o cambio de ruta, un grado de preferencia es calculado basado en la política de entrada (inbound). Esta preferencia es puesta dentro de los atributos de LOCAL_PREF. Si la ruta llega de un peer interno, el LOCAL-PREF ya ha sido transportada en la actualización y no debe ser recalculada. Cada ruta en el Adj-RIB-In ahora es procesada por el proceso de selección de rutas e introducida dentro del Local-RIB. El proceso de selección primero mira en los atributos NEXT_HOP y AS-PATH de la ruta. La dirección IP especificada en el NEXT_HOP debe ser alcanzable a través de una entrada en la tabla de ruteo local. El AS_PATH no debe contener el número del AS local. Si los dos atributos son cumplidos, la ruta es aceptada o ignorada basándose en la política de entrada; de otra forma, la ruta es ignorada. En caso de múltiples rutas hacia el mismo destino, la ruta con la mayor

preferencia es aceptada. En caso de que sea la misma preferencia, una regla compleja de lazo-roto “tie-breaking” asegura que solamente una de las rutas hacia el mismo destino es aceptada. El RFC1771 habla con más detalle de la regla “tie-breaking”.

Rutas en el Local-RIB ahora son puestas dentro de la tabla de ruteo local. La dirección de siguiente salto verdadera es tomada de las entradas de rutas locales hacia la dirección IPv4 especificada en el atributo NEXT_HOP.

Todas las rutas en el Local-RIB y todas las rutas en la tabla de ruteo local son elegibles a ser anunciadas hacia peers externos de este enrutador. Solamente rutas en el Local-RIB aprendidas de peers externos son elegibles a ser anunciadas hacia todos los peers internos de este enrutador al menos que reflexión (reflection) de rutas sea habilitado (RCF2796). La política de salida propaga las rutas hacia un peer especìfico Adj-RIB-Out. La política de salida podría realizar agregación o modificación del atributo de trayecto (path). Cambios en la Adj-RIB-Out causa el proceso de actualización que envìe una actualización hacia los peers.

4.5.2. Mensajes de encabezado BGP

Mensajes BGP son transportados en la cima de las conexiones TCP, las cuales pueden ser establecidas sobre IPv4 o IPv6. Las direcciones de origen y destino del datagrama dependen de la configuración del peer. Ellas siempre son unicast. Conexiones BGP usan el bien-conocido TCP puerto 179. Recordar que solamente una conexión BGP es establecida entre dos enrutadores peering. La figura 118 muestra el formato del encabezado de los mensajes BGP. El encabezado tiene un tamaño fijo de 19 bytes.

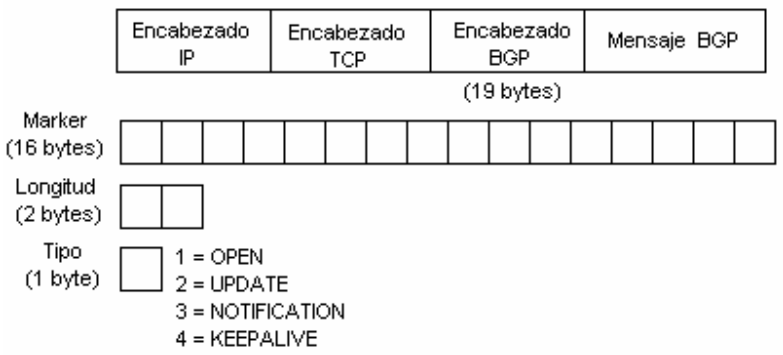


Figura 118. Formato del encabezado de mensajes BGP.

Los campos del encabezado BGP son los siguientes:

- Marker (16 bytes): Contiene datos de autenticación, si la autenticación fue negociada entre los dos peers. Todos los bits son puesto a uno, si autenticación no es usada o en el mensaje OPEN.
- Longitud (Lenght 2 bytes): La longitud total de los mensajes BGP, incluyendo encabezados. El valor debe ser entre 19 y 4096. El tamaño máximo de menajes de cualquier mensaje BGP es de 4096 bytes.
- Tipo (Type 1 byte): Indica los tipos de mensajes BGP, mostrados en la tabla 30.

Tabla 30. Tipos de mensajes BGP

Tipo	Nombre	Descripción
1	OPEN	Inicializa conexiones BGP y negocia parámetros de sesión.
2	UPDATE	Intercambia rutas BGP factibles (feasibles) y retiradas (withdrawn)
3	NOTIFICATION	Reporta errores o termina conexiones BGP
4	KEEPALIVE	Mantiene las conexiones BGP que no expiren.

4.5.3. Mensajes OPEN

Tan pronto la conexión TCP entre los peers BGP ha sido establecida, los enrutadores envían mensajes OPEN para inicializar la conexión BGP. Este mensaje verifica la validez de los peer y negocia parámetros usados por la sesión usando los campos mostrados en la figura 119. Para verificar la validez de un peer, cada lado de la conexión debe configurar la dirección IP y nùmero del AS del peer.

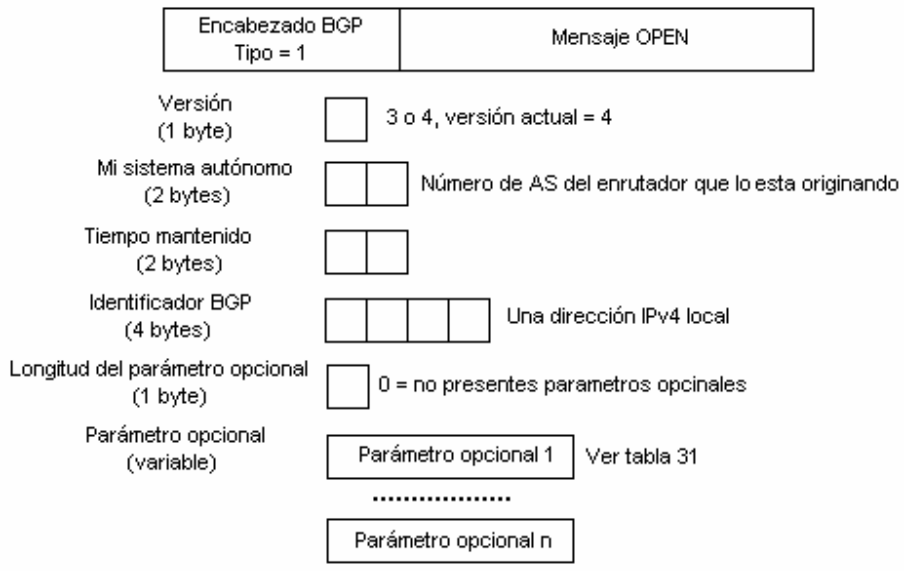


Figura 119. Mensaje OPEN de BGP

Los campos del mensaje OPEN son los siguientes:

- Versión (1 byte): Indica la versión BGP usada por el peer emisor. La versión actual es la 4. Ambos peer deben estar de acuerdo con la misma versión. La versión puede ser negociada. Cada peer usualmente indica la versión más alta soportada. Si el peer receptor no soporta esta versión, èste notifica al peer y termina la sesión.
- Mi sistema autónomo (2 bytes): Indica el número del AS del enrutador emisor. El enrutador receptor debe verificar que este nùmero sea el nùmero AS del peer. Si èste es incorrecto, el peer es notificado y la sesión es terminada. Si el nùmero del AS es el mismo del nùmero AS del enrutador receptor, el peers es interno (iBGP); de otra forma, el peer es externo (EBGP).
- Tiempo mantenido (Hold Time 2 bytes): Se propone un tiempo máximo en segundos que puede ser transcurrido antes de que cualquier mensaje BGP deba llegar a esta interfaz. El contador Hold es negociado por el valor màs pequeño anunciado por cualquier peer. Para mantener una conexión que no expire, los peers

envían mensajes “mantener con vida” KEEPALIVE una vez cada HoldTime de 3 segundos. Un tiempo de hold de cero indica que mensajes KEEPALIVE no necesitan ser enviados. El valor del tiempo de hold es 0 o mayor que 2.

- Identificador BGP (4 bytes): Cada enrutador debe ser identificado por un único, identificador global asignado. Al configurar, el identificador BGP es puesto como una dirección IPv4 de la interfaz. Esto significa que el enrutador debe tener al menos una dirección IPv4 configurada localmente, aùn si es ambiente IPv6-only. El mensaje es rechazado si el identificador BGP es igual al identificador BGP del receptor o si el identificador BGP es ilegal. Durante la selección de rutas, el identificador BGP puede ser usado para romper un lazo.
- Longitud del parámetro opcional (1 byte): Indica la longitud de los parámetros opcionales a ser negociados. Una longitud de cero indica que no hay parámetros opcionales.
- Parámetros opcionales: Cada parámetro opcional consiste de un trío <Type, Lenght, Value> (TLV). Ambos enrutadores deben conocer y estar de acuerdo de los parámetros opcionales; de otra forma, el peer es notificado del rechazo del parámetro. Esto podría continuar con la terminación de la sesión. Dos parámetros son especificados, se muestran en la tabal 31. La capacidad de parámetros opcionales BGP es muy importante para soportar IPv6.

Tabla 31. Parámetros opcionales

Tipo	Nombre	Descripción
1	Autenticación	El parámetro consiste de dos campos: Código de Autenticaron y Dato de Autenticación. El código de autenticación define el mecanismo de autenticación usado y còmo el marker y campo de dato de autenticación son computados.
2	Capacidades BGP	El parámetro consiste de uno o mas tríos de <Code,Lenght,Value> identificando diferentes capacidades BGP. Èste es definido en el RFC2842. El parámetro de capacidades puede aparecer màs de una vez en el mensaje OPEN. El código de capacidades puesto a 1 indica las capacidades de extensión de multi-protocolo, como lo define el RFC2858.

Las capacidades de extensiones de multi-protocolo tienen un campo de 4 bytes. Los primeros 2 bytes identifican al Identificador de Familia de Direcciones (Address Family Identifier “AFI”), el byte 3 es reservado, y el byte 4 define el Identificador de Familia de Direcciones Subsecuentes (Subsequent Address Family Identifier “SAFI”). AFI define el protocolo de capa de red usado en la extensión de multi-protocolo. SAFI define información adicional relacionada al protocolo, tal como si el protocolo usa envìo unicast (SAFI=1), envìo multicast (SAFI=2), o ambos (SAFI=3). Para soportar IPv6 las capacidades de extensión de multi-protocolo es puesto a <Code=1, Lenght=4, Value=hexadecimal 0x0002 0001).

4.5.4. Mensajes de actualización (UPDATE)

Un mensaje UPDATE transporta ruta(s) anunciadas por el peer que lo origina. Èste es dividido en tres secciones, como se muestra en la figura 120. La primera sección especifica el NLRI IPv4 que el peer emisor esta retirando (withdrawing). La segunda sección define

todos los atributos del trayecto (path) asociados con las factibles (feasibles) NLRI IPv4 seguidas en la sección tres. Multiples NLRI con el mismo sistema de atributos del trayecto (path) pueden ser puestos en un solo mensaje UPDATE.

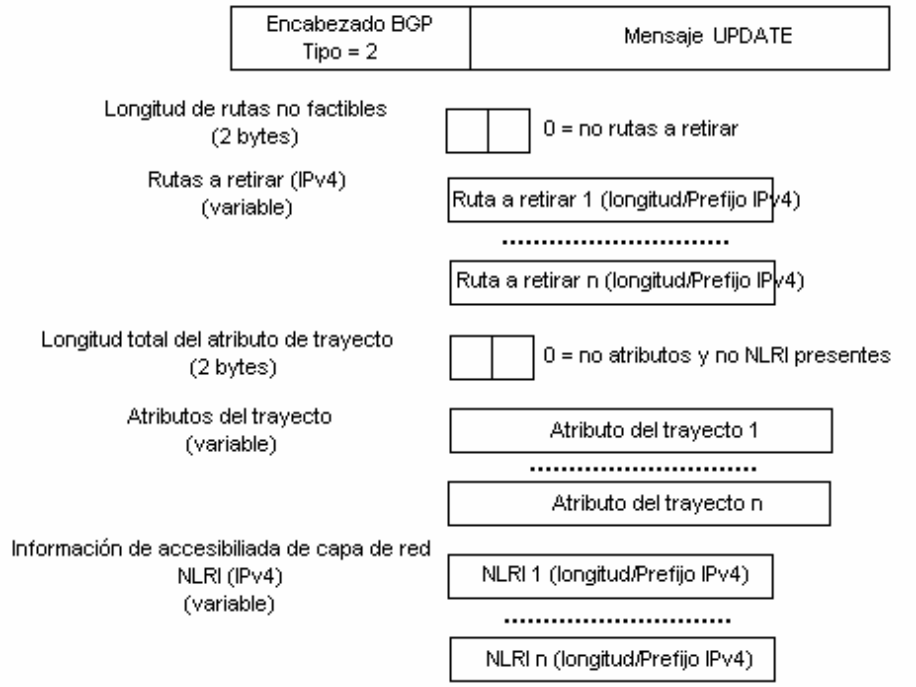


Figura 120. Mensaje UPDATE BGP.

Los campos de los mensajes UPDATE son listados a continuación:

- Longitud de rutas no factibles (Unfeasibles 2 bytes): Define la longitud del campo de rutas a retirar (withdrawn). Puesto a 0, indica que el peer que lo origina no tiene rutas a retirar (withdrawn) con este mensaje.
- Rutas a retirar (Withdrawn): Una lista de NLRI IPv4 que ya no más son vàlidas. Cada NLRI es codificado como <longitud, Prefijo> y representa un prefijo IPv4. El campo de longitud de un 1-byte define la longitud del campo del prefijo correspondiente. El campo prefijo es rellenado de octetos completos de bits cero. Porque el NLRI son prefijos IPv4, este campo nunca puede ser usado con rutas retiradas IPv6. Ver la sección de “Extensión BGP para IPv6” para màs detalles.
- Longitud Total del atributo de Trayecto (Path 2 bytes): Define la longitud del campo de atributos del trayecto (path).
- Atributos del Trayecto (Path): Contiene una lista de atributos del trayecto que pertenecen a los NLRI factibles anunciados. Los atributos son explicados en la sección “Atributos BGP”.
- Información de accesibilidad de capa de red: Una lista de NLRI IPv4 que son anunciadas con esta actualización. Cada NLRI es codificado como <Longitud, prefijo> y representa un prefijo IPv4. El campo de longitud de 1-byte define la longitud del correspondiente campo del prefijo. El campo de prefijo es rellenado de octetos completos de bits cero. Porque los NLRI son prefijos IPv4, este campo

nunca puede ser usado para anunciar rutas IPv6. Ver la sección de "Extensión BGP para IPv6" para màs detalles.

4.5.5. Atributos BGP

Atributos del trayecto (path) provee información adicional de los NLRI anunciados. Cada atributo del trayecto tiene un encabezado de atributos de 2-bytes, como se muestra en la figura 121.

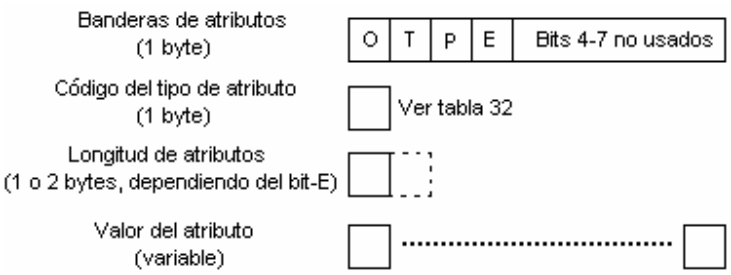


Figura 121. Atributos de trayecto (path) de BGP.

El encabezado de atributos de trayecto es explicado en la siguiente lista:

- Bit O (bit opcional): Define si los atributos son opcionales (puesto a uno) o bien conocidos (puesto a cero). Un atributo bien conocido deber ser reconocido y soportado para cada enrutador BGP. Atributos opcionales podrían no ser reconocidos por algunos enrutadores.
- Bit T (bit Transitivo): Define si los atributos son transitivos (puesto a uno) o no transitivos (puesto a cero). Atributos transitivos deben ser siempre pasados cuando el NLRI es anunciado hacia otro peer. Atributos bien conocidos deben ser siempre transitivos.
- Bit P (bit parcial): Aplica solamente a atributo opcional transitivo. Si cualquier enrutador a lo largo de la actualización del trayecto no reconoce el atributo opcional transitivo, se debe poner el bit P. Esto indica que al menos un enrutador en el trayecto hacia la ruta no reconoce el atributo. Este bit siempre debe ser puesto a cero para atributo opcional no transitivo o bien conocidos.
- Bit E (bit de longitud extendida): Define si el campo longitud del atributo es de 1 byte (puesto a cero) o 2 bytes (puesto a uno). Longitud extendida podría ser usado si el atributo de los datos es mayor que 255 bytes.
- Código del atributo (1 byte): Define el tipo de atributo. La tabal 32 lista y explica algunos de los más comunes atributos. Para mayores detalles dirigirse al RFC1771 o de cualquier RFC extendido de BGP.

Tabla 32. Atributos BGP[20]

Tipo	Nombre/Bandera	Descripción
1	ORIGIN (Bien conocido)	Define el origen original de esta ruta. 0=IGP, 1=EGP, 2=Incompleto
2	AS_PATH (Bien conocido)	Una secuencia de números de AS que esta ruta ha cruzado durante su actualización. El nùmero AS mas a la derecha

[20] Y. Rekhter, T. Li (1995). RFC1771 "A Border Gateway Protocol 4 (BGP-4)". Standards Track. Pp 20.

		define el AS origen. Cada AS cruzado es **prepended**. Previene loops y puede ser usado para las políticas.
3	NEXT_HOP (Bien conocido)	Especifica la dirección IPv4 de siguiente salto. No puede ser usado por IPv6.
4	MED (No transitivo opcional)	El MULTI_EXIT_DIS (MED) indica un deseo de preferencia (4 bytes) de esta ruta hacia el peer. El màs bajo es el mejor. Diseñado para múltiples conexiones BGP entre dos AS para compartir carga de tráfico entrante.
5	LOCAL_PREF (Bien conocido)	Define una preferencia local (4 byte) de esta ruta. El más alto es el mejor. Este es usualmente calculado sobre rutas que llegan de peers externos y son preservadas hacia peers internos. Diseñado para compartir carga de tráfico saliente.
6	ATOMIC_AGGREGATE (Bien conocido)	Específica que uno de los enrutadores tiene seleccionada esta ruta menos-especìfica sobre una ruta màs-especìfica.
7	AGGREGATOR (Transitivo opcional)	El identificador BGP del enrutador que agregò (aggregated) rutas dentro esta ruta.
8	COMMUNITY (Transitivo opcional)	Transporta una etiqueta informacional de 4 bytes. Puede ser usada por el proceso de selección de ruta. Definido en el RFC1997.
14	MP_REACH_NLRI (No transitivo opcional)	Anuncia NLRI multi-protocolo. Usado para prefijos IPv6. Mas sobre el tema en la sección “Extensiones BGP para IPv6”.
15	MP_UNREACH_NLRI (No transitivo opcional)	Retirados (wihtdraws) NLRI multi-protocolo. Usado por prefijos IPv6. Mas sobre el tema en la sección “Extensiones BGP para IPv6”.

4.5.6. Mensajes de Notificación y “mantener con vida” KEEPALIVE.

Mensajes de Notificación son usados para reportar errores. Un campo de Código de Error de 1 byte, especifica la principal categoría del error. Un campo de Sub-código provee el error actual siguiendo el campo de Código de Error. Por razones de eficiencia, datos acerca del error son puestos en el campo de Datos (Data). En el RFC1771 en la sección 4.5 se describen todos los códigos de errores. Documentos adicionales de extensiones de BGP agregan sub-códigos de errores. Mensajes de error para la extensión de BGP para IPv6 son especificados en el RFC2858.

Mensajes KEEPALIVE no contienen datos cualesquiera, sòlo el mensaje del encabezado BGP con el tipo de mensaje 4. Son usados para prevenir que expire una conexión BGP.

4.5.7. Extensiones BGP para IPv6

BGP-4 transporta solamente tres piezas de información que son verdaderas específicamente para IPv4:

- NLRI (factibles y Retiradas) en los mensajes de UPDATE contienen un prefijo IPv4.
- NEXT_HOP atributos del trayecto en el mensaje de UPADATE contienen una dirección IPv4.
- Identificador BGP està en el mensaje OPEN y en el atributo AGGREGATOR.

Para hacer BGP-4 disponible para otros protocolos de capa de red, el NLRI multi-protocolo y su información de siguiente salto deben ser agregados. El RFC 2858 extiende BGP para soportar múltiples protocolos de capa de red. IPv6 es una de los protocolos soportados, como se describe también en el RFC2545. Para acomodar los nuevos requerimientos para soportar multi-protocolos, BGP-4 agrega dos nuevos atributos para anunciar y retirar (withdraw) NLRI multi-protocolo. El identificador BGP permanece sin cambios. Enrutadores BGP con extensión IPv6 aùn necesitan una dirección local IPv4. Para establecer una conexión BGP e intercambiar prefijos IPv6, los enrutadores peering necesitan anunciar el parámetro opcional de capacidades BGP, para indicar el soporte IPv6. Conexiones BGP y selección de rutas permanecen sin cambios. Cada implementación necesita extender el RIB para acomodar rutas IPv6. Políticas necesitan tomar en consideración NLRI IPv6 e información de siguiente salto para la selección de rutas.

Un mensaje UPDATE anuncia solamente sistemas de rutas no factibles NLRI IPv6 con la longitud de campo a cero y no transporta NLRI IPv4. Todos los anuncios o rutas IPv6 a retirar (withdraw) son transportadas dentro del MP_REACH_NLRI y MP_UNREACH_NLRI. UPDATE debe transportar los atributos del trayecto ORIGIN y AS_PATH; en conexiones IBGP; èste debe también transportar LOCAL_PREF. El atributo de NEXT_HOP no debe ser transportado. Si el mensaje UPDATE contiene el atributo de NEXT_HOP, el peer receptor debe ignorarlo. Todos los demás atributos pueden ser transportados y son reconocidos.

Un mensaje de UPDATE podría anunciar NLRI IPv6 y NLRI IPv4 con los mismos atributos de trayecto. En este caso, todos los campos pueden ser usados. Para NLRI IPv6, el atributo de NEXT_HOP, debe ser ignorado como sea. NLRI IPv4 e IPv6 son separados en los correspondientes RIB.

4.5.7.1.Atributo del trayecto MP_REACH_NLRI

Este atributo opcional no-transitivo permite el intercambio de NLRI IPv6 factibles hacia un peer, a lo largo con su dirección IPv6 de siguiente salto. El NLRI y el siguiente salto son entregados en un atributo, como se representa en la figura 122.

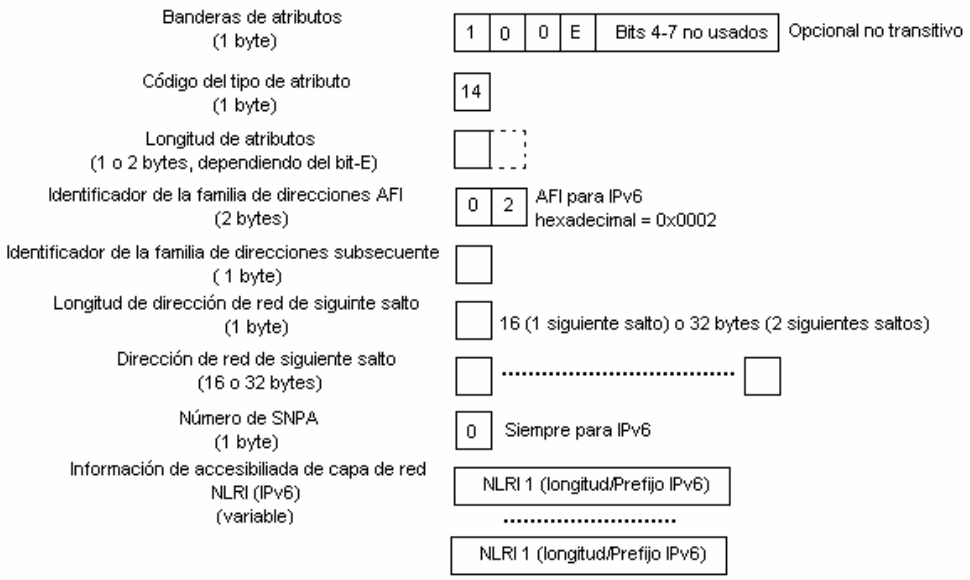


Figura 122. El atributo del trayecto MP_REACH_NLRI para IPv6.

Los campos comprendidos en el atributo del trayecto MP_REACH_NLRI se detallan en la siguiente lista:

- Identificador de la familia de direcciones (Address Family Identifier "AFI"): Define el protocolo de capa de red. IPv6 usa el valor de 0x0002 (hexadecimal) especificado en el RFC1700.
- Identificador de la familia de direcciones subsecuente (Subsequent Addres Family Identifier "SAFI" 1 byte): Define si el protocolo usa envió unicast (SAFI=1), multicast (SAFI=2), o ambos (SAFI=3).
- Longitud de la dirección de red del siguiente salto (1 byte): Define el número de bytes usados para el campo de dirección de siguiente salto. IPv6 pone este campo a 16 o 32, dependiendo del número de direcciones de siguiente salto dado.
- Dirección de red de siguiente salto: Contiene la dirección IPv6 de siguiente salto de esta ruta IPv6. Este campo es actualizado cuando anuncia esta ruta hacia un peer externo. El enrutador escoge su propia dirección IPv6 global/sitio del enlace hacia el peer externo. Este campo generalmente no es actualizado cuando anuncian esta ruta hacia un peer interno. Si la dirección IPv6 de siguiente salto y la dirección IPv6 del peer comparten un enlace común –por ejemplo, un enlace entre dos peers externos- la dirección de enlace-local del enlace común deber ser agregado como una segunda dirección de siguiente salto. Al contrario, cuando anuncian esta ruta hacia un peer interno, la dirección de enlace-local recibida por un peer externo necesita ser removida.
- Nùmero de SNPA (1 byte): Define el número de Subnetwork Points of Attachment (SNPA) seguidos a la derecha después de este campo. SNPA transporta información adicional asociada con el enrutador asociado con la dirección de siguiente salto. IPv6 no usa este campo y lo pone a cero. Por lo tanto, campo de datos SNPA no proseguirán.

- Información de la accesibilidad de la capa de red (Network Layer Reachability Information): Una lista de NLRI IPv6 que son anunciados con este atributo. Cada NLRI es codificado como <longitud, prefijo>. El campo de Longitud de 1-byte define la longitud del correspondiente campo Prefijo. El campo Prefijo es rellenado con octetos de bits cero. La longitud de este campo es la longitud restante después de deducir la longitud de todos los campos previos de la longitud del atributo.

4.5.7.2.Atributo del trayecto MP_UNREACH_NLRI

Este atributo opcional no-transitivo permite enviar a los peer múltiples rutas IPv6 a retirar no màs vàlidas. Como se ilustra en la figura 123, èste básicamente contiene una lista de prefijos IPv6 que el peer debe remover de sus RIB.

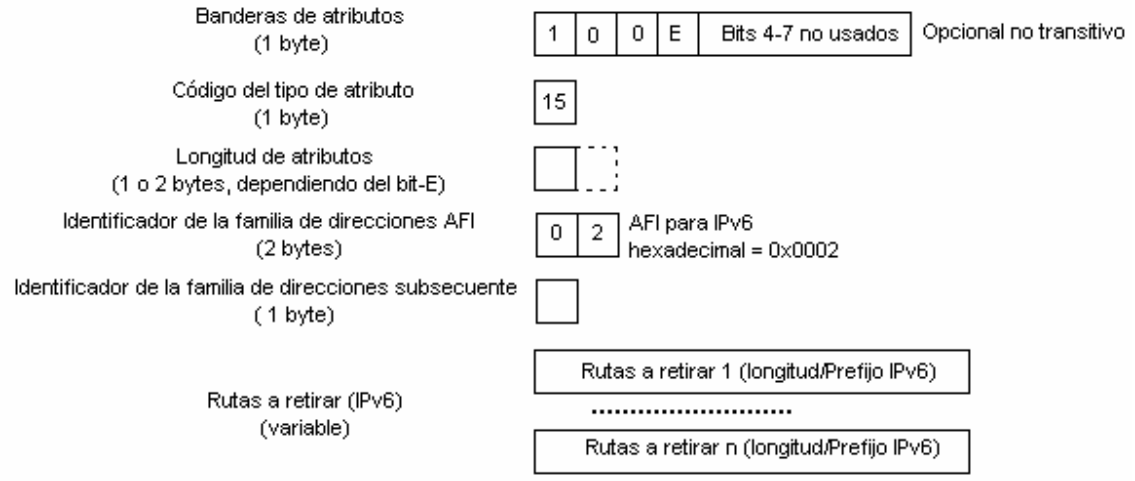


Figura 123. El atributo del trayecto MP_UNREACH_NLRI para IPv6.

Los campos comprendidos en el atributo del trayecto MP_UNREACH_NLRI se detallan en la siguiente lista:

- Identificador de la familia de direcciones (Address Family Identifier “AFI” 2-bytes): Define el protocolo de capa de red. IPv6 usa el valor 0x0002 (hexadecimal).
- Identificador de la familia de direcciones subsecuente (Subsequent Addres Family Identifier “SAFI” 1 byte): Define si el protocolo usa envio unicast (SAFI=1), multicast (SAFI=2), o ambos (SAFI=3).
- Rutas retiradas (withdrawn rutes): Una lista de NLRI IPv6 que son retiradas de servicio. Cada NLRI es codificada <longitud, prefijo>. El campo de Longitud de 1-byte define la longitud del correspondiente campo Prefijo. El campo Prefijo es rellenado con octetos de bits cero. La longitud de este campo es la longitud restante después de deducir la longitud de todos los campos previos de la longitud del atributo.

Capitulo

5.IPv6 en las redes académicas avanzadas

Resumen

En este capìtulo se trataran los antecedentes de la red de Internet de hoy en día, ya que por el uso que se le ha venido dando se ha encausado a un ámbito totalmente comercial, motivando a comunidades universitarias (principalmente por la necesidad de mayores recursos), tomar la iniciativa de desarrollar una nueva red denominada Internet2 o NREN (National Research and Education Networks); asimismo se plantea la situación actual de estas redes a nivel mundial, enfocándose esencialmente en México en la red de CUDI (Corporación Universitaria para el Desarrollo de Internet), que es el organismo que maneja el proyecto de la red Internet2 en México.
Además se obtendrá una visión del soporte que se tiene de la nueva versión del protocolo de Internet IPv6 y de la integración que se ha hecho de este protocolo en las redes avanzadas de Internet2.

5.1. ANTECEDENTES DE INTERNET

Hoy en día, Internet es la red mundial más grande de redes de computadoras y ha supuesto una revolución sin precedentes en el mundo de la informática y de las comunicaciones. Los inventos del telégrafo, teléfono, radio y ordenador sentaron las bases para esta integración de capacidades nunca antes vivida. Internet es a la vez una oportunidad de difusión mundial, un mecanismo de propagación de la información y un medio de colaboración e interacción entre los individuos y sus ordenadores, independientemente de su localización geográfica.

Los inicios de Internet nos remontan a los años 60. En plena guerra fría, Estados Unidos crea una red exclusivamente militar, con el objetivo de que, en el hipotético caso de un ataque ruso o desastre natural, se pudiera tener acceso a la información militar desde cualquier punto del país. ARPA fue la encargada de iniciar un proyecto que consistía en dar respuesta a la necesidad de crear un sistema de comunicaciones que sobreviviera a cualquier conflicto militar. Los 19 componentes de la agencia ARPA (cuyos miembros procedían de universidades y laboratorios científicos), se reunieron en 1967 para desarrollar un sistema más perfeccionado, dando como resultado la aparición de ARPANET (1969). Ésta era una red compuesta por ordenadores en la que todos los nodos (o intersecciones) tenían la misma importancia, así, si uno desaparecía no afectaba la red.

La Agencia de Proyectos de Investigación Avanzada (ARPA, Advanced Research Projects Agency) cambió su nombre a Agencia de Proyectos de Investigación Avanzada para la Defensa (DARPA, Defense Advanced Research Projects Agency) en 1971, más tarde retomó su antigua denominación ARPA en 1993, para volver a DARPA en 1996.

En principio, la red contaba con 4 ordenadores distribuidos entre distintas universidades del país.

- Nodo 1: UCLA - Universidad de Los Ángeles, California.
- Nodo 2: SRI - Instituto de Investigaciones de Stanford.
- Nodo 3: UCSB Universidad de California en Santa Bárbara.
- Nodo 4: Universidad de Utah.

Dos años después, ya contaba con unos 40 ordenadores conectados. Tanto fue el crecimiento de la red que su sistema de comunicación se quedó obsoleto. Entonces dos investigadores, Robert E. Kahn y Vinton G. Cerf, crearon el Protocolo TCP/IP, que se convirtió en el estándar de comunicaciones dentro de las redes informáticas de ARPANET (actualmente seguimos utilizando dicho protocolo), al principio sòlo fue TCP el cual dio lugar a una versión que sólo permitía circuitos virtuales. Esto llevó a la reorganización del TCP original en dos protocolos: uno sencillo, IP, que se encargarà tan sólo de dar una dirección a los paquetes y de reenviarlos; y un TCP que se dedicara a una serie de funcionalidades como el control del flujo y la recuperación de los paquetes perdidos. Para aquellas aplicaciones que no precisan los servicios de TCP, se añadió un protocolo alternativo llamado UDP (User Datagram Protocol, protocolo de datagramas de usuario) dedicado a dar un acceso directo a los servicios básicos del IP. Esto genera una de las

primeras definiciones de Internet: "una serie de redes conectadas entre sí, específicamente aquellas que utilizan el protocolo TCP/IP" y se utiliza el término "Internet" como redes TCP/IP interconectadas.

Las primeras implementaciones de TCP se hicieron para grandes sistemas en tiempo compartido como Tenex y TOPS 20. Cuando aparecieron los ordenadores de sobremesa (desktop), TCP era demasiado grande y complejo como para funcionar en ordenadores personales. David Clark y su equipo de investigación del MIT empezaron a buscar la implementación de TCP más sencilla y compacta posible. La desarrollaron, primero para el Alto de Xerox (la primera estación de trabajo personal desarrollada en el PARC de Xerox), y luego para el PC de IBM. Esta implementación operaba con otras de TCP, pero estaba adaptada al conjunto de aplicaciones y a las prestaciones de un ordenador personal, y demostraba que las estaciones de trabajo, al igual que los grandes sistemas, podían ser parte de Internet.

En los años 80, el desarrollo de redes Lan, PC y estaciones de trabajo permitió que la naciente Internet floreciera. La tecnología Ethernet, desarrollada por Bob Metcalfe en el PARC de Xerox en 1973, es la dominante en Internet, y los PCs y las estaciones de trabajo los modelos de ordenador dominantes. El cambio que supone pasar de unas pocas redes con un modesto número de hosts (el modelo original de ARPANET), a tener muchas redes, dio lugar a nuevos conceptos y a cambios en la tecnología. En primer lugar, hubo que definir tres clases de redes (A, B y C) para acomodar todas las existentes. La clase A representa a las redes grandes, a escala nacional (pocas redes con muchos ordenadores); la clase B representa redes regionales; por último, la clase C representa redes de área local (muchas redes con relativamente pocos ordenadores).

Como resultado del crecimiento de Internet, se produjo un cambio de gran importancia para la red y su gestión. Para facilitar el uso de Internet por sus usuarios, se asignaron nombres a los hosts de forma que resultara innecesario recordar sus direcciones numéricas. Originalmente había un número muy limitado de máquinas, por lo que bastaba con una simple tabla con todos los ordenadores y sus direcciones asociadas.

El cambio hacia un gran número de redes gestionadas independientemente (por ejemplo, las Lan) significó que no resultara ya fiable tener una pequeña tabla con todos los hosts. Esto llevó a la invención del DNS (Domain Name System, sistema de nombres de dominio) por Paul Mockapetris de USC/ISI. El DNS permitía un mecanismo escalable y distribuido para resolver jerárquicamente los nombres de los hosts (por ejemplo, www.acm.org o www.ati.es) en direcciones de Internet.

El incremento del tamaño de Internet resultó también un desafío para los routers. Originalmente había un sencillo algoritmo de enrutamiento que estaba implementado uniformemente en todos los routers de Internet. A medida que el número de redes en Internet se multiplicaba, el diseño inicial no era ya capaz de expandirse, por lo que fue sustituido por un modelo jerárquico de enrutamiento con un protocolo IGP (Interior Gateway Protocol, protocolo interno de pasarela) usado dentro de cada región de Internet y un protocolo EGP (Exterior Gateway Protocol, protocolo externo de pasarela) usado para mantener unidas las regiones. El diseño permitía que distintas regiones utilizaran IGP

distintos, por lo que los requisitos de coste, velocidad de configuración, robustez y escalabilidad, podían ajustarse a cada situación. Los algoritmos de enrutamiento no eran los únicos en poner en dificultades la capacidad de los routers, también lo hacía el tamaño de la tablas de direccionamiento. Se presentaron nuevas aproximaciones a la agregación de direcciones (en particular CIDR, Classless Interdomain Routing, enrutamiento entre dominios sin clase) para controlar el tamaño de las tablas de enrutamiento.

ARPANET siguió creciendo y abriéndose al mundo, y cualquier persona con fines académicos o de investigación podía tener acceso a la red. Las funciones militares se desligaron de ARPANET y fueron a parar a MILNET, una nueva red creada por los Estados Unidos. La NSF (National Science Fundation) crea su propia red informática llamada NSFNET, que más tarde absorbe a ARPANET, creando así una gran red con propósitos científicos y académicos. El desarrollo de las redes fue abismal, y se crean nuevas redes de libre acceso que más tarde se unen a NSFNET, formando el embrión de lo que hoy conocemos como INTERNET.

El desarrollo de NSFNET fue tal que hacia el año 1990 ya contaba con alrededor de 100, 000 servidores.

El CERN (Centro Europeo de Investigación de Partículas) crea las páginas Web, con el objetivo de comunicarse con otros científicos europeos. En 1993 un estudiante norteamericano escribió el código del primer explorador Web, el Mosaic, que se distribuía de forma gratuita por la red, y permitía tener acceso a gráficos y documentos de texto dentro de Internet. Esto supuso una auténtica revolución, y a partir de ese momento, Internet no ha parado de crecer. En el año 1996 existían cerca de 90, 000 sitios Web.

Actualmente es casi imposible calcular los sitios Web que existen y los servidores a los que tenemos acceso. Internet se ha desarrollado en esta última década mucho, y en parte es debido a los fines comerciales de las empresas. Internet ya no es la red de investigación ni militar para lo que fue creada, ahora Internet es, ante todo, un negocio, y eso ha sido lo que ha empujado su desarrollo.

5.1.1. Crecimiento de Internet

Con la reciente comercialización del Internet, las posibilidades para transferir información han aumentado en grandes proporciones. El Internet puede ser usado para enviar texto hacia millones de personas que tienen acceso a e-mail, software de transferencia de archivos, imágenes, archivos de audio, y otra información localizada en cualquier parte del mundo, o usarlo para trabajos en colaboración con otras personas ubicadas en otro estado u otro continente.

El predecesor de Internet, el ARPANET, fue la primera red de backbone transcontinental de los Estados Unidos, la cual se involucrò en la investigación de las redes de computadoras en el àrea de packet-switched. El crecimiento de Internet creció màs allá de las expectativas de las personas que originalmente lo concibieron. Lo que iniciò como un experimento para investigación llegò a ser un éxito comercial. Por otra parte, el Internet, como parte del éxito del World Wide Web (WWW), creció tempranamente 100% por año

desde 1988. Tràfico en Internet ha crecido a tazas de 400% por año recientemente[21]. En muchas instancias, la tecnología de Internet no es unida a las demandas del público en general.

Investigadores encontraron aùn más problemas con el Internet. Investigadores de universidades e instalaciones gubernamentales que originalmente usaron el Internet, encontraron que èste no podía cubrir todas sus necesidades. Primero, el flujo de personas que usan el Internet, están causando problemas de tráfico. Los enlaces de transporte de muchos paquetes de datos de los enrutadores, presentan problemas al enviar la información rápidamente hacia su siguiente destino, como lo que sucede con el flujo de agua, cuando se introduce a un tubo màs pequeño, a esto se le ha llamado embotellamiento.

Otro problema son las limitaciones del ancho de banda. El ancho de banda disponible del medio, restringe el número de información transmitida en un tiempo dado. Por instancia, cables de fibra óptica provee mayor ancho de banda que los cables de cobre. El limitado ancho de banda causa grandes problemas para los investigadores. Tecnologías de gran ancho de banda existen, pero aùn no se encuentran completamente expandidas en Internet.

Adicionalmente, la comunidad investigadora ha encontrado particular dificultad para hacer trabajos de colaboración con otros investigadores en Internet. Ellos necesitan un Internet en el que puedan transmitir tremendas cantidades de datos. La velocidad que el Internet actual ofrece, simplemente no es lo suficientemente rápida para ameritar hacer investigación a larga distancia.

5.2. LAS INICIATIVAS PARA LA SIGUIENTE GENERACIÓN DE INTERNET

Debido a la problemática presentada en Internet, el gobierno federal y universidades investigadoras, ambas de EEUU, están desarrollando la siguiente generación de Internet. Esta siguiente generación permitirá capacidades de control y comunicaciones que son difíciles de imaginar en la conectividad limitada de hoy en día. Para el desarrollo de estas capacidades, el gobierno federal ha creado la iniciativa de la Siguiente Generación de Internet (Next Generation Internet NGI), y por otra parte 200 de las universidades de investigación màs prestigiadas de la nación de EEUU, hasta finales del 2006, han organizado sus esfuerzos en un proyecto llamado Internet2. Èsta es una red experimental de telecomunicaciones que usarà tecnología avanzada para mover información (video, voz y datos), de 100 a 1000 veces màs rápido que el Internet actual.

5.2.1. NGI (Next Generation Internet)

Como respuesta a esto, el Presidente Clinton y el Vicepresidente Gore, anuncian la iniciativa para la Siguiente Generación de Internet (NGI) en octubre de 1996, basado sobre una fuerte investigación y programas de desarrollo a travès de Agencias Federales, que eventualmente extenderá nuevas tecnologías de red y aplicaciones hacia la Internet actual. Este tiene tres principales metas:

[21] http://www.wise-intern.org/journal/1997/HEIDEMAN.PDF. Pp 9.

- Investigación: El ànimo de esta meta es promover la experimentación con la siguiente generación de tecnologías de red
- Red de Pruebas: El ànimo de esta meta es desarrollar una red de prueba de siguiente generación para conectar universidades e instituciones federales de investigación a velocidades que sean suficientes para demostrar nuevas tecnologías y que soporte futuras investigaciones
- Aplicaciones: El ànimo de esta meta es demostrar nuevas aplicaciones, disponibles por la redes NGI, que reúna misiones y metas nacionales importantes

5.2.2. Internet2

Internet2 fue formada por 34 universidades EEUU de investigación reunidas en un hotel de Chicago el 1 de octubre de 1996, comisionada para establecer un proyecto que adoptara el desarrollo de las capacidades de red que podría, no solamente soportar investigación y educación, también podrían eventualmente hacer mejoras dentro del Internet comercial global.

En octubre de 1997, las 120 universidades y empresas participantes a la fecha en el proyecto Internet2 formalizaron su compromiso creando la "Corporación Universitaria para el Desarrollo Avanzado de Internet" (UCAID). La preside el profesor estadounidense Douglas E. Van Houweling, de la Universidad de Michigan. Aunque también participan empresas del rubro de la informática y de las telecomunicaciones.

Hay que mencionar que el nombre de Internet2 que se le ha dado a las redes avanzadas ha sido manejado por el proyecto de EEUU, y en otras redes como DANTE en Europa son denominadas NREN (National Research and Education Networks) y básicamente son denominadas como las responsables, sobre una nación, de proveer redes de comunicación y servicios para la comunidad de educación e investigación de su país. La red NREN típicamente conecta otras redes a nivel regional o metropolitano[22]. Ambos términos generalmente son manejados sin distinción, ya que las dos hacen referencia a lo mismo.

El proyecto Internet2 (I2), seguido por comunidades de investigación y educación desde 1996, es un esfuerzo de colaboración sin ánimo de lucro para desarrollar y desplegar aplicaciones avanzadas de red y tecnología vitales para las misiones de investigación y educación de las instituciones de educación superior, buscando de esta forma acelerar la creación del Internet del mañana y fomenta entre la sociedad Académica, Industria y gobierno los días de Internet en sus inicios.

La comunidad de Internet2 esta:
- permitiendo una nueva generación de aplicaciones de red,
- creando una red con capacidades avanzadas para investigación y educación, y
- fomentando la transferencia de tecnología y experiencia al Internet global

[22] http://www.dante.net/server/show/conWebDoc.396

La comunidad de Internet2, liderado por más de 200 universidades estadounidenses, màs de 70 compañías y más de 40 organizaciones afiliadas, incluyendo laboratorios de investigación del gobierno EEUU, están trabajando con más de otras 30 redes similares de organizaciones de educación e investigación en países alrededor del mundo. Miembros de Internet2 en acuerdo con iniciativas nacionales, de estado y regional en los Estados Unidos, son coordinados por organizaciones internacionales tal como la IETF (Internet Engineering Task Force). Los esfuerzos de Internet2 son enfocados en:

- *Aplicaciones de Red Avanzados* están permitiendo la colaboración entre la gente, así como el acceso interactivo a la información y recursos de forma que no podrían ser posibles en el Internet comercial de hoy en día. Aprendizaje a distancia interactivo, acceso remoto a únicos instrumentos científicos, tiempo de acceso real a grandes bases de datos, y alta definición de video-streaming, son todos posibles con el alto desempeño de las redes.
- *Nuevas Capacidades de Red* tal como la Calidad de Servicio (QoS), Multicast, e IPv6 están siendo desarrolladas y probadas intensamente en la redes usadas por los miembros de Internet2. Estas capacidades soporta aplicaciones avanzadas de red de hoy en día, y llegaran a ser disponibles en el Internet comercial del mañana para proveer confiable desempeño que requieran aplicaciones avanzadas.
- *Middleware*, el software behind-the-scenes, està proveyendo seguridad en directorios y otros servicios requeridos por aplicaciones de red avanzadas. En el Internet de hoy, las aplicaciones usualmente tienen que proveer estos servicios ellos mismos, lo cual entorpece la competencia y la incompatibilidad de estándares. Para promover la estandarización y la interoperabilidad, middleware hará aplicaciones de red avanzadas más fáciles de usar.
- *Redes de alto desempeño* están enlazando los campus y laboratorios de más de 200 instituciones miembros de Internet. Las redes de alto desempeño que participan en el proyecto Internet2 proveen el ambiente en la cual nuevas aplicaciones de red y capacidades pueden ser desarrolladas y probadas.

El proyecto de las universidades “Internet2” y las federales “NGI”, son iniciativas paralelas y complementarias en los Estados Unidos de América. Internet2 y NGI ya están trabajando juntas en muchas áreas. Por ejemplo, con la participación en un programa NSF y NGI, más de 150 universidades de Internet2 han recibido competitivamente concesiones para soportar conexiones al backbone de redes avanzadas tal como Abilene y al backbone de servicios de red de alta velocidad (vBNS).

Internet2 està también formando sociedades con iniciativas de redes avanzadas similares alrededor del mundo. Trabajando en conjunto ayudarà a asegurar una cohesiva e interoperable infraestructura de red avanzada para investigación y educación, y la continua interoperabilidad del Internet global.

Es necesario dejar muy claro que Internet2 no quitarà a la Internet que existe actualmente, ya que estas redes trabajan en paralelo, para que en Internet2 se desarrollen los proyectos sòlo de investigación, evitando que entren aplicaciones del sector privado o pùblico ajenas a los enfoques de educación y/o investigación; esta ventaja permitirá obtener los resultados

màs rápido y eficientemente y de esta manera poderlos divulgar lo màs pronto posible a todo el mundo, para que puedan disponer de los mismos y continuar con la búsqueda de soluciones a problemas prioritarios en distintas áreas del conocimiento.

5.2.2.1.Topología de la Red de Internet2 de EEUU

La red de Internet2 está compuesta por redes principales o backbones ubicadas geográficamente en EEUU, a los cuales se conectan los llamados gigaPoPs y backbones internacionales de la misma forma a su vez se conectan gigaPoPs o nodos tales como Universidades. Un gigaPoP es una red regional (con ancho de banda del orden de los gigabits por segundo) conectada a Internet2. Por ejemplo en EEUU el MIT, la Universidad de Boston y la Universidad de Harvard conforman el gigaPoP llamado BOS.

A continuación la se muestra un esquema (muy generalizado) de Internet2:

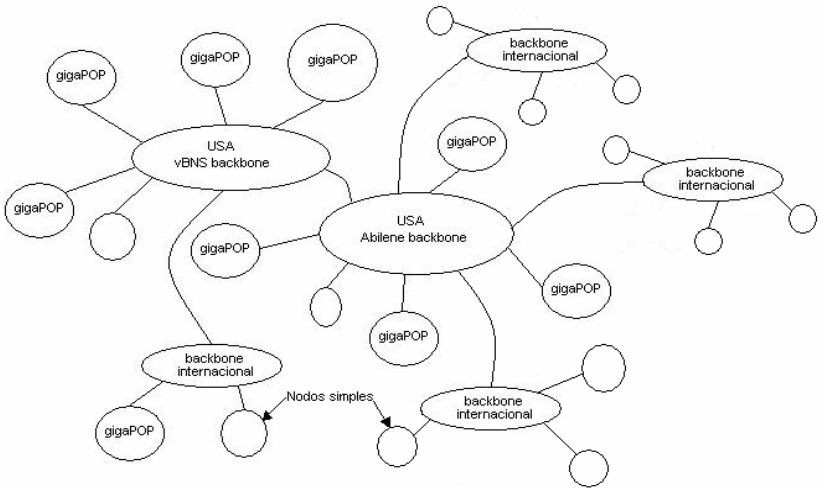


Figura 124. Esquema generalizado de la red de Internet2.

En la figura 124 se puede visualizar que actualmente existen dos grandes backbones en USA (aunque hoy en día el backbone Abilene es mucho mayor en ancho de banda, 10 Gbps con la meta de ofrecer 100 Gbps y con IPv6 nativo), de los cuales se distribuyen enlaces hacia backbones en otros países. Uno de estos backbones internacionales es CUDI (Corporación Universitaria para el desarrollo de Internet).

Abilene es una red dual stack (IPv4/IPv6) de alto desempeño desarrollada por Internet2 en sociedad con Juniper Networks, Nortel Networks, Qwest Communications, y la Universidad de Indiana. Una meta importante del proyecto Abilene es el proveer una red de backbone para Internet2. Abilene usa enrutadores IP de alto desempeño que son el acceso a los gigaPOPs localizados en algunas docenas de ubicaciones en toda la nación, para soportar la infraestructura de Internet2. Abilene dispone de facultades y personal de universidades de Internet2 y laboratorios de investigación para desarrollar aplicaciones y servicios de red avanzados.

La vBNS es una red proveída bajo una extensión de un acuerdo de cooperativa entre la National Science Foundation (NSF) y Worldcom para aprobar instituciones NSF de investigación con el más alto aprendizaje. Èsta continúa jugando un rol clave en el proyecto de Internet2. Abilene provee una alternativa complementaria para las vBNS y otras redes de investigación. Cada gigaPOP o conexiones de instituciones realiza su propia decisión acerca de cual red se provee las capacidades necesarias de aplicaciones avanzadas desarrolladas en su campus.

La figura 125 muestra la situación, hasta la fecha de la recopilación de la información, del backbone de la red de Internet2 de EEUU con sus conexiones internacionales y nacionales, así como los miembros que la conforman conectados a ella y las velocidades de conexión que maneja. Por las dimensiones del mapa es imposible verse a detalle en la figura, pero para mayor detalle el mapa puede ser encontrado en formato PDF con el nombre January2007-AbileneMap en un disco anexo a la tesis o también el publicado en la página del proyecto de Internet2 de EEUU.

