Universidad del Azuay

Facultad de Ciencia y Tecnología

Escuela de Ingeniería Electrónica

"Análisis del protocolo de Internet versión 6 e implementación en una red de área local en el laboratorio de redes de la Facultad"

Trabajo de graduación previo a la obtención del título de Ingeniero Electrónico

Autores:
Roberto José Cobos Delgado
Cristóbal Jaramillo Jaramillo

Director:
Ing. Leopoldo Vázquez Rodríguez

Cuenca, Ecuador

2007

Dedico este trabajo a mi familia, especialmente a mis padres.
Cristóbal Jaramillo J.

A mi esposa y a mi hijo.
Roberto Cobos D.

A todas las personas que nos ayudaron y guiaron durante la realización de este trabajo, entre los principales: Lcdo. Leopoldo Vázquez Rodríguez, Ing. Jordi Palet Martínez, Ing. Luís Espinoza, Ing. Juan Pablo León. A nuestras familias por el apoyo incondicional brindado a lo largo de estos años.

RESUMEN

Este trabajo trata sobre el protocolo IPV6, especialmente en lo relacionado con capacidad de direccionamiento, características de la cabecera, seguridad en el control de accesos y encriptamiento, calidad de servicio y aplicaciones móviles.

En el desarrollo del trabajo, se presenta una descripción del nuevo protocolo, destacando las razones por las cuales ha sido desarrollado y es necesaria su implantación.

En la parte central de esta tesis se trata sobre el proceso de configuración e implementación de una red bajo el protocolo IPV6. Esto se lo ha hecho mediante un *router virtual* desarrollado sobre una plataforma Linux®.

Finalmente, se recoge las conclusiones y recomendaciones resultantes del análisis y pruebas efectuados.

ABSTRACT

This work is related to the IPV6 protocol, especially in its addressing capacity, header characteristics, access control security and encryption, quality of service and mobile applications.

In the development of this work a description of the new protocol is presented, high lighting the reasons to develop it and why it is necessary its implementation.

In the main part of this Thesis, the configuration and implementation of a network under this protocol is described. A virtual router has been developed over a Linux® platform for testing the network.

Finally, the conclusions and recommendations produced by the analysis and tests made are presented.

INDICE DE CONTENIDOS

CAPÍTULO 3: DIRECCIONAMIENTO EN IPV6

CAPÍTULO 4: SEGURIDAD Y CALIDAD DE SERVICIO

CAPÍTULO 5: IMPLEMENTACIÓN DE LA RED IPV6

Cobos Delgado Roberto José
Jaramillo Jaramillo Fabián Cristóbal
Trabajo de Graduación
Lcdo. Leopoldo Vázquez Rodríguez
Diciembre del 2007

"Análisis del protocolo de Internet versión 6 e implementación en una red de área local en el laboratorio de redes de la Facultad"

RESEÑA HISTORICA DE IP

1.1 Reseña de IPV4

IPV4 es la Versión 4 del protocolo IP (Internet Protocol). Es el estándar actual de Internet para identificar dispositivos conectados a una red. Utiliza direcciones IP de 32 bits, lo cual limita la cantidad de direcciones a 4.294.967.296 (2 elevado a 32). Esto crea un evidente problema, la escasez de direcciones en el futuro.

IPV4 fue desarrollada por los años 70 para facilitar la comunicación y el compartimiento de la información entre investigadores del gobierno y académicos de los EEUU. En ese entonces el sistema fue limitado con escasos puntos de acceso, consecuentemente los desarrolladores no tuvieron la visión de requerimientos de seguridad y calidad de servicio. IPV4 ha sobrevivido más de 30 años y ha sido una parte integral de la revolución de Internet pero aun así la era de los diseños de sistemas más inteligentes requiere de mayores necesidades que IPV4 no les ofrece o los limita.

1.2 ¿Qué es IPV6?

Es la nueva versión del Protocolo de Internet, siendo una evolución de IPV4 utilizado ampliamente en redes de última tecnología y con gran aplicación especialmente en instituciones académicas. Soluciona el problema actual de direccionamiento debido a la ampliación de 32 a 128 bits en la dirección de red, o sea de 2^32 direcciones (4.294.967.296) a 2^128 direcciones (3.402823669 e38, o sea sobre 1.000 sixtillones).

El protocolo es instalado como una actualización de software en la mayoría de dispositivos y sistemas operativos. En caso de actualizaciones de hardware y sistemas operativos, usualmente ya se contempla la utilización de IPV6 y solo se necesita de una activación o configuración. Actualmente los mecanismos de transición permiten una introducción paso a paso de IPV6, sin poner en riesgo la infraestructura de IPV4.

IPV6 ha sido desarrollado basándose en una amplia experiencia de los creadores y usuarios de IPV4. Demuestra y establece mecanismos que han sido retenidos, superando limitaciones conocidas, y extendiendo la flexibilidad. Es además un protocolo diseñado para manejar la tasa de crecimiento de Internet y para enfrentarse con requerimientos de demanda en servicios, movilidad y seguridad de punto a punto.

El Grupo de Trabajo en Ingeniería de Internet (IETF) fue quien empezó a desarrollar un sucesor de el protocolo IPV4 al inicio de los años 90 al mismo tiempo que se realizaban esfuerzos paralelos para resolver los problemas de cantidad de direcciones IPs disponibles. En el año de 1993 la IETF inicio el Protocolo de Internet de Nueva Generación (IPNG) con el objetivo de definir los diferentes propósitos y fijar recomendaciones para procedimientos futuros.

1.3 ¿Por qué IPV6?

Por razones históricas, las organizaciones y agencias de gobierno en los Estados Unidos, utilizan aproximadamente el 60% del direccionamiento de IPV4, el restante 40% es compartido por el resto del mundo. De los 6.4 billones de personas en el mundo según información proporcionada en el año 2006, aproximadamente 330 millones viven en Norteamérica, 807 millones en Europa y 3.6 billones en Asia, esto significa que el 5% de la población del mundo que vive en Estados Unidos tiene el 60% de espacio de direccionamiento asignado. De los 3.6 billones de personas que viven en Asia aproximadamente 364 millones tienen acceso a Internet, con un crecimiento exponencial. Esto explica porque el desarrollo de IPV6 en Asia es más común que en Europa y EEUU.

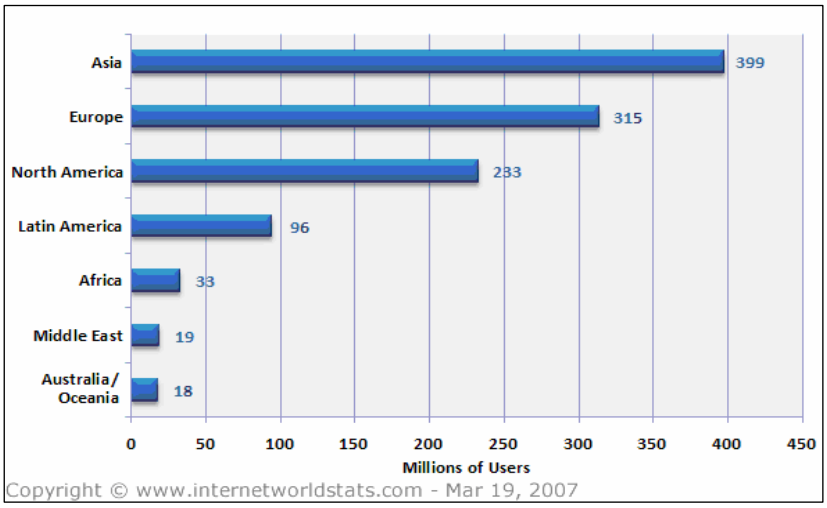


Fig.1.1 Usuarios de Internet en el Mundo

Existe la necesidad de considerar una adopción de IPV6, considerando que su desarrollo siga en marcha para estar a la par de la madurez con la que cuenta IPV4. Los aspectos aun no desarrollados podrían contemplarse en los próximos años, de igual manera que ocurrió con IPV4.

Algunas empresas no encuentran suficientes razones para adoptar esta nueva tecnología, sin embargo esto es muy importante para organizaciones que prestan atención para la introducción de la misma, ya que es algo inevitable a través del tiempo.

Se podrían considerar los siguientes como los principales cambios:

1.3.1 Espacio de dirección extendido

El formato de la dirección es extendido de de 32 bits a 128 bits. Esto proporciona una dirección de IP para cada grano de arena del planeta. Además, tiene en cuenta también una estructuración jerárquica de los espacios de dirección en favor de un ruteo global optimizado.

1.3.2 Autoconfiguración

Tal vez la nueva característica más intrigante de IPV6 es el mecanismo de autoconfiguración. Cuando un dispositivo arranca en el mundo de IPV6, se activa y solicita un prefijo de red, a este se le puede asignar uno o más prefijos de red

desde un router IPV6. Usando la información del prefijo el dispositivo puede configurarse para una o mas direcciones globales válidas IPs usando también su dirección MAC o un numero aleatorio privado para asignar una dirección única IP.

En el mundo IPV4 se debe asignar una dirección IP a cada dispositivo, esto puede ser una configuración manual o mediante DHCP.

La autoconfiguración simplifica la tarea de los administradores de red y reduce substancialmente el costo del mantenimiento de redes IP. Consideremos el caso de un hogar en el futuro donde se necesitará una dirección IP para cada dispositivo, lo cual vendría a ser indispensable, asumiendo por ejemplo la compra de un televisor que necesite la asignación de una dirección IP dentro de la red, lo cual mediante DHCP se necesitaría hacer una nueva configuración desde el servidor, mientras que con IPV6 se lograría hacerlo de forma automática. La autoconfiguración permite también la conexión de dispositivos móviles, tales como teléfonos celulares o portátiles cuando ingresen a redes vecinas.

1.3.3 Simplificación en el formato de la cabecera

La cabecera en IPV6 es más simplificada que la de IPV4 y tiene una extensión formada por 40 bytes, esta permite un procesamiento rápido, es básicamente acomodada en dos partes, 16 bytes para las direcciones de fuente y destino y 8 bytes para la información general de la cabecera.

1.3.4 Soporte mejorado para opciones y extensiones

IPV4 integra las opciones en la cabecera base, en tanto que IPV6 lleva las opciones en las llamadas extensiones de cabecera, las cuales son insertadas solo si son necesarias, esto permite un procesamiento rápido de los paquetes. Las especificaciones base describen un grupo de 6 extensiones de cabecera, incluyendo cabeceras para ruteo, IPV6 móvil, calidad de servicio y seguridad.

1.3.5 ¿Por qué se precisa de IPV6?

La cantidad de direcciones IPV4 tiene un límite teórico de 4.3 billones. Sin embargo, los métodos iniciales de distribución asignaron las direcciones ineficazmente. Consecuentemente, algunas organizaciones obtuvieron bloques de

direcciones más grandes que lo que necesitaban, y las direcciones que podrían ser utilizadas en otros lugares no están disponibles. Si fuese posible reasignar el espacio de direcciones IPV4, podría ser utilizado más efectivamente, pero este proceso no es posible, y una reasignación global y renumeración simplemente no es práctico. Se debe estar enterado de la realidad actual, como que el espacio de direcciones IPV4 se acerca al agotamiento, sólo cerca del 14% de la población de mundo tiene el Acceso a Internet. Si se quiere proporcionar el acceso a Internet a sólo 20% de la población del mundo, se necesitaría de direcciones IPV6. Y este cálculo no considera que en el futuro se necesitara billones de direcciones de IP para dispositivos. Los vendedores en todas las industrias han desarrollado monitoreo, control, y sistemas de administración basados en IP.

El direccionamiento extendido y la restauración del modelo original de punta a punta de Internet permite la eliminación de la traducción de direcciones de red (NAT), en el cual pocas direcciones públicas de IPV4 son usadas para conectar al Internet un gran número de usuarios con direcciones privadas, mediante un mapeo de direcciones internas a direcciones públicas. NAT fue introducido como una solución temporal a la limitación de direcciones en IPV4. Este servicio es muy común en redes en IPV4, lo cual crea serios inconvenientes en el manejo y operación: en el orden de hacer mapeo de direcciones, NAT modifica la dirección del nodo final de la cabecera IP, con frecuencia la aplicación de niveles de Gateways (ALG) son usados en conjunto con NAT para proporcionar un nivel de aplicación transparente. Hay una lista larga de aplicaciones y protocolos que presentan problemas cuando se usa un ambiente NAT, IPSec y las aplicaciones punto a punto son dos ejemplos conocidos. La ampliación del limitado espacio de direccionamiento (principal beneficio del NAT) no se justifica en IPV6, por lo tanto no se aplica en el.

Los índices de penetración de banda ancha en países como Corea del sur, Japón, Alemania, Francia y en EEUU, continúa acelerado y en muchos de los casos alcanzan el 65% o más. Este nivel de conexión siempre activa con una capacidad de ancho de banda sustancial (comparado con el servicio de dial-up) representa una gran oportunidad para que los dispositivos sean conectados al Internet, siendo los fabricantes de productos electrónicos los más beneficiados, por ejemplo las consolas de videojuegos con opción a partidas en línea. En Japón algunos proveedores de telecomunicaciones brindan servicios de televisión (películas, contenido de audio, etc.) sobre redes IP, además de aplicaciones de domótica en

las cuales refrigeradoras, estufas, calentadores de agua y bañeras, son conectados para facilitar tareas como control de encendido, control remoto, reparaciones (soporte en línea) y para propósitos telemetría y monitoreo. Al final el resultado de este proceso de habilitación de redes es una gran cantidad de dispositivos que necesitan direccionamiento, muchos de los cuales no tendrán un estándar de interfaz de usuario. En estos casos las direcciones de IPV6 son acopladas con las características de autoconfiguración e IPV6 móvil, esto ayudará a los usuarios en una nueva era computarizada en sus hogares.

El crecimiento de industrias wireless (celular y redes inalámbricas basadas en protocolos 802.11x, 802.16, 802.20, UMTS, UWB, MIMO, etc.) ha sido una revolución en este campo. Actualmente en algunos países como Italia y Gran Bretaña, el número de teléfonos celulares excede el número de personas. En este mundo de continua accesibilidad y confiabilidad en la capacidad de acceso a la información en cualquier momento, los requerimientos de movilidad para usuario final vienen a ser de fundamental importancia. Desde el punto de vista de servidores de telecomunicaciones, especialmente de quienes soportan acceso de tipo multimedia (3G, WiMax), proveen IP como un método de transporte y encaminamiento para la información.

Teléfonos celulares y PDAs, ya pueden conseguir acceso al Internet, juegos con otros usuarios, realizar llamadas, transferencia y reproducción de video. En lugar de soportar todas estas aplicaciones que utilizan protocolos diferentes de transporte y creación de aplicaciones intermedias para facilitar la comunicación, es más eficiente proveer la infraestructura existente de red del Internet y una red de la compañía. Posteriormente se verá desde una perspectiva técnica, que el IPV6 Móvil es muy elegante en su diseño, sosteniendo a usuarios de móvil en una manera altamente eficiente y proporcionando mecanismos sumamente eficaces de cobertura para usuarios que mantienen estas conexiones cuando atraviesan entre distintas redes, incluso si las redes no utilizan el mismo tipo de medio de acceso.

Por muchas de estas razones, la mayor parte del mundo de la tecnología está ya adoptando IPV6, en el caso de China existe una gran inversión de millones de dólares para el desarrollo de una nueva red basada principalmente en este protocolo. La Unión Europea ha gastado también millones para la investigación y desarrollo tecnológico del “backbone” IPV6 y servicios innovadores que proveen muchas de las características beneficiosas de IPV6. Por su parte la India con una

clase media de crecimiento y una fuerte presencia en el mundo de las comunicaciones por Internet ha demostrado un interés sustancial por el desarrollo y el uso de IPV6. En Junio del 2003 y posteriormente en Julio del 2005 el gobierno de los Estados Unidos adoptó la utilización de IPV6. En otros países tales como Australia, Taiwán, Singapur, Inglaterra, Egipto, también presentan gran interés por esta tecnología.

Existen todavía algunas incógnitas acerca del valor de IPV6 para las empresas, y vale la pena conceder a cada organización la necesidad de evaluar cuidadosamente los beneficios del protocolo para su uso interno y determinar el mejor momento para su utilización. En algunas instancias, las organizaciones pueden encontrar las formas más adecuadas para utilizar IPV6 con el fin de resolver asuntos críticos sin una migración por completo de sus redes. La adopción puede ocurrir en un momento de expansión de la red, con un plan que minimice el impacto de la integración, pero que también asegure que todo esté listo en el momento de poner en marcha sin poner la infraestructura IPV4 actual en riesgo.

Con estas nuevas estructuras y extensiones IPV6 provee la introducción de servicios de nueva generación. Existirán dispositivos y servicios en el mercado en un futuro cercano que no podrán ser desarrollados con IPV4, esta nueva tendencia de mercado y oportunidades de negocio para vendedores y proveedores de servicios. Las primeras oportunidades son sustanciales como las oportunidades de adaptar los productos actuales con una actualización de tecnología IPV6.

Cuando consideramos todas estas ventajas, surgiría la pregunta “¿Por qué no usar IPV6?”, el principal problema son los conceptos errados que tiene la mayoría de personas acerca de IPV6, como por ejemplo:

- “La introducción de IPV6 pone en riesgo la infraestructura y servicios actuales”: El principal objetivo en el desarrollo de IPV6 fue crear un mecanismo de integración que permita la coexistencia de ambos protocolos.

- “El protocolo IPV6 es inmaduro y no probado, su límite de tiempo ha sido superado, o si este es capaz de manejar los requerimientos”: Esto es parcialmente verdad, ya que la implementación se ha realizado básicamente en routers y sistemas operativos aproximadamente por una década y ha sido probado y optimizado extensivamente.

- “El costo de introducción de IPV6 es muy alto”: Todo cambio en tecnología implica un costo, tanto en hardware como en software, las organizaciones necesitarán capacitarse para el manejo de esta tecnología. Se tiene que considerar que las redes en IPV4 continuamente se vuelven más complejas debido a que deben adaptarse a nuevos servicios (VoIP, mensajería instantánea, video conferencia, IPTV)

- “Con la autoconfiguración creemos que no es capaz de controlar o monitorear el acceso a la red”: Esta frase es aplicable para redes que ampliamente utilizan la autoconfiguración, existen dos maneras de controlar esta situación: “Stateful” y “Stateless”, en el caso del modo “stateful” servirá para las redes que utilicen DHCP (IPV4), y en modo “stateless” funcionarán con DHCPV6 (IPV6)

- “Nuestro ISP no nos ofrece servicios de IPV6”: No se necesita que el ISP provea de IPV6 en su red corporativa o privada, existen mecanismo de transición que permiten la utilización de IPV6 sobre la infraestructura IPV4 de un ISP.

- “Sería muy costoso y complicado el mantenimiento del “backbone”: No existe un orden establecido de la actualización de los dispositivos, se podría aprovechar el cambio paulatino de dispositivos (vida útil) para implementar dispositivos que soporten IPV6.

- “Sería muy costoso y complicado la migración de todas las aplicaciones a IPV6”: Los esfuerzos necesarios para migrar las aplicaciones sobre IPV6 son a menudo menos de lo esperado. Si una aplicación está correctamente diseñada, se ejecutará sobre IPV6 sin ninguna modificación. Para aplicaciones que necesitan modificaciones y no están disponibles, se puede optar por una red de doble pila en la cual se use IPV4 para aplicaciones de acceso IPV4 y en IPV6 para aplicaciones de acceso IPV6.

- “Nosotros tenemos suficientes direcciones IPV4, no necesitamos IPV6”: Ignorar IPV6 asumiendo que se tienen suficientes direcciones en IPV4, sería un grave error, ya que se crearía un aislamiento con las redes actuales.

1.3.6 Cuando es el momento indicado para IPV6?

Actualmente la tendencia mundial va hacia la migración a IPV6, de no considerar esta transición se podría tener una exclusión del mundo global de las comunicaciones en un futuro.

Como principales indicadores de un momento adecuado para considerar un cambio a un sistema soportado por IPV6, podemos considerar los siguientes:

- Cuando se necesite realizar una ampliación o utilizar NAT.
- Cuando se llega a saturar el espacio de direccionamiento.
- Cuando se desea configurar la red para aplicaciones que estén basadas en características avanzadas de IPV6.
- Cuando se necesita seguridad para una conexión punto a punto de un gran número de usuarios, considerando el problema que representa las implementaciones de NAT.
- Cuando se reemplaza hardware o aplicaciones que han cumplido su ciclo de vida, asegurándose que los nuevos productos soporte IPV6.
- Cuando se desea que la red cuente con este protocolo.

Se debe tener en cuenta las siguientes precauciones para una correcta preparación para IPV6:

Capacitación al personal interno, técnicos de red y creación de una red de prueba.

Crear escenarios de integración basados en su red y requerimientos.

Considerar como punto fundamental el soporte técnico de IPV6 al momento de adquirir nuevo hardware y software.

1.4 Situación Actual de IPV6

1.4.1 Asia

IPV6 en este continente es una realidad. La alta población y el acelerado crecimiento de Internet combinado con la limitación de direccionamiento IPV4 no permiten otra opción.

Japón fue uno de los primeros países en tomar el liderato. En marzo del 2001, publicaron “e-Japan Priority Policy Program”, anunciando la construcción de una gran red IPV6.

Japón es una sala de muestras para vendedores con dispositivos que soportan IPV6. Sony, por ejemplo anunció que en un futuro cercano todos sus dispositivos contarán con un funcionamiento IPV6.

Podemos encontrar ya disponibles refrigeradoras y microondas con ruteo y soporte de IPV6 incluido. Se pueden manejar estos dispositivos con un panel a través de acceso web e e-mail.

Otra empresa que se encuentra con este soporte es Sanyo. Cuenta con una cámara digital y una televisión que trabaja con esta nueva tecnología, la cámara puede subir las fotos digitales a un enlace determinado en el hogar de su propietario, mientras este se encuentra en redes inalámbricas públicas. La televisión puede utilizarse y operarse de forma remota, cuenta con la opción de reproducir las imágenes o películas en diferentes lugares a diferentes usuarios.

También Canon desarrollo un sistema de cámaras web que pueden ser controladas remotamente. Se puede observar el ambiente en el que se encuentran los niños, mascotas, o la máquina de café mientras está camino a casa.

Nokia por su cuenta dispone de una terminal de Internet que combina redes inalámbricas, identificación por radio frecuencia (RFID), y tecnología de IPV6 móvil. Esto demostró que es posible el uso para dispositivos móviles de servicios sobre Internet de una forma segura y certificada.

En la primera mitad del 2006, China tuvo planificado desarrollar un backbone que estaría constituido por 300 redes de campus, incluyendo 100 universidades, 100 institutos y 100 empresas. Los 5 mayores operadores de telecomunicaciones cumplieron con un papel clave en el desarrollo de este proyecto. El proyecto fue completado exitosamente antes del tiempo estimado y logró conectarse a redes IPV6 extranjeras por medio del Internet a mediados del 2006. Se están desarrollando gradualmente en cada ciudad redes de área metropolitana, con IPV6 formando parte fundamental en el desarrollo. IPV6 es usado en industrias como la militar, meteorología, sismología, arquitectura inteligente y redes digitales a nivel de hogares.
Las grandes compañías en este país se enfocan en IPV6, tales IBM.

Otros países en Asia están desarrollando sus propios grupos de trabajo para esta tecnología, entre estos están India, Corea, Tailandia y Taiwán, en la mayoría de estos países IPV6 tienen un fuerte apoyo gubernamental.

1.4.2 Europa

La comisión europea ha liderado y ha apoyado la introducción de IPV6 desde el 2000.
Esta cree que IPV6 es esencial para la competitividad en el área económica.

“Telia Sweden” fue uno de los primeros ISPs en ofrecer comercialmente servicios de IPV6. Actualmente algunos ISPs ofrecen servicios de IPV6, pero en el fondo muchos de estos están preparados para la introducción y pueden reaccionar rápido frente al crecimiento de la demanda en el mercado, en caso de algunas empresas ofrecen servicios a nivel mundial.

El espacio de direccionamiento y la movilidad que soportan IPV6, ofrecen una buena base para el desarrollo de voz sobre IP (VoIP), removiendo algunas limitaciones de IPV4, las cuales permiten que la movilidad sea más conveniente para un uso global.

La compañía alemana “Telekom”, que se formó a principios del 2004, creen que en el año 2020 todas las comunicaciones globales de telefonía serán basadas totalmente en IPV6.

Los vendedores de carros utilizarán también el protocolo IP para sus negocios. Renault, tiene un prototipo de automóvil que trabaja con un sistema basado en IPV6 que se ha trabajo conjuntamente con Cisco. Está equipado con un router Cisco que implementa IPV6 móvil y cuenta con una red interna, la cual permite monitorear, controlar y mantener; para acceso a información sobre clima, tráfico y rutas. También permite que los pasajeros se conecten a redes inalámbricas o bluetooth para navegar en la web o mirar televisión digital con dispositivos que soporten IPV6.

Otros vendedores de carros como BMW, Chrysler y Audi, están trabajando en proyectos iguales. Se rumora que un auto IP, en el futuro va a tener un mínimo de 20 direcciones IP.

1.4.3 Estados Unidos

Se asume que este país será el último en adoptar IPV6, por la simple razón de espacio de direccionamiento no es crítica.

En el verano del 2003, la situación cambió significativamente con el anuncio del departamento de defensa de migrar su red a IPV6 a lo largo del 2008.
Esta decisión aceleró el mercado de IPV6 no solo en Estados Unidos, sino en todo el mundo, esto demuestra como IPV6 es ampliamente utilizado en el campo militar, ya que precisa de velocidad y requerimiento de nuevos servicios.

CAPITULO 2
ESTRUCTURA DE IPV6

2.1 Estructura general de la cabecera

En primer lugar, se analizará la descripción de la cabecera de un paquete IPV4:

Fig.2.1 Cabecera de un paquete IPV4

Como se puede observar, la longitud mínima de la cabecera IPV4 es de 20 bytes (cada fila de la tabla supone 4 bytes). A ello se debe añadir las opciones, que dependen de cada caso.

A continuación se especifican los campos y sus tamaños:

- Versión (4 bits)
- Cabecera (4 bits)
- TOS (Type Of Service) – Tipo de Servicio (1 byte)
- Longitud Total (2 bytes)
- Identificación (2 bytes)
- Indicador – Indicador (4 bits)
- Desplazamiento de Fragmentación (12 bits – 1.5 bytes)
- TTL (Time To Live) – Tiempo de Vida (1 byte)
- Protocolo (1 byte)
- Checksum – Código de Verificación (2 bytes)
- Dirección Fuente de 32 bits (4 bytes)
- Dirección Destino de 32 bits (4 bytes)

Analizando la cabecera IPV6:

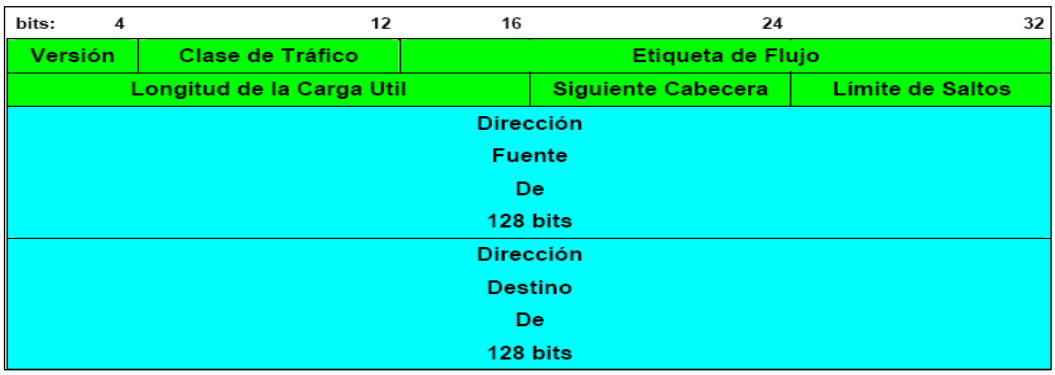


Fig.2.2 Cabecera de un paquete IPV6

En el caso de la cabecera IPV6 se tiene:

• Versión (4 bits)
• Clase de Tráfico (1 byte)
• Etiqueta de Flujo (20 bits)
• Longitud de Carga Útil (2 bytes)
• Siguiente Cabecera (1 byte)
• Límite de Saltos (1 byte)
• Dirección de Fuente (16 bytes)
• Dirección de Fuente (16 bytes)

2.2 Campos de la Cabecera

En IPV6, cinco campos de la cabecera IPV4 han sido suprimidos:

- Longitud de la cabecera
- Identificación
- Banderas
- Desplazamiento de Fragmento
- Suma de Comprobación

El campo de longitud de la cabecera fue removido porque este no es necesario en una cabecera de longitud formada. El IPV4, la mínima longitud de cabecera es de 20 bytes, pero si son añadidas varias opciones esta puede extenderse en rangos de 4 bytes hasta los 60 bytes. Sin embargo con IPV4 la información de la longitud total

de la cabecera es importante. En IPV6 el campo opciones esta definido por las extensiones de cabecera.

Los campos de identificación, banderas y desplazamiento de fragmento manejan la fragmentación de un paquete de la cabecera de IPV4. La fragmentación permite que un gran paquete sea enviado por una red que soporte solo paquetes de menor tamaño, un router en IPV4 divide el paquete en varios segmentos, el host de destino recibe los paquetes y los reensambla. Si un solo paquete se pierde o tiene un error toda la transmisión tiene que realizarse nuevamente lo cual es muy ineficiente. En IPV6 el host aprende el tamaño de la UNIDAD MAXIMA DE TRANSMISIÓN (MTU), a través de un proceso llamado ruta de descubrimiento MTU (path MTU discovery). Si el host que envía la información en IPV6 quiere fragmentar el paquete lo va a realizar a través de la extensión de cabeceras. Los routers IPV6 a través de la ruta del paquete no van a realizar una fragmentación como se realiza en IPV4, por lo tanto los campos de identificación, banderas y desplazamiento de fragmento son removidos de la cabecera de IPV6 y son insertados en la extensión de cabecera por el host de origen, si es necesario.

El campo de la cabecera “Suma de Comprobación” fue removido para obtener mayor velocidad de procesamiento. Si los routers no tienen que actualizar y revisar la Suma de Comprobación, el procesamiento será más rápido. Actualmente el riesgo para no detectar errores y paquetes no enrutados es mínimo.

El campo Clase de tráfico remplaza al campo Tipo de servicio. IPV6 tiene un mecanismo diferente para manejar preferencias.

Los campos “Tipo de Protocolo” y “Tiempo de Vida” fueron renombrados y levemente modificados. El campo “Etiqueta de Flujo” fue añadido.

2.2.1 Campos de la Cabecera de IPV6

2.2.1.1 Versión (4 bits)

Este es un campo de 4 bits que contiene la versión del protocolo. En el caso de IPV6, es el número 6. La versión numero 5 no puede ser usada porque esta ya está siendo implementada por una versión experimental del IP.

