Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

GUÍA PARA LA TRANSFORMACIÓN DE LOS SERVICIOS DE ASIGNACIÓN DINÁMICA DE IP'S (DHCP) Y RESOLUCIÓN DE NOMBRES (DNS), UTILIZANDO LA NUEVA GENERACIÓN DE IP'S (IPV4 – IPV6)

Mónica Dávila López
Asesorada por: Ing. Cresencio Chan Canek

GUATEMALA, OCTUBRE DE 2004

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA

FACULTAD DE INGENIERÍA

GUÍA PARA LA TRANSFORMACIÓN DE LOS SERVICIOS DE ASIGNACIÓN DINÁMICA DE IP'S (DHCP) Y RESOLUCIÓN DE NOMBRES (DNS), UTILIZANDO LA NUEVA GENERACIÓN DE IP'S (IPV4 – IPV6)

TRABAJO DE GRADUACIÓN

PRESENTADO A JUNTA DIRECTIVA DE LA

FACULTAD DE INGENIERÍA

POR

MÓNICA DÁVILA LÓPEZ

Asesorado por: Ing. Cresencio Chan Canek

AL CONFERÍRSELE EL TÍTULO DE

INGENIERA EN CIENCIAS Y SISTEMAS

GUATEMALA, OCTUBRE DE 2004

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA

FACULTAD DE INGENIERÍA

NÓMINA DE JUNTA DIRECTIVA

DECANO	Ing. Sydney Alexander Samuels Milson
VOCAL I	Ing. Murphy Olympo Paiz Recinos
VOCAL II	Ing. Amahán Sánchez Álvarez
VOCAL III	Ing. Julio David Galicia Celada
VOCAL IV	Br. Kenneth Issur Estrada Ruiz
VOCAL V	Ing. Pedro Antonio Aguilar Polanco

TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO

DECANO	Ing. Sydney Alexander Samuels Milson
EXAMINADOR	Ing. Ricardo Morales
EXAMINADORA	Inga. Elizabeth Domínguez
EXAMINADORA	Inga. Virginia Victoria Tala Ayerdi
SECRETARIO	Ing. Pedro Antonio Aguilar Polanco

HONORABLE TRIBUNAL EXAMINADOR

Cumpliendo con los preceptos que establece la ley de la Universidad de San Carlos de Guatemala, presento a su consideración mi trabajo de graduación titulado:

GUÍA PARA LA TRANSFORMACIÓN DE LOS SERVICIOS DE ASIGNACIÓN DINÁMICA DE IP'S (DHCP) Y RESOLUCIÓN DE NOMBRES (DNS), UTILIZANDO LA NUEVA GENERACIÓN DE IP'S (IPV4 – IPV6)

Tema que me fuera asignado por la Dirección de la Escuela de Ingeniería en Ciencias y Sistemas, con fecha enero de 2003.

_____

Mónica Dávila López

Guatemala, octubre de 2004

Ingeniero
Carlos Alfredo Azurdia Morales
Coordinador de Privados y Revisión de Tesis
Escuela de Ciencias y Sistemas

Estimado Ingeniero:

Por medio de la presente, me permito informarle que he asesorado el trabajo de graduación titulado: GUÍA PARA LA TRANSFORMACIÓN DE LOS SERVICIOS DE ASIGNACIÓN DINÁMICA DE IP'S (DHCP) Y RESOLUCIÓN DE NOMBRES (DNS), UTILIZANDO LA NUEVA GENERACIÓN DE IP'S (IPV4 – IPV6), elaborado por la estudiante Mónica Dávila López, a mi juicio el mismo cumple con los objetivos propuestos para su desarrollo.

Agradeciéndole de antemano la atención que le preste a la presente, me suscribo de usted,

Atentamente,

Cresencio Chan Canek
Ingeniero en Ciencias y Sistemas
Asesor

El Director de la Carrera de Ingeniería en Ciencias y Sistemas de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer el dictamen del asesor, con el visto bueno del revisor de tesis y del licenciado en Letras, al trabajo de graduación titulado **GUÍA PARA LA TRANSFORMACIÓN DE LOS SERVICIOS DE ASIGNACIÓN DINÁMICA DE IP'S (DHCP) Y RESOLUCIÓN DE NOMBRES (DNS), UTILIZANDO LA NUEVA GENERACIÓN DE IP'S (IPV4 – IPV6),** presentado por el estudiante **Mónica Dávila López**, aprueba el presente trabajo y solicita la autorización del mismo.

ID Y ENSEÑAD A TODOS

Ing. Luis Alberto Vettorazzi España
DIRECTOR
Ingeniería en Ciencias y Sistemas

Guatemala, 21 de octubre de 2004

El Decano de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer la aprobación por parte del Director de la Escuela de Ingeniería en Ciencias y Sistemas, al trabajo de graduación titulado: **GUÍA PARA LA TRANSFORMACIÓN DE LOS SERVICIOS DE ASIGNACIÓN DINÁMICA DE IP'S (DHCP) Y RESOLUCIÓN DE NOMBRES (DNS), UTILIZANDO LA NUEVA GENERACIÓN DE IP'S (IPV4 – IPV6),** presentado por el estudiante universitario **Mónica Dávila López** procede a la autorización para la impresión del mismo.

IMPRÍMASE:

Ing. Sydney Alexander Samuels
DECANO

Guatemala, 21 de octubre de 2004

DEDICATORIA A

Dios	Por haberme dado la oportunidad de llegar a culminar la meta propuesta desde niña.
Mis padres	José Angel Dávila y Blanca Lydia López de Dávila, por ser ejemplo a seguir, por darme todo su apoyo incondicional, por ser las piezas más fuertes en el tablero de mi vida. Papá y mamá, este logro es para ustedes. Ya podemos decir: ¡ Lo hemos logrado !
Mi hermana	Sonia Dávila López, mi dulce compañía, inseparable en juegos y estudios.
Mi hijo	Diego Fabián Morales Dávila, el tesoro de mi vida a quien espero darle todo mi apoyo y ser un ejemplo a seguir.
Mi esposo	Otto Morales, por ser mi apoyo incondicional.
Mis compañeros	En especial a Blanca Castillo, Xiomara Vivar, Karla Santos, Siomara Simón, Luis Santizo, Norma Díaz, Carlos Santucci, Daniel Girón, y a todos aquellos con quienes compartí mis alegrías y tristezas y que juntos como un equipo hemos salido triunfadores.
Catedráticos	A cada uno de ellos, mi admiración y respeto.

AGRADECIMIENTOS A

Dios Por ser la luz que vigila y guía mis pasos.

Mi papá Por todo su apoyo y sus consejos. Gracias papá.

Mi mamá Por ser compañera idónea en mi vida, por ser quien me da sus consejos, por estar siempre junto a mí en mis triunfos y fracasos y por ser la persona que siempre me motiva a seguir alcanzando metas. Mamá gracias por ser como eres.

Mi asesor Ing. Cresencio Chan Canek, por haberme orientado y llegar a culminar el último escalón de mi meta.

Amigos y compañeros de carrera **A** todos, mi respeto.

USAC Con capucha y sin capucha, ¡ Siempre en la lucha !

Un agradecimiento especial a la Inga. Sonia Castañeda, por motivarme a seguir en la lucha.

ÍNDICE GENERAL

ÍNDICE DE ILUSTRACIONES

FIGURAS

TABLAS

GLOSARIO

ARP Protocolo de resolución de direcciones (*Address Resolution Protocol*, por sus siglas en inglés).

AUI Interfaz de unidad de conexión (*Attachment Unit Interface*, por sus siglas en inglés). Conector usado con la *Ethernet* de alambre grueso. Hay una conexión AUI entre una computadora y un transceptor *Ethernet*.

BOOTP Protocolo de arranque (*BOOTstrap Protocol*, por sus siglas en inglés). Protocolo que usa una computadora cuando comienza a obtener información necesaria para configurar el *software* del protocolo. BOOTP emplea IP o UDP para difundir una solicitud y recibir una respuesta antes de que el IP se haya configurado por completo.

Cabecera base Cabecera obligatoria encontrada al inicio de un datagrama IPV6.

Cabecera de extensión Cabecera opcional usada en el protocolo IPV6.

Configuración de protocolo Paso que debe ejecutar la computadora para asignar valores a los parámetros antes de emplear el *software* del protocolo. En general, la configuración del protocolo

	requiere de un sistema que obtenga una dirección de protocolo.
Datagrama IP	Se usa de manera indistinta con paquete de datos o mensaje de red para identificar una unidad de información que se intercambia. Forma de un paquete enviado por una interred TCP/IP. Cada datagrama tiene una cabecera que identifica tanto al transmisor como al receptor, seguida de datos.
DHCP	Protocolo de configuración dinámica de *host* (*Dynamic Host Configuration Protocol,* por sus siglas en inglés). Protocolo que usan las computadoras para obtener información de configuración. El DHCP permite asignar una dirección IP a una computadora sin requerir que un administrador configure la información sobre la computadora en la base de datos de un servidor.
DNS	Sistema de nombres de dominio (*Domain Name System*, por sus siglas en inglés). Sistema automatizado que sirve para traducir nombres de computadoras a direcciones IP equivalentes. Un servidor DNS responde a una consulta buscando el nombre y devolviendo la dirección. Esto ahorra a cada sistema de la red el mantener una lista de todos los sistemas con los que quiere comunicarse. Lo usan las compuertas de correo.
Dominio	Parte de la jerarquía de nombres de computadora usada en Internet. Por ejemplo, las organizaciones comerciales registran sus nombres bajo el dominio .com.

Enrutador Bloque de construcción básico de una interred. Un errutador es una computadora que se conecta a dos o más redes y reenvía paquetes de acuerdo con la información encontrada en su tabla de enrutamiento. Los enrutadores de la *Ethernet* ejecutan el protocolo IP.

Ethernet Difundida tecnología de red de área local que usa una topología de canal compartido y acceso CSMA/CD. La *Ethernet* básica opera a 10 Mbps; la *Ethernet* rápida opera a 100 Mbps.

Fragmentación Técnica usada por el IP para dividir un datagrama grande en datagramas más pequeños llamados fragmentos. El destino final reconstruye los fragmentos.

FTP (*File Transfer Protocol*, por sus siglas en inglés). Es un protocolo usado para intercambiar ficheros entre dos máquinas, y además permite hacer *login* en la otra máquina para realizar funciones limitadas tales como borrar ficheros, crear directorios, listar directorios, etc.

Host Computadora de usuario final conectada a una red. En una interred, todas las computadoras se clasifican como *hosts* o enrutadores.

IANA	Autoridad de número asignados en Internet (*Internet Assigned Number Authority*, por sus siglas en inglés). Organización responsable de asignar los números usados por los protocolos TCP/IP. Por ejemplo, la IANA asigna direcciones IP.
ICMP	Protocolo de control de mensajes de interred (*Internet Control Message Protocol*, por sus siglas en inglés). Protocolo usado por el IP para informar de errores y excepciones, es una extensión del protocolo IP. El ICMP también incluye mensajes informativos usados por algunos programas como *ping*.
IETF	Grupo de trabajo de ingeniería de Internet (*Internet Engineering Task Force*, por sus siglas en inglés).
Internet	Red de computadoras basada en TCP/IP y protocolos relacionados. Además, es una red pública que interconecta negocios, universidades, instalaciones gubernamentales y centros de investigación.
IP	(*Internet Protocol*, por sus siglas en inglés). Define el protocolo de encaminamiento de los paquetes a través de la red. Para ello cuenta con un sistema de direccionamiento, conocido como direcciones IP.

IPng Protocolo de Internet: la siguiente generación (*Internet Protocol-the Next Generation*, por sus siglas en inglés). Nombre genérico usado durante los debates iniciales de un nuevo protocolo que sucedería al IPV4. Los investigadores propusieron varios protocolos posibles para el IPng.

IPSec Seguridad del protocolo de Internet, es el estándar para la seguridad de los paquetes en IPV4.

IPV4 Protocolo de Internet versión 4 (*Internet Protocol Version 4*, por sus siglas en inglés). Versión del IP actualmente usada en Internet. El IPV4 usa direcciones de 32 *bits*.

IPV6 Protocolo de Internet versión 6 (*Internet Protocol Version 6*, por sus siglas en inglés). Protocolo específico que ha sido propuesto por el IETF como sucesor del IPV4. El IPV6 usa direcciones de 128 *bits*.

ISO Organización Internacional de Normalización (*International Organization for Standardization*, por sus siglas en inglés). Organización conocida por haber propuesto el modelo de referencia de siete capas de la historia temprana de la conectividad de datos.

ISP Proveedor de servicio Internet (*Internet Service Provider*, por sus siglas en inglés). Organización comercial que provee acceso a Internet a sus suscriptores.