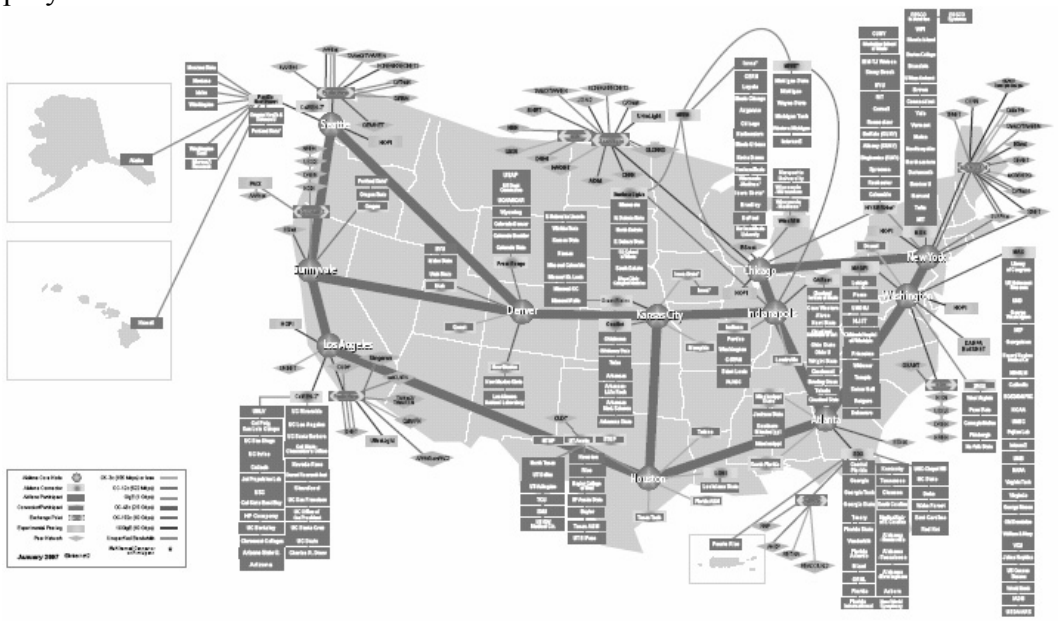


Figura 125. Red de Internet2 de los EEUU.

5.2.2.2.Grupos de Trabajo

Los grupos de trabajo (Working Groups "WG") son una parte integral de la comunidad de Internet2. Internet2 desarrolla actividades que toman lugar dentro de los grupos de trabajo, y por medio de ellos es por donde se provee el soporte a la red. Típicamente la meta de los grupos de trabajo podría incluir:

- Establecimiento de proyectos y reuniones
- Distribución de carga de trabajo entre los miembros
- Desarrollar documentación que haga cronologías del progreso del WG
- Publicar o enlazar a documentos apropiados en el sitio Web de Internet2
- Asegurar la implementación de proyectos y documentos

Algunos WGs son abiertos a todos los miembros de la comunidad de Internet2. Unos pocos son abiertos solamente a miembros universitarios o por invitación del responsable del WG, donde la membresía necesita ser limitada a causas de la naturaleza del trabajo. La tabla 33 muestra los WG's que actualmente tiene el proyecto Internet2.

Tabla 33. Grupos de trabajo de Internet2

Grupo de Trabajo	**Grupo de Trabajo**
Bulk Transport WG	perfSONAR WG
Grouper WG	Presence and Integrated Communications
HEPKI-TAG (PKI Technical) WG	ResearchChannel WG
IPv6 WG	Salsa-Computer Security Incidents - Internet2 (Salsa-CSI2)
MACE-Directories (MACE-Dir) WG	Salsa-Federated Wireless NetAuth (Salsa-FWNA)
MACE-Shibboleth WG	Salsa-NetAuth WG
Multicast WG	Signet WG
Orthopaedic Surgery WG	SIP.edu

5.2.2.2.1. Grupo de Trabajo IPv6

El grupo de trabajo IPv6 de Internet2 ha desarrollado con éxito una red IPv6 dentro de la infraestructura Abilene, èsta, a su vez se ha podido integrar con otras redes semejantes de gran escala. Dentro de los propósitos del grupo de trabajo es hacer de IPv6 una herramienta segura para la comunidad de Internet2, y al mismo tiempo, apoyar al trabajo que se hecho en el Internet global en materia de IPv6 por medio de la realización de pruebas, impulsando la difusión y mejorando la infraestructura IPv6 y software en el contexto de Internet2.

vBNS+ y Abilene operan IPv6. El backbone IPv6 Abilene esencialmente es el mismo que el backbone de Abilene IPv4, aunque no todos los conectores tengan habilitado IPv6. La figura 126 muestra el backbone IPv6 de Abilene, para mayores detalles el mapa puede ser encontrado en el disco anexo a la tesis con el nombre AbileneBackboneIPv6 o en la pagina de Internet2.

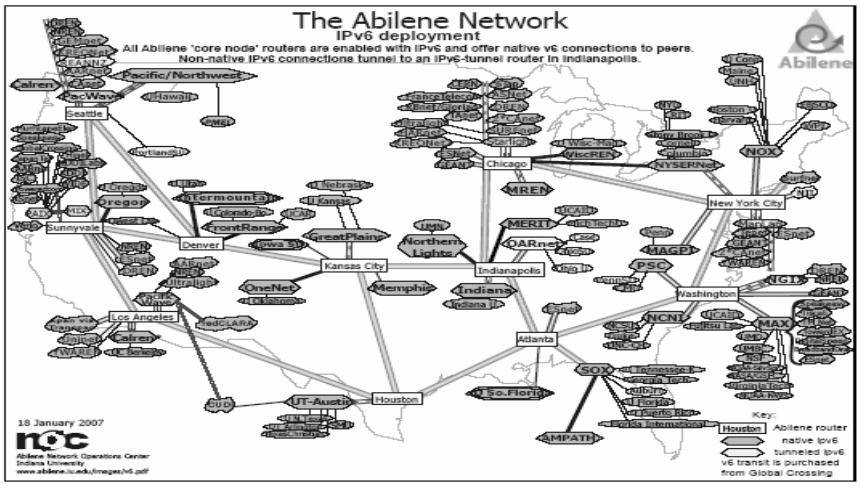


Figura 126. Backbone IPv6 de Abilene

La forma en que se puede establecer una conexión por medio de IPv6, es mediante el enrutador que brinda la conexión hacia Abilene o conector Abilene upstream, con base en las siguientes dos condiciones: si este soporta IPv6, la universidad podrá estar en la disposición de conectarse nativamente, y si el conector Abilene upstream no soporta IPv6, se podría establecer la conexión mediante un túnel IPv6-in-IPv4 entre un enrutador IPv6 ubicado en el campus de la universidad y el enrutador Abilene IPv6 de túneles en Indianápolis, Indiana. Un conector IPv6 necesita correr Multiprotocol-BGP (MBGP) con Abilene.

El espacio actual de direcciones IPv6 Abilene de producción (sTLA) consiste de un bloque /32 de direcciones (2001:468::/32) y un pTLA para pruebas de 6bone (3ffe:3700::/24) que no volverá a usarse por razones de la conclusión del proyecto de 6bone. Abilene asigna bloques de direcciones /40 a conectores y bloques de direcciones /48 a participantes directamente conectados[23] de su bloque de producción, para màs detalles visitar el sitio del WG de IPv6.

5.2.2.3.Memorandums de Entendimientos Internacionales de Internet2

Internet2 ha formado relaciones, a nivel de peer, con organizaciones fuera de EEUU que tienen proyectos similares a Internet2 en ámbitos y objetivos. Actualmente Internet2 ha formado sociedad con más de 50 organizaciones internacionales y redes semejantes. La tabla 34 muestra las redes con la que ha establecido sociedad.

Tabla 34. MoUs de Internet2

América		**Europa y el medio oriente**	
CANARIE	Canadá	ARNES	Eslovenia
CLARA	Latino América y el Caribe	BELNET	Bélgica
CEDIA	Ecuador	CARNET	Croacia
CNTI	Venezuela	CESnet	Repùblica Checa
CR2Net	Costa Rica	DANTE	Europa
CUDI	México	DFN-Verein	Alemania
REUNA	Chile	FCCN	Portugal
Retina	Argentina	GARR	Italia
RNP (FAPESP)	Brasil	GIP-RENATER	Francia
SENACYT	Panamá	GRNET	Grecia
Asia y Pacifico		HEAnet	Irlanda
AAIREP	Australia	HUNGARNET	Hungría
APAN	Asia-Pacìfico	Israel-IUCC	Israel
ANF	Corea	MCIT(EUN/ENSTINET)	Egipto
APRU	Singapur	NORDUnet	Dinamarca
CERNET, CSTNET, NSFCNET	China	PSNC/PIONIER (Fue POL-34)	Polonia
CDAC, ERNET	India	Qatar Foundation	Qatar
JAIRC	Japón	RedIRIS	España
JUCC	Hong Kong	RESTENA	Luxemburgo
MYREN/MDeC	Malasia	RIPN	Rusia
NECTEC/UNINET	Tailandia	SANET	Eslovaquia

[23] http://ipv6.internet2.edu/Abilene_IPv6_Addressing.shtml

NGI-NZ	Nueva Zelanda	Stichting SURF	Países Bajos
SingAREN	Singapur	SWITCH	Suiza
TANet2	Taiwán	TERENA	Europa
África		JSIC, UKERNA	Reino Unido
TENET	Sudáfrica		

5.3. REDES AVANZADAS EN EL MUNDO

En términos generales, las Redes Avanzadas se agrupan en el mundo de acuerdo a zonas geográficas. Así, las Redes Nacionales de Investigación y Educación (NREN) o Redes Avanzadas de cada país, van integrando consorcios que no son otra cosa que redes mayores, integradas en una gran troncal (backbone). Estas redes mayores, a su vez establecen Memorandums de Entendimiento (MoU) o asociaciones que les permiten interconectarse, permitiendo la interconexión total de las Redes Avanzadas.

A continuación se presenta las tablas de las Redes Avanzadas en el mundo, de acuerdo a zonas geográficas, consorcios y países[24]:

Tabla 35. Redes Educativas Avanzadas de la Región Asia-Pacifico

ACSys	Advanced Computational Systems	Australia
AARNET	Australia's Academic and Research Network	Australia
ANF	Advanced Network Forum	Corea
APRU	Association of Pacific Rim Universities	Singapur
APAN	Asia Pacific Advanced Network	Tailandia
CERNET	China Education and Reserach Network	China
WIDE	Widely Integrated Distributed Environment	Japón
NECTEC	Towards a Knowledge-Based Economy/Society	Tailandia
NGINZ	Next Generation Internet	Australia
SingAREN	Singapore Advanced Research and Education Network	Singapur
TANeT	Taiwan Academic Network	Japón
CSTNET	Chinese Scientific and Technoloy NETwork	China
NSFCNET	The National Natural Science Foundation of China	China
JAIRC	Profile of Japan Advanced Internet Research Consortium	Japón
JUCC	The University of Hong Kong	Hong Kong
NGI-NZ	Next Generation Internet in New Zealand	Australia
UNINET	Inter-University Network	Tailandia

Tabla 36. Redes Educativas Avanzadas de la Región Europa-Medio Oriente

ARNES	Academic and Research Network of Slovenia	Eslovenia
BELNET	Belgisch National Onderzoeksnetwerk	Bulgaria
CARNET	Croatian Academic and Research Network	Croacia
CESnet	Czech NREN operator	República Checa
DANTE	Delivery of Advanced Network Technology to Europe	Europa
DFN-Verein	Deutschen Forschungsnetz	Alemania
GARR	Consortium GARR	Italia
GRNET	Greek Research and Technology Network	Grecia
GÉANT	pan-European research and education network	Inglaterra

[24] http://www.cudi.edu.mx/internacional/main.html
http://www.reuna.cl/joomla/index.php?option=com_content&task=view&id=120&Itemid=144

HEAnet	Ireland's National Education and Research Network	Irlanda
HUNGARNET	The Hungarian Academic Computer Network	Hungría
INFN	National Institute of Nuclear Physics	Italia
Israel-IUCC	Internet 2 in Israel	Israel
IUCC	Inter-University Computation Center	Israel
JISC	The Joint Information Systems Committee	Inglaterra
NORDUnet	Nordic Internet	Dinamarca
POL 34	Polish Electrical Power Network Telecommunication Operator	Polonia
REDIRIS	Red Interconexión de los Recursos InformáticoS	España
RENATER	Réseau National Télécommunications	Francia
RESTENA	Réseau Téléinformatique de l'Education Nationale et de la Recherche	Luxemburgo
RIP	Russian Institute for Public Networks	Rusia
SANET	Slovak Academic Network	Eslovaquia
StichtingSURF		Países Bajos
SWITCH	Swiss Education & Research Network	Suiza
TERENA	Trans-European Research and Education Networking Association	Países Bajos
UKERNA	United Kingdom Education and Research Network Association	Reino Unido
GRNET	Greek Research and Technology Network	Grecia
PSNC/PIONIER	Poznan Supercomputing and Networking Center	Polonia

Tabla 37. Redes Educativas Avanzadas de la Región Norteamérica

Abilene	Advance Networking for Leading-Edge Research and Education	USA
Canarie	Canada's Advanced Internet Development Organization	Canadá
CENIC	Corporation for Education Network Initiatives in California	USA
UCAID	University Corporation for Advanced Internet Development	USA
vBNS	very High Performance Backbone Network Service	USA

Tabla 38. Redes Educativas Avanzadas de la Región de Sudamérica

ANSP	Academic Network at São Paulo	Brasil
Arandu	Arandú	Paraguay
ADSIB	Bolivia (Ex BolNet Red Boliviana de Comunicación de datos	Bolivia
CLARA	Cooperación Latinoamericana de Redes Avanzadas	Latinoamérica
CR2NeT	Red Nacional de Investigación	Costa Rica
CEDIA	Consorcio Ecuatoriano para el desarrollo de Internet Avanzado	Ecuador
FCCN	Fundación para la Computación Científica Nacional	Brasil
FUNDACYT	Fundación para la Ciencia y Tecnología	Ecuador
RAGIE	Red Avanzada Guatemalteca de Investigación y Educación	Guatemala
RAICES	Red Avanzada de Investigación	El Salvador
RAP	Red Académica Peruana	Perú
RAU	Red Académica de Uruguay	Uruguay
RACCIUN	Red Académica de Centros de Investigación y Universidades Nacionales de Alta Velocidad	Venezuela
RETINA	REd TeleINformática Académica de Argentina	Argentina
REUNA	Red Universitaria Nacional de Chile	Chile
RNP	Rede Nacional de Ensino e Pesquisa	Brasil
UNICAUCA	Red Académica Universitaria de Colombia	Colombia
RedUniv	Red Universitaria de la Republica de Cuba	Cuba

Tabla 39. Redes Educativas Avanzadas de la Región de África

KENET	Kenya Education Network	Kenya
MAREN	Malawi Research and Education Network	Malawi
MCIT	Ministry of Communications and Information Technology	Egipto
RENU	Research and Education Network for Uganda	Uganda
TENET	Tertiary Education Network	Sudafrica

5.4. DANTE

Dante (Delivery of Advanced Network Technology to Europe), fue establecida en Reino Unido Cambridge en 1993, construye y opera redes avanzadas para investigación y educación, sin fines lucrativos. Es propiedad de NREN's Europeos, y trabaja en sociedad con ellos y en cooperativas con comisiones Europeas. DANTE provee la infraestructura de comunicación de datos esencial para el desarrollo de la comunidad de investigación global.

Durante los primeros años de la existencia de DANTE, RARE (Réseaux Associés pour la Recherche Européenne/European ASsociation of Research Networks) fue el propietario legal y único accionista. El 25 de marzo de 1994, la propiedad de la compañía fue formalmente transferida a once organizaciones de investigación de red nacionales. Desde entonces, han sido agregadas sòlo cuatro a las lista de accionistas: CESNET, HEANet, RENATER y RESTENA.

Estas quince organizaciones accionistas, listadas en la tabla 40, son NREN´s o cuerpos gubernamentales que financia la NREN de sus países. Usualmente un departamento de gobierno es un accionista de DANTE en vez del NREN de su país, esto es porque el estatus legal de la NREN en sus leyes nacionales no permite compartir propiedad con otra organización.

Tabla 40. NRENs Accionistas de DANTE[25]

Organización	**País**
ARIADNET(GRNET)	Grecia
ARNES	Eslovenia
CESNET	Republica Checa
DFN	Alemania
FCCN	Portugal
GARR	Italia
HEAnet	Irlanda
HEFC-E en nombre de JISC(UKERNA/JANET)	Reino Unido
HUNGARNET	Hungría
NORDUnet	Países nórdicos (Dinamarca, Finlandia, Islandia, Noruega y Suecia)
RedIRIS	España
RENATER	Francia
RESTENA	Luxemburgo
SURFnet	Países Bajos
SWITCH	Suiza

[25] http://www.dante.net/server/show/nav.205

El propósito de DANTE es planear, construir y operar redes de investigación pan-Europeas. Desde que fue establecido juega un rol importante en las cinco generaciones de redes consecutivas de investigación pan-Europeas: EuropaNET, TEN-34, TEN-155, GEANT que pasa a ser GEANT2.

Redes de investigación pan-Europea están organizadas de forma que provea una red de backbone pan-Europea la cual conectarà las redes de investigación nacional de cada país Europeo.

Las actividades de DANTE típicamente incluyen:

- Dirección de proyectos.
- Desarrollar infraestructura de comunicación de datos.
- Determinación de la viabilidad del proyecto.
- Seguimiento de ejercicios.
- Tecnología de red de investigación.
- Desarrollo y disposición de los servicios de red.
- Difusión de la información y ayuda a clientes.

5.4.1. Proyectos de DANTE

La mayoría de trabajos desarrollados por DANTE es organizada en la forma de proyectos. Los proyectos usualmente reciben co-financiamiento de Comisiones Europeas y pueden durar típicamente entre 18 meses y 4 años. Una condición para recibir financiamiento es que las actividades de investigación deben ser realizadas por identidades de al menos tres estados miembros de la Unión Europea. DANTE ha estado involucrado en tales proyectos desde hace varios años. Estos proyectos típicamente involucran numerosos socios alrededor de Europa de NREN´s Europeos (usualmente algunas o muchas), y a veces también de otras organizaciones. Los socios en cada proyecto deben seleccionar una organización para actuar como el administrador o coordinador.

La tabla 41 da una breve descripción de los proyectos que administra actualmente DANTE y que ha administrado, así como con los que ha estado involucrado.

Tabla 41. Proyectos DANTE

Proyecto	Descripción
ADMINISTRADOS ACTUALMENTE	
GÉANT2	La principal actividad del proyecto GEANT2 es la operación y administración de la red multi-gigabit de comunicaciones de datos pan-Europea (también conocida como GEANT2) la cual es reservada especialmente para uso de investigación y educación. Paralelamente a la red, el proyecto està corriendo una programa integrado, para reunir actividades de investigación dentro de tecnologías y servicios de red.
GEANT (GN1)	Anteriormente a GEANT2, la principal actividad de DANTE fue la operación y administración de GEANT, la generación anterior de la red de educación e investigación.
ALICE	Un proyecto para desarrollar una infraestructura de red IP dentro de la región de Latino América para conectarse hacia Europa.

EUMEDCONNECT	Iniciativa pionera para desarrollar una infraestructura de red IP dentro de la región Mediterránea.
TEIN2	El propósito de TEIN2 es desarrollar una infraestructura de red IP a través de la región Asia-Pacìfico, y entre Asia y Europa.
ORIENT	Es un proyecto colaborativo de Sino-European para conectar las redes de investigación y educación de China y Europa.
ADMINISTRADOS ANTERIORMENTE	
QUANTUM	Un proyecto que desarrollò la tercera generación de red TEN-155 entre 1998 y 2001.
TEN-34	Proyecto que desarrollò la segunda generación de red de investigación pan-Europea, cual fue llamada TEN-34 en 1997.
EuropaNET	Proyecto que corrió de 1990 a 1997 y conectò redes universitarias nacionales a través de Europa, así como también ofreció conectividad completa al Internet global.
6NET	Proyecto para demostrar còmo la tecnología IPv6 tiene la capacidad para continuar con el crecimiento de Internet.
6LINK	Proyecto que aspira fortalecer un mejor entendimiento de las más importantes actividades para el desarrollo y despliegue de IPv6.
CAESAR	Estudio para evaluar la posibilidad de una interconexión directa entre GEANT y redes de investigación de Latino América.
CAPE	Estudio de la viabilidad para la conectividad entre Europa y la región Asia-Pacifico.
Q-Med	Un proyecto para desarrollar servicios de red de alta calidad hacia las NRENs de Israel y Chipre.
PHARE	Proyecto para conectar once ciudades del Centro y Este de Europa hacia el resto de las redes de investigación Europeas.
SEQUIN	Proyecto animado a definir e implementar un end-to-end aproximado a la Calidad de Servicio (QoS) que podría operar a través de múltiples administradores de dominios y explotar una combinación de tecnologías ATM e IP.
SEEREN	Iniciativa animada a proveer conectividad hacia GEANT a aquellas ciudades del sudeste de Europa que no eran socios del proyecto GEANT. SEEREN fue adminstrada por el GRNET.
SERENATE	Estudio estratégico dentro de las redes de educación e investigación Europeas dirigido a Europa.
SEEFIRE	Este proyecto construyò una base de conocimiento substancial acerca del estado de la infraestructura de la red en la región del sur y el este de Europa, y los métodos potenciales para reducir la división digital que afecta muchos países en estas zonas.
EN LOS QUE HA ESTADO INVOLUCRADO	
EGEE	El proyecto Enabling Grids for E-sciencE (EGEE) su principal objetivo es construir avances recientes de tecnología grid para proveer una infraestructura de producción grid en Europa.
SEEFIRE	El “South-East Europe Fibre Infraestructure for Research and Education” investigarà las opciones disponibles de infraestructura de red, y las estrategias para el desarrollo de redes de investigación y educación en el Sudeste de Europa.
EARNEST	“Education and Research Networking Evolution Study” es el sucesor del proyecto SERENATE.

5.4.2. GEANT2

GÉANT2 además de ser la séptima generación de redes de investigación y educación pan-Europea, es la red más grande que se ha construido nunca para la comunidad académica

europea, y se ha adaptado a las necesidades crecientes de los investigadores de Europa proporcionando recursos mejorados para manejar las grandes cantidades de datos que generan algunas iniciativas de investigación científica avanzada, sin deteriorar el servicio de alto rendimiento que se le proporciona a la comunidad en general.

GÉANT2, establecida oficialmente el 1 de septiembre de 2004, sucesora de la red GEANT, ofrece a las instituciones europeas acceso a un mayor conjunto de recursos y conocimientos especializados a través de conexiones de alta capacidad con organizaciones de investigación en Norteamérica, Japón, América Latina, el Mediterráneo, Oriente Medio y Sudáfrica, y una inminente conexión a la región de Asia Pacífico.

El proyecto GEANT2 es una colaboración entre 30 NREN, y dos grandes organizaciones DANTE y TERENA. Las 30 NREN participantes se conectan directamente a la red. Una de las 30 NREN es NORDUnet, la regional NREN Nordica, la cual conecta Dinamarca, Finlandia, Islandia, Noruega y Suecia. La tabla 42 muestra las NREN conectadas a red GEANT2.

Tabla 42. NRENs conectadas a la red GEANT2.

NREN	**País**	**NREN**	**País**
ACOnet	Australia	LATNET	Letonia
ARNES	Eslovenia	LITNET	Lituania
BELNET	Bulgaria	NIIF	Hungría
CARNet	Croacia	NORDUnet	Nórdicos
CESNET	Checoslovaquia	PSNC	Polonia
CYNET	Chipre	RedIRIS	España
DFN	Alemania	RENATER	Francia
EENet	Estonia	RESTENA	Luxemburgo
FCCN	Portugal	RoEduNet	Rumania
GARR	Italia	SANET	Eslovaquia
GRNET	Grecia	SURFnet	Países Bajos
HEAnet	Irlanda	SWITCH	Suiza
ISTF	Bulgaria	UKERNA	Reino Unido
IUCC	Israel	ULAKBIM	Turquía
JSCC	Rusia	University of Malta	Malta

La red GEANT2 ofrece servicios básicos IP, complementándolos con servicios avanzados, ofreciendo garantía en el ancho de banda y niveles de calidad de servicio (QoS). El portafolio de servicios proveídos por la red son los siguientes, para más detalles visitar www.geant2.net

- VPN (Virtual Private Network), es una red construida sobre una infraestructura común de red, èsta ofrece la posibilidad a un usuario de ser una red privada y dedicada sin la necesidad de invertir en una infraestructura especial dedicada.
- IPv6, es el protocolo diseñado por la IETF para remplazar al actual IPv4. Como con GEANT, GEANT2 inicialmente opero con servicios de dual-stack IPv4/IPv6, èste ha estado en completa producción en GEANT desde octubre de 2003, NRENs conectados a GEANT actualmente ofrecen IPv6 como un protocolo nativo de CORE en la red.

- Multicast, ofrece mejores características para la entrega de paquetes. Èste entrega tràfico eficientemente cuando los datos deben ser enviados de un origen a múltiples usuarios y de múltiples orígenes a múltiples usuarios, enviando sòlo el tràfico a los usuarios que están interesados en èl. El servicio Multicast que fue piloteado en GEANT ahora también es disponible en GEANT2. La tecnología que ha sido usada para hacer esto, aùn esta en desarrollo, pero es considerada suficientemente estable para proveer servicios de producción. Multicast permite proveer facilidades de videoconferencias y facilidades de ecuación a distancia a los usuarios de la red.
- MPLS (Multiprotocol label switching), es una tecnología usada para levantar trayecto de etiquetas switcheadas (label-switched paths "LSPs") sobre una red IP. El etiquetado aplicado a los paquetes de datos usando MPLS, habilita a los nodos de hacer una decisión instantánea hacia donde los paquetes deben ser reenviados.

5.4.2.1.Topología de la red GEANT2

La infraestructura de comunicación de datos que sirve a la comunidad de educación e investigación en Europa està organizada en forma jerárquica formando una red de redes. Por ejemplo, la comunicación de datos entre dos usuarios puede viajar a travès de redes a nivel de campus, después a nivel regional, y después viajar a un nivel nacional antes de pasar por la red pan-Europea GEANT2, y de esta misma forma pero de forma descendente hasta llegar al otro usuario final. La figura siguiente muestra la estructura que sigue la red GEANT2.

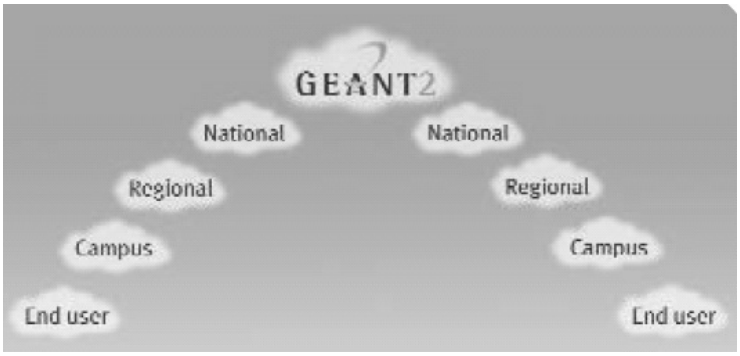


Figura 127. Estructura de la red GEANT2

La red GEANT2, mostrada en la figura 128, representa la séptima generación de redes de investigación y educación pan-Europea. Para mayores detalles el mapa puede ser encontrado en el disco anexo a la tesis con el nombre Geant2Backbone o en la página de Geant2.

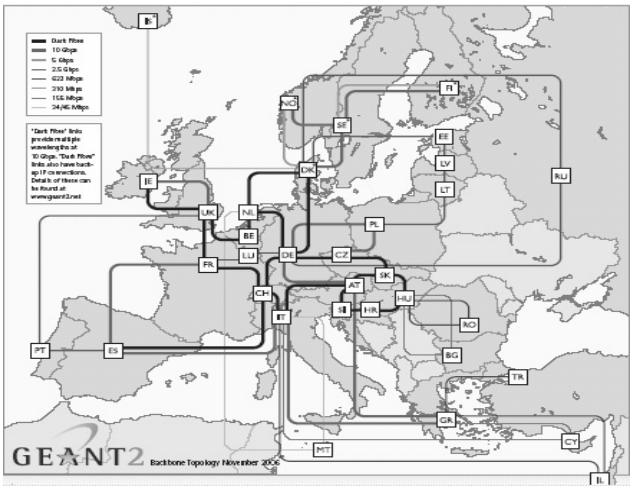


Figura 128. Red GEANT2

El diseño de la red GEANT2 se ha enfocado en maximizar la operación y flexibilidad en los servicios. La arquitectura de la red hace uso de una estructura flexible basada en la combinación de enrutadores IP y switches.

Especificaciones de la red

- Con un total de 50, 000 Km. de red
- Red basada en múltiples enlaces de 10 Gbps
- Innovadora red híbrida basada en switches y enrutadores IP
- Conexiones Point to Point crean "virtual private network" para la demanda de los usuarios
- Conecta 34 países a trabes de 30 NRENs
- 12, 000 Km. de fibra obscura, para mayor control en el desempeño y los costos
- Extensa conexión internacional hacia otras regiones del mundo.

El proyecto de la red GEANT2 casi ha finalizado. 39 de los 44 enrutadores están completamente instalados y operando. El primer enlace entrò en servicio e iniciò en Diciembre de 2005, entre Suiza e Italia, y entre Suiza y Alemania. 14 de las 18 rutas de fibra obscura están en operación. 24 de los 25 puntos de presencia (PoPs) están completamente equipados y en servicio. Múltiples Wavelenghts 10 Gbps estarán empleadas en el CORE de la red.

La nueva red operará hasta el 2008, y planea tener lista la plataforma para desarrollar la siguiente generación de red.

5.4.2.2.IPv6 en GEANT2

Como se mencionò anteriormente, la red GEANT2 inicialmente operò con servicios de dual-stack IPv4/IPv6, y actualmente ya se encuentra en operación como protocolo nativo del CORE. Para el manejo del protocolo de enrutamiento, IGP se ha migrado de OSFP a

IS-IS para tener soporte en ambas versiones del protocolo (IPv4 e IPv6). Para el protocolo de enrutamiento BGP se manejan sesiones TCP separadas para IPv4 e IPv6.

El plan de numeración IPv6 que se ha manejado tiene la siguiente estructura:

2001:0798/32 (sTLA) Asignado a DANTE por RIPE
2001:0798:0/35 Para GÉANT2
2001:0798:00/40 GÉANT2 Core Backbone
2001:0798:4/35 Delegación de /40, /48 para proyectos
2001:0798:E/35 Reservado para migración

8 rangos de /36 reservado para delegación a NREN

5.4.2.3.Conectividad Global de GEANT2

Redes de investigación alrededor del mundo usan una diversa gama de tecnologías, procedimientos y propuestas operacionales lo que ha motivado al establecimiento de una red a escala global para la investigación y la cooperación internacional. De tal forma GEANT2 se ve en la tarea de consolidar arreglos existentes, relacionándolos y ampliándolos para desarrollar un concepto integrado para la interconexión global.

Las siguientes regiones han establecido enlaces con GEANT2 la cual ha ayudado a desarrollar a NREN dentro de estas regiones.

- Sureste y Este de Europa a través de SEEREN2
- El Mediterráneo a través de EUMEDCONNECT
- Latino América a través de ALICE
- La región Asia-Pacifico a través de TEIN2.

La figura 129 muestra los enlaces establecidos con las regiones a nivel global con los que ha establecido acuerdos GEANT2. Para mayores detalles el mapa puede ser encontrado en el disco anexo a la tesis con el nombre GlobalConnectivityGeant2 o en la página de Geant2.

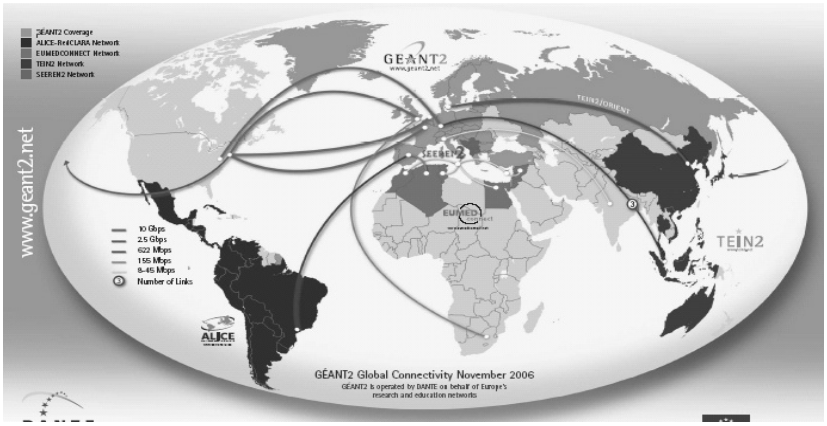


Figura 129. Conectividad Global de GEANT2.

5.5. SEEREN2

La iniciativa SEEREN2 (South East European Research and Education Networking), construirá acciones cooperativas y complementarias, que envuelva la participación de ocho organizaciones que representan las NRENs de Albania, Bosnia Herzegovina, Bulgaria, anterior Repùblica Yugoslava de Macedonia, Grecia, Hungría, Serbia-Montenegro, Rumania. De las ocho NRENs, cuatro están conectadas a GEANT2 (Grecia, Hungría, Rumania y Bulgaria), y las otras cuatro no lo están, en consecuencia son las principalmente beneficiadas de SEEREN2.

La figura 130 muestra el backbone de SEEREN2 y sus conexiones a GEANT2.

Figura 130. Backbone de SEEREN2

5.6. EUMEDCONNECT

El proyecto EUMEDCONNECT es pionero de la iniciativa de establecer y operar una red basada en IP en la región del Mediterráneo. La red EUMEDCONNECT sirve a las comunidades de educación e investigación de la región, y èsta es enlazada a la red Pan-Europea GEANT. Los países beneficiados de este proyecto son Argelia, Chipre, Egipto, Israel, Jordania, Líbano, Malta, Marruecos, Palestina, Siria, Túnez y Turquía.

La figura 131 muestra el backbone de EUMEDCONNECT. Para mayores detalles el mapa puede ser encontrado en el disco anexo a la tesis con el nombre EumedConnect o en la página de Geant2.

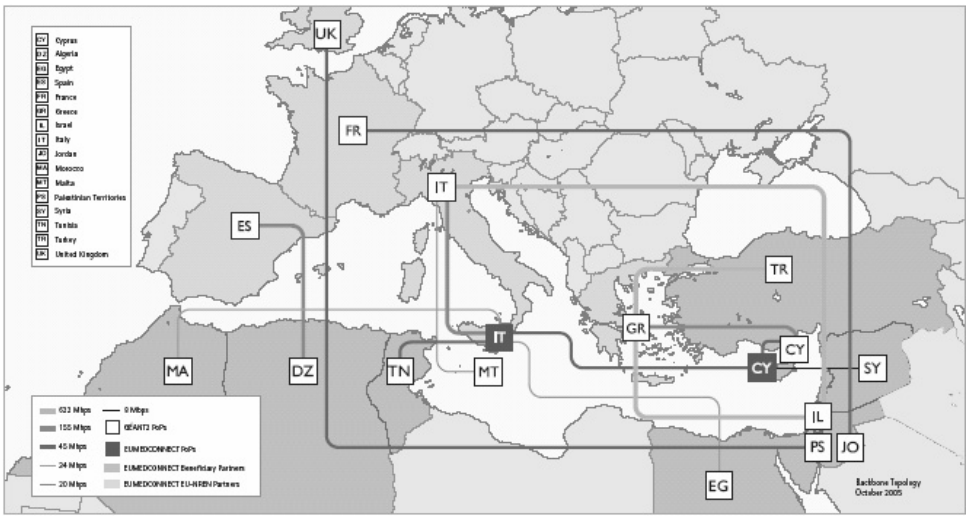


Figura 131. Red EUMEDCONNECT

5.7. TEIN2

TEIN2 (Trans-Eurasia Information Network) es la primera red de educación e investigación de gran escala de la región Asia-Pacìfico. Èsta conecta once países en la región (Australia, China, Hong Kong, Indonesia, Japón, Corea, Malasia, Las Filipinas, Singapur, Tailandia y Vietnam), y provee directa conectividad a la red Europea GEANT2.
La figura 132 muestra el backbone de TEIN2 y sus conexiones hacia GEANT2. Para mayores detalles el mapa puede ser encontrado en el disco anexo a la tesis con el nombre TEIN2Topology o en la página de Geant2.

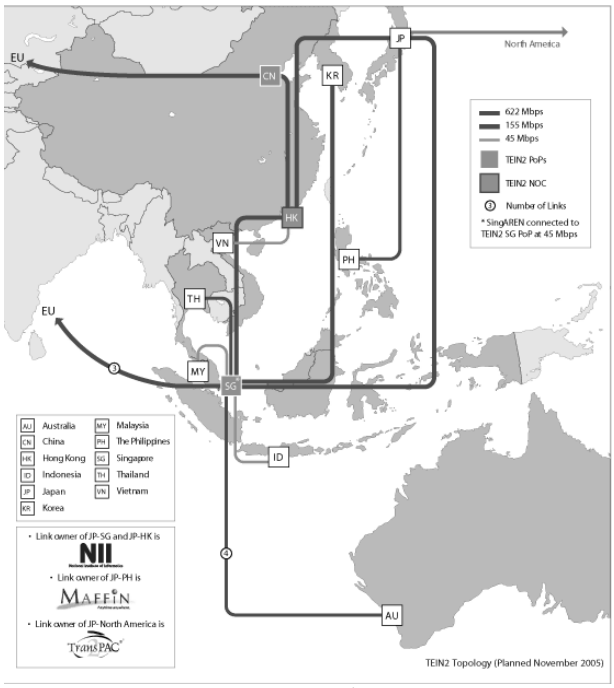


Figura 132. Red TEIN2

5.8. ALICE

El proyecto ALICE (América Latina Interconectada Con Europa**)** coordinado por DANTE, creò la primera red de investigación y educación para la zona de Latinoamérica. Conocida como RedCLARA, comenzó a funcionar el 1 de septiembre de 2004 proveyendo conectividad a 155 Mbps, en una topología de "anillo", enlazando a las redes de investigación y educación nacionales de Argentina, Brasil, Chile, Panamá y México, conectándolas con GÉANT2 a 622 Mbps mediante un enlace entre São Paulo, Brasil, y Madrid, España. A partir de esta conexión inicial, la red ha crecido de forma rápida, con la adhesión constante de socios regionales a la comunidad RedCLARA.

La RedCLARA ayuda a los investigadores latinoamericanos a competir en la carrera mundial de la investigación y permite una colaboración sin precedentes con sus colegas europeos. Anteriormente, la colaboración de investigación entre Latinoamérica y Europa se veía obstaculizada por la falta de una conexión exclusiva entre estas dos regiones. El proyecto ALICE ha abordado directamente este problema, ofreciendo con RedCLARA una conexión directa con la red europea GEANT2.

El 9 de junio de 2003, en la ciudad de Valle de Bravo, México, se realizó la firma de los estatutos para la formación de la Cooperación Latino Americana de Redes Avanzadas (CLARA), organización que es contraparte de DANTE en la región, así 18 países latinoamericanos se unen para integrar una red que potenciará el desarrollo de la Investigación y el Desarrollo (I+D) en la región. CLARA es una asociación civil, sin fines de lucro, con sede en el Uruguay.

La tabla 43 muestra las NREN regionales que se encuentran actualmente conectadas a la RedCLARA:

Tabla 43. NRENs conectadas a la RedCLARA

NREN	**Pais**	**NREN**	**Pais**
RETINA	Argentina	RAGIE	Guatemala
ADSIB	Bolivia	UNITEC	Honduras
RNP	Brasil	CUDI	Mexico
RENATA	Colombia	RENIA	Nicaragua
CR2Net	Costa Rica	RedCyT	Panama
RedUniv	Cuba	Arandu	Paraguay
REUNA	Chile	RAAP	Peru
CEDIA	Ecuador	RAU	Uruguay
RAICES	El Salvador	REACCIUN	Venezuela

La iniciativa CLARA tiene dos vertientes: la formación de una infraestructura que integre a las redes avanzadas latinoamericanas y la creación de una organización no gubernamental que represente los intereses de esta red de organizaciones**.**

Los objetivos de CLARA son:

- Coordinación entre las Redes Académicas Nacionales de América Latina y con otros bloques.

- Cooperación para la promoción del desarrollo científico y tecnológico.
- Planificación e implantación de servicios de redes para la interconexión regional.
- Desarrollo de una red regional (Red CLARA) para interconectar a las redes nacionales académicas y de investigación que serán operadas por sus Asociados.

Los miembros de CLARA coordinan sus esfuerzos para llevar las distintas aplicaciones y nuevas tecnologías a las NREN que la integran. De esta forma los ingenieros de las distintas NREN miembros de CLARA se integran para formar grupos de trabajo (GT) en las siguientes materias:

- Videoconferencia
- Voz sobre IP
- Seguridad
- Multicast
- IPv6
- Enrutamiento Avanzado
- Mediciones

Al igual que otras NREN, CLARA ha firmado los siguientes MoUs (Memorandum of Understanding) con las siguientes redes avanzadas:

- APAN (Consorcio Asia-Pacìfico de redes)
- Internet2 (UCAID)

5.8.1. Topología de RedCLARA

La troncal (backbone) de RedCLARA está compuesta por cinco nodos enrutadores principales, conectados en una topología de anillo. Cada nodo principal representa a un PoP para RedCLARA, y cada uno de ellos está ubicado en un país de América Latina.

Los cinco principales nodos IP de RedCLARA están ubicados en São Paulo (Brasil - BR), Buenos Aires (Argentina - AR), Santiago (Chile - CL), Panamá (PA) y Tijuana (México - MX). Todas las conexiones de las redes nacionales latinoamericanas (NREN) a RedCLARA serán a través de uno de estos cinco nodos. La troncal de CLARA está interconectada con la red paneuropea GÉANT2 a través del enlace del PoP de CLARA en São Paulo con el punto de acceso de GÉANT2 en Madrid (España - ES).

Cuando una NREN latinoamericana hace conexión con RedCLARA, lo hace a través de uno de los cinco nodos principales de la troncal de CLARA; esta conexión le brinda a estas NREN y sus miembros (clientes) acceso a RedCLARA, otorgándoles un Punto de Intercambio.

La figura 133 muestra el backbone de la RedCLARA y su conexión a GEANT2 por medio de Madrid. Para mayores detalles el mapa puede ser encontrado en el disco anexo a la tesis con el nombre RedCLARATopology o en la página de Geant2.

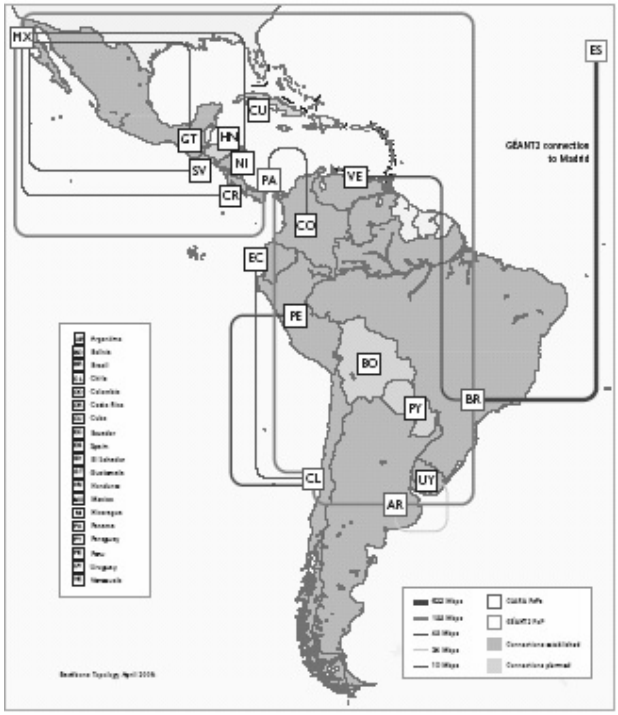


Figura 133. Topología de la RedCLARA

5.8.2. IPv6 en la RedCLARA

El Grupo de Trabajo de IPv6 (GTv6) se estableció para analizar la implementación, operación inicial y el uso de IPv6 en la RedCLARA y las redes nacionales de los Asociados, emitiendo recomendaciones, generando documentos y sirviendo de apoyo a las NRENs.
Dentro de los objetivos del GTv6 se encuentran:

- Apoyar el despliegue y la operación inicial de IPv6 en la RedCLARA.
- Ayudar al despliegue y la operación inicial de IPv6 en las redes de las NRENs.
- Investigar y utilizar aplicaciones con soporte IPv6 para CLARA.

De las 18 NRENs que integran a la RedCLARA sòlo 11 actualmente tienen soporte IPv6. La tabla 44 nombra a las once NRENs de RedCLARA con IPv6.

Tabla 44. NRENs conectadas a la RedCLARA con soporte IPv6

NREN	País	NREN	País
RETINA	Argentina	RENIA	Nicaragua
RNP	Brasil	RedCyT	Panamá
REUNA	Chile	RAAP	Perú
CEDIA	Ecuador	RAU	Uruguay
RAGIE	Guatemala	REACCIUN	Venezuela
CUDI	México		

La RedCLARA a partir del 9 de Agosto del 2005 habilitò nativamente IPv6 en los cinco nodos principales del backbone, para el 19 de Noviembre del mismo año, se habilitò Multicast IPv6 en el backbone. Para el manejo del protocolo de enrutamiento IGP se ha manejado desde el inicio de la red IS-IS para tener soporte en ambas versiones del protocolo (IPv4 e IPv6). Para el protocolo de enrutamiento EGP se manejan MBGP para tener el soporte para IPv6.

El plan de numeración IPv6 que se ha manejado tiene la siguiente estructura:

2001:1348::/32 (sTLA) Asignado a CLARA por LACNIC[26]
2001:1348:00XX/48 RedCLARA Core Backbone

5.9. CUDI

Como en otros países en México se toma la iniciativa de implementar y desarrollar redes de alto desempeño para el desarrollo de investigación que requiera un amplio ancho de banda y de esta manera poder estar a la vanguardia de las comunicaciones y experimentos en nuestro país.

Desde la década de los 90, las universidades mexicanas empezaron a tener proyectos de aplicaciones avanzadas con equipamientos de alta tecnología, que les permiten una mayor eficiencia en los procesos de educación y de investigación que llevan a cabo.

Algunas universidades mexicanas buscaron conectarse directamente a la red Internet2 de Estados Unidos. UCAID respondió que sería necesario hacer un consorcio de universidades mexicanas, ya que sería muy ineficiente conectar universidades de manera individual.

Es así como el 8 de abril de 1999 se constituye la Corporación Universitaria para el Desarrollo de Internet (CUDI), la cual es el organismo que maneja el proyecto de la red Internet2 en México y busca impulsar el desarrollo de aplicaciones que utilicen esta red, fomentando la colaboración en proyectos de investigación y educación entre sus miembros. CUDI es una asociación civil de carácter privado, sin fines de lucro, integrada por las universidades del país.

Esta organización nace con el impulso de 8 universidades del país, la cuales se comprometieron a pagar a prorrota los costos de la red que no se pudieran sufragar con otras fuentes, estas universidades son las siguientes:

- Centro de Investigación Científica y de Educación Superior de Ensenada (CICESE)
- Instituto Politécnico Nacional (IPN)
- Instituto Tecnológico y de Estudios Superiores de Monterrey (ITESM)
- Universidad Autónoma Metropolitana (UAM)
- Universidad Autónoma de Nuevo León (UANL)
- Universidad de Guadalajara (UDG)

[26] Latin American and Caribbean Internet Addresses Registry

- Universidad de las Américas Puebla (UDLAP)
- Universidad Nacional Autónoma de México (UNAM)

Apoyándose en este compromiso, Teléfonos de México (Telmex) y Avantel han aportado sin costo a la RedCUDI 8 000 kilómetros de red dorsal de alta capacidad. A cambio de la donación se acordó que la red sería una red privada que no comercializarà servicios de telecomunicaciones y que se utilizaría exclusivamente para aplicaciones educativas y de investigación

5.9.1. Administración CUDI

La administración de CUDI recae en su **Consejo Directivo**, que es el órgano de gobierno encargado por Asamblea de miembros del manejo de la Asociación Civil. Su presidencia rota anualmente entre los Asociados Académicos de la organización.

El Consejo, a su vez, se apoya en el trabajo de tres comités:

- El **Comité de Membresías**, tiene a su cargo evaluar las solicitudes de nuevas membresías, y desarrollar e implementar estrategias de promoción.
- El **Comité de Aplicaciones y Asignación de Fondos**, promueve el desarrollo de aplicaciones que utilicen la red, supervisa la correcta utilización de los fondos asignados y promueve el establecimiento de comunidades (Educación, Salud, Grids, Bibliotecas Digitales, Ciencias de la Tierra, Laboratorios, Biodiversidad, Astronomía)
- El **Comité de Desarrollo de la Red (CDR)** que aprueba el diseño de la red y supervisa su operación. Èste està conformado por grupos de trabajo (End to End, QoS, Enrutamiento I2, IPv6, Multicast, Tecnologías Audiovisuales, MPLS, Seguridad, Ingeniería y Desarrollo de la Red)

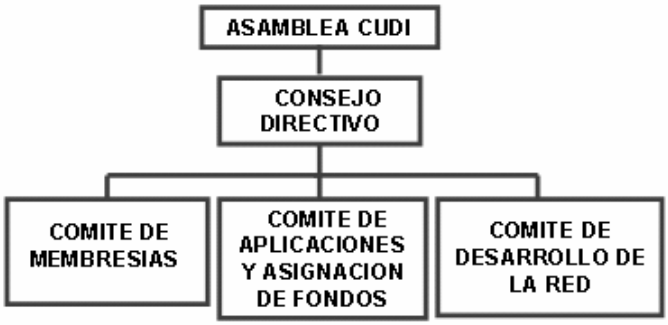


Figura 134. Estructura administrativa de CUDI

5.9.2. Membresía CUDI

Actualmente la membresía de CUDI se integra por las principales universidades y centros de investigación del país. Adicionalmente, forman parte de la membresía de CUDI, empresas que apoyan la investigación y educación en el país.

CUDI está formada actualmente por cuatro categorías de miembros:

- **Asociado Académico** (22) Universidades que adquieren el compromiso financiero de absorber a prorrata el costo de mantener la red operando. Forman parte del Consejo Directivo.
- **Afiliados Académicos** (49) Universidades que únicamente desean conectarse a la red y absorben los costos directos de su conexión a la red dorsal.
- **Asociados Institucionales** (4) Instituciones no universitarias que realizan una aportación mayor a la asociación y forman parte del Consejo Directivo.
- **Afiliados empresariales** (2) Instituciones no universitarias que realizan una aportación menor a la asociación.