2.2.1.2 Clase de tráfico (1 byte)

Este campo reemplaza el campo “tipo de servicio” en IPV4. Este facilita el manejo de datos en tiempo real y cualquier otro tipo de datos que requieran un manejo especial, esta característica se puede usar para el envío de la información distinguiendo entre diferentes clases o prioridades de paquetes de IPV6

2.2.1.3 Etiqueta de Flujo (20 Bits)

Este campo ordena paquetes que requieren un mismo tratamiento, para facilitar el manejo del tráfico en tiempo real. El host de origen puede identificar una secuencia de paquetes con un grupo de opciones. Los routers mantienen una pista del flujo y pueden procesar los paquetes pertenecientes a este flujo más eficientemente ya que no tienen que reprocesar cada cabecera de los paquetes. La etiqueta de flujo y la dirección del nodo de origen son los datos que identifican a un flujo. Los nodos que no soportan las opciones del campo de la etiqueta de flujo son obligados a pasar el campo sin cambiarlo cuando reenvía el paquete y a ignorarlo cuando reciben el paquete. Todos los paquetes pertenecientes al mismo flujo tienen la misma dirección IP de origen y de destino.

2.2.1.4 Longitud de la Carga Útil (2 Bytes)

Este campo especifica la carga útil por ejemplo la longitud de los datos transmitidos después de la cabecera IP. El cálculo de la longitud en IPV6 es diferente al de IPV4, la longitud del campo en IPV4 incluye la longitud de la cabecera de, mientras que en IPV6 la longitud de la carga útil contiene solo los datos posteriores a la cabecera IPV6. Las cabeceras de extensión son consideradas parte de la carga útil y son incluidos en el cálculo de la longitud.

El hecho de que la longitud de la carga útil tiene un límite de 2 bytes, implica un tamaño máximo de paquete de la carga útil de 64KB. IPV6 utiliza “Cabecera de extensión Jumbograma”, mediante lo cual soporta grandes tamaños de paquetes si es necesario. Los Jumbogramas son relevantes solo cuando el nodo IPV6 tiene conectados links que tienen un MTU mayor a 64KB (especificado en RFC2675).

2.2.1.5 Siguiente Cabecera (1 Byte)

En IPV4 este campo es llamado “Protocolo” pero en IPV6 fue renombrado debido a la nueva organización de los paquetes IP. Si la siguiente cabecera es UDP ó TCP, este campo va a contener los mismos números de protocolos que en IPV4, por ejemplo numero de protocolo 6 para TCP o 17 para UDP. Si las extensiones de cabeceras son usadas en IPV6, este campo contiene un tipo de “Siguiente cabecera de extensión”. Las cabeceras de extensión están ubicadas entre la cabecera IP la cabecera TCP ó UDP. En la tabla se puede observar los posibles valores en el campo de siguiente cabecera.

Valor	Descripción
0	En IPV4: reservado y no usado En IPV6: Siguiente cabecera "Cabecera de opciones hop-by-hop"
1	Protocolo de control de mensajes de Internet (ICMPv4) soportado por IPV4
2	Protocolo de administración del grupo Internet (IGMPv4) soportado por IPV4
4	IPV4
6	TCP
8	Protocolo de puerta de enlace exterior (EGP)
9	"IGPany" Puerta de enlace privada interior (usada por cisco para el protocolo de enrutamiento IGRP)
17	UDP
41	IPV6
43	Cabecera de enrutamiento
44	Cabecera de fragmentación
45	Protocolo de enrutamiento de inter-dominio (IDRP)
46	Protocolo de reserva de recursos (RSVP)
47	Encapsulación de enrutamiento general (GRE)
50	Cabecera de carga útil con seguridad de encriptación
51	Cabecera de autenticación
58	ICMPv6
59	Ausencia de siguiente cabecera en IPV6
60	Cabecera de opciones de destino
88	Protocolo mejorado de enrutamiento de puerta de enlace interior

	(EIGRP propietario de CISCO)
89	OSPF
108	Protocolo de compresión de carga útil IP
115	Protocolo de encapsulación de capa 2 (L2TP Layer 2 tunneling protocol)
132	Protocolo de transmisión de control de cadena (SCTP)
135	Cabecera de movilidad (IPV6 mobile)
136-254	Valores no asignados
255	Reservado

Tabla 2.1 Valores del campo de siguiente cabecera

2.2.1.6 Limite de saltos (1 Byte)

Este campo es análogo al campo TTL en IPV4. El campo TTL contiene el numero de segundos, indicando cuanto tiempo puede existir el paquete en la red antes de ser destruido. En IPV4 la mayoría de routers simplemente decrementan este valor en uno por cada salto. Este campo fue renombrado como Limite de Saltos en IPV6. El valor de este campo ahora expresa el número de saltos en lugar del número de segundos. Cada nodo de reenvío decrementa este valor en uno. Si un router recibe un paquete con limite de saltos de 1, este lo decrementa a 0, descarta el paquete, y envía un mensaje ICMPv6 “Limite de saltos excedido durante el flujo” para el origen.

2.2.1.7 Dirección de Origen (16 Bytes)

Este campo contiene la dirección IP del origen del paquete.

2.2.1.8 Dirección de Destino (16 Bytes)

Este campo contiene la dirección IP del destinatario del paquete. Este puede ser el destinatario final o, por ejemplo si esta presenta la cabecera de enrutamiento, la dirección del router del próximo salto.

2.3 Extensiones de cabecera

La cabecera IPV4 puede ser extendida de un mínimo de 20 bytes a un máximo de 60 bytes para especificar opciones tales como las de seguridad, la fuente de ruteo, o Timestamping. Esta capacidad es raramente utilizada porque disminuye el desempeño.

Mientras más simple es la cabecera de un paquete, más rápido es el procesamiento. IPV6 ofrece una nueva forma de tratar con las opciones que han mejorado substancialmente procesamiento: maneja las opciones en encabezamientos adicionales “Extensiones de Cabecera”, éstas se insertan dentro de un paquete sólo si las opciones son necesarias.

La especificación IPV6 actual (RFC 2460) define seis extensiones de cabecera:

- Cabecera de opciones hop-by-hop (salto a salto)
- Cabecera de encaminamiento
- Cabecera de fragmentación
- Cabecera de opciones de destino
- Cabecera de autenticación
- Cabecera de encriptación.

Puede haber cero, uno, o más de extensiones de cabecera en un paquete IPV6. Las extensiones de cabecera son colocadas entre la cabecera IPV6 y la capa superior de la cabecera de protocolo. Cada extensión de cabecera es identificada por el campo de “Siguiente Cabecera (Next Header)”. Además, son examinadas o son procesadas sólo por el nodo identificado en el campo de la “dirección del destino de la cabecera IPV6”. Si la dirección en el campo de la “dirección del Destino” es una multicast, las extensiones de cabecera son examinadas y son procesadas por todos los nodos que pertenecen a ese grupo del multicast. Las extensiones de cabecera deben ser procesadas estrictamente en el orden en la que ellas aparecen en la cabecera del paquete.

Hay una excepción de la regla que dice que sólo el “nodo de destino” procesará una extensión de cabecera. Si la extensión de cabecera es una “cabecera de opciones de Salto a Salto”, la información que lo lleva debe ser examinada y debe ser procesada por cada nodo a lo largo del camino del paquete. La cabecera de opciones salto a salto, si está presente, debe seguir inmediatamente el encabezamiento IPV6. Y debe ser indicado por el valor 0 en el campo de “siguiente cabecera” en la cabecera IPV6.

Esta arquitectura es muy flexible para desarrollar extensiones de cabecera para usos en necesidades futuras. Nuevas extensiones de cabecera pueden ser definidas y utilizadas sin cambiar la cabecera IPV6. Un buen ejemplo es la “cabecera de Movilidad” definido para el Móvil IPV6.

La figura a continuación muestra como se utilizan las extensiones de cabecera:

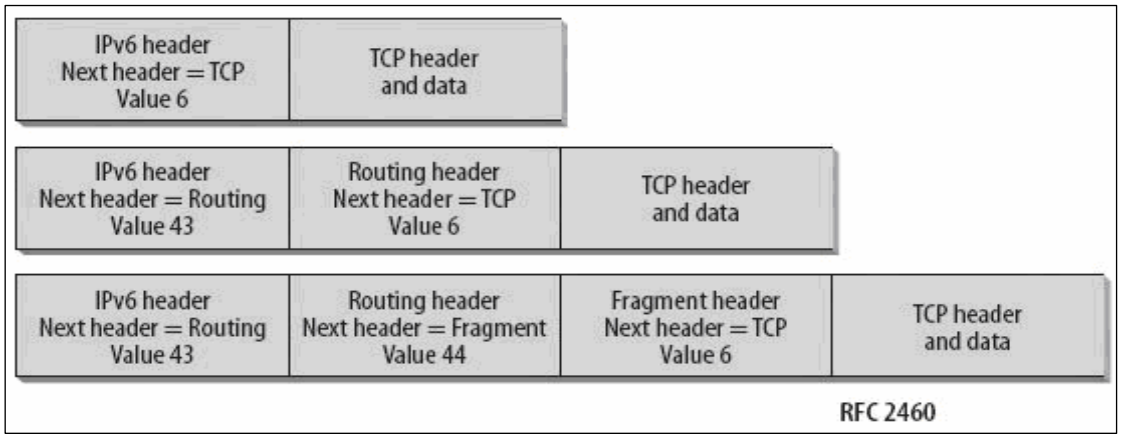


Fig.2.3 Uso de Extensiones de Cabecera

Cada extensión de cabecera tiene una longitud múltiplo de ocho bytes para que encabezamientos subsiguientes siempre puedan ser alineados. Si un nodo es requerido a procesar la “siguiente cabecera” pero no puede identificar el valor en el campo de “siguiente cabecera”, se desecha el paquete y se envía un mensaje de “Problema del Parámetro ICMPV6” a la fuente del paquete.

Si más de una extensión de cabecera es utilizada en un solo paquete, la cabecera deberá utilizar el siguiente orden:

- Cabecera IPV6

- Cabecera de opciones hop-by-hop (salto a salto)
- Cabecera de opciones de destino (para opciones a procesadas por el primer destino que aparece en el campo de la “dirección de destino IPV6”, además los destinos subsiguientes se listarán en la “cabecera de encaminamiento”)
- Cabecera de enrutamiento
- Cabecera de fragmentación
- Cabecera de autenticación
- Cabecera de encriptación
- Cabecera de opciones de destino (para opciones a ser procesadas sólo por el destino final del paquete)
- Cabecera de capa superior

2.3.1 Cabecera de opciones hop-by-hop (salto a salto):

Esta cabecera transporta información opcional que debe ser examinada por cada nodo a través de la ruta del paquete, esta se encuentra a continuación de la cabecera IPV6 y es indicada por el valor cero de la “siguiente cabecera”. Con IPV4 la única manera para que un router determine si es necesario examinar un datagrama es que por lo menos analice parcialmente los datos de la capa superior en todos los datagramas. Este proceso vuelve lento sustancialmente el proceso de ruteo. Con IPV6, si no se presenta la cabecera de opciones hop-by-hop el router sabe que no se requiere de procesamiento de información específica y puede encaminar el paquete inmediatamente hacia el destino final. Si es que existe la cabecera de opciones hop-by-hop, el router necesita solo examinar esta cabecera y no realizar más procesos en el paquete.

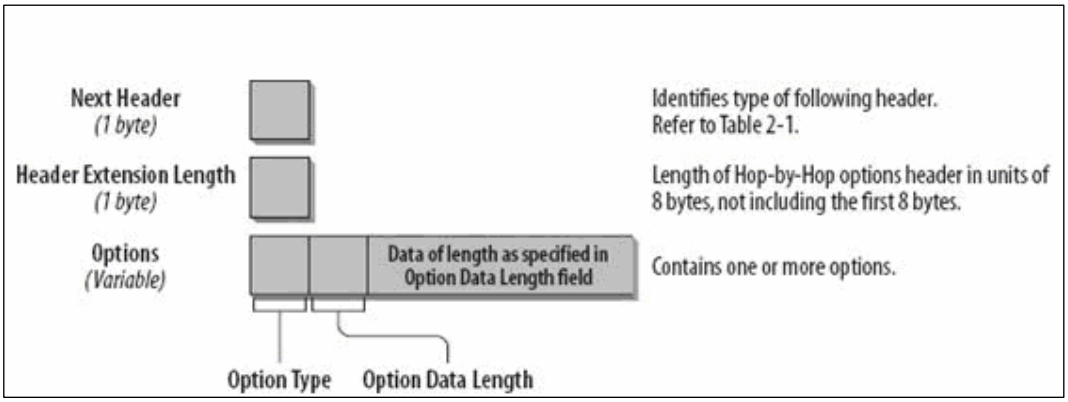


Fig.2.4 Formato de la cabecera de opciones hop-by-hop

A continuación se detalla cada campo:

2.3.2 Siguiente Cabecera (1 byte):

Identifica el tipo de cabecera que está a continuación de la cabecera de opciones hop-by-hop.
Los valores que puede tomar este campo son los listados en la tabla de “siguiente cabecera” previamente presentada en campos de cabecera IPV6.

2.3.3 Longitud de Extensión de cabecera (1 byte):

Este campo identifica la longitud de la cabecera de opciones hop-by-hop en unidades de 8 bytes. La longitud calculada no incluye los primeros bytes. Si la cabecera es menor a 8 bytes este campo es cero.

2.3.4 Opciones (tamaño variable):

Puede ser una o más opciones, la longitud de esta opción es variable y se especifica su tamaño en el campo de “longitud de extensión de cabecera”.

En el campo de tipo de opción, el primer byte contiene información acerca de cómo debe ser tratado en el caso de que el nodo de procesamiento no reconozca esta opción, el valor de los primeros 2 bits especifica la acción que se va a ejecutar:

- **00** : Salta y continua procesando
- **01** : Descarta el paquete.

- **10** : Descarta el paquete y envía un mensaje ICMP código 2 para la dirección del origen del paquete indicando un tipo de opción no reconocido.
- **11** : Descarta el paquete y envía un mensaje ICMP de código 2 para la dirección de origen del paquete, solo si el destino no es una dirección de multicast.

El tercer bit del campo de tipo de opciones, especifica si la información de opción puede cambiar el valor=1 o no cambiar valor = 0.

2.3.5 Tipo de Opción Jumbogram

Este tipo de opción hop-by-hop soporta el envió de jumbogramas IPV6. El campo de longitud de carga útil soporta un tamaño máximo de paquete de 65,535 bytes. Las opciones de carga útil “jumbo” (RFC 2675), permiten que grandes paquetes sean enviados.

En la cabecera de IPV6 de un paquete con opciones de carga útil “jumbo”, el campo de la longitud de carga útil toma el valor de cero.

El valor de tipo de opción de 194 indica opciones de carga útil “jumbo”. El campo de la longitud de carga útil jumbo tiene 32 bits y por lo tanto soporta la transmisión de paquetes comprendidos entre 65,536 y 4,294,967,295 bytes.

2.3.6 Opción de Alerta de Router

Este tipo de opción indica al router que el paquete contiene información importante que debe ser procesada cuando se reenvía el paquete. La opción es usada en su mayoría por MLD (Multicast Listener Discovery) y RSVP (Resource Reservation Protocol), esto se especifica en 2711.

RSVP usa control de paquetes que contienen información que necesita ser interpretada o actualizada por los routers a través de la ruta. Este control de paquetes usa la cabecera de opciones hop-by-hop, entonces solo los routers procesan el paquete. Los paquetes de datos regulares no necesitan esta extensión de cabecera y por lo tanto se reenvían nuevamente sin una revisión futura en el router.

Los primeros 3 bits en el campo de tipo de opción son configurados en cero. Un router que no conoce esta opción ignora y reenvía el paquete. En los restantes 5 bits del primer byte el tipo de opción 5 es especificado. El campo de longitud de datos de opción contiene el valor 2, el cual indica que el siguiente valor del campo tiene una longitud de 2 bytes (valores definidos en RFC 2711).

- **0** : El paquete contiene un mensaje MLD.
- **1** : El paquete contiene un mensaje RSVP.
- **2** : El paquete contiene un mensaje “Active Networks”.
- **3** – **35** : El paquete contiene un nivel anidado de reservación agregada. (RFC 3175, RSVP)
- **36-65, 535** : Reservada por IANA.

2.3.7 Cabecera de Fragmentación

En un host IPV6 que quiere enviar un paquete a un destino IPV6 la ruta para descubrimiento del MTU para determinar el máximo tamaño del paquete que puede ser usado en la ruta hacia el destino. Si el paquete que va a ser enviado es más grande que el MTU permitido, el host de origen fragmenta el paquete. A diferencia de IPV4, con IPV6 el router a través de la ruta no fragmenta paquetes. La fragmentación ocurre solo cuando el host de origen envía el paquete. El host de origen maneja en reensamblado del paquete. La cabecera de fragmentación esta identificada por el valor de la siguiente cabecera de 44 en la cabecera anterior.

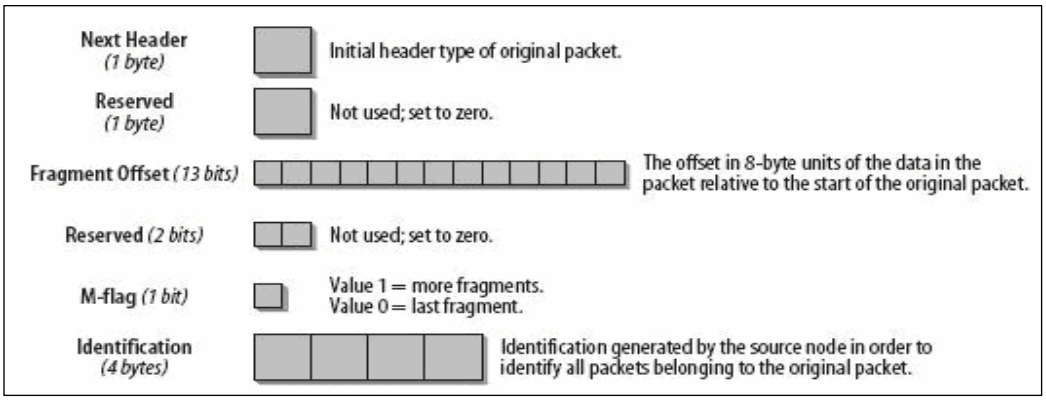


Fig.2.5 Formato de la Cabecera de Fragmentación

A continuación se describe cada campo:

- *Siguiente cabecera (1 byte)*

El campo de siguiente cabecera identifica el tipo de cabecera que esta a continuación. Se utiliza los mismos valores que se describieron en tabla de la página 17.

- *Reservado (1 byte)*

No utilizado, toma el valor de 0.

- *Desplazamiento de la fragmentación (13 bits)*

Indica el desplazamiento en unidades de 8-bytes de la información en relación a la ubicación en el paquete original.

- *Reservado (2 bits)*

No utilizado, toma el valor de 0.

- *Bandera M (1 bit)*

El valor de 1 indica más fragmentos, el valor de 0 indica que es el último fragmento.

- *Identificación (4 bytes)*

Generado por el host de origen para identificar todos los paquetes pertenecientes al paquete original (paquete fragmentado). Este campo es usualmente implementado como un contador, incrementándose en uno por cada paquete que necesita ser fragmentado por el host de origen.

El paquete inicial no fragmentado hace referencia al paquete original, el cual tiene partes no fragmentables como la cabecera IPV6 más cualquier otra cabecera de extensión que debe ser procesada por los nodos a través de la ruta hacia el destino. La parte fragmentable del paquete original consiste del cualquier cabecera de extensión que necesite solo ser procesada por el destino final, mas cabeceras de capa superior y datos.

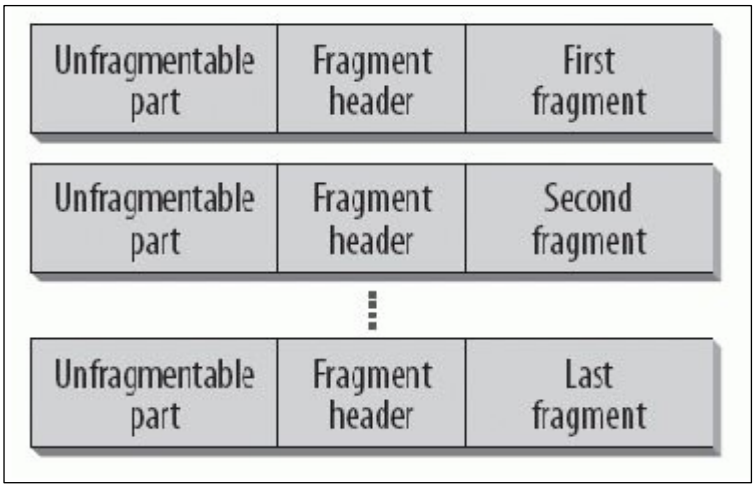


Fig.2.6 Cabecera de Fragmentación

La parte no fragmentable del paquete original aparece en cada fragmento, seguida de la cabecera de fragmentación y luego la información fragmentada. La cabecera IPV6 del paquete original tiene que ser levemente modificada. El campo “longitud” representa la longitud del fragmento (sin incluir la cabecera IPV6) y no la longitud del paquete original.

El nodo de destino une todos los fragmentos y los reensambla. Los fragmentos deben tener direcciones de origen y destino idénticas y el mismo valor de identificación para que sean reensamblados. Si todos los fragmentos no llegan al destino dentro de 60 segundos después del primer fragmento, el destino va a descartar todos los paquetes y enviara un mensaje ICMPv6 “Tiempo de reensamblaje de fragmento excedido” al origen.

2.3.8 Cabecera de opciones de destino

La cabecera de opciones de destino transporta información opcional que es examinada solo por el nodo de destino (la dirección de destino en la cabecera IPV6). El valor de la siguiente cabecera 60 identifica este tipo de cabecera. La cabecera de opciones de destino puede aparecer dos veces en un paquete IPV6; cuando es insertada antes de cabecera de enrutamiento, conteniendo información que va a ser procesada por los routers listó en la cabecera de ruteo y cuando es insertada antes de la cabecera de protocolos de capa superior, conteniendo información para el destino final del paquete.

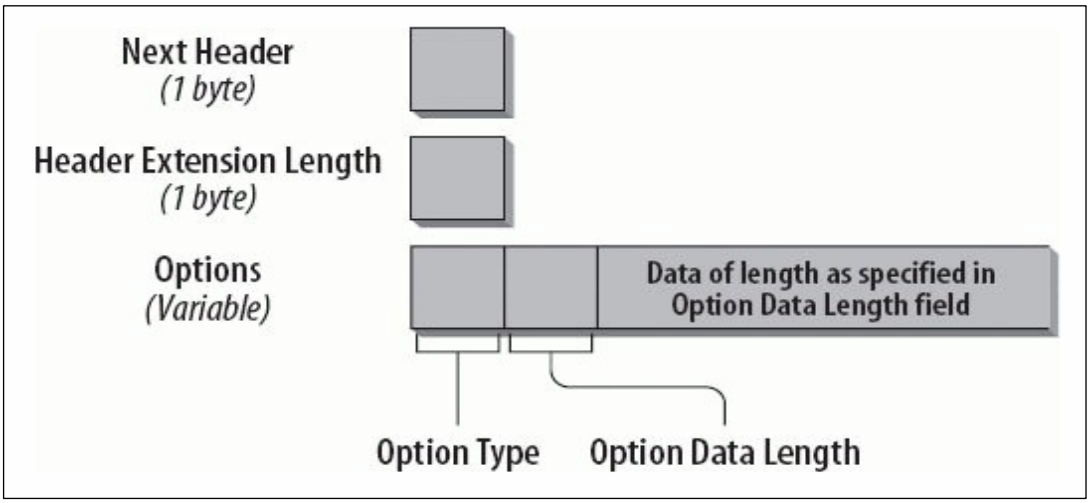


Fig. 2.7 Cabecera de Opciones de Destino

- *Siguiente cabecera (1 byte)*

Este campo identifica el tipo de cabecera que esta a continuación de la cabecera de opciones de destino.

- *Longitud de cabecera de extensión (1 byte)*

Este campo identifica la longitud de la cabecera de opciones de destino en unidades de 8 bytes. El cálculo de la longitud no incluye los primeros 8 bytes.

- *Opciones (tamaño variable)*

Este campo puede contener varias opciones. La longitud del campo opciones es variable y es determinada en el campo longitud de cabecera de extensión.

CAPITULO 3
DIRECCIONAMIENTO EN IPV6

Espacio de direccionamiento IPV6

Los 32 bits del espacio de direccionamiento de IPV4 proveen teóricamente un máximo de 2^{32} direcciones, que equivale aproximadamente a 4,29 billones de direcciones. La población actual del mundo alcanza aproximadamente 6,4 billones de personas. Si fuese posible el uso del 100% de el espacio de direccionamiento de IPV4, no fuese posible proveer una dirección por cada persona en el planta, de hecho, solo una pequeña fracción del espacio de direccionamiento puede ser usada. En los inicios de IP, nunca se imagino la existencia del Internet tal como la conocemos hoy, por lo cual grandes bloques de direcciones fueron asignados sin consideraciones para el enrutamiento global y asuntos de conservación de direcciones. Estos rangos de direcciones no pueden ser recuperados, la cual causa que direcciones sin utilizar que no están libres para ser asignadas.

Si se busca un acceso a solo el 20% de la población mundial, el espacio de direccionamiento de IPV4 nunca podría cubrir la demanda. Los cálculos demuestran que se requeriría alrededor de 390 bloques de direcciones IPV4 clase A (/8), pero solo quedaban 64 bloques de direcciones case A sin asignar según la IANA a finales del 2005. La evolución del Internet y varios servicios muestran que en el futuro, no solo se necesitara direcciones para usuarios y computadoras, también se necesitara una gran cantidad de direcciones para todos los tipos de dispositivos que necesitan una conexión permanente al Internet tales como teléfonos celulares, PDAs, cámaras web, refrigeradoras, carros y muchos mas artículos. Los fabricantes de carros son un ejemplo claro de este requerimiento, han diseñado “the networked car of the future” (carro comunicado a través de la red), este necesita por lo menos 20 direcciones IP por carro. Estas direcciones serán usadas para monitoreo y mantenimiento así como para acceso a servicios como información sobre el tiempo y información del trafico, este es un carro prototipo de la Renault integrado con un router Cisco que implementa IPV6.

El espacio de direccionamiento de IPV4 con su definición de clases de direcciones (A, B, C, D, E) permite 2,113,389 redes, con la introducción del enrutamiento de interdominio sin clase (CIDR), este numero se extendió ligeramente. Comparado esto con IPV6, podemos observar que el espacio de direccionamiento con el prefijo

actual para un direccionamiento de unicast global (binario 001) permite un total de 2^{45} redes con /48, o dicho de otra forma 35,184,372,088,832 redes, cada red posteriormente puede ser dividida en 65,536 subredes usando los 16 bits restantes.

3.1 Tipos de direcciones

En IPV4 tenemos las direcciones de unicast, broadcast, y multicast. Con IPV6, las direcciones de broadcast ya no serán usadas, las direcciones de multicast pasan a ocupar su función. Esta es una buena noticia ya que los paquetes de broadcast son un problema en la mayoría de las redes. Las direcciones anycast, un nuevo tipo de direcciones introducidas con el RFC 1546, ya fueron utilizadas en IPV4 pero probablemente serán usadas mas eficientemente en IPV6.

Las direcciones de IPV6 se clasifican dentro estas 3 categorías:

- ✓ **Unicast** estas direcciones identifican únicamente a una interfaz de un nodo IPV6. El paquete enviado a una dirección unicast es entregado a la interfaz identificada por esta dirección.

- ✓ **Multicast** estas direcciones identifican a un grupo de interfaces IPV6. El paquete enviado a una dirección multicast es procesado por todos los miembros de grupo multicast.

- ✓ **Anycast** estas direcciones son asignadas a varias interfaces (comúnmente en múltiples nodos). El paquete enviado a una dirección anycast es entregado a solo una de estas interfaces, generalmente a las más cercanas.

A una sola interface se le puede asignar varias direcciones IPV6 de cualquier tipo (unicast, multicast, y anycast). Un nodo por lo tanto puede ser identificado por la dirección de cualquiera de sus interfaces. Es posible también asignar una dirección de unicast a varias interfaces por razones de carga compartida, pero si se realiza esto, se necesita asegurar que el hardware y los drivers lo soporten.

3.2 Notación de las direcciones

En IPV6 las direcciones tienen 128 bits, o 16 bytes. Las direcciones están divididas en ocho bloques hexadecimales de 16 bits separados por dos puntos, por ejemplo:

2001:DB8:0000:0000:0202:B3FF:FE1E:8329

Para facilitar las cosas, es posible realizar algunas abreviaciones. Por ejemplo, los ceros iniciales en un bloque de 16 bits pueden ser omitidos, la dirección del ejemplo anterior quedaría así:

2001:DB8:0:0:202:B3FF:FE1E:8329

Dos veces dos puntos pueden remplazar a ceros consecutivos, si se aplica esta regla la dirección quedaría así:

2001:DB8::202:B3FF:FE1E:8329

Se debe tener en cuenta que los 2 puntos dobles solo pueden aparecer una sola vez en la dirección. La razón de esta regla es que la computadora siempre utiliza una representación binaria de 128 bits completa de la dirección, incluso si la dirección expuesta es simplificada. Cuando la computadora encuentra dos doble punto, esta expande con tantos ceros como sean necesarios para completar los 128 bits. Si la dirección tiene dos veces dos dobles puntos, la computadora no va a poder interpretar cuantos ceros tiene que completar en cada abreviación. Por lo tanto la dirección IPV6 2001:DB8:0000:0056:0000:ABCD:EF12:1234 puede ser representada de las siguientes formas (nótese las posible posiciones de los dos puntos):

2001:DB8:0000:0056:0000:ABCD:EF12:1234
2001:DB8:0:0056:0:ABCD:EF12:1234
2001:DB8::0056:0:ABCD:EF12:1234
2001:DB8:0:0056::ABCD:EF12:1234

En ambientes donde se trabaja con IPV4 e IPV6, otra forma conveniente de representar una dirección IPV6 es poner los valores de la dirección IPV4 dentro de

los cuatro bytes de más bajo en orden de la dirección. La dirección IPV4 192.168.0.2 puede ser representada como x:x:x:x:x:x:192.168.0.2, y una dirección 0:0:0:0:0:0:192.168.0.2 puede ser escrita como ::192.168.0.2 o si se prefiere ::C0A8:2.

3.2.1 Notación del prefijo

Los prefijos globales de ruteo son los bits más representativos de la dirección IP, usados para identificar las subredes o un tipo especifico de dirección. La notación de los prefijos es muy similar a la utilizada en ruteo de interdominio sin clase del direccionamiento IPV4, y también es comúnmente usado direccionamiento de subredes IPV4. La notación añade la longitud del prefijo, definiendo el número de bits anteponiendo un slash, de la siguiente forma:

Dirección IPV6/longitud del prefijo

La longitud del prefijo especifica cuantos bits, empezando por los más significativos, definen el prefijo. Esto es otra forma de representar las mascara de subred, la cual en IPV4 define cuantos bits de la dirección están representando la red. El prefijo es utilizado para definir la subred a la que pertenece una interfaz y es usado en el procesamiento de la información de los routers. El siguiente ejemplo explica como el prefijo debe ser interpretado, considerando la dirección IPV6 2E78:DA53:1200::/40. Para facilitar el procedimiento se pasa de hexadecimal a binario como se muestra a continuación.