LAN Red de área local (*Local Area Network*, por sus siglas en inglés). Red que usa tecnología diseñada para abarcar un área geográfica pequeña. Por ejemplo, la *Ethernet* es una tecnología de LAN adecuada para uso en un sólo edificio. Las LAN tienen retardos de propagación menores que las WAN.

LocalTalk Tecnología de LAN desarrollada por *Apple Computer Corporation* que usa una topología de canal. *LocalTalk* usa protocolos *AppleTalk*.

Máscara de dirección Cifra de 32 *bits* que específica los *bits* de una dirección IP que corresponde a una red y a una subred. Las direcciones de *bits* no cubiertas por la máscara corresponden a la parte del *host*. También llamado máscara de subred.

Máscara de subred Sinónimo de máscara de dirección.

MTU Unidad máxima de transmisión (*Maximum Transmission Unit*, por sus siglas en inglés). Cantidad máxima de datos que pueden transmitirse por una red en un sólo paquete. Cada tecnología de red define una MTU (por ejemplo, la MTU de la *Ethernet* es de 1500 octetos).

Multibase Cualquier computadora *host* que se conecta a más de una red. En la mayor parte de los sistemas de protocolos, una computadora multibase tiene más de una dirección.

NAT Traductor de direcciones de red (*Network Address Translator*, por sus siglas en inglés).

NFS (*Network File System*, por sus siglas en inglés). Protocolo usado para compartir sistemas de ficheros en un ambiente heterogéneo de equipos, sistemas operativos y redes.

NIC Centro de información de red (*Net Information Center*, por sus siglas en inglés); además, tarjeta de interfaz de red (*Network Interface Card*, por sus siglas en inglés). Es el responsable de administrar Internet, las direcciones TCP/IP y los nombres de red. Dispositivo de *hardware* que se enchufa en una computadora y la conecta a una red. Llamado con frecuencia adaptador de red.

Nodo Término usado informalmente para hacer referencia a un enrutador o a una computadora conectada a una red. El término se deriva de la teoría de las gráficas.

Pila Término informal de la implantación de un grupo de programas de protocolo, se refiere a que los diagramas de capas de protocolos muestran éstos como una pila vertical.

PING Inquisidor de paquetes de interred (*Packet Internet Groper*, por sus siglas en inglés). Programa usado para probar la conectividad de una red. *Ping* envía una solicitud de contestación ICMP al destino y reporta si recibe el regreso de contestación ICMP esperado.

Protocolo	Diseño que especifica los detalles sobre la manera en que se relacionan las computadoras, incluyendo el formato de los mensajes que intercambian y el manejo de los errores.
QoS	Calidad de servicio (*Quality of Service*, por sus siglas en inglés), es denominada así ya que permite la facilidad de entrega de datos en tiempo real.
RARP	(*Reverse Address Resolution Protocol*, por sus siglas en inglés). Es una variación del ARP, que permite a estaciones sin unidad de almacenamiento fijo obtener su propia dirección IP.
Relleno de *bytes*	Técnica de protocolo en la que los datos se cambian insertando *bytes* adicionales para distinguir entre valores y campos de control de paquete.
Retransmisión	Reenvío de un paquete que ya se había expedido. Los protocolos de transportación usan la retransmisión para lograr confiabilidad.
RFC (petición de comentarios)	Es la documentación que mantiene el NIC en cuanto a protocolos Internet, direccionamiento, ruteo, configuración y otros temas de Internet relacionados.
RIP	Protocolo de ruteo de información (*Router Information Protocol*, por sus siglas en inglés). Se utiliza para intercambiar información entre ruteadores.

Segmento Tramo de cable que forma una red de canal. Pueden conectarse varios segmentos mediante puentes y enrutadores. Un concentrador simula un segmento.

Servidor Cuando se comunican dos programas por una red, el cliente es el que inicia la comunicación, y el programa que espera ser contactado es el servidor. Cada programa puede actuar como servidor para un servicio y como cliente para otro.

SMTP Protocolo sencillo de transferencia de correo (*Simple Mail Transfer Protocol*, por sus siglas en inglés). Protocolo usado para la transferencia de correo electrónico de una computadora a otra por medio de Internet. El SMTP es parte del grupo de programas de protocolo TCP/IP.

SNAP Punto de conexión de subred (*SubNetwork Attachment Point*, por sus siglas en inglés). Parte de la cabecera LLC/SNAP del IEEE usada para identificar el tipo de un paquete. La cabecera completa es de 8 octetos, de los que la parte del SNAP ocupa los último cinco.

SNMP Protocolo sencillo de administración de redes (*Simple Network Management Protocol*, por sus siglas en inglés). Protocolo que especifica la manera en que una estación de administración de red se comunica con el *software* agente de los dispositivos remotos, como enrutadores. El SNMP define el formato de los mensajes y su significado.

TCP	(*Transmission Control Protocol*, por sus siglas en inglés). Es un protocolo de transporte que ofrece una conexión bidireccional y fiable entre dos aplicaciones, orientado a conexión, lo que significa que los participantes deben establecer la conexión antes de enviar datos.
TCP/IP	Grupo de programas de protocolo usado en Internet. Aunque contiene muchos protocolos, el TCP y el IP son dos de los más importantes.
TELNET	(*Telecomunication – Network*, por sus siglas en inglés). Permite hacer uso de servicios de terminal virtual en máquinas remotas y trabajar de forma interactiva.
Topología	Término que describe la forma general de una red. Las topologías comunes incluyen canal, anillo, estrella y punto a punto.
UDP	(*User Datagram Protocol*, por sus siglas en inglés). Es un protocolo de transporte que ofrece un servicio no fiable y no orientado a conexión entre dos aplicaciones.
URL	Localizador uniforme de recursos (*Uniform Resource Locator*, por sus siglas en inglés). Forma sintáctica usada para identificar una página de información en la WWW.

WAN Red de área amplia (*Wide Area Network,* por sus siglas en inglés). Red que usa tecnología diseñada para abarcar un área geográfica grande. Por ejemplo, una red satelital es una WAN, puesto que un satélite puede redifundir a todo un continente. Las WAN tienen un mayor retardo de propagación que las LAN.

Web Sinónimo de *World Wide Web* (WWW).

WWW Red mundial (*Word Wide Wed*, por sus siglas en inglés). Sistema de hipermedios usado en Internet en el que una página de información puede contener texto, imágenes, fragmentos de audio o vídeo y referencias a otras páginas.

OBJETIVOS

General

Definir una guía para la transformación de servicios en Internet utilizando el nuevo protocolo de los IP´s denominado IPng (IPV6), para establecer un antecedente de esta nueva tecnología de los protocolos de comunicación en redes.

Específicos

- Introducir el concepto de los protocolos de la nueva generación como lo es el protocolo IPV6.

- Realizar una investigación del protocolo anterior al IPV6, el protocolo IPV4, para establecer las causas o avances del nuevo protocolo.

- Describir las estructuras que componen los protocolos IPV4 – IPV6.

- Desarrollar una guía para la transformación de los servicios DHCP, utilizando los protocolos IPV4 – IPV6.

- Desarrollar una guía para la transformación de los servicios DNS, utilizando los protocolos IPV4 – IPV6.

RESUMEN

El término Internet se deriva del término *internetworking* (trabajo en interred) que, como lo implica, quiere decir el concepto de redes conectándose con otras redes. Cuando las personas usan el término Internet, describen probablemente el Internet global que evolucionó de un proyecto de investigación Federal en la década de 1960.

El protocolo de Internet (IP) tiene sus raíces más tempranas en las redes militares de 1970, pero es en la pasada década cuando se hace imparable en el mundo de las redes. Hoy en día, IP se ha establecido por sí sola como el vehículo primario para nuestro sistema global de comercio electrónico permitiendo un amplio rango de aplicaciones cliente servidor.

El protocolo IP (*Internet Protocol*) fue diseñado para interconexión de redes. IP se ocupa de la transmisión de bloques de datos, llamados datagramas de origen a destino, donde orígenes y destinos son *hosts* identificados por direcciones de una longitud fija. IP también se encarga de la fragmentación y reensamblado de datagramas, si éste fuera necesario.

El protocolo IP implementa dos funciones básicas: Direccionamiento y fragmentación.

El módulo Internet usa las direcciones contenidas en la cabecera de los datagramas para hacer llegar a estos a sus destinos. Asimismo, existen otros campos en la cabecera que permiten gestionar la fragmentación y posterior reensamblado de datagramas, para poder transmitir a través de redes que trabajen con tamaños de paquetes pequeños.

El módulo Internet reside en cada *host* integrado en la Internet, y en cada *gateway* interconectando redes. Estos módulos siguen reglas comunes para interpretar las direcciones y para realizar la fragmentación y el reensamblado de datagramas. Adicionalmente, estos módulos (especialmente en los *gateways)* están provistos de mecanismos para tomar decisiones sobre el enrutamiento de los datagramas.

Sin embargo, en el diseño inicial no se previó lo siguiente:

- El reciente crecimiento exponencial de Internet y el inminente agotamiento del espacio de direcciones IPV4
- El crecimiento de Internet y la capacidad de los enrutadores troncales de Internet para mantener grandes tablas de enrutamiento
- La necesidad de una configuración más sencilla
- El requisito de seguridad en el nivel de IP
- La necesidad de facilitar la entrega de datos en tiempo real, también denominada calidad de servicio (QoS, *Quality of Service*)

Para resolver estas preocupaciones, el grupo de trabajo de ingeniería de Internet (IETF) ha desarrollado un conjunto de protocolos y estándares conocidos como IP versión 6 (IPV6). El diseño de IPV6 se ha diseñado intencionalmente para que afecte lo menos posible a los protocolos de nivel superior e inferior al evitar que se agreguen aleatoriamente nuevas características.

IPV6, la siguiente generación de protocolo de Internet, fue aprobado por el *Internet Engineering Steering Group* el 17 de noviembre de 1994, como una propuesta de estándar. Desde ese momento, un gran número de organizaciones de usuarios finales, grupos de estándares y vendedores de redes han estado trabajando juntos en la especificación y pruebas de implementaciones de IPV6.

El protocolo de IPV6 fue diseñado con importantes características que permitirán tener más y mejores redes superando las limitaciones del protocolo IPV4 usado actualmente. Entre las características más importantes destacan las de espacio de direcciones prácticamente infinito, autoconfiguración de computadoras y ruteadores, computación móvil, mayor soporte para seguridad, herramientas de calidad de servicio, manejo de tráfico multimedia en tiempo real, aplicaciones multicast y mecanismos para la transición gradual de IPV4 a IPV6; características que harán posible, por ejemplo, la coexistencia de la telefonía, las comunicaciones móviles e inalámbricas y los medios audiovisuales en redes más grandes, eficientes y seguras.

A pesar de que las espectativas son muy promisorias, pasar de un protocolo a otro es bastante complicado, no sólo el costo que llevaría transformar toda una red a otra si se hace radicalmente el cambio, sino también que el tiempo que conlleva cada prueba que se realiza para que el protocolo IPV4 soporte al nuevo protocolo IPV6.

En éste trabajo se hará un análisis del proceso de transición de los protocolos IPV4 e IPV6 utilizando los servicios DNS y DHCP. DHCP se basa en el conocido modelo cliente-servidor. Utiliza un protocolo de comunicaciones muy sencillo (basado en UDP sobre IP).

Los clientes de una red que utilicen este protocolo utilizan direcciones IP que les "alquila" un servidor (no tiene que ser local). Cada vez que un cliente se inicia, pide una dirección IP o una renovación de la que tiene alquilada actualmente. El cliente recibe, junto con la dirección, algunos parámetros adicionales: pasarela (*gateway*) por defecto, servidor WINS (*Windows Internet Name Service*), servidor DNS, etc.

Lo que DHCP consigue es que la asignación y liberación de las direcciones IP en una red sea dinámica y automática; se evita las duplicidades y se optimiza el consumo de direcciones. La intervención del administrador de redes, aun en grandes configuraciones es mínima.

El protocolo de configuración de *host* dinámica (DHCP) proporciona un espacio de trabajo para pasar información de configuración a los *hosts* sobre una red TCP/IP. DHCP se basa en el protocolo BOOTP, añadiendo la capacidad de localización automática de direcciones de red reutilizables y opciones de configuración adicionales.

El mecanismo que implementa una jerarquía de nombres de máquinas en las redes se llama sistema de dominio de nombres (*Domain Name System*, DNS; a partir de ahora utilizaremos las siglas anglosajonas para referirnos a este sistema, por ser reconocidas internacionalmente, y más familiares).