La tabla 45 muestra a los miembros conectados con CUDI[27]

Tabla 45. Miembros de CUDI

ASOCIADOS ACADEMICOS (22)				
BUAP	CICESE	CCONACYT*	DGEST****	DGTVE
ILCE	IPN	ITESM	INS**	UNIPOL**
UAX	UACJ	UAL	UANL	UAT
UAEH	UAEM	UAM	UDG	UDLAP
UNAM	UV			
AFILIADOS ACADEMICOS (49)				
BID	CAPMED	CasaUniv.Calif.	CIMMYT	CINVESTAV
COLPOS	COLNAL	CONABIO	FMS	IIE
IMP	INEGI	ITAM	ITESI	ITSON
LANIA	TAMU	TESE	UAA	UABC
UABJO	UAChapingo	UACH	UADEC	UAEMEX
UASLP	UADY	UATX	UAG	UAGRO
UAN	UAQ	UCOL	UGTO	UJAD
UJAT	UIA	ULSA	UNACH	UMICH
UM	UPN	UPAEP	UQROO	UR
USON	UVM	UNESCO	UNITEC	
ASOCIADOS INSTITUCIONALES (4)				
Avantel	CISCO México	CONACyT	TELMEX	
AFILIADOS EMPRESARIALES (2)				
Centro NETEC	VITECH			
Centros Públicos-CONACYT (28)*				
CIAD	CIATEC	CIATEJ	CIATEQ	CIBNOR
CICY	CIDE	CIDESI	CIDETEQ	CIESAS
CENTRO GEO	CIMAT	CIMAV	CIO	CIQA
COLEF	COLMEX	COLMICH	COLSAN	COMIMSA
ECOSUR	FIDERH	FLACSO	IMORA	INAOE
INECOL	INFOTEC	IPICYT		
Institutos Nacionales de Salud (14)**				
INCan	INC	INCMyNSZ	INER	INNN
INP	INPER	INPRF	INSP	HIM
INR	CNTS	CENATRA	INMEGEN	
Subsistema de Universidades Politecnicas (18)***				
UPA	UPBC	UPCHI	UPDGO	UPFIM

[27] http://www.cudi.edu.mx/members/miembros_cudi.pdf

UPGPDGO	UPGTO	UPEMOR	UPQ	UPP
UPPUEBLA	UPSLP	UPSIN	UPTLX	UPTGO
UPVM	UPZAC	UPZMG		
Institutos Tecnologicos (61)****				
CENIDET	ITCd.Madero	ITCelaya	ITAgs.	ITChihuahua
ITConkal	ITDurango	ITLaguna	ITLeón	ITMérida
ITMinatitlán	ITMorelia	ITOaxaca	ITOrizaba	ITPuebla
ITQueretaro	ITSaltillo	ITTepic	ITTijuana	ITVeracruz
ITAcapulco	ITApizaco	ITBoca Rio	ITCampeche	ITCancun
ITCdCuauhte.	ITCdGuzman	ITCdJuarez	ITCdValles	ITCerro Azul
ITChetumal	ITChihuahua II	ITChilpancingo	ITCIIDET	ITColima
ITComitan	ITComitancillo	ITCuliacan	ITDelicias	ITHermosillo
ITIstmo	ITJiquilpan	ITCuenca Papaloapan	ITPaz	ITLazaro Cardenas
ITMochis	ITMatamoros	ITMexicalli	ITNogales	ITNuevo Laredo
ITPachuca	ITParral	ITSLP	ITTlajomulco	ITToluca
ITTuxtepec	ITValle Oax	ITVillahermosa	ITZacatecas	ITZacatepec
ITZitacuaro				

Al igual que otras NREN, CUDI ha firmado los siguientes MoUs (Memorandum of Understanding) con las siguientes redes avanzadas:

- CANARIE (Canadá)
- CLARA (Latinoamérica)
- CENIC (USA)
- RedIRIS (España)
- RETINA (Argentina)
- REUNA (Chile)
- UCAID (USA)

5.9.3. Topología de la RedCUDI

En la actualidad la RedCUDI cuenta con un backbone donado por TELMEX y AVANTEL de más de 8,000 kilómetros de enlaces de alta capacidad que operan a una velocidad de 155 megabits por segundo. Esta red dorsal abarca todo el territorio nacional. Se cuenta además con tres enlaces de la misma velocidad que permiten la interconexión con las principales redes académicas de Estados Unidos y del resto del mundo. A través de estos enlaces es posible tener acceso a más de 45 redes similares de Europa, Asia, Oceanía y América Latina que interconectan a más de 3,000 universidades y centros de investigación.

La figura 135 muestra el backbone de la RedCUDI y sus conexiones internacionales. Para mayores detalles el mapa puede ser encontrado en el disco anexo a la tesis con el nombre BackboneRedCUDI o en la página de CUDI.

Figura 135. Backbone de la RedCUDI

5.9.4. IPv6 en la RedCUDI

Desde sus inicios la red de Internet2 de México ha funcionado con IPv4 sin embargo, actualmente ya se tiene soporte del protocolo IPv6 en el Backbone, por lo que paulatinamente se ha empezado a utilizar IPv6 desde los equipos centrales hasta los equipos terminales de los integrantes de la red.

La conexión en RedCUDI con IPv6 se puede realizar actualmente mediante una conexión punto a punto entre un equipo de la RedCUDI y su equipo remoto, ambos con soporte IPv6. Esta conexión puede ser de dos tipos: Nativa o por Túnel (IPv6/IPv4).

A partir del 2006 se empezò a utilizar direcciones del bloque propio 2001:1228::/32 adjudicado por LACNIC, que sustituyó a los bloques:

- El de pruebas pNLA (asignado por la UNAM del proyecto 6BONE): 3FFE:8070:1006::/48
- El de producción sNLA (asignado por la UNAM de ARIN): 2001:0448:03::/40 (que substituyó al 2001:0448:0003::/48)

Cabe mencionar que el objetivo del trabajo se basa esencialmente en el desarrollo del direccionamiento del bloque propio adjudicado a RedCUDI por LACNIC el 15 de Noviembre de 2005, y demás documentos que conlleva la implementación y puesta en operación de dicho bloque de direcciones IPv6 en la RedCUDI.

La figura 136 muestra el Backbone de IPv6 que operaba en la RedCUDI hasta el 16 de Octubre de 2001. Para mayores detalles el mapa puede ser encontrado en el disco anexo a la tesis con el nombre RedCUDIIPv6-2001 o en la página del grupo de trabajo IPv6 de CUDI. El siguiente capìtulo detallarà el trabajo realizado en el nuevo bloque de direcciones IPv6, así como el estado actual del backbone IPv6 en la RedCUDI.

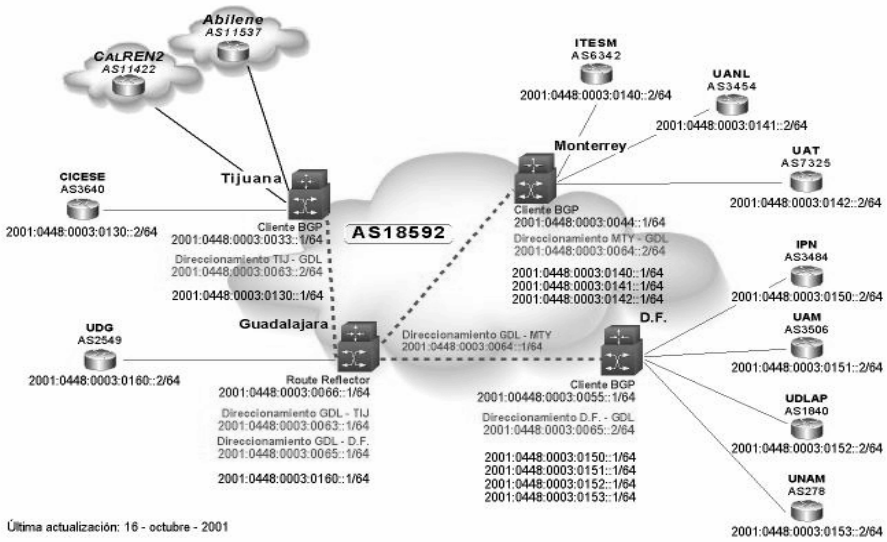


Figura 136. Backbone de la RedCUDI con soporte IPv6.

Capítulo

6.Actualización del soporte IPv6 en la RedCUDI

Resumen

El grupo de trabajo de IPv6 de la RedCUDI, con apoyo del proyecto IPv6 de la UNAM, realizaron trabajos de pruebas sobre la RedCUDI con bloques de direcciones IPv6 asignados por la UNAM de su prefijo de 6BONE, y posteriormente con direcciones de producción obtenidos de ARIN. La RedCUDI después de estar involucrada con el manejo de éste protocolo, por medio del grupo de trabajo IPv6 solicita un bloque de direcciones IPv6 a LACNIC, el cual asigna un bloque /32 de direcciones IPv6 el 15 de noviembre de 2005.

En este capìtulo se desarrolla la actualizacion del direccionamiento llevado a cabo en la RedCUDI a partir del bloque adquirido de LACNIC, para sustituir los actuales en uso, y demás documentos que son necesarios para el buen uso interno del prefijo IPv6, así como las recomendaciones utilizadas para la puesta en producción del nuevo bloque de direcciones IPv6.

6.1. ANTECEDENTES DE IPv6 EN LA RedCUDI

En diciembre de 1998 se inicia el proyecto IPv6 de la Universidad Nacional Autónoma de México (UNAM), para junio de 1999 la UNAM registra el primer nodo de 6BONE[28] (red de prueba IPv6 para asistir la evolución y desarrollo de IPv6) en México y para Septiembre de 1999 es aceptada como nodo de Backbone de 6BONE, obteniendo un bloque de direcciones IPv6 para prueba pTLA (Pseudo TLA) 3FFE:8070::/28, con este hecho la UNAM fue el primer nodo de 6BONE en México y el tercero en Latinoamérica.

En octubre de 2000 la UNAM adquiere un bloque de producción sTLA (sub Top-Level Aggregation) 2001:0448::/35 de ARIN, el cual es usado para asignar direcciones IPv6 a las instituciones en México, y Latinoamérica que estén interesadas en tener accesos a servicios basados en IPv6; el 29 de junio de 2005 adquiere otro bloque sTLA (2001:1218::/32) de LACNIC, el cual sustituirá a los dos anteriores.

Desde sus inicios la red de Internet2 de México ha funcionado con IPv4 sin embargo, la tendencia mundial es la migración a IPv6 desde el Backbone hasta los equipos terminales de los integrantes de esta red. Con esto se integra el grupo de trabajo IPv6 en Abril del 2000, para realizar trabajos de IPv6 en la red de Internet2 de México (RedCUDI), con la participación del CIMAT, IPN, ITESM, LANIA, UACH, UAEH, UAL, UDG y UNAM, quien fue el encargado de coordinarlos. Dentro de sus objetivos se encuentran:

- Instalar IPv6 en la red de Internet 2 de México (RedCUDI).
- Realizar pruebas de desempeño con IPv6.
- Utilizar y desarrollar aplicaciones con soporte IPv6.
- Realizar pruebas en colaboración con otros Grupos de Trabajo y Comités.

En Agosto del 2000, CUDI se registra en 6BONE en el nodo de backbone de la UNAM, de la cual adquiere un bloque IPv6 pNLA (3FFE:8070:1006::/48) para pruebas.

En abril de 2001, se establece un plan de trabajo que contempla en una primera etapa, la instalación y configuración de IPv6 en el equipo de acceso a Internet2 de la UNAM y la puesta en operación del primer túnel de IPv6 sobre IPv4 por Internet2 entre las redes de Internet2 de México (RedCUDI) y EUA (ABILENE) levantando una sesión con el protocolo BGP4+, después se configurò la primer conexión nativa de IPv6 entre CUDI y la UNAM.

En diciembre de 2001, se instala nativamente IPv6 en el backbone de Internet2 de CUDI. Para llevarlo acabo se migraron las versiones de IOS de todos los enrutadores Cisco 7200, del backbone a la versión c7200-js-mz.122-8.T.bin, con soporte IPv6. En los Switches ATM Cisco BPX 8620 no hubo necesidad de realizar algún cambio, ya que son de capa 2. Con esto se establecieron conexiones IPv6 nativas sobre ATM entre todos los equipos del backbone, que en ese momento contemplaba a los nodos de Ciudad de México, Guadalajara, Monterrey y Tijuana. Se configura BGP4+ como protocolo EGP y RIPng

[28] http://www.ipv6.unam.mx/

como protocolo IGP. Como siguiente etapa se continúo con la configuración de IPv6 en los equipos de acceso de las universidades conectadas a la red de CUDI[29].

El 20 de junio de 2002 se logra configurar con éxito la primera conexión IPv6 nativa entre las redes de Internet2 de México y EE.UU., CUDI y ABILENE respectivamente (Figura 137).

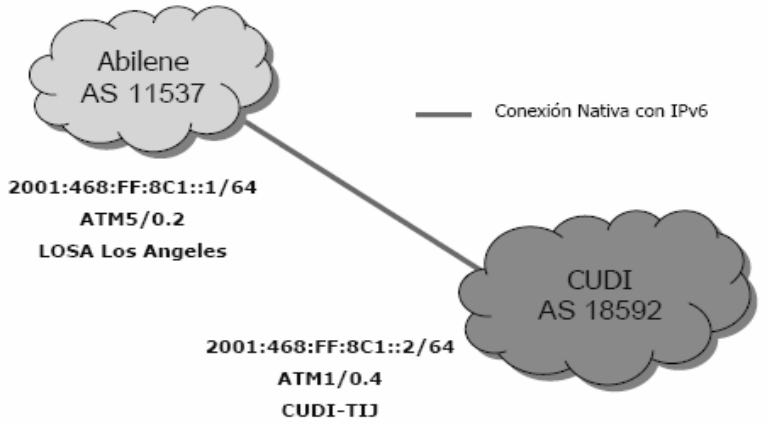


Figura 137. Conexión nativa IPv6 de CUDI-Abilene

En el 2003, del bloque /48 de direcciones IPv6 de producción asignado a CUDI por la UNAM, se incrementò el bloque de direcciones IPv6 a un /40.

- Primer prefijo sNLA 2001:0448:0003::/48 (producción)
- Prefijo actualizado sNLA 2001:0448:03::/40 (producción)

En la reunión de primavera de 2005 celebrada en Veracruz México, el grupo de trabajo IPv6 propone la solicitud de un bloque de direcciones IPv6 propio para producción de tipo sTLA (/32) para la red de Internet2 de México (CUDI) ante LACNIC, el cual se justificaba con los siguientes puntos:

- Los cambios de tamaño de los prefijos de producción a nivel mundial
- Las facilidades ofrecidas para obtener bloques propios
- Uso màs eficiente del espacio de direcciones adjudicado
- Facilitara mejor el crecimiento de Internet2.

lo que involucraba realizar en la RedCUDI:

- Realizar encuesta a los miembros de CUDI acerca del soporte IPv6 que se tenía a la fecha.
- Nuevo direccionamiento IPv6 para la interconexión de los equipos de backbone y de estos con los equipos de los Asociados Académicos de la RedCUDI.
- Plan de asignación de bloques /48 de direcciones IPv6 a los miembros de CUDI
- Desarrollo de políticas de Ruteo IPv6 que regirá a la RedCUDI.

[29] http://ccc.inaoep.mx/~cferegrino/cursos/redscomp/primera_cx_ipv6.pdf

- Desarrollo de políticas de Asignación que regirán a los miembros de CUDI.
- Procedimientos que se deberá llevar acabo para la asignación de bloques IPv6 a los miembros de CUDI que lo soliciten.
- Actualización para la asignación de nombres DNS de los equipos de Backbone y de èste con los Asociados Académicos.
- Desarrollo de un plan de renumeración en el CORE de la RedCUDI.
- Desarrollo y actualización de documentación en la que se vea involucrado con el uso del nuevo bloque de direcciones IPv6.

Para que LACNIC adjudicara un bloque de direcciones IPv6 sTLA (/32) para producción a CUDI, se tuvieron que cumplir con los criterios que se solicitan dentro de las políticas de adjudicación de un espacio de direcciones IPv6 de LACNIC[30]:

- Ser un ISP.
- No ser un sitio final (usuario final).
- Documentar un plan detallado sobre los servicios y la conectividad en IPv6 a ofrecer.
- Anunciar en el sistema de rutas inter-dominio de Internet un único bloque, que agregue toda la asignación de direcciones IPv6 recibida, en un plazo no mayor de 12 meses.
- Ofrecer servicios en IPv6 a clientes en Latinoamérica.
- Llenar la platilla de solicitud de bloque (http://lacnic.net/templates/ipv6-template-sp.txt).
- Hacer el pago correspondiente.
    - Para un bloque /32 son $2,500 dls anuales.
    - Sin pago hasta nuevo aviso (reemplazo al anterior).

En agosto de 2005, se establece la conexión por IPv6 entre las redes de CUDI y CLARA.

El 15 de noviembre de 2005, por conducto del grupo de trabajo IPv6 de CUDI, se adquiere finalmente de LACNIC un bloque de direcciones IPv6 (2001:1228::/32) para servicios de producción, el cual sustituyò paulatinamente a los bloques de pruebas y producción que tenìa asignado la UNAM a esta organización.

6.2. DIRECCIONAMIENTO IPv6 EN LA RedCUDI

Después de adquirir el bloque /32 de direcciones IPv6 de LACNIC, el grupo de trabajo IPv6, presenta en Enero de 2006, ante el CDR, una primera propuesta de direccionamiento IPv6 para la RedCUDI, quedando aceptado posteriormente en Marzo de 2006, dicho direccionamiento se presenta a continuación.

[30] http://www.lacnic.org/sp/politicas/ipv6.html

6.2.1. Estado del soporte IPv6 en los equipos del Backbone.

Como referencia se presenta un resumen del soporte IPv6 en los equipos del Backbone de la RedCUDI (Figura 135), de acuerdo al modelo y versión de IOS instalada.

- Backbone

El Backbone de la RedCUDI tiene hasta el momento seis PoPs (Puntos de Presencia), en los cuales existen los equipos listados en las tablas 46 y 47; el estado del soporte IPv6 en los mismos es el siguiente:

1. Los ruteadores Cisco 7200 que están en las instalaciones de Telmex y Avantel soportan IPv6 después de instalarles una versión del IOS estable y reciente.
2. A los Switches ATM de Cisco modelo BPX 8620, al ser de capa 2, no ha sido necesario cambiarles nada.
3. Para los ruteadores Cisco 10000 que están en las instalaciones de Avantel no existe actualmente una versión del IOS reciente y estable con el soporte de IPv6.

Equipos en Telmex

Tabla 46. Versiones del IOS en los equipos de Backbone en Telmex

Nodo de Backbone	Equipo	Versión IOS	Imagen
Guadalajara	Cisco 7204 NPE-200	12.2(13)T9	c7200-jk9s-mz.122-13.T9.bin
Monterrey	Cisco 7204 NPE-200	12.2(13)T1	c7200-p-mz.122-13.T1.bin
Tijuana	Cisco 7206VXR NPE-G1	12.4(2)T2	c7200-adventerprisek9-mz.124-2.T2.bin
México D.F.	Cisco 7204 NPE-200	12.2(13)T1	c7200-p-mz.122-13.T1.bin
Cd. Juárez	Cisco 7206VXR NPE-G1	12.4(2)T2	c7200-adventerprisek9-mz.124-2.T2.bin

Equipos en Avantel

Tabla 47. Versiones del IOS en los equipos de Backbone en Avantel

Nodo de Backbone	Equipo	Versión IOS	Imagen
México	Cisco 10008 PRE-1	12.0(21)SX	c10k-p10-mz.120-21.SX.bin
Monterrey	Cisco 10008 PRE-1	12.0(21)SX	c10k-p10-mz.120-21.SX.bin
Cancún	Cisco 7206VXR NPE-400	12.2(25)S5	c7200-ik91s-mz.122-25.S5.bin

- Nodos de acceso

En los nodos de los Asociados Académicos ha sido necesario:

1. Cambiar el IOS de los ruteadores Cisco por una versión que soporta IPv6.
2. En general cambiar el sistema operativo de los equipos de otras marcas por versiones que soportan IPv6.

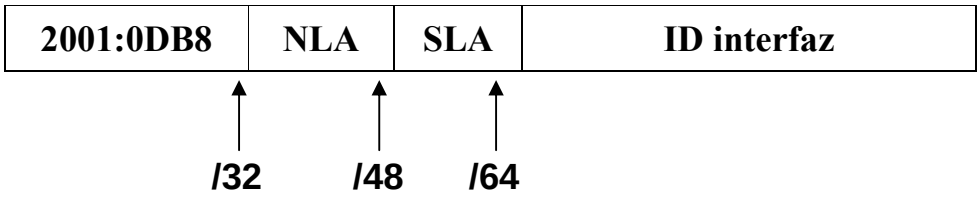
6.2.2. Prefijo IPv6 para la RedCUDI

El prefijo adjudicado a CUDI es el 2001:1228::/32, pero por razones de seguridad y privacidad de CUDI, el direccionamiento presentado en la tesis, se maneja con direcciones IPv6 de documentación especificadas en el RFC3849. Diversos documentos sobre el direccionamiento fueron elaborados por el grupo de trabajo IPv6, unos de carácter interno y otros externos, como los RFCMX, los cuales pueden ser encontrados en diversos formatos en la página http://rfc.cudi.edu.mx/.

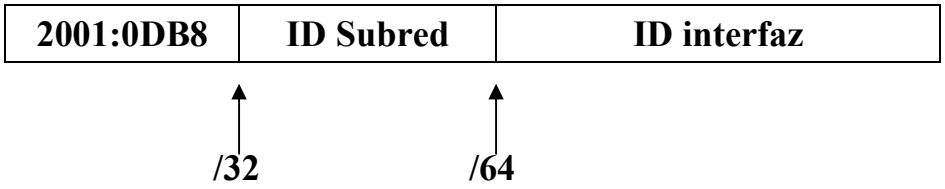
Del prefijo adjudicado de LACNIC:

sTLA 2001:0DB8:: /32[31]

El direccionamiento se diseñò tomando como referencia el formato IPv6 unicast globales dado por el RFC2374 como sigue:



Actualmente el formato para direcciones IPv6 unicast globales es actualizado por el RFC3587 como se muestra:



6.2.3. Asignación de bloques de direcciones.

A partir del bloque delegado a la red de Internet2 de México (RedCUDI) se asigna los siguientes bloques de direcciones para el Backbone y Asociados Académicos como se muestra en la tabla 48, de la cual se pueden obtener:

- 16 /40 para el Backbone
- 65504 /48 para los Asociados Académicos
- 16 /40 reservado

[31] G. Huston, A. Lord, P. Smith (2004). RFC3849 “IPv6 Address Prefix Reserved for Documentation”. Informational. Pp 1.

Tabla 48. Asignación de bloques de direcciones.

Rango		Asignación
Inicio	Final	
2001:0DB8:$X_0$0::/40	2001:0DB8:X_0F::/40	Backbone
2001:0DB8:$X_1$000::/48	2001:0DB8:EFFF::/48	Asociados Académicos
2001:0DB8:X_{15}0::/40	2001:0DB8:X_{15}F::/40	Reservado

6.2.4. Direccionamiento para el Backbone

De la tabla 48 se tiene que el bloque de direcciones que corresponde al direccionamiento del Backbone es del rango 2001:0DB8:$X_0$0::/40 al 2001:0DB8:X_0F::/40, donde el primer dígito (hay que tener presente que nos referimos como un dígito a un valor de 4 bits en su representación hexadecimal), del tercer campo de izquierda a derecha, tomando como delimitador ":", de la dirección IPv6, indica que son direcciones de backbone, y el dígito siguiente denota a un proveedor o carrier. Como se mencionó en el capitulo 5, el backbone de la red de Internet2 de México (RedCUDI) fue donado por TELMEX y AVANTEL para brindar la interconexión de los equipos del CORE, de aquí que el bloque asignado para las conexiones del backbone se divida tomando en cuenta a los proveedores; de donde sólo se utilizan 2 bloques /40 para TELMEX y AVANTEL, quedando 13 bloques /40 libres para futuras asignación y 1 bloque /40 reservado, como se muestra en la tabla 49.

Tabla 49. Direccionamiento para el Backbone

2001:0DB8:B_0P_0::/40	Reservado
2001:0DB8:B_0P_1::/40	Backbone con TELMEX
2001:0DB8:B_0P_2::/40	Backbone con AVANTEL
2001:0DB8:B_0P_3::/40	
2001:0DB8:B_0P_4::/40	
2001:0DB8:B_0P_5::/40	
2001:0DB8:B_0P_6::/40	
2001:0DB8:B_0P_7::/40	
2001:0DB8:B_0P_8::/40	
2001:0DB8:B_0P_9::/40	
2001:0DB8:B_0P_A::/40	
2001:0DB8:B_0P_B::/40	
2001:0DB8:B_0P_C::/40	
2001:0DB8:B_0P_D::/40	
2001:0DB8:B_0P_E::/40	
2001:0DB8:B_0P_F::/40	

6.2.4.1. Direccionamiento ::/64 para las conexiones del Backbone-Backbone y Backbone-Asociados Académicos.

Con lo anterior quedan identificados los bloques /40 asignados a cada proveedor, y se prosigue con la asignación de los ocho bits restantes aún no utilizados en este campo de la dirección IPv6. Partiendo de los bits inmediatos después del dígito identificador de proveedor, se continúa con la asignación del tercer dígito encontrado de izquierda a derecha del tercer campo de la dirección IPv6, el cual fue asignado para representar el tipo de conexión que se tiene entre los equipos; es decir, la conexión entre los equipos puede ser

entre equipos de backbone-backbone, backbone-asociados y aunque no pueda ocurrir pero se contempla en el direccionamiento la conexión de equipos de backbone-afiliados; para el cuarto digito se identifica el tipo de enlace utilizado, los cuales pueden ser de tipo Nativo, Túnel, Dirección de Loopback, VPNs y para realizar Pruebas, con lo que se obtienen direcciones /48. Los 16 bits restantes del cuarto campo de izquierda a derecha de la dirección IPv6 son utilizados para identificar la localización de los nodos que participan en la conexión. El formato general para asignación de direcciones IPv6, para la interconexión de los equipos de backbone, se muestra a continuación.

$$2001{:}1228{:}BPT_CT_E{:}OODD{::}/64$$

32	36	40	44	48	56	64
/--B--/	--P--/	--T_C--/	--T_E--/	-OO-/	-DD-/	
4	4	4	4	8	8	

Donde:

B: Backbone
- B_0 = Backbone
- B_1 = Reservado

P: Proveedor
- P_0 = Reservado
- P_1 = TELMEX
- P_2 = AVANTEL

TC: Tipo de conexión
- TC_0 = Backbone – Backbone
- TC_1 = Backbone – Asociado
- TC_2 = Backbone – Afiliado

TE: Tipo de enlace
- A = Nativo
- B = Túnel
- C = Loopback
- D = VPN
- E = Prueba

OO: Origen DD: Destino

El cuarto campo de izquierda a derecha de la dirección IPv6 se divide en dos partes de 8 bits para identificar la ubicación geográfica de los nodos origen y destino en donde se establece la conexión. Para la ubicación geográfica de los nodos del backbone, se realizo mediante la asignación de un código único a cada uno de los estados de la Republica Mexicana, de este modo los dígitos obtenidos son los valores que toman “OO” para el origen y “DD” para el destino, en la tabla 50 se muestra la asignación detallada de los códigos para cada estado de la República Mexicana.

Tabla 50. Asignación de códigos a los estados de la República Mexicana.

Código	Estado	Código	Estado
00	Reservado	11	Morelos
01	Aguascalientes	12	Nayarit
02	Baja California Norte	13	Nuevo León
03	Baja California Sur	14	Oaxaca
04	Campeche	15	Puebla
05	Coahuila	16	Querétaro
06	Colima	17	Quintana Roo
07	Chiapas	18	San Luis Potosí
08	Chihuahua	19	Sinaloa
09	D.F.	1A	Sonora
0A	Durango	1B	Tabasco
0B	Estado de México	1C	Tamaulipas
0C	Guanajuato	1D	Tlaxcala
0D	Guerrero	1E	Veracruz
0E	Hidalgo	1F	Yucatán
0F	Jalisco	20	Zacatecas
10	Michoacán	21	Internacional

De acuerdo a la tabla 49, el código que le corresponde a cada nodo actual conectado por su ubicación geográfica en la red de Internet2 (RedCUDI), son los siguientes.

OO o **DD** = 02 -> Tijuana
OO o **DD** = 13 -> Monterrey
OO o **DD** = 09 -> México D. F.
OO o **DD** = 0F -> Guadalajara
OO o **DD** = 08 -> Cd. Juárez
OO o **DD** = 17 -> Cancún
OO o **DD** = 21 -> Internacional

Para el direccionamiento de los equipos de backbone-backbone, se debe conocer por medio de qué proveedor se realiza la interconexión entre los nodos del CORE, que se encuentran ubicados geográficamente en Tijuana, Ciudad Juárez, Monterrey, Guadalajara, México D.F., y Cancún.

La tabla 51 muestra los enlaces establecidos entre cada nodo del backbone y por medio de que proveedor lo realizan. Para interpretar la tabla, primero se posiciona en la fila del nodo que se requieren conocer sus enlaces establecidos; después mediante la intersección de las columnas con la fila, se encuentran las conexiones establecidas con el nodo y por medio de qué proveedor la establece.

Tabla 51. Conexión entre los nodos del backbone de la RedCUDI

Conexión/ Nodo	**Tijuana**	**Cd. Juárez**	**Monterrey**	**Guadalajara**	**México D.F.**	**Cancún**
Tijuana				Telmex		
Cd. Juárez			Telmex			
Monterrey		Telmex		Telmex	Avantel	
Guadalajara	Telmex		Telmex		Telmex	
México D.F.			Avantel	Telmex		Avantel

Cancún					Avantel	

En la tabla 52 se muestra una lista de los Asociados Académicos directamente conectados a los nodos de backbone de la red de Internet2 (RedCUDI) mediante enlaces E3, agrupados por el proveedor que les brinda conectividad y por su ubicación geográfica.

Tabla 52. Nodo de Asociados Académicos

TELMEX	**AVANTEL**
TIJUANA	
Centro de Investigación Científica y de Educación Superior de Ensenada (CICESE)	
CIUDAD JUAREZ	
Universidad Autónoma de Ciudad Juárez (UACJ)	
MONTERREY	
Instituto Tecnológico y de Estudios Superiores de Monterrey (ITESM-MTY)	Universidad Autónoma de la Laguna (UAL)
Universidad Autónoma de Nuevo León (UANL)	Universidad Autónoma de Tamaulipas (UAT)
GUADALAJARA	
Universidad de Guadalajara (UDG)	
MEXICO D.F.	
InTELMEX*	Avantel VPNs*
Instituto Politécnico Nacional (IPN)	Benemérita Universidad Autónoma de Puebla (BUAP)
Instituto Tecnológico y de Estudios Superiores de Monterrey (ITESM-CEM)	Consejo Nacional de Ciencia y Tecnología (CONACYT)
Secretaria de Educación Pública (SEP)	Instituto Latinoamericano de Comunicación Educativa (ILCE)
Universidad Autónoma Metropolitana (UAM)	Universidad Autónoma del Estado de Hidalgo (UAEH)
Universidad de las Américas Puebla (UDLAP)	Universidad Autónoma del Estado de Morelos (UAEMor)
Universidad Nacional Autónoma de México (UNAM)	Universidad Veracruzana (UV)
UNINET VPNs*	
CANCUN	

* Se asigna numeración como Asociado Académicos, porque se encuentran conectados con sus equipos directamente al nodo D.F. (TELMEX y AVANTEL) por un enlace E3.

Con toda la información anterior obtenida, la tabla 53 muestra cómo queda finalmente el direccionamiento de las conexiones entre los equipos de Backbone-Backbone y de Backbone-Asociado-Académico de la red de Internet2 (RedCUDI); cabe mencionar que el tipo de enlace que se tiene entre estos equipos es de tipo nativo.

Tabla 53. Conexiones backbone-backbone y backbone-Asociados Académicos

MEXICO	
	Conexión Backbone-Backbone
2001:0DB8:$B_0P_2TC_0$A:0913::/64	MEX-MTY
2001:0DB8:$B_0P_1TC_0$A:0F09::/64	GDL - MEX
2001:0DB8:$B_0P_2TC_0$A:0917::/64	MEX-CAN
	Conexión Backbone-Asociado
2001:0DB8:$B_0P_2TC_1$A:0901::/64	Avantel VPNs*
2001:0DB8:$B_0P_2TC_1$A:0902::/64	BUAP
2001:0DB8:$B_0P_2TC_1$A:0903::/64	CONACYT
2001:0DB8:$B_0P_2TC_1$A:0904::/64	ILCE
2001:0DB8:$B_0P_1TC_1$A:0905::/64	InTELMEX*
2001:0DB8:$B_0P_1TC_1$A:0906::/64	IPN
2001:0DB8:$B_0P_1TC_1$A:0907::/64	ITESM-CEM*
2001:0DB8:$B_0P_1TC_1$A:0908::/64	SEP*
2001:0DB8:$B_0P_2TC_1$A:0909::/64	UAEH
2001:0DB8:$B_0P_2TC_1$A:090A::/64	UAEMor
2001:0DB8:$B_0P_1TC_1$A:090B::/64	UAM
2001:0DB8:$B_0P_1TC_1$A:090C::/64	UDLAP
2001:0DB8:$B_0P_1TC_1$A:090D::/64	UNAM
2001:0DB8:$B_0P_1TC_1$A:090E::/64	UNINET VPNs*
2001:0DB8:$B_0P_2TC_1$A:090F::/64	UV
MONTERREY	
	Conexión Backbone-Backbone
2001:0DB8:$B_0P_1TC_0$A:0813::/64	CD.J - MTY
2001:0DB8:$B_0P_1TC_0$A:0F13::/64	GDL- MTY
2001:0DB8:$B_0P_2TC_0$A:0913::/64	MEX - MTY
	Conexión Backbone-Asociado
2001:0DB8:$B_0P_1TC_1$A:1301::/64	ITESM-MTY
2001:0DB8:$B_0P_2TC_1$A:1302::/64	UAL
2001:0DB8:$B_0P_1TC_1$A:1303::/64	UANL
2001:0DB8:$B_0P_2TC_1$A:1304::/64	UAT
CD. JUAREZ	
	Conexión Backbone-Backbone
2001:0DB8:$B_0P_1TC_0$A:0813::/64	CD.J-MTY
	Conexión Backbone-Asociado
2001:0DB8:$B_0P_1TC_1$A:0801::/64	UACJ
TIJUANA	
	Conexión Backbone-Backbone
2001:0DB8:$B_0P_1TC_0$A:0F02::/64	GDL-TIJ
	Conexión Backbone-Asociado
2001:0DB8:$B_0P_1TC_1$A:0201::/64	CICESE

GUADALAJARA	
	Conexión Backbone-Backbone
2001:0DB8:$B_0P_1TC_0$A:0F02::/64	GDL-TIJ
2001:0DB8:$B_0P_1TC_0$A:0F13::/64	GDL-MTY
2001:0DB8:$B_0P_1TC_0$A:0F09::/64	GDL-MEX
	Conexión Backbone-Asociado
2001:0DB8:$B_0P_1TC_1$A:0F01::/64	UDG
CANCÚN	
	Conexión Backbone-Backbone
2001:0DB8:$B_0P_2TC_0$A:0917::/64	MEX-CAN
	Conexión Backbone-Asociado
	?

* Se asigna numeración como Asociado Académicos, porque se encuentran conectados con sus equipos directamente al nodo D.F. (TELMEX y AVANTEL) por un enlace E3.

6.2.5. Direccionamiento para los Asociados Académicos

De la tabla 48 se tiene que el bloque de direcciones para las asignaciones a los Asociados Académicos corresponden del rango 2001:0DB8:$X_1$0::/40 al 2001:0DB8:EF::/40, de donde se pueden asignar 65504 bloques /48. El primer dígito con valor a "X_1" del tercer campo de la dirección IPv6 de izquierda a derecha, representa a una dirección de asignación, la cual podrá ir aumentando de valor según se vaya requiriendo, hasta llegar al valor de "E" que será el límite de asignaciones que se podrá hacer a los Asociados Académicos, nótese que el valor de F queda reservado.

En los bloques de asignación se identifica al POP por medio de los dos dígitos seguidos de "X_1", estos dígitos toman el valor que le corresponde por la ubicación geográfica del POP (ver tabla 50), que brinda la conexión al Asociado Académico. El cuarto dígito se utilizará para numerar a cada Asociado Académico que tenga conexión por dicho POP y que requiera bloque de direcciones IPv6, estos valores van de 1 a F, en donde el 0 se reserva.

6.2.5.1.Estructura interna de los miembros de la RedCUDI

La RedCUDI esta formada por miembros que integran las principales universidades y centros de investigación del país. Adicionalmente, forman parte de la membresía de RedCUDI, empresas que apoyan la investigación y educación en el país. Actualmente se cuenta con cuatro categorías de miembros:

- Asociado Académico: Universidades que adquieren el compromiso financiero de absorber a prorrata el costo de mantener la red operando. Forman parte del Consejo Directivo.
- Afiliados Académicos: Universidades que únicamente desean conectarse a la red y absorben los costos directos de su conexión a la red dorsal.
- Asociados Institucionales: Instituciones no universitarias que realizan una aportación mayor a la asociación y forman parte del Consejo Directivo.
- Afiliados empresariales: Instituciones no universitarias que realizan una aportación menor a la asociación.

Los Asociados Académicos son los miembros que funcionan como nodo de acceso al Backbone de la RedCUDI mediante enlaces E3, y a través de ellos, los Afiliados Académicos se conectan a la RedCUDI con enlaces E1, esto se muestra en la figura 138.

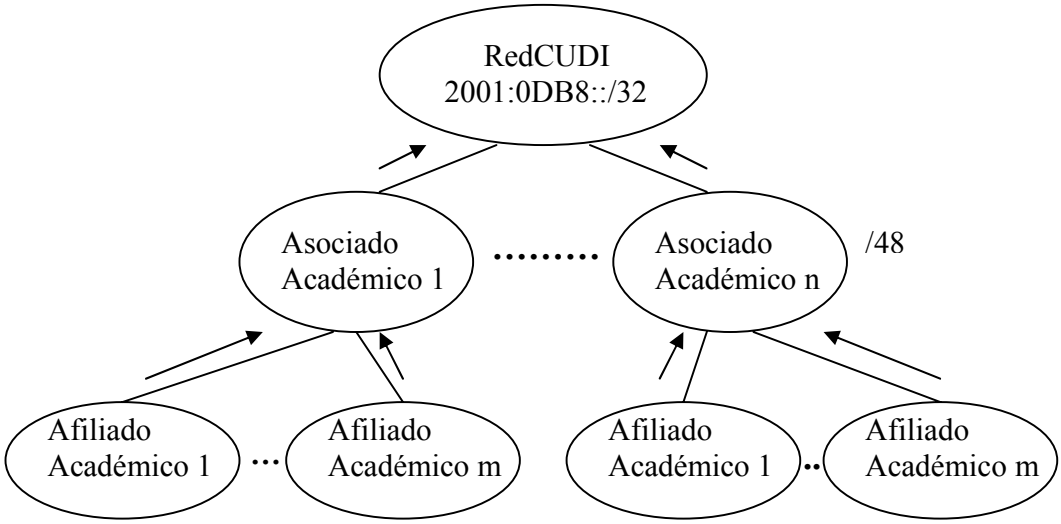


Figura 138. Estructura de conexión de los miembros de RedCUDI

6.2.5.2.Direccionamiento de bloques ::/48 para la Asignación a los Asociados Académicos.

El formato general para la asignación de bloques /48 de direcciones IPv6 a los Asociados Académicos, tomando en cuenta la ubicación del POP por donde se conecta el Asociado Académico al Backbone de la RedCUDI, del bloque 2001:0DB8:$X_1$000::/48 al 2001:0DB8:EFFF::/48, se muestra continuación.

2001:1228:XEEY::/48

Donde:

X: Identifica a una dirección de asignación
X = X_1 Primeras asignaciones
……
X = E Décimo cuartas asignaciones

Y: Identifica al número del Asociado Académico
Y = 0 Reservado
…….
Y = F

EE: identifica el estado en donde esta ubicado el POP origen

02 = Tijuana	09 = México DF	08 = Cd. Juárez	21 = Internacional
13 = Monterrey	0F = Guadalajara	17 = Cancún	

La tabla 54 muestra los bloques de direcciones /48 asignadas a cada uno de los Asociados Académicos por medio del POP que les brinda conectividad al backbone de la RedCUDI, basándose en el formato general para la asignación de bloques /48.

Tabla 54. Asignaciones de bloques /48 a los Asociados Académicos

2001:0DB8:$X_1$000::/48 - 2001:0DB8:$X_1$00F::/48	RESERVADO
	TIJUANA
2001:0DB8:$X_1$020::/48	Reservado
2001:0DB8:$X_1$021::/48	CICESE
	MONTERREY
2001:0DB8:$X_1$130::/48	Reservado
2001:0DB8:$X_1$131::/48	ITESM
2001:0DB8:$X_1$132::/48	UAL
2001:0DB8:$X_1$133::/48	UANL
2001:0DB8:$X_1$134::/48	UAT
	MÉXICO D.F
2001:0DB8:$X_1$090::/48	Reservado
2001:0DB8:$X_1$091::/48	BUAP
2001:0DB8:$X_1$092::/48	CONACYT
2001:0DB8:$X_1$093::/48	DGTVE
2001:0DB8:$X_1$094::/48	ILCE
2001:0DB8:$X_1$095::/48	IPN
2001:0DB8:$X_1$096::/48	SUBSIS. UNIVER. POLITECNICAS
2001:0DB8:$X_1$097::/48	UAEH
2001:0DB8:$X_1$098::/48	UAEMor
2001:0DB8:$X_1$099::/48	UAM
2001:0DB8:$X_1$09A::/48	UDLAP
2001:0DB8:$X_1$09B::/48	UNAM
2001:0DB8:$X_1$09C::/48	UV
	GUADALAJARA
2001:0DB8:$X_1$0F0::/48	Reservado
2001:0DB8:$X_1$0F1::/48	UDG
	CD. JUAREZ
2001:0DB8:$X_1$080::/48	Reservado
2001:0DB8:$X_1$081::/48	UACJ
	CANCÚN
2001:0DB8:$X_1$170::/48	Reservado

6.2.5.3.Ejemplo de Asignación de Bloques de direcciones IPv6 del Asociado Académico UNAM a sus Afiliados directamente conectados.

Cada Asociado Académico a partir del bloque /48 asignado por CUDI, podrá hacer su propio direccionamiento para los Afiliados Académicos que se conectan por medio de él a la RedCUDI. En esta sección se mostrará un ejemplo puramente explicativo y de carácter didáctico, de cómo el Asociado Académico UNAM hará el direccionamiento a sus Afiliados Académicos mediante bloques /56, a partir de su bloque /48 asignado por CUDI mostrado en la tabla 54.

Del bloque 2001:0DB8:$X_1$09B::/48 que pertenece a la UNAM, se puede realizar 256 asignaciones de bloques /56 a los Afiliados Académicos que se conectan por medio de él a la RedCUDI, con el siguiente formato general.

2001:0DB8:$X_1$09B:AA::/56

Donde

AA: número de Afiliado
AA = 00 Reservado
…….
AA = FF

La tabla 55 muestra los bloques de direcciones /56 asignados a cada uno de los Afiliados Académicos que se conectan a la RedCUDI por medio del Asociado Académico UNAM, tomando de referencia el formato anterior.

Tabla 55. Asignación de bloques /56 a los Afiliados Académicos de la UNAM.

2001:0DB8:$X_1$09B:00::/56	Reservado
2001:0DB8:$X_1$09B:01::/56	BID
2001:0DB8:$X_1$09B:02::/56	CONABIO
2001:0DB8:$X_1$09B:03::/56	CINVESTAV
2001:0DB8:$X_1$09B:04::/56	ColNal
2001:0DB8:$X_1$09B:05::/56	CUDI
2001:0DB8:$X_1$09B:06::/56	IMP
2001:0DB8:$X_1$09B:07::/56	ITAM
2001:0DB8:$X_1$09B:08::/56	Segmento de I2 UNAM
2001:0DB8:$X_1$09B:09::/56	TAMU
2001:0DB8:$X_1$09B:0A::/56	ULSA
2001:0DB8:$X_1$09B:0B::/56	UP

6.2.6. Direccionamiento para VPNs

Una de las facilidades que se permite como alternativa de conexión para enlazar un Afiliado Académico a la RedCUDI es por medio de VPNs, al igual que en los otros tipos de conexión, también se necesita un direccionamiento IPv6 para VPNs. El formato general para la asignación de direcciones IPv6 por medio de VPNs es el siguiente.

2001:1228:BP$T_C$$T_E$:R#VPN::/64

Donde

B: identifica a una dirección de Backbone
B_0 = Backbone

P: identifica el proveedor
P_1 = Telmex
P_2 = Avantel

TC: identifica entre qué tipos de equipos se realiza la conexión
TC_0 = Backbone-Backbone
TC_1 = Backbone-Asociado
TC_2 = Backbone-Afiliado

TE: identifica el tipo de enlace a utilizar
A = Nativo
B = Túnel
C = Loopback
D = VPN
E = Pruebas

R: identifica un valor reservado para uso futuro
R = 0 Reservado

#VPN: identifica el número asignado de VPN
#VPN = 000 Reservado
#VPN = 001 primer VPN
#VPN = 002 segunda VPN
…..
#VPN = FFF

6.2.7. Direccionamiento Loopback y Pruebas

Las direcciones de loopback forman parte de la configuración de los equipos del Backbone, y son utilizadas para levantar sesiones del protocolo de enrutamiento BGP, por tal razón se diseñó un direccionamiento IPv6 para las direcciones de loopback que usaran los equipos de backbone de la RedCUDI. Por otro lado se debe contar con un direccionamiento IPv6 para la realización de pruebas en la RedCUDI, de tal forma que se evite el empleo de los bloques de direcciones IPv6 de producción.

El formato de las direcciones de loopback empleado en los equipos del backbone de la RedCUDI, utiliza el mismo esquema del direccionamiento de las conexiones de Backbone, en donde las variables llamadas B, P, T_C, y T_E, del tercer campo de izquierda a derecha de la dirección IPv6, siguen manteniendo el valor que les corresponde. En el cuarto campo de la dirección IPv6, el primer dígito de 8 bits representado por PX, indica el código del estado, donde se encuentra el POP del equipo que contendrá la dirección de loopback, como se indicó en la tabla 50, los restantes 8 bits del cuarto campo, son reservados y puestos todos a uno, es decir (FF), lo cual identifica a una dirección IPv6 de loopback. Para la última variable identificada por #I, identifica el número de interfaz de loopback dentro del enrutador. Para las direcciones de prueba, al igual que las direcciones de loopback, contienen las mismas variables, lo cual implica un mismo esquema de direccionamiento IPv6, pero con la excepción que las direcciones de prueba son /56 y no manejan la variable #I.

2001:1228:BPT$_C$T$_E$:PXFF::#I/128 Loopback
2001:1228:BPT$_C$T$_E$:PX::/56 Pruebas

Donde

B: identifica a una dirección de Backbone
B_0 = Backbone

P: identifica el proveedor
P_1 = TELMEX
P_2 = AVANTEL

TC: identifica entre qué tipos de equipos se realiza la conexión
TC_0 = Backbone

TE: identifica el tipo de enlace a utilizar
A = Nativo
B = Túnel
C = Loopback
D = VPN
E = Prueba

PX: identifica el código del Estado en donde se encuentra el POP
02 = Tijuana
13 = Monterrey
09 = México D.F.
0F = Guadalajara
08 = Cd. Juárez
17 = Cancún
21 = Internacional

#I: identifica el número de interfaz de Loopback

#I = 0001 -> Loopback 1
#I = 0002 -> Loopback 2
#I = 0003 -> Loopback 3

En la tabla 56 se muestran las direcciones de loopback bajo el criterio antes descrito, agrupadas por medio del POP y del proveedor.

Tabla 56. Asignación de direcciones de Loopback.

	TELMEX	**AVANTEL**	**RANGO**
Tijuana	2001:0DB8:$B_0P_1TC_0$C:02FF::/64		
	2001:0DB8:$B_0P_1TC_0$C:02FF::/128		No Asignada
	2001:0DB8:$B_0P_1TC_0$C:02FF::1/128		Loopback 1
	2001:0DB8:$B_0P_1TC_0$C:02FF::2/128		Loopback 2
	2001:0DB8:$B_0P_1TC_0$C:02FF::3/128		Loopback 3
Guadalajara	2001:0DB8:$B_0P_1TC_0$C:0FFF::/64		
	2001:0DB8:$B_0P_1TC_0$C:0FFF::/128		No Asignada
	2001:0DB8:$B_0P_1TC_0$C:0FFF::1/128		Loopback 1

	2001:0DB8:$B_0P_1TC_0C$:0FFF::2/128		Loopback 2
	2001:0DB8:$B_0P_1TC_0C$:0FFF::3/128		Loopback 3
Monterrey	2001:0DB8:$B_0P_1TC_0C$:13FF::/64	2001:0DB8:$B_0P_2TC_0C$:13FF::/64	
	2001:0DB8:$B_0P_1TC_0C$:13FF::/128	2001:0DB8:$B_0P_2TC_0C$:13FF::/128	No Asignada
	2001:0DB8:$B_0P_1TC_0C$:13FF::1/128	2001:0DB8:$B_0P_2TC_0C$:13FF::1/128	Loopback 1
	2001:0DB8:$B_0P_1TC_0C$:13FF::2/128	2001:0DB8:$B_0P_2TC_0C$:13FF::2/128	Loopback 2
	2001:0DB8:$B_0P_1TC_0C$:13FF::3/128	2001:0DB8:$B_0P_2TC_0C$:13FF::3/128	Loopback 3
México D.F.	2001:0DB8:$B_0P_1TC_0C$:09FF::/64	2001:0DB8:$B_0P_2TC_0C$:09FF::/64	
	2001:0DB8:$B_0P_1TC_0C$:09FF::/128	2001:0DB8:$B_0P_2TC_0C$:09FF::/128	No Asignada
	2001:0DB8:$B_0P_1TC_0C$:09FF::1/128	2001:0DB8:$B_0P_2TC_0C$:09FF::1/128	Loopback 1
	2001:0DB8:$B_0P_1TC_0C$:09FF::2/128	2001:0DB8:$B_0P_2TC_0C$:09FF::2/128	Loopback 2
	2001:0DB8:$B_0P_1TC_0C$:09FF::3/128	2001:0DB8:$B_0P_2TC_0C$:09FF::3/128	Loopback 3
Cd. Juárez	2001:0DB8:$B_0P_1TC_0C$:08FF::/64		
	2001:0DB8:$B_0P_1TC_0C$:08FF::/128		No Asignada
	2001:0DB8:$B_0P_1TC_0C$:08FF::1/128		Loopback 1
	2001:0DB8:$B_0P_1TC_0C$:08FF::2/128		Loopback 2
	2001:0DB8:$B_0P_1TC_0C$:08FF::3/128		Loopback 3
Cancún		2001:0DB8:$B_0P_2TC_0C$:17FF::/64	
		2001:0DB8:$B_0P_2TC_0C$:17FF::/128	No Asignada
		2001:0DB8:$B_0P_2TC_0C$:17FF::1/128	Loopback 1
		2001:0DB8:$B_0P_2TC_0C$:17FF::2/128	Loopback 2
		2001:0DB8:$B_0P_2TC_0C$:17FF::3/128	Loopback 3

6.2.8. Maqueta del direccionamiento IPv6 en el core de RedCUDI

La figura 139 y 140, muestra el Backbone de TELMEX y AVANTEL de la RedCUDI con el direccionamiento IPv6 que le corresponde a cada uno de los equipos de Backbone-Backbone y Backbone-Asociados Académicos, según el direccionamiento desarrollado en la sección 6.2. Para mayores detalles los mapas pueden ser encontrados en el disco anexo a la tesis con los nombres de backbone2-ipv6-2-DireDoc(02-2007) y backbone2-ipv6-3-DireDoc(02-2007).