Hexadecimal	Binario	Numero de Bits
2E 78	0010 1110 0111 1000	16
DA 53	1101 1010 0101 0011	16
12	0001 0010	8
		Total: 40 bits

Tabla.3.1 Conversión Hexadecimal a Binario

La notación reducida (remplazando una secuencia de ceros con los dobles dos puntos) es también posible en la representación del prefijo. Se debe tener mucho cuidado ya que con frecuencia se presentan 2 o más rangos de ceros en una dirección, y solo uno puede ser comprimido.

Para representar un caso de lo mencionado anteriormente utilizamos la siguiente dirección 2001:DB8:0000:0056:0000:ABCD:EF12:1234/64, una forma errada de representar el prefijo seria la siguiente:

2001:DB8::56/64

Para verificar esta notación, se debe expandir la dirección nuevamente siguiendo las reglas, obteniendo la siguiente dirección 2001:DB8:0000:0000:0000:0000:0000:0056, con 2001:DB8:0000:0000 como los 64 bits de prefijo. De modo que la compresión lleva a una interpretación equivocada. Esto no es idéntico a la dirección y prefijo original. Para asegurarnos que la interpretación de la dirección no sea ambigua, se debe representar así:

2001:DB8:0:56::/64

3.2.2 Prefijos de enrutamiento globales

La tabla que se muestra a continuación resume la asignación actual de prefijos reservados y direcciones especiales, tales como direcciones para un enlace local o direcciones de multicast. La mayor parte de este espacio de direccionamiento (sobre el 80%) no esta asignado, lo cual queda para futuras asignaciones.

Asignación	**Prefijo Binario**	**Prefijo Hexa.**
No asignadas	0000 0000	::0/8
Reservadas	0000 001	
Unicast Global	001	2000::/3
Enlace local Unicast	1111 1110 10	FE80::/10
Direcciones Locales	1111 110	FC00::/7
Administración Privada	1111 1101	FD00::/8
Multicast	1111 1111	FF00::/8

Tabla.3.2 Asignación actual de prefijos y direcciones especiales

3.3 Direcciones Especiales

La primera parte del espacio de direccionamiento IPV6 con el prefijo 0000 0000 están reservadas, fuera de este rango, las direcciones especiales se definen de la siguiente manera:

3.3.1 Direcciones no especificadas

Las direcciones no especificadas tienen un valor de 0:0:0:0:0:0:0:0 y se le conoce como direcciones de todo ceros. Estas son comparables con 0.0.0.0 en IPV4. Estas indican la ausencia de una dirección valida, y esta puede, por ejemplo, ser usada como dirección de origen por el host durante el proceso de arranque cuando este envía una solicitud de información para la configuración de la dirección. Esta dirección se puede abreviar como ::. Nunca se debe asignar dinámicamente o estáticamente a una interfaz y no debe aparecer como una dirección IP de destino o dentro de una cabecera de enrutamiento IPV6.

3.3.2 Direcciones de loopback

En IPV4 la dirección de loopback es la 127.0.0.1, esta dirección es muy útil para la ubicación de fallas y pruebas de la pila de protocolos IP por que esta puede ser usada para enviar un paquete a la pila de protocolos sin que salga la información a la subred. Con IPV6, la dirección de loopback trabaja de la misma forma y es representada como 0:0:0:0:0:0:0:1, abreviada como ::1. Nunca debe ser asignada de forma estática o dinámica en una interfaz.

3.3.3 Direcciones IPV6 con direcciones IPV4 inmersas.

Debido a que la transición de IPV6 va a ser gradual, dos tipos especiales de direcciones han sido definidas para garantizar la compatibilidad con IPV4, ambas son descritas en el RFC 429.

3.3.4 Direcciones IPV6 compatibles con IPV4

Este tipo de direcciones son usadas para transmitir paquetes IPV6 a través de un túnel, dinámicamente sobre una infraestructura de ruteo IPV4. Los nodos IPV6 que usan esta técnica, se les asigna una dirección especial unicast IPV6 que transporta

la dirección IPV4 dentro de los 32 bits de más bajo orden. Este tipo de dirección hasta ahora ha sido utilizada raramente.

3.3.5 Direcciones IPV6 apuntadas a IPV4

Este tipo de direcciones son usadas para representar las direcciones de los nodos IPV4 como direcciones IPV6. Un nodo IPV6 puede usar esta dirección para enviar un paquete a un nodo IPV4. La dirección también transporta la dirección IPV4 en los 32 bits de más bajo orden.

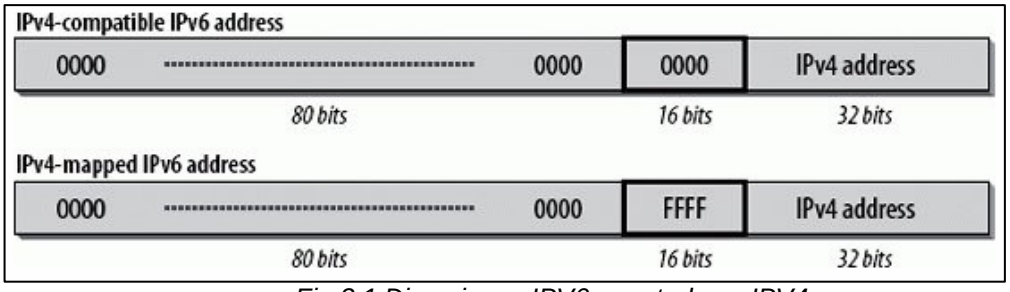


Fig.3.1 Direcciones IPV6 apuntadas a IPV4

Los dos tipos de direcciones son muy parecidas, la única diferencia es los 16 bits que se encuentran en el medio. Cuando estos toman el valor de 0, la dirección es una IPV6 compatible con IPV4, si estos bits son 1 es una dirección IPV6 mapeada a IPV4.

3.3.6 Dirección *6to4*

La IANA tiene permanente asignado un identificador de 13 bits TLA para operaciones 6to4 dentro del rango de direcciones unicast global (001). 6to4 es uno de los mecanismos definidos para lograr que host o redes IPV6 se comuniquen sobre una infraestructura puramente IPV4. El identificador del TLA 6to4 es 0x0002 y el formato de la trama es como el que se muestra en la figura.

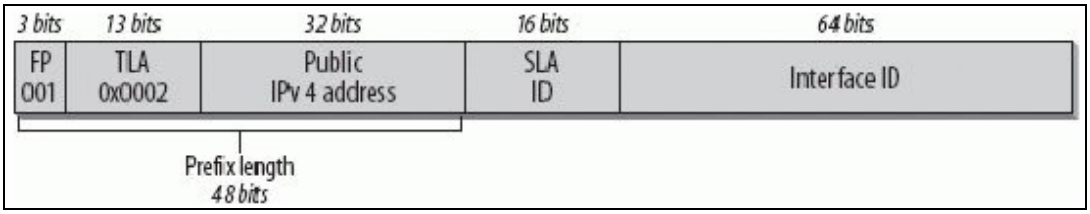


Fig.3.2 Dirección 6to4

El prefijo tiene una longitud total de 48 bits. La dirección IPV4 en el prefijo debe ser una dirección IPV4 pública y es representada en notación hexadecimal. En efecto, si se configura una interface para 6to4 con la dirección IPV4 62.2.84.115, la dirección 6to4 es 2002:3E02:5473::/48. A través de esta interface, todos los host IPV6 en este enlace pueden encapsular sus paquetes sobre la infraestructura IPV4.

3.3.7 Direcciones ISATAP

El protocolo de direccionamiento automático de túnel dentro de un sitio (Intra-Site Automatic Tunnel Addressing Protocol - ISATAP) es un mecanismo que hace túneles automáticamente especificado en el RFC 4214. Este es diseñado para dos nodos separados por una infraestructura IPV4. Trata la red IPV4 como un gran enlace de capa de red y permite que los dos nodos formen automáticamente un túnel entre ellos usando cualquier formato de direcciones IPV4. Windows XP incluye una implementación de ISATAP. ISATAP usa un tipo de identificador de 0XFE para especificar direcciones IPV6 con direcciones IPV4 inmersas.

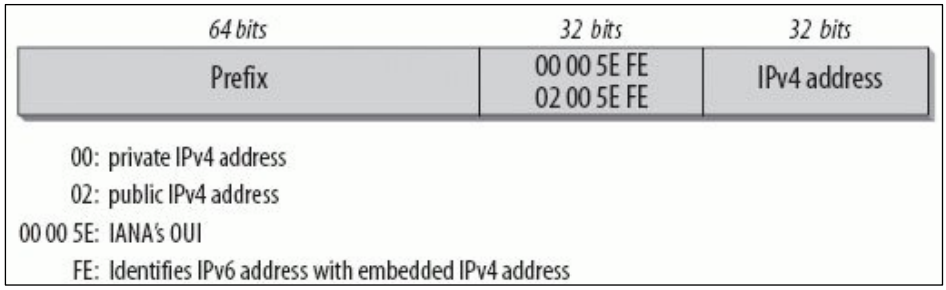


Fig.3.3 Direcciones ISATAP

Los primeros 64 bits siguen el formato de las direcciones globales unicast. La IANA es propietaria del identificador único organizacional (OUI) 00-00-5E y especifica el formato EUI-48 de las asignación del identificador de la interface. Dentro de los primeros 16 bits, el tipo de identificador muestra si la dirección IPV4 proviene de un rango privado (0000) o una dirección única global (0200). Los siguientes 8 bits contienen un identificador que indica si la dirección IPV6 contiene una dirección IPV4 inmersa, el valor de este identificador es 0xFE. Los últimos 32 bits contienen la dirección IPV4 inmersa, la cual puede escribirse en notación decimal o hexadecimal.

Asumiendo que se tiene la dirección IPV4 192.168.0.1 y el host es asignado un prefijo de 64 bits de 2001:db8:510:200::/64. La dirección ISATAP para este host es

2001:db8:510:200:0:5EFE:192.168.0.1. Alternativamente se puede usar la notación hexadecimal para la dirección IPV4, en este caso la dirección es 2001:db8:510:200:0:5EFE:C0A8:1. La dirección de enlace local para este host es FE80:: 5EFE:192.168.0.1.

3.3.8 Direcciones Teredo

Teredo es un mecanismo diseñado para proveer conectividad para host los cuales están bajo uno o más NATs. Se realiza mediante la formación de túneles para los paquetes IPV6 dentro de UDP. El mecanismo consta de los clientes, servidores y relevos (relays) teredo. Los relays Teredo son routers IPV6 situados entre el servicio de teredo y la red IPV6 nativa. Este método esta definido en el RFC4380. Como muchos usuarios de Internet privados están detrás de NATs, se ha considerado que la utilización de este método va a ser muy común hasta que los ISPs pasen sus redes a IPV6. El formato de la dirección teredo es el siguiente:

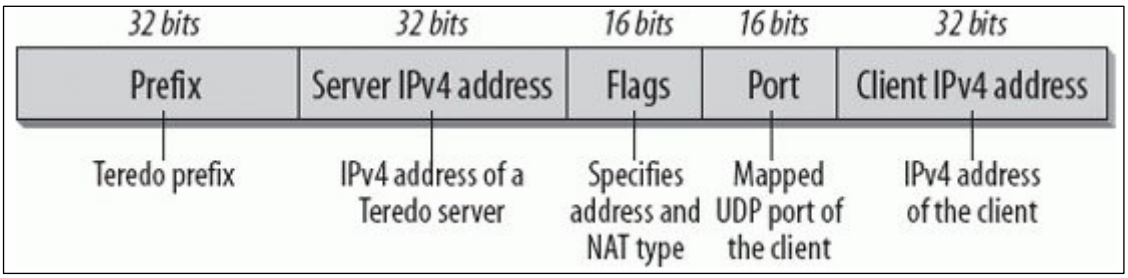


Fig.3.4 Direcciones Teredo

El prefijo tiene una longitud de 32 bits. El prefijo de servicio IPV6 Teredo global es 2001:0000:/32. El campo “dirección IPV4 del servidor” tiene una longitud de 32 bits y contiene la dirección IPV4 del servidor teredo. El campo “banderas” (flags) tiene 16 bits y especifica el tipo de dirección y NAT en uso. Los 16 bits del campo “puerto” (port) contiene el puerto UDP apuntado del servicio teredo en el cliente y el campo “dirección IPV4 del cliente” contiene la dirección IPV4 apuntada del cliente.
En este formato, tanto el puerto apuntado UDP como la dirección IPV4 del cliente son modificadas, cada bit en la dirección y el puerto son invertidos.

3.4 Direcciones Anycast

Las direcciones unicast son diseñadas para proveer redundancia y balanceo de carga en situaciones donde muchos host o routers proveen el mismo servicio. Anycast se definió para ser utilizado por servicios como DNS y HTTP.

En la práctica, anycast no fue implementado para lo que fue diseñado. Con frecuencia el método conocido como unicast compartido es utilizado, este método es aplicado asignando una dirección unicast regular a varias interfaces y creando varias entradas en la tabla de ruteo.

Dentro de una red donde un grupo de routers pueden proveer acceso a un dominio de enrutamiento común, se les puede asignar una dirección común. Cuando un cliente envía un paquete a esta dirección, este va a ser reenviado al siguiente router habilitado. IPV6 móvil también usa direcciones de anycast.

Cuando se utiliza direcciones anycast, hay que considerar que el hecho de que el emisor no tiene control sobre que interface el paquete fue entrado, esta decisión es tomada por el protocolo de enrutamiento. Cuando el emisor envía varios paquetes a una dirección de anycast, los paquetes pueden llegar a destinos diferentes, si hay una serie de peticiones y respuestas o si el paquete tiene que ser fragmentado, esto puede causar los problemas.

La dirección anycast del router de subred, es un ejemplo y su formato es el siguiente:

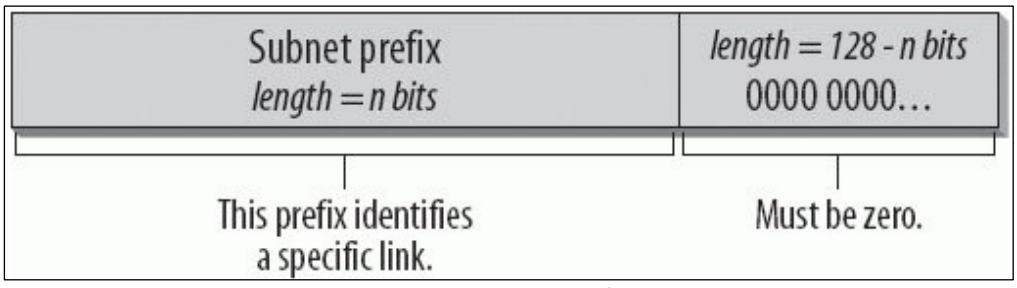


Fig.3.5 Formato de dirección Anycast

Básicamente, se ve como una unicast normal con un prefijo especificando la subred y un identificador con todos los valores en cero. Un paquete enviado a esta dirección va a ser entregado a un router en una subred.

Las direcciones anycast de subred reservada pueden tener uno de dos formatos, como se muestra a continuación:

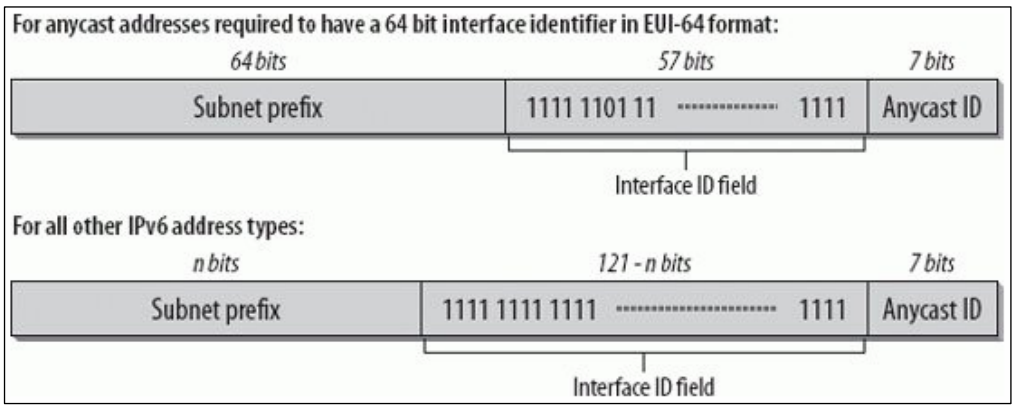


Fig.3.6 Formatos de direcciones Anycast de subred reservada

3.5 Direcciones de Multicast

La dirección multicast es un identificador para un grupo de nodos identificados por los bits de alto orden FF. O 1111 1111 en notación binaria. Un nodo puede pertenecer a más de un grupo multicast. Cuando un paquete es enviado a una dirección multicast, todos los miembros del grupo multicast procesan el paquete. Multicast existe en IPV4, pero este concepto ha sido redefinido y mejorado para IPV6. El formato de una dirección multicast es el siguiente:

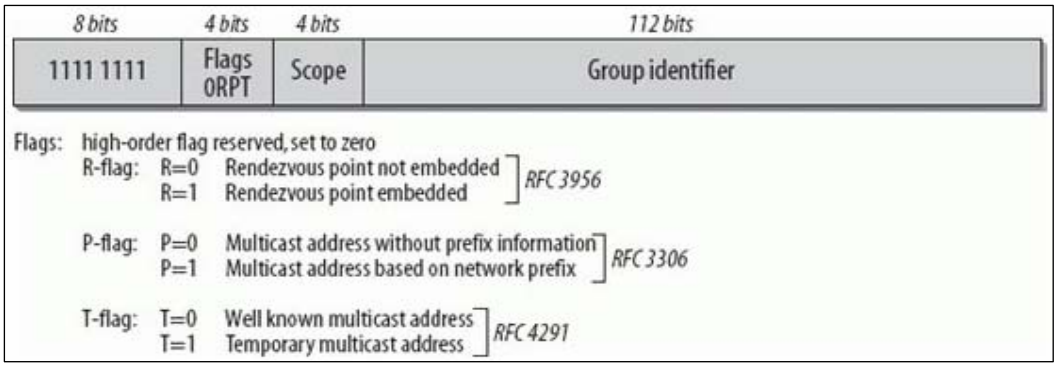


Fig.3.7 Formato de dirección Multicast

El primer byte identifica la dirección como una dirección de multicast. Los siguientes cuatro bits son usados para banderas, definidos de la siguiente manera: el primer bit de la bandera tiene que ser cero; reservado para usos futuros. El segundo bit indica si la dirección multicast inmerso un punto centralizado (rendezvous point). Un punto centralizado es un punto de distribución para una corriente multicast

especifica en una red multicast (RFC 3956). El tercer bit indica si la dirección multicast tiene inmerso un prefijo de información. El ultimo bit del campo bandera indica si la dirección es asignada permanentemente, es decir, una dirección multicast bien conocida asignada por la IANA o una dirección multicast temporal. El valor de cero en el último bit define una dirección bien conocida; el valor de uno indica una dirección temporal. El campo “alcance” (scope) es situado para limitar el alcance de la dirección multicast. Los posibles valores se muestran en la siguiente tabla:

Valor	**Descripción**
0	Reservado
1	Alcance a interface local
2	Alcance a conexión local
3	Reservado
4	Alcance a administrador local
5	Alcance a sitio local
6, 7	No asignados
8	Alcance a organización local
9, A, B, C, D	No asignados
E	Alcance global
F	Reservado

Tabla.3.2 Posibles Valores de “Alcance” (Scope)

Las divisiones de zonas de alcance que sean otros a los mencionados en la tabla, tienen que ser definidos y configurados por el administrador de la red. Los alcances reservados no pueden ser utilizados.

3.6 Direcciones de unicast globales

Las direcciones de unicast globales son identificadas por el prefijo binario 001.

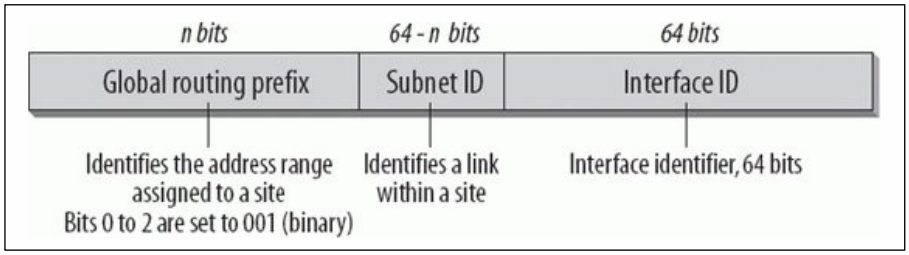


Fig.3.8 Formato de dirección Unicast

Los prefijos globales de enrutamiento identifican el rango de direcciones asignado a un sitio. Esta parte de la dirección es asignada por un servicio de registro internacional y el proveedor de servicio de Internet (ISP) y tiene una estructura jerárquica. El identificador de la Subred (Subnet ID) identifica una conexión dentro de un lugar. A una conexión puede asignarse varios identificadores de subred, el administrador local del sitio asigna esta parte de la dirección. El identificador de la internas (Interface ID) identifica a una internas en una subred y debe ser única dentro de la subred.

SEGURIDAD Y CALIDAD DE SERVICIO

4.1 Conceptos generales de seguridad

Las prácticas de seguridad Estándar envuelve dos triadas: la CIA y la AAA.

4.1.1 CIA (Confidentiality, Integrity, Availabilty):

4.1.1.1 Confidentiality (Confidencialidad):

La información almacenada o transmitida no puede ser leída o alterada por una parte no autorizada.

4.1.1.2 Integrity (Integridad):

Cualquier modificación de información transmitida o almacenada puede ser detectada.

4.1.1.3 Availability (Disponibilidad):

La información en cuestión es fácilmente accesible a usuarios autorizados todo el tiempo.

4.1.2 AAA (Authentication, Authorization, Accounting):

4.1.2.1 Authentication (Autenticación):

Las formas comunes de la autenticación incluyen los nombres de usuario y las contraseñas o las combinaciones de tarjeta/PIN de ATM.

4.1.2.2 Authorization (Autorización):

Asegurar que el usuario o grupo autenticados tienen los derechos apropiados para conseguir acceso a la información a la que ellos procuran acceder. Las implementaciones comunes incluyen las listas del control del acceso (ACLs).

4.1.2.3 Accounting (Contabilidad):

El acto de recolectar información en el uso del recurso. El diario de un servidor de HTTP sería una forma común de la contabilidad.

Estos requisitos de seguridad necesitan ser proporcionados por dos elementos básicos de la seguridad: codificación (proporciona confidencialidad) y sumas de comprobación seguras (proporciona integridad). Las combinaciones convenientes de estos dos elementos pueden ser utilizadas para proporcionar servicios más complejos, tal como la autenticidad.

Hay dos formas de codificación que son utilizadas comúnmente. La primera es llamada "Criptografía Clave Secreta," llamada también “codificación clave simétrica”, que requiere de emisor y receptor para convenir en un secreto compartido (es decir, una clave o contraseña), esto es utilizado para cifrar y desencriptar la información cambiada. Los algoritmos claves, simétricos y comunes son D, 3DES, la IDEA, RC-4, y AES.

El segundo es llamado "Criptografía Clave Pública," llamado también codificación asimétrica. Un algoritmo asimétrico de codificación utiliza un par de claves que consisten en una clave conocida y distribuida públicamente y una clave privada individual. Cuándo un mensaje es encriptado utilizando una clave pública y desencriptado por el receptor con una llave privada, sólo el receptor destinado es capaz de ver el mensaje encriptado. Esta forma de codificación puede ser utilizada para establecer un cambio confidencial de datos. Si además, el mensaje fue encriptado también con clave privada de emisor y entonces desencripta el receptor con una clave pública, los servicios de seguridad de la autenticación del origen de datos y “nonrepudiation” son agregados. Los algoritmos claves, asimétricos y comunes son RSA y ElGamal.

4.2 Elementos de seguridad en IPV6.

IPsec describe mecanismos generales de seguridad que pueden ser utilizados con ambos protocolos, IPV6 e IPV4. Esto significa que IPV6 no es más seguro que IPV4. La diferencia en la seguridad es que IPsec puede ser instalado separadamente para IPV4, mientras que es un obligatorio y la parte esencial del stack IPV6, y por lo tanto disponible con cualquier implementación.

La especificación de IPsec define protocolos para el encabezamiento de la autenticación (AH) y el Encapsulando encabezamiento de Carga Útil de Seguridad (ESP). Con IPV6, estos encabezamientos son incluidos como extensiones de cabecera. Una implementación de IPsec debe soportar ESP y puede soportar AH. Con la especificación más antigua, el soporte para ambos protocolos fue requerido. El requisito para el soporte AH apoyo ha sido quitado porque ESP puede ser utilizada para proporcionar la integridad, que en la mayoría de los casos tiene probado ser suficiente.

IPsec diferencia dos modos del transporte:

Modo Transporte

El SA (Security Associations) es hecho entre dos nodos finales y define la codificación o la autenticación para la carga útil de todos paquetes de IP para esa conexión. El encabezamiento de IP no es encriptado.

Modo Túnel

El SA es hecho generalmente entre dos gateways de seguridad. El paquete entero inclusive el encabezamiento original de IP es encriptado o es autenticado encapsulándolo en un nuevo encabezamiento. Esto es la base para una red privada virtual (VPN).

4.2.1 Cabecera de Autenticación

La cabecera de autenticación (AH) proporciona la integridad y la autenticación (no confidencialidad) para todos los datos transportados en un paquete de IP. Soporta diferentes mecanismos de autenticación. Está especificado en RFC 4302 (volviendo obsoleto al RFC 2402) y es indicado por el valor de protocolo 51 en el encabezamiento anterior.

La AH está localizada entre las cabeceras de IPV6 y la capa superior (TCP, UDP, ICMP). Si las extensiones de cabecera están presentes, tiene que ser colocada después del "Hop-by-Hop", "Routing" y de "Fragmentos de extensiones de cabecera".

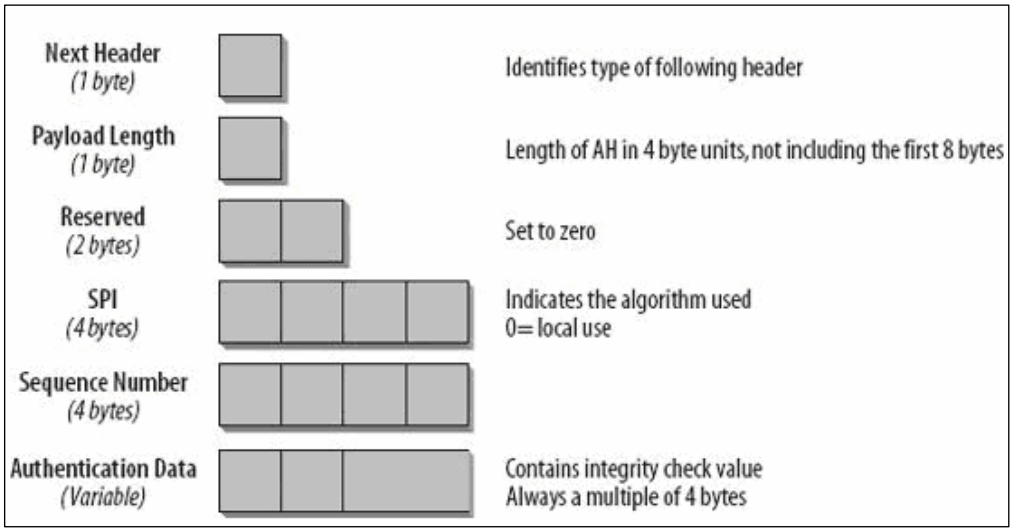


Fig. 4.1 Formato de la Cabecera de Autenticación

Detallando los campos:

4.2.1.1 Siguiente Cabecera (1 byte)

El campo de siguiente cabecera identifica el tipo de cabecera que sigue a la AH.

4.2.1.2 Longitud de la carga útil (1 byte)

Describe la longitud de la cabecera en unidades de cuatro bytes, sin incluir los primeros ocho bytes en el cálculo. Esta indicación de la longitud es necesaria porque los datos de la autenticación en la AH pueden diferir en longitud dependiendo del algoritmo utilizado.

4.2.1.3 Reservado (2 bytes)

No usados, puestos en 0.

4.2.1.4 Índice del Parámetro de la seguridad (SPI) (4 bytes)

Las Asociaciones de seguridad (SA) son los acuerdos entre semejantes de comunicación. Tres elementos forman parte del acuerdo: una clave, un mecanismo de codificación o autenticación, y los parámetros adicionales para el algoritmo. SAs es unidireccional, y cada servicio de seguridad requiere un SA. Esto significa que dos semejantes de comunicación que quieren cifrar y autenticar una necesidad de

comunicación de dos vías necesitan cuatro SAs (un par para la codificación y un par para la autenticación).

Arbitrario de 32 bits. Utilizado por el receptor para identificar el SA a que un paquete entrante pertenece. El campo de SPI es obligatorio, y este mecanismo se utiliza para trazar el tráfico de entrada al unicast SAs debe ser soportado por todas las implementaciones AH. Si una implementación de IPsec soporta multicast, debe soportar además multicast SAs que utiliza el algoritmo de-multiplex, utilizado con el fin de trazar datagramas de entrada de IPsec a SAs. Los valores de SPI de 1 hasta 255 son reservados por la IANA para uso futuro. El valor de SPI de 0 es reservado para el uso local e implementación específica y no debe ser enviado en la red.

4.2.1.5 Número de Secuencia (4 bytes)

Este número de 32 bits, tiene que ser puesto por el emisor, pero es el receptor quien decide actuar sobre el. Asegura que los paquetes con datos idénticos no se reenvíen de forma repetida. Esto previene los ataques de repetición en un unicast o el único-emisor SA. Para un multi-emisor SA, las características "sin contestación" de AH no están disponibles, porque la AH no tiene un medio para sincronizar contadores de paquetes entre múltiples emisores. En el establecimiento de un SA, el valor es puesto en 0 en el emisor y en el receptor. El primer paquete siempre tiene el valor 1, que es aumentado en uno para cada paquete consecutivo. Cuándo el valor de 232 es alcanzado, el contador es reinicializado en 0.

4.2.1.6 Valor de Chequeo de Integridad (longitud variable)

Este campo contiene la suma de comprobación (Valor de Chequeo de Integridad, ICV) para el paquete. La longitud depende del algoritmo escogido en establecer el SA. Es siempre un múltiplo de cuatro bytes.

La especificación de AH en el RFC 4302 define una nuevo Número de Secuecia (ESN) Prolongado (64-BIT). No se puede ver en la Figura 5-1 porque sólo 32 bits (orden bajo) del ESN son transmitidos. El orden alto de 32 bits se mantiene como parte del contador del número de secuencia, tanto parar el transmisor como el receptor, y se incluyen en el cómputo del ICV. El número de secuencia de 64 bits es una nueva opción diseñada para soportar las implementaciones de alta velocidad

de IPsec. El uso de un ESN prolongado se negocia en la configuración del SA. El defecto con IKEV2 es ESN, a menos que de 32 bits sea negociado explícitamente.