El DNS especifica la sintaxis de los nombres, y las reglas para delegar autoridad sobre los nombres; además de especificar la implementación de un sistema distribuido que relaciona eficientemente nombres con direcciones.

INTRODUCCIÓN

La tecnología ha logrado avances impresionantes en cuanto a comunicación se refiere. En los últimos años, el Internet es una de las vías más importantes de comunicación a nivel mundial en la cual se ha visto un desarrollo tecnológico impresionante tanto para las compañías que desarrollan *software* como para las compañías que desarrollan *hardware*. Estos cambios han llevado al congestionamiento del Internet.

La versión actual de IP (conocida como versión 4 o IPV4) no ha cambiado sustancialmente desde la publicación de RFC 791 en 1981. IPV4 ha demostrado su robustez, facilidad de implementación e interoperabilidad, y ha superado la prueba que representa ampliar una red interna para convertirla en un servicio global de las dimensiones actuales de Internet. Esto es un tributo a su diseño inicial.

Las direcciones IPV4 son relativamente escasas, lo que ha obligado al grupo de trabajo de ingeniería de Internet (IETF, *Internet Engineering Task Force*) a desarrollar un conjunto de protocolos y estándares conocidos como IP versión 6 (IPV6). Esta nueva versión, antes denominada IP: la siguiente generación (*IP-The Next Generation* o IPng), incorpora los conceptos de muchos métodos propuestos para actualizar el protocolo IPV4. El diseño de IPV6 se ha diseñado intencionalmente para que afecte lo menos posible a los protocolos de nivel superior e inferior al evitar que se agreguen aleatoriamente nuevas características.

Es por ello que este trabajo tiene como fin proporcionar al lector, la transición de un protocolo a otro utilizando para ello los servicios de asignación dinámica de IP's (DHCP) como también la resolución de nombres (DNS), describiendo las partes en que se compone el protocolo IPV4, posteriormente definiendo los componentes del protocolo IPV6.

En el primer capítulo se definen conceptos básicos sobre las diferentes topologías que se utilizan para la construcción de las redes como también los protocolos que existen para la comunicación entre las mismas.

En el segundo capítulo se describe la estructura de direccionamiento del protocolo IPV4, como también los componentes que forman el datagrama de dicho protocolo.

En el capítulo 3, se define el protocolo IPV6, la nueva nomenclatura a utilizar para las direcciones con tamaño de 128 *bits*. Las nuevas extensiones de la cabecera del Internet, como también la definición del protocolo ICMPV6 que es el encargado de la detección de errores de dichos paquetes o datagramas.

En el cuarto capítulo se define el concepto y funcionamiento de los servicios de asignación dinámica de IP's (DHCP) y resolución de nombres (DNS), sobre el protocolo IPV4.

En el capítulo 5, se implementa la guía para la transformación de los servicios DHCP y DNS, utilizando la nueva generación de IP's (IPV4 – IPV6).

1. INTERNET Y TERMINOLOGÍA DE REDES

1.1 Historia del Internet

El término Internet se deriva del término *internetworking* (trabajo en interred) que, como lo implica, quiere decir el concepto de redes conectándose con otras redes. Cuando las personas usan el término Internet, describen probablemente el Internet global que evolucionó de un proyecto de investigación Federal en la década de 1960.

En la década de 1960, El Departamento de Defensa de EE.UU. investiga centros localizados alrededor del país necesitados de compartir datos y recursos de la computadora. Una red de computadoras desarrollada para este propósito específico fue llamada Agencia del Proyecto de la Investigación Avanzada conectada para Trabajo en Red (*Advanced Research Project Agency NETwork,* ARPANET). Desde 1985, la Fundación Nacional de la Ciencia (*National Science Foundation,* NSF) financió el segmento columna vertebral primaria de la Red Internacional para EE.UU., llamado *NSFNet*. Al final de Abril de 1995, *NSFNet* pasó los deberes de la columna vertebral primaria de la Red Internacional a compañías comerciales tal como Advanced Network Services, Inc. y Sprint Corporation.

Protocolos Red Internacional y normas son establecidas por comités Red Internacional comprendidos por profesionales desde la comunidad Red Internacional. Uno de estos comités, Tablero de Arquitectura Red Internacional (Internet *Architecture Board*), vigila la arquitectura Red Internacional y Tarea Fuerza para Ingeniería Red Internacional (*Internet Engineering Task Force* IETF), que establece normas tal como esquemas de direccionamiento IP.

1.2 ¿Qué es una red informática?

Se puede definir una red informática como un sistema de comunicación que conecta ordenadores y otros equipos informáticos entre sí, con la finalidad de compartir información y recursos.

A través de la compartición de información y recursos en una red, los usuarios de los sistemas informáticos de una organización podrán hacer un mejor uso de los mismos, mejorando de este modo el rendimiento global de la organización. Entre las ventajas que supone el tener instalada una red, pueden citarse las siguientes:

- Mayor facilidad en la comunicación entre usuarios
- Reducción en el presupuesto para *software*
- Reducción en el presupuesto para *hardware*
- Posibilidad de organizar grupos de trabajo
- Mejoras en la administración de equipos y programas
- Mejoras en la integridad de datos
- Mayor seguridad para acceder a la información

1.3 Tipos de redes

1.3.1 Extensión

De acuerdo con la distribución geográfica, se habla de redes:

- Locales o LAN.
- Metropolitanas o MAN.
- Extensas o WAN.

1.3.1.1 LAN

Las redes de área local, generalmente llamadas LAN *(Local Area Networks)*, son redes de propiedad privada dentro de un solo edificio o campus de hasta unos cuantos kilómetros de extensión. Se usan ampliamente para conectar computadoras personales y estaciones de trabajo en oficinas de compañías y fábricas con objeto de compartir recursos (por ejemplo, impresoras) e intercambiar información.

1.3.1.2 MAN

Una red de área metropolitana, o MAN *(Metropolitan Area Network)* es básicamente una versión más grande de una LAN y normalmente se basa en una tecnología similar. Podría abarcar un grupo de oficinas corporativas cercanas o una ciudad y podría ser privada o pública.

Una MAN puede manejar datos y voz, e incluso podría estar relacionada con la red de televisión por cable local. Una MAN sólo tiene uno o dos cables y no contiene elementos de conmutación, los cuales desvían los paquetes por una de varias líneas de salida potenciales. Al no tener que conmutar, se simplifica el diseño.

1.3.1.3 WAN

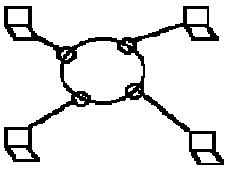
Una red de área amplia, o WAN *(Wide Area Network),* se extiende sobre un área geográfica extensa, a veces un país o un continente; contiene una colección de máquinas dedicadas a ejecutar programas de usuario (es decir, de aplicación). Siguiendo el uso tradicional, llamaremos a estas máquinas *hosts***.** Las *hosts* están conectadas por una subred de comunicación, o simplemente subred. El trabajo de la subred es conducir mensajes de una *host* a otra, así como el sistema telefónico conduce palabras del que habla al que escucha.

1.3.2 Topología

1.3.2.1 Anillo

Es una de las tres principales topologías de red. Las estaciones están unidas una con otra formando un círculo por medio de un cable común. Las señales circulan en un solo sentido alrededor del círculo, regenerándose en cada nodo.

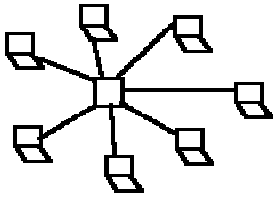
Figura 1. Topología de anillo



1.3.2.2 Estrella

Es otra de las tres principales topologías. La red se une en un único punto, normalmente con control centralizado, como un concentrador de cableado.

Figura 2. Topología estrella



1.3.2.3 Bus

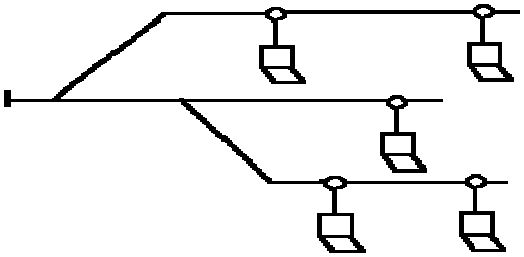
Es la tercera de las topologías principales. Las estaciones están conectadas por un único segmento de cable. A diferencia del anillo, el bus es pasivo, no se produce regeneración de las señales en cada nodo.

Figura 3. Topología de bus

1.3.2.4 Árbol

Esta estructura de red se utiliza en aplicaciones de televisión por cable, sobre la cual podrían basarse las futuras estructuras de redes que alcancen los hogares. También se ha utilizado en aplicaciones de redes locales analógicas de banda ancha.

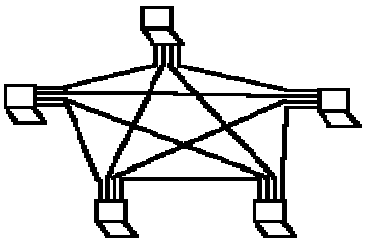
Figura 4. Topología de árbol



1.3.2.5 Trama

Esta estructura de red es típica de las WAN, pero también se puede utilizar en algunas aplicaciones de redes locales (LAN). Los nodos están conectados cada uno con todos los demás.

Figura 5. Topología de trama



1.4 Medio físico

El medio físico es sobre el que se envían las señales eléctricas para realizar la transmisión de la información.

1.4.1 Cables de cobre

Los cables de cobre utilizados para transmisión son conductores clásicos que en ocasiones no son de este metal, sino aleaciones que mejoran las características eléctricas del cable.

Los tipos de cables más utilizados para la transmisión de datos son:

1.4.1.1 Coaxial

El término coaxial quiere decir eje común ya que un cable coaxial está formado por un conductor central rodeado de una capa de material aislante o dieléctrico, rodeada a su vez por una malla de hilos conductores cubierta por una funda de material aislante y protector, formando así cuatro capas concéntricas.

1.4.1.2 Twinaxial

Este tipo de cable es una variación del coaxial que dispone de dos conductores centrales, envueltos cada uno en un aislante.

1.4.1.3 Par trenzado apantallado *(*STP*, Shielded Twisted Pair)*

Este tipo de cable está formado por grupos de dos conductores cada uno con su propio aislante trenzados entre sí y rodeados de una pantalla de material conductor, recubierta a su vez por un aislante. Cada grupo se trenza con los demás que forman el cable y, el conjunto total se rodea de una malla conductora y una capa de aislante protector. Esta disposición reduce las interferencias externas, las interferencias entre pares y la emisión de señales producidas por las corrientes que circulan por el cable.

1.4.1.4 Par trenzado sin pantalla *(*UTP*, Unshielded Twisted Pair)*

En este tipo de cable, los conductores aislados se trenzan entre sí en pares y todos los pares del cable a su vez. Esto reduce las interferencias entre pares y la emisión de señales.

1.4.2 Fibra óptica

Las fibras se utilizan como guías de haces de luz láser sobre los cuales se modulan las señales que transmiten la información, permitiendo que la luz describa trayectorias curvadas, necesarias para poder instalar las redes en los edificios.

1.4.3 Radio

Las ondas de radio fueron el primer medio utilizado para transmitir información y, gracias a los avances tecnológicos como la telefonía celular y el auge de los equipos portátiles, se están convirtiendo en uno de los medios de transmisión más utilizados en la actualidad.

1.4.4 Luz

La luz se utilizó aún antes que la radio para transmitir información, ya los griegos utilizaban espejos para comunicarse con sus barcos en el mar. Pero ha sido necesario mejorar los sistemas de producción de luz láser para permitir transmitir información electrónica con velocidades similares a los cables.

1.5 Protocolos de red

1.5.1 IPX/SPX

Internet *Packet eXchange / Sequenced Packet eXchange.* Es el conjunto de protocolos de bajo nivel utilizados por el sistema operativo de red *Netware de Novell*. SPX actúa sobre IPX para asegurar la entrega de los datos.

1.5.2 DECnet

Es un protocolo de red propio de *Digital Equipement Corporation* (DEC), que se utiliza para las conexiones en red de los ordenadores y equipos de esta marca y sus compatibles. Uno de sus componentes, LAT *(Local Area Transport*, transporte de área local), se utiliza para conectar periféricos por medio de la red y tiene una serie de características de gran utilidad como la asignación de nombres de servicio a periféricos o los servicios dedicados.

1.5.3 X.25

Es un protocolo utilizado principalmente en WAN y, sobre todo, en las redes públicas de transmisión de datos. Funciona por conmutación de paquetes, esto es, que los bloques de datos contienen información del origen y destino de los mismos para que la red los pueda entregar correctamente aunque cada uno circule por un camino diferente.