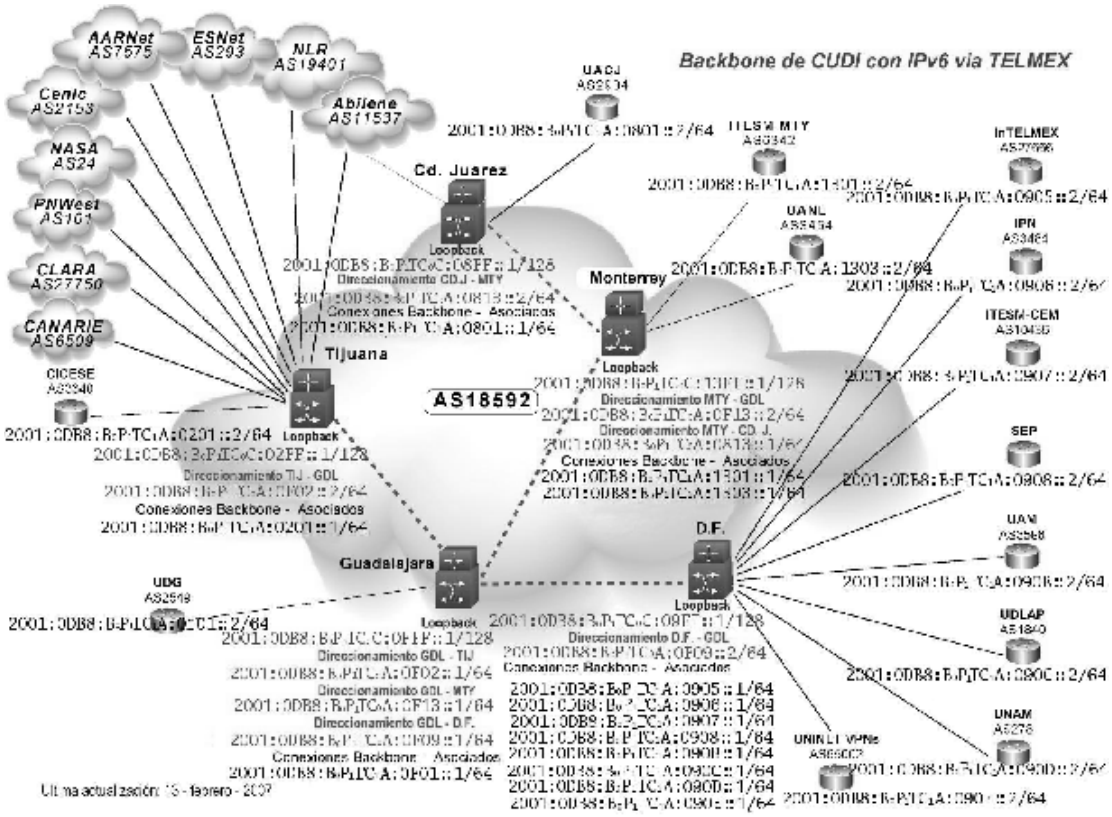


Figura 139. Direccionamiento IPv6 de los equipos de Backbone del proveedor TELMEX

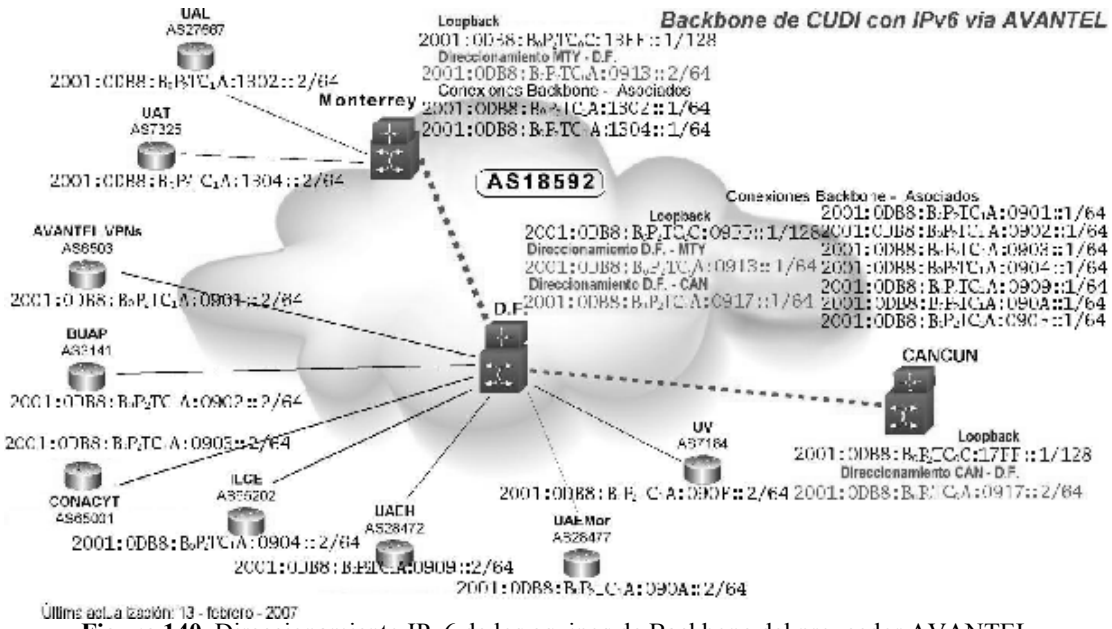


Figura 140. Direccionamiento IPv6 de los equipos de Backbone del proveedor AVANTEL

6.3. RENUMERACIÓN IPv6 EN RedCUDI

Después de haber sido aprobado el nuevo direccionamiento IPv6 por el CDR, se procedió a preparar y trabajar en un plan de renumeración y configuración de las conexiones entre los equipos del Backbone de la RedCUDI.

Durante mayo y junio de 2006 se trabajó con el NOC de CUDI en el plan de renumeración y configuración del Backbone, y finalmente, a principios de julio se anunció a los miembros del CDR que se encontraba renumerado parte del Backbone, y que el bloque IPv6 ya estaba siendo anunciado a Internet.

Cabe mencionar que nuevamente se configuraron los protocolos BGP4+ y RIPng mientras se instalan nuevas versiones del IOS en los equipos que soporten IS-IS para IPv6 que sustituirá finalmente a RIPng.

Posteriormente se reconfiguraron el resto de las conexiones y enlaces a las instituciones cuyos equipos de acceso a RedCUDI soportaban IPv6 y ya tenían una conexión previamente.

A continuación se muestra el documento, que sirvió de recomendación para llevar a cabo la renumeración del backbone de RedCUDI.

6.3.1. Procedimiento de Renumeración

El proceso de renumeración en RedCUDI consistió en remplazar el prefijo existente en producción asignado por la UNAM (2001:0448:03::/40), por el prefijo nuevo adjudicado por LACNIC (2001:1228::/32). Durante la renumeración, los prefijos-enlace del prefijo viejo, los cuales habían sido asignados a enlaces en la red, se reemplazaron por prefijos-enlace del nuevo prefijo. Las interfaces de los sistemas de la red se configuraron con direcciones IPv6 del prefijo-enlace del nuevo prefijo, y cualquier dirección del viejo prefijo en servicios como DNS o configurados dentro de los Switches, enrutadores y en las aplicaciones se remplazaron por la dirección apropiada del nuevo prefijo.

El procedimiento de renumeración se pudo haber aplicado a la red de forma completa o por partes. Para el caso de RedCUDI, le renumeración se trato por partes, ya que cada miembro realizó separadamente su propia renumeración.

6.3.2. Afectaciones por la renumeración

En una amplia variedad de lugares en los componentes de la red, se pueden encontrar direcciones IPv6 configuradas del viejo prefijo, las cuales son afectadas por el proceso de renumeración. A continuación se muestra la lista que se recomendó al NOC[32] tomar en cuenta antes de realizar la renumeración, y de esta forma evitar que se viera afectado algún servicio de la red.

[32] Centro de Operación de la Red CUDI

- Prefijos-enlaces asignados a los enlaces.
- Direcciones IPv6 asignadas a interfaces de Switches y enrutadores.
- Información de enrutamiento propagada por Switches y enrutadores.
- Filtros de Ingreso/Egreso.
- Listas de Acceso (ACLs).
- Entradas DNS.
- Información de configuración por DHCP.
- Direcciones IPv6 embebidas (embedded) en archivos de configuración.
- etc.

6.3.3. Antes de la Renumeración

1. La configuración del viejo prefijo debe estar funcionando de manera estable.
2. Obtener un nuevo prefijo de producción del RIR de la zona (LACNIC).
3. Diseño del direccionamiento con el nuevo bloque de direcciones IPv6 otorgado a CUDI, basándose en la estructura de la red.
4. Recopilación de información detallada acerca de cada prefijo actualmente asignado en cada interfaz de los enrutadores.
5. Elaboración de un documento en el que se establezca una relación del viejo prefijo, con el nuevo prefijo, y la interfaz a la que esta configurado, para informar qué prefijo nuevo sustituirá al prefijo viejo.
6. El documento elaborado tendrá que ser publicado para que todos lo miembros de la RedCUDI estén enterados de la renumeración.

6.3.4. Pasos de la Renumeración

La renumeración de la RedCUDI se realizó por fases, la tabla 57 muestra los tiempos tomados para la realización de la renumeración.

Tabla 57. Tabla cronológica de la renumeración en RedCUDI

Fase	Nombre	Mayo 2006	Junio 2006	Julio 2006	Agosto 2006
0	Diseño del Direccionamiento				
1	Renumeración Backbone Manual/Automático				
2	Levantamiento de las sesiones de IGP/RIP				
3	Levantamiento de las sesiones EGP/BGP4+				
4	Establecimientos de sesiones con los Asociados Académicos con bloque propio IPv6				
5	Asignación de nombres a las conexiones con IPv6				
6	Configuración en el				

	servidor DNS																
7	Configuración de IS-IS para IPv6 y pruebas*																

* Fecha aún por definir de acuerdo a la disponibilidad de las tarjetas PCMCIA que será confirmado por TELMEX

1. Documentar la relación del viejo prefijo con el nuevo prefijo y la interfaz del equipo al que pertenece, para que sirva de guía al momento de realizar la renumeración, y se eviten errores. La tabla 58 muestra dicha relación.

Tabla 58. Relación de prefijos a configurar en los equipos de Backbone de RedCUDI

Nodo	Conexión	Interfaz	Dirección vieja	Dirección nueva
Tijuana (Telmex)	Tij-Gdl	ATM1/0.1	2001:448:3:63::2	2001:0DB8:$B_0P_1TC_0$A:0F02::2
	Tij-CICESE	ATM1/0.3	2001:448:3:130::1	2001:0DB8:$B_0P_1TC_1$A:0201::1
		Loopback0	2001:448:3:33::1	2001:0DB8:$B_0P_1TC_0$C:02FF::1
	Tij-Canc	Tunnel1	2001:448:3:68::1	2001:0DB8:$B_0P_1TC_0$B:0217::1
Guadalajara (Telmex)	Gdl-DF	ATM1/0.1	2001:448:3:65::1	2001:0DB8:$B_0P_1TC_0$A:0F09::1
	Gdl-Mty	ATM1/0.2	2001:448:3:64::1	2001:0DB8:$B_0P_1TC_0$A:0F13::1
	Gdl-Tij	ATM1/0.3	2001:448:3:63::1	2001:0DB8:$B_0P_1TC_0$A:0F02::1
	Gdl-UDG	ATM1/0.4	2001:448:3:160::1	2001:0DB8:$B_0P_1TC_1$A:0F01::1
		Loopback0	2001:448:3:66::1	2001:0DB8:$B_0P_1TC_0$C:0FFF::1
Monterrey (Telmex)	Mty-Gdl	ATM1/0.1	2001:448:3:64::2	2001:0DB8:$B_0P_1TC_0$A:0F13::2
	Mty-DF	No existe*	Conexión x Avantel	
	Mty-CdJuarez	ATM1/0.5	2001:448:3:67::1	2001:0DB8:$B_0P_1TC_0$A:0813::1
	Mty-ITESM	ATM1/0.3	2001:448:3:140::1	2001:0DB8:$B_0P_1TC_1$A:1301::1
		Loopback0	2001:448:3:44::1	2001:0DB8:$B_0P_1TC_0$C:13FF::1
D.F. (Telmex)	DF-Gdl	ATM1/0.1	2001:448:3:65::2	2001:0DB8:$B_0P_1TC_0$A:0F09::2
	DF-Canc	No existe*	Conexión x Avantel	
	DF-Mty	No existe*	Conexión x Avantel	
	DF- UAEMor	ATM1/0.7		2001:0DB8:$B_0P_1TC_1$A:090A::2
		Loopback2	2001:448:3:55::1	2001:0DB8:$B_0P_1TC_0$C:09FF::1
	DF-UDLAP old DF-IPN new	Tunnel0	2001:448:3:152::1	2001:0DB8:$B_0P_1TC_1$B:0906::1
	DF-UNAM old DF- UAEMor new	Tunnel1	2001:448:3:153::1	2001:0DB8:$B_0P_1TC_1$B:090A::1
Cd. Juárez (Telmex)	Cd.Juarez-Mty	ATM2/0.1	2001:448:3:67::2	2001:0DB8:$B_0P_1TC_0$A:0813::2
		Loopback0	2001:448:3:77::1	2001:0DB8:$B_0P_1TC_0$C:08FF::1
Cancún (Avantel)	DF-Canc	No existe*	Conexión x Avantel	
		Loopback0	2001:448:3:88::1	2001:0DB8:$B_0P_2TC_0$C:17FF::1
	Tij-Canc	Tunnel0	2001:448:3:68::2	2001:0DB8:$B_0P_1TC_0$B:0217::2

* Conexiones realizadas por equipos de Avantel, y actualmente no tienen soporte IPv6.

2. Realizar un respaldo final de las configuraciones actuales, de cada uno de los equipos que se verán involucrados en la renumeración.
3. La renumeración del Backbone puede realizarse de dos maneras:
    a. Manual
        i. Configurar las direcciones IPv6 de enlaces del prefijo nuevo, en cada una de las interfaces de los enrutadores, según lo muestra la tabla 57, sin borrar aún las direcciones IPv6 de enlaces del prefijo viejo.

ii. Probar conectividad y mantener operando en paralelo ambas direcciones del viejo y nuevo prefijo.
iii. Después de haber levantado las sesiones de protocolos IGP y EGP, y verificar que el prefijo nuevo se encuentren funcionando de manera estable en la red, borrar manualmente los prefijos viejos en las interfaces de los enrutadores.

b. Automático
   i. (RFC2894)
      1. Instalar claves de seguridad en cada enrutador.
      2. Hacer una asociación de seguridad entre los enrutadores.
      3. Configurar los mensajes de Router-Renumering, para la renumeración de los enrutadores.
      4. Los enrutadores se re-configurarán automáticamente y guardaran su nueva configuración en la memoria no volátil.
      5. Los enrutadores envían los mensajes de re-configuración a sus vecinos, los cuales propagaran el Router-Renumering. Los mensajes son enviados a la dirección Multicast de todos los enrutadores.
   ii. Demonio
      1. Instalar en plataforma FreeBSD la utilidad de **rrenumd**.
      2. Configurar un archivo o Script, en donde se configurará los mensajes de Router-Renumering, según **rrenumd.conf**.
      3. Después del periodo de transición, los prefijos viejos se convirtierán en inválidos y serán eliminados automáticamente.

4. Levantamiento de las sesiones IGP/RIPng entre los enrutadores del backbone. No hubo necesidad de realizar nada, ya que RIPng ya se encontraba habilitado en cada interfaz del enrutador.
5. Levantamiento de las sesiones EGP/BGP4+, tanto iBGP y eBGP, entre los enrutadores del backbone y peer´s de CUDI.
   a. Migración de las sesiones iBGP al nuevo prefijo.
   b. Migración de las sesiones eBGP al nuevo prefijo, de cada uno de los peer´s de RedCUDI.
6. Actualización DNS
7. Migración del protocolo de enrutamiento RIPng a IS-IS. Pendiente, hasta que se aumente la capacidad de las memorias a los enrutadores para que soporten IS-IS.

6.4. ENCUESTA DEL SOPORTE IPv6 A LOS MIEMBROS DE CUDI

Durante el mes de agosto de 2006, el CDR (Comité de desarrollo de la red) realiza una encuesta a los miembros de CUDI (Asociados y Afiliados, ambos académicos), que tuvo como principal propósito recoger información de la situación de las instituciones dentro de RedCUDI. La encuesta fue dividida en 9 secciones, las cuales son las siguientes:

- Estado Actual de la Institución.
- Dirección de Informática y Telecomunicaciones.

- Departamento de Telecomunicaciones.
- Recursos Humanos.
- Conectividad de la Institución.
- Infraestructura de Comunicaciones.
- Infraestructura de Seguridad.
- Infraestructura de Videoconferencia.
- Infraestructura de Voz.

Dentro de estas secciones de la encuesta se elaboraron preguntas relacionadas a temas de interés particular para el grupo de trabajo IPv6, las cuales son mostradas en la tabla 59.

Tabla 59. Preguntas de interés para el grupo de trabajo IPv6.

Sección	**Pregunta**
Recursos Humanos	¿Qué necesidades de capacitación requiere que le proporcione el CDR-CUDI?
Conectividad de la Institución	Bloque(s) de Direcciones IPv6 asignadas:
Infraestructura de Comunicaciones	¿Utiliza Switches en su LAN? :
	¿Qué fabricante de switch? ¿Qué modelos de switch? En caso de contar con fabricantes, especificar el nombre y modelo de cada uno:
	¿Utiliza Ruteadores al interior de tu Red?:
	¿Qué marcas de ruteadores? ¿Qué modelos de ruteadores?. En caso de contar con fabricantes, especificar el nombre y modelo de cada uno:
	¿Qué protocolo IGP (ruteo interno) utiliza en tu Red [RIP | OSPF | EIGRP | Rutas Estáticas]? En caso de utilizar varios protocolos, favor de indicarlo:

Se debe destacar que al momento de realizar la encuesta se contaba con sólo 141 miembros que conformaban a los Asociados Académicos y Afiliados Académicos, de los cuales sólo 28 contestaron la encuesta, tomando en cuenta que los grupos de Asociados Académicos como los Centros Públicos de Investigación CONACYT, Dirección General de Educación Superior y tecnología, Institutos Nacionales de Salud y el Subsistema de Universidades Politécnicas fueron contabilizados por separado cada una de sus instituciones integrantes, es decir de forma desagrupada, lo que da como resultado los 141 miembros Académicos. La tabla 60 muestra los miembros que contestaron la encuesta y en qué categoría se encuentran inscritos en CUDI.

Tabla 60. Miembros que contestaron la encuesta realizado por el CDR.

Tipo de Miembro	**Miembros**	**Integrantes**
Asociado Académico	Centros Públicos de Investigación CONACYT	CIATEJ CIDE CIDETEQ CIMAT COLMICH COLSAN CENTROGEO INAO INECOL
	ILCE	
	IPN	

	ITESM	
	Institutos Nacionales de Salud	INER INPER INSP
	Subsistema de Universidades Politécnicas	UP-Tulancingo
	UAX	
	UAL	
	UDG	
	UAEM	
	UV	
Afiliados Académicos	IIE	
	LANIA	
	UAG	
	UASLP	
	ULSA	
	UNISON	
	UR	

La tabla 61 muestra la información obtenida de las encuestas contestadas por los miembros listados en la tabla 60, y tomando en cuenta solamente las preguntas de interés particular para el grupo de trabajo IPv6 mencionadas en la tabla 59. Los miembros son listados alfabéticamente, e indicando por medio de la columna AA que es un Asociado Académico y con AF que es un Afiliado Académico.

Tabla 61. Resultados sobre IPv6 obtenidos de la encuesta realizada por el CDR.

#	Institución			Cuenta con Bloque(s) de Direcciones IPv6 asignadas:		Solicita capacitación IPv6 por medio del CDR-CUDI	
		AA	AF	SI	NO	SI	NO
1	CIATEJ*	X			X		X
2	CIDE*	X			X		X
3	CIDETEQ*	X			X		X
4	CIMAT*	X			X	X	
5	COLMICH*	X			X		X
6	COLSAN*	X			X		X
7	CENTROGEO*	X			X		X
8	IIE		X		X		X
9	ILCE	X			X		X
10	INAOE*	X		X			X
11	INECOL*	X			X	X	
12	INER***	X			X		X
13	INPER***	X			X		X
14	INSP***	X			X		X
15	IPN	X			X		X

16	ITESM	X			X		X
17	LANIA		X		X	X	
18	UAG		X		X		X
19	UAL	X		X		X	
20	UASLP		X		X		X
21	UAX	X			X	X	
22	UDG	X		X			X
23	ULSA		X		X	X	
24	UNISON		X		X		X
25	UAEM	X			X		X
26	UP-Tulancingo****	X			X		X
27	UR		X		X		X
28	UV	X			X		X

*Centros Públicos de Investigación CONACYT
**Dirección General de Educación Superior Tecnológica
***Institutos Nacionales de Salud
****Subsistemas de Universidades Politécnicas

Los datos obtenidos en la tabla 61, son graficados, de forma que se obtiene una gráfica por cada pregunta. Hay que mencionar que las gráficas obtenidas serán de forma desagrupada, ya que existen Asociados Académicos que representan a grupos de integrantes, por ejemplo, los miembros Centros Públicos de Investigación CONACYT, Dirección General de Educación Superior Tecnológica, Institutos Nacionales de Salud y Subsistemas de Universidades Politécnicas. Cabe señalar esto, porque en el reporte que se realizó para el CDR se añadieron otras gráficas en las que se tomaba en cuenta a cada miembro que representa un grupo como un todo, pero para propósitos de la tesis con las graficas a las que se les llamo en el reporte "desagrupadas" es suficiente, ya que están más apegadas a la realidad.

6.4.1. Gráficas obtenidas de los resultados arrojados de la encuesta realizada por el CDR de CUDI

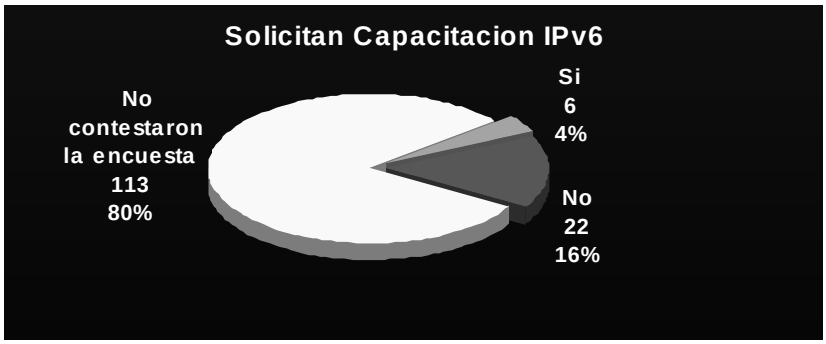
- Gráfica de los miembros que cuentan con bloques de direcciones IPv6

La gráfica 1 muestra los resultados obtenidos de la tabla 60, de la pregunta "¿Cuentan con bloque(s) de direcciones IPv6 asignadas?". De los 141 miembros académicos que conforman el 100% de los integrantes de CUDI, sólo 28 miembros contestaron la encuesta, que representa el 20% del total, y los 113 (80%) restante se abstuvieron de contestar la encuesta. El 2% de los miembros, en este caso 3, contestaron que cuentan con bloque(s) de direcciones IPv6, contra un 18%, 25 miembros, que dijeron no contar con bloques de direcciones IPv6.

Gráfica 1. Resultados de encuesta de los miembros que cuentan con bloques IPv6 (Desagrupados)

- Gráficas de los miembros que solicitan capacitación de IPv6 a CUDI

La gráfica 2 muestra los resultados obtenidos de la tabla 60, de la pregunta abierta "¿Qué necesidades de capacitación requiere que le proporcione el CDR-CUDI?". Esta gráfica muestra que 6 (4%) de los integrantes académicos de CUDI, que contestaron la encuesta, están solicitando capitación IPv6 a CUDI, mientras 22 (16%), de los que contestaron la encuesta, no solicitan capitación IPv6, y los restante 113 (80%), son los que se abstuvieron de contestar la encuesta.



Gráfica 2. Resultados de encuesta de los miembros que solicitan capitación de IPv6 a CUDI (Desagrupados).

6.4.2. Investigación realizada acerca de bloques ipv6 con los que cuentan los miembros de CUDI.

Al mismo tiempo que se realiza la encuesta del CDR, el grupo de trabajo IPv6 realiza una recopilación interna de información, acerca de los bloques de direcciones IPv6 con los que cuentan cada uno de los miembros de CUDI. De tal investigación se obtuvieron las siguientes tablas en las que se muestra información relacionada con miembros que actualmente tienen bloques de direcciones IPv6 y los que tuvieron en algún tiempo. Los miembros actuales que ya no cuentan con bloques de direcciones IPv6 principalmente son atribuidos a la conclusión de la red de prueba de 6BONE el 6 de junio de 2006. De los miembros que actualmente cuentan con bloques de direcciones IPv6, algunos lo han

recibido del bloque propio IPv6 de producción de la UNAM 2001:448::/32, y otros cuentan con bloques propios adjudicados.

La tabla 62, muestra los 16 miembros que tuvieron bloques de direcciones IPv6, hasta el cierre del proyecto de 6BONE. La tabla 63, muestra los miembros que hasta la fecha de la elaboración de este documento, cuentan con bloque de direcciones IPv6. AVANTEL, ITESM, TELMEX, UAEH, UDG y UNAM son los miembros que cuentan con bloque propio IPv6 adjudicado.

Tabla 62. Miembros que han tenido bloques de direcciones IPv6

#	**Institución**		
		AA	**AF**
1	CIC-IPN	X	
2	CICESE	X	
3	INAOE[1]	X	
4	INFOTEC[1]	X	
5	ITAM		X
6	ITESM	X	
7	ITMerida[2]	X	
8	ITO[2]	X	
9	LANIA		X
10	UABC		X
11	UAEH	X	
12	UAL	X	
13	UCOL		X
14	UDG	X	
15	ULSA		X
16	UNAM	X	

Tabla 63. Miembros que tienen actualmente bloque de direcciones IPv6

#	Institución		
		AA	AF
1	AVANTEL*	X	
2	INAOE[1]	X	
3	INFOTEC[1]	X	
4	ITESM	X	
5	TELMEX (UNINET)*	X	
6	UAEH	X	
7	UDG	X	
8	ULSA		X
9	UNAM	X	

*Asociados Institucionales
1.-Centros Públicos de Investigación CONACYT
2.-Dirección General de Educación Superior Tecnológica
3.-Institutos Nacionales de Salud

6.4.2.1.Gráficas obtenidas de la investigacion realizada por el grupo de trabajo IPv6

Las gráficas obtenidas en esta sección, fueron elaboradas con la misma metodología del apartado anterior. Es decir en el reporte presentado al CDR se manejaron otros tipos de gráficas, pero para propósitos de la tesis solo se utilizaran las gráficas que en el reporte se les llamó desagrupadas, ya que están más apegadas a la realidad.

- Gráficas de los miembros que tienen o han tenido bloques IPv6

La gráfica 3 muestra los resultados logrados de la investigación, presentando lo siguiente: 125 (88%) de los miembros académicos de CUDI, tanto Afiliados como Asociados, nunca han tenido bloque de direcciones IPv6, los 16 (12%) restantes son los que sí han tenido bloque de direcciones IPv6. De este porcentaje 11 (8%) son Asociados Académicos y 5 (4%) son Afiliados Académicos.

Gráfica 3. Resultados obtenidos de la investigación realizada. Miembros que tienen o han tenido bloque de direcciones IPv6 (Desagrupados).

- Gráficas de los Asociados Académicos que tienen o han tenido bloques IPv6

La gráfica 4 muestra los resultados logrados de la investigación, presentando lo siguiente: 84 (89%) del total de los Asociados Académicos nunca han tenido bloque de direcciones IPv6, y los 11 (11%) restante son los Asociados Académicos que tienen o han tenido un bloque IPv6. Del porcentaje que han tenido un bloque, 6 (6%) de los Asociados Académicos actualmente cuentan con un bloque, y el restante 5 (5%) lo tuvieron en algún momento.

Gráfica 4. Resultados obtenidos de la investigación realizada. Asociados Académicos que tienen o han tenido bloque de direcciones IPv6 (Desagrupados).

- Gráficas de los Afiliados Académicos que tienen o han tenido bloques IPv6

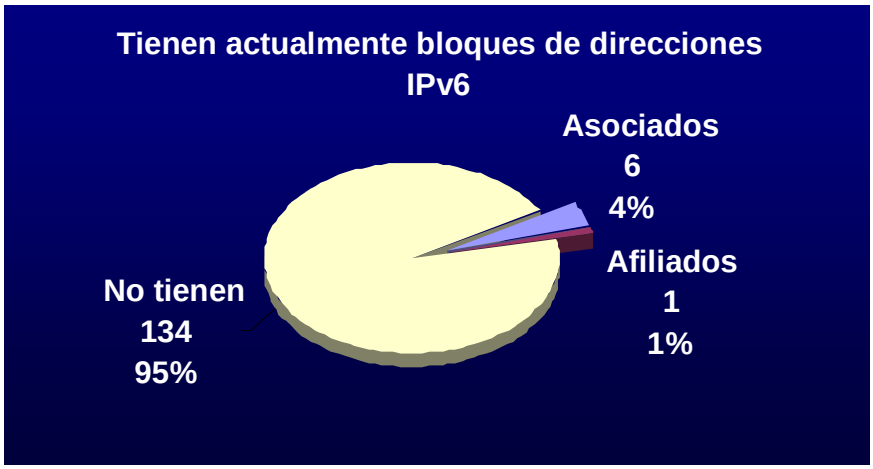
La gráfica 5 muestra los resultados logrados de la investigación, presentando lo siguiente: 41 (89%) del total de los Afiliados Académicos nunca han tenido bloque de direcciones IPv6, mientras los restantes 5 (11%) ya han tenido alguna relación con IPv6. Del porcentaje que ya han tenido IPv6, 1 (2%) de los Afiliados Académicos actuales cuentan con un bloque y los otro 4 (9%) lo tuvieron en algún momento. El número total de Afiliados Académicos hasta el momento de la elaboración del reporte al CDR fue de 46 miembros.

Gráfica 5. Resultados obtenidos de la investigación realizada. Afiliados Académicos que tienen o han tenido bloque de direcciones IPv6.

- Gráficas de los miembros académicos que cuentan con bloques de direcciones IPv6 en la red de CUDI hasta el momento de realizar el reporte al CDR.

La gráfica 6 muestra la cantidad de miembros académicos que actualmente cuentan con bloque de direcciones IPv6, lo que da una idea del soporte IPv6 que se tiene internamente en los miembros académicos de CUDI. Los resultados obtenidos de la investigación, reflejados por la gráfica son los siguientes: 134 (95%) de los miembros académicos, con los que cuenta actualmente la red de CUDI, no tienen bloque de direcciones IPv6, y los 7 (5%) restantes son la cantidad de miembros académicos que actualmente sí cuentan con un

bloque. Del porcentaje que cuentan con bloques de direcciones IPv6, 6 (4%) son Asociados Académicos y 1 (1%) son Afiliados Académicos.



Gráfica 6. Resultados obtenidos de la investigación realizada. Miembros académicos totales que actualmente cuentan con bloque de direcciones IPv6 en la red de CUDI (Desagrupados).

6.5. POLÍTICAS DE RUTEO IPv6 Y DE ASIGNACIÓN DE BLOQUES DE DIRECCIONES IPv6 EN RedCUDI

Con la adquisicion del nuevo bloque de direcciones IPv6 asignado a CUDI por medio de LACNIC, y el desarrollo de la documentacion del direccionamiento y renumeración con el hasta entonces nuevo bloque IPv6, se elaboraron en paralelo propuestas de documentos sobre políticas de uso y asignaciones de bloques IPv6 en CUDI, con la intención que llegaran a formar parte de los RFCMX.

Un RFC (Request for Comments), es un documento estándar que describe protocolos, sistemas o procedimientos utilizados por la comunidad de Internet. En México, la RedCUDI retoma el término y le agrega la terminación MX (México) con el fin de generar los documentos de estándares y disposiciones de uso, desarrollo e implementación de la RedCUDI y sus aplicaciones.

El objetivo de los RFCMX es tener una serie de documentos que sean base y lineamientos para los desarrollos de las aplicaciones, las tecnologías a utilizar y aquellas propuestas que van encaminadas a utilizar y a mejorar la RedCUDI.

Después de arduo trabajo sobre dichos documentos y puestos a discusión en el CDR, fueron finalmente aceptados y publicados sus últimas versiones el 12 de marzo de 2007 en la pagina http://rfc.cudi.edu.mx, bajo los nombres RFCMX-3 “Políticas de ruteo IPv6 en RedCUDI” y RFCMX-4 “Políticas de asignación de bloque de direcciones IPv6 en CUDI”.

Con la publicación de estos RFCMXs, se inició y abrió la posibilidad de manera formal a los miembros de CUDI, de iniciar el proceso de solicitud de bloques de direcciones IPv6 por un año, mientras justifican su bloque propio ante NIC-México.

Para la solicitud de bloques de direcciones IPv6, por parte de los miembros de CUDI, se elaboró un documento que lleva por nombre "Procedimiento y Requisitos para la recepción de bloques IPv6 de parte de CUDI v1.1", el cual tiene como propósito guiar al miembro interesado en adquirir un bloque de direcciones IPv6 ante CUDI, y fue publicado, al mismo tiempo que los RFCMX, en la página del grupo de trabajo de IPv6.

A continuación se muestran los documentos antes mencionados, cabe señalar que los clientes se verán obligados a cumplir con las políticas y lineamientos establecidos en los documentos llamados RFCMX, con el objeto de un buen uso de los recursos de la red.

6.5.1. RFCMX-3 “Políticas de ruteo IPv6 en RedCUDI”

En este documento se describieron las políticas de Ruteo IPv6 que servirán de lineamientos a los miembros actuales y posteriores que se conecten a la red de Internet2 de México (RedCUDI), así como el tráfico que se permitirá transportar o no en esta red. La implementación de estas políticas ayudará a prevenir el crecimiento incontrolable de las tablas de enrutamiento y detener los anuncios no autorizados.

6.5.1.1.Obligaciones de los Asociados Académicos

Los Asociados Académicos tienen la obligación de implementar y operar su ruteo IPv6, de manera que el desempeño de la red CUDI y la del propio Asociado, no se vea afectado negativamente. Particularmente en lo que se refiere a las redes anunciadas por los Asociados Académicos, y a las redes recibidas de enlaces internacionales.

6.5.1.2.Políticas de Ruteo IPv6 en RedCUDI

Las políticas de ruteo para IPv6 a implementar son referentes a: las redes que no deben ser anunciadas hacia la red, conocidas como “Improper routes” (direcciones de red que no deben ser anunciadas globalmente hacia Internet) y al tamaño de las redes recibidas por parte de los Asociados Académicos conocido como “Import routing policy” (políticas de acceso para las redes de los Asociados y Afiliados Académicos).

Con base en el direccionamiento IPv6 realizado, se establecieron las siguientes Políticas de Ruteo IPv6 en RedCUDI.

- Aceptación temporal de tráfico IPv6 con fines experimentales, proveniente de los miembros de CUDI, en un periodo de hasta 1 año siempre y cuando este tráfico no afecte a las demás aplicaciones y en general al tráfico académico existente, pudiéndose fijar límites y los filtros necesarios. Esta política estará sujeta a modificaciones cuando se considere pertinente.

- Aceptación de los prefijos 6to4 (solo /16), siempre y cuando este tráfico no afecte a las demás aplicaciones y en general al tráfico académico existente, pudiéndose fijar limites y los filtros necesarios. Esta política estará sujeta a modificaciones cuando se considere pertinente.

- En la RedCUDI sólo se anunciará el bloque propio /32, y se realizará agregación y sumarización del mismo sin embargo, se aceptarán bloques /32 de los miembros de CUDI del tipo agregable global, definidos internacionalmente en la IANA, previo acuerdo de agregación y sumarización de su parte, para ser anunciados.
  De tal manera que se garantice el anuncio hacia las redes académicas internacionales (Abilene, Cenic, CLARA y GÉANT).

- De acuerdo a la "Import routing policy" cualquier otro bloque /32 del tipo agregable global, será aceptado únicamente de otras redes avanzadas e instituciones de carácter académico, o de otras entidades con las cuales CUDI haya establecido algún convenio, previo acuerdo de agregación y sumarización de su parte.

- Sólo se aceptarán bloques /48 del tipo agregable global de los miembros de CUDI que hayan sido asignados por la misma corporación.

De lo anterior se presenta la siguiente recomendación:

A las instituciones pertenecientes a CUDI se les recomienda adquirir su propio bloque de direcciones IPv6, a fin de evitar el anuncio de bloques de mayor tamaño y de garantizar la salida hacia las redes internacionales.

- Cualquier otro bloque /48 será filtrado temporalmente y se solicitará a quien este anunciándolo que deje de hacerlo.
  En caso de seguir recibiendo al bloque /48 se enviaran tres avisos a la otra parte, y se dará de baja la conexión en algunos casos, con previo aviso; y en otros se pondrá necesariamente un filtro permanente, en el equipo de backbone correspondiente.

- Los prefijos conocidos como "Improper routes" mostrados en la tabla 64 no se anunciarán para evitar la propagación de estas rutas.

Tabla 64. Prefijos que deben ser filtrados.

Notación IPv6	Tipo de dirección
::/128	No especificada
::/96	Reservada por IETF*
::1/128	Loopback
::ffff./96	Mapeadas-IPv4
2001:DB8::/32	Para documentación
FC00::/7	Unicast local única
FE80::/10	Unicast de enlace local
FEC0::/10	Reservada por IETF**
FF00::/8	Multicast
3FFE::/16	6Bone ***

* Fue definida como un prefijo de dirección IPv6 IPv4-compatible, actualmente ha sido revocada por IETF.
** Fue definida como un prefijo de dirección de ámbito de sitio local, actualmente ha sido revocada por IETF.
*** Prefijo utilizado en la red mundial experimental 6Bone, ya no debe usarse en el Internet como lo establece el RFC 3701.

6.5.1.2.1. Políticas de Ruteo IPv6 sobre prefijos de 6Bone

Se aceptaron (no filtrado) prefijos de 6BONE temporalmente hasta /32 de los miembros de CUDI por plazos hasta de 9 meses, que no pudieron extenderse más allá del 6 de junio de 2006 como lo establece el RFC 3701.

- **De Entrada:**
  Se aceptaron los prefijos de 6Bone de los miembros de CUDI mientras renumeraron sus redes, en plazos de 1 ó 2 meses desde el momento de su conexión.
- **De salida:**
  Se anunciaron los prefijos de 6Bone de aquellas instituciones localizadas físicamente en México que hubieron manifestaron su compromiso de renumerar en el plazo establecido desde el momento de su conexión a CUDI.
- Esta política de no filtrado tuvo vigencia hasta el 6 de junio de 2006, y a partir de esta fecha se tienen que filtrar estos prefijos.

6.5.2. RFCMX-4 "Políticas de asignación de bloque de direcciones IPv6 en CUDI"

En este documento se describieron las políticas de Asignación que servirán de lineamientos a los miembros actuales y posteriores que requieran un bloque de direcciones IPv6 de parte de CUDI para conectarse a la red de Internet2 de México (RedCUDI), a partir del bloque propio adjudicado a CUDI por LACNIC.

6.5.2.1.Políticas de Asignación de bloque de direcciones IPv6 en RedCUDI

Asignación:

- Se asignaran bloques por parte de CUDI sin costo alguno a los Asociados y/o Afiliados ambos Académicos, mientras no apliquen o tengan experiencia para justificar a NIC-México su bloque propio, por un periodo de 1 año, con posibilidad de renovación, de acuerdo a los procedimientos de asignación establecidos.
- Se requerirá que cada miembro empiece a usar el bloque asignado en un plazo máximo de 3 meses.
- Las asignaciones serán de bloques /48 para los Asociados Académicos y estos asignarán bloques /56 a sus Afiliados Académicos.
- Los bloques asignados únicamente podrán usarse y anunciarse en la RedCUDI teniendo como proveedor de servicio, al Asociado correspondiente; en el caso de los Afiliados y a CUDI en el caso de los Asociados. Para los Afiliados Académicos que se conectan por VPN, el bloque asignado por CUDI, únicamente podrá usarse y anunciarse a través de esta conexión.
- Sólo se aceptaran bloques /48 del tipo agregable global, definidos internacionalmente en la IANA, de los miembros de CUDI que hayan sido asignados por la misma corporación.

De lo anterior se presenta la siguiente recomendación:

A las instituciones pertenecientes a CUDI se les recomienda adquirir su propio bloque de direcciones IPv6, a fin de evitar el anuncio de bloques de mayor tamaño y de garantizar la salida hacia las redes internacionales.

6.5.3. Procedimiento y Requisitos para la recepción de bloques IPv6 de parte de CUDI v1.1

Tomando como referencia el RFCMX-4 "Políticas de asignación de bloque de direcciones IPv6 en CUDI", los miembros que requieran solicitar un bloque IPv6 /48 calificando para una asignación inicial, deberán cumplir con los siguientes requisitos:

1. Ser un Asociado Académico, es decir, Universidades, Instituciones de Educación Superior e Investigación del país, que adquieren el compromiso financiero de absorber a prorrata el costo de mantener la red operando. Forman parte del Consejo Directivo;
2. Tener vigente las cuotas anuales por concepto de membresía;
3. No ser un sitio final (usuario final);
4. Documentar un plan detallado sobre los recursos, servicios y conectividad en IPv6 que empleara;
5. En caso de tener Afiliados, documentar un listado de quienes son, y direccionamiento a emplear;
6. Anunciar en el sistema de rutas inter-dominio de Internet un único bloque (/48), que agregue toda la asignación de direcciones IPv6 recibida, en un plazo no mayor de 12 meses;
7. Ofrecer servicios en IPv6 a miembros localizados físicamente en México.

Antes de hacer la solicitud de un bloque IPv6, se recomienda leer el RFCMX-3 "Políticas de ruteo IPv6 en RedCUDI".

Para solicitar un bloque IPv6 la organización debe llenar el "Formulario IPv6" (www.ipv6.unam.mx/Internet2/Formulario-Bloque-IPv6-v1.txt) y enviarlo después a la cuenta de correo del coordinador del Grupo de Trabajo de IPv6.

Una vez aprobada la solicitud de asignación inicial, deberá comunicarse el contacto técnico del Asociado Académico con el responsable del Grupo de Trabajo de IPv6, con copia al personal del NOC de CUDI, para programar la conexión IPv6 del Asociado al Backbone de la RedCUDI a partir del bloque delegado, y para revisar los detalles técnicos de la misma.

6.5.3.1.Formulario para solicitar Bloque IPv6 en CUDI

```
#  No remover el número de la version
CUDI México IPV6 Template 20070228-2-SP

#  Envíe esta solicitud a azael@ipv6.unam.mx

#  Información sobre el Asociado Académico que esta solicitando
#  el bloque IPv6.
```

Si el Asociado Académico ya tiene algún recurso registrado con
NIC-MEXICO, informar solamente su "ownerID".
En caso de no saber cuál es el "ownerID", consultar algún recurso
adjudicado a su organización en el servidor WHOIS
de NIC-MEXICO.
[http://www.nic.mx/es/Busqueda.Who_Is]

0a. ID. del Asociado Académico (OwnerID):
0b. Nombre de la Institución:
0c. Dirección Postal:
0d. Ciudad:
0e. Estado:
0f. País:
0g. Código Postal:
0h. Teléfono

Puntos de contacto con el Asociado Académico.
Será necesario informar contacto técnico y
de membresía con CUDI.
Informar también el "UserID" de los puntos de contacto.
El “User-ID” es una clave única que consiste de las iniciales
del contacto técnico (4 caracteres).

1a. Nombre del contacto técnico:
1b. E-correo del contacto técnico:
1c. Nombre del contacto institucional:
1d. E-correo del contacto institucional:
1e. ID contacto técnico (UserID):

Brindar información sobre el Asociado Académico que solicita el
bloque IPv6.

2a. Información del Asociado Académico:

Informar el plan para despliegue de la red IPv6 del
Asociados Académicos, el plan de utilización de la direcciones IPv6
y plan de sub-asignaciones de direcciones IPv6 para los
Afiliados.

3a. Fecha:
3b. Plan de utilización:
3c. Plan de Asignación:

Brindar información sobre la estructura de la red
IPv6 y tipos de servicios que serán usados.

4. Información Adicional:

No remover esta línea
Final del formulario

6.6. ACTUALIZACIÓN DE LOS NOMBRES DNS DE LOS EQUIPOS DE BACKBONE.

En mayo de 2007, después de la aceptación y publicación de los RFCMX, se desarrolla una nueva propuesta para la asignación de recursos DNS, a cada una de los enlaces de los equipos de backbone de la RedCUDI, del tipo AAAA o A6 para la parte IPv6, ya que hasta el momento sólo hay configurados registros A para IPv4 en los servidores DNS.

A continuación se muestra dicho documento; destacando que hasta el momento de redactar esta parte de la tesis no ha sido aprobada, ya que resta ponerlo a discusión ante el comité del CDR.

6.6.1. Nombres DNS de los enlaces de los equipos de backbone de la RedCUDI.

De la misma forma que se le da un nombre DNS a un recurso de Internet, también es posible asignar un nombre DNS a las IP de las interfaz de los enrutadores del backbone de RedCUDI, de manera que al utilizar herramientas de diagnostico como el traceroute, los resultados obtenidos, devuelvan los nombres de los saltos (hop´s) por los que pasa el paquete para llegar a su destino final.

A continuación se muestra los resultados obtenidos al realizar un traceroute del nodo de Tijuana-TELMEX al nodo de Cd. Juarez-TELMEX[33] por medio de IPv4; no es realizado por IPv6 porque aun no se han configurado los registros AAAA o A6[34] (El RFC3363 recomienda usar AAAA para producción y A6 para pruebas) en los servidores DNS de CUDI.

Traceroute del nodo CUDI-TIJ-TELMEX(Tijuana, B.C) a 200.23.60.146.

Router Response:

Type escape sequence to abort.
Tracing the route to juarez-mty7200.core.cudi.edu.mx (200.23.60.146)

1 gdl-tijuana.core.cudi.edu.mx (200.23.60.242) 36 msec 40 msec 36 msec
2 mty7200-gdl.core.cudi.edu.mx (200.23.60.233) 56 msec 56 msec 56 msec
*3 juarez-mty7200.core.cudi.edu.mx (200.23.60.146) 72 msec * 68 msec*

El resultado muestra que, se debe pasar por tres saltos (hop´s), para alcanzar a la dirección IPv4 200.23.60.146 con nombre *juarez-mty7200.core.cudi.edu.mx*. Un salto (hop) es un

[33] http://www.noc.cudi.edu.mx/
[34] R. Bush, A. Durand, B. Fink, O. Gudmundsson, T. Hain (2002). RFC3363 "Representing Internet Protocol version 6 (IPv6) Addresses in the Domain Name System (DNS)". Informational. Pp. 1.

término usado para describir a cada enrutador que un paquete alcanza para llegar a su destino final. Cada salto muestra el nombre DNS que le corresponde, según se ha configurado en el servidor DNS; en seguida, las direcciones IP de las interfaces del enrutador alcanzado, y finalmente los contadores obtenidos de las pruebas propias de la herramienta traceroute realizadas.

Los nombres DNS obtenidos en la prueba efectuada, dan un sentido erróneo de la dirección que están siguiendo los paquetes para alcanzar a su destino. Por ejemplo, el primer salto obtenido indica, que el paquete enviado de la prueba traceroute, va del origen *gdl* (Guadalajara-TELMEX) hacia el destino *tijuana* (Tijuana-TELMEX); el segundo salto indica que el paquete va del origen *mty7200* (Monterrey-TELMEX) hacia el destino *gdl* (Guadalajara-TELMEX), y así mismo el tercero, va del origen *juarez* (Cd. Juárez-TELMEX) hacia el destino *mty7200* (Monterrey-TELMEX).

La figura 141 muestra en forma gráfica el sentido en que se realiza la prueba traceroute, y los puntos rojos mostrados son los saltos que entrega la prueba, lo que demuestra la manera en que han sido asignados los nombres DNS se encuentra configurados de manera incorrecta.

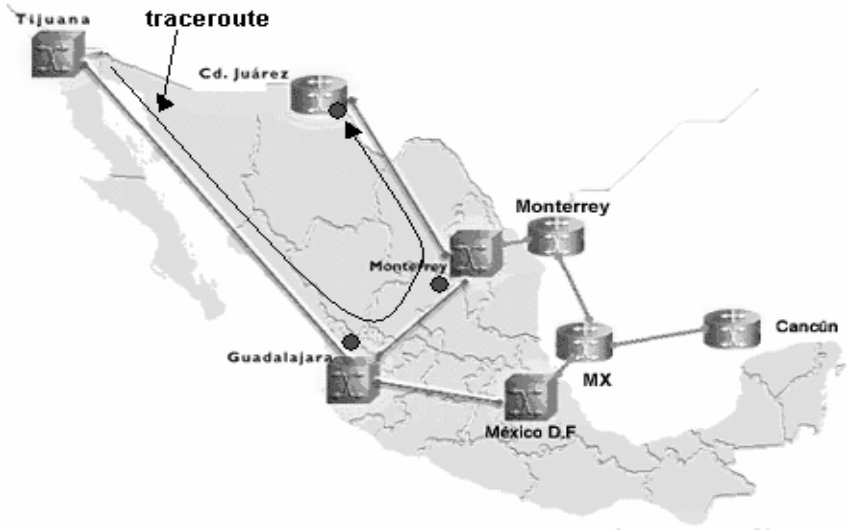


Figura 141. Traceroute de Tijuana a Cd. Juárez.

Se propone cambiar los nombres DNS de forma que indiquen la dirección que los paquetes toman para alcanzar su destino, de esta forma, el primer salto obtenido de la prueba anterior de traceroute seria de *tijuana* (Tijuana-TELMEX) hacia *gdl* (Guadalajara-TELMEX), de la misma forma para el segundo salto, de *gdl* (Guadalajara-TELMEX) hacia *mty7200* (Monterrey-TELMEX), y el tercer salto de *mty7200* (Monterrey-TELMEX) hacia *juarez* (Cd. Juárez-TELMEX). Lo que al realizar la misma prueba de traceroute de Tijuana-TELMEX hacia Cd. Juárez-TELMEX reflejaría lo siguiente.

Traceroute del nodo CUDI-TIJ-TELMEX (Tijuana, B.C) a 200.23.60.146.

Router Response:

Type escape sequence to abort.
Tracing the route to mty7200-juarez.core.cudi.edu.mx (200.23.60.146)

1 tijuan-gdl.core.cudi.edu.mx (200.23.60.242) 36 msec 40 msec 36 msec
2 gdl-mty7200.core.cudi.edu.mx (200.23.60.233) 56 msec 56 msec 56 msec
*3 mty7200-juarez.core.cudi.edu.mx (200.23.60.146) 72 msec * 68 msec*

La tabla 65 muestra la relación de los nombres DNS de las interconexiones de los equipos de backbone de la RedCUDI con sus respectivas direcciones IPv6. Los nombres sin estar en negritas, son los asignados actualmente, y los nombres, en negritas, son los nombres propuestos para mejorar la interpretación de los resultados obtenidos por herramientas de diagnóstico como traceroute.

Tabla 65. Nombres DNS de los enlaces del Backbone de la RedCUDI.

Dir. IPv6	Enlace	Nombre DNS
Nodo Tijuana (TELMEX)		
2001:0DB8:$B_0P_1TC_0$A:0F02::2/64	Tij-Gdl	tijuana-gdl.core.cudi.edu.mx
		gdl-tijuana.core.cudi.edu.mx
Nodo Guadalajara (TELMEX)		
2001:0DB8:$B_0P_1TC_0$A:0F09::1/64	Gdl.-D.F.	gdl-mexico7200.core.cudi.edu.mx
		mexico7200-gdl.core.cudi.edu.mx
2001:0DB8:$B_0P_1TC_0$A:0F13::1/64	Gdl-Mty	gdl-mty7200.core.cudi.edu.mx
		mty7200-gdl.core.cudi.edu.mx
2001:0DB8:$B_0P_1TC_0$A:0F02::1/64	Gdl-Tij	gdl-tijuana.core.cudi.edu.mx
		tijuana-gdl.core.cudi.edu.mx
Nodo Monterrey (TELMEX)		
2001:0DB8:$B_0P_1TC_0$A:0F13::2/64	Mty-Gdl	mty7200-gdl.core.cudi.edu.mx
		gdl-mty7200.core.cudi.edu.mx
2001:0DB8:$B_0P_1TC_0$A:0813::1/64	Mty-Cd.J	mty-juarez.core.cudi.edu.mx
		juarez-mty.core.cudi.edu.mx
No existe*	Mty Tel-Mty Ava	mty7200-mty10k.core.cudi.edu.mx
		mty10k-mty7200.core.cudi.edu.mx
Nodo D.F. (TELMEX)		
2001:0DB8:$B_0P_1TC_0$A:0F09::2/64	D.F.-Gdl	mexico7200-gdl.core.cudi.edu.mx
		gdl -mexico7200.core.cudi.edu.mx
No existe*	D.F. Tel-D.F. Ava	mexico7200-mexico10k.core.cudi.edu.mx
		mexico10k-mexico7200.core.cudi.edu.mx
Nodo Cd. Juárez (TELMEX)		
2001:0DB8:$B_0P_1TC_0$A:0813::2/64	Cd.J-Mty	juarez-mty7200.core.cudi.edu.mx
		mty7200-juarez.core.cudi.edu.mx
Nodo D.F. (AVANTEL)		
2001:0DB8:$B_0P_2TC_0$A:0913::1/64	D.F.-Mty	mexico10k-monterrey.core.cudi.edu.mx
		monterrey-mexico10k.core.cudi.edu.mx
2001:0DB8:$B_0P_2TC_0$A:0917::1/64	D.F.-Cancún	mexico10k-cancun.core.cudi.edu.mx
		cancun-mexico10k.core.cudi.edu.mx
No existe*	D.F. Ava-D.F. Tel	mexico10k-mexico7200.core.cudi.edu.mx
		mexico7200-mexico10k.core.cudi.edu.mx

2001:0DB8:$B_0P_2TC_0A$:0913::2/64	Mty-D.F	mty10k-mexico.core.cudi.edu.mx
		mexico-mty10k.core.cudi.edu.mx
No existe*	Mty Ava-Mty Tel	mty10k-mty7200.core.cudi.edu.mx
		mty7200-mty10k.core.cudi.edu.mx
2001:0DB8:$B_0P_2TC_0A$:0917::2/64	Cancún-D.F.	cancun-mexico10k.core.cudi.edu.mx
		mexico10k -cancun.core.cudi.edu.mx

* Conexiones realizadas por equipos de Avantel, y actualmente no tienen soporte IPv6.