El checksum es calculado sobre los siguientes campos:

- Todos los campos de la cabecera IP o los campos de extensiones de cabecera antes de la cabecera de autenticación (AH) que no cambian en el tránsito o cuyo valor al llegar al destino pueden ser predichos. Por ejemplo, si una extensión de cabecera de ruteo está presente, la última dirección en la extensión de cabecera de ruteo es utilizada para el cálculo. El campo de la Clase, la Etiqueta del Flujo, y el Límite del Salto no están incluidos en el cálculo.

- Todos los campos de la cabecera autenticación.

- Otras extensiones de cabecera presentes y la carga útil.

- Los bits de orden alto del ESN (si es empleado) y cualquier pad implícito requerido por el algoritmo de integridad.

Los algoritmos siguientes son considerados convenientes para IPsec:

- Códigos de mensajes clave de autenticación (MACs) se basan en algoritmos simétricos de codificación

- Funciones de un solo sentido (por ejemplo, MD5, SHA-1, SHA-256, etc.)

Las debilidades aparentemente se hacen presentes en MD5; sin embargo, ellos no deben afectar el uso de MD5 con HMAC.

La cabecera de autenticación puede ser usada en modo túnel y transporte, como se muestra en la siguiente figura:

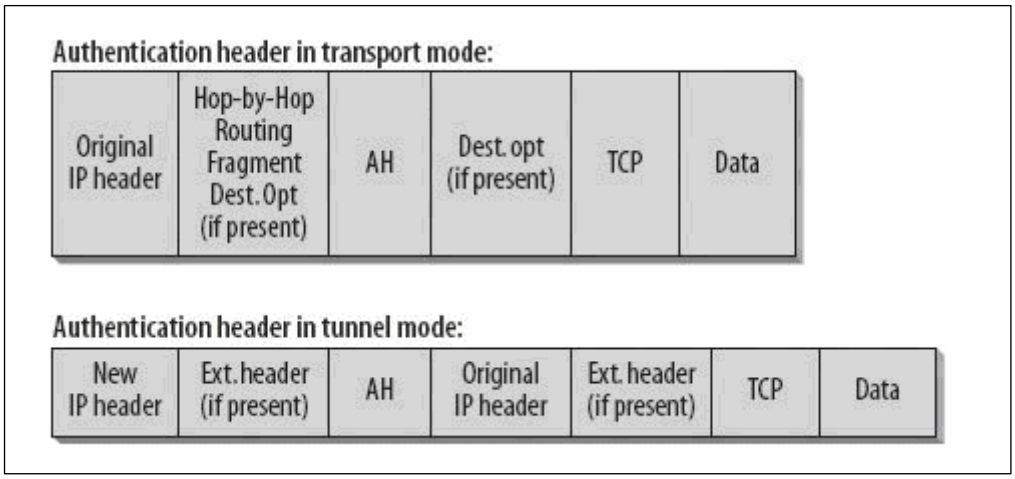


Fig. 4.2 Modos de transporte de la Cabecera de Autenticación

En el modo de transporte, la carga útil entera, inclusive los campos de la cabecera IPV6, que no cambian en el tránsito, son asegurados. En el modo de túnel, el paquete interior contiene la dirección de IP de emisor y receptor. El encabezamiento exterior de IP contiene la dirección de IP de los puntos finales de túnel. En este caso, el paquete original completo, incluso los campos del encabezamiento exterior que no cambia en el tránsito, son asegurados.

4.2.2 Encapsulado de la cabecera de seguridad de carga útil

La cabecera de seguridad de carga útil (ESP) proporciona la integridad, la confidencialidad, la autenticación del origen de datos, el servicio anti-repetición, y confidencialidad limitada de Flujo de Tráfico para todos datos de punta a punta transportados en un paquete de IP. El conjunto de servicios proporcionados es negociado en el establecimiento del SA. La ESP está definida en RFC 4303 y está indicada por un valor de protocolo de 50 en el encabezamiento anterior.

La ESP está localizada al frente del transporte (por ejemplo, UDP o TCP), control de red (por ejemplo, ICMP), o ruteo (por ejemplo, OSPF) cabecera de protocolo.

El formato de la ESP se muestra a continuación:

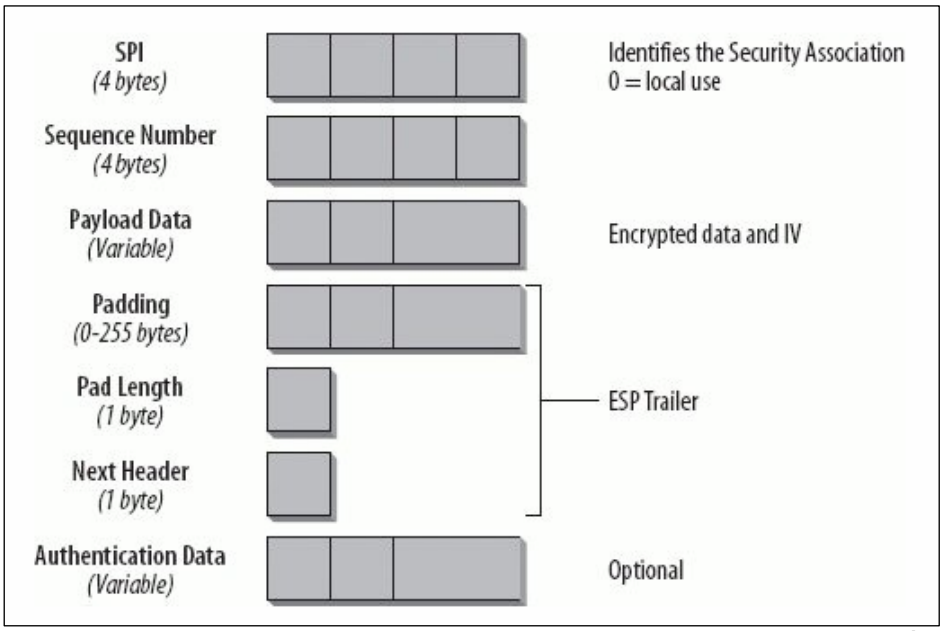


Fig. 4.3 Formato de Encapsulado de la cabecera de seguridad de carga útil

4.2.2.1 Índice del Parámetro de Seguridad (SPI, 4 bytes)

Valor arbitrario de 32 bits. Utilizado por el receptor para identificar el SA al que un paquete entrante pertenece. El campo de SPI es obligatorio, y este mecanismo para trazar el tráfico de entrada al unicast SAs debe ser soportado por todas implementaciones de ESP. Si una implementación de IPsec soporta multicast, debe soportar adicionalmente multicast SAs que utiliza el algoritmo de-multiplex especificado para este fin para trazar datagramas de entrada de IPsec a SAs. Los valores de SPI de 1 a 255 son reservados por la Autoridad de Asignación de Números de Internet para uso futuro (IANA). El valor de SPI de 0 es reservado para el uso local e implementación específica y no debe ser mandado por el cable.

4.2.2.2 Número de Secuencia (4 bytes)

Este número de 32 bits es un contador ascendente. Tiene que ser puesto por el emisor, pero es la elección del receptor si actúa sobre el. Asegura que los paquetes con datos idénticos no se repitan. Esto previene los ataques de repetición en un unicast o el solo-emisor SA. Para un multi-emisor SA, las características anti repetición del ESP no están disponibles, porque la ESP no tiene un medio para sincronizar los contadores de paquete entre múltiples emisores. En el establecimiento de un SA, el valor es puesto a 0 en el emisor y en el receptor. El

primer paquete siempre tiene el valor 1, que aumenta en uno para cada paquete consecutivo. Cuándo el valor 232 es alcanzado, el mostrador es reinicializado a 0.

4.2.2.3 Datos de carga útil (longitud variable)

Contiene los datos encriptados así como el vector de la inicialización de codificación (IV) si es requerido por el mecanismo de codificación.

4.2.2.4 *Padding* (0 a 255 bytes)

Usado para alinear el paquete a un múltiplo de 4 bytes y para alcanzar un tamaño mínimo de paquete si el mecanismo de codificación lo requiere.

4.2.2.5 Longitud de *Pad* (1 byte)

Indica el número de bytes precedentes de *pad*.

4.2.2.6 Siguiente Cabecera (1 byte)

Identifica el tipo de cabecera que sigue a la cabecera de ESP. Utiliza los valores listados en la tabla 2-1. Para facilitar la generación rápida y desechando el tráfico de *padding* con el apoyo de confidencialidad de flujo de tráfico, el valor de protocolo 59 (no siguiente cabecera) designa un paquete "falso". El receptor de un paquete falso lo debe desechar sin crear un mensaje de error.

4.2.2.7 Valor del Chequeo de integridad (la longitud variable)

El Valor del Chequeo de Integridad (ICV) es un campo de la longitud variable que contiene un checksum computado sobre la cabecera ESP, la carga útil, y campos de remolque de ESP. El campo de ICV es opcional. Está presente sólo si el servicio de la integridad es escogido, y es proporcionado por un algoritmo separado de la integridad o un algoritmo combinado del modo que utiliza un ICV. La longitud del campo es especificada por el algoritmo de integridad escogido y asociado con el SA.

El Padding, la longitud del Pad, y los campos de siguiente cabecera forman parte del remolque de ESP. El algoritmo de codificación es especificado manualmente e

incluido en el SA para la corriente de paquete o negociado dinámicamente por el protocolo clave de intercambio.

4.3 Principios de la QoS.

El modelo actual de IP trata a todos los paquetes de la misma forma. Todos son reenviados con el tratamiento de "mejor esfuerzo" con el criterio "primero en llegar, primero servido". El cuál da un camino al paquete a través de la red dependiendo de los *routers* disponibles, tablas de enrutamiento, y carga general de la red.

Los protocolos de QoS tienen la tarea de proporcionar las diferentes cadenas de datos con prioridades y garantizar las calidades tales como tiempos, anchura de banda y demora. Existen actualmente dos arquitecturas principales: Los Servicios integrados (IntServ) y los Servicios Diferenciados (DiffServ). Ambas arquitecturas utilizan las políticas del tráfico y pueden ser combinadas para tener en cuenta QoS tanto en una LAN como en una WAN.

Las políticas del tráfico pueden ser utilizadas para hacer la transmisión de datos, dependiendo de ciertos criterios por ejemplo, si hay suficientes recursos disponibles para reenviar los datos según sus requisitos de QoS. Las políticas del tráfico pueden controlar también las cadenas de datos y hacer los ajustes o las restricciones si es necesario. Además, asegurándose que los requisitos de QoS para el tráfico sensible a demora puedan ser utilizados también con fines comerciales, tales como el control de costos dependiendo de los diferentes niveles de servicio.

4.3.1 Servicios Integrados

La Arquitectura de Servicios Integrados (IntServ) está basada en el paradigma de la anchura de banda y todos los recursos relacionados por el flujo son reservados en una base de punta a punta. Esto presupone que los routers almacenan información acerca de flujos y analizan cada paquete para determinar si pertenece a un flujo específico de manera de reenviar el paquete según los criterios para ese flujo específico.

RSVP (Protocolo de Reservación de Recursos, RFC 2205) forma parte de la arquitectura de IntServ. RSVP es un protocolo de señalización usado para reservar

anchura de banda y otros recursos de QoS a través de una red de IP. IntServ combinado con RSVP puede tener una aplicación compleja y a causa de su escalabilidad limitada, es inadecuado ofrecer una solución general de QoS para el Internet global.

4.3.2 Servicios Diferenciados

Mientras IntServ ofrece la capacidad de asignar anchura de banda a flujos diferentes, la arquitectura de Servicios Diferenciados (DiffServ) fue diseñada para hacer una diferenciación menos específica de clases para aumentar su escalabilidad y el valor práctico en redes grandes y en el Internet.

Los Servicios diferenciados son especificados en RFCs 2474 y 2475. RFC 2474, "Definición del campo de Servicios Diferenciados (Campo DS) en las cabeceras IPV4 e IPV6," especifica el campo DS. Esto es implementado en el campo ToS en la cabecera IPV4 y en el campo de la clase del tráfico de la cabecera IPV6. El campo DS es utilizado por routers de DiffServ para determinar el QoS reenviando los requisitos de paquetes. Los nodos de comunicación pueden clasificar su comunicación por una conducta llamada Conducta Por-Salto (PHB). Basado en el PHB, los paquetes reciben el tratamiento específico en routers de DiffServ.

Un domino DiffServ (DS) es un grupo contiguo de routers DS que trabajan con una política común del servicio aplicada en todos routers. Un dominio DS está definido por routers de frontera DS. Los routers de la frontera clasifican las cadenas de datos entrantes y aseguran que todos los paquetes que atraviesen el dominio sean marcados apropiadamente, utilizan una Conducta Por-Salto del set disponible para el dominio. Los routers dentro del dominio escogen las reglas de reenvío basadas en los valores de DiffServ en paquetes los cuales se envían a los PHBs correspondientes. En los Servicios Diferenciados Codepoint (DSCP; refiérase a la figura a continuación) el valor puede utilizar por defecto (DSCP=0) o una cartografía individual configurada para el dominio. Un dominio DS consiste generalmente en una red o un conjunto de redes, que constituye una unidad administrativa.

Una región DS es un conjunto de dominios DS contiguos. Las regiones DS pueden asegurar los servicios DS para senderos de cruce de dominio. Los dominios singulares pueden utilizar cartografías individuales de definiciones y PHB-codepoint de PHB internamente. Entre los dominios dentro de una región, los

acondicionadores del tráfico son responsables de proporcionar una correcta traducción del PHBs de las diferentes cartografías. Si las políticas de los grupos de PHB, y cartografías de codepoint son las mismas en todos los dominios dentro de la región, no se necesitará de acondicionadores de tráfico.

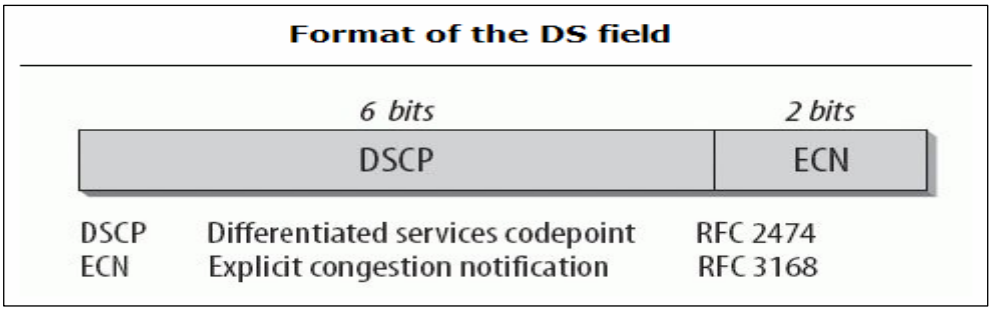


Fig. 4.4 Formato del campo DS

The codepoint pools

Pool	Codepoint space	Assignment policy
1	xxxxx0	Standard use
2	xxxx11	Experimental/local use
3	xxxx01	Experimental/local use; potential standard use in the future

Fig. 4.5 Conjuntos de Codepoints

El clasificador de paquetes escoge los paquetes de una cadena de datos basada en la información en los encabezamientos de paquete y según reglas predefinidas. Hay dos tipos de clasificadores: el Clasificador Agregado de Conducta (BA) clasifica paquetes basados en los campos DS, y el Clasificador Multi Campo (MF) clasifica paquetes basados en campos, ya sea, diferentes de campos de cabecera o una combinación de campos de cabecera, tal como la dirección de la fuente o el destino, el campo DS, el número de protocolo, el puerto de la fuente o el destino, o la información tal como interfaz entrante.

4.4 QoS en el protocolo IPV6

Los diseñadores de IPV6 no se han enfocado en requerir mecanismos específicos para QoS, sino en la ofrecer tanta flexibilidad como sea posible en soportar diferentes mecanismos de QoS.

4.4.1 Cabecera IPV6:

En la cabecera IPV6 hay dos campos que puede ser utilizados para QoS: la clase de Tráfico y el campo de etiquetador de flujo.

4.4.2 Clase de Tráfico:

El uso del campo de Clase del Tráfico de 1 byte es especificado en el RFC 2474. Este RFC introduce el término "campo DS" para el campo de la Clase del Tráfico. La meta de esta especificación es que los routers de DiffServ tienen un conjunto conocido de rutinas DS, que son determinados por el valor en el campo DS. Estos valores de DSCP son conducidos a Conductas de Por-Salto (PHB) (Figura 4.4).

El campo de DSCP dentro del campo DS (los seis bits más significativos del campo DS) es utilizado para el codepoint, que especifica el PHB. Con este campo, 64 codepoints diferentes pueden ser especificados. Este conjunto de codepoints ha sido dividido en tres partes para controlar la asignación de PHBs. (Figura 4.5).

Un conjunto de 32 codepoints recomendados (conjunto 1) es asignado por la estandardización formal; un conjunto de 16 más codepoints (conjunto 2) es reservado para el uso experimental o local; el conjunto final de 16 codepoints (conjunto 3) está inicialmente disponible para el uso experimental o local, pero debe ser utilizado como un conjunto de capacidad excesiva si el conjunto 1 se agota.

El PHBs especifica cómo los paquetes deben ser reenviados. Un PHB predefinido denominado por un codepoint DS de "todo-ceros" debe ser proporcionado por cualquier router DS. El PHB predefinido describe el "mejor esfuerzo" reenviando la conducta disponible en routers existentes. Tales paquetes son reenviados sin adherir a cualquier política de prioridad; es decir, la red entregará tantos de estos paquetes como le sea posible, tan pronto como sea posible, basándose en recursos existentes tales como la capacidad de la memoria o procesamiento. Los paquetes recibidos con un codepoint indefinido deben ser reenviados también como si fueran marcados para la conducta predefinida.

El campo DS no especifica PHBs; especifica codepoints. El número de codepoints es limitado a 64, mientras que el número de PHBs es ilimitado. Allí son

recomendados rutas de codepoints a PHBs. Estas rutas pueden ser definidas individualmente dentro de dominios administrativos, que hace el número de PHBs posible ilimitado.

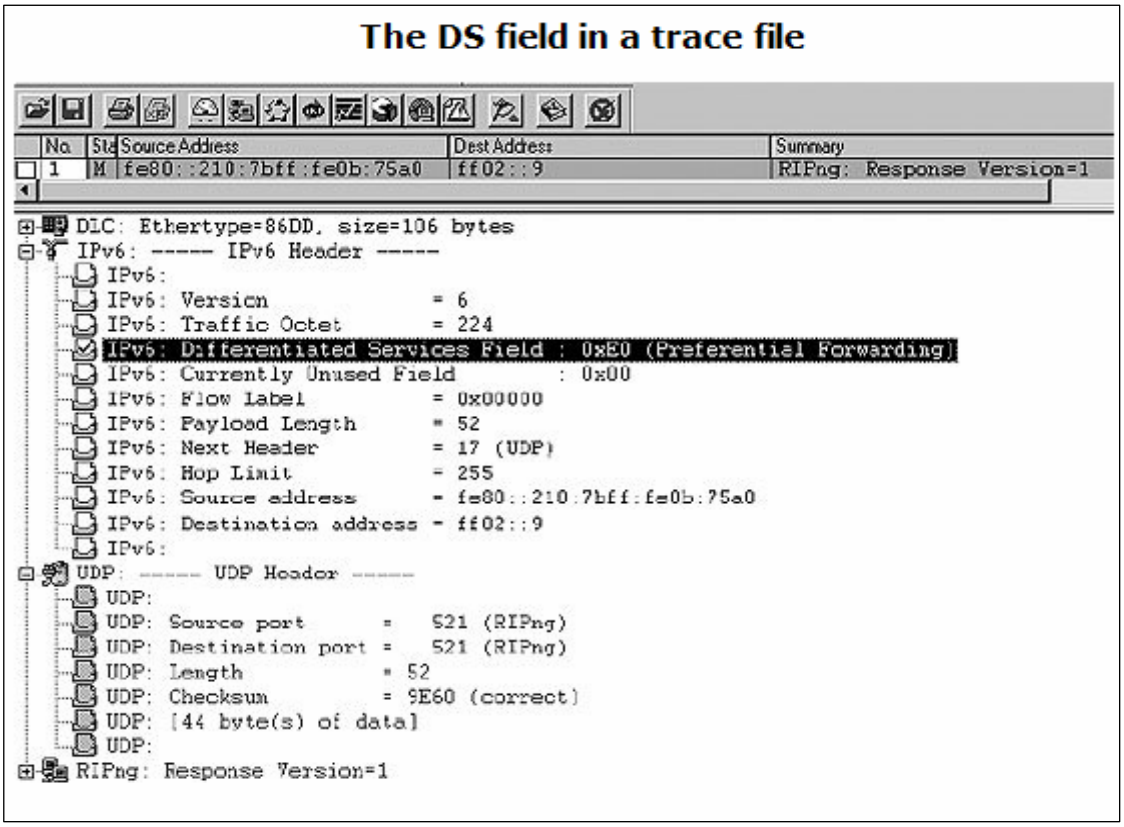


Fig. 4.6 Campo DS en un archivo rastreado

Los restantes dos bits del campo DS no son utilizados según RFC 2474, y son especificados en RFC 3168, "Adición de la Notificación Explícita de la Congestión (ECN) a IP". Ellos proporcionan cuatro posibles codepoints (00 a 11) que son utilizados para la Notificación de la Congestión. Generalmente la sobrecarga de un router sólo podría ser determinada basándose en la pérdida de paquete. Con el uso de estas Notificación de la Congestión de Codepoints, un router puede señalar sobrecarga antes de la pérdida de paquete. Este método es semejante al "relevo de cadena" de BECNs (Backwards Explicit Congestion Notification) y FECNs (Forwards Explicit Congestion Notification)

Los dos bits son utilizados de la siguiente manera:

00: El paquete no utiliza ECN.

01/10: El emisor y el receptor son ECN-PERMITIDOS.

11: El router señala la congestión.

El campo de 20 bits de la etiqueta de flujo en la cabecera IPV6 puede ser utilizado por una fuente para marcar paquetes que solicitan el manejo especial por los routers IPV6, tal como "nondefault QoS" o servicio de tiempo real. Una etiqueta de flujo es asignada a un flujo por el nodo de la fuente del flujo. Entre un emisor y un receptor, puede haber múltiples flujos activos en paralelo, junto con el cambio de paquetes sin requisitos de QoS. Nuevas etiquetas de flujo deben ser escogidas al azar de la gama 00001 a FFFFF

Los host o los routers que no soportan las funciones del campo de Etiqueta de Flujo (la mayor parte de aplicaciones actuales, que no serán modificadas para utilizar la etiqueta del flujo, o que no necesita el manejo de QoS) son requeridos para poner en valor de "todo cero" al mandar un paquete, para pasar el campo sin cambiar al reenviar un paquete, y para ignorar el contenido de campo al recibir un paquete.

Todos los paquetes que pertenecen al mismo flujo deben ser mandados con la misma dirección de la fuente de IP, con la dirección del Destino de IP, con los puertos idénticos de la fuente y el destino, y con una etiqueta del flujo "no cero". Si cualquiera de estos paquetes incluye un encabezamiento de Opciones de Salto Por Salto, todos deben ser originados con el mismo contenido de encabezamiento de Opciones de Salto Por Salto (excluyendo el campo de próximo encabezamiento de la cabecera de opciones de salto por salto, que es permitido diferir). Si cualquier paquete incluye una extensión de cabecera, todos estos deben ser creados con el mismo contenido en todas las extensiones de cabecera, incluyendo la extensión de cabecera que ruteo (otra vez excluyendo el campo de siguiente cabecera en la extensión de cabecera de ruteo). Los routers o los receptores son permitidos de verificar que estas condiciones sean satisfactorias. Si una infracción de estas reglas de la consistencia es detectada, un mensaje de error correspondiente es devuelto, indicando la ubicación exacta de la infracción de la regla.

El manejo de la etiqueta del flujo en routers es eficiente, y cuando se utiliza IPsec, está siempre disponible porque el encabezamiento IPV6 no es encriptado por ESP ni autentificado por AH (en el modo del transporte). Esto implica que la integridad de la información en los campos DS no puede ser garantizado por IPsec.

Un flujo está definido como una sucesión de paquetes de un emisor a un unicast específico, anycast, o la dirección de multicast marcada como un flujo por el emisor. Un flujo no es asociado necesariamente con una conexión del transporte. Un host corriendo múltiples sesiones con otro host debe ser capaz de asignar una etiqueta diferente del flujo a cada sesión. La especificación original define un flujo basado en cinco criterios, la nueva especificación define un flujo basado en tres criterios (Direcciones de etiqueta de flujo de fuente y destino). La razón para esto es que estos tres campos están siempre disponibles para el examen por routers, mientras que la fuente y el número del puerto del destino pueden ser escondidos por ESP.

4.4.3 Extensiones de Cabecera IPV6

Como se hablo anteriormente, dos extensiones de cabecera IPV6 se pueden utilizar para señalar los requisitos QoS:

La extensión de cabecera de ruteo se puede utilizar para solicitar una ruta específica, indicando una sucesión de nodos para ser utilizada ("una ruta de fuente floja" en terminología IPV4). Sin embargo, el uso de esta extensión de cabecera requiere que el solicitante tenga conocimiento acerca de la ruta preferida (es decir, la topología de la red y parámetros sensibles de QoS, etc.). Para prevenir los ataques en el sistema de ruteo, un paquete es mandado en respuesta a un paquete recibido que incluye una cabecera de ruteo que no debe incluir un cabecera de ruteo que es generada automáticamente "invirtiendo" la cabecera de ruteo recibida (como a menudo es hecho en IPV4 una ruta de fuente floja) a menos que la integridad y la autenticidad de la dirección recibida de la fuente IP y cabecera de ruteo se pueda verificar.

La cabecera de opciones de Salto Por Salto puede ser utilizada para transportar un máximo de una alarma de señalización de mensaje de un router por paquete IP (RFC 2711) a cada router en la vía de tráfico sensible a QoS, indicando que cada router debe procesar específicamente el paquete de IP. El uso de la cabecera de Opciones de Salto Por Salto permite un procesamiento rápido por el router porque ningún análisis de cabeceras de protocolo de más alto-nivel es requerido. Los routers que no son capaces de reconocer la opción de alerta de router son requeridos para ignorar esta opción y seguir el procesamiento del encabezamiento. También, a los routers no se les permite cambiar la opción mientras el paquete está

en tránsito. Los tipos de la alarma del router que han sido definidos son mostrados en la siguiente tabla:

Value	Description
0	IP packet contains a Multicast Listener Discovery message.
1	IP packet contains an RSVP message.
2	IP packet contains an Active Networks messagethe sender is attempting to load a program into the router for executing customized functions.
3-35	IP packet contains an Aggregated Reservation Nesting Level (RFC 3175, RSVP)
36-65,535	Reserved to IANA for future use.

Fig. 4.7 Tipos de Alarma de Router

4.4.4 Arquitectura del Switch de Etiqueta IPV6 (6LSA)

Una nueva propuesta que utiliza el campo de Etiqueta de Flujo es la Arquitectura de Switch de Etiqueta IPV6 (6LSA). Un nuevo uso del campo de Etiqueta de Flujo es propuesto, la arquitectura 6LSA es descrita, y un método de paquetes obligatorios a reenviar la Equivalencia las Clases (FECs) es discutido.

La arquitectura 6LSA es semejante en muchos aspectos a la Conmutación de la Etiqueta de Multiprotocol (MPLS). Las etiquetas son asignadas a paquetes IPV6, que son utilizados para proporcionar servicios QoS a través de un dominio 6LSA. 6LSA utiliza el campo de Etiqueta de Flujo en vez de un encabezamiento de calce, tanto para llevar la información de etiqueta, evitando fragmentación y ciertos asuntos de desempeño, como para proporcionar una capa de punta a punta de 3 etiquetas para QoS.

La arquitectura 6LSA es todavía un concepto muy nuevo y necesitará atravesar la evaluación por la IETF. Mientras su aceptación y su despliegue son desconocidos, su desarrollo es un signo de que la comunidad de QoS para IPV6 continúa trabajando con fines de proporcionar mejores modelos.

CAPITULO 5
IMPLEMENTACION DE LA RED IPV6

5.1 Introducción a la red IPV6 y su configuración

5.1.1 Topologías de Red

Una topología de red define cómo están conectadas computadoras, impresoras, dispositivos de red y otros dispositivos. En otras palabras, una topología de red describe la disposición de los cables y los dispositivos, así como las rutas utilizadas para las transmisiones de datos. La topología incluye enormemente en el funcionamiento de la red.

Las redes pueden tener topología física y una topología lógica. La topología física se refiere a la disposición física de los dispositivos y los medios. Las topologías físicas más comunes son las siguientes:

- Bus
- Anillo
- Estrella
- Estrella extendida
- Jerárquica
- Malla

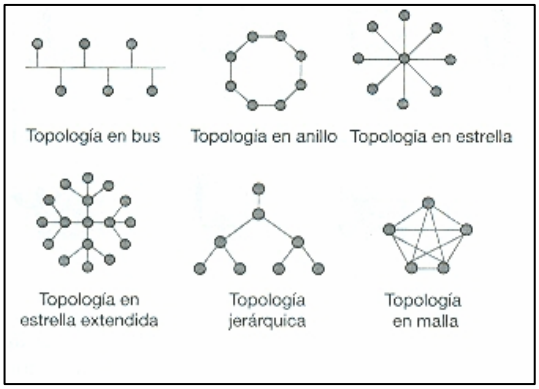


Fig. 5.1 Topologías Físicas

La topología lógica define cómo acceden los hosts a los medios para enviar datos.

La siguiente figura muestra distintas topologías conectadas por dispositivos de red e ilustra una red de complejidad media típica de un colegio o una empresa pequeña.

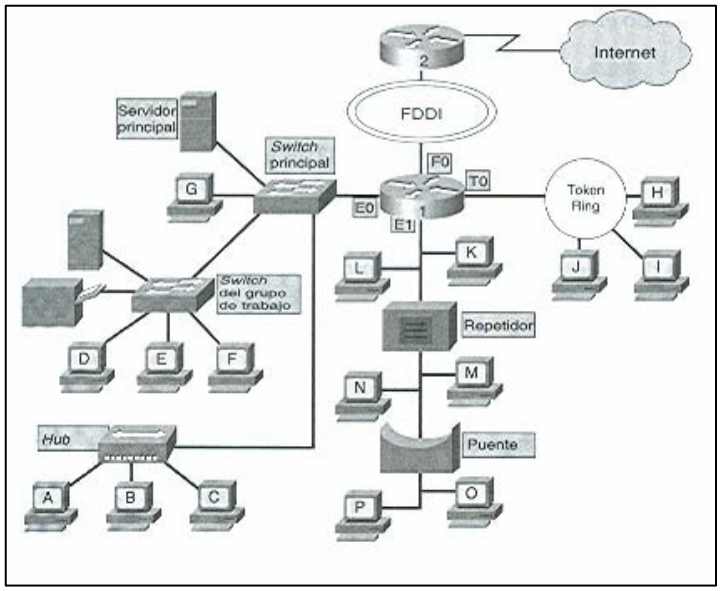


Fig. 5.2 Topologías de Red

5.1.1.1 Topología de Bus

Comúnmente conocida como bus lineal, una topología en bus conecta todos los dispositivos utilizando un solo cable. Este cable va de una computadora a la siguiente, al igual que un autobús de línea va de una ciudad a otra.