1.5.4 TCP/IP

Este no es un protocolo, sino un conjunto de protocolos, que toma su nombre de los dos más conocidos: TCP (*Transmission Control Protocol*, protocolo de control de transmisión) e IP (*Internet Protocol,* protocolo de Internet). Esta familia de protocolos es la base de la red Internet, la mayor red del mundo. Por lo cual, se ha convertido en el más extendido.

1.5.5 *AppleTalk*

Este protocolo está incluido en el sistema operativo del ordenador *Apple Macintosh* desde su aparición y permite interconectar ordenadores y periféricos con gran sencillez para el usuario, ya que no requiere ningún tipo de configuración por su parte; el sistema operativo se encarga de todo.

1.5.6 NetBEUI

NetBIOS Extended User Interface (Interfaz de usuario extendido para *NetBIOS*). Es la versión de Microsoft del *NetBIOS (Network Basic Input/Output System,* sistema básico de entrada /salida de red), que es el sistema de enlazar el *software* y el *hardware* de red en los PCs. Este protocolo es la base de la red de Microsoft Windows para Trabajo en Grupo.

1.6 Equipos de Red

1.6.1 NIC/MAU (Tarjeta de red)

Network Interface Card (Tarjeta de interfaz de red) o *Medium Access Unit* (unidad de acceso al medio). Es el dispositivo que conecta la estación (ordenador u otro equipo de red) con el medio físico. A veces, es necesario, además de la tarjeta de red, un transceptor. Este es un dispositivo que se conecta al medio físico y a la tarjeta.

1.6.2 Concentradores

Son equipos que permiten estructurar el cableado de las redes. La variedad de tipos y características de estos equipos es muy grande. En un principio eran sólo concentradores de cableado, pero cada vez disponen de mayor número de capacidades, como aislamiento de tramos de red, capacidad de conmutación de las salidas para aumentar la capacidad de la red, gestión remota, etc.

1.6.3 Repetidores

Son equipos que actúan a nivel físico. Prolongan la longitud de la red uniendo dos segmentos y amplificando la señal, pero junto con ella amplifican también el ruido. La red sigue siendo una sola, con lo cual, siguen siendo válidas las limitaciones en cuanto al número de estaciones que pueden compartir el medio.

1.6.3 *Bridges*

Son equipos que unen dos redes actuando sobre los protocolos de bajo nivel, en el nivel de control de acceso al medio. Los *bridges* producen las señales, con lo cual no se transmite ruido a través de ellos.

1.6.4 *Router*s

Son equipos de interconexión de redes que actúan a nivel de los protocolos de red. Permite utilizar varios sistemas de interconexión mejorando el rendimiento de la transmisión entre redes. Su funcionamiento es más lento que los *bridges* pero su capacidad es mayor. Permiten, incluso, enlazar dos redes basadas en un protocolo, por medio de otra que utilice un protocolo diferente.

1.6.6 *Gateways*

Son equipos para 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.

2. PROTOCOLO IPV4

2.1 Introducción

El protocolo IP *(Internet Protocol)* fue diseñado para interconexión de redes. IP se ocupa de la transmisión de bloques de datos, llamados datagramas de origen a destino, donde orígenes y destinos son *hosts* identificados por direcciones de una longitud fija. IP también se encarga de la fragmentación y reensamblado de datagramas, si fuera necesario.

El protocolo IP implementa dos funciones básicas: Direccionamiento y fragmentación.

El módulo Internet usa las direcciones contenidas en la cabecera de los datagramas para hacer llegar a estos a sus destinos. Asimismo, existen otros campos en la cabecera que permiten gestionar la fragmentación y posterior reensamblado de datagramas, para poder transmitir a través de redes que trabajen con tamaños de paquetes pequeños.

El módulo Internet reside en cada *host* integrado en la Internet, y en cada *gateway* interconectando redes. Estos módulos siguen reglas comunes para interpretar las direcciones y para realizar la fragmentación y el reensamblado de datagramas. Adicionalmente, estos módulos (especialmente en los *gateways)* están provistos de mecanismos para tomar decisiones sobre el enrutamiento de los datagramas.

2.2 Direccionamiento IPV4

Para identificar cada máquina en la Internet, se le asigna un número denominado dirección IP. Este número es asignado de tal forma que se consigue una gran eficiencia al encaminar paquetes, ya que codifica la información de la red en la que está conectado, además de la identificación del *host* en concreto.

Cada dirección de Internet tiene una longitud fija de 32 *bits*. Los *bits* de las direcciones IP de todos los *host* de una red determinada comparten un prefijo común. Conceptualmente, cada dirección IP es una pareja formada por una identidad de red y una identidad de *host*, donde la identidad de red identifica a la red, e identidad de *host*, a un *host* determinado dentro de esa red.

Para que exista una flexibilidad en la asignación de direcciones, existen tres formatos básicos de representación de direcciones. La elección de uno de éstos formatos dependerá del tamaño de la red. Además de los tres formatos básicos, existe uno para *Multicasting*, usado para envío de mensajes a un grupo de *hosts*, y otro reservado para un uso futuro.

La estructura de los diferentes formatos es la que sigue:

Tabla I. Estructura de los formatos de direccionamiento IPV4

Clase A

1	Identificador de Red	Identificador de *Host*
	7 *bits*	24 *bits*

Clase B

1	0	Identificador de Red	Identificador de *Host*
		14 *bits*	16 *bits*

Clase C

1	1	0	Identificador de Red	Identificador de *Host*
			21 *bits*	8 *bits*

Clase D

1	1	1	0	Dirección *Multicast*
				28 *bits*

Clase E

1	1	1	1	0	Espacio reservado para futuro uso
					27 *bits*

Se puede observar que los primeros *bits* identifican la clase de dirección IP, que va seguido de un prefijo de identificación de red, y seguido de un identificador de *host*. La clase D se usa para transmitir un mismo mensaje a un grupo de *hosts* determinado.

La clase A se usa para grandes redes que tengan más de 2^{16} (65536) *hosts*. La clase B se usa para redes de tamaño intermedio, entre 2^{8} (256) y 2^{16} *hosts*. Finalmente, la clase C corresponde a redes con menos de 256 *host*s.

El cuarto tipo, el D, se dedica a tareas de *Multicasting*.

Para asegurar que la parte de identificación de red de una dirección Internet es única, todas las direcciones son asignadas por una autoridad central, el Centro de Información de Red (NIC*, Network Information Center*).

Esta autoridad central tan sólo asigna el prefijo de red de la dirección y delega la responsabilidad de asignar las direcciones de *host* individuales a la organización solicitante. A las redes de área local con pocos ordenadores (menos de 255) se le asignan direcciones de la clase C, pues se espera que surjan un gran número de ellas. A redes muy grandes, como ARPANET, se les asigna la clase A, ya que se espera que no surjan demasiadas.

A la hora de trabajar con direcciones IP, usamos la notación decimal. La dirección expresada de esta forma vendrá dada por cuatro enteros positivos separados por puntos, donde cada entero se corresponde con el valor de un octeto de la dirección IP.

Según lo comentado anteriormente, una dirección IP identifica a un *host*, pero esto no es estrictamente cierto. Por ejemplo, si un *gateway* está conectado a dos redes diferentes, no podemos asignarle una dirección IP única, ya que las dos redes tienen su propia dirección de red. En este caso, hay que asignar una dirección diferente según la conexión, con lo que la dirección IP no especificaría una máquina en particular, sino una conexión a una red.

Según esto, un *gateway* que conecte 'n' redes tendrá 'n' diferentes dirección IP, según la conexión establecida.

Otra consecuencia es que si un *host* se mueve de una red a otra, su dirección IP deberá cambiar según la red en la que se encuentre.

Como debilidades del protocolo podemos indicar que si una red crece por encima de lo que su clase le permite direccionar (por ejemplo una red de clase C que crezca por encima de los 255 *host*) deberá cambiar todas sus direcciones a la clase B, proceso muy costoso y en el que sería muy difícil encontrar errores.

2.2.1 Direcciones especiales

Existen algunas combinaciones de 0's y 1's que no se asignan como dirección IP, sino que tienen asociado un significado especial.

Las distintas combinaciones son las indicadas a continuación:

Tabla II. Direcciones especiales

Toda a 0's		Identifica al propio *host*
Todo a 0's	Identificador de *host*	Identifica al *host* en su red.
Todo a 1's		Multidifusión limitada en la propia red.
Identificador de red	Todo a 1's	Multidifusión a todos los *host*s de la red indicada
127	Contenido	Bucle local

Los dos primeros casos sólo pueden ser usados al arrancar el sistema (en máquinas sin unidad de almacenamiento fijo) y nunca se usan como una dirección de destino válida. En cualquier caso, sólo se usan de forma temporal mientras el *host* aprende su dirección IP.

El tercer caso es la denominada dirección de multidifusión de red local, o dirección de multidifusión limitada, que permite difundir un mensaje a toda la red local independientemente de su dirección IP asignada. Un *host* puede usar esta dirección como parte de un procedimiento de comienzo antes de conocer su dirección IP o la dirección IP de su red.

La dirección de multidifusión dirigida a una red nos permite enviar un mensaje a todas las estaciones situadas en una red determinada. Es una herramienta muy potente, ya que permite enviar un sólo paquete que será difundido en toda la red. Esta dirección se usa de forma restringida, ya que supone una gran carga de trabajo en redes grandes.

La dirección de bucle local está diseñada para pruebas y comunicación entre procesos en la máquina local. Si un programa envía un mensaje a esta dirección, el módulo Internet le devolverá los datos sin enviar nada a la red. De hecho, nunca debe haber en la red un paquete de este tipo, ya que no es una dirección de red válida.

2.2.2 Subredes

En el direccionamiento IP a cada red física se le asigna una única dirección de red, los *hosts* de esa red llevan su dirección de red incluida en su dirección individual.

Este esquema de direccionamiento tiene un fallo: el crecimiento exponencial de Internet. Cuando el protocolo IP fue diseñado, nadie imaginó que pudiera hacer cientos de miles de pequeñas redes de ordenadores personales.

Al existir tantas redes, aparte del problema administrativo de asignar direcciones a todas ellas, existe el problema de que las tablas de encaminamiento de los *gateways* son excesivamente largas, y la ocupación de ancho de banda de la red usada en transmitir esas tablas es alta. Para solucionar esto, se debe disminuir el número de direcciones de red asignadas sin alterar el esquema de direccionamiento original. Para conseguirlo hay que hacer que un mismo prefijo de red IP que pueda ser compartido por múltiples redes físicas.

Este objetivo se alcanzará modificando los procedimientos de encaminamiento y todas las máquinas que se conectan a esas redes deben entender las convenciones usadas.

Para asignar una única dirección IP a varias redes físicas, se usa la máscara de subred, conceptualmente, ya que el añadir subredes sólo varía la interpretación de las direcciones IP ligeramente. El resultado es una forma de direccionamiento jerárquico que conlleva un encaminamiento jerárquico.

La ventaja de usar direccionamiento jerárquico es que permite el crecimiento con facilidad, ya que un *gateway* no necesita conocer con tanto detalle los destinos remotos como los cercanos. Una desventaja es la dificultad de establecer el sistema, y mucho más de cambiarlo una vez establecido.

El estándar IP para subredes especifica que para cada red física en una localización que use subredes hay que escoger una máscara de subred de 32 *bits*. De esta forma, la parte local asociada con el identificador de *host* se puede dividir en dos, una asociada con el identificador de subred, y otra con el *host* en particular. En la máscara adquieren valor 1 los *bits* situados en las posiciones para la indicación de la clase de dirección y para el prefijo de red. Y dentro de la parte local, adquieren valor 1 los *bits* destinados a identificar la subred.

2.3 El datagrama de Internet

2.3.1 El datagrama Internet

Un datagrama es la unidad básica de transferencia entre la Internet, y se descompone en cabecera y datos. La estructura de un datagrama Internet es la siguiente:

Tabla III. Datagrama de Internet

Versión	Long.cab	Tipo de servicio	Longitud total	
Identificación			*Flags*	Offset fragmento
Tiempo de vida	Protocolo		FCS cabecera	
Dirección IP fuente				
Dirección IP destino				
Opciones				Relleno
DATOS				

A continuación describiremos cada uno de los campos:

2.3.2 Versión

Este campo ocupa 4 *bits*, e indica el tipo de formato de datagrama. Para el formato descrito, su valor es 4 (IP versión 4).

2.3.3 Longitud de la cabecera

Este campo ocupa 4 *bits*, y especifica la longitud de la cabecera medida en palabras de 32 *bits*, el mínimo valor posible para una cabecera correcta es 5 (5, 32, 160 *bits*), ya que el campo de opciones puede estar presente o no.