Capítulo

7.Servicios y aplicaciones IPv6 en RedCUDI

Resumen

Los servicios y aplicaciones son parte fundamental de las facilidades y herramientas ofrecidas a los usuarios de la red para el manejo de la misma, al igual que Internet comercial, Internet2 (CUDI) maneja sus propias soluciones, las cuales van más orientadas a servicios en tiempo real (videoconferencias, trabajo en grupo a distancia, etc.). En este capítulo se hablará de forma general de los servicios y aplicaciones que cuentan con soporte IPv6 utilizadas en el Internet comercial, debido a que algunas de estas herramientas pueden ser retomadas dentro de Internet2 para llevar a cabo las tareas de los investigadores, y de algunos otros servicios y aplicaciones que corren sobre Internet2, ya que el soporte IPv6 en estos, básicamente depende de los desarrolladores de los proyectos.

7.1. ANTECEDENTES

Desde diciembre del año 2001, está funcionando el Backbone de la red de Internet2 de nuestro país (RedCUDI) con conexiones IPv6 nativas sobre ATM. Con este hecho, se tuvo lista esta red para proporcionar servicios de producción con la nueva versión de IP, y se avanzó en pruebas e implantación de nuevas tecnologías que permitirán llevar a cabo los objetivos planeados para Internet2.

A la par que se llevaban a cabo los trabajos para la instalación de IPv6 en el backbone de RedCUDI, y hasta la fecha actual, se han estado realizando pruebas de IPv6 dentro del grupo de trabajo IPv6 de la UNAM y de CUDI, con los diferentes equipos, sistemas operativos y aplicaciones que tienen disponibles. Dentro de tales pruebas destacan las siguientes[35]:

- Pruebas de Stacks IPv4/IPv6 (Win NT 4, Win 2000, Win XP, Vista, Solaris 2.5, 7 y 8, BSD, Nortel Networks, Cisco Systems, 3Com, Huawei, Foundry, Quagga, etc).
- Protocolos de enrutamiento (Rutas estáticas, Ripng, OSPFv3, ISIS y BGP4+)
- Túneles (Manuales, y Automáticos, 6to4, ISATAP, Teredo, etc.)
- Aplicaciones de videoconferencia con soporte IPv6.
- Multicast con IPv6
- Aplicaciones básicas (ping, netstat, route, traceroute, telnet, etc.)
- Herramientas de seguridad e IPsecv6 (tcpwrappers, openssh, ssh, etc)
- Con diferentes aplicaciones con soporte para IPv6 (www, FTP, Correo, DNS, etc).
- De desempeño con IPv4 e IPv6 en ruteadores y switches.
- Software traductor IPv4/IPv6 para Windows (Toolnet6 y MSR).
- Análisis de tráfico en IPv6 (Ethereal, WireShark, tcpdump, etc) .
- IPv6 sobre ATM.
- QoS.
- NAT.
- Computación móvil

Asimismo, como parte de los objetivos del grupo de trabajo IPv6 de CUDI y en colaboración con el comité de aplicaciones, y algunos otros grupos involucrados de CUDI, se llevó acabo el desarrollo de algunos proyectos dentro de los cuales destacan:

- Desarrollo y programación de aplicaciones para IPv6 (programación de sockets).
- VoIPv6.
- Colaboratorios
- Control Remoto de Telescopios.
- Control Remoto “Microscopio Electrónico”
- GRIDS computacionales (GRAMA, GRid Académica MexicanA).
- Realidad Virtual Compartida.
- Opera Oberta con Multicast IPv6.

[35] www.ipv6.unam.mx/documentos/IPv6_UNAM.pdf,
http://www.cudi.edu.mx/primavera2002/presentaciones/Red-IPv6_de_CUDI.pdf

Como se mencionó, los trabajos en materia de IPv6 han seguido realizándose hasta la actualidad, dentro de los proyectos próximos y pruebas a llevar a cabo, destacan los siguientes:

- Instalación de servicios de túneles IPv6/IPv4, por medio de Teredo, Tunnel Broker y 6to4 Relay para ofrecer conectividad IPv6 a los usuarios de CUDI, que aún no tenga soporte IPv6 en sus equipos.
- Realizar pruebas con soporte Multicast IPv6 como parte del proyecto OSTN de CUDI.
- Realizar pruebas de videoconferencias por IPv6 mediante el uso de ISABEL.
- Instalación de herramientas de administración para IPv6.

7.2. SERVICIOS Y APLICACIONES IPv6

Para hacer uso de la red existente IPv6 en RedCUDI, no sólo basta con tener habilitado IPv6 en los equipos de la red, sino también beneficiarse de aplicaciones que lo soporten. Las aplicaciones y servicios del Internet comercial, al igual que las redes, están sufriendo de un período de transición, mientras dura la migración de las redes IP de la versión 4 a la 6, y en consecuencia, en este espacio de tiempo, surgirán escenarios donde aplicaciones IPv4 e IPv6 tendrán que coexistir e incluso ínteroperar. Actualmente, el número de aplicaciones con soporte nativo IPv6 esta creciendo significativamente, y otras más, que aún no lo tienen, están siendo trasladadas por desarrolladores a nivel mundial.

De igual forma que en el Internet comercial utiliza servicios y aplicaciones para trabajar con los recursos de la red, Internet2 puede aprovechar dichos servicios y aplicaciones para que los usuarios puedan efectuar sus labores, ya que la mayoría de las investigaciones realizadas dentro de Internet2 van más orientados a servicios en tiempo real (videoconferencias, tele-inmersión, bibliotecas digitales, trabajo en grupo a distancia, etc.), la cuales, dependiendo de sus desarrolladores, pueden o no correr sobre IPv6.

Internet2 no retomará los servicios actuales de Internet para seguirlos desarrollando, pero posiblemente en algunas circunstancias será necesario utilizar algunos de ellos para llevar acabo las labores de los usuarios, o para el manejo de algunas herramientas, es por eso, que en este capítulo serán tratados de manera general.
Algunos de los servicios disponibles en Internet comercial aparte de la Web son el acceso remoto a otras máquinas (SSH y telnet), transferencia de archivos (FTP), correo electrónico (SMTP), boletines electrónicos (news o grupos de noticias), conversaciones en línea (IRC y chats), mensajería instantánea, transmisión de archivos (P2P, P2M, Descarga Directa), etc.

A continuación de manera general se hablará del funcionamiento de algunos de estos servicios y aplicaciones utilizados en el Internet comercial, y así mismo, se dará una lista con los nombres de las aplicaciones y versiones respectivas que actualmente ya cuentan con soporte IPv6. En la sección siguiente, se mencionarán y describirán algunas de las aplicaciones que han sido, o están siendo utilizadas en la red de Internet2 de México (CUDI); tratando de enfocarnos principalmente a las de mayor interés en cada comunidad.

7.2.1. Modelo Cliente/Servidor

La mayoría de las aplicaciones que trabajan en entornos de red están clasificadas como aplicaciones cliente/servidor. Dichas aplicaciones, como clientes FTP, navegadores y clientes de correo electrónico, disponen de dos componentes que les permite funcionar: la parte del cliente y la parte del servidor. La primera, está localizada en la computadora del usuario y es la solicitante del servicio. La parte del servidor está localizada en una computadora remota y ofrece servicios concretos en función de la solicitud del cliente.

Una aplicación cliente/servidor funciona repitiendo continuamente la misma rutina: a una solicitud del cliente le sigue una respuesta del servidor. Por ejemplo, un navegador accede a una página Web a través de un URL (Localizados universal de recursos, Uniform Resource Locutor), el cual se convierte en una dirección IP de un servidor Web remoto. Tras localizar dicho URL, el servidor identificado por el mismo, responde a la petición. A continuación, y en función de la información recibida desde ese servidor Web, el cliente puede efectuar otra petición al mismo servidor o solicitar otra página Web de un servidor diferente.

7.2.1.1.Sockets

Asociado al modelo cliente servidor, existe otro concepto importante inherente a este modelo, el cual es el responsable de que se puedan establecer conexiones lógicas entre clientes y servidores para el intercambio de información.

Para que dos programas puedan comunicarse o establecer una conexión lógica entre sí es necesario que se cumplan ciertos requisitos:

- Que un programa sea capaz de localizar al otro.
- Que ambos programas sean capaces de intercambiarse cualquier secuencia de octetos, es decir, datos relevantes a su finalidad.

Para ello son necesarios los tres recursos que originan el concepto de socket:

- Un protocolo de comunicaciones, que permite el intercambio de octetos.
- Una dirección del Protocolo de Red (Dirección IP, si se utiliza el Protocolo TCP/IP), que identifica una computadora.
- Un número de puerto, que identifica a un programa dentro de una computadora.

Por lo tanto, un socket es un fichero existente en la máquina cliente y en la máquina servidora, que sirve en última instancia para que el programa servidor y el cliente lean y escriban la información. Esta información será la transmitida por las diferentes capas de red.

7.2.2. Acceso Remoto

En ocasiones se necesita trabajar con una máquina ubicada en un lugar remoto sin conocer concretamente donde está. Simplemente se sabe que existe esa máquina, que permite hacer ciertas cosas consideradas de interés y se requiere utilizar.

Para poder trabajar con esa máquina, se necesita establecer una conexión remota entre la PC local y la PC remota. Ésta te permitirá utilizar todos los recursos de esa máquina, sin que ella tenga que estar en el mismo lugar donde se encuentra la PC local. Este tipo de conexión es totalmente transparente, de hecho es como si se estuviera conectado directamente al ordenador en cuestión.

Las aplicaciones de acceso remoto, permiten acceder a un servidor remoto, emulando un terminal que se encuentra físicamente corriendo en el PC local. Una vez que la conexión queda establecida, las aplicaciones de acceso remoto actúan de intermediario entre el ordenador local y el equipo remoto, es decir, todo lo que se escribe en el teclado local pasa directamente al otro ordenador, de la misma forma, todo lo que el otro ordenador intenta mostrar por pantalla pasa directamente al monitor local.

A continuación se mencionan algunas de las aplicaciones utilizadas actualmente, para realizar conexiones remotas entre dos ordenadores ubicados físicamente en lugares diferentes.

7.2.2.1.Telnet

Telnet es un software de emulación de terminal que ofrece acceso remoto a otras computadoras. Permite acceder a un host de Internet y ejecutar comandos. Telnet es utilizado con frecuencia para la administración remota de servidores y de equipos de red como routers y Switches. El cliente Telnet suele denominarse host local mientras que el servidor Telnet, usa un software especial llamado daemon, recibe el nombre de host remoto; en la figura 142 se muestra cómo se establece una conexión telnet.

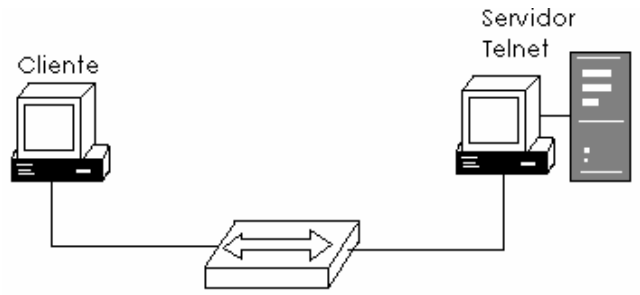


Figura 142. Conexión Telnet

Para efectuar una conexión desde un cliente Telnet, se debe seleccionar una opción de conexión. Es necesario introducir en la línea de comandos el nombre del host y el tipo de terminal. El nombre del host es la dirección IP (DNS) de la computadora remota con la cual se desea conectar. El tipo de terminal describe la emulación de terminal que quiere que su computadora local lleve a cabo. La operación Telnet no utiliza ninguno de los recursos de

la máquina que transmite. En su lugar, transmite las pulsaciones de teclas a la computadora remota y devuelve la pantalla resultante de la petición al monitor local. Todas las tareas de procesamiento y almacenamiento tienen lugar en la computadora remota.

Cuando se introduce una DNS para una localización Telnet, el nombre de la misma debe traducirse a su dirección IP asociada antes de que la conexión pueda establecerse. La aplicación Telnet trabaja principalmente con las tres capas superiores del modelo OSI: la capa de aplicación (comandos), la capa de presentación (formatos, normalmente ASCII) y la capa de sesión (transmisión). Los datos son entonces pasados a la capa de transporte, donde son segmentados y en donde se añade el número de puerto y la comprobación de errores. A continuación, la capa de red recibe los datos y en ella se añade la cabecera IP (que contiene la dirección IP origen y destino). Después, el paquete viaja a la capa de enlace de dato, donde se encapsula en una trama de datos, se añaden las direcciones MAC origen y destino y una trama de información final. Si la computadora origen no dispone de la dirección MAC de la computadora destino, se efectúa una solicitud ARP para IPv4, y Neighbor Discovery en el caso de IPv6. Una vez determinada esta dirección, la trama viaja a través del medio físico (en formato binario) hasta el siguiente dispositivo. Telnet es una buena herramienta para la resolución de problemas de red porque comprueba las siete capas del modelo OSI y permite la realización de diagnósticos remotos.

Cuando los datos alcanzan el host destino, la capa de enlace de datos, la capa de red y la capa de transporte reensamblan los comandos de datos originales. El host remoto ejecuta dichos comandos y devuelve el resultado a la computadora local del cliente, usando el mismo proceso de encapsulación que entregaron los comandos originales. Todo este proceso se repite continuamente, enviando comandos y recibiendo resultados, hasta que el trabajo del cliente se haya completado, momento en el cual éste finaliza la sesión.

La llegada de la navegación gráfica ha relegado a Telnet a usos más restringidos, normalmente relacionados con la administración remota de equipos, o consulta de recursos muy especializados (bibliotecas, BBS, Archie, bases de datos).

7.2.2.2.ssh

SSH (Secure SHell) es el nombre de un protocolo y del programa que lo implementa, y sirve para acceder a máquinas remotas a través de una red. Permite manejar por completo el ordenador mediante un intérprete de comandos, y también puede redirigir el tráfico de X para poder ejecutar programas gráficos si se tiene un Servidor X arrancado.

Además de la conexión a otras máquinas, SSH nos permite copiar datos de forma segura (tanto ficheros sueltos, como simular sesiones FTP cifradas), gestionar claves RSA para no escribir claves al conectar a las máquinas y pasar los datos de cualquier otra aplicación por un canal seguro tunelizado mediante SSH.
SSH trabaja de forma similar a como se hace con telnet. La diferencia principal es que SSH usa técnicas de cifrado que hacen que la información que viaja por el medio de comunicación vaya de manera no legible y ninguna tercera persona pueda descubrir el usuario y contraseña de la conexión ni lo que se escribe durante toda la sesión.

El protocolo SSH cuenta con dos versiones. La primera de ellas se mantiene por motivos de compatibilidad, pero se recomienda generalmente el uso de la segunda, por su mayor seguridad. OpenSSH es una implementación, usable en sistemas Linux, tanto en el cliente como en el servidor, las versiones disponibles permiten usar tanto SSHv1 como SSHv2.

SSH consta de tres componentes:

- Protocolo de transporte: Que normalmente opera sobre TCP/IP dando autenticidad, confidencialidad e integridad.
- Protocolo de autenticación de usuario: Que autentica al usuario ante el servidor.
- Protocolo de conexión: Que multiplexa un canal cifrado en diversos canales lógicos.

Este protocolo requiere que los servidores tengan "llaves", las cuales son usadas por los clientes cada vez que se conectan a un servidor para verificar que no fue suplantado. Una llave es un número codificado y cifrado en un archivo. Para el cifrado de llaves, OpenSSH ofrece los algoritmos RSA y DSA (de los que se recomienda DSA para la versión 2).

A continuación se muestra la tabla 66 con el nombre de las aplicaciones de Acceso Remoto que soportan IPv6, versión y sistema operativo en la que corren.

Tabla 66. Aplicaciones de Acceso Remoto con soporte IPv6.

Aplicación	Versión	Sistema Operativo
Servidores		
OpenSSH	4.6/4.6p1	Unix, Linux, BSD, MACOS
tightvnc	1.3.9	Unix, Linux, Windows
lshd	0.9.10	Unix, Linux, BSD
PortForwarder	2.8.0	Windows
rexecd, rlogind, rshd	0.17	Unix, Linux, BSD
Clientes		
OpenSSH	4.6/4.6p1	Unix, Linux, BSD, MACOS
Putty	0.60	Unix, Linux, Windows
Tera Term Pro	2.3	Windows
tightvnc	1.3.9	Unix, Linux, Windows
lsh	0.9.10	Unix, Linux, BSD
SRP	2.1.2	Unix, Linux, BSD, Windows
rexec, rlogin, rsh	0.17	Unix, Linux, BSD

7.2.3. IRC

IRC (Internet Relay Chat) es un protocolo de comunicación en tiempo real basado en texto, que permite debates en grupo o entre dos personas y que está clasificado dentro de la mensajería instantánea. Las conversaciones se desarrollan en los llamados canales de IRC, designados por nombres que habitualmente comienzan con el carácter # o & (este último sólo es utilizado en canales locales del servidor). Es un sistema de charlas ampliamente utilizado por personas de todo el mundo.

Los usuarios del IRC utilizan una aplicación cliente para conectarse con un servidor, en el que funciona una aplicación IRCd (IRC Daemon o servidor de IRC) que gestiona los canales y las conversaciones.

IRC es un Protocolo de red que utiliza TCP así como opcionalmente SSL. Un servidor de IRC se puede conectar a otros servidores IRC para expandir la red IRC. Los usuarios acceden a las redes de IRC conectando un cliente a un servidor. Existen muchas implementaciones de clientes IRC así como de servidores. La mayoría de los servidores IRC no necesitan que los usuarios se registren, aunque de cualquier manera se necesita que los usuarios establezcan un alias antes de conectarse.

IRC es un protocolo que envía sus mensajes en texto plano, lo que significa que es posible (aunque tal vez no conviene) utilizar IRC mediante un cliente de flujo de bytes básico como netcat o telnet.

A continuación se muestra la tabla 67 con el nombre de las aplicaciones IRC que soportan IPv6, versión y sistema operativo en la que corren.

Tabla 67. Aplicaciones IRC con soporte IPv6.

Aplicación	Versión	Sistema Operativo
Servidores		
ircd	2.10.3p3	Unix, Linux, BSD, Windows
ircd-hybrid	7.2.2	Unix, Linux, BSD, Windows
Clientes		
irc	2.10.3p3	Unix, Linux, BSD, MS-Windows
bitchx	1.0c19	Unix, Linux, BSD, Windows
epic	5.0.3.4	Unix, Linux, BSD
xchat	2.8.2	Unix, Linux, BSD, Windows
irssi	0.8.11	Unix, Linux, BSD
ircii	20060725	Unix, Linux, BSD
KSirc	3.5.7	Unix, Linux, BSD
Turbo IRC	6	Unix, Linux, BSD, Windows
IRC Bots y Proxies		
eggdrop	1.6.18	Unix, Linux, BSD
ezbounce	1.04c	Unix, Linux, BSD
muh	2.1rc1	Unix, Linux, BSD
bnc	2.8.8	Unix, Linux, BSD, Windows

7.2.4. NewsGroups

Los grupos de noticias (newsgroups en inglés) son un medio de comunicación dentro del sistema Usenet (USErs NETwork, sistema de tablas de boletines más grande del mundo, y es hecho solamente de mensajes de correos electrónicos) en el cual los usuarios leen y envían mensajes textuales a distintos tablones distribuidos entre servidores con la posibilidad de enviar y contestar a los mensajes.

El sistema es técnicamente distinto, pero funciona de forma similar a los grupos de discusión de la World Wide Web. Como ésta misma, como el correo electrónico y la mensajería instantánea, los grupos de noticias funcionan a través de Internet.

Hay programas cliente para leer y escribir a grupos de noticias, generalmente integrados con un programa cliente de correo electrónico. Los mensajes suelen ser temáticos y el tráfico es enorme, por lo que solo aparecen los mensajes más recientes. Algunos grupos de noticias son moderados.

Existen 9 jerarquías principales, cada una dedicada a discusiones sobre un tipo de temas:

- comp.*: Temas relacionados con las computadoras.
- news.*: Discusión del propio Usenet.
- sci.*: Temas científicos.
- humanities.*: Discusión de humanidades (como literatura o filosofía).
- rec.*: Discusión de actividades recreativas (como juegos y aficiones).
- soc.*: Socialización y discusión de temas sociales.
- talk.*: Temas polémicos, como religión y política.
- misc.*: Miscelánea (todo lo que no entre en las restantes jerarquías).
- alt.*: Salió como alternativa a talk, pero es usada por los usuarios P2P.

A continuación se muestra la tabla 68 con el nombre de las aplicaciones NEWS que soportan IPv6, versión y sistema operativo en la que corren.

Tabla 68. Aplicaciones NEWS con soporte IPv6.

Aplicación	Versión	Sistema Operativo
Servidores		
inn	2.4.3	Unix, Linux, BSD
sn	0.3.3	Unix, Linux
leafnode	1.11.6	Unix, Linux, BSD
diablo	5.0-REL	Unix, Linux, BSD
Clientes		
tin	1.8.3	Unix, Linux, BSD
thunderbird	2.0.0.4	Unix, Linux, BSD
KNode	3.5.7	Unix, Linux, BSD
sylpheed	2.4.0	Unix, Linux, BSD, MACOS, Windows
mnews	1.22	Unix, Linux, BSD, Windows

7.2.5. ftp y tftp

FTP (Protocolo de transferencia de archivos, File Transfer Protocol) esta diseñado para descargar ficheros (recuperarlos de un servidor de Internet) o actualizarlos (dejarlos en uno de estos servidores). La capacidad de descargar o actualizar ficheros es una de las tareas más valoradas en Internet. Resulta especialmente útil para personas que necesitan acceder a computadoras remotas por distintas causas o para aquellos que deben actualizar software o drivers de forma inmediata. Los administradores de red raramente pueden esperar varios días para obtener los drivers necesarios que habiliten sus servidores de red y vuelvan ponerlos en funcionamiento. Internet puede ofrecer estos ficheros inmediatamente usando FTP. Al igual que ocurre con el correo electrónico o con Telnet, FTP es una aplicación cliente/servidor que requiere que haya software de servidor funcionando en un host al cual se puede acceder a través de aplicaciones en el cliente.

Una sesión FTP se establece del mismo modo que otra Telnet. Al igual que ocurre con este último tipo, una sesión FTP permanece activada hasta que el cliente la finalice o hasta que se produzca algún error de comunicación. Cuando se establece una conexión a un proceso FTP o daemon, es necesario suministrar una identificación de usuario y una contraseña. Habitualmente se utiliza Anonymous como identificación de usuario y su dirección de correo electrónico como contraseña. Este tipo de conexión recibe el nombre de FTP anónimo. Una vez determinada su identidad, un comando link abre la conexión entre la máquina local y el servidor FTP. Éste es similar a lo que ocurre con una sesión Telnet, en la que el cliente envía comandos que se ejecutan en el servidor y los resultados son devueltos al primero. Esta característica le permite crear y cambiar carpetas, borrar y renombrar ficheros y ejecutar muchas otras tareas asociadas con el mantenimiento de archivos.

La principal función de FTP es transferir ficheros entre los servidores y las computadoras de los usuarios y viceversa. Cuando se copian ficheros desde un servidor, FTP establece una segunda conexión (un enlace de datos entre las computadoras) a través de la cual se transfieren los datos. Dicha transferencia puede producirse en código ASCII (Código normalizado americano para el intercambio de información, American Standard Code for Information Interchange) o en modo binario. Estos modos determinan el tipo de fichero de datos que será transferido entre ambas estaciones. El formato ASCII devuelve una representación entendible para los humanos del número en siete caracteres ASCII. En el modo binario los números sólo tienen cuatro bytes (en comparación de los 7 de su equivalente ASCII), la representación binaria utiliza menos tiempo para enviar a través de la conexión serie de la computadora. Sin embargo, existen importantes ventajas en el uso de representación ASCII. Una vez terminada la transferencia del fichero, la conexión de datos finaliza automáticamente. Una vez concluida por completo la sesión de copia y movimiento de ficheros, se puede efectuar una operación de log off, lo que cierra el comando link y finaliza la sesión.

TFTP (Protocolo trivial de transferencia de archivos, Trivial File Transfer Protocol) es un servicio sin conexión que usa UDP. TFTP se utiliza en routers y Switches para transferir ficheros de configuración e imágenes del software del sistema operativo (ejemplo IOS de Cisco), y para intercambiar archivos entre sistemas que soporten TFTP. Esta diseñado de forma que resulte fácil y rápido de implementar. Por ello, no dispone de muchas de las características habituales de un FTP normal. Lo único que puede hacer es leer y escribir ficheros (o correo electrónico) desde o hacia un servidor remoto. No puede listar directorios y no tiene posibilidad de autentificar al usuario. Resulta útil en algunas LAN ya que opera más rápido que FTP en entornos estables. Otro protocolo que tiene la capacidad de descargar ficheros es http. Sin embargo, tiene la limitación de que sólo es capaz de descargar ficheros, no actualizarlos.

A continuación se muestra la tabla 69 con el nombre de las aplicaciones FTP que soportan IPv6, versión y sistema operativo en la que corren.

Tabla 69. Aplicaciones FTP con soporte IPv6[36].

Aplicación	Versión	Sistema Operativo
Servidores		
proFTPD	1.3.0a	Unix, Linux, BSD
moftpd	1.2b1	Unix, Linux, BSD
Pure-FTPd	1.0.21	Unix, Linux, BSD
oftpd	0.3.7	Unix, Linux, BSD
MSRIPv6 FTP server		Windows NT/2000
tnftpd/lukemftpd	5	Linux, BSD, MACOS
wu-ftpd	2.6.2	Linux, BSD
vsftpd	2.0.5	Unix, Linux
Libra FTP	1.3.4	Unix, Linux
tftpd	0.17	Unix, Linux, BSD
Clientes		
NcFTP	3.1.8	Windows XP/.NET
NcFTP	3.2.0	Unix, MACOS
Lftp	3.5.6	Unix, Linux, BSD, Windows
gFTP	2.0.18	Unix, Linux, BSD, Windows
FFFTP	1.92c	Windows
Windows-FTP	MSRIPv6	Windows
MacOS X-FTP	10.3	MACOS
Lukemftp/tnftp	20061217	Linux, BSD
cftp	0.11.2	Unix, Linux, BSD
Port-FTP	1.1.6	Unix, Linux, BSD, Windows
Wget	1.10.2	Unix, Linux, BSD, Windows
fget	0.4.1	Unix, Linux, BSD
Squid	2.6	Unix, Linux, BSD
tftp	0.17	Unix, Linux, BSD

7.2.6. DNS

Internet esta construido con un esquema de direccionamiento jerárquico. Esto permite el enrutamiento basado en clases de direcciones, en oposición a las direcciones individuales. El problema que esto crea al usuario es asociar la dirección correcta con el sitio de Internet adecuado. La única diferencia entre las direcciones 198.151.11.12 y 198.151.11.21 es un dígito cambiado de posición, y se complica aún más con direcciones IPv6, por ejemplo la dirección 2001:DB8:1:1:0:5EFE:A0A:A01. Es muy fácil olvidar una dirección a un sitio concreto porque no hay nada que asocie los contenidos de dicho sitio con su dirección.

Para asociar los contenidos de un sitio con su dirección, se desarrolló un sistema de denominación de dominios DNS es un sistema usado en Internet para traducir nombres de dominio en sus correspondientes direcciones IP. Un dominio es un grupo de computadoras que están agrupadas por su localización geográfica o su tipo de negocio y que están identificadas por una cadena de caracteres y/o números, normalmente un nombre de abreviatura que representa la dirección numérica del sitio Internet. En Internet existen más de 200 dominios de nivel superior, como por ejemplo:

[36] http://6net.iif.hu/ipv6_apps/, http://www.join.uni-muenster.de/Implementationen/Software.php?lang=en#FTPclients

- .us. Estados Unidos
- .uk. Reino Unido
- .es. Epaña
- .mx. México

Existen también nombres más genéricos, entre los que se incluyen:

- .edu. Sitios educativos
- .com. Sitios comerciales
- .gov. Sitios gubernamentales
- .org. Sitios no lucrativos
- .net. Servicios de red
- .mil. Sitios militares de Estados Unidos

7.2.6.1.El sistema de denominación de dominio

El servidor DNS es un dispositivo de una red que responde a una petición realizada por los clientes y que traduce un nombre de dominio en la dirección IP asociada. El sistema DNS esta configurado como una jerarquía que crea diferentes niveles de servidores DNS.

Si un servidor DNS local es capaz de traducir un nombre de dominio a su dirección IP, lo hace y devuelve el resultado al cliente. En caso de no poder hacerlo, pasa la solicitud al servidor root de nivel superior, el cual devuelve al servidor DNS local la dirección IP del servidor DNS autoritativo de nivel inferior, que de la misma forma vuelve a realizar la consulta, en caso de no conocer el recurso, se vuelve a realizar la misma operación, una y otra vez, hasta que se encuentre el servidor DNS autoritativo de la zona que pueda resolver el recurso solicitado. Si no se puede localizar el nombre de dominio en este último servidor, se considera que es un error y se devuelve el correspondiente mensaje. Cualquier aplicación que use nombres de dominio para representar direcciones IP utiliza servidores DNS para traducir dicho nombre en su correspondiente dirección IP.

Las direcciones IPv6 se representan en el Sistema de Nombres de Dominio (DNS) mediante registros AAAA (también llamados registros de quad-A, por analogía con los registros A para IPv4). El concepto de AAAA fue una de las dos propuestas, al tiempo que la arquitectura IPv6 estaba siendo diseñada. La otra propuesta utilizaba registros A6 y otras innovaciones como las etiquetas de cadena de bits (bit-string labels) y los registros DNAME.

Mientras que la idea de AAAA es una simple generalización del DNS IPv4, la idea de A6 fue una revisión y puesta a punto del DNS para ser más genérico, y de ahí su complejidad. El RFC 3363 recomienda utilizar registros AAAA hasta tanto se pruebe y estudie exhaustivamente el uso de registros A6. La RFC 3364 realiza una comparación de las ventajas y desventajas de cada tipo de registro.

A continuación se muestra la tabla 70 con el nombre de las aplicaciones DNS que soportan IPv6, versión y sistema operativo en la que corren.

Tabla 70. Aplicaciones DNS con soporte IPv6[37].

Aplicación	Versión	Sistema Operativo
Bind	9.3.2-P2	UNIX, Linux, BSD, Win 2000, XP, 2003, NT 4.
djbdns	1.05	UNIX, Linux, BSD
geta	19990419	UNIX, KAME, BSD
MaraDNS	1.2	UNIX, BSD, Linux
openldap	2.0.23	UNIX, Linux, BSD
pdnsd	1.1.7a	UNIX, Linux, BSD
Sun ONE Directory server	5.1	AIX 4.3.3, Tru64 Unix, Solaris8-9, HP-UX 11, WinNT 4 y Win2000 también soportado, pero sin IPv6.
totd	1.3	UNIX, Linux, BSD

7.2.7. http

El http (Protocolo de transferencia de hipertexto, HyperText Transfer Protocol) funciona con la World Wide Web, la parte más utilizada y de un crecimiento mayor de todo Internet. Una de las razones principales del gran crecimiento de la Web es la facilidad con la que se puede acceder a la información. Un navegador es una aplicación cliente/servidor, lo cual significa que requiere componentes en ambas partes que funcione. Este tipo de software presenta datos multimedia en páginas Web que usan texto, gráficos, sonido y video. Las páginas Web se crean con un formato propio llamado HTML (Lenguaje de marcado de hipertexto, Hypertext Markup Language). HTML dirige al navegador para generar la página Web. Además, HTML dispone de marcas para la ubicación del texto, ficheros y objetos que se transfieren desde el servidor Web al navegador.

Los hiperenlaces hacen que la navegación por la World Wide Web sea sencilla. Un hiperenlace es un objeto en un página Web (palabra, frase o imagen) que, al hacerse clic sobre él, transfiere el control a otra página Web. La página Web contiene (con frecuencia oculta dentro de su descripción HTML) una dirección conocida como URL (Localizador Universal de Recursos, Uniform Resource Locutor).

La tabal 71 muestra los componentes de una dirección URL estándar (en este caso http://www.ipv6.unam.mx/Internet2/)

Tabla 71. Componentes de una URL.

http://	**www.**	**ipv6.unam.mx**	**/Internet2/**
Indica al navegador el protocolo que debe usar.	Identifica el tipo de sitio con el que esta intentando conectar el navegador.	Representa el dominio del sitio Web.	Identifica la carpeta del servidor en la que esta localizada la pagina Web. Además, como no se ha especificado ningún nombre, el navegador carga la página por defecto indicada en el servidor.

Cuando se abre un navegador, lo primero que se ve es la página de inicio (o “home”). El URL de la página de inicio se encuentra dentro de las especificaciones de configuración del

[37] http://6net.iif.hu/ipv6_apps/

propio navegador, y puede cambiarse cuando se quiera. Desde esta página de inicio, se puede hacer clic en cualquiera de sus hiperenlaces o escribir un URL en la barra de dirección del navegador. Éste, entonces, examina el protocolo para determinar si debe abrir otro programa, y determina la dirección IP del servidor Web. Tras esto, la capa de transporte, la capa de red, la capa de enlace de datos y la capa física inician una sesión con el servidor Web. Los datos transferidos al servidor HTTP contienen el nombre de la carpeta de la localización de la página Web (estos datos también pueden contener el nombre de fichero concreto de una página HTML). Si no se indica ningún nombre, el servidor usa un nombre por defecto (el especificado en la configuración del servidor).

El servidor responde a la petición enviando todo el texto, audio, video e imágenes especificadas en las instrucciones HTML al cliente Web. El navegador reensambla todos estos ficheros para crear el aspecto de la página Web y después finaliza la sesión. Si se hace clic en otra página (ya sea del mismo servidor o de otro distinto), el proceso vuelve a repetirse.

A continuación se muestra la tabla 72 con el nombre de las aplicaciones http que soportan IPv6, versión y sistema operativo en la que corren.

Tabla 72. Aplicaciones http con soporte IPv6[38].

Aplicación	Versión	Sistema Operativo
Servidores		
Apache	2.2	UNIX, Linux, BSD, Windows
Fnord!	1.6	UNIX, Linux, BSD
thttpd	2.25b	UNIX, Linux, BSD, SunOS, Solares.
mini_httpd	1.19	UNIX, Linux, BSD, SunOS, Solares.
boa	0.94.13	Unix, Linux, BSD.
bozohttpd	5.09	NetBSD, probablemente otros BSD.
caudium	1.4	Unix, Linux, BSD, Solaris, AIX, Darwin/MacOS X.
Clientes		
Mozilla/Firefox	2.0.0.4	Unix, Linux, BSD, MACOS, Windows
Opera	9.21	Unix, Linux, BSD, MACOS, Windows
Safari	1.2	MACOS
Internet Explorer	6/7	Windows NT4, 2000,2003/WindowsXP/Vista
Konqueror	3.5.7	Unix, Linux, BSD/Standard KDE Browser
Lynx	2.8.6	Unix, Linux, BSD, Windows
W3m	0.5.2	Unix, Linux, BSD, Windows
Links	0.99	Unix, Linux, BSD
mMosaic	3.6.9	Unix, Linux, BSD
Netscape	9.0b1	Linux, MACOS, Windows
Wget	1.10.2	Unix, Linux, BSD, Windows
fget	0.4.1	Unix, Linux, BSD
Proxy y Cache		
wwwoffle	2.9	Unix, Linux, BSD, Windows
Squid	2.6	Unix, Linux, BSD
httrack	3.41-2	Unix, Linux, BSD, Windows

7.2.8. snmp

[38] http://6net.iif.hu/ipv6_apps/, http://www.ipv6.org/v6-apps.html

El SNMP (Protocolo simple de administración de redes, Simple Network Management Protocol) es un protocolo de la capa de aplicación que facilita el intercambio de información de administración entre dispositivos de red. SNMP permite a los administradores de la red gestionar los recursos de la misma, buscar y resolver problemas y planificar su crecimiento.

Una red SNMP administrada está compuesta por los siguientes tres componentes:

- Administración de dispositivos. Es un nodo de red que contienen un agente SNMP y que reside en una red administrada. Los dispositivos administrados recopilan y almacenan información y la hacen accesibles a los NMS usando SNMP. Los dispositivos administrados, a veces llamados también elementos de red, pueden ser routers y servidores de acceso, Switches y puentes, hubs, hosts o impresoras.
- Agente. Es un software de administración de red que reside en un dispositivo administrado. Un agente dispone de un conocimiento local sobre la información de administración y la traslada a un formato compatible con SNMP.
- NMS (Sistema de administración de la red, Network Management System). Ejecuta aplicaciones de monitorización y controlan dispositivos administrados. Los NMS proporcionan el grueso del procesamiento y los recursos de memoria necesarios para la administración de la red. En cualquier red administrada deben existir uno o varios NMS.

A continuación se muestra la tabla 73 con el nombre de las aplicaciones SNMP que soportan IPv6, versión y sistema operativo en la que corren.

Tabla 73. Aplicaciones SNMP con soporte IPv6.

Aplicación	Versión	Sistema Operativo
Nagios	2.9	Unix, Linux
mrtg	2.15.2	Unix, Linux, Windows
cacti	0.8.6j	Unix, Linux, Windows
NTOP	3.3	Unix, Linux, BSD, MACOS, Windows
cricket	1.0.5	Unix, Linux, BSD, Windows
Rancid	2.3.1	Unix, Linux, BSD, MACOS
Argus	3.4	Unix, Linux
ASPath-Tree	4.2	Unix, Linux, BSD
icpld	1.1.4	Unix, Linux, BSD, MACOS
CiscoView	5.4	Unix

7.2.9. smtp

Los servidores de correo se comunican entre si mediante el SMTP (Protocolo de transferencia simple de correo, Simple Mail Transfer Protocol) para enviar y recibir correo. El protocolo SMTP transporta mensajes de correo electrónico en formato ASCII usando TCP.

Cuando un servidor de correo recibe un mensaje destinado a uno de sus clientes, lo almacena y espera a que dicho cliente lo recupere. La forma de recuperar estos mensajes es variada: los clientes pueden usar programas que acceden directamente a los ficheros del

servidor de correo o utilizar uno de los muchos protocolos de red. Los protocolos de cliente de correo más populares son POP3 (Protocolo de oficina de correos versión 3, Post Office Protocol Versión 3) e IMAP4 (Protocolo de acceso a mensajes de Internet versión 4, Internet Messaging Access Protocol Versión 4), que usan TCP para transportar los datos. Aunque los clientes de correo usan estos protocolos especiales para recuperar los mensajes, el utilizado habitualmente para enviarlos es SMTP. Debido a que, habitualmente, se utilizan dos protocolos diferentes (y dos servidores distintos) para enviar y recuperar el correo, es habitual que los clientes de correo sólo puedan hacer una tarea al mismo tiempo. Por consiguiente es necesario este comportamiento de recibir y enviar los mensajes de forma separada.

Cuando se configura un cliente de correo, se debe prestar atención tanto al servidor de transmisión de correo (SMTP) como a los de recepción (POP o IMAP). SMTP no ofrece demasiadas posibilidades en temas de seguridad y no precisa de ninguna autenticación. Para prevenir que usuarios no autorizados "pirateen" mensajes de correo de sus servidores, los administradores no suelen permitir que hosts que no conforman parte de su propia red usen su servidor SMTP para enviar correo.

A continuación se muestra la tabla 74 con el nombre de las aplicaciones de correo electrónico (e-mail) que soportan IPv6, versión y sistema operativo en la que corren.

Tabla 74. Aplicaciones e-mail con soporte IPv6[39].

Aplicación	Versión	Sistema Operativo
Agentes de transferencia de Correos		
exim	4.67	Unix, Linux, BSD
zmailer	2.99.55	Unix, Linux, BSD
sendmail	8.14.1	Unix, Linux, BSD
qmail	1.05	Unix, Linux, BSD
postfix	2.4	Unix, Linux, BSD
courier	0.55.1	Unix, Linux, BSD
Inframail	2	Unix, Linux, BSD, Windows
Agente de recuperación de correos		
fetchmail	6.2.2	Unix, Linux, BSD
Agentes de usuarios		
mutt	1.5.15	Unix, Linux, BSD
sylpheed	2.4.2	Unix, Linux, BSD, MACOS, Windows
KMail	3.5.7	Unix, Linux, BSD
mozilla-mail	2.0.0.4	Unix, Linux, BSD, MACOS, Windows
thunderbird	2	Unix, Linux, BSD, MACOS, Windows
ximian-evolution	1.4.5	Unix, Linux
pine	4.64	Unix, Linux, BSD, Windows
Demonios Mailbox		
solidpop3d	0.15	Unix, Linux, BSD
courier-pop3d	0.55.1	Unix, Linux, BSD
courier-imapd	0.42.2	Unix, Linux, BSD
cyrus-imapd	2.1.3	Unix, Linux, BSD
dovecot	0.99.10.6	Unix, Linux
bincimapd	1.2.13	Unix, Linux, BSD, MACOS

[39] http://www.deepspace6.net/docs/ipv6_status_page_apps.html

Webmail		
courier-webmail	0.42.2	Unix, Linux, BSD

7.2.10. Videoconferencias

Videoconferencia es la comunicación simultánea bidireccional de audio y video, permitiendo mantener reuniones con grupos de personas situadas en lugares alejados entre sí. Adicionalmente, pueden ofrecerse facilidades telemáticas o de otro tipo como el intercambio de informaciones gráficas, imágenes fijas, transmisión de ficheros desde el pc, etc. La videoconferencia proporciona importantes beneficios como el trabajo colaborativo entre personas geográficamente distantes y una mayor integración entre grupos de trabajo.

Uno de los aspectos más importantes en una videoconferencia es el enlace de comunicación. Esto debido a que la realización de una videoconferencia demanda un ancho de banda considerable. Entre mayor sea el ancho de banda, la calidad de la videoconferencia aumenta.

Los enlaces de comunicación pueden establecerse sobre satélite, cable, fibra óptica etc., y sus velocidades de conexión pueden ir desde los 64 kbps, hasta 2 Mbps o más de acuerdo con el ancho de banda que se tenga.

Los datos se comprimen en el equipo de origen, viajan comprimidos a través del circuito de comunicación y se descomprimen en el destino. La calidad de las imágenes que se perciben está en función del nivel de compresión y de la capacidad de transmisión de datos. Los tipos de enlaces con los que trabaja videoconferencia son: Internet, Internet2 I2, ISDN y Dedicado.

7.2.10.1. Los Estándares

Permiten conexiones entre diversas marcas de fabricantes de equipos de videoconferencia, siempre y cuando cumplan con las normas internacionales propuestas por la ITU (Unión Internacional de Telecomunicaciones). A continuación se mencionan dos estándares de la ITU más usados en videoconferencias, y el estándar de la IETF utilizado par las videoconferencias y telefonía.

- **Estándar H.320:** El H.320, describe normas para la videoconferencia punto a punto y multipunto en las Redes Digitales de Servicios Integrados ISDN. Este estándar gobierna los conceptos básicos para el intercambio de audio y vídeo en el proceso de comunicación.
- **Estándar H.323:** El H.323, basado en el protocolo de Internet IP, define la forma cómo los puntos de la red transmiten y reciben llamadas, compartiendo las capacidades de transmisión de audio, vídeo y datos. Los protocolos más importantes dentro del H.323 son H.225 y H.245.
- **SIP (Protocolo de Inicio de Sesión):** Es una Arquitectura Multimedia de Internet definida por la IETF. Se puede usar SIP para Voz sobre IP, videoconferencia, mensajería instantánea así como en aplicaciones de telefonía móvil de tercera generación. Es el protocolo de señalización de la capa de aplicación que se usa para

establecer, modificar y terminar sesiones multimedia. Las aplicaciones de SIP son variadas: voz, video, juegos, mensajería, telepresencia, control de llamadas, etc.

SIP se apoya en otros estándares de comunicación entre computadoras, como el Protocolo para Descripción de Sesión (Session Description Protocol - SDP), el Protocolo de Tiempo Real (Real-Time Protocol RTP), TCP, UDP, entre otros.

7.2.10.2. Modalidades

Las modalidades que existen en videoconferencia son:

- **Punto a Punto:** Se establece una conexión en la que participan dos sitios. Su gestión se realiza mediante la negociación bilateral entre los dos sitios, marcando a una IP o a un número ISDN. Pueden llevarse a cabo los siguientes tipos de sesión:

  Un profesor hacia un alumno
  Un profesor con un grupo de alumnos
  Un grupo hacia otro grupo

- **Multipunto:** Es posible establecer una conexión en la que participen más de dos sitios, cada terminal recibe así permanentemente las imágenes de las otras salas y las visualiza simultáneamente en pantallas separadas o en una sola pantalla utilizando la técnica de división de pantalla. Se utiliza un MCU para poder realizar la conexión entre las sedes participantes.

7.2.10.3. Requerimientos

Codecs: Es una abreviatura de Codificador-Descodificador. Describe una especificación desarrollada en software, hardware o una combinación de ambos, capaz de transformar un archivo con un flujo de datos (stream) o una señal. Los códecs son los responsables de transformar las señales de video y audio a muchas calidades diferentes. La tabla 75 siguiente muestra algunos codecs utilizados en las videoconferencias.

Tabla 75. Codecs Audio y Video

Audio	Video
G.711	H.261
G.722	H.263
G.728	H.263+
G.729	H.263++
	H.264

Sistema de audio: Se compone de audio de entrada y audio de salida. Como sistema de audio, se puede lograr lo siguiente: Acústica, cancelación de eco y supresión de ruidos, adaptándose a las características acústicas de la sala.

El audio de entrada se conforma por:

- Microfonía Inalámbrica y/o Alámbrica

- Mezcladora

El audio de salida se conforma por:
- Bocinas Plafón o Base
- Amplificador
- Mezcladora

Sistema de video: El sistema de video permite observar la imagen del sitio remoto y del sitio local, como es el caso de diapositivas, gráficas, videos, por mencionar algunos.

El sistema de video se conforma por:
- Cámara robótica
- Videoproyector
- Televisor (es)
- Cámara de documentos

Iluminación: La mejor iluminación para videoconferencia es la fluorescente difusa. Es importante minimizar las sombras para transmitir una imagen clara al sitio remoto. Además el uso de luces de baja energía fluorescentes que operan entre los 30 y 50 kHz debe evitarse, ya que interfiere con el funcionamiento adecuado de controles remotos utilizados para el manejo de las salas. Idealmente la sala no debe contar con ventanas, si las tuviese deben cubrirse con cortinas o persianas.

Enlaces: Como se mencionó anteriormente, el enlace de comunicación es un factor crucial en una videoconferencia ya que entre mayor sea el ancho de banda, mejor será la calidad de la videoconferencia.

Velocidad de transmisión: La velocidad estándar de transmisión en una videoconferencia es de 384 kbps. A esta velocidad se cuenta con una calidad de video óptima para juntas y presentaciones.

A continuación se muestra la tabla 76 con el nombre de las aplicaciones de videoconfenrecias que soportan IPv6, versión y sistema operativo en la que corren.

Tabla 76. Aplicaciones de Videoconferencias con soporte IPv6.

Aplicación	Versión	Sistema Operativo
6UMS (IPv6 Unified Instant Messaging Systems)	Proyecto de varias universidades Europeas (Euro6IX). Sistema de mensajera para conexiones punto a punto son soporte IPv6	Linux, Solaris
GnomeMeeting o Ekiga	2.0.9	Unix, Linux, Windows
Isabel	4.10.r0-7	Linux
MPEG-4IP	1.5	Unix, Linux, BSD, MACOS, Windows
Vic y Rat	2.8ucl1.1.5/4.2.25	Unix, Linux, BSD, Windows
VOCAL	1.5.0	Unix, Linux, BSD, Windows
Open H.323	1.12.2	Unix, Linux, BSD, Windows
Access Grid	3.0.2	Unix, Linux, BSD, MACOS, Windows

7.2.11. VoIP

La Voz sobre IP (VoIP) es una tecnología que permite la transmisión de voz a través de las redes IP (Internet). Esto significa que se envía la señal de voz en forma digital, dentro de paquetes, en lugar de enviarla en forma de circuitos como una compañía telefónica convencional o PSTN (Red Telefónica Conmutada).

En Internet, los datos se envían en pequeños fragmentos (paquetes) que se dispersan eligiendo el camino más corto (menos saturado) y se recomponen en el destino. Éste funcionamiento, óptimo para los paquetes de datos, no fue pensado en un principio para enviar voz en tiempo real; por lo que las comunicaciones IP eran de muy mala calidad, debido a retardos y ecos.

Actualmente la tecnología en las redes ha avanzado lo suficiente para ofrecer telefonía IP a una calidad más que aceptable. Y ya son muchas las empresas que ofrecen estos servicios. El tráfico de Voz sobre IP puede circular por cualquier red IP, incluyendo aquellas conectadas a Internet, como por ejemplo redes de área local (LAN).

7.2.11.1. Funcionalidad

VoIP puede facilitar tareas que serían más difíciles de realizar usando las redes telefónicas tradicionales:

- Las llamadas telefónicas locales pueden ser automáticamente enrutadas al teléfono VoIP, sin importar en dónde esté conectado a la red local o a Internet.
- Números telefónicos gratuitos para usar con VoIP, están disponibles en Estados Unidos de América, Reino Unido y otros países de organizaciones como Usuario VoIP.
- Los agentes de Call center usando teléfonos VoIP pueden trabajar en cualquier lugar con conexión a Internet lo suficientemente rápida.
- Algunos paquetes de VoIP incluyen servicios extra, por los que la PSTN normalmente cobra un cargo extra, o que no se encuentran disponibles en algunos países, como son las llamadas de 3 a la vez, retorno de llamada, remarcación automática, o identificación de llamadas.

7.2.11.2. Movilidad

Los usuarios de VoIP pueden viajar a cualquier lugar en el mundo y seguir haciendo y recibiendo llamadas de la siguiente forma:

- Los subscriptores de los servicios de las líneas telefónicas pueden hacer y recibir llamadas locales fuera de su localidad.
- Los usuarios de Mensajería Instantánea basada en servicios de VoIP pueden también viajar a cualquier lugar del mundo y hacer y recibir llamadas telefónicas.
- Los teléfonos VoIP pueden integrarse con otros servicios disponibles en Internet, incluyendo videoconferencias, intercambio de datos y mensajes con otros servicios

en paralelo con la conversación, audio conferencias, administración de libros de direcciones e intercambio de información.

7.2.11.3. Protocolos

Los protocolos son el lenguaje que utilizarán los distintos dispositivos VoIP para su conexión. Esta parte es importante ya que de ella dependerá la eficacia y la complejidad de la comunicación. A continuación se mencionan por orden de antigüedad (de más antiguo a más nuevo) algunos protocolos usados en VoIP:

- H.323 - Protocolo definido por la ITU-T
- SIP - Protocolo definido por la IETF
- Megaco (También conocido como H.248) y MGCP - Protocolos de control
- Skinny Client Control Protocol - Protocolo propiedad de Cisco
- MiNet - Protocolo propiedad de Mitel
- CorNet-IP - Protocolo propiedad de Siemens
- IAX - Protocolo original para la comunicación entre PBXs Asterisk (obsoleto)
- Skype - Protocolo propietario peer-to-peer utilizado en la aplicación Skype
- IAX2 - Protocolo para la comunicación entre PBXs Asterisk en reemplazo de IAX
- Jingle - Protocolo abierto utilizado en tecnología Jabber

A continuación se muestra la tabla 77 con el nombre de las aplicaciones VoIP que soportan IPv6, versión y sistema operativo en la que corren.

Tabla 77. Aplicaciones VoIP con soporte IPv6.

Aplicación	Versión	Sistema Operativo
IP PBX		
6VOICE	2.0	Linux
Asterisk	1.4.4	Unix, Linux, BSD, MACOS
SER	0.9.6	Unix, Linux, BSD
VOCAL	1.5.0	Unix, Linux, BSD, Windows
OpenSER	1.2.2	Unix, Linux, BSD
SoftPhone		
Bonephone	0.8.9d-alpha	Linux
Linephone	2.0.0	Unix, Linux, BSD, Windows
Kphone	1.0.2	Unix, Linux, BSD
SJPHONE	1.60.299	Unix, Linux, MACOS

7.2.12. Multicast

La tecnología multicast representa un servicio de red en el cual un único flujo de datos, proveniente de una determinada fuente, se puede enviar simultáneamente a diversos receptores interesados. Cabe a la infraestructura de red transportar este flujo de datos, replicándolo cuando sea necesario, para todos los receptores que registren interés en recibir estos datos.