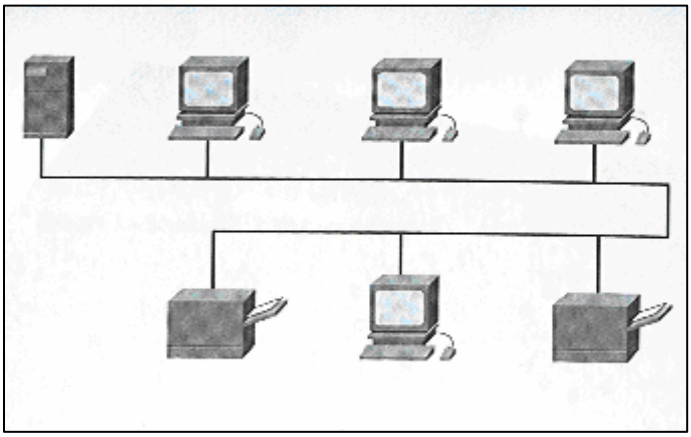


Fig. 5.3 Topología en Bus

Con una topología en bus física, el segmento de cable principal debe finalizar con un terminal que absorba la señal cuando esta alcanza el final de la línea o cable. Si no hay un terminador, la señal eléctrica que representa los datos rebotará al otro extremo del cable, provocando errores en la red.

5.1.1.2 Topologías en estrella y estrella extendida

La topología física en estrella es la más utilizada en las LAN Ethernet. Una vez instalada, la topología en estrella se parece a los radios de una rueda de bicicleta. La topología en estrella está constituida por un punto de conexión central que es un dispositivo (como un hub, un switch o un router) donde se encuentran todos los segmentos de cable. Cada uno de los hosts de la red está conectado al dispositivo central con su propio cable.

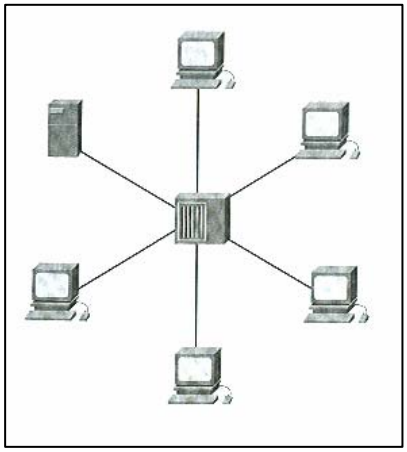


Fig. 5.4 Topología en Estrella

Aunque la implementación de una topología en estrella física es más costosa que la de la topología en bus física, sus ventajas contrarrestan ese coste adicional. Como cada host está conectado al dispositivo central con su propio cable, cuando este cable tiene un problema, sólo ese host se ve afectado; el resto de la red permanece operativa. Esta ventaja es extremadamente importante y debido a ella casi todas las nuevas LAN Ethernet que se diseñan tienen una topología en estrella física.

Un punto de conexión central podría ser deseable para la seguridad o el acceso restringido, pero esto también es un importante inconveniente de la topología estrella. Si falla el dispositivo central, la red entera se desconecta.

Cuando una red en estrella se expande para incluir un dispositivo de red adicional conectado al dispositivo de red principal, se conoce como topología en estrella extendida.

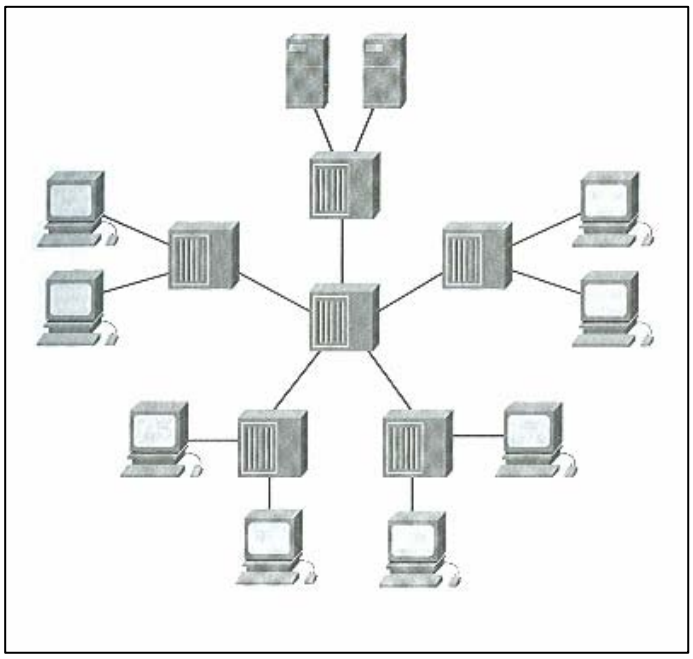


Fig. 5.5 Topología en estrella extendida

5.1.1.3 Topología en anillo

La topología en anillo lógica es otra topología importante en la conectividad LAN. Como su nombre indica, los hosts están conectados en forma de anillo o en círculo. A diferencia de la topología en bus física, la topología en anillo no tiene principio o fin que deba terminarse. Los datos se transmiten en un sentido, al contrario que en la topología en bus lógica. Una trama viaja alrededor del anillo, parando en todos los nodos. Si un nodo quiere transmitir los datos, tiene permiso de añadir esos datos, así como la dirección de destino, que extrae los datos de la trama. La ventaja de utilizar este tipo de método es que no hay colisiones de los paquetes de datos.

Hay dos tipos de anillos:

- Anillo simple
- Anillo doble

En un anillo simple, todos los dispositivos de la red comparten un solo cable y los datos viajan en una única dirección. Cada dispositivo espera su turno para enviar datos por la red. La mayoría de las topologías de anillo simple están cableadas realmente como una estrella.

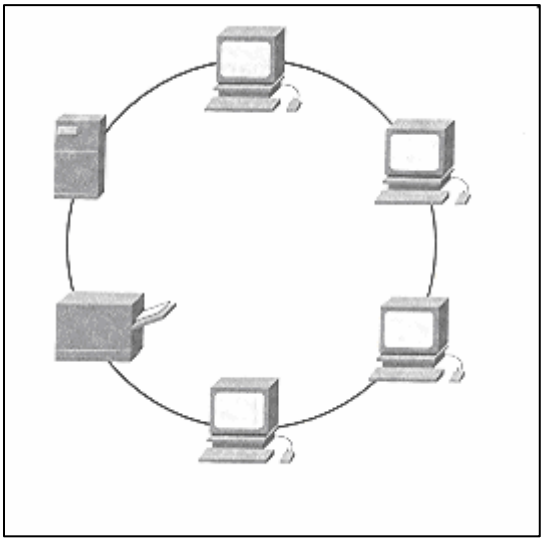


Fig. 5.6 Topología en Anillo

En un anillo doble, dos anillos permiten que los datos se envíen en ambas direcciones. Esta configuración crea redundancia (tolerancia a fallos), lo que significa que si uno de los anillos falla, los datos pueden transmitirse por el otro. Además. Si ambos anillos fallan, una “reiniciación” en el fallo puede devolver la topología a un anillo.

5.1.1.4 Topología jerárquica

Una topología jerárquica es similar a una topología en estrella extendida. La principal diferencia es que no utiliza un nodo central. En su lugar, utiliza un nodo troncal del que parten ramas a otros nodos. Existen dos tipos de topologías en árbol: el árbol binario (cada nodo se divide en dos enlaces) y el árbol backbone (un tronco backbone tiene nodos rama con enlaces colgando de él).

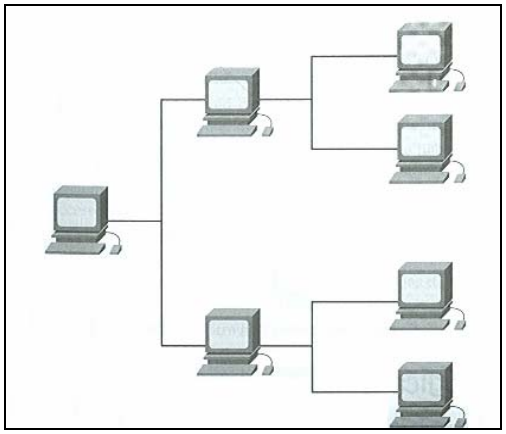


Fig. 5.7 Topología Jerárquica

5.1.1.5 Topologías en malla completa y malla parcial

La topología en malla completa conecta todos los dispositivos (nodos) con todos los demás para conseguir redundancia y tolerancia a fallos. El cableado en una topología en malla completa tiene diferentes ventajas e inconvenientes. La ventaja es que cada nodo está conectado físicamente con todos los demás, creándose una conexión redundante. Si falla cualquiera de los enlaces, la información puede fluir por otros muchos enlaces para alcanzar su destino. El principal inconveniente es que para algo más que un pequeño número de nodos, la cantidad de medios para los enlaces y el número de conexiones en las líneas puede ser abrumador. La implementación de una topología en malla completa es costosa y compleja. Normalmente se implementa en WAN entre routers.

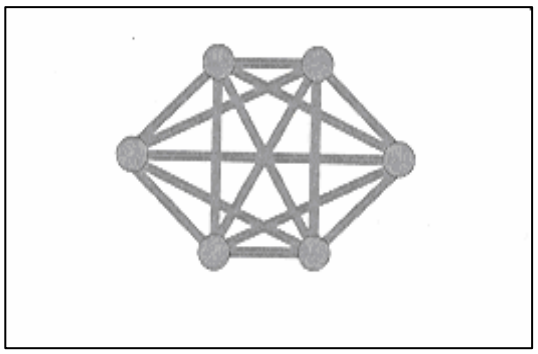


Fig. 5.8 Topología en Malla Completa

En una topología en malla parcial, al menos uno de los dispositivos mantiene múltiples conexiones con otros sin estar mallado por completo. Una topología en malla parcial todavía proporciona redundancia al contar con varias rutas

alternativas. Si una ruta no se puede utilizar, los datos toman otra diferente, aunque sea más larga. La topología en malla parcial se utiliza en muchos backbones de telecomunicaciones, así como en Internet.

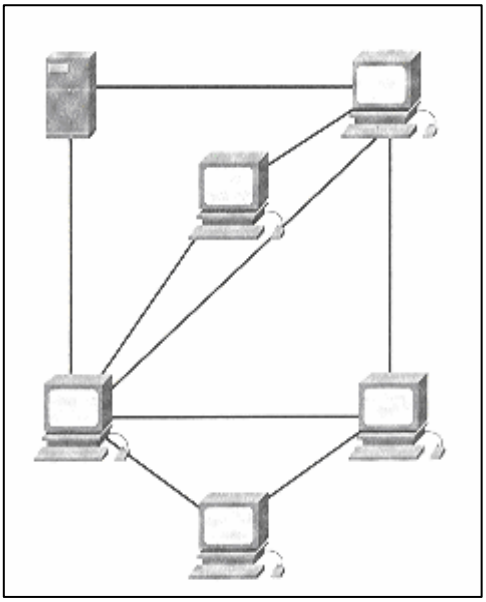


Fig. 5.9 Topología en Malla Parcial

5.1.1.6 Topología lógica

Una topología lógica de red se refiere a cómo los hosts se comunican a través del medio. Los dos tipos más comunes de topologías lógicas son difusión y transmisión de testigos.

La topología de difusión simplemente significa que cada host dirige sus datos a una NIC en particular, a una dirección de multidifusión o a una dirección de difusión en el medio de red. No hay un orden que las estaciones deban seguir para utilizar la red. El primero que llega es el primero que sirve. Ethernet también funciona de este modo.

La segunda topología lógica es la transmisión de testigos, que controla el acceso a la red pasando un testigo electrónico secuencialmente a cada host. Cuando un host recibe el testigo, puede enviar datos por la red. Si el host no tiene datos que enviar, pasa el testigo al siguiente host, y el proceso se vuelve a repetir. Token Ring y FDDI son dos ejemplos de redes que utilizan la transmisión de testigos, y ambas son ejemplos de transmisión de testigos en una topología en anillo física.

5.1.2 Dispositivos de Red

El equipamiento conectado directamente a un segmento de red se denomina dispositivo. Los dispositivos se dividen en dos clasificaciones:

5.1.2.1 Dispositivo de usuario final. Incluyen computadoras, impresoras, escáneres, y otros dispositivos que proporcionan servicios directamente al usuario.

5.1.2.2 Dispositivo de red. Abarcan todos los dispositivos que conectan los dispositivos de usuario final para permitir que se comuniquen.

Los dispositivos de usuario final que proporcionan al usuario una conexión a la red también se conoce como hosts.

Los dispositivos de red proporcionan el transporte de los datos que deben ser transferidos entre dispositivos de usuario final. Los dispositivos de red extienden las conexiones por cable, concentran las conexiones, convierten los formatos de los datos y administran las transferencias de datos. Ejemplos de dispositivos que realizan estas funciones son los repetidores, los hubs, los puentes switches y los routers. Las siguientes secciones ofrecen una visión general de algunos de los dispositivos de red más comunes.

5.1.3 Tarjeta de interfaz de red.

Las tarjetas de interfaz de red (NIC) están consideradas como dispositivos de la capa 2 porque cada una de ellas tiene un código único, denominado dirección de control de acceso al medio (MAC, Media Access Control). Dicha dirección controla la comunicación de datos para el host en la LAN. Las NIC controlan el acceso del host al medio. La siguiente figura muestra una NIC

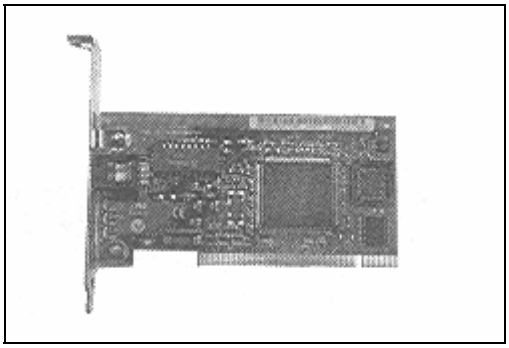


Fig. 5.10 Tarjeta de Interfaz de Red

5.1.4 Switches

Los switches (o conmutadores) de capa 2, también conocidos como switches LAN o switches de grupo de trabajo, a menudo sustituyen a los hubs compartidos y trabajan con las infraestructuras de cables existentes para garantizar que los switches estén instalados con el mínimo de alteración de las redes existentes.

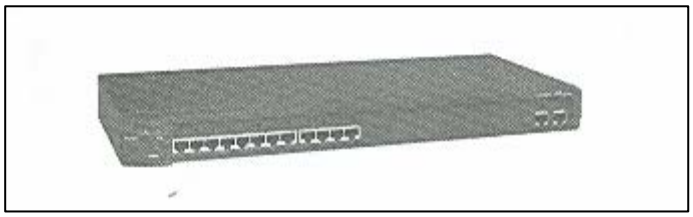


Fig. 5.11 Switch

Los switches o dispositivos de la capa de enlace de datos que, al igual que los puentes, permiten interconectar múltiples segmentos de LAN físicos en redes sencillas más grandes. De forma similar a los puentes, los switches remiten e inundan el tráfico en base a las direcciones MAC. Como la conmutación se lleva a cabo en el hardware, es significativamente más rápido que la función de conmutación la realice un puente utilizando software. Piense en un puerto de switch como en un micropuente. Cada puerto de switch actúa como un puente separado y proporciona a cada host el ancho de banda completo del medio. Este proceso se conoce como microsegmentación.

La microsegmentación permite la creación de segmentos privados o dedicados: un host por segmento. Cada host recibe acceso instantáneo al ancho de banda completo y no tiene que competir con otros hosts por un ancho de banda disponible. En los switches dúplex (full-duplex), como sólo un dispositivo está conectado a cada uno de los puertos del switch no se producen colisiones.

Sin embargo, como ocurre con un puente, un switch remite un mensaje de difusión a todos los segmentos del switch. Por consiguiente, se considera que todos los segmentos en un entorno conmutado están en el mismo dominio de difusión.

Fig. 5.12 Switch Cisco Catalyst 8500

5.1.5 Routers

Un router es un tipo de dispositivo de internetworking que pasa paquetes de datos entre redes basándose en direcciones de la capa 3. Un router puede tomar decisiones acerca de la mejor ruta para la distribución de datos por la red.

Trabajar en la capa 3 permite al router tomar decisiones basándose en las direcciones de red, en lugar de las direcciones MAC individuales de la capa 2. Los routers también pueden conectar diferentes tecnologías de capa 2, como Ethernet, Token Ring y FDDI (Interfaz de dados distribuidos por fibra). Los routers también se conectan con frecuencia con conexiones en serie y ATM (Modo de transferencia asíncrono). Sin embargo, debido a su capacidad de enlutar paquetes en base a la información de la capa 3, los routers se han convertido en el backbone de Internet y ejecutan el protocolo IP.

El propósito de un router es examinar los paquetes entrantes (datos de la capa 3), elegir la mejor ruta para ellos a través de la red y, después, conmutarlos al puerto de la salida apropiado. Los routers son el dispositivo regulador del tráfico más importante en las redes grandes. Virtualmente, permiten que cualquier tipo de computadora se comunique con otra en cualquier parte del mundo.

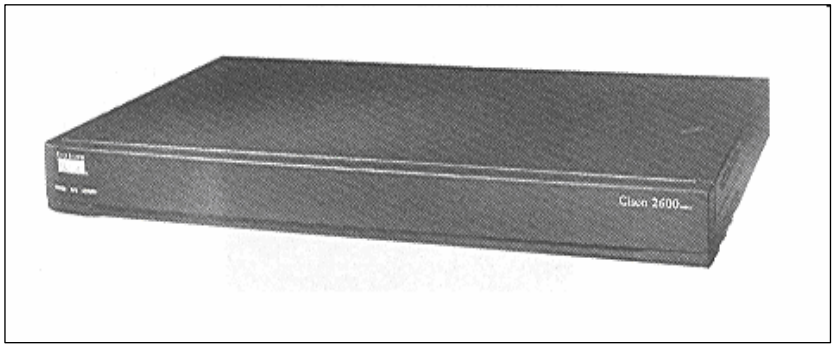


Fig. 5.13 Router

5.2 Implementación

La instalación de IPV6 en la plataforma Windows XP (la cual se utilizara), se debe realizar mediante comandos en una ventana de DOS, ya que no permite una configuración de modo gráfico del protocolo, a diferencia del sistema operativo Windows Vista, el cual ofrece una interfaz gráfica.

5.2.1 Instalación de IPV6

En una ventana de DOS se ingresa el comando “ipv6 install”.

Fig. 5.14 IPV6 Install

Para comprobar mediante comandos que la instalación se haya realizado correctamente se ejecuta el comando “ipconfig” el cual muestra la dirección asignada automáticamente y las diferentes seudointerfaces asociadas al funcionamiento de IPV6. También se puede utilizar el comando “ipv6 if”

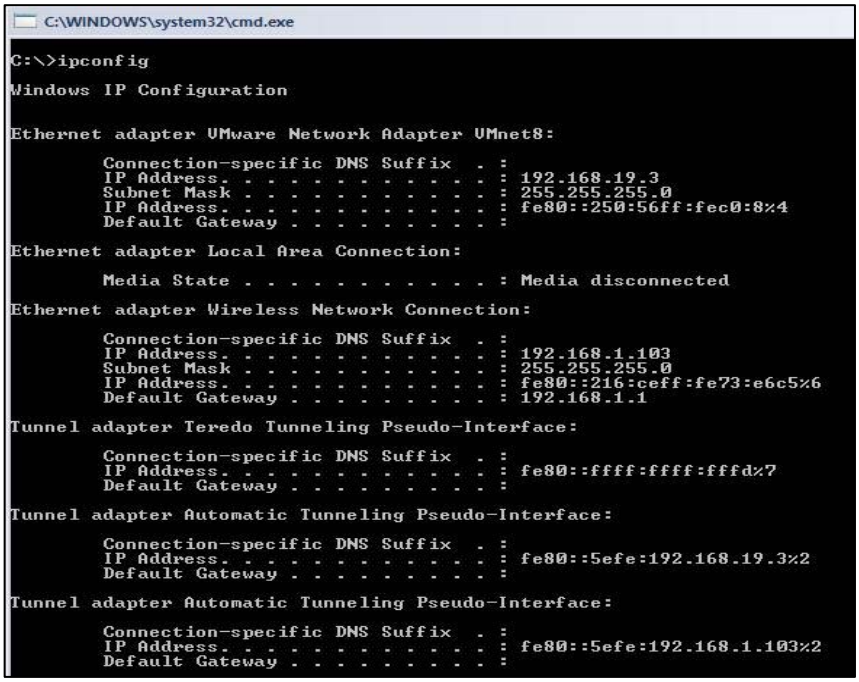


Fig. 5.15 Ipconfig

Para revisar si IPV6 se encuentra instalado de forma gráfica, se lo puede revisar en propiedades de las conexiones de red.

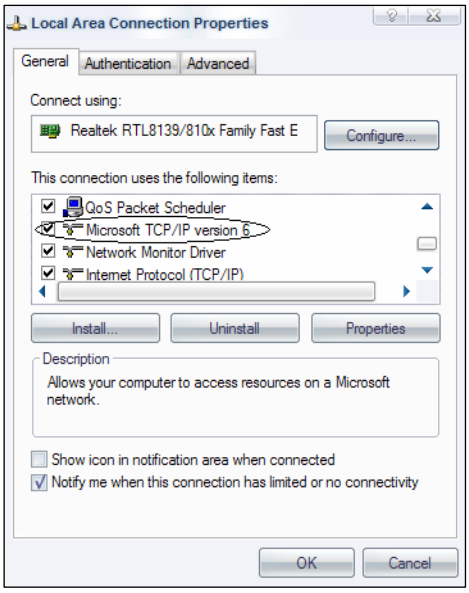


Fig. 5.16 Propiedades de Conexiones de Red

Ningún cambio en la configuración del protocolo podrá realizarse de manera gráfica. Para desinstalar el protocolo de utiliza el comando “ipv6 uninstall”.

5.2.2 Modos de configuración en XP/2003

Sirven para obtener información sobre el estado y realizar la configuración de interfaces, direcciones, caches, rutas, etc.

Dos grupos de comandos:

– ipv6.exe (hasta Windows XP SP1)

Algunos cambios no son permanentes y se pierden cuando se reinicia el PC. Se pueden ejecutar en cada inicio con un script.cmd

– netsh interface ipv6 (desde Windows XP SP2 y Server 2003)

5.2.3 Pruebas de conectividad

Se utiliza el siguiente formato del comando para realizar un ping:

• ping6 [-t] [-a] [-n count] [-l size] [-w timeout] [-s srcaddr] [-r] dest

– t Ping the specifed host until interrupted
– a Resolve addresses to hostnames
– n Number of echo requests to send
– l Send buffer size
– w Timeout in milliseconds to wait for each reply
– s Source address to use
– r Use routing header to test reverse route also

Para comprobar que IPV6 funciona correctamente en la propia PC (loopback), se ejecuta en una ventana de consola:

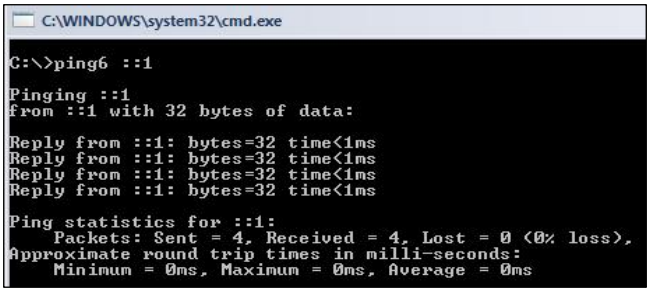


Fig. 5.17 Ping Loopback

Esto significa que IPV6 esta instalado correctamente y es funcional. Como se puede notar en el ejemplo anterior, el comando ping es sustituido por el comando ping6 para comprobar conexión en IPV6, tanto en loopback como en computadoras conectadas en red como se muestra en la figura.

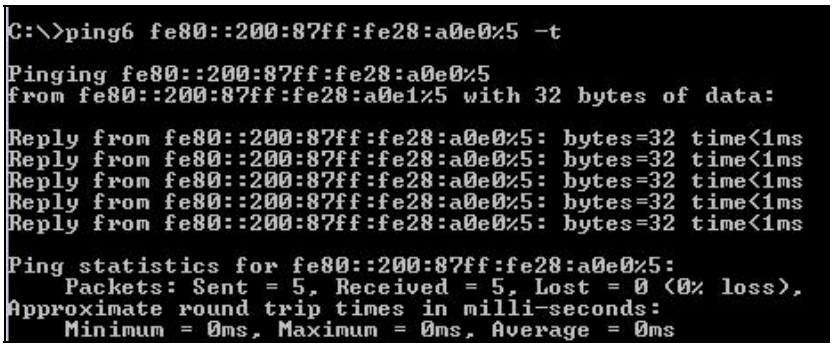


Fig. 5.18 Ping6 a una dirección

5.2.4 Configuración manual de una IP

Para setear una dirección manualmente se realiza mediante el comando:

ipv6 adu ifindex/dirección

Donde “ifindex” es un valor numérico que identifica la interfaz a la que se le esta aplicando la configuración.

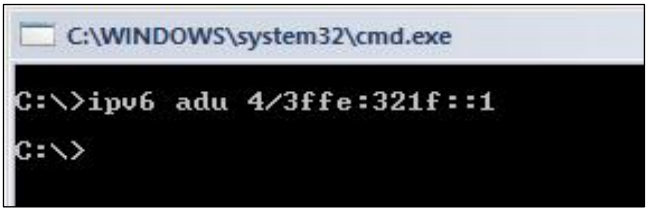


Fig. 5.19 Seteo Manual de una dirección IP

Para definir la mascara de red se utiliza el comando “ipv6 rtu red/prefijo ifindex”

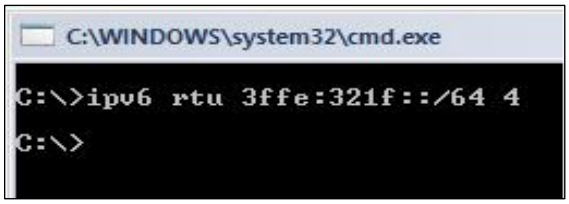


Fig. 5.20 Definición de Máscara de Red

5.3 Comparaciones entre redes en IPV4 e IPV6

Tema	IPV4	IPV6	Ventajas IPV6
Espacio de direccionamiento	4294967296 direcciones	3.40282366920938E+38 direcciones	Prácticamente espacio ilimitado
Tamaño de la dirección	32 bits	128 bits	Mas cantidad de direcciones
Notación	4 grupos decimales de 0 a 255 separados por puntos	8 grupos hexadecimales de 0 a FFFF separados por 2 puntos	Notación hexadecimal (no se necesita conversión)
Cabecera	12 campos	8 campos	Mejora la eficiencia en el procesamiento de los paquetes
Configuración	Manual o DHCP	Universal Plug and Play (UPnP) con o sin DHCP	Menor gasto de operaciones y reducción de errores
Broadcast / Multicast	Usa ambos	No se usa Boadcast; existen diferentes formas de multicast	Mejor eficiencia en el ancho de banda
Soporte Anycast	No	Completo	Soporta nuevas aplicaciones móviles
Configuración de la red	Gran parte manual	Facilita la renumeración de hosts y routers	Fácil migración y menor necesidad de operación
Soporte de calidad de servicio (QoS)	Tipos de servicio (Tos) usando diferentes servicios	Clase de flujo y etiquetas de flujo	Mayor control de QoS
Seguridad	Usa IPSec para la protección de paquetes	IPSec se usa como llave tecnológica para la protección de datos y el control de paquetes	Red unificada para la seguridad con o cual se genera un ambiente computacional de mayor seguridad
Movilidad	Difícil implementar	Optimización de enrutamiento y movilidad jerárquica	Mejor eficiencia y escalabilidad

Tabla 5.1 Comparaciones entre redes entre redes en IPV4 e IPV6

5.4 Migración IPV4 a IPV6.

IPV6 e IPV4 coexistirán por varios años, esto implica que una amplia variedad de técnicas hagan posible la coexistencia y faciliten la transición. Estas técnicas están separadas dentro de tres categorías principales:

Técnicas "Dual-stack"

Permiten que IPV4 e IPV6 coexistan en los mismos dispositivos y redes

Técnicas de *Tunneling*

Permiten el transporte de trafico IPV6 sobre la infraestructura IPV4 existente

Técnicas de Transición

Permiten que nodos que solo soportan IPV6 se comuniquen con nodos que solo soportan IPV4

La migración a IPV6 va a ser un proceso que se puede realizar paso a paso, empezando por un simple host o una subred, a continuación se analizan algunas de las técnicas más utilizadas que nos pueden ayudar a cumplir este propósito.

5.4.1 Técnicas "Dual-stack" (doble pila)

Un nodo dual-stack tiene un soporte para ambas versiones del protocolo. A este tipo de nodo a menudo se lo conoce como un nodo IPV6/IPV4. Las implementaciones deben contar con un control de la configuración que habilite o deshabilite una de las pilas, por lo tanto este tipo de nodo puede tener tres modos de operación. Cuando la pila IPV4 es habilitada y la IPV6 deshabilitada el nodo se comporta como un nodo solo-IPV4. Cuando la pila IPV6 es habilitada y la IPV4 deshabilitada el nodo se comporta como un nodo solo-IPV6. Cuando ambas pilas la IPV4 y la IPV6 están habilitadas, el nodo puede usar ambos protocolos. Un nodo IPV6/IPV4 tiene por lo menos una dirección para cada versión de protocolo, este utiliza mecanismos IPV4 para la configuración de una dirección IPV4 (configuración estática o DHCP) y utiliza mecanismos IPV6 para configurar una dirección IPV6 (configuración estática ó autoconfiguración).

Una red dual-stack es una infraestructura en la cual tanto los paquetes IPV4 como IPV6 son procesados por los routers. La desventaja de esta técnica es que se debe

contar con una completa actualización del software de la red para correr las dos pilas separadas. Para el manejo de la red, en algunos sistemas operativos se tiene comandos separados dependiendo del protocolo (por ejemplo, ping para IPV4 y ping6 para IPV6), y esto implica contar con mas memoria y procesamiento del CPU.

5.4.2 Técnicas de *Tunneling*.

Los mecanismos de *tunneling* pueden ser usados para desarrollar una infraestructura de reenvió IPV6 manteniendo como base la infraestructura inicial IPV4 asegurando que esta no debe ser modificada o actualizada. El *tunneling* es también conocido como encapsulación. Con la encapsulación un protocolo (en este caso, IPV6) es encapsulado en una cabecera de otro protocolo (en este caso, IPV4) y procesado sobre la infraestructura del segundo protocolo (IPV4). Existen 2 tipos diferentes de *tunneling*:

Tunneling configurado manualmente de IPV6 sobre IPV4

Los paquetes IPV6 son encapsulados en paquetes IPV4 para ser transportados sobre la infraestructura de enrutamiento IPV4. Este es un túnel punto a punto que necesita ser configurado manualmente.