2.3.4 Tipo de servicio

Este campo ocupa 8 *bits*, e indica como deberá ser tratado el datagrama. Se divide, a su vez, en cinco subcampos, de la forma siguiente:

Tabla IV. Estructura del campo tipo de servicio

Prioridad	D	T	R	Sin uso
3b	1b	1b	1b	2b

Los 3 *bits* de prioridad, con valores comprendidos entre cero (prioridad normal) y siete (control de red), permiten al remitente indicar la importancia del datagrama. Aunque la mayor parte del *software* y de los *gateways* no usa este campo, es un concepto importante porque permite que en un momento determinado los comandos de control tengan prioridad sobre los datos. Por ejemplo, sin este campo sería imposible implementar algoritmos de control de congestión que no se vieran afectados por la congestión que están intentando controlar.

Los *bits* D, T y R especifican el tipo de transporte que el datagrama solicita. Si están activos, sus significados son:

D activado El datagrama solicita bajo retardo

T activado El datagrama solicita alta capacidad

R activado El datagrama solictia alta fiabilidad.

Es posible que en uno o varios nodos del camino no exista alguna de las facilidades solicitadas, así, estos *bits* son más una ayuda a los algoritmos de encaminamiento que una petición de servicio.

2.3.5 Longitud del datagrama

Este campo ocupa 16 *bits*, e indica la longitud total del datagrama, incluyendo la cabecera y los datos, la longitud se indica en octetos. Con esto, se permite especificar una longitud de hasta 65536 octetos, sin embargo, los datagramas largos resultan intratables a muchos *host*s y redes. El mínimo tamaño que debería aceptar un *host* es de 576 octetos. Se recomienda que los *host*s sólo envíen datagramas de más de 576 octetos y tienen la seguridad de que el destinatario podrá aceptarlos.

El tamaño de 576 octetos se elige para permitir un tamaño razonable del bloque de datos para ser transmitido junto con la cabecera. Así, este tamaño permite un tamaño para el bloque de datos de 512 octetos, junto con 64 octetos para la cabecera. El tamaño máximo de una cabecera es de 64 octetos, y una cabecera normal ronda los 20 octetos, proporcionando un margen de actuación.

Para que el datagrama se transmita de un nodo a otro de la red, deberá ser transportado en un paquete de la red física subyacente. La idea de transportar un datagrama en una trama de red se denomina encapsulamiento.

Para la red física, el datagrama IP es como cualquier mensaje intercambiado entre dos ordenadores, sin que reconozca ni el formato de datagrama ni la dirección de destino IP.

En el caso ideal, todo el datagrama IP cabría en una sola trama de red, haciendo que la transmisión fuese eficiente. Pero como el datagrama puede atravesar en su camino diferentes tipos de redes físicas, no existe una longitud máxima de datagrama que se ajuste a todas ellas. A la longitud máxima de transferencia de datos por trama de una red física se le conoce como unidad de transferencia máxima (MTU*, Maximum Transmision Unit*).

Cuando un datagrama se envía por una red con un MTU menor que su longitud, entonces el datagrama se divide en partes denominadas fragmentos. Al proceso se le conoce como fragmentación, y será comentado posteriormente.

2.3.6 Identificación

Este campo ocupa 16 *bits*, y contiene un número entero que identifica al datagrama. Este número suele asignarse con un contador secuencial en la máquina origen que va asignándolos según nuevos datagramas. Este campo es indispensable en el proceso de reensamblado de fragmentos, cuando un datagrama fue fragmentado.

2.3.7 *Flags*

Este campo ocupa 3 *bits*, e incluye varios *Flags* de control :

Bit 0: Reservado, debe ser 0

Bit 1: (DF) 0 = el datagrama puede fragmentarse,

1 = el datagrama NO puede fragmentarse

Bit 2: (MF) 0 = es el último fragmento

1 = existen más fragmentos

El primer *bit* significativo (*bit* 1) del campo *Flags* es el de no fragmentación, se llama así porque si está activo implica que el datagrama no puede fragmentarse. Este *bit* resulta útil en casos de pruebas de redes y en algunas aplicaciones especiales donde se necesita que el datagrama llegue sin fragmentar. En el caso de que el *gateway* sea incapaz de enviarlo sin fragmentarlo, envía un mensaje de error a la máquina origen.

El *bit* de menor peso del campo *flags* (*bit* 2), es el *bit* de más fragmentos. Este *bit* es útil para la máquina destino, que así puede determinar si ha recibido todos los fragmentos correspondientes a un datagrama. Cuando el *bit* está a cero, indica que es el último fragmento del datagrama. Así, con este *bit* y con el campo de offset de fragmento, la máquina puede comprobar si ya ha recibido todos los fragmentos y puede reensamblar el datagrama original. La máquina destino no puede guiarse sólo por el *bit* de más fragmentos, porque es posible que se reciba el último fragmento antes de recibir algún fragmento intermedio, ya que IP no provee un método para que los datagramas lleguen ordenados.

2.3.8 *Offset* del fragmento

Este campo ocupa 13 *bits*, y especifica el desplazamiento desde el comienzo del campo de datos del datagrama original hasta el comienzo del campo de datos del fragmento, expresado en múltiplos de 8 octetos.

2.3.9 Tiempo de vida

Este campo ocupa 8 *bits*, e indica cuanto tiempo, en segundos, está el datagrama autorizado a permanecer en el sistema Internet. La idea es simple: cuando una máquina pone un datagrama en la Internet, le asigna un tiempo máximo de existencia del mismo. Los *gateways* y *host*s que van procesando el datagrama deben ir decrementando el campo tiempo de vida, y descartarlo de la Internet cuando el tiempo haya expirado.

Es difícil para los *gateways* estimar el tiempo exacto transcurrido desde que el datagrama salió de la máquina anterior, ya que no conocen el retardo inducido por las redes. Para solventar este problema, se siguen dos normas:

a.- Cada *gateway* por el que pasa el datagrama decrementará en 1 el valor del campo.

b.- Para tener en cuenta los casos de *gateways* con gran retardo de tránsito, al llegar el paquete a un *gateway*, éste almacenará la hora local de llegada, y en el momento de enviarlo decrementará el valor del campo según el número de segundos que haya estado en el sistema esperando ser enviado.

Cuando el campo alcanza el valor cero, el datagrama es descartado y se envía un mensaje de error al origen. La idea del tiempo de vida es interesante porque evita que los datagramas estén eternamente circulando por la red en el caso de que las tablas de encaminamiento estén corruptas y los *gateways* envíen los datagramas en círculo.

2.3.10 Protocolo

Este campo ocupa 8 *bits*, e indica cuál fue el protocolo de alto nivel que ha creado los datos que están en el campo datos. La asignación de estos valores se hace por una autoridad centralizada (IANA*, Institute Assigned Numbers Authority*), para que exista acuerdo a través de toda Internet.

2.3.11 FCS cabecera

Este campo ocupa 16 *bits*, y asegura la integridad de la cabecera. La máquina origen ejecuta una serie de operaciones matemáticas sobre el conjunto de la cabecera y pone el resultado en este campo. El receptor hará la misma operación y comparará el resultado para asegurarse de que los datos de la cabecera son correctos. Sólo es verificada la cabecera, para no sobrecargar de trabajo a los *gateways*. Al entregarse estos datos sin comprobar, serán los protocolos de alto nivel los que realicen su propio chequeo.

2.3.12 Dirección IP origen

Este campo ocupa 32 *bits*, e indica la dirección IP de la máquina origen.

2.3.13 Dirección IP destino

Este campo ocupa 32 *bits*, e indica la dirección IP de la máquina destino.

2.3.14 Opciones

Este campo tiene una longitud variable, y puede estar o no presente en la cabecera del datagrama. Esta opcionalidad se refiere a datagramas en particular, no a la implementación específica, cualquier módulo Internet debe implementar esta funcionalidad, tanto en *host*s como en *gateways*.

Cada opción tendrá un campo llamado código de opción, de 1 octeto de longitud, que puede ser suficiente según la opción, si no es así, este campo vendrá seguido de un campo llamado longitud, también de un octeto, y de un campo conteniendo los datos específicos de la opción de longitud variable.

La estructura de un campo llamado código de opción es la siguiente:

Tabla V. Estructura del campo de opciones

Copia	Clase de opción	Número de opción
1 *bit*	2 *bits*	5 *bits*

2.3.14.1 Copia

El primer *bit* del campo es el de copia. Cuando este *bit* está a uno, indica que la opción deberá ser copiada a los diferentes fragmentos en caso de que el datagrama sea fragmentado. Si está a cero, entonces la opción deberá ser copiada sólo en el primer fragmento y no en el resto.

2.3.14.2 Clase de opción

Indica la clase de opción indicada, las diferentes clases son:

00 Datagrama o control de red

01 Reservado para uso futuro

10 Medida y control de errores

11 Reservado para uso futuro

2.3.15 Relleno

La cabecera de un datagrama IP esta alineada a 32 *bits*. Este campo se usa para asegurar que sea así. El sobrante hasta conseguir un tamaño múltiplo de 32 (*bits*), se rellena con 0's.

2.4 Fragmentación

La fragmentación de un datagrama IP es necesaria cuando el tamaño de un datagrama resulta intratable para alguna de las redes que debe atravesar para llegar a su destino.

El campo identificador es usado junto con los de dirección origen, dirección destino y protocolo, para identificar fragmentos a reensamblar. El módulo Internet del origen del paquete debe asignar un identificador único para cada datagrama, que el destino usa para identificar a que datagramas originales pertenecen que fragmento.

El *flag* más fragmentos, está a 1 si el datagrama no es el último fragmento. El campo *offset* de fragmento identifica la localización del fragmento en el datagrama original, indicando el desplazamiento sobre su comienzo.

La estrategia de fragmentación está diseñada para que un datagrama sin fragmentar tenga toda la información relativa a fragmentación a 0's (más fragmentos = 0*, offset* fragmento = 0*)*. Si un datagrama es fragmentado, todos sus fragmentos (menos el último) deben de estar alineados a 8 octetos (su longitud en *bits* debe ser múltiplo de 64).

Para fragmentar un datagrama Internet, un módulo Internet crea n nuevos datagramas y copia los contenidos de la cabecera a todos ellos. El campo datos del datagrama original es dividido en n partes, las cuales deben estar alineadas a 8 octetos. La primera porción de datos se copia en el primer datagrama generado, y se cambia su campo longitud, haciéndolo coincidir con la longitud del primer datagrama. El *flag* más fragmentos es puesto a 1. La segunda porción de datos es copiada en el segundo datagrama, se cambia su campo longitud y más fragmentos de forma similar, y se especifica el desplazamiento en el campo *offset* de fragmento.

Este proceso se repite hasta el último fragmento generado, que tendrá el *flag* más fragmentos a 0, y que no deberá estar alineado a 8 octetos necesariamente.

Para reensamblar los fragmentos, el módulo Internet en el destino combina los fragmentos que tengan el mismo valor en los campos identificador, dirección origen, dirección destino y protocolo. La recombinación se hace copiando la parte de datos de cada fragmento en la posición relativa indicada en el campo *offset* de fragmento. El primer fragmento deberá tener el campo *offset* de fragmento a cero, y el último fragmento el *flag* más fragmentos a cero.

2.5 Encaminamiento del datagrama IP

Se denomina encaminamiento al proceso de elegir un camino por el que enviar un paquete, y *router* al sistema encargado de realizar esa decisión.

El propósito del encaminamiento Internet es el de proveer al usuario de una red virtual de envío de datagramas IP sin conexión (diferentes datagramas pueden seguir diferentes caminos), de una forma transparente y sin importar el número o tipo de redes físicas que el datagrama debe atravesar para llegar a su destino.

Una Internet está formada por múltiples redes físicas interconectadas por máquinas actuando de *gateways*. Cada uno de éstos está unido a dos o más redes físicas.

Los *host*s, al contrario que los *gateways*, suelen estar conectados a una sola red física.

Tanto los *gateways* como los *hosts* participan en el encaminamiento IP. Un *host* conectado a varios *gateways* decidirá por cual de ellos enviar el datagrama, y un *gateway* conectado a varias redes decidirá a cual de ellas enviar el datagrama. Un *host* y un *gateway* pueden coexistir en la misma máquina física, pero el protocolo IP los considera entes totalmente diferentes.

De forma general, podemos dividir el encaminamiento en dos tipos: directo e indirecto.

El encaminamiento directo es la base del sistema Internet, y consiste en la comunicación de dos *hosts* enganchados a la misma red física. El remitente deberá encapsular el datagrama en una trama física, mapear la dirección IP en una dirección física (por ejemplo usando el protocolo ARP, descrito posteriormente), y enviar la trama resultante directamente al destinatario.