En redes TCP/IP, estos receptores son representados por una dirección de grupo o dirección multicast. Cada fuente envía paquetes hacia una dirección de grupo (multicast), en el cual estarán asociados diversos receptores. Estos receptores, a su vez se pueden vincular y desvincular en forma dinámica. Es tarea de los dispositivos de red y en particular de los enrutadores, determinar cuáles de sus interfaces poseen receptores interesados en un grupo multicast y cuáles deberán recibir una copia de los paquetes enviados para ese grupo.

El multicast está orientado hacia aplicaciones del tipo "uno para muchos" y "muchos para muchos". En estos casos, presenta claras ventajas cuando se le compara con los mecanismos de transmisión unicast y broadcast. En unicast, es necesario que la fuente replique varios flujos de datos idénticos con el objeto de transmitirlos a cada uno de los receptores, generando desperdicio de banda. Por otro lado, el sistema broadcast envía los datos a toda la red de forma indiscriminada. Esto también da como resultado el desperdicio de recursos, pues implica en transporte de datos para todas las estaciones de la red, aunque el número de receptores deseosos de que ese contenido sea reducido. Con multicast, la fuente de tránsito envía una única copia de los paquetes hacia una dirección de grupo multicast. La infraestructura de red replica estos paquetes de forma inteligente, encaminando los datos de acuerdo con la topología de receptores interesados en esa información.

Entre las diversas aplicaciones que pueden obtener ganancias con el uso de multicast están: videoconferencia, aprendizaje a distancia, distribución de software, noticias e informaciones de mercado, conciertos al vivo, actualización de bases de datos, juegos distribuidos, procesamiento competidor, simulacros distribuidos etc.

A continuación se muestra la tabla 78 con el nombre de las aplicaciones multicast que soportan IPv6, versión y sistema operativo en la que corren.

Tabla 78. Aplicaciones multicast con soporte IPv6.

Aplicación	**Versión**	**Sistema Operativo**
Servidor/Cliente		
VLC	0.8.6	Unix, Linux, BSD, Windows
Windows Media Player	11	Windows
pcm6cast	0.11	Unix, Linux
RAT	4.2.23	Unix, Linux, BSD, Windows
icecast	2.3.1	Unix, Linux, BSD, Windows

7.2.13. Seguridad

En la actualidad, las organizaciones son cada vez más dependientes de sus redes informáticas y un problema que las afecte, por mínimo que sea, puede llegar a comprometer las continuidad de las operaciones.

La seguridad informática ha adquirido gran auge, dada las condiciones cambiantes y las nuevas plataformas computacionales disponibles. La posibilidad de interconectarse a través de redes, ha abierto nuevos horizontes que permiten explorar más allá de las fronteras de la organización.

Por lo tanto, la seguridad en redes es mantener bajo protección los recursos y la información con que se encuentra en la red, a travès de procedimientos basados en una política de seguridad tales que permitan el control de lo actuado. Para que un sistema se pueda definir como seguro se debe dotar de cuatro características al mismo:

- Integridad: La información no puede ser modificada por quien no está autorizado
- Confidencialidad: La información sólo debe ser legible para los autorizados
- Disponibilidad: Debe estar disponible cuando se necesita
- Irrefutabilidad: (No-Rechazo o No Repudio) Que no se pueda negar la autoría

La seguridad en la redes informáticas por sí sola conllevas muchas cosas que se deben de considerar para mantenerlas libres de amenazas, es por tal razón que en esta sección se mencionaran sólo algunas herramientas de seguridad que soportan el protocolo IPv6, utilizadas por los administradores, para gestionar las redes informáticas.

A continuación se muestra la tabla 79 con el nombre de algunas herramientas de seguridad que soportan IPv6, versión y sistema operativo en la que corren.

Tabla 79. Herramientas de seguridad con soporte IPv6.

Aplicación	**Versión**	**Sistema Operativo**
Túneles		
IPSEC	Linux 2.6	Linux
IPSEC	4.0	FeeeBSD
IPSEC	2.7	OpenBSD
IPSEC	1.5	NetBSD
yavipin	0.9.6	Desde Linux 2.4
openvpn	2.0.9	Unix, Linux, BSD, windows
FreeS/WAN	2.06	Linux
openswan	2.4.8	Linux
strongswan	4.1.3	Linux
Auditoria		
Nmap	4.21	Unix, Linux, BSD, MACOS, Windows
halfscan6	0.2	Linux
nessus	3.1.4	Unix, Linux, BSD, MACOS, Windows
Sniffers		
tcpdump	3.9.6	Unix, Linux, BSD
libpcap	0.9.6	Unix, Linux, BSD
ethereal	0.99.0	Unix, Linux, BSD, MACOS, Windows
wireshark	0.99.5	Unix, Linux, Windows
cold	1.0.14alpha	Unix, Linux, BSD
ndpmon	1.1b	Linux
winpcap	2.2	Windows
windump	3.5.2	Windows
Wrappers		
tcpwrappers	7.6	Unix, Linux, BSD
tcpd	0.2.0	Unix, Linux, BSD
firewall		
iptables	1.3.7	Linux
ipfilter	4.1.23	Unix, Liux, BSD

7.3. APLICACIONES EN INTERNET2

Actualmente existen en CUDI dos comités que están directamente involucrados con el uso de servicios y aplicaciones utilizados en la RedCUDI: el Comité de Desarrollo de la Red (CDR) y el Comité de Aplicaciones y Asignación de Fondos.

Los grupos de trabajo End to End, Enrutamiento I2, Multicast, IPv6, Seguridad, QoS, mpls, NOC I2, VNOC, Ingeniería y Conectividad, los cuales conforman el CDR de CUDI, tienen como principal función brindar a los usuarios de CUDI una red de alta capacidad con tecnología de vanguardia, en la cual puedan desarrollarse nuevas aplicaciones avanzadas de investigación y colaboración a distancia, sin las dificultades de ancho de banda y congestión de información presentadas por el primer Internet.

En contraparte, el Comité de Aplicaciones y Asignación de Fondos, conformado por las comunidades de Astronomía, Bibliotecas Digitales, Educación, Ecología, Ciencias de la Tierra, Grids Súpercomputo, Laboratorios, Matemáticas, y Salud, son los encargados de promover el desarrollo de nuevas aplicaciones avanzadas que utilicen la red. Estas aplicaciones de tecnología avanzadas de redes de telecomunicaciones y cómputo, van orientadas principalmente a la investigación y colaboración en tiempo real entre investigadores y estudiantes; principalmente en temas concernientes a la comunidad en cuestión.

Hoy en día todavía no es posible imaginar todas las posibles aplicaciones que pueden aparecer con Internet2, pero en México existen dos aplicaciones recurrentes en todas las instituciones educativas de nivel superior[40]:

- La educación a distancia (e-learning)
- Las bibliotecas digitales

Otras aplicaciones no menos importantes se están desarrollando en las áreas de:

- Telemedicina (e-salud)
- Súper cómputo compartido (Grids)
- Sistemas de información geográfica
- Tele-inmersión
- Visualización (realidad virtual)
- Laboratorios virtuales o Colaboratorios
- Control a Distancia

De esto se puede decir que las aplicaciones de Internet2 sólo tienen como lìmite la imaginación de los que trabajan en el desarrollo de las mismas. A continuación se da una breve descripción de las aplicaciones antes mencionadas.

[40] http://internet2.dgsca.unam.mx/seminario_nov99/ponencias/cudi/Antonioli/ANTONIOLIWalther.html

7.3.1. Educación a distancia

Aprendizaje asistido por tecnologías de la información. La educación a distancia fomenta el uso intensivo de las tecnologías de la información facilitando la creación, adopción y distribución de contenidos, así como la adaptación del ritmo de aprendizaje y la disponibilidad de las herramientas de aprendizaje independientemente de límites horarios o geográficos, permitiendo al alumno intercambiar opiniones y aportes a través de las tecnologías de la información.

Las herramientas que componen esta estrategia de educación son, por un lado, iguales utilidades de almacenamiento para aprender en Internet, utilidades para la presentación de los contenidos (textos, animaciones, gráficos, vídeos) y por otro, herramientas de comunicación síncrona o asíncrona entre alumnos o entre alumnos y tutores de los cursos (correo electrónico, chat, foros, blogs, wikis). Pero, más allá de las herramientas ocupadas, la educación a distancia, como todo proceso educativo, requiere de un diseño instructivo o instruccional sólido y que tome en cuenta, además de las consideraciones pedagógicas, las ventajas y limitaciones de Internet y el comportamiento de los usuarios de la misma.

7.3.2. Bibliotecas digitales

Los proyectos de bibliotecas digitales buscan promover la adopción del medio digital en las actividades de investigación, enseñanza y aprendizaje. Las bibliotecas digitales son espacios virtuales que facilitan el acceso, el uso, la diseminación y la generación del conocimiento. En estas actividades, es fundamental la disponibilidad de una red de alto desempeño, como lo es Internet2.

El ancho de banda permitirá a materiales tales como vídeo y audio digitales continuos a la investigación. Las imágenes, el audio y el vídeo pueden, por lo menos desde un punto de vista de la salida de la información, remplazar los materiales textuales. Esto también facilitará una investigación más extensa en los problemas difíciles de ordenar, de poner en un índice, y de proporcionar el acceso intelectual a esta clase de materiales.

Esta técnica utiliza gráficos complejos de alta resolución y animación para proporcionar menores cantidades de información textual. Otras capacidades de Internet2, tales como la capacidad de proporcionar consulta en tiempo real o ayuda del experto vía comunicación audio o video como parte de un interfaz, también ofrecen oportunidades de enriquecer y de ampliar el estado actual en sistemas del acceso y de extracción de información.

Para implementar exitosamente proyectos de bibliotecas digitales se requiere la formación de equipos multidisciplinarios que involucran áreas como recuperación de información, sistemas distribuidos, diseño de interfaces, ambientes colaborativos, ciencias de la información y bibliotecología.

7.3.3. Telemedicina y Salud

Se define como telemedicina la prestación de servicios de medicina a distancia. Para su implementación se emplean usualmente tecnologías de la información y las

comunicaciones. La telemedicina puede ser tan simple como dos profesionales de la salud discutiendo un caso por teléfono hasta la utilización de avanzada tecnología en comunicaciones e informática para realizar consultas, diagnósticos y hasta cirugías a distancia y en tiempo real.

Telemedicina significa medicina practicada a distancia, incluye diagnóstico y tratamiento, como también la educación médica, es un recurso tecnológico que posibilita la optimización de los servicios de atención en salud, ahorrando tiempo y dinero y facilitando el acceso a zonas distantes para tener atención de especialistas. Otra de las utilidades que presta el uso de la transmisión de datos médicos sobre redes adecuadas, es la educación, donde los alumnos de medicina y enfermería pueden aprender semiología remotamente, apoyados por su profesor y con la presencia del paciente. Así la telemedicina presta los siguientes servicios:

- Servicios complementarios e instantáneos a la atención de un especialista (obtención de una segunda opinión).
- Diagnósticos inmediatos por parte de un médico especialista en un área determinada.
- Educación remota de alumnos de las escuelas de enfermería y medicina.
- Servicios de archivo digital de exámenes radiológicos, ecografías y otros.

Todo esto se traduce en una disminución de tiempos entre la toma de exámenes y la obtención de resultados, o entre la atención y el diagnóstico certero del especialista, el cual no debe viajar o el paciente no tiene que ir a examinarse, reduciendo costos de tiempo y dinero.

7.3.4. Laboratorios virtuales

Un laboratorio virtual es un ambiente heterogéneo, distribuido que permite a un grupo de investigadores situados alrededor del mundo para trabajar en conjunto en la solución de proyectos. Como con cualquier otro laboratorio, las herramientas y las técnicas son específicas al dominio de la investigación, pero los requisitos básicos de la infraestructura se comparten a través de disciplinas. Aunque está relacionado con algunas de las aplicaciones de la tele-inmersión, el laboratorio virtual no asume a priori la necesidad de un ambiente de tele-presencia compartido.

Los componentes de un laboratorio virtual incluyen:

- Servidores capaces de manejar simulaciones de alta escala de datos
- Bases de datos que contienen la información específica de la aplicación tal como el inicio y límite de la simulación, observaciones experimentales, requisitos del cliente, apremios de fabricación, así como recursos específicos de la aplicación (estas bases de datos son dinámicas y distribuidas, pueden también ser muy grandes).

- Instrumentos científicos que están conectados con la red. (por ejemplo, datos basados en los satélites, movimiento de tierra y censores de la calidad del aire, instrumentos astronómicos tales como los recursos de radio distribuidos, etc).
- Herramientas de colaboración, a veces incluyendo la tele-inmersión.
- Activos del software (cada laboratorio virtual se basa alrededor del software especializado para la simulación, análisis de datos, descubrimiento y reducción, y visualización).

7.3.5. Tele-inmersión o Tele-presencia

La Tele-inmersión permite a usuarios de sitios geográficamente distribuidos colaborar en tiempo real en un ambiente compartido, simulado, híbrido como si estuvieran en el mismo cuarto físico.

Es lo último en las tecnologías de multimedia:

- Exploración del ambiente 3d,
- Tecnologías descriptivas y de la visualización,
- Tecnologías de audio
- Robótica

Los requisitos considerables para el sistema de la tele-inmersión, tal como el gran ancho de banda, tiempo de espera bajo y la baja variación del tiempo de espera, le hacen uno de las aplicaciones más desafiantes.

En un ambiente tele-inmersión los ordenadores reconocen la presencia y los movimientos de individuos y los objetos físicos y virtuales, y los proyectan en ambientes realistas, geográficamente distribuidos. Esto requiere el muestreo y resíntesis del ambiente físico, así como las caras de los usuarios y sus cuerpos.

Los ambientes tele-inmersos por lo tanto facilitarán la interacción entre los usuarios, los modelos y las simulaciones originados en un ordenador. Este nuevo paradigma para la interacción del humano-ordenador esta en la categoría de las aplicaciones más avanzadas de la red y como tal, es el último desafío técnico para Internet2.

Tal sistema representa la unificación de la realidad virtual y videoconferencia. Esta combinación, es un nuevo paradigma para las comunicaciones humanas y la colaboración.

7.3.6. Súper cómputo (Grids)

La computación en grid o en malla es un nuevo paradigma de computación distribuida en el cual todos los recursos de un número indeterminado de computadoras son englobados para ser tratados como un único superordenador de manera transparente.

Es una tecnología innovadora que permite utilizar de forma coordinada todo tipo de recursos (entre ellos cómputo, almacenamiento y aplicaciones específicas) que no están

sujetos a un control centralizado. En este sentido es una nueva forma de computación distribuida, en la cual los recursos pueden ser heterogéneos (diferentes arquitecturas, supercomputadores, clusters...) y se encuentran conectados mediante redes de área extensa.

Estas computadoras englobadas no están conectadas o enlazadas firmemente, es decir no tienen porque estar en el mismo lugar geográfico, pueden estar en diferentes puntos del mundo englobados por medio de una red de alta capacidad. Universidades, laboratorios de investigación, empresas, etc., se asocian para formar grid para lo cual utilizan algún tipo de software que implemente este concepto.

Las características de esta arquitectura serían:

- Capacidad de balanceo de sistemas: no habría necesidad de calcular la capacidad de los sistemas en función de los picos de trabajo, ya que la capacidad se puede reasignar desde la granja de recursos a donde se necesite
- Alta disponibilidad. Con la nueva funcionalidad, si un servidor falla, se reasignan los servicios en los servidores restantes
- Reducción de costes: Con esta arquitectura los servicios son gestionados por "granjas de recursos". Ya no es necesario disponer de "grandes servidores" y se puede hacer uso de componentes de bajo coste

7.3.7. Visualización o Realidad virtual

La realidad virtual consiste en compartir entre lugares remotos, ambientes de realidad virtual para aprovechar las cualidades de inmersión e interacción con modelos tridimensionales y combinarlas con redes avanzadas para apoyar trabajos colaborativos.

La combinación de las técnicas de realidad virtual con los sistemas avanzados de comunicación, abre una nueva forma de trabajo con un alto impacto, sin embargo representa un alto reto para las redes dadas las cantidades de pixeles que se pueden llegar a manejar y la fluidez necesaria en la interacción.

7.3.8. Observatorios Virtuales Solares

Tienen como propósito a nivel internacional, el poner a disposición de la comunidad astronómica datos observaciónales colectados por largos periodos de tiempo que posibilitan desarrollar investigaciones de frontera y que al mismo tiempo es una excelente herramienta electrónica que tiene un fuerte impacto educativo desde nivel básico hasta superior y de posgrado.

En el caso particular de la Astronomía, además de los Observatorios, requiere de Cómputo de Alto Rendimiento para hacer simulaciones numéricas de los fenómenos que ocurren en el cosmos (desde perturbaciones solares que viajan través del medio interplanetario, hasta la formación y evolución de nuestro universo), y actualmente se enfrenta al problema de manipular enormes Bases de Datos (del orden 5TB actualmente y para el próximo ciclo

solar, aproximadamente hasta 23TB), lo que lleva a la Astronomía a ser usuaria de software, hardware y redes de comunicación muy especializada.

7.3.9. Sistemas de información geográfica

Los sistemas de información geográfica (SIG) y la Percepción remota (PR) son herramientas computacionales que permiten brindar respuestas rápidas y eficaces a problemas sociales, que además se han convertido en valiosos apoyos para los estudios del medio ambiente, permitiendo recabar, analizar y visualizar relaciones que de otra manera sería muy difícil presentar.

Por ejemplo, en materia de ecología se le puede preguntar al sistema sobre la existencia de residuos peligrosos, las especificaciones de uso de suelo, sobre las posibilidades del desarrollo urbano en determinado lugar o ¿dónde poner un campo de golf?, entre otras preguntas, por lo que las únicas limitantes son, por un lado, la imaginación, y por el otro la información que no se tenga disponible en el sitio.

7.3.10. Control a Distancia

Consiste en la Animación y simulación de sistemas mecánicos simples mediante una plataforma base con la ayuda de ambientes virtuales para el control virtual y la instrumentación de sistemas mecánicos y mecatrónicos.

El uso posible de esta tecnología es para poder controlar equipos que puedan estar en medios ambientes peligrosos y/o hostiles; para el entrenamiento de personal en situaciones en la que no se cuenta con los equipos físicamente, así como también en el área de medicina, en donde se podrían realizar operaciones a distancia, entrenamiento de estudiantes, etc.

Capítulo

8.Pruebas y Resultados

Resumen

Motivado por la necesidad de complementar el trabajo de tesis en cuestión y además para consolidar bien los conocimientos del protocolo IPv6 en la práctica, se realizaron pruebas con diferentes equipos de telecomunicaciones, facilitados por diferentes fabricantes, para evaluar y probar el soporte IPv6 y la interoperabilidad entre ellos. En este capítulo se mostraran de manera muy general; por motivos de confidencialidad demandada por los fabricantes de los equipos, algunas de las pruebas IPv6 llevadas internamente en el Laboratorio de Tecnologías Emergentes de Red (NETLab) de la Dirección General de Servicios de Cómputo Académico (DGSCA) de la UNAM, mientras durò mi estancia en el proyecto que llevó por nombre “Actualización del Soporte IPv6 en la RedCUDI”.

8.1. INTRODUCCIÓN

Como se mencionó en el capítulo 6, el grupo de trabajo IPv6 de la UNAM con el grupo de trabajo IPv6 de CUDI han estado realizando trabajos y pruebas muy estrechamente, en materia del protocolo IPv6, por lo que ambas instituciones están proveyendo un aporte cuantioso de experiencia y conocimientos tanto teóricos como prácticos a la sociedad Mexicana en cuestión de la siguiente generación del protocolo IP.

En la actualidad ambos grupos de trabajo siguen realizando numerosas pruebas con el stack de IPv6, para evaluar el desempeño y la interoperabilidad de los equipos y aplicaciones. Para poder llevar a cabo dichas pruebas, ha sido necesario invitar a participar a varios distribuidores de equipos de telecomunicaciones de renombradas marcas, para que en carácter de préstamo, faciliten algunos de los enrutadores o switches que actualmente ya tienen habilitado dentro de su sistema operativo, las facilidades y características proporcionadas por las especificaciones de IPv6.

Para que los equipos fuesen prestados para realizar las pruebas con el stack de IPv6, dentro del laboratorio NetLab de la DGSCA-UNAM, ha sido necesario firmar acuerdos de confidencialidad entre ambas partes involucradas, de forma que ninguno de los resultados obtenidos sean publicados, y además únicamente sean entregados al fabricante implicado en cuestión. Es por tal motivo que el desarrollo de este capítulo en particular, se hablará de forma muy general de las pruebas llevadas a cabo con el stack de IPv6 con los diferentes equipos de telecomunicaciones disponibles, dentro del laboratorio, durante mi estancia en el proyecto que llevó por nombre “Actualización del Soporte IPv6 en la RedCUDI”.

La forma en que se describirán las pruebas realizadas, será tomando en cuenta los siguientes puntos:

- Se describirá cada prueba.
- Se mostrará un diagrama con el escenario de la prueba a realizar, sin mencionar marcas y modelos de los enrutadores y switches utilizados.
- Se mencionarán los parámetros a configurar en los equipos involucrados para realizar la prueba.
- En algunos casos se mostrarán resultados de ping6 o traceroute6 en caso de ser necesario.
- Se mostrarà la captura de paquetes por medio de un sniffer en caso de ser necesario.
- Se dará una conclusión de lo sucedido dentro de la prueba realizada.

Dentro de las pruebas realizadas a mencionar se desarrollarán las siguientes:

1. De auto-configuración de direcciones IPv6
2. De túnel manual 6in4
3. De túnel automático ISATAP
4. De túnel automático 6to4
5. De conectividad con RIPng
6. De conectividad IPv6 con OSPFv3 y redistribución de rutas de BGP4+

7. De conectividad IPv6 con IS-IS como protocolo IGP y BGP4+ como protocolo EGP, con la implementación de un servidor Web (Apache) y DNS (BIND)

8.2. PRUEBA DE AUTO-CONFIGURACIÓN DE DIRECCIONES IPv6

La figura 143 muestra la maqueta de prueba realizada, para verificar la capacidad de auto-configuración stateless de direcciones IPv6 en una interfaz de un host (PC WinXP), conectado directamente al segmento de red, en donde se reciben mensajes del tipo Anuncio de Enrutador, generados por el Ruteador, en el cual se ha configurado las direcciones IPv6 3ffe:8070::1/64 y 4001:448::1/64 en la interfaz ethe1/1.

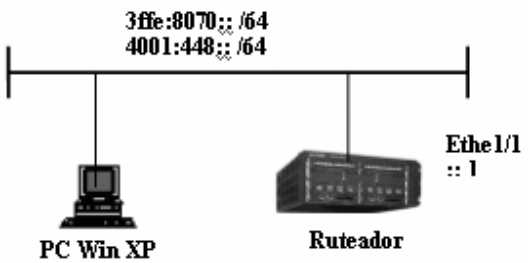


Figura 143. Maqueta de prueba de auto-configuración stateless

La tabla 80 describe los pasos de configuración llevados a cabo en los equipos de red involucrados, para realizar la prueba de auto-configuración stateless.

Tabla 80. Configuración de equipos en la prueba de auto-configuración stateless

Host	Ruteador
Habilitar el soporte IPv6 en WinXP	Habilitar IPv6 de forma global
	Configurar la direcciones IPv6 en la interfaz ethe1/1
	Por default los Anuncios de Enrutador están habilitados por cada prefijo IPv6 configurado.
	Deshabilitar el Anuncio de Enrutador de alguno de los prefijos IPv6 configurado, si es deseado.

A continuación la tabla 81, muestra los resultados obtenidos en el host WinXP, antes y después de recibir los mensajes de Anuncio de Enrutador generados por Router.

Tabla 81. Resultados de antes y después del mensaje de Anuncio de Enrutador

Antes de Anuncios de Enrutador
c:\>**ipv6 if 4** **Interfaz 4: Ethernet: Conexión de área local** GUID {6B053574-6E20-4D13-9CDC-CB13E7FE9FCE} cable desconectado usa descubrimiento de vecinos usa descubrimiento de enrutador dirección de capa de enlace: **00-0e-7b-a1-54-16** preferred link-local **fe80::20e:7bff:fea1:5416**, duración infinite multidifusión interface-local ff01::1, 1 referencias , no reportable multidifusión link-local ff02::1, 1 referencias , no reportable multidifusión link-local ff02::1:ffa1:5416, 1 referencias , último informador multidifusión link-local ff02::1:ff00:2, 1 referencias , último informador multidifusión link-local ff02::1:fff8:6ce8, 1 referencias , último informador enlace MTU 1500 (enlace MTU 1500) límite de saltos actual128 tiempo alcanzable 19500ms (base 30000ms) intervalo de retransmisión 1000ms transmisiones DAD 1 longitud de prefijo de sitio predeterminada 48
Después de Anuncio de Enrutador
C:\>**ipv6 if 4** **Interfaz 4: Ethernet: Conexión de área local** GUID {F4590D40-D8DD-4A14-A95F-EA643143C3DA} zonas: link 10 admin 5 site 2 usa descubrimiento de vecinos usa descubrimiento de enrutador dirección de capa de enlace: **00-0e-7b-16-54-16** preferred global **3ffe:8070::44c6:72d8:be87:3779**, duración 6d23h57m26s/23h54m39s (temporal) preferred global **3ffe:8070::20e:7bff:fe16:5416**, duración 29d23h57m53s/6d23h57m53s (público) preferred global **4001:448::74ab:660d:57f9:4d54**, duración 6d23h26m40s/23h23m53s (temporal) preferred global **4001:448::20e:7bff:fe16:5416**, duración 29d23h57m53s/6d23h57m53s (público) preferred link-local fe80::20e:7bff:fe16:5416, duración infinite multidifusión interface-local ff01::1, 1 referencias , no reportable multidifusión link-local ff02::1, 1 referencias , no reportable multidifusión link-local ff02::1:ff16:5416, 3 referencias , último informador multidifusión link-local ff02::1:fff9:4d54, 1 referencias , último informador multidifusión link-local ff02::1:ff87:3779, 1 referencias , último informador enlace MTU 1500 (enlace MTU 1500) límite de saltos actual64 tiempo alcanzable 27000ms (base 30000ms) intervalo de retransmisión 1000ms transmisiones DAD 1 longitud de prefijo de sitio predeterminada 48

La tabla 82 muestra los resultados de los ping6 realizados por ambos equipos, por cada prefijo configurado.

Tabla 82. Resultados de conectividad

ping6 del host WinXP a Router por el prefijo 4001:448::/64
c:\>**ping6 4001:448::1** Haciendo ping **4001:448::1** de **4001:448::74ab:660d:57f9:4d54** con 32 bytes de datos: Respuesta desde 4001:448::1: bytes=32 tiempo<1m Respuesta desde 4001:448::1: bytes=32 tiempo<1m Respuesta desde 4001:448::1: bytes=32 tiempo<1m Respuesta desde 4001:448::1: bytes=32 tiempo<1m Estadísticas de ping para 4001:448::1: Paquetes: enviados = 4, recibidos = 4, perdidos = 0 (0% perdidos), Tiempos aproximados de ida y vuelta en milisegundos: Mínimo = 0ms, Máximo = 0ms, Media = 0ms
ping ipv6 del Router al host WinXP por el prefijo 4001:448::/64

Router#**ping ipv6 4001:448::20e:7bff:fea1:5416**

Sending 1, 16-byte ICMPv6 Echo to **4001:448::20e:7bff:fea1:5416**
timeout 5000 msec, Hop Limit 64
Type Control-c to abort
Reply from 4001:448::20e:7bff:fea1:5416: bytes=16 time<1ms Hop Limit=64
Success rate is 100 percent (1/1), round-trip min/avg/max=0/0/0 ms.

ping6 del host WinXP a Router por el prefijo 3ffe:8070::/64

c:\>**ping6 3ffe:8070::1**

Haciendo ping **3ffe:8070::1**
de **3ffe:8070::44c6:72d8:be87:3779** con 32 bytes de datos:
Respuesta desde 3ffe:8070::1: bytes=32 tiempo<1m
Respuesta desde 3ffe:8070::1: bytes=32 tiempo<1m
Respuesta desde 3ffe:8070::1: bytes=32 tiempo<1m
Respuesta desde 3ffe:8070::1: bytes=32 tiempo<1m
Estadísticas de ping para 3ffe:8070::1:
Paquetes: enviados = 4, recibidos = 4, perdidos = 0 (0% perdidos),
Tiempos aproximados de ida y vuelta en milisegundos:
Mínimo = 0ms, Máximo = 0ms, Media = 0ms

ping ipv6 del Router al host WinXP por el prefijo 3ffe:8070::/64

Router#**ping ipv6 3ffe:8070::20e:7bff:fe16:5416**

Sending 1, 16-byte ICMPv6 Echo to **3ffe:8070::20e:7bff:fea1:5416**
timeout 5000 msec, Hop Limit 64
Type Control-c to abort
Reply from 3ffe:8070::20e:7bff:fea1:5416: bytes=16 time<1ms Hop Limit=64
Success rate is 100 percent (1/1), round-trip min/avg/max=0/0/0 ms.

La figura 144 muestra la captura de paquetes IPv6 con ayuda del sniffer Ethereal realizados durante la prueba. La captura detalla un paquetes del tipo Anuncio de Enrutador del prefijo 4001:448::/64 para realizar la autoconfiguración en el host WinXP, enviado por Router.

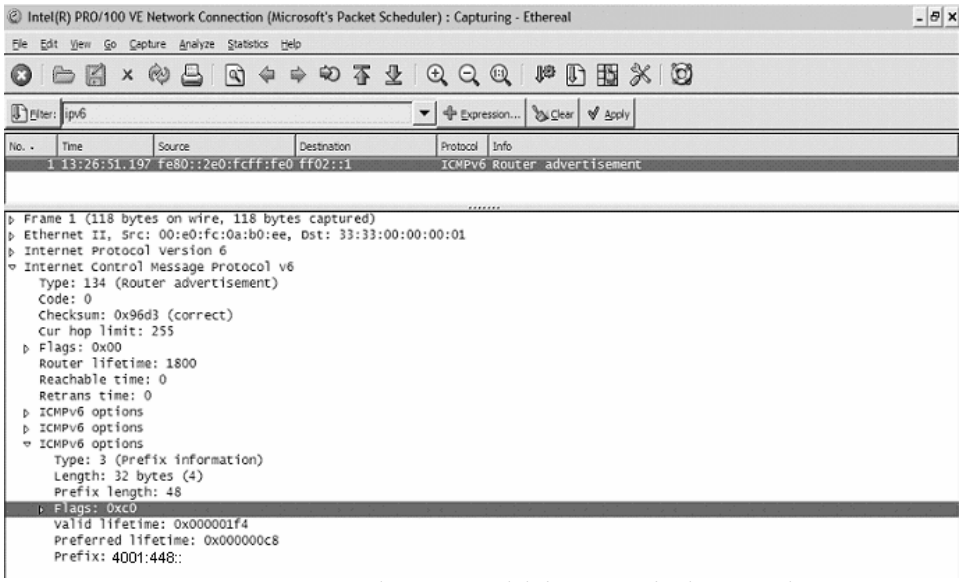


Figura 144. Captura de paquetes del tipo Anuncio de Enrutador

Como se puede observar, la prueba resultó exitosa, ya que por medio de los mensajes de Anuncio de Enrutador, se auto-configuraron en la interfaz 4 del host WinXP, las direcciones IPv6 de los prefijos preconfigurados en la interfaz ethe1/1 de Router. Por

razones de seguridad, se autoconfiguran dos direcciones IPv6 en la interfaz del host WinXP (una temporal y una pública), por cada prefijo IPv6 recibido en los mensajes de Anuncio de Enrutador. La conectividad es de forma nativa, ya que los paquetes IPv6 no están encapsulados dentro de paquetes IPv4, sino que van sobre la trama Ethernet.

8.3. TÚNEL MANUAL 6in4

La figura 145 muestra la maqueta de prueba realizada para verificar la conectividad IPv6 de túneles manuales 6in4 sobre una infraestructura de red IPv4 existente. Dicha prueba consistió en configurar y dar conectividad IPv6 por medio de un túnel manual 6in4 a un enrutador ubicado en un segmento de red IPv4 diferente, al que se ubicaba el host con plataforma FreeBSD; ambos equipos con soporte dual-stack IPv6/IPv4.

A cada uno de los equipo involucrados se levanta una interfaces de túnel 6in4, en la cual se configuran las direcciones IPv4 origen y destino de ambos lados del túnel, así como las direcciones IPv6 a utilizar en ambos lados del túnel, teniendo siempre en cuenta que estas direcciones IPv6 deben pertenecer a un mismo segmento de red, y por último una ruta estática IPv6 en la tabla de ruteo de los equipos, la cual establezca que todos los paquetes IPv6 dirigidos al segmento de red IPv6 usada, se envíen por la interfaz de túnel 6in4 antes configurada.

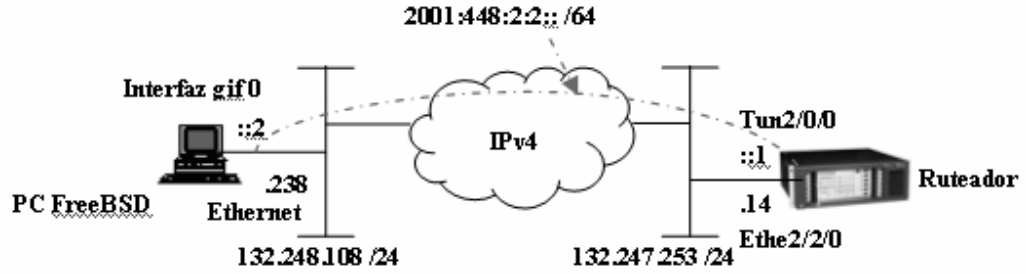


Figura 145. Maqueta de prueba de túnel manual 6in4.

La tabla 83 describe los pasos de configuración llevados a cabo en los equipos de red involucrados, para realizar la prueba de túnel manual 6in4 para proporcionar conectividad IPv6.

Tabla 83. Configuración de equipos en la prueba de túnel manual 6in4.

Host[41]	**Ruteador**
Configurar todos los parámetros necesarios para que el host tenga completa conectividad IPv4.	Configurar todos los parámetros necesarios para que el Router tenga completa conectividad IPv4.
Probar conectividad IPv4 por medio de ping al otro lado del túnel; en este caso Router, y que el resultado sea exitoso.	Probar conectividad IPv4 por medio de ping al otro lado del túnel; en este caso FreeBSD, y que el resultado sea exitoso.
Habilitar en caso de ser necesario, el soporte IPv6 en el Kernel	Habilitar IPv6 de forma global
Con ayuda de un comando de FreeBSD, crear la interfaz de túnel genérico, con	Con ayuda de un comando del enrutador, crear una interfaz virtual de túnel, la cual coincida con el índice de

[41] http://www.freebsd.org/doc/es_ES.ISO8859-1/books/handbook/network-ipv6.html

nombre gif0	la interfaz física del enrutador, por donde se establecerá el túnel manual 6in4.
Con ayuda de un comando de FreeBSD, configurar las direcciones IPv4 origen y destino del túnel manual 6in4, en la interfaz gif0	Con ayuda de un comando del enrutador, configurar en la interfaz virtual del túnel creada, el tipo de túnel a utilizar, en este caso 6in4
Con ayuda de un comando de FreeBSD, configurar en la interfaz gif0, la dirección IPv6 asignada a este lado del túnel manual 6in4.	Con ayuda de un comando del enrutador, configurar la interfaz física origen, por donde se establece el túnel manual 6in4, en la interfaz virtual del túnel creada
Con ayuda de un comando de FreeBSD, configurar una ruta por default estática a la interfaz gif0, para que todo el tráfico IPv6 sea enviado por dicha interfaz.	Con ayuda de un comando del enrutador, configurar la dirección IPv4 destino, del lado opuesto del túnel 6in4, en la interfaz virtual del túnel creada
	Con ayuda de un comando del enrutador, configurar en la interfaz virtual del túnel creada, la dirección IPv6 asignada a este lado del túnel manual 6in4.
	Con ayuda de un comando del enrutador, configurar una ruta estática a la interfaz virtual de túnel creada, para que todo el trafico IPv6 sea enviado por dicha interfaz.

A continuación la tabla 84, muestra los resultados obtenidos de la configuración del túnel manual 6in4 en ambos equipos involucrados en la prueba, es decir en cada punto final del túnel creado.

Tabla 84. Resultado de la configuración del túnel manual 6in4, en ambos lados del túnel.

Ruteador

```
#
 ipv6
#
interface Ethernet2/2/0
 ip address 132.247.253.14 255.255.255.0
 #
interface Tunnel2/0/0
 ipv6 address 2001:448:2:2::1/64
 tunnel-protocol ipv6-ipv4
 source Ethernet2/2/0
 destination 132.248.108.238
#
#
 ip route-static 0.0.0.0 0.0.0.0 132.247.253.254
#
#
 ipv6 route-static : :  0  tun 2 / 0 / 0
#
```

FreeBSD

```
[FreeBSD]$ ifconfig gif0
gif0:
flags=8051<UP,POINTOPOINT,RUNNING,MULTICAST> mtu 1280
        tunnel inet 132.248.108.238 --> 132.247.253.14
        inet6 fe80::213:20ff:fe61:5da4%gif0 prefixlen 64 scopeid 0x4
        inet6 2001:448:2:2::2 --> 2001:448:2:2::1 prefixlen 128

[FreeBSD]$ netstat -rn
Routing tables
Internet:
Destination        Gateway            Flags      Refs      Use      Netif  Expire
default            132.248.108.254    UGS        0         586      myk0
132.248.108.224/27 link#1             UC         0         0        myk0
132.248.108.254    00:d0:58:f3:6d:41  UHLW       2         0        myk0   1199
```

Internet6:				
Destination	Gateway	Flags	Netif	Expire
::/96	::1	UGRS	lo0	
default	fe80::2d0:58ff:fef3:6d41%myk0	UG	myk0	
::1	::1	UH	lo0	
::ffff:0.0.0.0/96	::1	UGRS	lo0	
2001:448:2:2::1	**2001:448:2:2::2**	UH	**gif0**	

La tabla 85 muestra los resultados de la herramienta traceroute, tanto IPv4 e IPv6, realizados entre ambos equipos involucrados en el túnel manual 6in4, lo cual demuestra el éxito del túnel.

Tabla 85. Resultados de conectividad

Traceroute IPv4 de Ruteador a FreeBSD
[Ruteador] trace 132.248.108.238 traceroute to 132.248.108.238(132.248.108.238) 30 hops max,40 bytes packet 1 132.247.253.254 16 ms 3 ms 3 ms 2 132.247.255.222 16 ms 3 ms 3 ms 3 132.247.251.201 16 ms 3 ms 3 ms 4 132.247.251.194 16 ms 3 ms 3 ms 5 132.247.251.6 16 ms 3 ms 3 ms 6 192.100.200.226 16 ms 3 ms 3 ms 7 132.248.108.238 16 ms 3 ms 3 ms
Traceroute IPv6 de Ruteador a FreeBSD
[Ruteador] trace ipv6 2001:448:2:2::2 traceroute to 2001:448:2:2::2 30 hops max,60 bytes packet 1 2001:448:2:2::2 16 ms 5 ms 5 ms
Traceroute IPv4 de FreeBSD a Ruteador
[FreeBSD]$ traceroute 132.247.253.14 traceroute to 132.247.253.14 (132.247.253.14), 64 hops max, 40 byte packets 1 unam-ipv6-1.ipv6.unam.mx (132.248.108.254) 1.322 ms 1.289 ms 1.438 ms 2 ve2-dgsca-dist.ge.unam.mx (192.100.200.225) 2.149 ms 0.967 ms 0.961 ms 3 132.247.251.5 (132.247.251.5) 0.952 ms 0.922 ms 0.945 ms 4 132.247.251.193 (132.247.251.193) 1.024 ms 0.948 ms 0.937 ms 5 132.247.251.202 (132.247.251.202) 1.056 ms 1.042 ms 1.061 ms 6 132.247.255.221 (132.247.255.221) 1.361 ms 1.310 ms 1.225 ms 7 132.247.253.14 (132.247.253.14) 4.799 ms 6.293 ms 5.010 ms
Traceroute IPv6 de FreeBSD a Ruteador
[FreeBSD]$ traceroute6 2001:448:2:2::1 traceroute6 to 2001:448:2:2::1 (2001:448:2:2::1) from 2001:448:2:2::2, 64 hops max, 12 byte packets 1 2001:448:2:2::1 5.836 ms 4.654 ms 3.591 ms

Al final de la prueba se consiguió la conectividad IPv6 exitosa entre el host FreeBSD y el enrutador, mediante el uso de un túnel manual estático 6in4. En esta prueba no se guardó ninguna captura de paquetes generados por las herramientas como ping y traceroute; por tal motivo no se ha podido mostrar de forma gráfica, como es que los paquetes IPv6 viajan encapsulados en paquetes IPv4, pero con la ayuda de traceroute se comprueba el éxito de la misma.

Como se puede observar en la tabla 85, el uso de la herramienta traceroute por medio de IPv4, para alcanzar la interfaz opuesta de uno de los dispositivos involucrados en el túnel, se tenía que pasar por 7 saltos, lo que para el resultado de la misma prueba con la misma herramienta pero por medio de IPv6, sólo se hacia un salto, esto debido a que el túnel 6in4 hace aparentar que solamente existe un salto entre ambos equipos.

8.4. TÚNEL AUTOMÁTICO ISATAP

La figura 146 muestra la maqueta de prueba realizada, para verificar la conectividad IPv6 de un túnel automático ISATAP sobre una infraestructura de red IPv4 existente. Dicha prueba consistió en configurar y probar la conectividad IPv6 por medio de un túnel automático ISATAP a dos dispositivos con soporte dual-stack IPv6/IPv4 ubicados en un mismo segmento de red IPv4.

El prefijo IPv6 usado para realizar la prueba ISATAP fue el 2001::/64, el cual es configurado en Ruteador, como dirección IPv6 ISATAP en la interfaz de túnel. En el hosts WinXP solamente se configura la dirección IPv4 del Ruteador y habilita la interfaz de túnel ISATAP, la cual por medio de mensajes del tipo Solicitud de Enrutador, pide al Ruteador el prefijo IPv6, para autoconfigurar la dirección IPv6 ISATAP en su interfaz, quedando habilitada de esta forma la conectividad IPv6.

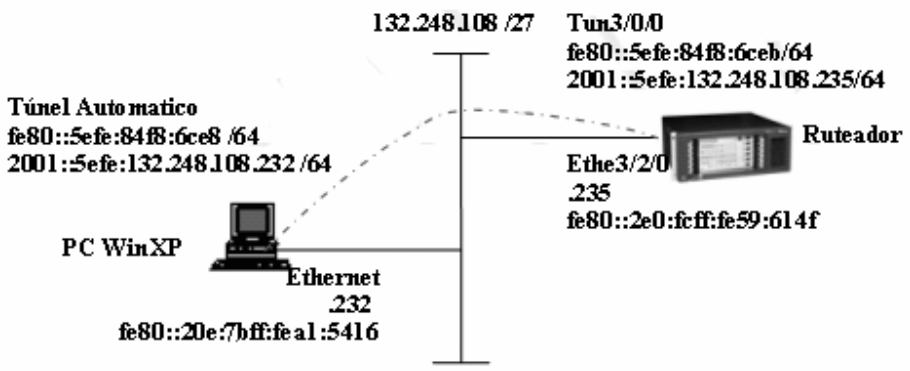


Figura 146. Maqueta de prueba de túnel automático ISATAP.

La tabla 86 describe los pasos de configuración llevados a cabo en los equipos de red involucrados, para realizar la prueba de túnel automático ISATAP para proporcionar conectividad IPv6.

Tabla 86. Configuración de equipos en la prueba de túnel automático ISATAP.

Host[42]	**Ruteador**
Configurar todos los parámetros necesarios para que el host tenga completa conectividad IPv4.	Configurar todos los parámetros necesarios para que el Router tenga completa conectividad IPv4.
Probar conectividad IPv4 por medio de ping al otro lado del túnel; en este caso Router, y que el resultado sea exitoso.	Probar conectividad IPv4 por medio de ping al otro lado del túnel; en este caso WinXP, y que el resultado sea exitoso.
Habilitar en caso de ser necesario, el soporte IPv6 en WinXP	Habilitar IPv6 de forma global
Con ayuda de un comando de WinXP, configurar la dirección IPv4 del Ruteador, en la interfaz de túnel automático ISATAP	Con ayuda de un comando del enrutador, crear una interfaz virtual de túnel, la cual coincida con el índice de la interfaz física del enrutador, por donde se establecerá el túnel automático ISATAP.
Con ayuda de un comando de WinXP, habilitar la interfaz de túnel automático	Con ayuda de un comando del enrutador, configurar en la interfaz virtual del túnel creada, el tipo de

[42] http://www.join.uni-muenster.de/Dokumente/Howtos/Howto_ISATAP.php?lang=en

ISATAP, para que se envíen los mensajes de Solicitud de Enrutador, y se autoconfigure la dirección IPv6 ISATAP.	túnel a utilizar, en este caso ISATAP
	Con ayuda de un comando del enrutador, configurar la interfaz física origen, por donde se establece el túnel automático ISATAP, en la interfaz virtual del túnel creada
	Con ayuda de un comando del enrutador, configurar en la interfaz virtual del túnel creada, la dirección IPv6 asignada a este lado del túnel automático ISATAP, con eui-64.
	Con ayuda de un comando del enrutador, configurar una ruta estática a la interfaz virtual de túnel creada, para que todo el tráfico IPv6 sea enviado por dicha interfaz.

A continuación la tabla 87, muestra los resultados obtenidos de la configuración del túnel automático ISATAP en ambos equipos involucrados en la prueba, es decir en cada extremo final del túnel creado.

Tabla 87. Resultado de la configuración del túnel automático ISATAP.

Ruteador

```
#
 ipv6
#
interface Ethernet3/2/0
 ip address 132.248.108.235 255.255.255.224
 #
#
 interface Tunnel3/0/0
 ipv6 address 2001::/64 eui-64
 tunnel-protocol ipv6-ipv4 isatap
 source Ethernet3/2/0
#
 ipv6 route-static 2001:: 16 Tunnel3/0/0
#
```

WinXP

```
Interfaz 2: Pseudo-interfaz de protocolo de túnel automático
  GUID {48FCE3FC-EC30-E50E-F1A7-71172AEEE3AE}
  no usa descubrimiento de vecinos
  no usa descubrimiento de enrutador
  preferencia de enrutamiento 1
  Dirección IPV4 incrustada EUI-64: 132.248.108.232
  dirección de capa de enlace de enrutador: 132.248.108.235
    preferred global 2001::5efe:132.248.108.232, duración infinite (manual)
    preferred link-local fe80::5efe:132.248.108.232, duración infinite
  enlace MTU 1280 (enlace MTU 65515)
  límite de saltos actual128
  tiempo alcanzable 27500ms (base 30000ms)
  intervalo de retransmisión 1000ms
  transmisiones DAD 0
  longitud de prefijo de sitio predeterminada 48

c:\>ipv6 rt
2001::/64 -> 2/fe80::5efe:132.248.108.235 pref 1if+0=1 duración infinite (dinámica)
```

La tabla 88 muestra los resultados de la herramienta ping6, realizados entre ambos equipos involucrados en el túnel automático ISATAP, lo cual demuestra la conectividad IPv6 lograda por medio del túnel automático ISATAP.

Tabla 88. Resultados de conectividad del túnel automático ISATAP.

ping6 del host WinXP a Router
c:\>**ping6 2001::5EFE:84F8:6CEB** Haciendo ping **2001::5EFE:84F8:6CEB** de **2001::5efe:84f8:6ce8** con 32 bytes de datos: Respuesta desde 2001::5EFE:84F8:6CEB: bytes=32 tiempo<1m Respuesta desde 2001::5EFE:84F8:6CEB: bytes=32 tiempo<1m Respuesta desde 2001::5EFE:84F8:6CEB: bytes=32 tiempo<1m Respuesta desde 2001::5EFE:84F8:6CEB: bytes=32 tiempo<1m Estadísticas de ping para 2001::5EFE:84F8:6CEB: Paquetes: enviados = 4, recibidos = 4, perdidos = 0 (0% perdidos), Tiempos aproximados de ida y vuelta en milisegundos: Mínimo = 0ms, Máximo = 0ms, Media = 0ms
ping ipv6 del Router al host WinXP
Router#**ping ipv6 2001::5EFE:84F8:6CE8** **----------** Sending 1, 16-byte ICMPv6 Echo to **2001::5EFE:84F8:6CE8** timeout 5000 msec, Hop Limit 64 Type Control-c to abort Reply from 2001::5EFE:84F8:6CE8: bytes=16 time<1ms Hop Limit=64 Success rate is 100 percent (1/1), round-trip min/avg/max=0/0/0 ms.

Al final de esta prueba se consiguió la conectividad IPv6 exitosa entre el host WinXP y el enrutador, mediante el uso de un túnel automático ISATAP. En esta prueba no se guardó ninguna captura de paquetes IPv6 encapsulados en IPv4, por tal motivo no se ha podido mostrar de forma gráfica el encapsulamiento, pero con la ayuda de ping6 se demuestra la conectividad IPv6 que se ha logrado entre ambos equipos por medio del túnel automático ISATAP.

Al habilitar la interfaz de túnel automático ISATAP e indicar al host WinXP la direccion IPv4 del enrutador ISATAP, automáticamente se envia un paquete de Solicitud de Enrutador encapsulado en IPv4 a la inetrfaz de túnel ISATAP del enrutador. El enrutador al recibir la solicitud, anuncia automáticamente el prefijo IPv6 ISATAP antes configurado en su interfaz de túnel ISATAP, lo cual provoca la autoconfiuguración de la interfaz de túnel ISATAP en el Host WinXP, y crea una ruta estática para el prefijo ISATAP por la interfaz de túnel automático ISATAP.

8.5. TÚNEL AUTOMÁTICO 6to4

La figura 147 muestra la maqueta de prueba realizada, para verificar la conectividad IPv6 de un túnel automático 6to4 sobre una infraestructura de red IPv4 existente. Dicha prueba consistió en configurar y probar la conectividad IPv6 por medio de un túnel automático 6to4, a dos PC con sistemas operativos WinXp y Linux/Debian, ambas con soporte dual-stack IPv6/IPv4 ubicadas en un mismo segmento de red IPv4.

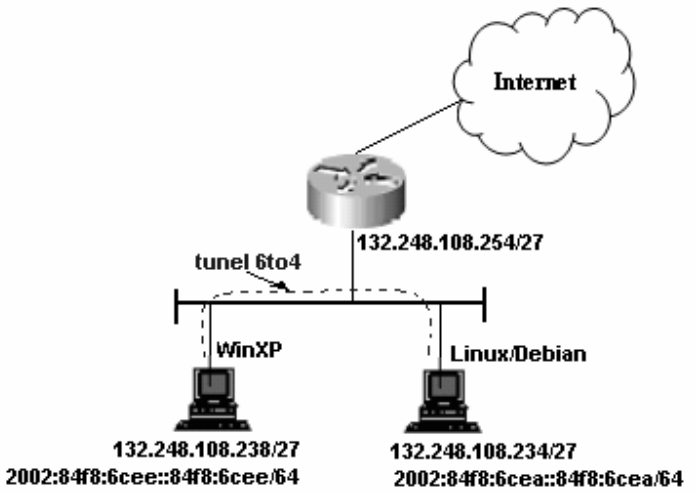


Figura 147. Maqueta de prueba de túnel automático 6to4.

La tabla 89 describe los pasos de configuración llevados a cabo en los equipos de red involucrados, para realizar la prueba de túnel automático 6to4 para proporcionar conectividad IPv6.

Tabla 89. Configuración de equipos en la prueba de túnel automático 6to4.

Host WinXP	**Host Linux/Debian**
Configurar todos los parámetros necesarios para que el host tenga completa conectividad IPv4.	Configurar todos los parámetros necesarios para que el host tenga completa conectividad IPv4.
Probar conectividad IPv4 por medio de ping al otro lado del túnel; en este caso Linux/Debian, y que el resultado sea exitoso.	Probar conectividad IPv4 por medio de ping al otro lado del túnel; en este caso WinXP, y que el resultado sea exitoso.
Habilitar en caso de ser necesario, el soporte IPv6 en WinXP	Habilitar en caso de ser necesario, el soporte IPv6 en Linux/Debian
Poner a la interfaz de 6to4 en estado de enable, lo que generar automáticamente la dirección 6to4 en la interfaz	Se levanta la interfaz sit0 de túneles IPv6-in-IPv4, para poder configurarle la dirección 6to4
Configurar un ruta estática a la red 2000::/3 por la interfaz de túnel automático 6to4	Se configura la dirección IPv6 6to4 en la interfaz sit0
	Configurar un ruta estática a la red 2000::/3 por la interfaz de túnel automático 6to4, sit0

A continuación la tabla 90, muestra los resultados obtenidos de la configuración del túnel automático 6to4 en ambos equipos involucrados en la prueba, es decir en cada extremo final del túnel creado.