Tunneling automático de IPV6 sobre IPV4

El nodo IPV6 puede usar diferentes tipos de direcciones, tales como direcciones 6to4 ó ISATAP, para que dinámicamente se transporte por la infraestructura de enrutamiento IPV4 los paquetes del túnel IPV6. Estas direcciones unicast especiales IPV6 transportan una contienen IPV4 en algún parte de los campos de la dirección IPV6.

5.4.3 Mecanismos de Transición.

En este tema se describe mecanismos de transición que están disponibles hoy en día. En base a un análisis del entorno y de los requerimientos se puede determinar la herramienta óptima o la combinación de herramientas que permita cumplir el objetivo.

5.4.3.1 6to4 (RFC 3056)

Específica un mecanismo para que sitios IPV6 se comuniquen con otro sobre una red IPV4 sin una configuración explicita de túnel. Este mecanismo es llamado 6to4. La amplia área de redes IPV4 es tratada como una capa de enlace punto a punto unicast, y el dominio nativo IPV6 se comunica mediante routers 6to4, también conocidos como gateways 6to4. Hay que tomar en cuenta que solo se necesita que el gateway maneje el mecanismo 6to4. No deben realizarse cambios a los host dentro de una red 6to4. Se entiende que este mecanismo de transición será usado durante el periodo de coexistencia de IPV4 e IPV6, no será usado como una solución permanente. Los paquetes IPV6 son encapsulados en IPV4 por el gateway 6to4. Anteriormente en el capitulo 3 se hablo del formato de este tipo de direcciones.

5.4.3.2 ISATAP

El protocolo “Intra-Site Automatic Tunnel Addressing Protocol” (ISATAP) fue diseñado para proveer conectividad IPV6 para nodos dual-stack sobre una red basada en IPV4. Este protocolo trata las redes IPV4 como una gran capa de enlace de red y permite que los nodos dual-stack tengan entre ellos un túnel automático. Se puede utilizar el mecanismo de *tunneling* automático sin importar si te utiliza direcciones IV4 privadas ó globales. La dirección ISATAP contiene la dirección IPV4 dentro del identificador de la interface EUI-64 (descrito en el capitulo 3).

5.4.3.3 Teredo

6to4 hace que IPV6 funcione sobre una infraestructura IPV4 usando direcciones públicas IPV4. ISATAP permite el funcionamiento de host IPV6 dentro de un sitio sin importar si se utiliza direcciones IPV4 privadas ó publicas. Teredo fue diseñado para permitir que IPV6 este disponible para hosts que estén detrás de uno o mas NATs por el *tunneling* de los paquetes sobre UDP. Muchos usuarios de Internet, especialmente usuarios residenciales, pueden acceder al Internet solo a través de NATs (Network Address Translation).

5.5 Ruteo en IPV6

Al igual que en IPV4 se maneja un enrutamiento entre dominios sin clase (CIDR). Las versiones mas recientes de los protocolos manejan direcciones IPV6 y las diferentes estructuras de cabecera. En la figura se puede observar cuales son los protocolos de ruteo disponibles para IPV6.

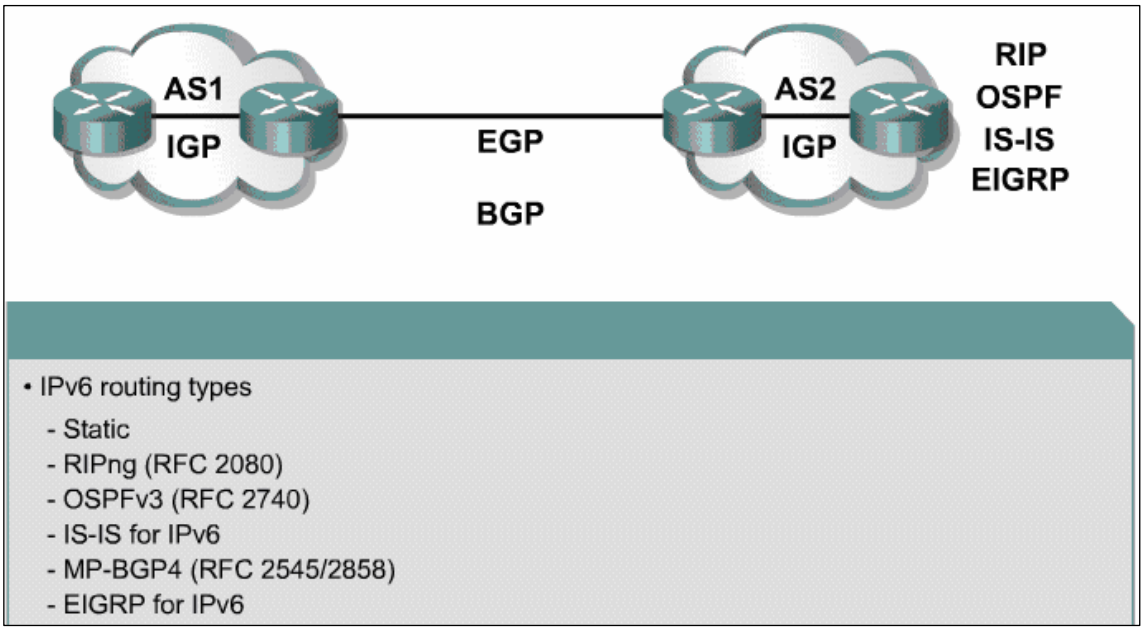


Fig. 5.21 Protocolos de Ruteo

5.5.1 Ruteo Estático

El ruteo estático es usado y configurado de la misma forma que IPV4. El router debe ser capaz de determinar una dirección de link local para cada uno de sus routers vecinos para asegurarse de que la dirección de la interfaz del mensaje redireccionado sea la del router vecino correcto. Este requerimiento significa básicamente usar una dirección unicast global como dirección de próximo salto lo cual no es recomendado dentro del enrutamiento.

5.5.2 RIPng

El protocolo de información de enrutamiento de próxima generación (Routing Information Protocol next generation) es un protocolo de enrutamiento de vector distancia con un limite de 15 saltos y utiliza las técnicas del horizonte dividido y

posición reversa (envenenamiento) para evitar formar bucles de enrutamiento. Como principales características se pueden citar las siguientes:

- Basado en Rip IPV4 versión 2 (RIPv2) y similar a este.
- Usa IPV6 para transporte.
- Utiliza el grupo multicast FF02::9, para todos los grupos multicast de routers RIP, como la dirección de destino para las actualizaciones RIP.
- Las actualizaciones se envían por el puerto UDP 521.

5.5.3 OSPFv3

OSPF es un protocolo de puerta de enlace interior usado para distribuir información de ruteo entre routers de un sistema autónomo simple.

La implementación de este protocolo para IPV6 incluye las siguientes características:

- Basado en OSPF versión 2 (OSPFv2), con encaminamiento.
- Distribuye prefijos IPV6.
- Corre directamente sobre IPV6

Esta implementación adiciona los siguientes atributos específicos IPV6:

- Direcciones de 128 bits.
- Direcciones de link local
- Diferentes direcciones y situaciones por interfaz
- Autenticación (ahora utiliza IPsec)
- OSPFv3 opera directamente sobre una interfaz.

5.5.4 IS-IS

El protocolo sistema intermedio a sistema intermedio (Intermediate System-to-Intermediate System) no fue desarrollado originalmente para IP si no para proveer la funcionalidad de enrutamiento entre los routers de redes basadas en CLNP (Connectionless Network Protocol). Con la adición del soporte IPV4 (RFC 1195), el protocolo, a veces referido como IS-IS integrado, fue ampliamente adoptado como el protocolo de gateway interior elegido para muchos ISPs y grandes redes empresariales.

Sistema intermedio a sistema intermedio es el mismo que en IPV4 con las siguientes extensiones añadidas:

- Dos nuevos valores de longitud de tipo (TLV)
- Manejo completo de IPV6.
- Direcciones de interfaz IPV6
- Nuevo protocolo IDS

5.5.5 EIGRP

El protocolo de puerta de enlace interior ampliado (Enhanced Interior Gateway Protocol) puede ser usado para rutear prefijos IPV6. EIGRP IPV4 corre sobre un transporte IPV4, comunicándose solo con puntos IPV4, y publicando solo rutas IPV4. EIGRP para IPV6 continua con el mismo modelo. EIGRP para IPV4 y EIGRP para IPV6 son configurados y manejados por separado. Sin embargo, la configuración para IPV4 e IPV6 es similar y provee una operación familiarizada y continuidad.

5.5.6 Multiprotocolo BGP (MP-BGP)

El protocolo de puerta de enlace fronterizo versión 4 (BGP4) es el protocolo de tipo gateway exterior EGP (Exterior gateway protocol) usado para intercambiar rutas entre los sistemas autónomos en el Internet. BGP fue diseñado basándose en la experiencia ganada con EGP e implementando soporte para CIDR (classless interdomain routing).

El enrutamiento en la red implementada es mediante una ruta estática realizada por un Linux distribución Centos que cumple el papel de router mediante la habilitación de un registro para que realice el enrutamiento entre dos interfaces ethernet instaladas en el equipo, en cada una de las cuales se encuentran las redes a ser enrutadas. Cabe recalcar que las direcciones utilizadas son de tipo unicast global para poder enrutar, asumiendo que un organismo, en nuestro caso LACNIC, nos hubiese asignado este grupo de direcciones identificadas por el Prefijo Global de Ruteo 2000:1fff::/32 y los ID de subred 2000:1fff:0:a::/64 y el 2000:1fff:0:b::/64.

El esquema es el siguiente:

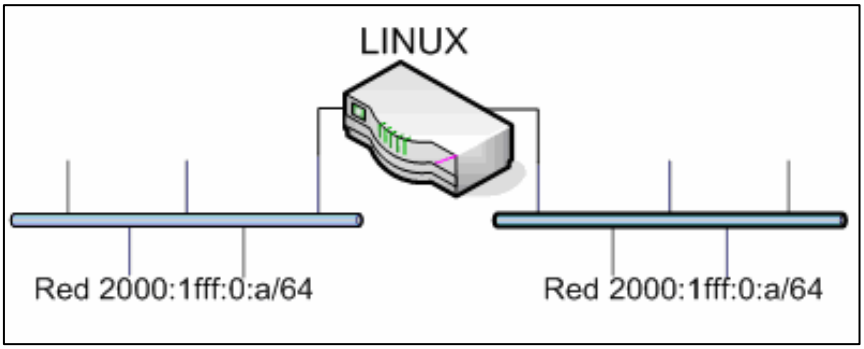


Fig. 5.22 Esquema de Red

La configuración necesaria en el sistema operativo Linux para que cumpla el papel de router es la siguiente:

- sysctl -w net.IPv6.conf.all.forwarding="1"
  Este comando habilita el ruteo en las 2 o más interfaces del equipo.

- ip addr add 2000:1fff:0:a::1/64 dev eth0
  Mediante este comando configuramos la dirección IP unicast global en la interfaz eth0

- ip addr add 2000:1fff:0:b::1/64 dev eth1
  Mediante este comando configuramos la dirección IP unicast global en la interfaz eth1

Los comandos necesarios para la configuración en el sistema operativo Windows XP se los puede realizar como se indico en el manual de implementación ó con los siguientes comandos:

- netsh
  interface ipv6
  add address 4 2000:1fff:0:a::2
  Este comando configura la dirección IP unicast global en la interfaz con index 4.

- netsh

  interface ipv6

  add route 2000:1fff:0:b::/64 4 2000:1fff:0:b::1

  Este comando define el gateway de la red ó en otras palabras crea una ruta para salir a la red vecina 2000:1fff:0:b::/64 por la interfaz local 4 a la IP 2000:1fff:0:a::1. El mismo procedimiento se debe realizar en las terminales pertenecientes a esta red y cambiando la red y el gateway para terminales de otras redes.

Por último se procede a probar conectividad mediante el comando ping6

- ping6 –I eth0 2000:1fff:0:a::2 (Linux)
- ping6 2000:1fff:0:a::2 (Windows XP)

5.3 Manual de Instalación de IPV6 en Windows XP

Instalación de IPV6

Se ejecuta el comando “ipv6 install” en una ventana de DOS, de esta manera se instala y habilita el protocolo IPV6.

Fig. 5.23 IPV6 Install

Para comprobar mediante comandos que la instalación se haya realizado correctamente se ejecuta el comando “ipconfig” el cual muestra la dirección asignada automáticamente y las diferentes seudointerfaces asociadas al funcionamiento de IPV6. También se puede utilizar el comando “ipv6 if”

```
C:\>ipconfig

Windows IP Configuration

Ethernet adapter VMware Network Adapter VMnet8:

        Connection-specific DNS Suffix  . :
        IP Address. . . . . . . . . . . . : 192.168.19.3
        Subnet Mask . . . . . . . . . . . : 255.255.255.0
        IP Address. . . . . . . . . . . . : fe80::250:56ff:fec0:8%4
        Default Gateway . . . . . . . . . :

Ethernet adapter Local Area Connection:

        Connection-specific DNS Suffix  . :
        IP Address. . . . . . . . . . . . : 192.168.1.5
        Subnet Mask . . . . . . . . . . . : 255.255.255.0
        IP Address. . . . . . . . . . . . : fe80::216:d4ff:fe12:b0f0%5
        Default Gateway . . . . . . . . . :

Ethernet adapter Wireless Network Connection:

        Connection-specific DNS Suffix  . :
        IP Address. . . . . . . . . . . . : 192.168.1.100
        Subnet Mask . . . . . . . . . . . : 255.255.255.0
        IP Address. . . . . . . . . . . . : fe80::216:ceff:fe73:e6c5%6
        Default Gateway . . . . . . . . . : 192.168.1.1

Tunnel adapter Teredo Tunneling Pseudo-Interface:

        Connection-specific DNS Suffix  . :
        IP Address. . . . . . . . . . . . : fe80::ffff:ffff:fffd%7
        Default Gateway . . . . . . . . . :

Tunnel adapter Automatic Tunneling Pseudo-Interface:

        Connection-specific DNS Suffix  . :
        IP Address. . . . . . . . . . . . : fe80::5efe:192.168.19.3%2
        Default Gateway . . . . . . . . . :

Tunnel adapter Automatic Tunneling Pseudo-Interface:

        Connection-specific DNS Suffix  . :
        IP Address. . . . . . . . . . . . : fe80::5efe:192.168.1.5%2
        Default Gateway . . . . . . . . . :

Tunnel adapter Automatic Tunneling Pseudo-Interface:

        Connection-specific DNS Suffix  . :
        IP Address. . . . . . . . . . . . : fe80::5efe:192.168.1.100%2
        Default Gateway . . . . . . . . . :
```

Fig. 5.24 Ipconfig

Para revisar si IPV6 se encuentra instalado de forma gráfica, se lo puede revisar en propiedades de las conexiones de red.

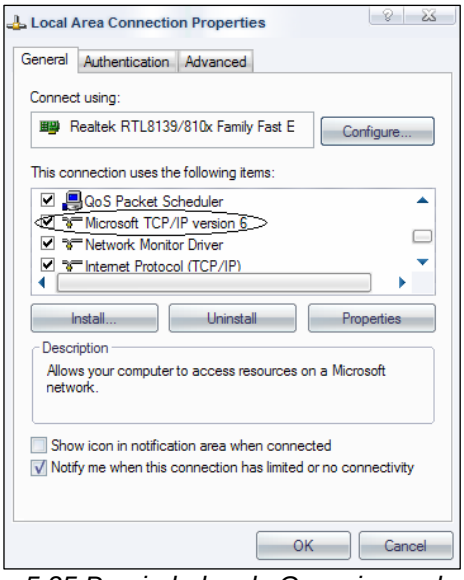


Fig. 5.25 Propiedades de Conexiones de Red

Ningún cambio en la configuración del protocolo podrá realizarse de manera gráfica. Para desinstalar el protocolo de utiliza el comando “ipv6 uninstall”.

Configuración manual de la dirección IPV6

Los parámetros que necesitamos son la dirección IP que vamos a utilizar y el index de la interfaz, este valor se lo puede obtener del comando ipconfig ubicado en la dirección de la interfaz que se va a utilizar luego del símbolo “%”, en nuestro caso 5, la sintaxis es la siguiente:

ipv6 adu [index int]/[dirección IPV6]

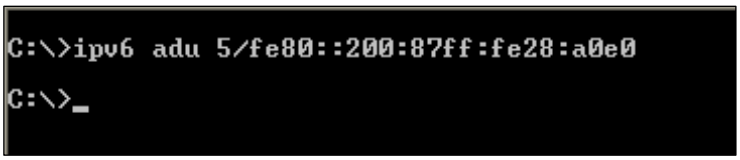


Fig. 5.26 Seteo Manual de una dirección IP

A continuación tenemos que definir cual es la red mediante su prefijo, la sintaxis es la siguiente:

ipv6 rtu [prefijo red]/[# de bits] [index int]

En el ejemplo se considera una red /64 o sea que se tomaran 64 bits para el prefijo de red lo que implica 64 bits para la dirección de red.

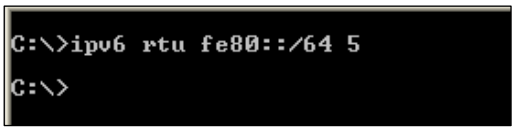
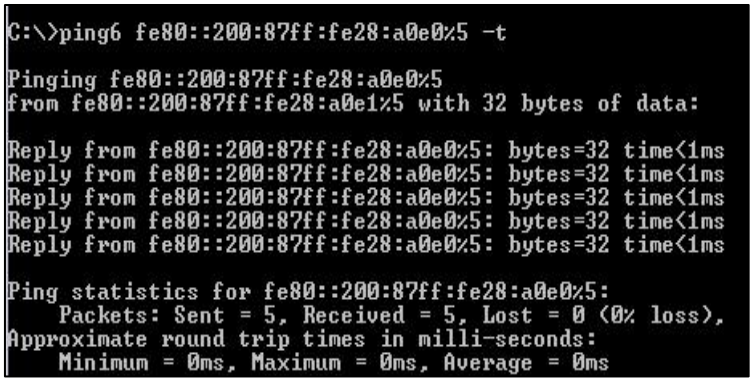


Fig. 5.27 Definición de Máscara de Red

Como siguiente paso se procede a comprobar la conectividad entre maquinas mediante el comando “ping6”, la sintaxis de este comando es la siguiente

Ping6 [dirección IPV6]%[index int]



5. 28 Ping a una dirección

CONCLUSIONES Y RECOMENDACIONES

Conclusiones

A lo largo del desarrollo de este trabajo se pudo observar que IPV6 es un protocolo mucho más completo que IPV4. Es un mecanismo pensado para el Internet actual, que ofrece muchas posibilidades, ventajas y sistemas incorporados para la optimización y eficiencia del mismo. Partiendo desde su capacidad de direcciones, hasta su eficacia en el tema de seguridad y calidad.

Una de las partes que no se puede dejar de mencionar acerca de IPV6, es la capacidad que tiene para dar soporte a IPs del protocolo IPV4, lo cual es de suma importancia, ya que resulta impensable la posibilidad de eliminar de un día a otro un protocolo tan utilizado como lo es IPV4. Para ser reemplazado IPV4 es necesario que la nueva arquitectura soporte ambos protocolos para lograr una migración gradual.

Definitivamente el protocolo IPV6 está pensado de una forma muy ambiciosa y con mucha visión a futuro. Es necesario dar soporte para las aplicaciones que en la actualidad se ejecutan en el protocolo IPV4; por lo cual, los cambios que se hicieron no son completamente diferentes, pero si determinantes para su eficiencia.

Uno de los puntos que se consideraron ampliamente para el desarrollo de este protocolo fue la capacidad de brindar soporte para dispositivos móviles. Generalmente los dispositivos móviles tienen una limitada capacidad de transmisión de datos; pero por la forma en que se enrutan las direcciones en IPV6, se reduce de forma considerable el tráfico de la red utilizando menor ancho de banda.

Con la implementación de la red utilizando el protocolo IPV6, pudimos concluir que se trata de una interfaz relativamente sencilla, ya que el sistema operativo con el que se trabajó, viene precargado con paquetes del protocolo, se manejan comandos fáciles para poder formar una red manejando direcciones IPV6.

Se alcanzaron todos los objetivos de esta tesis, ya que se logró implementar correctamente y con gran funcionalidad la red con el protocolo V6.

Recomendaciones

Las empresas, universidades, ISPs, y usuarios de Internet en general, deberían pensar en una rápida migración hacia IPV6 sabiendo que es un protocolo mucho más seguro y que lleva ya algún tiempo de aplicación a nivel mundial.

Aunque no está por demás recordar que la migración de IPV4 a IPV6 no es trivial; sobre todo porque para muchas de las empresas representa una gran inversión de dinero y un riesgo en las aplicaciones que pudieran ser no compatibles con la nueva versión del protocolo. Por este motivo se requiere un estudio, entendimiento y capacitación sobre el tema de la migración y mecanismos de transición, ya que el tiempo y el costo implicado para esta transición podrían resultar en vano el momento de suponer que toda la arquitectura y software viene preparada para soportar IPV6.

Muchas veces el temor de migrar hacia un nuevo sistema es el principal culpable de estancarse en una tecnología antigua, se debe tener en cuenta que este protocolo es un tema relativamente nuevo, a pesar de haber aparecido hace aproximadamente 10 años, se ha retomado el tema significativamente en estos años debido a la preocupación de la IANA y otros organismos por la insuficiencia de direcciones IPV4. Muchas personas están concientes de este inconveniente y buscan la mejor manera de solucionarlo, el único camino es IPV6.

Este nuevo protocolo ya es una realidad y que está al alcance de todos, no se debe esperar un colapso de una red que no fue creada para soportar tantos usuarios. Toda la nueva arquitectura y sistemas operativos actuales vienen ya acoplados para soporte IPV6, de manera de poder trabajar del mismo modo que se hacia con los sistemas y equipos IPV4.

Se recomienda a personas que quieran tratar con IPV6, un entendimiento y un estudio previo a este nuevo mecanismo, la implementación y los interfaces a pesar de parecer sencillos se necesita conocer los cambios en su estructura con respecto a IPV4, para poder implementar y configurar el protocolo de manera correcta. IPV6 es un tema tratado y estudiado por algún tiempo, no es una casualidad, hay empresas destinadas a promover y a educar a la gente sobre esto, como el caso del IPV6forum.

En resumen, IPV6 es un protocolo que tiene todas las de ganar debido a sus innumerables ventajas frente a V4, pero sobre todo se debe pensar en que es un cambio para mejorar uno de los servicios más utilizados en el mundo, el Internet, y que este protocolo puede ser el encargado de habilitar la llegada y ejecución de nuevas tecnologías con muchos beneficios para esta era.

GLOSARIO

ATM: (Asynchronous Transfer Mode - Modo de Transferencia Asíncrona) es una tecnología de telecomunicación desarrollada para hacer frente a la gran demanda de capacidad de transmisión para servicios y aplicaciones.

Backbone: Se refiere a las principales conexiones troncales de Internet. Está compuesta de un gran número de routers comerciales, gubernamentales, universitarios y otros de gran capacidad interconectados que llevan los datos entre países, continentes y océanos del mundo.

CentOS: (acrónimo de Community ENTerprise Operating System) es un clon a nivel binario de la distribución Red Hat Enterprise Linux, compilado por voluntarios a partir del código fuente liberado por Red Hat, empresa desarrolladora de RHEL.

CIDR: (Classless Inter-Domain Routing - Encaminamiento Inter-Dominios sin Clases) se introdujo en 1993 y representa la última mejora en el modo como se interpretan las direcciones IP. Su introducción permitió una mayor flexibilidad al dividir rangos de direcciones IP en redes separadas. De esta manera permitió:

- Un uso más eficiente de las cada vez más escasas direcciones IPV4.
- Un mayor uso de la jerarquía de direcciones ('agregación de prefijos de red'), disminuyendo la sobrecarga de los enrutadores principales de Internet para realizar el encaminamiento.

DCHP: (Dynamic Host Configuration Protocol) es un protocolo de red que permite a los nodos de una red IP obtener sus parámetros de configuración automáticamente. Se trata de un protocolo de tipo cliente/servidor en el que generalmente un servidor posee una lista de direcciones IP dinámicas y las va asignando a los clientes conforme éstas van estando libres, sabiendo en todo momento quién ha estado en posesión de esa IP, cuánto tiempo la ha tenido y a quién se la ha asignado después.

Gateway: Es un equipo que permite interconectar redes con protocolos y arquitecturas completamente diferentes a todos los niveles de comunicación. La traducción de las unidades de información reduce mucho la velocidad de transmisión a través de estos equipos.

Host: En informática o computación se refiere a una máquina conectada a una red de ordenadores. Puede ser un ordenador, un servidor de archivos, un dispositivo de almacenamiento por red, una máquina de fax, impresora, etc.

IANA: (Internet Assigned Numbers Authority), es la Agencia de Asignación de Números de Internet. Era el antiguo registro central de los protocolos Internet, como puertos, números de protocolo y empresa, opciones y códigos.

IETF: (Internet Engineering Task Force, en castellano Grupo de Trabajo en Ingeniería de Internet) es una organización internacional abierta de normalización, que tiene como objetivos el contribuir a la ingeniería de Internet, actuando en diversas áreas, tales como transporte, encaminamiento, seguridad. Fue creada en EE.UU. en 1986.

ISP: (Internet Service Provider - Proveedor de servicios de Internet) es una empresa dedicada a conectar a Internet a los usuarios o las distintas redes que tengan, y dar el mantenimiento necesario para que el acceso funcione correctamente. También ofrecen servicios relacionados, como alojamiento web o registro de dominios entre otros.

Jumbogram: Es una opción que permite que la longitud máxima de los datos transportados por IPV6 (16 bits, 65.535 bytes), se extienda hasta 64 bits. Se prevé su uso especialmente para tráficos multimedia, sobre líneas de banda ancha. Sin embargo estos paquetes no pueden ser fragmentados.

Loopback: es un interfaz de red virtual que siempre representa al propio dispositivo independientemente de la dirección IP que se le haya asignado.

Multicast: Multidifusión, es el envío de la información en una red a múltiples destinos simultáneamente, usando la estrategia más eficiente para el envío de los mensajes sobre cada enlace de la red sólo una vez y creando copias cuando los enlaces en los destinos se dividen. En comparación con multicast, los envíos de un punto a otro en una red se le denomina unidifusión (en inglés unicast), y el envío a todos los nodos en una red se le denomina difusión amplia.

NAT: (Network Address Translation - Traducción de Dirección de Red) es un mecanismo utilizado por routers IP para intercambiar paquetes entre dos redes que se asignan mutuamente direcciones incompatibles. Consiste en convertir en tiempo

real las direcciones utilizadas en los paquetes transportados. También es necesario editar los paquetes para permitir la operación de protocolos que incluyen información de direcciones dentro de la conversación del protocolo.

OSPF: (Open Shortest Path First) es un protocolo de encaminamiento jerárquico de pasarela interior o IGP (Interior Gateway Protocol).

PDA: Personal Digital Assistant, (Ayudante personal digital) es un computador de mano originalmente diseñado como agenda electrónica (calendario, lista de contactos, bloc de notas y recordatorios) con un sistema de reconocimiento de escritura.

RFC: Request For Comments (abreviado como RFC), que se traduce como "petición de comentarios", es un documento cuyo contenido es una propuesta oficial para un nuevo protocolo de la red Internet (originalmente de ARPANET), que se explica con todo detalle para que en caso de ser aceptado pueda ser implementado sin ambigüedades.

Cada RFC tiene un título y un número asignado, que no puede repertirse ni eliminarse aunque el documento se quede obsoleto.

TCP: (Transmission Control Protocol) fue creado entre los años 1973 - 1974 por Vint Cerf y Robert Kahn). Es uno de los protocolos fundamentales en Internet. Muchos programas dentro de una red de datos compuesta por ordenadores pueden usar TCP para crear *conexiones* entre ellos a través de las cuales enviarse un flujo de datos. El protocolo garantiza que los datos serán entregados en su destino sin errores y en el mismo orden en que se transmitieron. También proporciona un mecanismo para distinguir distintas aplicaciones dentro de una misma máquina, a través del concepto de puerto. TCP da soporte a muchas de las aplicaciones más populares de Internet, incluidas HTTP, SMTP y SSH.

Timestamping: El sellado de tiempo es un mecanismo online que permite demostrar que una serie de datos han existido y no han sido alterados desde un instante específico en el tiempo.

UDP: (User Datagram Protocol) es un protocolo del nivel de transporte basado en el intercambio de datagramas. Permite el envío de datagramas a través de la red sin que se haya establecido previamente una conexión, ya que el propio datagrama

incorpora suficiente información de direccionamiento en su cabecera. Tampoco tiene confirmación, ni control de flujo, por lo que los paquetes pueden adelantarse unos a otros; y tampoco sabemos si ha llegado correctamente, ya que no hay confirmación de entrega o de recepción. Su uso principal es para protocolos como DHCP, BOOTP, DNS y demás protocolos en los que el intercambio de paquetes de la conexión/desconexión son mayores, o no son rentables con respecto a la información transmitida, así como para la transmisión de audio y vídeo en tiempo real, donde no es posible realizar retransmisiones por los estrictos requisitos de retardo que se tiene en estos casos.

Unicast: Es un envío de información desde un único emisor a un único receptor.

BIBLIOGRAFIA

1. HAGEN Silvia / *IPV6 Essentials*, Segunda Edición / 2006 / Editorial O'Reilly.
2. FEIT Sydney / *TCP/IP* / Primera Edición / 1999 / Editorial McGraw-Hill.
3. VAN BEIJNUM Iljitsch van / *Running IpV6* / Primera Edición / 2006 / Editorial Apress.
4. POPOVICIU Ciprian / *Deploying IPV6 Networks* / 2006 / Editorial Cisco Press.
5. MALONE David, MURPHY Niall / *Network Administration* / 2005 / Editorial O'Reilly.
6. LARRABEITI David (2001) / *Comunicación de grupo en IPV6* / Manual de Linux.
7. RALLI UCENDO Carlos / *IPV6: Mecanismos de transición IPV4 – IPV6* / Folleto proporcionado en convención de LACNIC.
8. IPV6 Deployment Concepts / Editorial Cisco Press
9. Cisco Systems Inc. / *Guía del primer año CCNA I y II* / Tercera Edición / 2004 / Editorial Pearson Educación.
10. LUENGO Miguel / *Introducción a IPV6* / Guía de Redes Linux.
11. LOPEZ Alberto / *Calidad de Servicio en IPV6* / http://www.guiaslinux.org/component/option,com_remository/Itemid,0/func,fileinfo/id,159/
12. PALET Jordi / *Introducción a IPV6* / http://www.lacnic.net/documentos/lacnicix/ipv6_tutorial_jordi_palet.zip
13. Guide to Experiment on IPv6 Basic Configuration / http://www.huawei.com

ANEXOS

Anexo 1: RFC 2460

Network Working Group
Request for Comments: 2460
Obsoletes: 1883
Category: Standards Track

S. Deering
Cisco
R. Hinden
Nokia
December 1998

Internet Protocol, Version 6 (IPv6)

Specification

Status of this Memo

This document specifies an Internet standards track protocol for the Internet community, and requests discussion and suggestions for improvements. Please refer to the current edition of the "Internet Official Protocol Standards" (STD 1) for the standardization state and status of this protocol. Distribution of this memo is unlimited.