Para que un *host* determine si el destino del paquete pertenece a su misma red, sólo tiene que comparar la parte de la dirección IP que identifica a la red y compararla con la de la propia red.

En la práctica, el encaminamiento directo es la fase final o entrega del paquete, pues aunque este atraviese múltiples redes, al final en último *gateway* estará en la misma red física que el destino, y usará encaminamiento directo para enviarle el mensaje.

El encaminamiento indirecto se usa cuando el destino no está en la misma red física que el origen, en este caso, el origen enviará el datagrama a un *gateway*, que determinará a cual de las redes a las que está conectado enviará el datagrama.

El encaminamiento indirecto es más difícil que el directo, ya que el remitente debe identificar un *gateway* donde enviar el datagrama. El *gateway* debe entonces enviar el datagrama hacia su destino final.

En estas situaciones, se utiliza lo que se conoce como tabla de encaminamiento IP, donde cada máquina almacena información sobre posibles destinos y como llegar a ellos. Cuando el *software* de encaminamiento IP necesita transmitir un datagrama, consulta la tabla para decidir dónde enviar el datagrama.

En la tabla se almacenan parejas de dirección de red-dirección del siguiente *gateway*, indicando el *gateway* por el que seguir el camino hacia la red.

Una técnica usada para que las tablas de enrutamiento no sean demasiado grandes es consolidar múltiples entradas en un caso por defecto. Este tipo de técnica en muy útil en redes pequeñas que tienen una sola conexión con el resto de la Internet. En este caso, el encaminamiento consiste sólo en ver si la estación destino está en la propia red, si no es así, se envía en datagrama al único *gateway* de comunicación con el exterior.

También pueden almacenarse direcciones de máquinas específicas en la tabla, que sirven para dar más control al administrador y como medida de seguridad, detección, y corrección de errores.

2.6 El protocolo ARP

2.6.1 El protocolo ARP

Como hemos comentado en el punto anterior, dos máquinas que quieren comunicarse en la misma red física, sólo podrán hacerlo si conocen sus direcciones físicas.

Existen diversas formas de resolver el problema. Si en la red física pudiera escogerse la numeración de las estaciones (por ejemplo red PRONET 10), entonces podemos hacer que su número sea una función simple de su dirección IP.

Pero en el caso de una red *Ethernet*, el problema no resulta tan sencillo de resolver. Cada interfaz *Ethernet* tiene asignada una dirección *hardware* de 48 *bits*, así pues, es imposible codificar la dirección *hardware* en una dirección IP, además, si se sustituye el interfaz *Ethernet*, cambia la dirección física de la estación.

Para resolver este problema, se diseño el protocolo de resolución de direcciones *(*ARP*, Adress Resolution Protocol),* válido para todas las redes que soportan multidistribución.

La idea es simple, si una máquina A necesita saber la dirección física de una máquina B, envía por multidifusión un paquete especial que pide a la máquina con la dirección IP indicada que responda con su dirección física. Una vez recibida la respuesta, A puede enviar paquetes a B directamente, pues conoce su dirección física.

Debido a que la multidistribución es un recurso costoso (consume recursos de red, ya que todos los receptores deben procesar el paquete enviado), suele evitarse su uso lo más posible. Una de las formas de hacer esto es manteniendo en cada máquina una tabla relacionando direcciones IP con direcciones físicas. Además, como en cada petición ARP se encuentra la dirección IP y la dirección física del remitente, todas las máquinas activas pueden actualizar su tabla con el nuevo dato.

Al enviar un mensaje ARP de una máquina a otra, éste debe viajar en una trama física. Para que la máquina destino identifique la trama como ARP, debe llevar un valor en el campo de tipo de trama que lo identifique como tal. En *Ethernet*, este valor es 0806h (en hexadecimal).

El formato del mensaje ARP no es fijo, sino que depende que *hardware* de la red.

El formato de un mensaje ARP para *Ethernet* es el siguiente:

Tabla VI. Formato de un mensaje ARP

<table>
<tr><td colspan="2">Tipo de hardware</td><td>Tipo de protocolo</td></tr>
<tr><td>Long. dir. Física</td><td>long. dir. protocolo</td><td>Operación</td></tr>
<tr><td colspan="3">Dirección física remitente (octetos 0 a 3)</td></tr>
<tr><td colspan="2">Dirección física remitente (octetos 4 a 5)</td><td>dirección IP remitente (octetos 0 y 1)</td></tr>
<tr><td colspan="2">Dirección IP remitente (octetos 2 y 3)</td><td>Dirección física destinatario (octetos 0 y 1)</td></tr>
<tr><td colspan="3">Dirección física destinatario (octetos 2 a 5)</td></tr>
<tr><td colspan="3">Dirección IP destinatario (completa, octetos 0 a 3)</td></tr>
</table>

El campo tipo de *hardware* (16 bits) especifica el tipo de interfaz *hardware* del que se busca la dirección (1 para *Ethernet*). El campo tipo de protocolo (16 bits) indica el tipo de protocolo del que el origen ha enviado la dirección (0800h para IP).

Los campos de longitud de direcciones física y de protocolo permiten usar ARP con diferente *hardware* y protocolos.

El campo operación nos indica el tipo de operación en concreto, si es una petición ARP o una respuesta a una petición.

El resto de los campos indican las direcciones IP y físicas tanto del remitente como del destinatario.

2.6.2 El protocolo RARP

El protocolo RARP *(Reverse Adress Resolution Protocol)* es una variación de ARP, que permite a estaciones sin unidad de almacenamiento fija obtener su propia dirección IP.

Cuando una estación sin unidad de almacenamiento arranca, envía a la red un mensaje multidifusión con su dirección física (obtenida directamente del *hardware*). El servidor de direcciones buscará la dirección física del solicitante y le enviará un mensaje indicándole su dirección IP.

3. PROTOCOLO IPV6

3.1. Introducción

El protocolo de Internet (IP) tiene sus raíces más tempranas en las redes militares de 1970, pero es en la pasada década cuando se hace imparable en el mundo de las redes. Hoy en día, IP se ha establecido por si sola como el vehículo primario para nuestro sistema global de comercio electrónico permitiendo un amplio rango de aplicaciones cliente servidor.

IPV6, la siguiente generación de protocolo de Internet, fue aprobado por el *Internet Engineering Steering Group* el 17 de noviembre de 1994, como una propuesta de estándar. Desde ese momento un gran número de organizaciones de usuarios finales, grupos de estándares y vendedores de redes han estado trabajando juntos en la especificación y pruebas de implementaciones de IPV6.

La versión actual de IP (conocida como versión 4 o IPV4) ha demostrado su robustez, facilidad de implementación e interoperabilidad, y ha superado la prueba que representa ampliar una red interna para convertirla en un servicio global de las dimensiones actuales de Internet. Esto es un tributo a su diseño inicial.

Sin embargo, en el diseño inicial no se previó lo siguiente:

- El reciente crecimiento exponencial de Internet y el inminente agotamiento del espacio de direcciones IPV4.

Las direcciones IPV4 son relativamente escasas, lo que ha obligado a algunas organizaciones a utilizar el traductor de direcciones de red (NAT, *Network Address Translator*) para asignar múltiples direcciones privadas a una sola dirección IP pública. Aunque NAT permite reutilizar el espacio de direcciones privadas, no admite la seguridad basada en estándares en la capa de red o la asignación correcta de todos los protocolos de nivel superior y puede crear problemas cuando se conectan dos organizaciones que utilizan el espacio de direcciones privadas.

Además, la creciente proliferación de dispositivos y aparatos conectados a Internet apunta a que el espacio de direcciones públicas de IPV4 se agotará dentro de un tiempo.

- El crecimiento de Internet y la capacidad de los enrutadores troncales de Internet para mantener grandes tablas de enrutamiento.

Debido a la forma en la que se asignan los identificadores de red IPV4, existen normalmente más de 70.000 rutas en la tabla de enrutamiento de los enrutadores troncales de Internet. La infraestructura actual del enrutamiento de IPV4 en Internet es una combinación de enrutamiento plano y jerárquico.

- La necesidad de una configuración más sencilla.

La mayor parte de las implementaciones actuales de IPV4 deben configurarse manualmente o mediante un protocolo de configuración de direcciones con estado, como el protocolo de configuración dinámica de *host* (DHCP, *Dynamic Host Configuration Protocol*). Ante un número mayor de equipos y dispositivos que utilizan IP, existe la necesidad de emplear una configuración de direcciones más sencilla y automática, así como otros parámetros de configuración no basados en la administración de una infraestructura DHCP.

- El requisito de seguridad en el nivel de IP.

La comunicación privada a través de un medio público como Internet requiere servicios de cifrado que protejan los datos que se envían ante posibles observaciones o modificaciones durante el tránsito. Aunque ahora existe un estándar para ofrecer seguridad a los paquetes de IPV4 (conocida como seguridad de protocolo Internet o IPSec), es opcional y prevalecen las soluciones propietarias.

- La necesidad de facilitar la entrega de datos en tiempo real, también denominada calidad de servicio (QoS, *Quality of Service*).

Aunque existen estándares de QoS para IPV4, el tráfico en tiempo real se basa en el campo *Type of Service* (TOS o tipo de servicio) de IPV4 y en la identificación de la carga, normalmente mediante un puerto UDP o TCP. Por desgracia, el campo *Type of Service* de IPV4 presenta una funcionalidad limitada y con el tiempo han surgido distintas interpretaciones locales. Además, la identificación de la carga mediante un puerto TCP y UDP no es posible cuando la carga de paquetes IPV4 está cifrada.

Para resolver estas preocupaciones, el grupo de trabajo de ingeniería de Internet (IETF, *Internet Engineering Task Force*) ha desarrollado un conjunto de protocolos y estándares conocidos como IP versión 6 (IPV6). Esta nueva versión, antes denominada IP: la siguiente generación *(*IP-*The Next Generation o IPng*), incorpora los conceptos de muchos métodos propuestos para actualizar el protocolo IPV4.

El diseño de IPV6 se ha diseñado intencionalmente para que afecte lo menos posible a los protocolos de nivel superior e inferior al evitar que se agreguen aleatoriamente nuevas características.

3.2. Nomenclatura IPV6

Cuando se desarrolló la actual notación basada en direcciones de 32 *bits*, agrupados en 4 campos de 8 *bits*, el número de *hosts* máximos que por aquel entonces se podían direccionar resultaba muy lejano de las expectativas que por aquel entonces se tenían de la red Internet. Actualmente, aunque se ha visto frenada debido al uso de CIDR, el continuo y también desorbitado crecimiento esta a punto de desbordar la capacidad de IPV4.

3.2.1 Nomenclatura

El nuevo formato, usa una dirección de 128 *bits*, repartidas en 8 campos de 16 *bits,* de la siguiente manera:

23CF:0000:0000:0000:19FC:A96C:B456:FFFF

También se especifica la posibilidad de “comprimir” campos cuyo valor sea 0, con lo que si un campo es ...:0000:... se puede usar ...:0:... en su lugar. Así mismo, si algunos campos vecinos tienen de valor 0, también se pueden comprimir tal y como se explica a continuación, basándose en la dirección de antes:

23CF::FFFF:19FC:A96C:B456:FFFF

Donde se ha cambiado la cadena :0000:0000: por sólo :: aunque no se pueden comprimir a la vez, 2 grupos colocados en lugares no contiguos. Por ejemplo, en la dirección FFFF:0:0:0:FFFF:0:0:0, se podría comprimir en FFFF::FFFF:0:0:0 o en FFFF:0:0:0:FFFF::, pero nunca en FFFF::FFFF:: porque no se podría distinguir cual de los campos se usa de relleno. Explicado en otras palabras, cuando un campo vale ::, se rellena con 0 hasta completar la dirección.

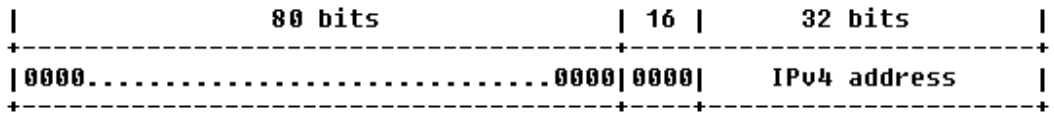
Como es lógico, las direcciones reservadas como *loopback*, de *broadcast* y usadas en *intranets*, se han visto modificadas, aunque, como se explicará mas adelante, estas últimas no necesitarán de ser actualizadas si no es preciso.