Tabla 90. Resultado de la configuración del túnel automático 6to4.

Host WinXP
C:\>**ipv6 if 3** **Interfaz 3**: Pseudo-interfaz de protocolo de túnel **6to4** GUID {A995346E-9F3E-2EDB-47D1-9CC7BA01CD73} no usa descubrimiento de vecinos no usa descubrimiento de enrutador preferencia de enrutamiento 1 preferred global **2002:84f8:6cee::84f8:6cee**, duración infinite enlace MTU 1280 (enlace MTU 65515) límite de saltos actual128

tiempo alcanzable 17000ms (base 30000ms)
intervalo de retransmisión 1000ms
transmisiones DAD 0
longitud de prefijo de sitio predeterminada 48

C:\>**ipv6 rt**
2000::/3 -> 3 pref 1if+0=1 duración infinite (manual)
::/0 -> 5/fe80::2d0:58ff:fef3:6d41 pref 256 duración 27m30s (configuración automática)

Host Linux/Debian

Linux/Debian# **ifconfig sit0**
sit0 Link encap:**IPv6-in-IPv4**
inet6 addr: **2002:84f8:6cea::84f8:6cea/64** Scope:Global
inet6 addr: ::132.248.108.234/96 Scope:Compat
inet6 addr: ::127.0.0.1/96 Scope:Unknown
UP RUNNING NOARP MTU:1480 Metric:1
RX packets:583 errors:0 dropped:0 overruns:0 frame:0
TX packets:575 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
RX bytes:53288 (52.0 KiB) TX bytes:64148 (62.6 KiB)

Linux/Debian# **netstat -A inet6 -rt**
Kernel IPv6 routing table

Destination	Next Hop	Flags	Metric	Ref	Use	Iface
::1/128	::	U	0	0	1	lo
::127.0.0.1/128	::	U	0	0	1	lo
::132.248.108.234/128	::	U	0	0	1	lo
::/96	::	U	256	0	0	sit0
2001:1218:1:6:2c0:4fff:fead:dcd2/128	::	U	0	9	1	lo
2002:84f8:6cea::84f8:6cea/128	::	U	0	608	1	**lo**
2002:84f8:6cea::/64	::	U	256	0	0	**sit0**
2000::/3	::	U	1	0	0	**sit0**
fe80::2c0:4fff:fead:dcd2/128	::	U	0	7	1	lo
fe80::/64	::	U	256	0	0	eth0
fe80::/64	::	U	256	0	0	sit0
ff02::1/128	ff02::1	UC	0	1	0	eth0
ff02::9/128	ff02::9	UC	0	1026	0	eth0
ff00::/8	::	U	256	0	0	eth0
ff00::/8	::	U	256	0	0	sit0
::/0	fe80::2d0:58ff:fef3:6d41	UGDA	1024	17	0	eth0

La tabla 91 muestra los resultados de la herramienta ping6 y traceroute6, realizados entre ambos equipos involucrados en el túnel automático 6to4, lo cual demuestra la conectividad IPv6 lograda por medio del túnel automático 6to4.

Tabla 91. Resultados de conectividad del túnel automático 6to4.

ping6 y tracert6 del host WinXP a Linux/Debian

C:\>**ping6 2002:84f8:6cea::84f8:6cea**

Haciendo ping **2002:84f8:6cea::84f8:6cea**
de **2002:84f8:6cee::84f8:6cee** con 32 bytes de datos:

Respuesta desde 2002:84f8:6cea::84f8:6cea: bytes=32 tiempo<1m
Respuesta desde 2002:84f8:6cea::84f8:6cea: bytes=32 tiempo<1m
Respuesta desde 2002:84f8:6cea::84f8:6cea: bytes=32 tiempo<1m
Respuesta desde 2002:84f8:6cea::84f8:6cea: bytes=32 tiempo<1m

Estadísticas de ping para 2002:84f8:6cea::84f8:6cea:
Paquetes: enviados = 4, recibidos = 4, perdidos = 0 (0% perdidos),
Tiempos aproximados de ida y vuelta en milisegundos:
Mínimo = 0ms, Máximo = 0ms, Media = 0ms

C:\>**tracert6 2002:84f8:6cea::84f8:6cea**

Traza a la dirección **2002:84f8:6cea::84f8:6cea**
desde **2002:84f8:6cee::84f8:6cee** sobre un máximo de 30 saltos:

1 <1 ms <1 ms <1 ms **2002:84f8:6cea::84f8:6cea** Traza completa.
Ping6 y traceroute6 de Linux/Debian a WinXP
Linux/Debian# **ping6 2002:84f8:6cee::84f8:6cee** PING 2002:84f8:6cee::84f8:6cee(2002:84f8:6cee::84f8:6cee) 56 data bytes 64 bytes from 2002:84f8:6cee::84f8:6cee: icmp_seq=1 ttl=128 time=2.98 ms 64 bytes from 2002:84f8:6cee::84f8:6cee: icmp_seq=2 ttl=128 time=0.731 ms 64 bytes from 2002:84f8:6cee::84f8:6cee: icmp_seq=3 ttl=128 time=0.742 ms 64 bytes from 2002:84f8:6cee::84f8:6cee: icmp_seq=4 ttl=128 time=0.739 ms --- **2002:84f8:6cee::84f8:6cee** ping statistics --- 4 packets transmitted, 4 received, 0% packet loss, time 3002ms rtt min/avg/max/mdev = 0.731/1.299/2.985/0.973 ms Linux/Debian# **traceroute6 2002:84f8:6cee::84f8:6cee** traceroute to **2002:84f8:6cee::84f8:6cee** (2002:84f8:6cee::84f8:6cee) from **2002:84f8:6cea::84f8:6cea**, 30 hops max, 16 byte packets 1 2002**:84f8:6cee::84f8:6cee** (2002:84f8:6cee::84f8:6cee) 0.945 ms 0.899 ms *

La figura 148 muestra la captura de paquetes IPv6 con ayuda del sniffer Ethereal. La captura detalla paquetes del tipo ICMPv6 del tipo Echo Request y Reply, del ping6 realizado de Linux/Debian a WinXP. En esta captura se muestra los datagramas de IPv6 encapsulados dentro de datagramas IPv4 logrando el túnel IPv6 sobre IPv4.

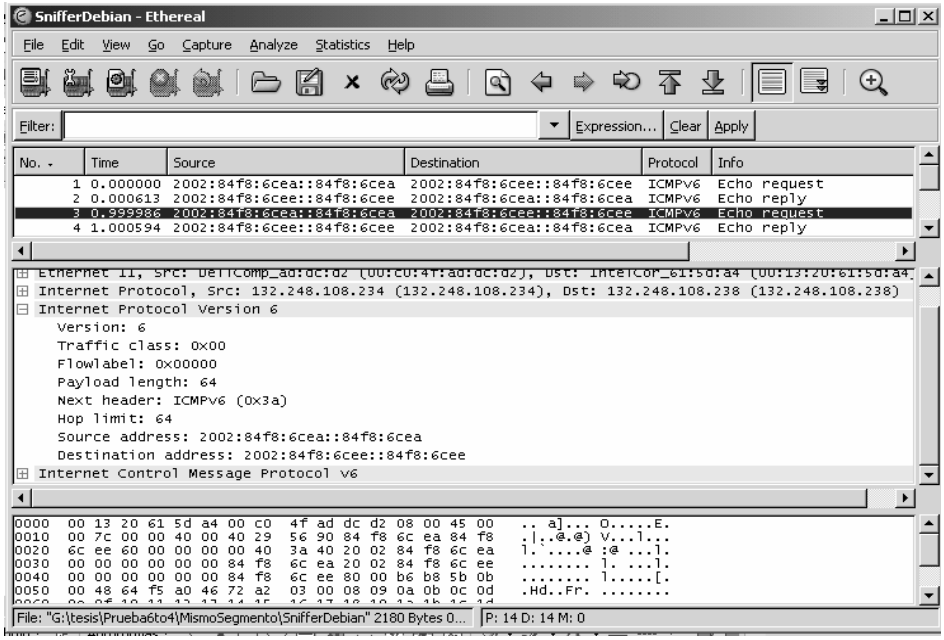


Figura 148. Captura de paquetes Ethereal

Al final de esta prueba se consiguió la conectividad IPv6 exitosa entre los host´s WinXP y Linux/Debian, mediante el uso de un túnel automático 6to4 en una infraestructura de red IPv4. La captura de paquetes obtenida por el sniffer Ethereal, demuestra que al realizar un ping6 de Linux/Debian a WinXP, los paquetes son encapsulados dentro de un datagrama IPv4, con dirección IPv4 origen del host Linux/Debian y dirección IPv4 destino del host WinXP.

El host Linux/Debian al hacer ping6 al host WinXP, inmediatamente reconoce que todo los paquetes enviados al prefijo 6to4 2002::/16, sea realizado por la interfaz de túnel manual 6to4. El paquete al ser recibido por la interfaz, conoce que dentro de la dirección IPv6 lleva consigo incrustada la dirección IPv4 real del host destino, con lo que encapsula el datagrama IPv6 dentro de un datagrama IPv4 colocando las direcciones IPv4 tanto origen y destino obtenidas de las direcciones IPv6. En este punto el proceso llevado a cabo para entregar el paquete al host destino es realizado por IPv4. En el otro extremo se recibe el paquete y se realiza de forma inversa todo el procedimiento, logrando con esto la conectividad IPv6 por medio de un túnel automático 6to4.

8.6. RIPng

La figura 149 muestra la maqueta de prueba realizada, para verificar la conectividad nativa IPv6 entre el host PC1 y PC2, ubicados en redes IPv6 diferentes. Como se puede observar en la maqueta, el hosts PC1 se encuentra conectado directamente al Ruteador-1, el cual funciona como gateway de la red 4001:448::/64, de igual forma en el otro extremo de la maqueta, el host PC2 se encuentra directamente conectado al Ruteador-2, el cual funciona como gateway de la red 3fee:8070:1::/64.

Para lograr la conectividad por IPv6 entre ambas redes es necesario el uso de un protocolo de ruteo dinámico IGP, el cual anuncie las redes en ambos enrutadores para hacerlas alcanzables. El protocolo de ruteo utilizado en la prueba es RIPng, el cual es un protocolo Vector-Distancia y utiliza como métrica los hops.

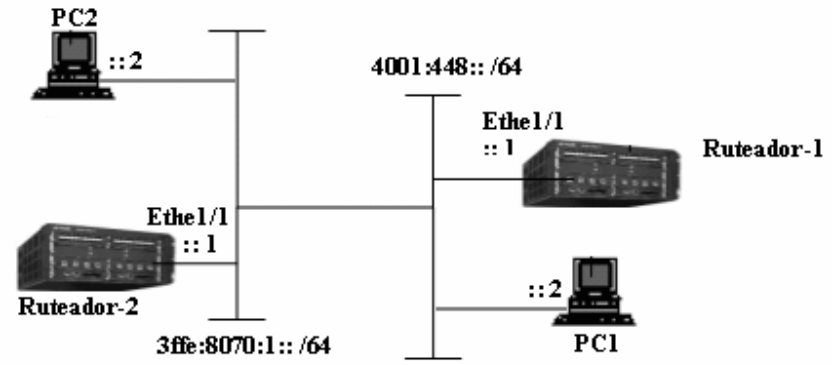


Figura 149. Maqueta de prueba del protocolo de ruteo dinámico RIPng

La tabla 92 describe los pasos de configuración llevados a cabo únicamente en el Ruteador-1 y PC1, para realizar la prueba de conectividad nativa IPv6, de dos hosts PC, conectados en redes diferentes IPv6, unidas por medio del protocolo de ruteo dinámico RIPng, debido a que la configuración en el Ruteador-2 y PC-2 son similares.

Tabla 92. Configuración de equipos para la prueba de conectividad IPv6 por RIPng

Ruteador-1	**PC1**
Habilitar IPv6 de forma global	Habilitar el soporte IPv6 en WinXP
Con ayuda de un comando del enrutador, habilitar de forma global, como protocolo de ruteo dinámico a RIP	Con ayuda de un comando de WinXP, configurar la dirección IPv6 que corresponde a la interfaz Ethernet del host

Con ayuda de un comando del enrutador, configurar la dirección IPv6 que corresponde a la interfaz Ethe1/1	Con ayuda de un comando de WinXP, configurar una ruta estática por default, para que todos los paquetes IPv6 sean enviados a la interfaz del Ruteador Ethe1/1, la cual será el Gateway del host
Con ayuda de un comando del enrutador, suprimir los mensajes del tipo Anuncio de Enrutador, en la interfaz Ethe1/1	
Con ayuda de un comando del enrutador, habilitar RIP en la interfaz Ethe1/1, por donde se intercambiaran los anuncios de RIP.	

A continuación la tabla 93 muestra solamente los resultados obtenidos en la configuración del Ruteador-1 y la PC1, en la prueba de conectividad IPv6 por medio del protocolo de ruteo dinámico RIPng, ya que la configuración de Ruteador-2 y PC-2 son similares y no hace falta mostrarla.

Tabla 93. Resultado de la configuración de los equipos para la prueba de conectividad IPv6 por RIPng.

Ruteador-1

```
!
ipv6 router rip
!
interface ethernet 1/1
ipv6 address 4001:448::1/64
 ipv6 enable
 ipv6 rip enable
 ipv6 nd suppress-ra
!
```

PC1 - WinXP

```
c:\>ipv6 if 4
Interfaz 4: Ethernet: Conexión de área local
  GUID {6B053574-6E20-4D13-9CDC-CB13E7FE9FCE}
  usa descubrimiento de vecinos
  usa descubrimiento de enrutador
  dirección de capa de enlace: 00-0e-7b-a1-54-16
    preferred global 4001:448::2, duración infinite (manual)
    preferred link-local fe80::20e:7bff:fea1:5416, duración infinite
    multidifusión interface-local ff01::1, 1 referencias , no reportable
    multidifusión link-local ff02::1, 1 referencias , no reportable
    multidifusión link-local ff02::1:ffa1:5416, 1 referencias , último informador
    multidifusión link-local ff02::1:ff00:2, 1 referencias , último informador,5 segundos para el reporte
  enlace MTU 1500 (enlace MTU 1500)
  límite de saltos actual128
  tiempo alcanzable 36500ms (base 30000ms)
  intervalo de retransmisión 1000ms
  transmisiones DAD 1
  longitud de prefijo de sitio predeterminada 48

c:\>ipv6 rt
::/0 -> 4/ fe80::20c:dbff:fef6:600 pref 1if+0=1 duración infinite (manual)
```

La tabla 94 muestra los resultados de la herramienta traceroute6, realizados entre los hosts PC1 y PC2 para verificar la conectividad IPv6 lograda entre ambos equipos ubicados en diferentes redes IPv6, demostrando con esto, que por medio de RIP, los enrutadores aprendieron las rutas de forma dinámica entre las redes IPv6 involucradas.

Tabla 94. Resultados de conectividad IPv6 obtenida por medio del protocolo de ruteo dinámico RIPng.

Traceroute6 de PC1 a PC2
c:\>tracert6 3ffe:8070:1::2 Traza a la dirección 3ffe:8070:1::2 desde 4001:448::2 sobre un máximo de 30 saltos: 1 <1 ms <1 ms <1 ms 4001:448::1 2 <1 ms <1 ms <1 ms 3ffe:8070:1::1 3 <1 ms <1 ms <1 ms 3ffe:8070:1::2 Traza completa.
Traceroute6 de PC2 a PC1
c:\>tracert6 4001:448::2 Traza a la dirección 4001:448::2 desde 3ffe:8070:1::2 sobre un máximo de 30 saltos: 1 <1 ms <1 ms <1 ms 3ffe:8070:1::1 2 <1 ms <1 ms <1 ms 4001:448::1 3 <1 ms <1 ms <1 ms 4001:448::2 Traza completa.

Al final de esta prueba se consiguió exitosamente la conectividad nativa IPv6 entre los hosts PC1 y PC2, conectados directamente a dos enrutadores diferentes, con redes IPv6 distintas. Para haber logrado la conectividad, a los dos enrutadores, se habilitó RIPng en cada una de las interfaces, por las cuales intercambiarían paquetes de enrutamiento, de forma que cada enrutador involucrado, aprendiera el trayecto o ruta para alcanzar a la red IPv6 que no se tenía directamente conectada.

Como se puede observar en la figura 149, la interfaz Ethe1/1 de los enrutadores son las que sirven de gateway al host directamente conectado y al mismo tiempo, son la interfaces que se ven directamente, por medio de la red broadcast, para intercambiar paquetes de enrutamiento. La tabla 94 muestra los resultados obtenidos después de haber realizado traceroute6, con lo que se comprueba que con ayuda de RIP, se aprendieron las redes IPv6 involucradas, de manera que para alcanzar comunicarse con el hosts opuesto, primero se tendría que pasar por los gateway´s de cada red IPv6.

8.7. OSPFv3 Y REDISTRIBUCIÓN DE RUTAS DE BGP4+

La figura 150 muestra la maqueta de prueba realizada, para verificar la conectividad IPv6 por medio de túneles IPv6, desde el host PC FreeBSD a la red IPv6 global a la que se encuentra conectada la UNAM. Como se puede observar el Ruteador-1, es el enrutador de producción IPv6 de la UNAM que se encuentra actualmente conectado por medio de túneles a la red IPv6 global. El Ruteador-2 se conecta al Ruteador-1 por medio de un túnel manual, y de la misma forma del Router-2 al host PC FreeBSD, el cual estará funcionando también como enrutador por tener habilitado el demonio de enrutamiento Quagga.

Como se sabe el protocolo EGP utilizado para lograr conocer rutas entre AS, es BGP4+, por lo tanto el Ruteador-1 tiene habilitado un proceso de BGP4+, y tiene peers con AS de la red IPv6 global. El Ruteador-2 y el host PC FreeBSD se manejaran como equipos que pertenecen un mismo AS, por lo que para que puedan aprender entre ellos rutas de manera dinámica se utilizó a OSPFv3 como protocolo IGP de enrutamiento dinámico. De modo que el AS 64516 pueda conocer las redes anunciadas en la red IPv6 global, es necesario que se establezca un peer con el AS 278. Pero al establecer peer sólo el Ruteador-2 conocerá las rutas anunciadas por el AS 278, debido a que dentro del AS 64516 sólo se tiene habilitado

un protocolo de enrutamiento IGP, y no se ha realizado iBGP4+. Para que el host FreeBSD conozca las rutas anunciadas por el Ruteador-1, se manejará redistribución de rutas de BGP4+ a OSPFv3, con esto evitando el uso de iBGP4+ y logrando la conectividad del host PC FreeBSD a la red global IPv6.

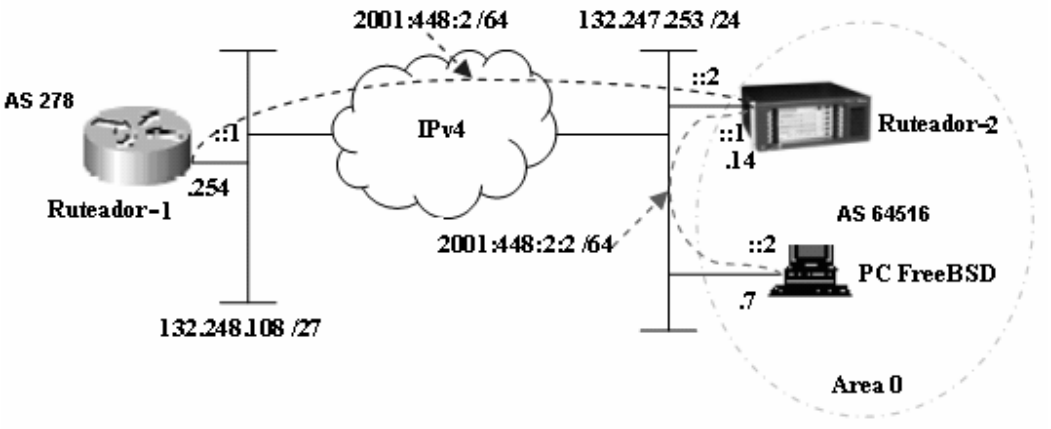


Figura 150. Maqueta de prueba OSPFv3 con redistribución de rutas de BGP4+

La tabla 95 describe los pasos de configuración llevado a cabo en el Ruteador-2 y PC FreeBSD en OSPFv3 y BGP4+, ya que la configuración realizada en el Ruteador-2 es la misma para el Ruteador-1 en la parte del túnel manual y para establecer peer por BGP4+. La configuración de los túneles manuales será omitida por haber ya sido expuestos en pruebas anteriores.

Tabla 95. Configuración de equipos para la prueba de conectividad IPv6 por medio de OSPFv3 y redistribución de rutas de BGP4+

Ruteador-2	**PC FreeBSD (Quagga)**
Habilitar IPv6 de forma global	Habilitar el soporte IPv6 en FreeBSD
Configurar la dirección IPv4 para lograr conectividad con Ruteador-1 y PC FreeBSD	Configurar la dirección IPv4 para lograr conectividad con Ruteador-1 y Ruteador-2
Configurar el túnel manual 6in4, para lograr conectividad IPv6 con el Ruteador-1	Configurar el túnel manual 6in4, para lograr conectividad IPv6 con el Ruteador-2
Configurar el túnel manual 6in4, para lograr conectividad IPv6 con PC FreeBSD	Habilitar el forwarding IPv6 en FreeBSD.
Habilitar de forma global un proceso de routing OSPFv3	Levantar el demonio de zebra en FreeBSD y conectarse a el por medio de telnet.
OSPFv3 usa un identificador de enrutador por medio de una dirección IPv4, por lo general se utiliza la dirección IPv4 de una interfaz loopback. Configurar el router-id	Levantar el demonio de ospfv3 en FreeBSD y conectarse a el por medio de telnet.
Configurar el área OSPFv3 al que pertenece el enrutador, en nuestro caso el área 0.0.0.0	Conectado al demonio de ospfv3, dentro del proceso global de OSPFv3, configurar el router-id
En la interfaz de túnel manual 6in4, habilitar el proceso de OSPFv3 indicando al área OSPFv3 al que pertenece la interfaz	Conectado al demonio de ospfv3, dentro del proceso global de OSPFv3, configurar las interfaces y àrea a las que pertenecen, para iniciar a anunciar prefijos IPv6.
Dentro del proceso OSPFv3 global, con ayuda de un comando del enrutador, importar las rutas de BGP dentro de OSPFv3	Conectado al demonio de ospfv3, dentro del proceso global de OSPFv3, configurar los prefijos IPv6 a anunciar

Habilitar de forma global un proceso de routing BGP4+	
BGP4+ usa un identificador de enrutador por medio de una dirección IPv4, por lo general se utiliza la dirección IPv4 de una interfaz loopback. Configurar el router-id	
Establecer peer con el Router-1 de forma que puedan intercambiar información de enrutamiento. En este punto, por default, los enrutadores solo intercambiaran prefijos de direcciones IPv4 unicast, los pasos siguientes habilitará el intercambio de prefijos IPv6 unicast.	
BGP4+ utiliza familia de direcciones para identificar al protocolo en el que se esta trabajando, las familias son IPv4 [unicast, multicast] e IPv6 [unicast, Multicast]. En nuestro caso ingresar a la familia IPv6 unicast.	
Habilitar dentro de la familia IPv6 unicast, al peer Router-1, para poder iniciar el intercambio de prefijos de direcciones IPv6 unicast.	
En caso de ser necesario, anunciar (inyectar) un prefijo IPv6 dentro de la familia IPv6 unicast, para que sea anunciada dentro de la base de datos BGP IPv6.	

A continuación la tabla 96 muestra los resultados obtenidos de la configuración del Ruteador-2 y la PC FreeBSD, en la prueba de conectividad IPv6 por medio del protocolo IGP de ruteo dinámico OSPFv3, y redistribución de rutas BGP4+, como protocolo EGP.

Tabla 96. Resultado de la configuración de los equipos para la prueba de OSPFv3 y redistribución de rutas BGP4+.

Ruteador-2

```
#
 ipv6
#
 interface Ethernet2/2/0
 ip address 132.247.253.14 255.255.255.0
#
 interface Tunnel2/0/0
 ipv6 address 2001:448:2::2/64
 tunnel-protocol ipv6-ipv4
 source Ethernet2/2/0
 destination 132.248.108.254
#
 interface Tunnel3/0/0
 ipv6 address 2001:448:2:2::1/64
 ospfv3 1 area 0.0.0.0
 tunnel-protocol ipv6-ipv4
 source Ethernet2/2/0
 destination 132.247.253.7
#
 bgp 64516
 router-id 132.247.253.14
 peer 2001:448:2::1 as-number 278
#
ipv6-family
network 2001:448:2:: 48
peer 2001:448:2::1 enable
#
```

```
 ospfv3 1
 router-id 132.247.253.14
 area 0.0.0.0
 import-route bgp
#
 ip route-static 0.0.0.0 0.0.0.0 132.247.253.254
#
```

PC FreeBSD (Quagga)

```
[FreeBSD]# ifconfig -a
myk0: flags=8943<UP,BROADCAST,RUNNING,PROMISC,SIMPLEX,MULTICAST> mtu 1500
        options=2b<RXCSUM,TXCSUM,VLAN_MTU,JUMBO_MTU>
        inet6 fe80::213:20ff:fe61:5da4%myk0 prefixlen 64 scopeid 0x1
        inet 132.247.253.7 netmask 0xffffff00 broadcast 132.247.253.255
        ether 00:13:20:61:5d:a4
        media: Ethernet autoselect (10baseT/UTP <half-duplex>)
        status: active
gif0: flags=8051<UP,POINTOPOINT,RUNNING,MULTICAST> mtu 1280
        tunnel inet 132.247.253.7 --> 132.247.253.14
        inet6 fe80::213:20ff:fe61:5da4%gif0 prefixlen 64 scopeid 0x4
        inet6 2001:448:2:2::2 --> 2001:448:2:2::1 prefixlen 128

zebra# show running-configure
Current configuration:
!
interface gif0
 ipv6 nd suppress-ra
!
interface myk0
 no ipv6 nd suppress-ra
!
ipv6 route ::/0 myk0
!
router-id 10.0.0.3
ipv6 forwarding
!

ospfv3# show running-configure
Current configuration:
!
interface gif0
 ipv6 ospf6 cost 1
 ipv6 ospf6 hello-interval 10
 ipv6 ospf6 dead-interval 40
 ipv6 ospf6 retransmit-interval 5
 ipv6 ospf6 priority 1
 ipv6 ospf6 transmit-delay 1
 ipv6 ospf6 instance-id 0
!
router ospf6
 router-id 132.247.253.7
  area 0.0.0.0 range 2001:448:2:2::/64
 area 0.0.0.0 range 2001:1218::/48
interface gif0 area 0.0.0.0
!
```

La tabla 97 muestra los resultados del ping6 realizado desde el PC FreeBSD hacia la página de www.ipv6forum.com, la cual por medio de una consulta DNS obtiene la dirección IPv6 de la página (2001:a18:1:20::22). La página es alcanzada a través del Ruteador-1, que esta directamente conectado a la red IPv6 global. También se muestra la tabla de routing IPv6 generada en PC FreeBSD después de haber realizado la redistribución de rutas BGP4+ a OSPFv3.

Tabla 97. Resultados de la prueba de conectividad IPv6 por OSPFv3 y redistribución de rutas BGP4+.

Tabla de Ruteo en PC FreeBSD

```
Ospfv3# show ipv6 ospf6 route
*N E2 2001:200::/32                fe80::2e0:fcff:fe0a:b0ee    myk0 00:00:11
*N E2 2001:200:e000::/35           fe80::2e0:fcff:fe0a:b0ee    myk0 00:00:11
*N E2 2001:208::/32                fe80::2e0:fcff:fe0a:b0ee    myk0 00:00:11
*N E2 2001:218::/32                fe80::2e0:fcff:fe0a:b0ee    myk0 00:00:11
*N E2 2001:220::/35                fe80::2e0:fcff:fe0a:b0ee    myk0 00:00:11
*N E2 2001:220:2000::/35           fe80::2e0:fcff:fe0a:b0ee    myk0 00:00:11
*N E2 2001:220:4000::/34           fe80::2e0:fcff:fe0a:b0ee    myk0 00:00:11
*N E2 2001:220:8000::/33           fe80::2e0:fcff:fe0a:b0ee    myk0 00:00:11
*N E2 2001:228::/35                fe80::2e0:fcff:fe0a:b0ee    myk0 00:00:11
*N E2 2001:238::/32                fe80::2e0:fcff:fe0a:b0ee    myk0 00:00:11
*N E2 2001:240::/32                fe80::2e0:fcff:fe0a:b0ee    myk0 00:00:11
*N E2 2001:250::/32                fe80::2e0:fcff:fe0a:b0ee    myk0 00:00:11
*N E2 2001:251::/32                fe80::2e0:fcff:fe0a:b0ee    myk0 00:00:11
*N E2 2001:254::/32                fe80::2e0:fcff:fe0a:b0ee    myk0 00:00:11
*N E2 2001:254::/33                fe80::2e0:fcff:fe0a:b0ee    myk0 00:00:11
*N E2 2001:254:8000::/33           fe80::2e0:fcff:fe0a:b0ee    myk0 00:00:11
*N E2 2001:256::/32                fe80::2e0:fcff:fe0a:b0ee    myk0 00:00:11
*N E2 2001:258::/32                fe80::2e0:fcff:fe0a:b0ee    myk0 00:00:11
*N E2 2001:260::/32                fe80::2e0:fcff:fe0a:b0ee    myk0 00:00:11
*N E2 2001:268::/32                fe80::2e0:fcff:fe0a:b0ee    myk0 00:00:11
*N E2 2001:278::/32                fe80::2e0:fcff:fe0a:b0ee    myk0 00:00:11
*N E2 2001:288::/32                fe80::2e0:fcff:fe0a:b0ee    myk0 00:00:11
 --More--
```

Ping6 de PC FreeBSD a www.ipv6forum.com

```
[FreeBSD]$ ping6 www.ipv6forum.com
PING6(56=40+8+8 bytes) 2001:448:2:2::2 --> 2001:a18:1:20::22
16 bytes from 2001:a18:1:20::22, icmp_seq=54 hlim=47 time=245.982 ms
16 bytes from 2001:a18:1:20::22, icmp_seq=55 hlim=47 time=249.206 ms
16 bytes from 2001:a18:1:20::22, icmp_seq=56 hlim=47 time=246.122 ms
16 bytes from 2001:a18:1:20::22, icmp_seq=57 hlim=47 time=244.560 ms
16 bytes from 2001:a18:1:20::22, icmp_seq=58 hlim=47 time=244.189 ms
^C
```

Al final de esta prueba se consiguió exitosamente la conectividad IPv6 por medio de túneles 6in4, con el uso de protocolos de ruteo dinámico IGP y EGP. El protocolo de ruteo dinámico IGP dentro del AS fue OSPFv3, por el cual se aprenden las rutas anunciadas de la red IPv6 global por medio de BGP4+. Al ser OSPFv3 un protocolo IGP, sólo se pueden anunciar rutas dentro del mismo AS, pero para conocer las rutas de otros AS, es necesario de un protocolo EGP, en nuestro caso BGP4+. Para hacer que las rutas conocidas por EGP puedan ser anunciadas o inyectadas a IGP es necesario del uso de redistribución de rutas.
La tabla 97 muestra los resultados obtenidos al mostrar las rutas aprendidas IPv6 de la red IPv6 global. Al momento que PC FreeBSD aprende las rutas de la red IPv6 global, se obtiene el acceso a cualquier red IPv6 del mundo, y por lo tanto a cualquier recurso disponible. Para comprobar la conectividad por IPv6, se realiza ping6 a www.ipv6forum.com, el cual es un recurso que se encuentra ubicado en Luxemburgo, en la dirección IPv6 (2001:a18:1:20::22).

8.8. PRUEBA DE CONECTIVIDAD IPv6 CON IS-IS COMO PROTOCOLO IGP Y BGP4+ COMO PROTOCOLO EGP, CON SERVIDOR WEB (APACHE) Y DNS (BIND)

La figura 151 muestra la maqueta de prueba realizada, para verificar la conectividad nativa IPv6 y el uso de servicios Web y DNS con soporte IPv6. La maqueta está compuesta de 4 ruteadores, de los cuales, dos se encuentran en el AS100, y los otros 2 se encuentran en

diferentes AS, el 200 y 300. El soporte de IPv4 e IPv6 está habilitado completamente, obteniendo con esto, conectividad y accesibilidad por ambos stack, de la misma forma que se encuentra actualmente RedCUDI.

Para que los enrutadores del AS100 puedan aprender las redes internas, es necesario del uso de un protocolo de ruteo dinámico IGP, en nuestro caso ISIS, en donde los dos enrutadores pertenecen al área 49.0001 de nivel L2. Para el caso de los enrutadores de los AS 200 y 300, no hubo necesidad de un protocolo IGP, ya que las redes se encuentran directamente conectadas a cada uno de ellos. Ahora bien, para que las redes puedan ser accesibles desde cada extremo de la maqueta, es decir desde el cliente WinXP hasta el Servidor Web/DNS y viceversa, se requiere de un protocolo de ruteo dinámico EGP que ayude a conocer las redes entre los AS´s, en nuestro caso BGP4+.

Para el AS100, hubo necesidad de realizar iBGP entre los enrutadores internos y eBGP con los enrutadores de los AS´s vecinos, para que de esta forma se aprendieran la totalidad de redes anunciadas dentro del AS y las anunciadas por los vecinos; logrando con esto una full-mesh. Para los AS´s, 200 y 300 sólo hubo necesidad de realizar eBGP y anunciar sus redes internas con el AS100, por medio del establecimiento de vecindad peer.

Ya que se obtuvo la conectividad IPv4 e IPv6 completa entre todas las redes; sólo resto configurar y levantar en el host servidor los demonios de apache y bind, para dar los servicios de web y dns. El host servidor es una pc i386 con plataforma FreeBSD, a la cual se bajaron, compilaron e instalaron los paquetes de software libre apache 2.2.4 y bind 9.4.0, ya que soportan IPv6.

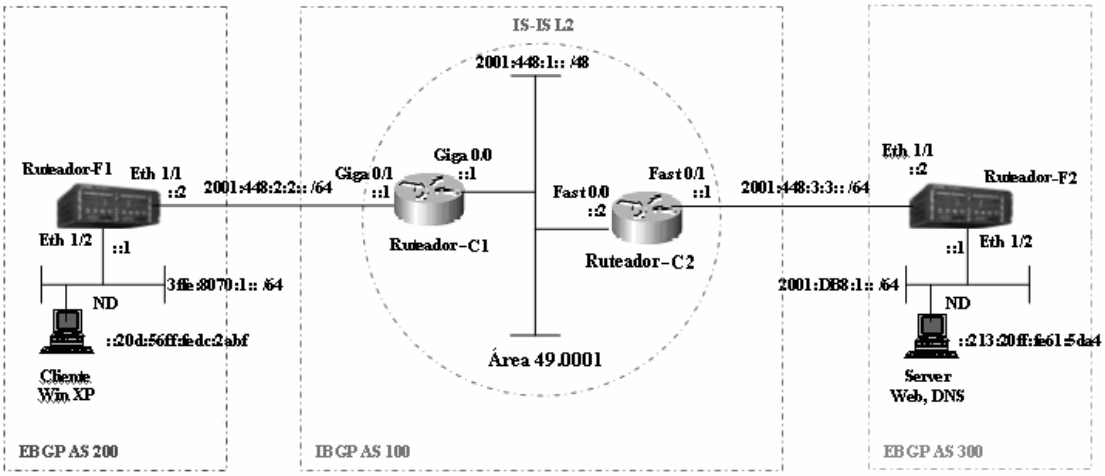


Figura 151. Maqueta de prueba ISIS y BGP4+, con servidores WEB y DNS

La tabla 98 describe los pasos de configuración efectuados únicamente en el Ruteador-C1 y Ruteador-F1, tanto en ISIS y BGP4+, ya que las configuraciones del Ruteador-C2 y Ruteador-F2, son exactamente lo mismo. Para la configuracion de direcciones IPv6 en el hosts Cliente WinXP y host Servidor Web/DNS, no se mostrarán, ya que fueron autoconfiguradas por medio de mensajes del tipo Anuncio de Enrutador. Y por último, la configuración de los demonios de apache y bind, sólo se mostrarán los archivos de configuración obtenidos (tabla 99), para ofrecer los servicios de web y dns.

Tabla 98. Configuración de equipos para la prueba de conectividad IPv6 por medio de ISIS y BGP4+, con servidores WEB y DNS

Ruteador-C1	**Ruteador-F1**
Habilitar IPv6 de forma global	Habilitar IPv6 de forma global
Configurar las direcciones IPv4 para lograr conectividad con Ruteador-C2 y Ruteador-F1	Configurar las direcciones IPv4 para lograr conectividad con Ruteador-C1 y con WinXP
Configurar las direcciones IPv6 para lograr conectividad con Ruteador-C2 y Ruteador-F1	Configurar las direcciones IPv6 para lograr conectividad con Ruteador-C1 y con WinXP
Habilitar de forma global un proceso de routing ISIS	Habilitar de forma global un proceso de routing BGP4+
Dentro del proceso de routing, configurar la entidad de red ISIS (NET)	Dentro del proceso BGP4+, configurar el número de AS, al que pertenece el enrutador
Dentro del proceso de routing, configurar globalmente al nivel L2 ISIS al que pertenece el enrutador	Establecer peer eBGP con el Ruteador-C1, tanto en IPv4 como en IPv6, de forma que puedan intercambiar información de enrutamiento.
Dentro de la interfaces Giga0/0 y Giga 0/1, habilitar el proceso de routing ISIS tanto IPv4 e IPv6	Dentro del proceso de rounting BGP4+, ingresar a la familia IPv4 unicast
Habilitar de forma global un proceso de routing BGP4+	Dentro de la familia IPv4 unicast, anunciar (inyectar) los prefijos IPv4, para que sea anunciada dentro de la base de datos BGP IPv4.
Dentro del proceso de rountig BGP4+, configurar el router-id	Dentro del proceso de rounting BGP4+, ingresar a la familia IPv6 unicast
Establecer peer iBGP con el Ruteador-C2, y peer eBGP con el Ruteador-F1, tanto en IPv4 como en IPv6, de forma que puedan intercambiar información de enrutamiento. En este punto, por default, los enrutadores sólo intercambiarán prefijos de direcciones IPv4 unicast, los pasos siguientes habilitará el intercambio de prefijos IPv6 unicast.	Habilitar dentro de la familia IPv6 unicast, al peer eBGP Ruteador-C2, para poder iniciar el intercambio de prefijos de direcciones IPv6 unicast.
Dentro del proceso de rountig BGP4+, anunciar directamente las redes IPv4, aquí no hay necesidad de ingresar a la familia IPv4 unicast.	Dentro de la familia IPv6 unicast, anunciar (inyectar) los prefijos IPv6, para que sea anunciada dentro de la base de datos BGP IPv6.
Dentro del proceso de rounting BGP4+, ingresar a la familia IPv6 unicast	
Habilitar dentro de la familia IPv6 unicast, al peer iBGP Ruteador-C2 y eBGP Ruteador-F1, para poder iniciar el intercambio de prefijos de direcciones IPv6 unicast.	
Dentro de la familia IPv6 unicast, anunciar (inyectar) los prefijos IPv6, para que sea anunciada dentro de la base de datos BGP IPv6.	

A continuación la tabla 99, muestra los resultados obtenidos de la configuración del Ruteador-C1, Ruteador-F1, Servidor WEB/DNS y del Cliente WinXP llevado a cabo para probar la interoperabilidad por IPv6, entre los diferentes AS´s y del soporte IPv6 que presentan las aplicaciones WEB y DNS.

Tabla 99. Resultado de la configuración de los equipos para la prueba de ISIS y BGP4+, con servidores WEB y DNS.

Ruteador-C1

```
Router1#show running
!
ipv6 unicast-routing
!
interface GigabitEthernet0/0
 ip address 192.168.1.1 255.255.255.0
 ip router isis
 ipv6 address 2001:448:1::1/48
 ipv6 router isis
!
interface GigabitEthernet0/1
 ip address 172.16.1.1 255.255.255.0
 ip router isis
 ipv6 address 2001:448:2:2::1/64
 ipv6 router isis
!
router isis
 net 49.0001.1111.1111.1111.00
 is-type level-2-only
!
router bgp 100
 network 172.16.1.0
 network 192.168.1.0
 neighbor 2001:448:1::2 remote-as 100
 neighbor 2001:448:2:2::2 remote-as 200
 neighbor 172.16.1.2 remote-as 200
 neighbor 192.168.1.2 remote-as 100
 !
 address-family ipv6
 neighbor 2001:448:1::2 activate
 neighbor 2001:448:2:2::2 activate
 network 2001:448:1::1/48
 network 2001:448:2:2::1/64
 !
```

Ruteador-F1

```
Ruteador-F1#show running-config
!
ipv6 unicast-routing
!
interface ethernet 1/1
ip address 172.16.1.2/24
ipv6 address 2001:448:2:2::2/64
!
interface ethernet 1/2
ip address 172.16.2.1/24
ipv6 address 3ffe:8070:1::1/64
!
router bgp
 local-as 200
 neighbor 172.16.1.1 remote-as 100
 neighbor 2001:448:2:2::1 remote-as 100
 address-family ipv4 unicast
 network 172.16.2.0/24
 network 172.16.1.0/24
 address-family ipv6 unicast
 network 3ffe:8070:1::/64
 neighbor 2001:448:2:2::1 activate
  !
```

Servidor DNS y WEB (FreeBSD)

```
[Servidor]# ifconfig -a
myk0: flags=8943<UP,BROADCAST,RUNNING,PROMISC,SIMPLEX,MULTICAST> mtu 1500
        options=2b<RXCSUM,TXCSUM,VLAN_MTU,JUMBO_MTU>
        inet6 fe80::213:20ff:fe61:5da4%myk0 prefixlen 64 scopeid 0x1
        inet 10.1.2.2 netmask 0xffffff00 broadcast 10.1.2.255
        inet6 2001:db8:1:0:213:20ff:fe61:5da4 prefixlen 64 autoconf
```

```
        ether 00:13:20:61:5d:a4
        media: Ethernet autoselect (100baseTX <full-duplex>)
        status: active
```

DNS (BIND)
Configurando la zona "ipv6.netlab.unam.mx" en el archivo named.conf:

```
zone "ipv6.netlab.unam.mx" {
        type master;
        file "master/ipv6.netlab.unam.mx";
        allow-query { any; };
};
```

Configurando el archivo de zona "ipv6.netlab.unam.mx" ubicado en el directorio master/:

```
; This file is automatically edited by the `make-localhost' script in
; the /etc/namedb directory.
;

$TTL    3600
@       IN      SOA     ns1.ipv6.netlab.unam.mx. admin.example.org.  (
                                2007032501          ; Serial
                                3600        ; Refresh
                                900         ; Retry
                                3600000     ; Expire
                                3600 )      ; Minimum
;DNS Servers

        IN      NS      ns1.ipv6.netlab.unam.mx.

;Machine Names

localhost  IN   A       127.0.0.1
                IN      AAAA    ::1


ns1     IN      A       10.1.2.2
        IN      AAAA    2001:db8:1:0:213:20ff:fe61:5da4

www     IN      A       10.1.2.2
        IN      AAAA    2001:db8:1:0:213:20ff:fe61:5da4

maquina1 IN     A       172.16.2.3
                IN      AAAA    3ffe:8070:1:0:20d:56ff:fedc:2abf

www4    IN      A       10.1.2.2

www6    IN      AAAA    2001:db8:1:0:213:20ff:fe61:5da4
```

WEB (APACHE)

Editar el archivo httpd.conf
Especificando el puerto que se va usar:

```
# Listen 80
Listen [::]:80
```

Especificando el directorio donde servirás los documentos que quieres publicar:

```
#
DocumentRoot "/usr/local/apache2/htdocs"
#
```

Cliente WinXP

c:\>ipv6 if 5

```
Interface 5: Ethernet: Local Area Connection
  Guid {085DBF9E-8E26-4084-9E66-3C49316CCB8B}
  uses Neighbor Discovery
  uses Router Discovery
  link-layer address: 00-0d-56-dc-2a-bf
    preferred global 3ffe:8070:1:0:7c67:4fe4:89e1:293e, life 6d22h6m22s/22h3m35s(temporary)
    preferred global 3ffe:8070:1:0:20d:56ff:fedc:2abf, life 29d23h56m55s/6d23h56m55s (public)
    preferred link-local fe80::20d:56ff:fedc:2abf, life infinite
```

```
   multicast interface-local ff01::1, 1 refs, not reportable
   multicast link-local ff02::1, 1 refs, not reportable
   multicast link-local ff02::1:ffdc:2abf, 2 refs, last reporter
   multicast link-local ff02::1:ffe1:293e, 1 refs, last reporter
 link MTU 1500 (true link MTU 1500)
 current hop limit 64
 reachable time 26000ms (base 30000ms)
 retransmission interval 1000ms
 DAD transmits 1
 default site prefix length 48

c:\>ipv6 rt
3ffe:8070:1::/64 -> 5 pref 8 life 29d23h59m7s (autoconf)
::/0 -> 5/fe80::20c:dbff:fef6:601 pref 256 life 29m7s (autoconf)
```

La tabla 100 muestra los resultados obtenidos de las tablas generadas de BGP4+ y ruteo del Ruteador-C1 y Ruteador-F1, para lograr la conectividad por IPv6 con ayuda del protocolo de ruteo ISIS y BGP4+ entre diferentes AS. De la misma forma se muestran los resultados obtenidos al hacer consultas de registros AAAA al servidor DNS y de los ping6 y traceroute6 realizados para verificar la conectividad IPv6 nativa lograda entre los AS´s.

Para el servidor Web se demuestra la conectividad IPv6 mediante las bitácoras generadas en el archivo “access_log” del servidor WEB, así como también la captura de pantalla tomada al navegador Web Mozilla al momento de acceder al servidor Web Apache por medio de IPv4 e IPv6.

Tabla 100. Resultados obtenidos de la prueba de ISIS y BGP4+, con servidores WEB y DNS

Resultados del Ruteador-C1 de los peers establecidos

```
Ruteador-C1#show bgp ipv6 unicast summary
BGP router identifier 192.168.10.1, local AS number 100
BGP table version is 11, main routing table version 11
5 network entries using 745 bytes of memory
6 path entries using 456 bytes of memory
5/4 BGP path/bestpath attribute entries using 620 bytes of memory
2 BGP AS-PATH entries using 48 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory
BGP using 1869 total bytes of memory
BGP activity 14/4 prefixes, 20/8 paths, scan interval 60 secs

Neighbor        V    AS  MsgRcvd  MsgSent   TblVer  InQ  OutQ  Up/Down  State/PfxRcd
2001:448:1::2   4   100       99      101       11    0     0  01:29:05            3
2001:448:2:2::2 4   200       94       91       11    0     0  01:17:34            1
```

Resultados de la tabla de ruteo del Ruteador-C1

```
Ruteador-C1#show ipv6 route
IPv6 Routing Table - 9 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       U - Per-user Static route
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
C   2001:448:1::/48 [0/0]
     via ::, GigabitEthernet0/0
L   2001:448:1::1/128 [0/0]
     via ::, GigabitEthernet0/0
C   2001:448:2:2::/64 [0/0]
     via ::, GigabitEthernet0/1
L   2001:448:2:2::1/128 [0/0]
     via ::, GigabitEthernet0/1
I2  2001:448:3:3::/64 [115/20]
     via FE80::20D:29FF:FE09:3D00, GigabitEthernet0/0
B   2001:DB8:1::/64 [200/0]
     via 2001:448:3:3::2
```

```
B  3FFE:8070:1::/64 [20/0]
     via FE80::20C:DBFF:FEF6:600, GigabitEthernet0/1
L   FE80::/10 [0/0]
     via ::, Null0
L   FF00::/8 [0/0]
     via ::, Null0
```

Resultados del Ruteador-F1 de los peers establecidos

```
Ruteador-F1#show ipv6 bgp summary
 BGP4 Summary
 Router ID: 172.16.1.2   Local AS Number : 200
 Confederation Identifier : not configured
 Confederation Peers:
 Maximum Number of Paths Supported for Load Sharing : 1
 Number of Neighbors Configured : 1, UP: 1
 Number of Routes Installed : 5
 Number of Routes Advertising to All Neighbors : 1
 Number of Attribute Entries Installed : 4
 Neighbor Address   AS#        State        Time     Rt:Accepted          Filtered     Sent     ToSend
 2001:448:2:2::1    100        ESTAB        1h27m 6s     4                    0          1        0
```

Resultados del Ruteador-F1, de las rutas aprendidas por BGP4+

```
Ruteador-F1#show ipv6 bgp routes
Total number of BGP Routes: 5
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
      E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
      S:SUPPRESSED F:FILTERED s:STALE
          Prefix                Next Hop              Metric    LocPrf  Weight        Status
1     2001:448:1::/48           2001:448:2:2::1       0         100     0             BE
       AS_PATH: 100
2     2001:448:2:2::/64         2001:448:2:2::1       0         100     0             BE
       AS_PATH: 100
3     2001:448:3:3::/64         2001:448:2:2::1                 100     0             BE
       AS_PATH: 100
4     2001:db8:1::/64           2001:448:2:2::1                 100     0             BE
       AS_PATH: 100 300
5     3ffe:8070:1::/64          ::                    0         100     32768         BL
       AS_PATH:
```

Traceroute6 del cliente WinXP al servidor WEB/DNS

```
c:\>tracert -d www.ipv6.netlab.unam.mx
Tracing route to www.ipv6.netlab.unam.mx [2001:db8:1:0:213:20ff:fe61:5da4]
over a maximum of 30 hops:
  1    <1 ms    <1 ms    <1 ms  3ffe:8070:1::1
  2    <1 ms    <1 ms    <1 ms  2001:448:2:2::1
  3     1 ms     1 ms     1 ms  2001:448:1::2
  4     2 ms     1 ms     1 ms  2001:448:3:3::2
  5     2 ms     1 ms     2 ms  2001:db8:1:0:213:20ff:fe61:5da4
Trace complete.
```

Traceroute6 del servidor WEB/DNS al cliente WinXP

```
[Servidor WEB/DNS]#traceroute6 3ffe:8070:1:0:20d:56ff:fedc:2abf
traceroute6 to 3ffe:8070:1:0:20d:56ff:fedc:2abf (3ffe:8070:1:0:20d:56ff:fedc:2abf) from 2001:db8:1:0:213:20ff:fe61:5da4, 64
hops max, 12 byte packets
 1  2001:db8:1::1  0.462 ms  0.373 ms  0.289 ms
 2  2001:448:3:3::1  1.133 ms  1.139 ms  1.033 ms
 3  2001:448:1::1  1.920 ms  2.011 ms  1.653 ms
 4  2001:448:2:2::2  1.930 ms  2.372 ms  2.000 ms
 5  3ffe:8070:1:0:20d:56ff:fedc:2abf  1.941 ms  2.125 ms *
```

Consultas de registros AAAA al servidor DNS, desde el cliente WinXP

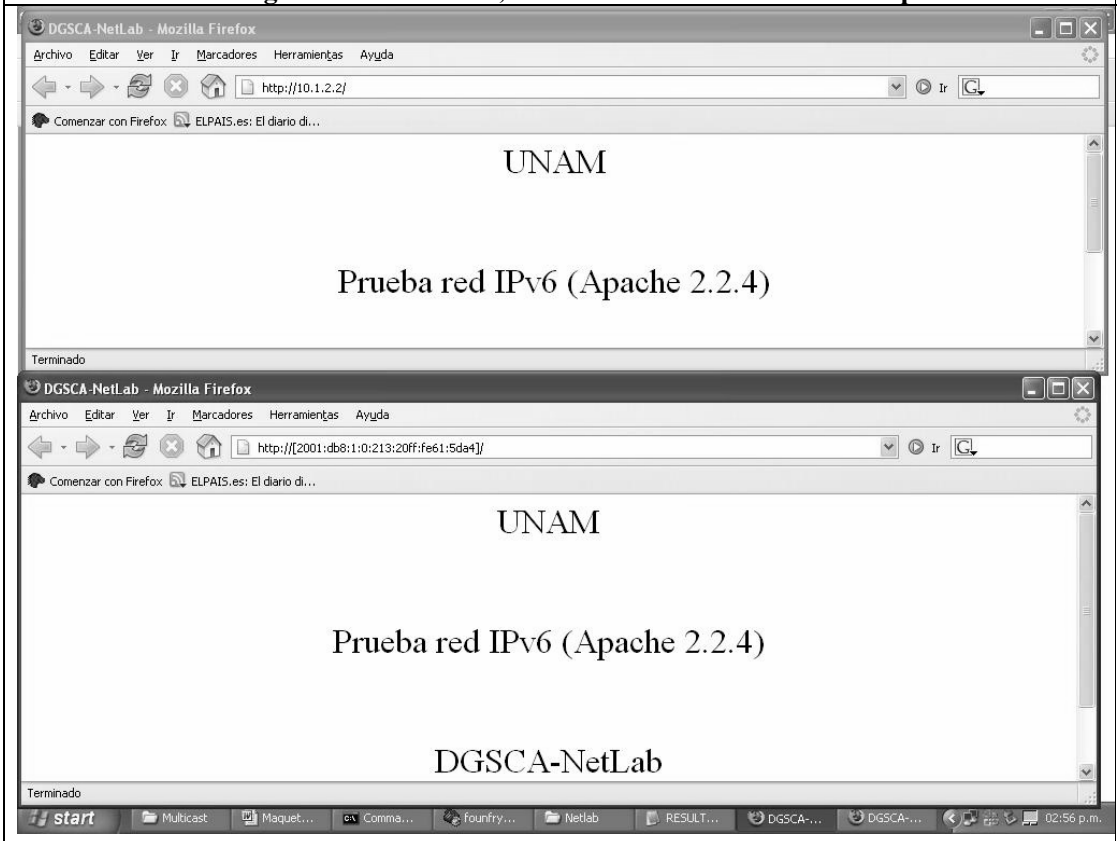
```
c:\>nslookup
> set q=AAAA
> www6.ipv6.netlab.unam.mx

www6.ipv6.netlab.unam.mx       AAAA IPv6 address = 2001:db8:1:0:213:20ff:fe61:5da4
ipv6.netlab.unam.mx     nameserver = ns1.ipv6.netlab.unam.mx
ns1.ipv6.netlab.unam.mx internet address = 10.1.2.2
ns1.ipv6.netlab.unam.mx AAAA IPv6 address = 2001:db8:1:0:213:20ff:fe61:5da4

> ns1.ipv6.netlab.unam.mx

ns1.ipv6.netlab.unam.mx AAAA IPv6 address = 2001:db8:1:0:213:20ff:fe61:5da4
```

ipv6.netlab.unam.mx nameserver = ns1.ipv6.netlab.unam.mx
ns1.ipv6.netlab.unam.mx internet address = 10.1.2.2

Resultados de la bitácora generada en el archivo "access_log" de Apache

172.16.2.3 - - [28/Mar/2007:14:42:11 -0600] "GET / HTTP/1.1" 200 712
172.16.2.3 - - [28/Mar/2007:14:42:11 -0600] "GET /favicon.ico HTTP/1.1" 404 209
172.16.2.3 - - [28/Mar/2007:14:42:11 -0600] "GET /favicon.ico HTTP/1.1" 404 209
3ffe:8070:1:0:88b9:1ca7:a44c:3dc - - [28/Mar/2007:14:43:43 -0600] "GET / HTTP/1.1" 200 712
3ffe:8070:1:0:88b9:1ca7:a44c:3dc - - [28/Mar/2007:14:43:44 -0600] "GET /favicon.ico HTTP/1.1" 404 209
3ffe:8070:1:0:88b9:1ca7:a44c:3dc - - [28/Mar/2007:14:43:44 -0600] "GET /favicon.ico HTTP/1.1" 404 209
172.16.2.4 - - [28/Mar/2007:14:51:50 -0600] "GET / HTTP/1.1" 200 712
10.1.2.2 - - [28/Mar/2007:14:52:07 -0600] "GET / HTTP/1.1" 200 712
10.1.2.2 - - [28/Mar/2007:14:52:08 -0600] "GET /favicon.ico HTTP/1.1" 404 209
172.16.2.3 - - [28/Mar/2007:14:52:40 -0600] "GET / HTTP/1.1" 304 -
3ffe:8070:1:0:88b9:1ca7:a44c:3dc - - [28/Mar/2007:14:53:51 -0600] "GET / HTTP/1.1" 304 -
3ffe:8070:1:0:85cb:c4b3:1f89:a456 - - [28/Mar/2007:14:55:29 -0600] "GET/ HTTP/1.1" 200 712

Resultados del Navegador Web Mozilla, consultando al servidor WEB Apache



Al final de esta prueba se consiguió exitosamente la conectividad nativa IPv6 entre los diferentes Sistemas Autónomos, y el funcionamiento de los servidores WEB y DNS. El AS100 estuvo conformado por dos enrutadores, los cuales utilizaron como protocolo de enrutamiento IGP a ISIS, con esto aprendiendo todos los prefijos comprendidos dentro del AS. El AS 200 y 300 al constar sólo de un enrutador, no hubo necesidad de habilitar algún protocolo IGP. Dentro del AS 200 y 300 se conectaron directamente a los enrutadores el cliente WinXP y el Servidor DNS, respectivamente, los cuales obtuvieron su dirección IPv6 mediante autoconfiguración. Al tener varios AS´s involucrados en la maqueta de prueba, se tuvo la necesidad de emplear a BGP4+ como protcolo EGP para lograr la conectividad completa entre ellos, tanto en IPv4 como en IPv6.