Copyright Notice

Abstract

This document specifies version 6 of the Internet Protocol (IPv6), also sometimes referred to as IP Next Generation or IPng.

Table of Contents

1 Introduction IP version 6 (IPv6) is a new version of the Internet Protocol, designed as the successor to IP version 4 (IPv4) [RFC-791]. The changes from IPv4 to IPv6 fall primarily into the following categories:

- Expanded Addressing Capabilities

IPv6 increases the IP address size from 32 bits to 128 bits, to support more levels of addressing hierarchy, a much greater number of addressable nodes, and simpler auto-configuration of addresses. The scalability of multicast routing is improved by adding a "scope" field to multicast addresses. And a new type of address called an "anycast address" is defined, used to send a packet to any one of a group of nodes.

- Header Format Simplification

Some IPv4 header fields have been dropped or made optional, to reduce the common-case processing cost of packet handling and to limit the bandwidth cost of the IPv6 header.

- Improved Support for Extensions and Options

Changes in the way IP header options are encoded allows for more efficient forwarding, less stringent limits on the length of options, and greater flexibility for introducing new options in the future.

- Flow Labeling Capability

A new capability is added to enable the labeling of packets belonging to particular traffic "flows" for which the sender requests special handling, such as non-default quality of service or "real-time" service.

- Authentication and Privacy Capabilities

Extensions to support authentication, data integrity, and (optional) data confidentiality are specified for IPv6.

This document specifies the basic IPv6 header and the initially- defined IPv6 extension headers and options. It also discusses packet size issues, the semantics of flow labels and traffic classes, and the effects of IPv6 on upper-layer protocols. The format and semantics of IPv6 addresses are specified separately in [ADDRARCH]. The IPv6 version of ICMP, which all IPv6 implementations are required to include, is specified in [ICMPv6].

2 Terminology

node - a device that implements IPv6.

router - a node that forwards IPv6 packets not explicitly addressed to itself. [See Note below].

host - any node that is not a router. [See Note below].

upper layer - a protocol layer immediately above IPv6. Examples are transport protocols such as TCP and UDP, control protocols such as ICMP, routing protocols such as OSPF, and internet or lower-layer protocols being "tunneled" over (i.e., encapsulated in) IPv6 such as IPX, AppleTalk, or IPv6 itself.

link - a communication facility or medium over which nodes can communicate at the link layer, i.e., the layer immediately below IPv6. Examples are Ethernets (simple or bridged); PPP links; X.25, Frame Relay, or ATM networks; and internet (or higher) layer "tunnels", such as tunnels over IPv4 or IPv6 itself.

neighbors - nodes attached to the same link.

interface - a node's attachment to a link.

address - an IPv6-layer identifier for an interface or a set of interfaces.

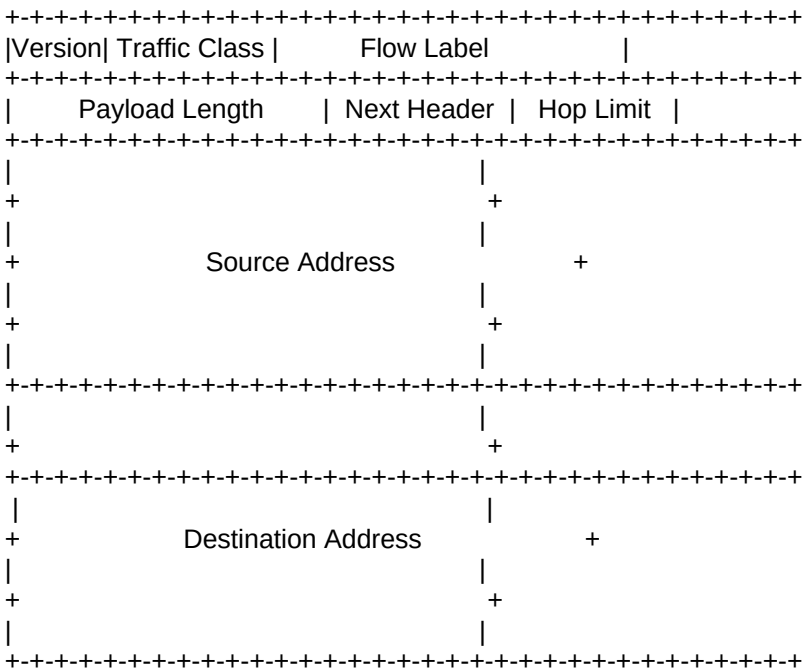
packet - an IPv6 header plus payload.

link MTU - the maximum transmission unit, i.e., maximum packet size in octets, that can be conveyed over a link.

path MTU - the minimum link MTU of all the links in a path between a source node and a destination node.

Note: it is possible, though unusual, for a device with multiple interfaces to be configured to forward non-self-destined packets arriving from some set (fewer than all) of its interfaces, and to discard non-self-destined packets arriving from its other interfaces. Such a device must obey the protocol requirements for routers when receiving packets from, and interacting with neighbors over, the former (forwarding) interfaces. It must obey the protocol requirements for hosts when receiving packets from, and interacting with neighbors over, the latter (non-forwarding) interfaces.

3 IPv6 Header Format



Version 4-bit Internet Protocol version number = 6.

Traffic Class 8-bit traffic class field. See section 7.

Flow Label 20-bit flow label. See section 6.

Payload Length 16-bit unsigned integer. Length of the IPv6 payload, i.e., the rest of the packet following this IPv6 header, in octets. (Note that any

extension headers [section 4] present are considered part of the payload, i.e., included in the length count.)

Next Header 8-bit selector. Identifies the type of header immediately following the IPv6 header. Uses the same values as the IPv4 Protocol field [RFC-1700 et seq.].

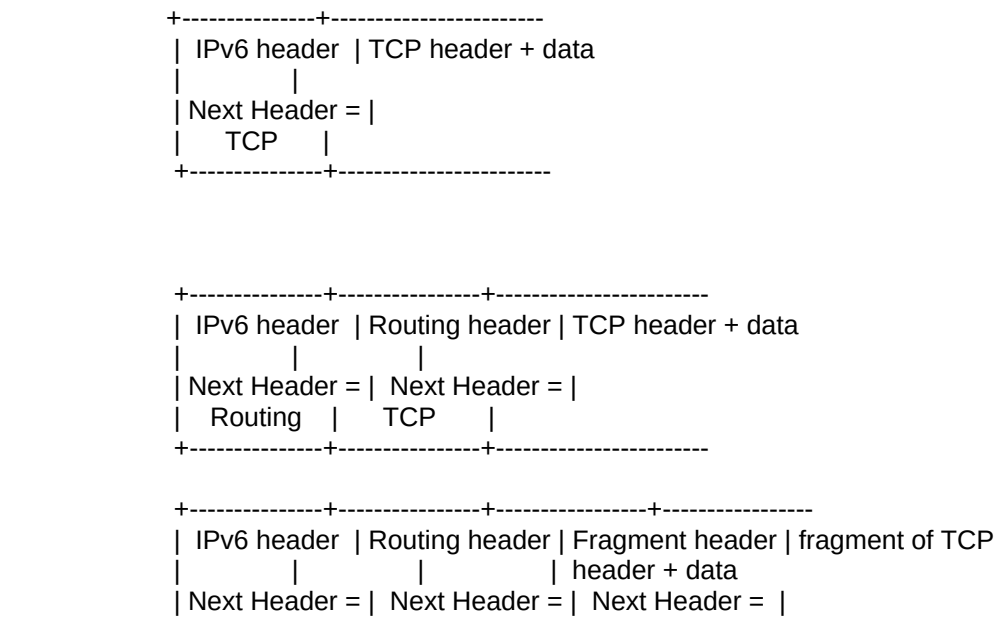
Hop Limit 8-bit unsigned integer. Decremented by 1 by each node that forwards the packet. The packet is discarded if Hop Limit is decremented to zero.

Source Address 128-bit address of the originator of the packet. See [ADDRARCH].

Destination Address 128-bit address of the intended recipient of the packet (possibly not the ultimate recipient, if a Routing header is present). See [ADDRARCH] and section 4.4.

4 IPv6 Extension Headers

In IPv6, optional internet-layer information is encoded in separate headers that may be placed between the IPv6 header and the upper- layer header in a packet. There are a small number of such extension headers, each identified by a distinct Next Header value. As illustrated in these examples, an IPv6 packet may carry zero, one, or more extension headers, each identified by the Next Header field of the preceding header:



```
| Routing   | Fragment  |     TCP      |
+-----------+-----------+--------------+----------------
```

With one exception, extension headers are not examined or processed by any node along a packet's delivery path, until the packet reaches the node (or each of the set of nodes, in the case of multicast) identified in the Destination Address field of the IPv6 header. There, normal demultiplexing on the Next Header field of the IPv6 header invokes the module to process the first extension header, or the upper-layer header if no extension header is present. The contents and semantics of each extension header determine whether or not to proceed to the next header. Therefore, extension headers must be processed strictly in the order they appear in the packet; a receiver must not, for example, scan through a packet looking for a particular kind of extension header and process that header prior to processing all preceding ones.

The exception referred to in the preceding paragraph is the Hop-by- Hop Options header, which carries information that must be examined and processed by every node along a packet's delivery path, including the source and destination nodes. The Hop-by-Hop Options header, when present, must immediately follow the IPv6 header. Its presence is indicated by the value zero in the Next Header field of the IPv6 header.

If, as a result of processing a header, a node is required to proceed to the next header but the Next Header value in the current header is unrecognized by the node, it should discard the packet and send an ICMP Parameter Problem message to the source of the packet, with an ICMP Code value of 1 ("unrecognized Next Header type encountered") and the ICMP Pointer field containing the offset of the unrecognized value within the original packet. The same action should be taken if a node encounters a Next Header value of zero in any header other than an IPv6 header.

Each extension header is an integer multiple of 8 octets long, in order to retain 8-octet alignment for subsequent headers. Multi- octet fields within each extension header are aligned on their natural boundaries, i.e., fields of width n octets are placed at an integer multiple of n octets from the start of the header, for n = 1,

2, 4, or 8.

A full implementation of IPv6 includes implementation of the following extension headers:

Hop-by-Hop Options
Routing (Type 0)
Fragment
Destination Options
Authentication
Encapsulating Security Payload

The first four are specified in this document; the last two are specified in [RFC-2402] and [RFC-2406], respectively.

4.1 Extension Header Order

When more than one extension header is used in the same packet, it is recommended that those headers appear in the following order:

IPv6 header
Hop-by-Hop Options header
Destination Options header (note 1)
Routing header
Fragment header

Authentication header (note 2)
Encapsulating Security Payload header (note 2)
Destination Options header (note 3)
upper-layer header

note 1: for options to be processed by the first destination that appears in the IPv6 Destination Address field plus subsequent destinations listed in the Routing header.

note 2: additional recommendations regarding the relative order of the Authentication and Encapsulating Security Payload headers are given in [RFC-2406].

note 3: for options to be processed only by the final destination of the packet.

Each extension header should occur at most once, except for the Destination Options header which should occur at most twice (once before a Routing header and once before the upper-layer header).

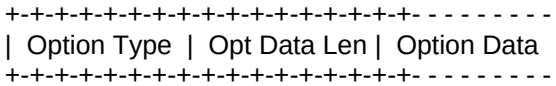
If the upper-layer header is another IPv6 header (in the case of IPv6 being tunneled over or encapsulated in IPv6), it may be followed by its own extension headers, which are separately subject to the same ordering recommendations.

If and when other extension headers are defined, their ordering constraints relative to the above listed headers must be specified.

IPv6 nodes must accept and attempt to process extension headers in any order and occurring any number of times in the same packet, except for the Hop-by-Hop Options header which is restricted to appear immediately after an IPv6 header only. Nonetheless, it is strongly advised that sources of IPv6 packets adhere to the above recommended order until and unless subsequent specifications revise that recommendation.

4.2 Options

Two of the currently-defined extension headers -- the Hop-by-Hop Options header and the Destination Options header -- carry a variable number of type-length-value (TLV) encoded "options", of the following format:



Option Type 8-bit identifier of the type of option.

Opt Data Len 8-bit unsigned integer. Length of the Option Data field of this option, in octets.

Option Data Variable-length field. Option-Type-specific data.

The sequence of options within a header must be processed strictly in the order they appear in the header; a receiver must not, for example, scan through the header looking for a particular kind of option and process that option prior to processing all preceding ones.

The Option Type identifiers are internally encoded such that their highest-order two bits specify the action that must be taken if the processing IPv6 node does not recognize the Option Type:

00 - skip over this option and continue processing the header.

01 - discard the packet.

10 - discard the packet and, regardless of whether or not the packet's Destination Address was a multicast address, send an ICMP Parameter Problem, Code 2, message to the packet's Source Address, pointing to the unrecognized Option Type.

11 - discard the packet and, only if the packet's Destination Address was not a multicast address, send an ICMP Parameter Problem, Code 2, message to the packet's Source Address, pointing to the unrecognized Option Type.

The third-highest-order bit of the Option Type specifies whether or not the Option Data of that option can change en-route to the packet's final destination. When an Authentication header is present

in the packet, for any option whose data may change en-route, its entire Option Data field must be treated as zero-valued octets when computing or verifying the packet's authenticating value.

0 - Option Data does not change en-route

1 - Option Data may change en-route

The three high-order bits described above are to be treated as part of the Option Type, not independent of the Option Type. That is, a particular option is identified by a full 8-bit Option Type, not just the low-order 5 bits of an Option Type.

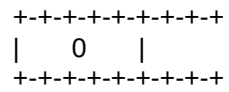
The same Option Type numbering space is used for both the Hop-by-Hop Options header and the Destination Options header. However, the specification of a particular option may restrict its use to only one of those two headers.

Individual options may have specific alignment requirements, to ensure that multi-octet values within Option Data fields fall on natural boundaries. The alignment requirement of an option is specified using the notation xn+y, meaning the Option Type must appear at an integer multiple of x octets from the start of the header, plus y octets. For example:

2n means any 2-octet offset from the start of the header.
8n+2 means any 8-octet offset from the start of the header, plus 2 octets.

There are two padding options which are used when necessary to align subsequent options and to pad out the containing header to a multiple of 8 octets in length. These padding options must be recognized by all IPv6 implementations:

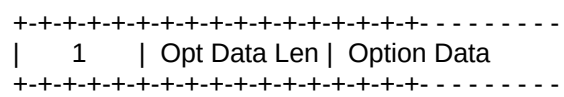
Pad1 option (alignment requirement: none)



NOTE! the format of the Pad1 option is a special case -- it does not have length and value fields.

The Pad1 option is used to insert one octet of padding into the Options area of a header. If more than one octet of padding is required, the PadN option, described next, should be used, rather than multiple Pad1 options.

PadN option (alignment requirement: none)

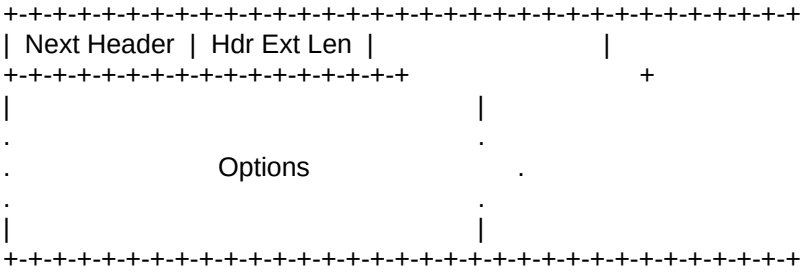


The PadN option is used to insert two or more octets of padding into the Options area of a header. For N octets of padding, the Opt Data Len field contains the value N-2, and the Option Data consists of N-2 zero-valued octets.

Appendix B contains formatting guidelines for designing new options.

4.3 Hop-by-Hop Options Header

The Hop-by-Hop Options header is used to carry optional information that must be examined by every node along a packet's delivery path. The Hop-by-Hop Options header is identified by a Next Header value of 0 in the IPv6 header, and has the following format:



Next Header 8-bit selector. Identifies the type of header immediately following the Hop-by-Hop Options header. Uses the same values as the IPv4 Protocol field [RFC-1700 et seq.].

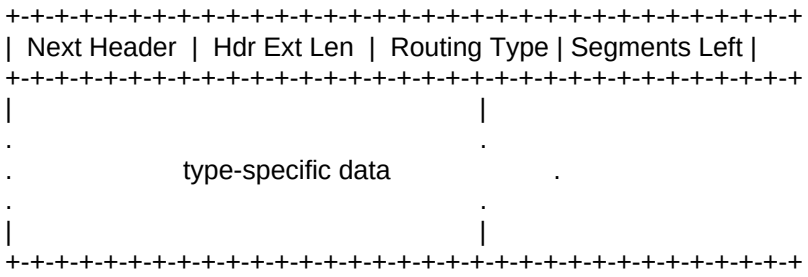
Hdr Ext Len 8-bit unsigned integer. Length of the Hop-by-Hop Options header in 8-octet units, not including the first 8 octets.

Options Variable-length field, of length such that the complete Hop-by-Hop Options header is an integer multiple of 8 octets long. Contains one or more TLV-encoded options, as described in section 4.2.

The only hop-by-hop options defined in this document are the Pad1 and PadN options specified in section 4.2.

4.4 Routing Header

The Routing header is used by an IPv6 source to list one or more intermediate nodes to be "visited" on the way to a packet's destination. This function is very similar to IPv4's Loose Source and Record Route option. The Routing header is identified by a Next Header value of 43 in the immediately preceding header, and has the following format:



Next Header 8-bit selector. Identifies the type of header immediately following the Routing header. Uses the same values as the IPv4 Protocol field [RFC-1700 et seq.].

Hdr Ext Len 8-bit unsigned integer. Length of the Routing header in 8-octet units, not including the first 8 octets.

Routing Type 8-bit identifier of a particular Routing header variant.

Segments Left 8-bit unsigned integer. Number of route segments remaining, i.e., number of explicitly listed intermediate nodes still to be visited before reaching the final destination.

type-specific data Variable-length field, of format determined by the Routing Type, and of length such that the complete Routing header is an integer multiple of 8 octets long.

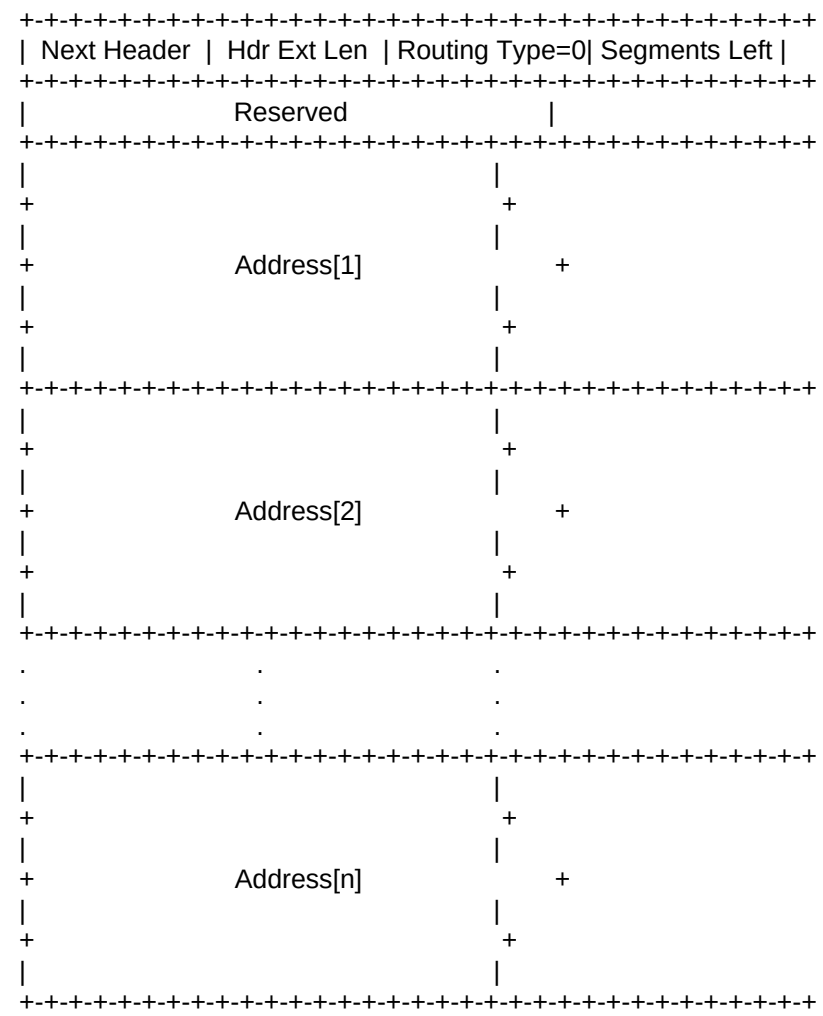
If, while processing a received packet, a node encounters a Routing header with an unrecognized Routing Type value, the required behavior of the node depends on the value of the Segments Left field, as follows:

If Segments Left is zero, the node must ignore the Routing header and proceed to process the next header in the packet, whose type is identified by the Next Header field in the Routing header.

If Segments Left is non-zero, the node must discard the packet and send an ICMP Parameter Problem, Code 0, message to the packet's Source Address, pointing to the unrecognized Routing Type.

If, after processing a Routing header of a received packet, an intermediate node determines that the packet is to be forwarded onto a link whose link MTU is less than the size of the packet, the node must discard the packet and send an ICMP Packet Too Big message to the packet's Source Address.

The Type 0 Routing header has the following format:



Next Header 8-bit selector. Identifies the type of header immediately following the Routing header. Uses the same values as the IPv4 Protocol field [RFC-1700 et seq.].

Hdr Ext Len 8-bit unsigned integer. Length of the Routing header in 8-octet units, not including the first 8 octets. For the Type 0 Routing header, Hdr Ext Len is equal to two times the number of addresses in the header.

Routing Type 0.

Segments Left 8-bit unsigned integer. Number of route segments remaining, i.e., number of explicitly listed intermediate nodes still to be visited before reaching the final destination.

Reserved 32-bit reserved field. Initialized to zero for transmission; ignored on reception.

Address[1..n] Vector of 128-bit addresses, numbered 1 to n.

Multicast addresses must not appear in a Routing header of Type 0, or in the IPv6 Destination Address field of a packet carrying a Routing header of Type 0.

A Routing header is not examined or processed until it reaches the node identified in the Destination Address field of the IPv6 header. In that node, dispatching on the Next Header field of the immediately preceding header causes the Routing header module to be invoked, which, in the case of Routing Type 0, performs the following algorithm:

```
if Segments Left = 0 {
     proceed to process the next header in the packet, whose type is
     identified by the Next Header field in the Routing header
  }
  else if Hdr Ext Len is odd {
       send an ICMP Parameter Problem, Code 0, message to the Source
       Address, pointing to the Hdr Ext Len field, and discard the
       packet
  }
  else {
     compute n, the number of addresses in the Routing header, by
     dividing Hdr Ext Len by 2

     if Segments Left is greater than n {
       send an ICMP Parameter Problem, Code 0, message to the Source
       Address, pointing to the Segments Left field, and discard the
       packet
     }
     else {
       decrement Segments Left by 1;
       compute i, the index of the next address to be visited in
       the address vector, by subtracting Segments Left from n

       if Address [i] or the IPv6 Destination Address is multicast {
         discard the packet
       }
       else {
         swap the IPv6 Destination Address and Address[i]

         if the IPv6 Hop Limit is less than or equal to 1 {
           send an ICMP Time Exceeded -- Hop Limit Exceeded in
           Transit message to the Source Address and discard the
           packet
         }
         else {
           decrement the Hop Limit by 1

resubmit the packet to the IPv6 module for transmission to the new destination
}
}
}
}
```

As an example of the effects of the above algorithm, consider the case of a source node S sending a packet to destination node D, using a Routing header to cause the packet to be routed via intermediate nodes I1, I2, and I3. The values of the relevant IPv6 header and Routing header fields on each segment of the delivery path would be as follows:

As the packet travels from S to I1:

Source Address = S	Hdr Ext Len = 6
Destination Address = I1	Segments Left = 3

Address[1] = I2
Address[2] = I3
Address[3] = D

As the packet travels from I1 to I2:

Source Address = S Hdr Ext Len = 6
Destination Address = I2 Segments Left = 2
Address[1] = I1
Address[2] = I3
Address[3] = D

As the packet travels from I2 to I3:

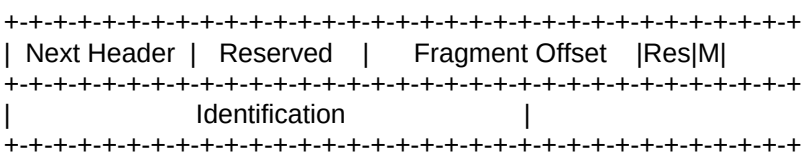
Source Address = S Hdr Ext Len = 6
Destination Address = I3 Segments Left = 1
Address[1] = I1
Address[2] = I2
Address[3] = D

As the packet travels from I3 to D:

Source Address = S Hdr Ext Len = 6
Destination Address = D Segments Left = 0
Address[1] = I1
Address[2] = I2
Address[3] = I3

4.5 Fragment Header

The Fragment header is used by an IPv6 source to send a packet larger than would fit in the path MTU to its destination. (Note: unlike IPv4, fragmentation in IPv6 is performed only by source nodes, not by routers along a packet's delivery path -- see section 5.) The Fragment header is identified by a Next Header value of 44 in the immediately preceding header, and has the following format:



Next Header 8-bit selector. Identifies the initial header type of the Fragmentable Part of the original packet (defined below). Uses the same values as the IPv4 Protocol field [RFC-1700 et seq.].

Reserved 8-bit reserved field. Initialized to zero for transmission; ignored on reception.

Fragment Offset 13-bit unsigned integer. The offset, in 8-octet units, of the data following this header, relative to the start of the Fragmentable Part of the original packet.

Res 2-bit reserved field. Initialized to zero for transmission; ignored on reception.

M flag 1 = more fragments; 0 = last fragment.

Identification 32 bits. See description below.

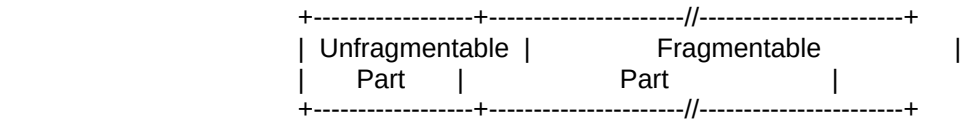
In order to send a packet that is too large to fit in the MTU of the path to its destination, a source node may divide the packet into fragments and send each fragment as a separate packet, to be reassembled at the receiver.

For every packet that is to be fragmented, the source node generates an Identification value. The Identification must be different than that of any other fragmented packet sent recently* with the same Source Address and Destination Address. If a Routing header is present, the Destination Address of concern is that of the final destination.

> * "recently" means within the maximum likely lifetime of a packet, including transit time from source to destination and time spent awaiting reassembly with other fragments of the same packet. However, it is not required that a source node know the maximum packet lifetime. Rather, it is assumed that the requirement can be met by maintaining the Identification value as a simple, 32-bit, "wrap-around" counter, incremented each time a packet must be fragmented. It is an implementation choice whether to maintain a single counter for the node or multiple counters, e.g., one for each of the node's possible source addresses, or one for each active (source address, destination address) combination.

The initial, large, unfragmented packet is referred to as the "original packet", and it is considered to consist of two parts, as illustrated:

original packet:

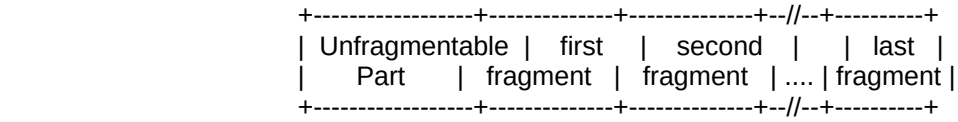


The Unfragmentable Part consists of the IPv6 header plus any extension headers that must be processed by nodes en route to the destination, that is, all headers up to and including the Routing header if present, else the Hop-by-Hop Options header if present, else no extension headers.

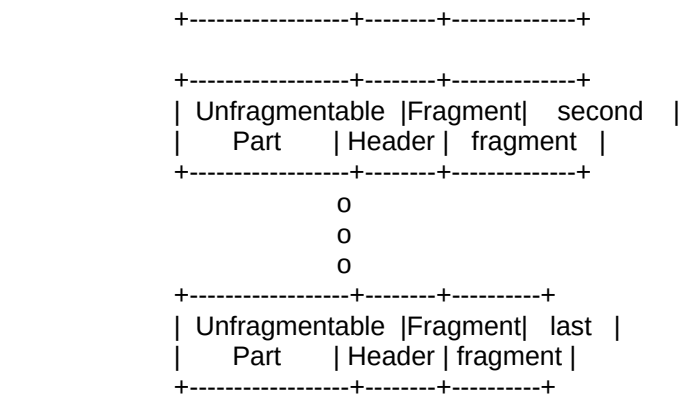
The Fragmentable Part consists of the rest of the packet, that is, any extension headers that need be processed only by the final destination node(s), plus the upper-layer header and data.

The Fragmentable Part of the original packet is divided into fragments, each, except possibly the last ("rightmost") one, being an integer multiple of 8 octets long. The fragments are transmitted in separate "fragment packets" as illustrated:

original packet:



fragment packets:



Each fragment packet is composed of:

(1) The Unfragmentable Part of the original packet, with the Payload Length of the original IPv6 header changed to contain the length of this fragment packet only (excluding the length of the IPv6 header itself), and the Next Header field of the last header of the Unfragmentable Part changed to 44.

(2) A Fragment header containing:

The Next Header value that identifies the first header of the Fragmentable Part of the original packet.

A Fragment Offset containing the offset of the fragment, in 8-octet units, relative to the start of the Fragmentable Part of the original packet. The Fragment Offset of the first ("leftmost") fragment is 0.

An M flag value of 0 if the fragment is the last ("rightmost") one, else an M flag value of 1.

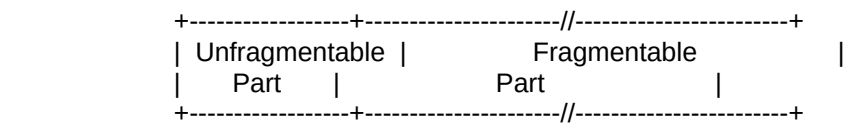
The Identification value generated for the original packet.

(3) The fragment itself.

The lengths of the fragments must be chosen such that the resulting fragment packets fit within the MTU of the path to the packets' destination(s).

At the destination, fragment packets are reassembled into their original, unfragmented form, as illustrated:

reassembled original packet:



The following rules govern reassembly:

An original packet is reassembled only from fragment packets that have the same Source Address, Destination Address, and Fragment Identification.

The Unfragmentable Part of the reassembled packet consists of all headers up to, but not including, the Fragment header of the first fragment packet (that is, the packet whose Fragment Offset is zero), with the following two changes:

The Next Header field of the last header of the Unfragmentable Part is obtained from the Next Header field of the first fragment's Fragment header.