Direcciones especiales:

		IPV4	IPV6	
Loopback	---	127.0.0.1	0:0:0:0:0:0:0:1	|| ::1
Unspecified Adresss	---	0.0.0.0	0:0:0:0:0:0:0:0	|| ::

También se ha pensado en las redes que comparten las dos versiones de IP, y se accederá a ellas del siguiente modo:

Figura No. 6. Las dos versiones de IP (IPV4 – IPV6)

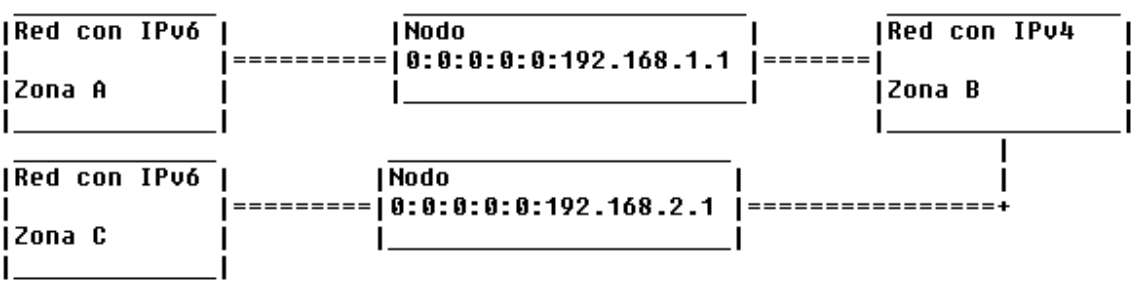


Fuente: IPV6, el futuro de Internet, http://www.undersec.com

Para conectarse a la máquina *Host1* desde *Host2*, se debe especificar la dirección IP en el formato de la versión 6, de la siguiente manera 0:0:0:0:0:FFFF:192.168.1.2 y *Host2* es visto desde *Host1* como 192.168.1.3

Si son varias redes las que hay en juego, se procede a practicar el *tunneling* entre versiones por parte de los nodos.

Figura 7. *Tunneling* entre versiones por parte de los nodos

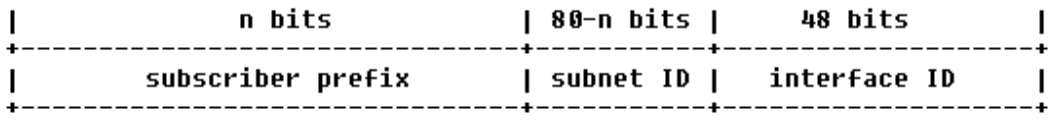


Fuente: IPV6, el futuro de Internet, http://www.undersec.com

De esta forma, se pueden comunicar sin problemas, la Zona A con la Zona C, sin necesidad de actualizar la Zona B y de una forma elegante y transparente para los usuarios.

Para identificar a un *host*, perteneciente a nuestra subred, en la IPV4, se hacía mediante la comparación de nuestra dirección con la máscara de red. Esto supuso una manera cómoda de trabajar, aunque propició el despilfarro de muchísimas direcciones IP, hasta que fue frenado de manera considerable por el uso de CIDR. Como es lógico, IPV6 realiza esta tarea de una manera más eficiente mediante identificadores. El uso de estos identificadores facilitará tanto la asignación como la identificación de una red, independientemente de su tamaño.

Figura 8. Uso de identificadores tanto para la asignación como la identificación de una red



Fuente: IPV6, el futuro de Internet, http://www.undersec.com

En la figura 8 se puede observar la zona de 80 *bits* para el identificador de subred y los 48 *bits* finales, que son asignados mediante la dirección *hardware* del interfaz de red, en este caso, una tarjeta de red que cumple el estándar IEEE-802. Esto a su vez, permitiría una manera muy fácil de asignar direcciones IP sin mayor complicación que la asignación de el ID de subred común, por lo que el administrador, no se tendría que preocupar de asignar las direcciones últimas de *host*.

3.3. Cabeceras IPV6

3.3.1 Cabecera estándar

También, como es lógico, el formato del cabecera de IP se ha visto alterado, y se han añadido muchas e interesantes opciones, algunas ya presentes en IPV4, y se ha procedido a la eliminación de las no usadas o deficientemente implementadas.

Según esto, una cabecera IPV6 quedaría dividida en los siguientes campos:

Versión (4 *bits*):

Identifica la versión, y lógicamente, tiene el valor 6

Traffic Class (Clase de tráfico) (8 *bits*):

Esto renueva y amplía en anterior concepto de TOS *(Type Of Service)*. Los valores de este campo no se encuentran estandarizados por el momento, por lo que su valor inicial, debería ser 00000000 por el momento.

Flow Label (Etiqueta de flujo) (20 *bits*):

En este campo se pueden comunicar diferentes valores, para el manejo del paquete para su enrutamiento, reserva de ancho de banda, que mejorará la recepción de vídeo y sonido sobre IP, a parte de descargar al *router* de parte del trabajo en lo que a administración de calidad de servicio se refiere.

Payload Length (Tamaño de *Payload*) (16 *bits* como *unsigned integer*):

Expresa la longitud de la carga del paquete en *bytes* (octetos)

Next Header (Siguiente cabecera) (8 *bits*):

Comunica el tipo de cabecera siguiente a la de IPV6. Por ejemplo, 6 para TCP. Los valores son los mismos que usa IPV4 aunque en la anterior especificación, este campo tomaba el nombre de *Protocol.*

Hop Limit (8 *bits* como unsigned integer):

Este campo sustituye al TTL *(Time To Live)* de IPV4, y tiene un valor inicial determinado por el *host* emisor en un rango de 0 a 255. Cuando un paquete llega a un *router*, a este valor se le resta 1 y si este llega a 0, es descartado por el mismo.

Source Address (Dirección origen) (128 *bits*):

Dirección de origen del paquete.

Destination Address (Dirección destino) (128 *bits*)

Aunque como se explica en la sección, se puede añadir la ruta por la cual se debe encaminar el paquete.

3.3.2 Extensiones de cabecera

Como novedad, se implementa el concepto extensiones de cabeceras. Estas, son como añadidos a la cabecera estándar, lo que hace más modulables y extensibles las posibilidades de IPV6. Estas extensiones, se identifican mediante el campo "*Next Header"*. Los nuevos identificativos son:

Hop-by-Hop Options (Valor de *next header* de la cabecera anterior 0)

Routing (Valor de *next header* de la cabecera anterior 43)

Fragment (Valor de *next header* de la cabecera anterior 44)

Destination Options (Valor de *next header* de la cabecera anterior)

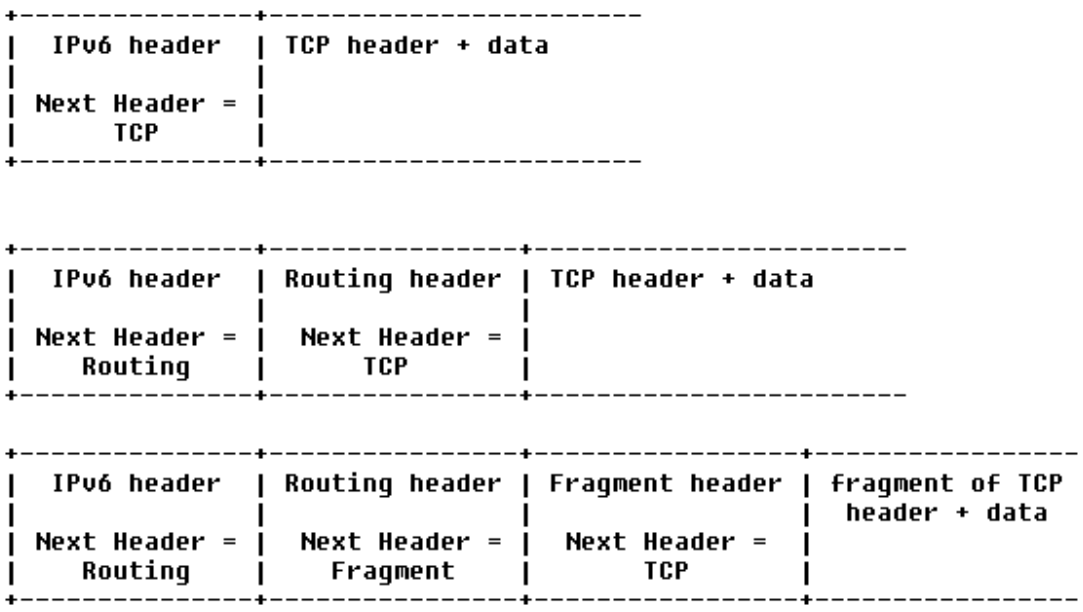
Authentication (Valor en el campo *next header* de la cabecera anterior 51)

Encapsulating Security Payload (Valor de *next header* de la cabecera anterior50)

Last header (Valor en el campo *next header* de la cabecera anterior 59)

Destination Options (Valor en el campo *next header* de la cabecera anterior 60)

Figura 9. Extensiones de cabecera



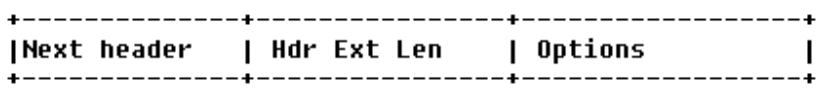
Fuente: IPV6, el futuro de Internet, http://www.undersec.com

La figura 9 puede ayudar a comprender el importante alcance de incluir esta capa dentro de un datagrama. Cabe destacar que una versión propietaria puede incluir sus propias extensiones. A continuación se definen brevemente las cabeceras de extensión del protocolo IPV6. Además se describirán los identificadores que más sobresalen:

3.3.2.1 *Hop-by-Hop Options* (Opciones salto por salto):

Son opciones, cuyo valor debe ser revisado por todos los nodos por los que el paquete es encaminado. Tiene el siguiente formato:

Figura 10. Formato de Hop-by-Hop Options



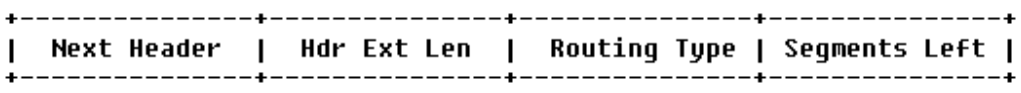
Fuente: IPV6, el futuro de Internet, http://www.undersec.com

En el campo *Next Header* se especifica el tipo de cabecera inmediatamente siguiente a esta, indicándolo mediante un selector de 8 *bits*. En el campo *Hdr Ext Len (Header Extended Lenght)* se especifica la longitud del campo *options*, exceptuando los primeros 8 *bytes*. Se indica como un valor de 8 *bits unsigned integer*. En el campo *Options*, se incluyen las opciones que se especifican en la sección 3.4.2. Su longitud es variable aunque la suma de la longitud de la cabecera *Hop-by-Hop* debe ser múltiplo de 8 *bytes*.

3.3.2.2 *Routing* (**Encaminamiento**)

Esta cabecera, posibilita una de las funciones más atractivas de la versión 6 de IP. Su función, es la de poder especificar el camino exacto que preferentemente debe recorrer un paquete hasta llegar a su destino. El formato es el siguiente (Figura 11):

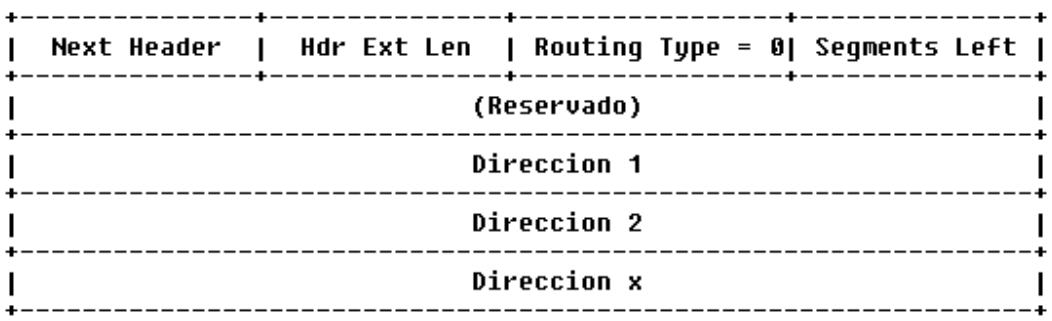
Figura 11. Formato del *Routing*



Fuente: IPV6, el futuro de Internet, http://www.undersec.com

En el campo *Next Header* se especifica el tipo de cabecera inmediatamente siguiente a ésta, indicándolo mediante un selector de 8 *bits*. En el campo *Hdr Ext Len* se especifica la longitud de la cabecera *routing*, exceptuando los primeros 8 *bytes*. Se indica como un valor de 8 *bits unsigned integer*. El campo *Routing Type* es un selector de 8 *bits* de tamaño, que indica el tipo de cabecera *routing* que de momento es 0. Este tipo de cabeceras de enrutado, tiene el siguiente formato:

Figura 12. Formato del campo *Routing Type*



Fuente: IPV6, el futuro de Internet, http://www.undersec.com

En este tipo de cabecera de enrutado, se especifican en el campo de direcciones la IP por la que el paquete debe pasar para llegar a su destino. Cuando un paquete llega a Dirección 1 ésta coloca su propia dirección en el último lugar y envía el paquete a la Dirección 2 que hace el mismo procedimiento y se decrementa en 1 el campo *Segments Left*. Cuando se llega al destino final, el paquete conserva la ruta completa por la que ha pasado y es utilizada para enviar el retorno.