En el AS100 se realizó iBGP para que los dos enrutadores involucrados tuvieran conocimiento de todos los prefijos IPv4 e IPv6 anunciados por BGP4+. Para el AS200 y 300 se realizó eBGP para que pudieran anunciar sus redes al AS100 y de esta forma conocer todos los prefijos IPv6 e IPv4 utilizados en la maqueta de prueba. Al realizar iBGP y eBGP se logra la conectividad completa entre todos los AS´s, y al tener acceso a cualquier parte de la red, desde cualquier punto, se pudieron ver a nivel de capa 3 del modelo OSI, el cliente WinXP y el Servidor, con esto ganado acceso a los servicios de WEB y DNS.

El servidor Web y DNS al tener soporte IPv6, se pudieron hacer consultas a ellos por medio de IPv4 e IPv6, de igual forma que sucede en cualquier red que esté utilizando ambos stack de IP, un ejemplo de ello, la RedCUDI. Con la ayuda de los DNS las redes IPv6 pueden ser manejadas de la misma forma que sucede en la actualidad, ya que cualquier recurso de la red, puede ser identificado sólo por el nombre asociado en el servidor DNS, evitando con esto la necesidad de aprenderse las grandes direcciones IPv6 utilizadas. Con esto se puede observar qué tan fácil puede ser, configurar una red para que opere tanto en IPv4 como IPv6 mientras dura el proceso de migración a IPv6, y utilizar cualquier servicio y aplicación que sea requerida.

Capítulo

9.Conclusión

Conclusión

Internet tiene sus inicios en los años 60´s en Estados Unidos con la aparición de la red de ARPANET, la cual tuvo como principal objetivo interconectar las computadoras de los centros de la defensa de EEUU, para evitar la pérdida de comunicación en el hipotético de algún ataque militar ruso. Debido a la gran aceptación obtenida, se fueron conectando más y más estaciones de trabajo, hasta el punto de necesitar otro protocolo de comunicación que pudiera cubrir las necesidades existentes. Se adopta al stack de protocolos TCP/IP como estándar de comunicación dentro de la red de ARPANET y posteriormente se convierten en los protocolos de comunicación de capa 3 y 4 del modelo OSI, más utilizado a nivel mundial, por la facilidad que presentaba para comunicar ordenadores de diferentes arquitecturas, hardware, software y fabricantes.

Desde la época de ARPANET, Internet ha tenido un gran crecimiento en forma desmedida en un lapso de tiempo muy corto, debido a la versatilidad que presenta para ofrecer una gran variedad de servicios, como son: publicidad, ventas en línea, entretenimiento, comunicación personal, correo electrónico, transacciones financieras, investigación, educación, trabajos de colaboración a distancia, etc. Internet creció más allá de las expectativas de las personas que originalmente lo concibieron y lo que inició como un experimento solamente de investigación, pronto llegó a ser un éxito totalmente comercial. Todo esfuerzo de planeación ha superado la infraestructura prevista de comunicaciones, de manera que las aplicaciones de investigación y educación son limitadas por la velocidad, ancho de banda, direcciones IP, retardo de transmisión, etc.

Es por eso que organismos gubernamentales, de investigación y educación en todo el mundo se unieron para crear una red paralela a Internet, que usará tecnología de punta para mover grandes cantidades de información, de 100 a 1000 veces más rápido que el Internet actual, restringiéndola únicamente para uso de investigación y educación.

La red de Internet2, sirve de plataforma para correr y desarrollar aplicaciones avanzadas que necesitan de gran ancho de banda, permitiendo la colaboración entre académicos e investigadores, así como el acceso interactivo a la información y recursos de forma que no podrían ser posibles en el Internet comercial de hoy en día. Aprendizaje a distancia e interactivo, acceso remoto a únicos instrumentos científicos, tiempo de acceso real a grandes bases de datos, y alta definición de video-streaming son todos posibles con el alto desempeño de las redes avanzadas de Internet2.

En Internet2 al igual que Internet y la mayoría de redes privadas, manejan tecnologías basadas en IP, por lo que utilizan el suite de protocolos TCP/IP para interconectarse. Debido a que la versión IP actual en uso es la 4 y presenta ciertos inconvenientes como la escasez de direcciones, explosión de las tablas de enrutamiento, demasiados sistemas autónomos conectados, así como también los añadidos que se le han hecho al protocolo como son QoS, IPSec y movilidad, han ocasionado un manejo ineficiente y deterioro del mismo. Es por tales motivos que la IETF (Internet Engineering Task Force) toma cartas en el asunto, y crea las recomendaciones para la siguiente generación del protocolo IP, llamándolo IPng ó IPv6.

El protocolo IP versión 6 es la siguiente generación del Protocolo de Internet que vendrá a suceder a IP versión 4, con los siguientes cambios importantes:

- Mayor direccionamiento IP.
- Plug & Play "Autoconfiguración"
- Direcciones Unicast, Anycast y Multicast
- Simplicidad en el formato de la cabecera IP.
- Mejoramiento en las extensiones y opciones.
- Capacidad de etiquetar el flujo.
- Capacidad de autenticación y privacidad.
- Renumeración y multi-homing
- Características de movilidad

Se dice que la característica más importante de IPv6 frente a IPv4 es la escalabilidad, ya que los puntos anteriores son particularidades básicas, pues la propia estructura del protocolo permite que crezca, o dicho de otro modo sea escalable, según las nuevas necesidades y aplicaciones o servicios lo vayan requiriendo.

Como es de saber, IP es un protocolo de capa 3 del modelo OSI, lo que lo hace el responsable de entregar los paquetes de datos o datagramas del host origen al host destino de los servicios y aplicaciones que corren sobre las redes IP, por lo que hablar de una migración de IPv4 a IPv6 involucra cambios significativos en las redes de datos, así como también en las de voz que utilizan VoIP, desde los equipos de CORE o Backbone, hasta los equipos finales de los usuarios, es decir las computadoras y teléfonos IP que son las generadoras del tráfico que fluye por la red.

Anteriormente se había mencionado, que la mayoría de las redes corren sobre el protocolo IP, resultando con esto, que todas las aplicaciones y servicios como son los protocolos de enrutamiento, servicios DNS, servicios WEB, de correo electrónico, de acceso remoto, de transferencia de archivos, de videoconferencia, de streaming de audio y video, de VoIP, de Chat, de administración, de aplicaciones de Internet2 y demás habidas no mencionadas, se vean en la necesidad de actualizarlas para soportar el stack de IPv6. Aunque este proceso de actualización ya ha sido iniciado desde hace unos años por muchos desarrolladores de las aplicaciones, también es indispensable que los administradores hagan su parte en sus redes, para que el proceso de migración marche paulatinamente y no sean tomados por sorpresa.

Como es de imaginar la migración a IPv6 no puede ser de una día a otro, por eso se han desarrollo diferentes mecanismos de transición de IPv4 a IPv6. El desarrollo de IPv6 depende en gran medida de la convivencia con las redes IPv4 existentes, y la coexistencia de ambos protocolos propiciará, a los administradores y usuarios de las redes, obtener la suficiente experiencia del manejo del mismo, y así mismo nos preparará a una exitosa migración completa de IPv6.

Hay que destacar, que la migración a IPv6 es inminente, se preguntaran ¿por qué hago esta aseveración? pues debido a que el pasado 20 de junio de 2007, LACNIC el Registro de direcciones de Internet de la región de Latino América y el Caribe, realizó un comunicado

de prensa en el que informa que el stock central de direcciones IPv4 para el año 2011, podría estar definitivamente agotado, por lo que se ven en la necesidad de anunciar el lanzamiento de una campaña regional para lograr que antes del 1º de enero de 2011 se logre la total adaptación de las redes de la región a la nueva versión seis del protocolo IP (IPv6)[43].

Por otra parte, desde los inicios de la red de Internet2 de México CUDI, ha estado funcionado con IPv4 como protocolo de la capa de red, sin embargo la tendencia mundial de las demás NREN´s, ha propiciado la migración a IPv6 desde los equipos de Backbone hasta los equipos terminales; es por tal motivo que se crea el grupo de trabajo IPv6 en CUDI, con el principal objetivo de migrar de forma nativa a IPv6 en toda la red de CUDI y realizar trabajos en colaboración con los demás grupos de trabajo de CUDI y NREN´s. Pasando por una serie de pruebas y conexiones por túneles de IPv6 sobre IPv4, en diciembre de 2001, el grupo de trabajo IPv6 de CUDI y en colaboración con el grupo de trabajo de IPv6 de la UNAM, logran exitosamente instalar nativamente IPv6 en los equipos de Backbone de RedCUDI, con un prefijo IPv6 asignado por la UNAM.

Después de obtener la suficiente experiencia con el protocolo, así como contar con la cantidad suficiente de clientes (Asociados y Afiliados Académicos) para brindar conectividad y bloques de direcciones IPv6, e igualmente cubrir con los requisitos solicitados por LACNIC para adjudicar un prefijo IPv6, el grupo de trabajo IPv6 anuncia en la reunión de primavera de 2005, la solicitud de un bloque de direcciones IPv6 propio para producción de tipos sTLA (/32) para la red de Internet2 de México (CUDI) ante LACNIC, y para el 15 de noviembre de 2005, se adquiere finalmente de LACNIC un bloque de direcciones IPv6 (2001:1228::/32) para servicios de producción, el cual fue sustituyendo paulatinamente a los bloques de pruebas y producción que tenía asignado la UNAM a esta organización.

La adquisición del nuevo prefijo IPv6 para CUDI, involucró una serie de trabajos que tendrían que llevarse a cabo en un lapso de tiempo no mayor a un año, el cual es solicitado por LACNIC para que se anuncie el prefijo IPv6 adjudicado a la red global IPv6. Dentro de los trabajos desarrollados, estuvo el del nuevo direccionamiento IPv6 para el backbone de CUDI y el de asignación de bloques de direcciones IPv6, para aquellos clientes (Asociados Académicos) de RedCUDI que lo soliciten, mientras adquieren la experiencia necesaria y califican ante el NIC México para obtener su bloque propio de direcciones IPv6 para producción.

Poco tiempo después de haber sido aceptado el documento que hace referencia hasta ese momento al nuevo direccionamiento IPv6 de RedCUDI por el CDR (Comité de desarrollo de la red), se procedió a preparar y trabajar en conjunto con el NOC, en un plan de renumeración y configuración de las conexiones entre los equipos de Backbone de RedCUDI. Finalmente para principios de julio de 2006 se anuncia formalmente que la RedCUDI se encontraba configurada con el nuevo prefijo IPv6 y que el bloque IPv6 ya estaba siendo anunciado a Internet. Posteriormente se prosiguió con la renumeración de los enlaces de los Asociados Académicos directamente conectados al Backbone, brindándoles

[43] http://www.lacnic.org/sp/anuncios/2007_agotamiento_ipv4.html

la confianza para que ellos mismos desarrollaran su propio direccionamiento IPv6, plan de renumeración y configuración de sus equipos, para que de la misma forma provean de conectividad IPv6 a los Afiliados Académicos.

La renumeración consistió en la configuración de las direcciones IPv6 en cada una de las interfaces de los equipos del backbone de RedCUDI según el diseño del documento de nuevo direccionamiento IPv6 de RedCUDI, así como en la re-configuración de las sesiones iBGP y eBGP del protocolo BGP4+, y RIPng como protocolo IGP, mientras se instalan nuevas versiones de IOS en los equipos de backbone para que soporten IS-IS.

De igual forma que los trabajos antes mencionados, se tuvieron que desarrollar otros documentos que sirven de reglamentos para el uso del prefijo IPv6 dentro de la RedCUDI. El de política de ruteo IPv6, establece la manera en que se manejará el enrutamiento IPv6 de los prefijos entre los clientes internos y entre los AS´s, así como el manejo del tráfico IPv6 en la red, obteniendo con esto la prevención del crecimiento incontrolable de las tablas de enrutamiento y detener los anuncios no autorizados de prefijos IPv6. Además se elaboraron los documentos que establecen la forma de asignar bloques de direcciones IPv6, por parte de CUDI a los Asociados Académicos, así como los procedimientos que tendrán que realizar cada uno de los clientes para la recepción de un prefijo IPv6.

Además el CDR realiza una encuesta a los miembros de CUDI, el mes de agosto de 2006, para tener una visión más amplia de las necesidades y de la situación de los clientes de CUDI en cuanto a requerimientos de equipos y capacitación para el manejo de la red. De dicha encuesta el grupo de trabajo IPv6 obtiene resultados en lo que se refleja que un 5% de los Asociados Académicos conectados a CUDI, hasta ese momento, tenían asignado un bloque de direcciones IPv6, y el 95% restante aún no tenían bloques de direcciones IPv6, y además un 4% de los miembros que contestaron la encuesta, solicitaron capacitación del protocolo IPv6 al CDR, con lo que se concluyó, brindar un taller de capacitación IPv6 para la reunión de otoño de CUDI en Villahermosa Tabasco, agendándola para el 15 y 16 de Octubre de 2007, antes de la inauguración formal del evento.

Aparte de todos los trabajos antes mencionados, se han venido efectuando una serie de pruebas de conectividad IPv6 con equipos de diferentes fabricantes para examinar la interoperabilidad y buen funcionamiento del stack IPv6, así como también la puesta en práctica de algunas aplicaciones de software libre que ya presentan un alto desarrollo en el soporte del protocolo IP versión 6. La mayoría de los equipos y aplicaciones han tenido un buen desempeño, y han trabajado como lo establece los estándares de la IETF, algunos otros han presentado algunos inconvenientes que han sido superados con la actualización del release o la versión del software.

Todo lo anterior indica, que el protocolo IPv6 no es tan joven y llega a ser lo suficientemente estable y maduro frente a otros protocolos de la misma índole, conquistando con esto, la aprobación aceptable del protocolo en el mercado de las telecomunicaciones y por lo tanto mayor desarrollo y estabilidad entre las aplicaciones de los usuarios de la red global de Internet. Gran parte de la comunidad de Internet aún no ha visto las bondades que ofrece IPv6, pero cada vez existen más adeptos gracias a las posibilidades y flexibilidad que ofrece el protocolo por si mismo, implicando con esto

grandes beneficios a la comunidad de administradores, desarrolladores y usuarios de Internet.

Hasta el momento todo ha salido bien y CUDI ya está anunciando su bloque IPv6 a Internet, y al mismo tiempo se cumple uno de los objetivos del Grupo de trabajo IPv6, el de configurar nativamente a RedCUDI con el protocolo IPv6 y así mismo quedan establecidos todos los lineamientos que servirán a los miembros actuales y posteriores que se adhieran a la red. Aún quedan muchas cosas por hacer, el camino es largo como sucede en cualquier red, pero un gran paso ya se ha dado, el resto contempla levantar servicios y animar a desarrolladores de aplicaciones de Internet2 que hagan uso de IPv6 y continuar brindando capacitación y documentación a toda la comunidad nacional e internacional en todo lo que tenga que ver con la siguiente generación de IP, el cual vendrá a revolucionar la forma de manejar las redes IP. RedCUDI actualmente se ubica como una de las primeras redes a nivel Nacional y Latinoamérica en brindar conectividad nativa IPv6.

IPv6 llegará a convertirse en el protocolo de redes más usado en el mundo de las telecomunicaciones, y todos los administrados de redes, usuarios y desarrolladores irán adaptándose poco a poco a él, hasta obtener una migración completa de las redes. IPv6 claramente está listo para redes de producción, ahora sólo queda mencionar una de las frases más utilizadas en el mundo de IPv6 “**are you ready for IPv6?**”.

Referencias Bibliograficas

LIBROS

[1] BEHROUZ A. FOROUZAN.
"Transmisión de datos y redes de comunicaciones"
Segunda Edición.
McGraw Hill, 2001, Aravaca Madrid.

[2] DESMEULES RÉGIS. (May 2003).
"Cisco Self-Study: Implementing Cisco IPv6 Networks (IPv6)"
Primera Edicion.
Editorial Cisco Press, May 2003, USA.

[3] HAGEN SILVIA.
"IPv6 Essentials"
Primera Edicion
CA. O`relly Media, Inc., 2002, Gravenstein Highway North, Sebastopol.

[4] MARC BLANCHET.
"Migrating to IPv6"
Jhon Wiley & Sons, Ltd., 2006, Québec Madrid.

[5] PETE LOSHIN.
"IPv6: Theory, Protocol, and practice"
Segunda Edición
Morgan Kaufmann Publisher and printer by Elsevier, 2004, San Francisco CA.

RFC´s

[RFCMX 0003] A. Fernández, M Ricárdez, R. Morales (2007). **"Políticas de ruteo IPv6 en RedCUDI"**. Políticas.

[RFCMX 0004] A. Fernández, M Ricárdez (2007). **"Políticas de asignación de bloque de direcciones IPv6 en CUDI"**. Políticas.

[RFC 1195] R. Callon (1990). **"Use of OSI IS-IS for Routing in TCP/IP and Dual Environments"**. Standards Track.

[RFC 1546] Partridge, T. Mendez, W. Milliken (1993). **"Host Anycasting Service"**. Informational

[RFC 1752] S. Bradner, A. Mankin (January 1995). **"The Recommendation for the IP Next Generation Protocol"**. Standards Track.

[RFC 1771] Y. Rekhter, T. Li (1995). **"A Border Gateway Protocol 4 (BGP-4)"**. Standards Track.

[RFC 1981] J. McCann, S. Deering, J. Mogul (1996). **"Path MTU Discovery for IP version 6"**. Standards Track.

[RFC 2080] G. Malkin, R. Minnear (1997). **"RIPng for IPv6"**. Standards Track.

[RFC 2373] R. Hinden, S. Deering (1998). **"IP Version 6 Addressing**

Architecture". Standards Track. Updates for RFC3513 "Internet Protocol Version 6 (IPv6) Addressing Architecture".

[RFC 2374] R. Hinden, M. O'Dell, S. Deering (1998). **"An IPv6 Aggregatable Global Unicast Address Format"**. Standards Track.

[RFC 2460] S. Deering, R. Hinden (December 1998). **"Internet Protocol, Version 6 (IPv6) Specification"**. Standards Track.

[RFC 2461] T. Narten, E. Nordmark, W. Simpson (1998). **"Neighbor Discovery for IP Version 6 (IPv6)"**. Standards Track.

[RFC 2462] S. Thomson, T. Narten (1998). **"IPv6 Stateless Address Autoconfiguration"**. Standards Track.

[RFC 2473] A. Conta, S. Deering (1998). **"Generic Packet Tunneling in IPv6 Specification"**. Standards Track.

[RFC 2526] D. Jonson, S. Deering (1999). **"Reserved IPv6 Subnet Anycast Addresses"**. Standards Track.

[RFC 2529] B. Carpenter, C. Jung (1999). **"Transmission of IPv6 over IPv4 Domains without Explicit Tunnels"**. Standards Track.

[RFC 2545] P. Marques, F. Dupont (1999). **"Use of BGP-4 Multiprotocol Extensions for IPv6 Inter-Domain Routing"**. Standards Track.

[RFC 2675] D. Borman, S. Deering, R. Hinden (August 1999). **"IPv6 Jumbograms"**. Standards Track.

[RFC 2710] S. Deering, W. Fenner, B. Haberman (1999). **"Multicast Listener Discovery (MLD) for IPv6"**. Standards Track.

[RFC 2740] R. Coltun, D. Ferguson, J. Moy (1999). **"OSPF for IPv6"**. Standards Track.

[RFC 2765] E. Nordmark (2000). **"Stateless IP/ICMP Translation Algorithm (SIIT)"**. Standards Track.

[RFC 2766] G. Tsirtsis, P. Srisuresh (2000). **"Network Address Translation - Protocol Translation (NAT-PT)"**. Standards Track.

[RFC 2858] T. Bates, Y. Rekhter, R. Chandra, D. Katz (2000). **"Multiprotocol Extensions for BGP-4"**. Standards Track.

[RFC 3041] T. Narten, R. Draves (2001). **"Privacy Extensions for Stateless Address Autoconfiguration in IPv6"**. Standards Track.

[RFC 3053] A. Durand, P. Fasano, I. Guardini, D. Lento (2001). **"IPv6 Tunnel Broker"**. Informational.

[RFC 3056] B. Carpenter, K. Moore (2001). **"Connection of IPv6 Domains via IPv4 Clouds"**. Standards Track.

[RFC 3363] R. Bush, A. Durand, B. Fink, O. Gudmundsson, T. Hain (2002). **"Representing Internet Protocol version 6 (IPv6) Addresses in the Domain Name System (DNS)"**. Informational.

[RFC 3489] J. Rosenberg, J. Weinberger, C. Huitema, R. Mahy (2003). **"STUN - Simple Traversal of User Datagram Protocol (UDP)Through Network Address Translators (NATs)"**. Standards Track.

[RFC 4193] R. Hinden, B. Haberman (2005). **"Unique Local IPv6 Unicast Address"**. Standards Track.

[RFC 4213] Gilligan, E. Nordmark (2005). **"Basic Transition Mechanisms for IPv6 Hosts and Routers"**. Standards Track.

[RFC 4214] F. Templin, T. Gleeson, M. Talwar, D. Thaler (2005). **"Intra-Site**

Automatic Tunnel Addressing Protocol (ISATAP)". Experimental.

[RFC 4380] C. Huitema (2006). **"Teredo: Tunneling IPv6 over UDP through Network Address Translations (NATs)"**. Standards Track.

[RFC 4446] A. Conta, S. Deering, M. Gupta (2006). **"Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification"**. Standards Track.

[RFCMX Por-definir] A. Fernández, M Ricárdez (2007). **"Direccionamiento IPv6 para la red de Internet2 de México (RedCUDI)"**. Estándar.

[draft-blanchet-v6ops-tunnelbroker-tsp-03] M. Blanchet, F. Parent (2005). **"IPv6 Tunnel Broker with the Tunnel Setup Protocol (TSP)"**. Internet-Draft.

PRESENTACIONES

[1] **"Cambio en el Prefijo y nuevo Direccionamiento IPv6 en la RedCUDI"**. Reunión de Primavera CUDI 2005. Veracruz, Veracruz, México 29 de Abril de 2005. Ing. Azael Fernández Alcántara.

[2] **"Día Virtual CUDI sobre IPv6"**. Día Virtual CUDI sobre IPv6. Cd. de México, México 27 de febrero 2004. Ing. Azael Fernández Alcántara.

[3] **"Experiencia de CUDI con IPv6 / Situación Actual de IPv6 en la RedCLARA"**. Seminario sobre IPv6 y Políticas Publicas. Videoconferencia a Caracas, Venezuela 17 y 18 de mayo 2005. Ing. Azael Fernández Alcántara.

[4] **"Informe del Estado IPv6 en la RedCUDI"**. Grupo de trabajo IPv6 en CUDI. Reunión del CDR-CUDI Videoconferencia México, México 21 de noviembre de 2006. Mark Ricárdez Zárate.

[5] **"Renumeración IPv6 de Redes y Equipos"**. Cuarto Foro Latinoamericano de IPv6 (FLIP-6). LACNIC IX. Guatemala, Guatemala 24 de mayo de 2006. Ing. Azael Fernández Alcántara.

DOCUMENTOS

[1] **"IPv6 en la Universidad Nacional Autónoma de México (UNAM)"**. Azael Fernández Alcántara, César Olvera Morales. Laboratorio de Interoperabilidad de la UNAM, 1999-2001.

[2] **"Nombres DNS de los enlaces de los equipos de backbone de la RedCUDI"**. Azael Fernández Alcántara, Mark Ricárdez Zárate. Grupo de trabajo IPv6 en CUDI mayo de 2007.

[3] **"Plan de Renumeración IPv6 en el CORE de CUDI v1.1"**. Azael Fernández Alcántara, Mark Ricárdez Zárate. Grupo de trabajo IPv6 en CUDI marzo de 2006.

[4] http://www.ipv6.unam.mx/Internet2/**Procedimiento-Bloques-IPv6-v1.1.pdf.**

MANUALES

[1] Cisco IOS IPv6 Commando Reference. Corporate Headquarters. Cisco Systems, Inc., USA.

[2] Foundry NetIron XMR/MLX. Configuration Guide. Foundry Networks.

[3] The ABCs of IP Version 6. Casimir Sammanasu. Cisco IOS Learning Service.

[4] Manual de Usuario Quidway NetEngine16E/08E/05 Routers. Versión T2-080103-20041116-C-3.10.

INTERNET

http://www.iana.org
http://www.ietf.org
http://www.cudi.edu.mx
http://rfc.cudi.edu.mx/
http://www.tcpipguide.com/free/t_IPv6GlobalUnicastAddressFormat.htm
http://www.consulintel.es/html/ForoIPv6/foroipv6.htm
http://www.ipv6style.jp
http://www.cisco.com/en/US/tech/tk872/tsd_technology_support_protocol_home.html
http://tools.ietf.org/wg/v6ops/draft-palet-v6ops-6in4-vs-6over4-00.txt
http://www.eurescom.de/~public-webspace/P1000-series/P1009/doc2_3.html
http://www.ipv6-es.com/02/docs/alberto_lopez_2.pdf
http://ntrg.cs.tcd.ie/undergrad/4ba2.02/ipv6/interop.html
http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/routing.htm#xtocid15
http://dialogica.com.ar/clicsmodernos/archives/engelbart_a.htm
http://www.calasanz-pereira.edu.co/sistemas/tutoriales/internet/evolucion.htm
http://mundointernet.iespana.es/historia.htm
http://www.razonypalabra.org.mx/anteriores/n1/marzo.html
http://www.ati.es/DOCS/internet/histint/histint1.html
http://alepscor.ua.edu/i2i_gradprojoverview.htm
http://www.wise-intern.org/journal/1997/HEIDEMAN.PDF.
http://ipv6.internet2.edu/faq.shtml
http://www.geant2.net/server/show/nav.00d009001
http://www.ipv6.unam.mx/
http://www.noc.cudi.edu.mx/

http://www.internet2.unam.mx/cgi-bin/ve.cgi?m=AAAAAAABSX&a=F&p=AAAAAAAAJL
http://ciberhabitat.gob.mx/universidad/internet2/textos/texto_internet2.htm
http://es.wikipedia.org/wiki/Internet
http://www.ipv6.unam.mx/internet2.html
http://www.idg.es/iworld/articulo.asp?id=106554
http://www.itq.edu.mx/vidatec/espacio/aisc/ARTICULOS/i2/internet_2.htm
http://www.ipv6.org/v6-apps.html
http://6net.iif.hu/ipv6_apps/
http://www.join.uni-muenster.de/Implementationen/Software.php?lang=en#Email
http://www.deepspace6.net/docs/ipv6_status_page_apps.html#mbox
http://www.rnp.br/es/multicast/sobre.html
http://www.cudi.edu.mx/convocatorias/2005_convocatoria/OVS.pdf
http://www.microsoft.com/technet/network/ipv6/default.mspx
http://people.debian.org/~csmall/ipv6/
http://www.freebsd.org/doc/en_US.ISO8859-1/books/handbook/network-ipv6.html
http://www.freebsd.org/doc/en_US.ISO8859-1/books/handbook/kernelconfig-config.html
http://www.cisco.com/en/US/tech/tk872/tsd_technology_support_protocol_home.html

Anexos.

Habilitacion y configuración de IPv6

La habilitación y configuración de IPv6 en diferentes equipos es muy importante a la hora de poner en práctica al protocolo en ambientes de prueba o producción. El siguiente anexo tiene como finalidad, brindar algunos de los comandos y sintaxis utilizados con más frecuencia en diferentes equipos y plataformas.

A. WINDOWS[44]

IPv6 para Windows XP "sin" service packs fue instalado como un avance del desarrollo del stack. El protocolo IPv6 para Windows Server 2003, Windows XP con SP1, y Windows XP con SP2 es un protocolo de producción.

IPv6 puede ser manipulado de varias formas, una de ellas son por medio de comandos ipv6.exe y la otra por medio de netsh.

A.1. Instalación

A continuación se describe las formas en que puede ser habilitado el stack de IPv6 en Windows.

Iniciar sesión en una computadora que esté corriendo Windows XP con SP1 y SP2 con una cuenta de usuario que tenga privilegios para cambiar la configuración de red.

1. Para instalar IPv6 mediante comandos, desde el escritorio de Windows XP pulsar en Inicio > Programas > Accesorios > Símbolo del sistema. En la ventana de comandos del sistema, escriba:

c:/> ***ipv6 install***

o si prefiere: *c:/>* ***netsh interface ipv6 install***

o de forma gráfica:

1. Pulsar Inicio > Panel de Control, y dar doble click en "Conexiones de Red".
2. Pulsar con el botón derecho en cualquier conexión de área local y luego pulsar en "Propiedades".
3. Pulsar Instalar.
4. En el cuadro Seleccionar tipo de componente de red pulsar en Protocolo, y después en "Añadir".
5. En el cuadro "Seleccionar protocolo de Red", pulsar en "Microsoft TCP/IP versión 6", y después pulsar "OK".
6. Pulsar Cerrar para guardar los cambios en la configuración de la red.

A.2. Comandos

Los comandos es la forma màs comùn en la que se configura IPv6 en la mayoría de los equipos Windows, la tabla A.2-1 muestra los comandos que se pueden aplicar desde una ventana de símbolo de sistema, de forma directa (ipv6.exe) o por medio de Netsh.

[44] http://www.microsoft.com/technet/network/ipv6/default.mspx

Tabla A.2-1. Equivalencia de comandos entre ipv6.exe y Netsh.

Comandos en ipv6.exe	Equivalente en Netsh
ipv6 install	**netsh interface ipv6 install**
ipv6 uninstall	**netsh interface ipv6 uninstall**
ipv6 [**-v**] **if** [*IfIndex*]	**netsh interface ipv6 show interface** [[**interface**=]*String*] [[**level**=]{**normal** | **verbose**}] [[**store**=]{**active** | **persistent**}]
ipv6 ifcr v6v4*V4Src V4Dst* [**nd**] [**pmld**]	**netsh interface ipv6 add v6v4tunnel** [[**interface**=]*String*] [**localaddress**=]*IPv4Address* [**remoteaddress**=]*IPv4Address* [[**neighbordiscovery**=]{**enabled** | **disabled**}] [[**store**=]{active | **persistent**}]
ipv6 ifcr 6over4*V4Src*	**netsh interface ipv6 add 6over4tunnel** [[**interface**=]*String*] [**localaddress**=]*IPv4Address* [[**store**=]{**active** | **persistent**}]
ipv6 ifc*IfIndex* {[**forwards**] | [**-forwards**]} {[**advertises**] | [**-advertises**]} [**mtu***#Bytes*] [**site** *SiteIdentifier*]	**netsh interface ipv6 set interface** [[**interface**=]*String*] [[**forwarding**=]{**enabled** | **disabled**}] [[**advertise**=]{**enabled** | **disabled**}] [[**mtu**=]*Integer*] [[**siteid**=]*Integer*] [[**metric**=]*Integer*] [[**store**=]{**active** | **persistent**}]
ipv6 ifd*IfIndex*	**netsh interface ipv6 delete interface** [[**interface**=]*String*] [[**store**=]{**active** | **persistent**}]
ipv6 adu*IfIndex/Address* [**life** *ValidLifetime*[*/PrefLifetime*]] [**anycast**] [**unicast**]	**netsh interface ipv6 add address** [[**interface**=]*String*] [**address**=]*IPv6Address* [[**type**=]{**unicast** | **anycast**}] [[**validlifetime**=]{*Integer* | **infinite**}] [[**preferredlifetime**=]{*Integer* | **infinite**}] [[**store**=]{**active** | **persistent**}]
ipv6 nc [*IfIndex* [*Address*]]	**netsh interface ipv6 show neighbors** [[**interface**=]*String*] [[**address**=]*IPv6Address*]
ipv6 ncf [*IfIndex* [*Address*]]	**netsh interface ipv6 delete neighbors** [[**interface**=]*String*] [[**address**=]*IPv6Address*]
ipv6 rc [*IfIndex* [*Address*]]	**netsh interface ipv6 show destinationcache** [[**interface**=]*String*] [[**address**=]*IPv6Address*]
ipv6 rcf [*IfIndex* [*Address*]]	**netsh interface ipv6 delete destinationcache** [[**interface**=]*String*] [[**address**=]*IPv6Address*]
ipv6 bc	**netsh interface ipv6 show bindingcacheentries**
ipv6 [-v] **rt**	**netsh interface ipv6 show routes** [[**level**=]{**normal** | **verbose**}] [[**store**=]{**active** | **persistent**}]
ipv6 rtu*Prefix IfIndex*[*/Address*]	**netsh interface ipv6 add route**

Comandos en ipv6.exe	Equivalente en Netsh
[**lifetime***Valid*[*/Preferred*]] [**preference** *P*] [**publish**] [**age**] [**spl***SitePrefixLength*]	[**prefix**=]*IPv6Address/Integer* [[**interface**=]*String*] [[**nexthop**=]*IPv6Address*] [[**siteprefixlength**=]*Integer*] [[**metric**=]*Integer*] [[**publish**=]{**no** | **yes** | **immortal**}] [[**validlifetime**=]{*Integer* | **infinite**}] [[**preferredlifetime**=]{*Integer* | **infinite**}] [[**store**=]{**active** | **persistent**}]
ipv6 spt	**netsh interface ipv6 show siteprefixes**
ipv6 spu*Prefix IfIndex* [**life** L]	**netsh interface ipv6 add route** [**prefix**=]*IPv6Address/Integer* [[**siteprefixlength**=]*Integer*] [[**store**=]{**active** | **persistent**}]
ipv6 gp	**netsh interface ipv6 show global** [[**store**=]{**active** | **persistent**}]
ipv6 [**-p**] **gpu DefaultCurHopLimit** *Hops*	**netsh interface ipv6 set global** [[**defaultcurhoplimit**=]*Integer*] [[**store**=]{**active** | **persistent**}]
ipv6 [**-p**] **gpu UseAnonymousAddresses** [**yes**|**no**|**always**|*Counter*]	**netsh interface ipv6 set privacy** [[**state**=]{**enabled** | **disabled**}] [[**store**=]{**active** | **persistent**}]
ipv6 [**-p**] **gpu MaxAnonDADAttempts** *Number*	**netsh interface ipv6 set privacy** [[**maxdadattempts**=]*Integer*] [[**store**=]{**active** | **persistent**}]
ipv6 [**-p**] **gpu MaxAnonLifetime** *Valid*[*/Preferred*]	**netsh interface ipv6 set privacy** [[**maxvalidlifetime**=]*Integer*] [[**maxpreferredlifetime**=]*Integer*] [[**store**=]{**active** | **persistent**}]
ipv6 [**-p**] **gpu AnonRegenerateTime** *Time*	**netsh interface ipv6 set privacy** [[**regeneratetime**=]*Integer*] [[**store**=]{**active>** | **persistent**}]
ipv6 [**-p**] **gpu MaxAnonRandomTime** *Time*	**netsh interface ipv6 set privacy** [[**maxrandomtime**=]*Integer*] [[**store**=]{**active** | **persistent**}]
ipv6 [**-p**] **gpu AnonRandomTime** *Time*	**netsh interface ipv6 set privacy** [[**randomtime**=]*Integer*] [[**store**=]{**active** | **persistent**}]
ipv6 [**-p**] **gpu NeighborCacheLimit** *Number*	**netsh interface ipv6 set global** [**neighborcachelimit**=]*Integer* [[**store**=]{**active** | **persistent**}]
ipv6 [**-p**] **gpu RouteCacheLimit** *Number*	**netsh interface ipv6 set global** [[**routecachelimit**=]*Integer*] [[**store**=]{**active** | **persistent**}]
ipv6 ppt	**netsh interface ipv6 show prefixpolicy** [[**store**=]{**active** | **persistent**}]
ipv6 ppu*Prefix***precedence** *PrecedenceValue***srclabel** *SourceLabelValue* [**dstlabel** *DestinationLabelValue*]	**netsh interface ipv6 add prefixpolicy** [[**maxvalidlifetime**=]*Integer*] [[**maxpreferredlifetime**=]*Integer*] [[**store**=]{**active** | **persistent**}]
ipv6 renew [*IfIndex*]	**Netsh interface ipv6 renew** [[**interface**=]*String*]

B. LINUX[45]

B.1. Instalación

El soporte IPv6 en Linux está incluido desde la versión 2.2. Como primer paso, comprobar si el kernel tiene soporte IPv6, escribiendo en el shell de Linux:

test -f /proc/net/if_inet6 && echo "El kernel actual soporta IPv6"

Si el kernel no soporta IPv6 se tiene que recompilar. Para mayor información consultar: http://www.bieringer.de/linux/IPv6/IPv6-HOWTO/IPv6-HOWTO-2.html#kernel

El siguiente paso será verificar que el módulo IPv6 esté cargado, escribiendo:

lsmod |grep -w 'ipv6' && echo "Módulo IPv6 satisfactoriamente cargado"

Si no esta cargado el módulo IPv6 ejecutar:

modprobe ipv6

Posteriormente comprobar la carga del mismo y que se active por defecto en el siguiente reinicio del equipo, consultando que en el archivo /etc/modules.conf esté configurada la siguiente línea:

alias net-pf-10 ipv6

Si se requiere desactivar el módulo, cambiar la línea anterior por la siguiente:

alias net-pf-10 off

De esta manera para ver la asignación de direcciones link-local en cualquier interfaz se hace de la siguiente manera:

/sbin/ip -6 addr show dev <interfaz> scope <link, site, global>

B.2. Instalar utilidades en debian para ipv6

Para hacer uso de herramientas IPv6 para probar la conectividad, instalar lo siguiente:

 # apt-get install iputils-ping
 # apt-get install iputils-tracepath

Para configurar la máquina como gateway, para proveer conectividad ipv6 a las máquinas de la red local, instalar el paquete:

[45] http://people.debian.org/~csmall/ipv6/

 # apt-get install radvd

B.3. Configuración manual

Para configurar manualmente direcciones IPv6 usar:

/sbin/ip -6 addr add <dirección IPv6>/<longitud del prefijo> dev <interfaz>

o

/sbin/ifconfig <interfaz> inet6 add <dirección IPv6>/<longitud del prefijo>

Cache de vecinos: *ip -6 neigh show*

Ruta de vecinos IPv6: *route –A inet6/netstat*

B.3.1. Isatap

Para configurar un cliente ISATAP en Linux, usar:

 # /sbin/ip tunnel add is0 mode isatap local <local IPv4 address> v4any <Server IPv4 address> ttl 64

 # /sbin/ip link set is0 up

B.3.2. 6to4

El siguiente script bash, ayuda a calcular direcciones ipv6 6to4 a partir de direcciones ipv4:

printf "2002:%02x%02x:%02x%02x::\n" a b c d

Sustituyendo a,b,c,d, por los bytes de la dirección ipv4 pública.

Para configurar un cliente 6to4, hacer lo siguiente:

 # ip tunnel add SeisACuatro mode sit ttl 0 remote any local a.b.c.d
 # ip link set dev SeisACuatro up
 # ip -6 addr add 2002:d53c:6668::/16 dev SeisACuatro
 # ip -6 route add 2000::/3 via ::192.88.99.1 dev SeisACuatro metric 1

Donde: a.b.c.d es la dirección IPv4 pública del host
2002:d53c:6668::/16 es la dirección obtenida del script aplicado anteriormente
::192.88.99.1 es la dirección del enrutador 6to4

C. FREEBSD[46]

El soporte de IPv6 está incluido en FreeBSD desde la versión 4.0.

C.1. Instalación

El soporte IPv6 está basado en trabajos hechos por el grupo KAME. En FreeBSD se instala por defecto el soporte IPv6. Se puede cambiar el kernel y habilitar más opciones de IPv6 como las que se encuentran en el archivo /sys/i386/conf/LINT (este archivo contiene una descripción de todas las opciones de configuración para el kernel)[47]. En versiones posteriores construir el archivo que contenga todas las opciones disponibles con:

 # cd /usr/src/sys/i386/conf && make LINT

Si por alguna razón IPv6 no se encuentra, se necesitará recompilar el kernel agregando las opciones en la línea INET6 en la configuración del mismo, como se presenta en la siguiente tabla:

Tabla C.1-1. Opciones de INET6

Opción	**INET6**		**# Protocolos de Comunicación IPv6**
pseudo-device	gif	4	# Al hacer un túnel IPv6 e IPv4
pseudo-device	faith	1	# Retransmisión IPv6-e-Ipv4 (traducción)
pseudo-device	stf	1	# Encapsulamiento IPv6 sobre Ipv4 (6to4)

La configuración por defecto del kernel está en el archivo /sys/i386/conf/GENERIC y se configura por defecto con todas las opciones anteriores, excepto el encapsulamiento 6to4 para habilitar esta opción se puede consultar (http://www.freebsd.org/cgi/ports.cgi?query=6to4&stype=all).

C.2. Auto-Configuración

Para obtener direcciones IPv6 por Auto-Configuración, simplemente seguir los siguientes pasos.

1.- Poner en el archivo /etc/rc.conf la siguiente línea: ipv6_enable="YES"

2.- Insertar cualquier información de host local o externo (si no se esta usando DNS) en el archivo /etc/hosts similar al siguiente ejemplo:

Tabla C.2-1. Opciones de configuración en /etc/hosts

[46] http://www.freebsd.org/doc/en_US.ISO8859-1/books/handbook/network-ipv6.html

[47] Pete Loshing, "IPv6 Theory, Protocol, and Practice", Morgan Kaufmann Series, 2nd Edition.

http://www.freebsd.org/doc/en_US.ISO8859-1/books/handbook/kernelconfig-config.html

127.0.0.1	localhost.dominio.com localhost	# dirección loopback IPv4
192.168.25.5	host.dominio.com host	# dirección IPv4 del host local
: : 1	Localhost	# dirección loopback IPv6
3ffe:80f0:1:1:201:2ff:fe00:2112	Nombre del host	# entrada para un host externo

3.- Finalmente, reiniciar la máquina y ejecutar el comando ifconfig <"Interfaz"> para comprobar que se están obteniendo direcciones IPv6.

C.3. Configuración manual

El procedimiento es similar. El primer paso consiste en poner en el archivo /etc/rc.conf lo siguiente:

ipv6_enable = "YES"
ipv6_network_interfaces = <"nombre de la interfaz">
ipv6_ifconfig_<"nombre de la interfaz"> = <"dirección IPv6"> prefixlen <"longitud del prefijo">
ipv6_defaultrouter = <"dirección IPv6 del router">%<"nombre de la interfaz">

C.4. Configuración de Rutas Estáticas

 # route add –inet6 default fe80::X:X:X:X%interface
 # route add –inet6 default X:X:X:X::X (Si dirección global)
 # route add –inet6 X:X:X:X:: -prefixlen YY X:X:X:X::X
 # route add –inet6 X:X:X:X:: -prefixlen YY fe80::X:X:X:X%interface

D. CISCO[48]

El soporte de IPv6 en el software de Cisco apareció por primera vez en versiones como la 11.3X del IOS como versión de prueba, y a partir de la versión 12.2T en forma más estable. Actualmente las características son soportadas en las versiones 12.0S, 12.xT, 12.2S, 12.2SB, 12.3, y 12.4 del IOS.

Para habilitar IPv6 en el enrutador de forma global ejecutar lo siguiente, con privilegios de administrador:

(config) # ***ipv6 unicast-routing***

[48] http://www.cisco.com/en/US/tech/tk872/tsd_technology_support_protocol_home.html

D.1. Configuración manual

Para configurar manualmente una dirección IPv6 en cualquier interfaz proceder de la siguiente manera:

Primero ingresar al modo de configuración con el comando "configure terminal". Después entrar ala interfaz deseada con:

(config)# ***interface*** <"interfaz">

y ejecutar:

(config-if)# ***ipv6 address*** <"dirección IPv6">/<"longitud del prefijo">

D.2. Configuración de RIPng

Para habilitar RIP en el enrutador, después de haber habilitado IPv6 globalmente, se necesita iniciar un proceso de Routing RIP IPv6, identificándolo con un nombre, en el modo de configuración.

(config)# ***ipv6 router rip*** NombreProceso

Para habilitar el proceso de RIPng sobre una interfaz específica, entrar en el modo de configuración de interfaz y habilitar el proceso de RIPng.

(config-if)# ***ipv6 rip*** NombreProceso ***enable***

D.3. Configuración de OSPFv3

Para habilitar un proceso de OSPFv3 en el enrutador, dentro del modo de configuración del enrutador aplicar:

(config)# ***ipv6 router ospf*** ID-Proceso

OSPFv3 utiliza un valor de 32 bits representado por una dirección IPv4, como ID-Router. En un enrutador habilitado solamente con IPv6, dentro del proceso OSPFv3 especificado, aplicar:

*(config-if)#****router-id*** xx.xx.xx.xx

Para habilitar OSPFv3 en una interfaz dada, meterse en el modo de configuración de la interfaz y aplicar:

(config-if)# ***ipv6 ospf*** ID-Proceso ***area*** ID-Area

D.4. Configuración de ISIS

Para habilitar un proceso de ISIS en el enrutador, dentro del modo de configuración del enrutador aplicar:

(config)# ***router isis*** Etiqueta-Area

Para configurar la identidad de red ISIS (NET), dentro del proceso de routing ISIS, aplicar:

(config-router)# ***net*** Titulo-Identidad-Red

Para habilitar el proceso de routing ISIS en una interfaz, dentro del modo de configuración de interfaz, aplicar:

(config-if)# ***ipv6 router isis*** Etiqueta-Area

D.5. Configuración de BGP4+

Para habilitar un proceso de BGP4+ en el enrutador, dentro del modo de configuración del enrutador aplicar:

(config)# ***router bgp*** AS-Numero

BGP usa un ID de Router para identificar a los peer´s hablando BGP. Frecuentemente un ID Router BGP es una valor de 32 bit representado por una dirección IPv4. Un enrutador habilitado únicamente con IPv6, se debe configurar manualmente el ID Router BGP. Dentro del modo de configuración del proceso de routing especificado, aplicar:

(config-router)# ***bgp router-id*** xx.xx.xx.xx

Para habilitar el intercambio de prefijos entre dos AS, se tiene que establecer peer entre dos enrutadores de los AS´s. Por default al establecer peering, sólo se intercambian prefijos IPv4 unicast. Dentro del modo de configuración del proceso routing especificado, aplicar:

(config-router)# ***neighbor*** {*ip-address* | *ipv6-address* | *peer-group-name*} ***remote-as*** AS-Numero

Para poder ingresar a la familia IPv6 unicast dentro del número de proceso BGP especificado, y establecer el intercambio de prefijos IPv6 y anunciar prefijos IPv6, aplicar:

(config-router)# ***address-family ipv6*** [unicast | multicast]

Para habilitar el intercambio de prefijos IPv6 con un peer especificado, aplicar:

(config-router-af)# ***neighbor*** {*ip-address* | *ipv6-address* | *peer-group-name*} ***activate***

Para poder anunciar o inyectar un prefijo IPv6 unicast dentro de la base de datos BGP para la familia IPv6 unicast, dentro de la familia IPv6 unicast, aplicar:

(config-router-af)# ***network*** Numero-Red/prefijo