The Payload Length of the reassembled packet is computed from the length of the Unfragmentable Part and the length and offset of the last fragment. For example, a formula for computing the Payload Length of the reassembled original packet is:

PL.orig = PL.first - FL.first - 8 + (8 * FO.last) + FL.last

where

PL.orig = Payload Length field of reassembled packet.
PL.first = Payload Length field of first fragment packet.
FL.first = length of fragment following Fragment header of first fragment packet.
FO.last = Fragment Offset field of Fragment header of last fragment packet.
FL.last = length of fragment following Fragment header of last fragment packet.

The Fragmentable Part of the reassembled packet is constructed from the fragments following the Fragment headers in each of the fragment packets. The length of each fragment is computed by subtracting from the packet's Payload Length the length of the headers between the IPv6 header and fragment itself; its relative position in Fragmentable Part is computed from its Fragment Offset value.

The Fragment header is not present in the final, reassembled packet.

The following error conditions may arise when reassembling fragmented packets:

If insufficient fragments are received to complete reassembly of a packet within 60 seconds of the reception of the first-arriving fragment of that packet, reassembly of that packet must be abandoned and all the fragments that have been received for that packet must be discarded. If the first fragment (i.e., the one with a Fragment Offset of zero) has been received, an ICMP Time Exceeded -- Fragment Reassembly Time Exceeded message should be sent to the source of that fragment.

If the length of a fragment, as derived from the fragment packet's Payload Length field, is not a multiple of 8 octets and the M flag of that fragment is 1, then that fragment must be discarded and an ICMP Parameter Problem, Code 0, message should be sent to the source of the fragment, pointing to the Payload Length field of the fragment packet.

If the length and offset of a fragment are such that the Payload Length of the packet reassembled from that fragment would exceed 65,535 octets, then that fragment must be discarded and an ICMP Parameter Problem, Code 0, message should be sent to the source of the fragment, pointing to the Fragment Offset field of the fragment packet.

The following conditions are not expected to occur, but are not considered errors if they do:

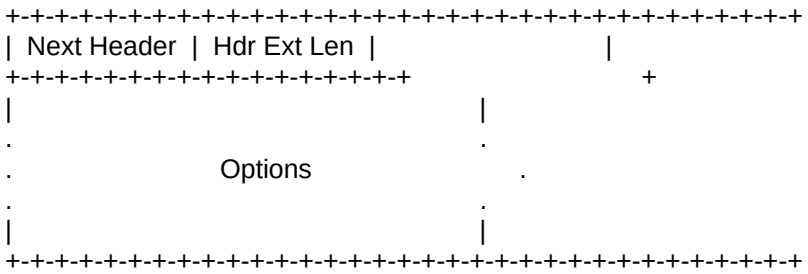
The number and content of the headers preceding the Fragment header of different fragments of the same original packet may differ. Whatever headers are present,

preceding the Fragment header in each fragment packet, are processed when the packets arrive, prior to queuing the fragments for reassembly. Only those headers in the Offset zero fragment packet are retained in the reassembled packet.

The Next Header values in the Fragment headers of different fragments of the same original packet may differ. Only the value from the Offset zero fragment packet is used for reassembly.

4.6 Destination Options Header

The Destination Options header is used to carry optional information that need be examined only by a packet's destination node(s). The Destination Options header is identified by a Next Header value of 60 in the immediately preceding header, and has the following format:



Next Header 8-bit selector. Identifies the type of header immediately following the Destination Options header. Uses the same values as the IPv4 Protocol field [RFC-1700 et seq.].

Hdr Ext Len 8-bit unsigned integer. Length of the Destination Options header in 8-octet units, not including the first 8 octets.

Options Variable-length field, of length such that the complete Destination Options header is an integer multiple of 8 octets long. Contains one or more TLV-encoded options, as described in section 4.2.

The only destination options defined in this document are the Pad1 and PadN options specified in section 4.2.

Note that there are two possible ways to encode optional destination information in an IPv6 packet: either as an option in the Destination Options header, or as a separate extension header. The Fragment header and the Authentication header are examples of the latter approach. Which approach can be used depends on what action is desired of a destination node that does not understand the optional information:

- If the desired action is for the destination node to discard the packet and, only if the packet's Destination Address is not a multicast address, send an ICMP Unrecognized Type message to the packet's Source Address, then the information may be encoded either as a separate header or as an option in the destination Options header whose Option Type has the value 11 in its highest-order two bits. The choice may depend on such factors as which takes fewer octets, or which yields better alignment or more efficient parsing.

- If any other action is desired, the information must be encoded as an option in the Destination Options header whose Option Type has the value 00, 01, or 10 in its highest-order two bits, specifying the desired action (see section 4.2).

4.7 No Next Header

The value 59 in the Next Header field of an IPv6 header or any extension header indicates that there is nothing following that header. If the Payload Length field of the IPv6 header indicates the presence of octets past the end of a header whose Next Header field contains 59, those octets must be ignored, and passed on unchanged if the packet is forwarded.

5 Packet Size Issues

IPv6 requires that every link in the internet have an MTU of 1280 octets or greater. On any link that cannot convey a 1280-octet packet in one piece, link-specific fragmentation and reassembly must be provided at a layer below IPv6.

Links that have a configurable MTU (for example, PPP links [RFC- 1661]) must be configured to have an MTU of at least 1280 octets; it is recommended that they be configured with an MTU of 1500 octets or greater, to accommodate possible encapsulations (i.e., tunneling) without incurring IPv6-layer fragmentation.

From each link to which a node is directly attached, the node must be able to accept packets as large as that link's MTU.

It is strongly recommended that IPv6 nodes implement Path MTU Discovery [RFC-1981], in order to discover and take advantage of path MTUs greater than 1280 octets. However, a minimal IPv6 implementation (e.g., in a boot ROM) may simply restrict itself to sending packets no larger than 1280 octets, and omit implementation of Path MTU Discovery.

In order to send a packet larger than a path's MTU, a node may use the IPv6 Fragment header to fragment the packet at the source and have it reassembled at the destination(s). However, the use of such fragmentation is discouraged in any application that is able to adjust its packets to fit the measured path MTU (i.e., down to 1280 octets).

A node must be able to accept a fragmented packet that, after reassembly, is as large as 1500 octets. A node is permitted to accept fragmented packets that reassemble to more than 1500 octets. An upper-layer protocol or application that depends on IPv6 fragmentation to send packets larger than the MTU of a path should not send packets larger than 1500 octets unless it has assurance that the destination is capable of reassembling packets of that larger size.

In response to an IPv6 packet that is sent to an IPv4 destination (i.e., a packet that undergoes translation from IPv6 to IPv4), the originating IPv6 node may receive an ICMP Packet Too Big message reporting a Next-Hop MTU less than 1280. In that case, the IPv6 node is not required to reduce the size of subsequent packets to less than 1280, but must include a Fragment header in those packets so that the IPv6-to-IPv4 translating router can obtain a suitable Identification value to use in resulting IPv4 fragments. Note that this means the payload may have to be reduced to 1232 octets (1280 minus 40 for the IPv6 header and 8 for the Fragment header), and smaller still if additional extension headers are used.

6 Flow Labels

The 20-bit Flow Label field in the IPv6 header may be used by a source to label sequences of packets for which it requests special handling by the IPv6 routers, such as non-default quality of service or "real-time" service. This aspect of IPv6 is, at the time of writing, still experimental and subject to change as the requirements for flow support in the Internet become clearer. Hosts or routers that do not support the functions of the Flow Label field are required to set the field to zero when originating a packet, pass the field on unchanged when forwarding a packet, and ignore the field when receiving a packet.

Appendix A describes the current intended semantics and usage of the Flow Label field.

7 Traffic Classes

The 8-bit Traffic Class field in the IPv6 header is available for use by originating nodes and/or forwarding routers to identify and distinguish between different classes or priorities of IPv6 packets. At the point in time at which this specification is being written, there are a number of experiments underway in the use of the IPv4 Type of Service and/or Precedence bits to provide various forms of "differentiated service" for IP packets, other than through the use of explicit flow set-up. The Traffic Class field in the IPv6 header is intended to allow similar functionality to be supported in IPv6.

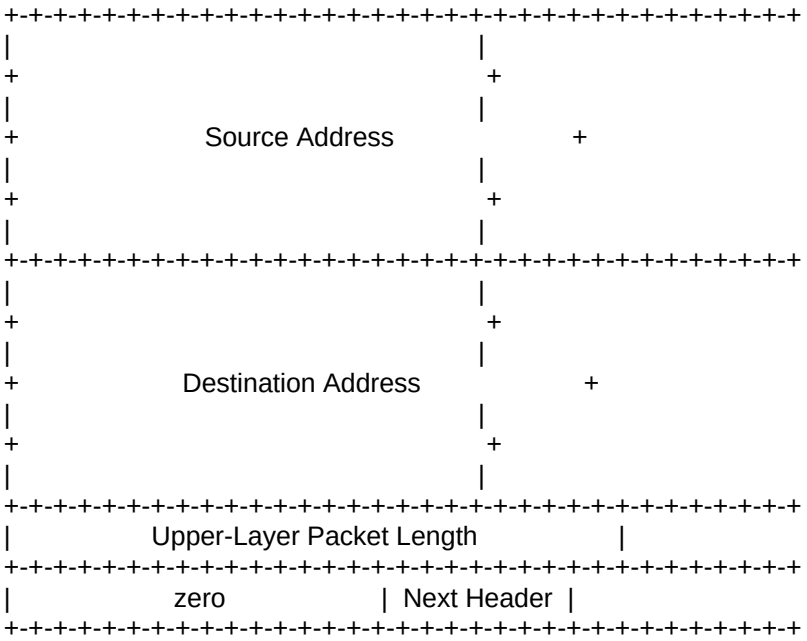
It is hoped that those experiments will eventually lead to agreement on what sorts of traffic classifications are most useful for IP packets. Detailed definitions of the syntax and semantics of all or some of the IPv6 Traffic Class bits, whether experimental or intended for eventual standardization, are to be provided in separate documents.

The following general requirements apply to the Traffic Class field:

- The service interface to the IPv6 service within a node must provide a means for an upper-layer protocol to supply the value of the Traffic Class bits in packets originated by that upper- layer protocol. The default value must be zero for all 8 bits.
- Nodes that support a specific (experimental or eventual standard) use of some or all of the Traffic Class bits are permitted to change the value of those bits in packets that they originate, forward, or receive, as required for that specific use. Nodes should ignore and leave unchanged any bits of the Traffic Class field for which they do not support a specific use.
- An upper-layer protocol must not assume that the values of the Traffic Class bits in a received packet are the same as the value sent by the packet's source.

8 Upper-Layer Protocol Issues
8.1 Upper-Layer Checksums

Any transport or other upper-layer protocol that includes the addresses from the IP header in its checksum computation must be modified for use over IPv6, to include the 128-bit IPv6 addresses instead of 32-bit IPv4 addresses. In particular, the following illustration shows the TCP and UDP "pseudo-header" for IPv6:



- If the IPv6 packet contains a Routing header, the Destination Address used in the pseudo-header is that of the final destination. At the originating node, that address will be in the last element of the Routing header; at the recipient(s), that address will be in the Destination Address field of the IPv6 header.
- The Next Header value in the pseudo-header identifies the upper-layer protocol (e.g., 6 for TCP, or 17 for UDP). It will differ from the Next Header value in the IPv6 header if there are extension headers between the IPv6 header and the upper- layer header.

- The Upper-Layer Packet Length in the pseudo-header is the length of the upper-layer header and data (e.g., TCP header plus TCP data). Some upper-layer protocols carry their own length information (e.g., the Length field in the UDP header); for such protocols, that is the length used in the pseudo- header. Other protocols (such as TCP) do not carry their own length information, in which case the length used in the pseudo-header is the Payload Length from the IPv6 header, minus the length of any extension headers present between the IPv6 header and the upper-layer header.

- Unlike IPv4, when UDP packets are originated by an IPv6 node, the UDP checksum is not optional. That is, whenever originating a UDP packet, an IPv6 node must compute a UDP checksum over the packet and the pseudo-header, and, if that computation yields a result of zero, it must be changed to hex FFFF for placement in the UDP header. IPv6 receivers must discard UDP packets containing a zero checksum, and should log the error.

The IPv6 version of ICMP [ICMPv6] includes the above pseudo-header in its checksum computation; this is a change from the IPv4 version of ICMP, which does not include a pseudo-header in its checksum. The reason for the change is to protect ICMP from misdelivery or corruption of those fields of the IPv6 header on which it depends, which, unlike IPv4, are not covered by an internet-layer checksum. The Next Header field in the pseudo-header for ICMP contains the value 58, which identifies the IPv6 version of ICMP.

8.2 Maximum Packet Lifetime

Unlike IPv4, IPv6 nodes are not required to enforce maximum packet lifetime. That is the reason the IPv4 "Time to Live" field was renamed "Hop Limit" in IPv6. In practice, very few, if any, IPv4 implementations conform to the requirement that they limit packet lifetime, so this is not a change in practice. Any upper-layer protocol that relies on the internet layer (whether IPv4 or IPv6) to limit packet lifetime ought to be upgraded to provide its own mechanisms for detecting and discarding obsolete packets.

8.3 Maximum Upper-Layer Payload Size

When computing the maximum payload size available for upper-layer data, an upper-layer protocol must take into account the larger size of the IPv6 header relative to the IPv4 header. For example, in IPv4, TCP's MSS option is computed as the maximum packet size (a default value or a value learned through Path MTU Discovery) minus 40 octets (20 octets for the minimum-length IPv4 header and 20 octets for the minimum-length TCP header). When using TCP over IPv6, the MSS must be computed as the maximum packet size minus 60 octets, because the minimum-length IPv6 header (i.e., an IPv6 header with no extension headers) is 20 octets longer than a minimum-length IPv4 header.

8.4 Responding to Packets Carrying Routing Headers

When an upper-layer protocol sends one or more packets in response to a received packet that included a Routing header, the response packet(s) must not include a Routing header that was automatically derived by "reversing" the received Routing header UNLESS the integrity and authenticity of the received Source Address and Routing header have been verified (e.g., via the use of an Authentication header in the received packet). In other words, only the following kinds of packets are permitted in response to a received packet bearing a Routing header:

- Response packets that do not carry Routing headers.
- Response packets that carry Routing headers that were NOT derived by reversing the Routing header of the received packet (for example, a Routing header supplied by local configuration).
- Response packets that carry Routing headers that were derived by reversing the Routing header of the received packet IF AND ONLY IF the integrity and authenticity of the Source Address and Routing header from the received packet have been verified by the responder.

Appendix A. Semantics and Usage of the Flow Label Field

A flow is a sequence of packets sent from a particular source to a particular (unicast or multicast) destination for which the source desires special handling by the intervening routers. The nature of that special handling might be conveyed to the routers by a control protocol, such as a resource reservation protocol, or by information within the flow's packets themselves, e.g., in a hop-by-hop option. The details of such control protocols or options are beyond the scope of this document.

There may be multiple active flows from a source to a destination, as well as traffic that is not associated with any flow. A flow is uniquely identified by the combination of a source address and a non-zero flow label. Packets that do not belong to a flow carry a flow label of zero.

A flow label is assigned to a flow by the flow's source node. New flow labels must be chosen (pseudo-)randomly and uniformly from the range 1 to FFFFF hex. The

purpose of the random allocation is to make any set of bits within the Flow Label field suitable for use as a hash key by routers, for looking up the state associated with the flow.

All packets belonging to the same flow must be sent with the same source address, destination address, and flow label. If any of those packets includes a Hop-by-Hop Options header, then they all must be originated with the same Hop-by-Hop Options header contents (excluding the Next Header field of the Hop-by-Hop Options header). If any of those packets includes a Routing header, then they all must be originated with the same contents in all extension headers up to and including the Routing header (excluding the Next Header field in the Routing header). The routers or destinations are permitted, but not required, to verify that these conditions are satisfied. If a violation is detected, it should be reported to the source by an ICMP Parameter Problem message, Code 0, pointing to the high-order octet of the Flow Label field (i.e., offset 1 within the IPv6 packet).

The maximum lifetime of any flow-handling state established along a flow's path must be specified as part of the description of the state-establishment mechanism, e.g., the resource reservation protocol or the flow-setup hop-by-hop option. A source must not re- use a flow label for a new flow within the maximum lifetime of any flow-handling state that might have been established for the prior use of that flow label.

When a node stops and restarts (e.g., as a result of a "crash"), it must be careful not to use a flow label that it might have used for an earlier flow whose lifetime may not have expired yet. This may be accomplished by recording flow label usage on stable storage so that it can be remembered across crashes, or by refraining from using any flow labels until the maximum lifetime of any possible previously established flows has expired. If the minimum time for rebooting the node is known, that time can be deducted from the necessary waiting period before starting to allocate flow labels.

There is no requirement that all, or even most, packets belong to flows, i.e., carry non-zero flow labels. This observation is placed here to remind protocol designers and implementors not to assume otherwise. For example, it would be unwise to design a router whose performance would be adequate only if most packets belonged to flows, or to design a header compression scheme that only worked on packets that belonged to flows.

Appendix B. Formatting Guidelines for Options

This appendix gives some advice on how to lay out the fields when designing new options to be used in the Hop-by-Hop Options header or the Destination Options header, as described in section 4.2. These guidelines are based on the following assumptions:

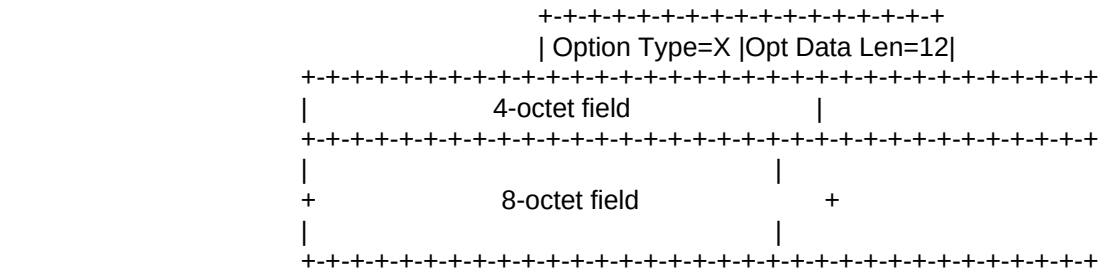
- One desirable feature is that any multi-octet fields within the Option Data area of an option be aligned on their natural boundaries, i.e., fields of width n octets should be placed at an integer multiple of n octets from the start of the Hop-by-
- Hop or Destination Options header, for n = 1, 2, 4, or 8.
- Another desirable feature is that the Hop-by-Hop or Destination Options header take up as little space as possible, subject to the requirement that the header be an integer multiple of 8 octets long.
- It may be assumed that, when either of the option-bearing headers are present, they carry a very small number of options, usually only one.

These assumptions suggest the following approach to laying out the fields of an option: order the fields from smallest to largest, with no interior padding, then derive

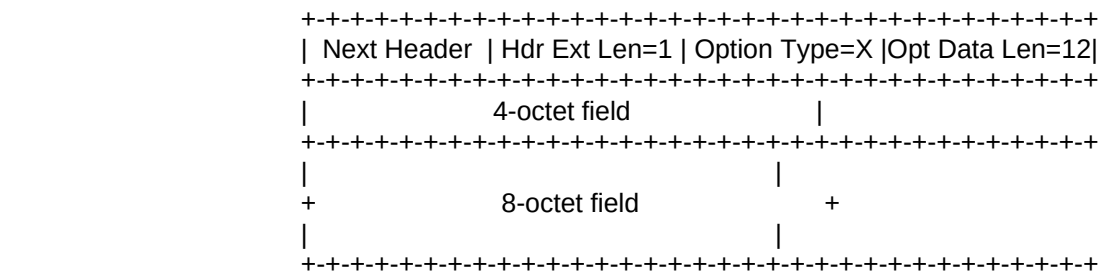
the alignment requirement for the entire option based on the alignment requirement of the largest field (up to a maximum alignment of 8 octets). This approach is illustrated in the following examples:

Example 1

If an option X required two data fields, one of length 8 octets and one of length 4 octets, it would be laid out as follows:

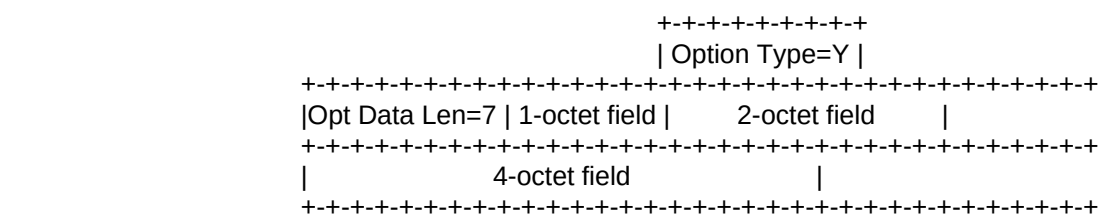


Its alignment requirement is 8n+2, to ensure that the 8-octet field starts at a multiple-of-8 offset from the start of the enclosing header. A complete Hop-by-Hop or Destination Options header containing this one option would look as follows:

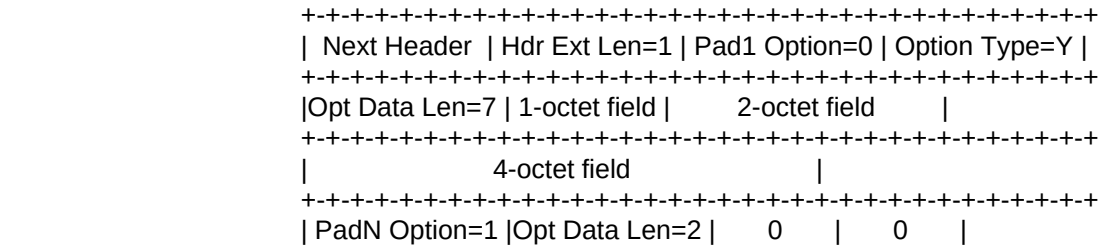


Example 2

If an option Y required three data fields, one of length 4 octets, one of length 2 octets, and one of length 1 octet, it would be laid out as follows:

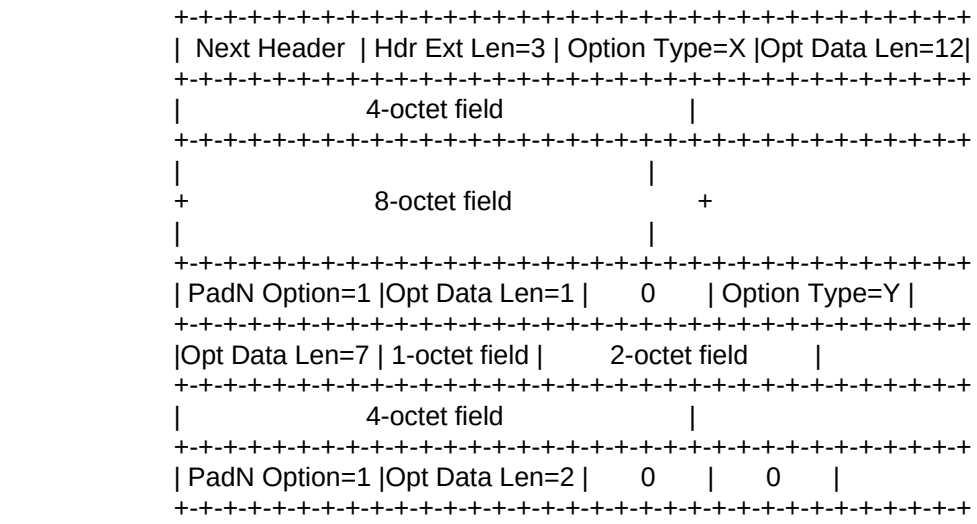


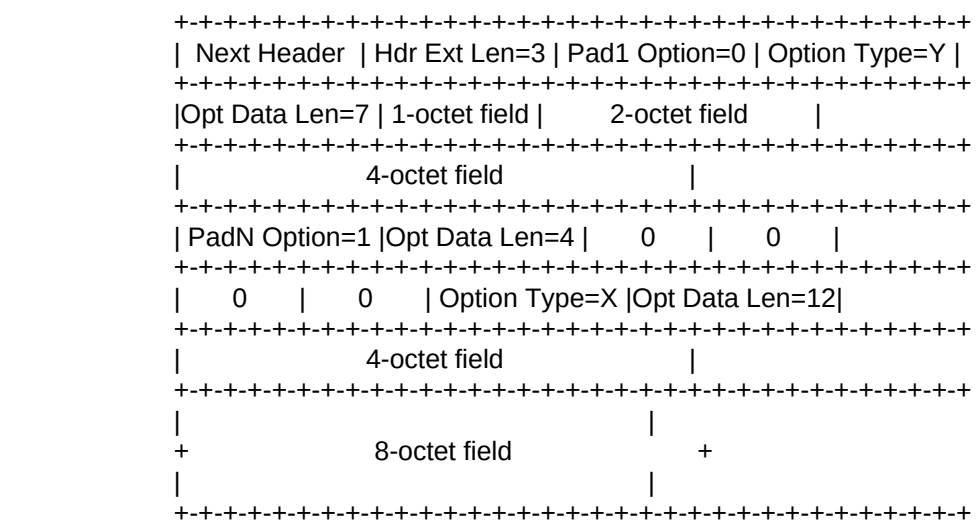
Its alignment requirement is 4n+3, to ensure that the 4-octet field starts at a multiple-of-4 offset from the start of the enclosing header. A complete Hop-by-Hop or Destination Options header containing this one option would look as follows:



Example 3

A Hop-by-Hop or Destination Options header containing both options X and Y from Examples 1 and 2 would have one of the two following formats, depending on which option appeared first:





Security Considerations

The security features of IPv6 are described in the Security Architecture for the Internet Protocol [RFC-2401].

Acknowledgments

The authors gratefully acknowledge the many helpful suggestions of the members of the IPng working group, the End-to-End Protocols research group, and the Internet Community At Large.

Authors' Addresses

Stephen E. Deering
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA

Phone: +1 408 527 8213

Fax: +1 408 527 8254
EMail: deering@cisco.com

Robert M. Hinden
Nokia
232 Java Drive
Sunnyvale, CA 94089
USA

Phone: +1 408 990-2004

Fax: +1 408 743-5677
EMail: hinden@iprg.nokia.com

References

[RFC-2401] Kent, S. and R. Atkinson, "Security Architecture for the Internet Protocol", RFC 2401, November 1998.

[RFC-2402] Kent, S. and R. Atkinson, "IP Authentication Header", RFC 2402, November 1998.

[RFC-2406] Kent, S. and R. Atkinson, "IP Encapsulating Security Protocol (ESP)", RFC 2406, November 1998.

[ICMPv6] Conta, A. and S. Deering, "ICMP for the Internet Protocol Version 6 (IPv6)", RFC 2463, December 1998.

[ADDRARCH] Hinden, R. and S. Deering, "IP Version 6 Addressing Architecture", RFC 2373, July 1998.

[RFC-1981] McCann, J., Mogul, J. and S. Deering, "Path MTU Discovery for IP version 6", RFC 1981, August 1996.

[RFC-791] Postel, J., "Internet Protocol", STD 5, RFC 791, September 1981.

[RFC-1700] Reynolds, J. and J. Postel, "Assigned Numbers", STD 2, RFC 1700, October 1994. See also: http://www.iana.org/numbers.html

[RFC-1661] Simpson, W., "The Point-to-Point Protocol (PPP)", STD 51, RFC 1661, July 1994.
CHANGES SINCE RFC-1883

This memo has the following changes from RFC-1883. Numbers identify the Internet-Draft version in which the change was made.

02) Removed all references to jumbograms and the Jumbo Payload option (moved to a separate document).

02) Moved most of Flow Label description from section 6 to (new) Appendix A.

02) In Flow Label description, now in Appendix A, corrected maximum Flow Label value from FFFFFF to FFFFF (i.e., one less "F") due to reduction of size of Flow Label field from 24 bits to 20 bits.

02) Renumbered (relettered?) the previous Appendix A to be Appendix B.

02) Changed the wording of the Security Considerations section to avoid dependency loop between this spec and the IPsec specs.

02) Updated R. Hinden's email address and company affiliation.

01) In section 3, changed field name "Class" to "Traffic Class" and increased its size from 4 to 8 bits. Decreased size of Flow Label field from 24 to 20 bits to compensate for increase in Traffic Class field.

01) In section 4.1, restored the order of the Authentication Header and the ESP header, which were mistakenly swapped in the 00 version of this memo.

01) In section 4.4, deleted the Strict/Loose Bit Map field and the strict routing functionality from the Type 0 Routing header, and removed the restriction on number of addresses that may be carried in the Type 0 Routing header (was limited to 23 addresses, because of the size of the strict/loose bit map).

01) In section 5, changed the minimum IPv6 MTU from 576 to 1280 octets, and added a recommendation that links with configurable MTU (e.g., PPP links) be configured to have an MTU of at least 1500 octets.

01) In section 5, deleted the requirement that a node must not send fragmented packets that reassemble to more than 1500 octets without knowledge of the destination reassembly buffer size, and replaced it with a recommendation that upper-layer protocols or applications should not do that.

01) Replaced reference to the IPv4 Path MTU Discovery spec (RFC- 1191) with reference to the IPv6 Path MTU Discovery spec (RFC- 1981), and deleted the Notes at the end of section 5 regarding Path MTU Discovery, since those details are now covered by RFC- 1981.

01) In section 6, deleted specification of "opportunistic" flow set-up, and removed all references to the 6-second maximum lifetime for opportunistically established flow state.

01) In section 7, deleted the provisional description of the internal structure and semantics of the Traffic Class field, and specified that such descriptions be provided in separate documents.

00) In section 4, corrected the Code value to indicate "unrecognized Next Header type encountered" in an ICMP Parameter Problem message (changed from 2 to 1).

00) In the description of the Payload Length field in section 3, and of the Jumbo Payload Length field in section 4.3, made it clearer that extension headers are included in the payload length count.

00) In section 4.1, swapped the order of the Authentication header and the ESP header. (NOTE: this was a mistake, and the change was undone in version 01.)

00) In section 4.2, made it clearer that options are identified by the full 8-bit Option Type, not by the low-order 5 bits of an Option Type. Also specified that the same Option Type numbering space is used for both Hop-by-Hop Options and Destination Options headers.

00) In section 4.4, added a sentence requiring that nodes processing a Routing header must send an ICMP Packet Too Big message in response to a packet that is too big to fit in the next hop link (rather than, say, performing fragmentation).

00) Changed the name of the IPv6 Priority field to "Class", and replaced the previous description of Priority in section 7 with a description of the Class field. Also, excluded this field from the set of fields that must remain the same for all packets in the same flow, as specified in section 6.

00) In the pseudo-header in section 8.1, changed the name of the "Payload Length" field to "Upper-Layer Packet Length". Also clarified that, in the case of protocols that carry their own length info (like non-jumbogram UDP), it is the upper-layer- derived length, not the IP-layer-derived length, that is used in the pseudo-header.

00) Added section 8.4, specifying that upper-layer protocols, when responding to a received packet that carried a Routing header, must not include the reverse of the Routing header in the response packet(s) unless the received Routing header was authenticated.

00) Fixed some typos and grammatical errors.

00) Authors' contact info updated.

Full Copyright Statement