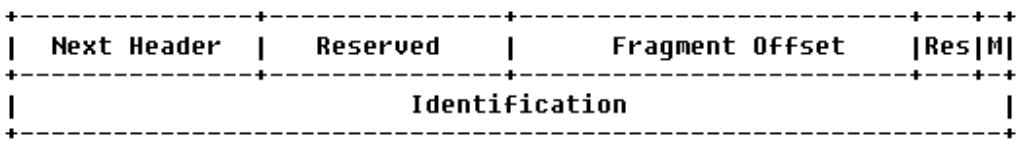
El campo *Segments Left* de la cabecera de *routing*, se indica el número se segmentos de red, que quedan para que el paquete llegue a su destino. En cada salto, este número se decrementa en 1. Su valor se comunica mediante un *unsigned integer.*

3.3.2.3 *Fragment* (Fragmentación)

Al contrario de la IPV4, la fragmentación únicamente debe ser efectuada en el *host* de origen. Si el *router* se encontrara con un paquete de un tamaño mayor al que puede manejar el siguiente salto, debe descartarlo y enviar al *host* de origen un ICMP que comunique la MTU del siguiente salto.

El formato de esta clase de cabeceras es el siguiente:

Figura 13. Formato *Fragment*



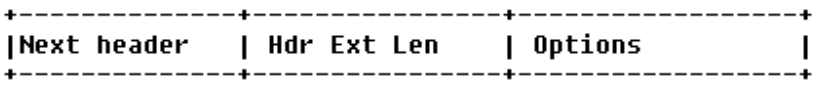
Fuente: IPV6, el futuro de Internet, http://www.undersec.com

En el campo *Next Header* se específica el tipo de cabecera inmediatamente siguiente a esta, indicándolo mediante un selector de 8 *bits*. El campo *Reserved* esta reservado para usos futuros y debe ser inicializado a 0 en el origen e ignorado en la recepción. El valor debe ser de 8 *bytes* del tipo *unsigned integer*. El campo *Fragment Offset* es de una longitud de 13 *bits*, e indica desplazamientos, en bloques de 8 *bytes* relativos al primer paquete de la serie. El campo *Res*, esta reservado para usos futuros y debe ser inicializado en origen a 0 e ignorado por el destino. La *flag* M indica, mediante un *bit*, si el paquete es el último (valor 0) o por el contrario es uno más de la serie (valor 1).

3.3.2.4 *Destination Options* (Opciones de destino)

Esta cabecera indica las opciones que deben procesarse en el destino o en los destinos en el caso de multienvíos. Si va acompañada por cabeceras de enrutamiento, la cabecera de opciones de destino colocada antes de la de enrutamiento se aplica a todos los nodos, mientras que la que se encuentra al final, es sólo aplicable a la dirección de destino. Su formato es el siguiente:

Figura 14. Formato *Destination Options*



Fuente: IPV6, el futuro de Internet, http://www.undersec.com

En el campo *Next Header* se especifica el tipo de cabecera inmediatamente siguiente a ésta, indicándolo mediante un selector de 8 *bits*. En el campo *Hdr Ext Len* se especifica la longitud de la cabecera *Destination Options*, exceptuando los primeros 8 *bytes*. Se indica como un valor de 8 *bits unsigned integer*. El campo *options,* de longitud variable, no se han especificado más opciones que las de relleno hasta que el tamaño sea múltiplo de 8 *bytes*.

3.3.2.5 *Authentication* (Autenticación)

Utiliza un algoritmo para asegurarse de que el paquete IPV6 no se ha alterado a lo largo de su camino. La cabecera también se asegura de que el paquete IPV6 haya llegado de la fuente enumerada en la cabecera IP.

Todas ellas, deben seguir un orden dentro del datagrama:

IPV6 *header*
Hop-by-Hop Options header
Destination Options header
Routing header
Fragment header
Authentication header
Encapsulating Security Payload header
Destination Options header
Upper-layer header

3.4. Detección de problemas y mantenimiento bajo IPV6

3.4.1 Mantenimiento

Con la nueva versión de IP, aparte de la eliminación de máscaras de red, y limitaciones en el número de direcciones disponibles, los desarrolladores del Internet *Protocol* han decidido facilitarle mucho la vida a los administradores. Una red que corra sobre IPV6, será una red fácil de administrar, con menos carga en los *routers* y con muchos problemas menos de tráfico que la "antigua" IPV4. Esto no quiere decir que una red sobre IPV6 sea el paraíso, porque el trabajo de administrador siempre será el de administrar.

Para el montaje de una red sobre IPV6 se han desarrollado nuevos protocolos, y se han extendido muchos de los existentes. Soluciones más completas como la del DHCPV6, permiten que un *host* no sólo obtenga una IP con la seguridad de que es única en su red, si no que sea también dado de alta en el servidor de DNS, solicitando un nombre directamente al servidor DHCPV6.

También el protocolo ARP tiene modificaciones, ya que sus funciones son mucho mejor cubiertas por nuevos procedimientos explicados en la sección 3.4.2 de este capítulo. Problemas como los traslados de equipos a otras redes y/o la reconfiguración de la misma para ampliarla, serán historia en la nueva versión IP.

3.4.2 *Neighbor Discovery*

El *Neighbor Discovery protocol* (protocolo de descubrimiento de vecindad), es el encargado de saber por qué *host*s/redes estamos rodeados. Esto también incluye descubrir las características del medio por donde van a circular los paquetes que salgan de nuestra máquina. El *Neighbor Discovery* hace uso de ICMPV6, para comunicarse. El *Neighbor Discovery* es una combinación de ARP, ICMP *Router Discovery* e ICMP *redirect*, y a su vez se han incorporado nuevas funciones. Entre las funciones de este protocolo, están las de:

Router Discovery (Descubrimiento de *router*): comunica al *host* y a otros *routers* la existencia de un nuevo *router*, o la permanencia / eliminación de los actuales.

Parameter Discovery (Descubrimiento de parámetros): da información a los nodos de una red, sobre el MTU, así como del número máximo de saltos para llegar al exterior.

Prefix Discovery (Descubrimiento de prefijo): comunica al nodo el prefijo de la red que tiene la dirección IPV6 de su subred.

Next-hop determination (Determinación del próximo salto): comunica el nodo más cercano. Este valor puede ser usado para determinar la ruta más corta por la cual se encaminarán los paquetes.

Address Autoconfiguration (Autoconfiguración de dirección): da a conocer a la máquina la IP de un interfaz de red.

Address resolution (Resolución de dirección): llamado también ARP. Se encarga de establecer la correspondencia entre la dirección IP y la de su capa de enlace (Dirección MAC de una *Ethernet* por ejemplo).

Neighbor Unreachability Detection (Detección de inalcanzabilidad de un vecino): detecta la caída o eliminación de un *router* y/o *host*.

Duplicate Address Detection (Detección de Duplicidad de direcciones): comprueba que no hay direcciones IP duplicadas dentro de una red. Puede ser usado antes de dar de alta un nuevo nodo.

Redirect (Redirección): informa a un nodo del mejor camino más corto para alcanzar un destino.

3.5. ICMPV6

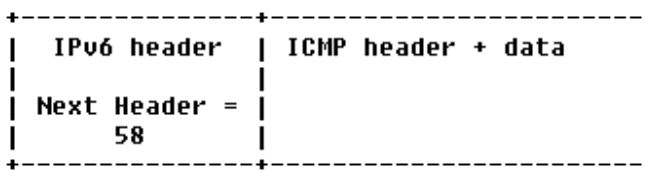
El *Internet Control Messages Protocol* (56 en el campo de *Next Header*), tiene el mismo uso que su antepasado, el ICMPV4. La misión de un ICMP es, sobre todo, la de informar.

3.5.1 Tipos de ICMPV6 y formato

Los mensajes de ICMP se han dividido en 2 clases, los que comunican errores y los que piden/dan información sobre un nodo. Para diferenciarlos, se han adjudicado una numeración del 0 al 127 a los mensajes que contienen información y del 128 al 255, sobre los que informan de algún tipo de error de una petición.

Un paquete ICMPV6 está formado por una cabecera IPV6, y es precedido inmediatamente por una cabecera con valor 58 en el campo *next header*:

Figura 15. **Formato de un paquete ICMPV6**



Fuente: IPV6, el futuro de Internet, http://www.undersec.com

Nótese que este procedimiento es diferente al de IPV4, y que un ICMP puede ser insertado en cualquier tipo de paquetes.

3.5.2 Tipos de ICMPV6 de información

Los mensajes de información pueden ser del tipo:

3.5.2.1 *Echo Request*

Un nodo puede enviar un ICMP *Echo Request* (más conocidos como *pings*), para saber el tiempo de respuesta de otro *host*.

La recepción de ICMP *Echo Request* debe ser comunicada a la capa superior de trasporte.

3.5.2.2 *Echo Reply*

El ICMP *Echo Reply* es enviado como respuesta a un ICMP *Echo Request*. El ICMP *Echo Reply* debe ser trasportado al proceso que originó el ICMP *Echo Request*.

3.5.3 Tipos de ICMPV6 de error

3.5.3.1 *Destination Unreachable*

Un ICMP *Destination Unreachable* es mandado por un *router*, o por cualquier nodo, para informar de la imposibilidad de que un paquete lleve a su destino. No se deberían mandar ICMPV6, si son ocasionados por problemas de congestión de la red. Estos ICMP se dividen en subclases, según el tipo de problema que haya ocasionado su emisión:

-Si el error es ocasionado por un envío de un paquete al nodo erróneo, este enviará un ICMPV6 con código 0.

-Si el error es ocasionado por un envío hacia un destino cerrado por causas administrativas (un *firewall,* por ejemplo), se debe enviar un ICMP de código 1.

-Si el error es ocasionado por la imposibilidad de resolver la dirección IP de un link, se enviará un ICMPV6 con código 3.

-Si el error es ocasionado por un fallo en la capa de trasporte, si el puerto está indisponible para la misma, se enviará un ICMP con código 4. Por ejemplo, un paquete TCP enviado a un puerto UDP.

Un nodo que ha recibido un ICMPV6 *Destination Unreachable* debe comunicarlo a la capa superior del proceso.

3.5.3.2 *Packet Too Big*

Un ICMP *Packet Too Big* es enviado cuando el tamaño máximo de un paquete es superior a la MTU del interfaz de red al que se ha enviado. También es enviado por un *router*, si el siguiente salto es tiene un MTU inferior al tamaño del paquete. Este ICMP puede ser usado para saber el MTU de un *path*.

3.5.3.3 *Time Exceeded*

Si un *router* recibe un paquete con el *Hop limit* a 0, o si es el quien lo tiene que poner a 0, el paquete es descartado y se envía un ICMPV6 *Time Exceeded*. Si un *host* no puede ensamblar un paquete en un tiempo x, descartará todos los fragmentos recibidos y también enviará un ICMP de esta clase. La llegada de un ICMPV6 debe ser notificada a la capa superior de trasporte.

3.5.3.4 *Parameter Problem*

Si un nodo IPV6 al procesar un paquete encuentra un error en uno de los parámetros de sus campos, enviará un ICMP *Parameter Problem* informando al destino de la situación del error en el paquete.

3.5.4 Seguridad e ICMPV6

Los ataques producidos por los ICMP enviados de forma masiva, generalmente para provocar un DOS *(Denial of Service)* y/o la caída de un nodo de una red y/o de una conexión, son lo suficientemente conocidos como para tener que volver a explicarlos. El protocolo IPV6 implementa medios de autentificación que pueden evitar los más comunes.

-Caída por recepción de envíos masivos de ICMP.

-Desconexión de un *host*, por el envío de un atacante al servidor, de ICMP con mensajes de error.

-Falsificación de ICMP.

4. DHCP y DNS en IPV4

4.1 DHCP

4.1.1 Definición

DHCP se basa en el conocido modelo cliente-servidor. Utiliza un protocolo de comunicaciones muy sencillo (basado en UDP sobre IP). Los clientes de una red que utilicen este protocolo utilizan direcciones IP que les "alquila" un servidor (no tiene que ser local). Cada vez que un cliente se inicia, pide una dirección IP o una renovación de la que tiene alquilada actualmente.

El cliente recibe, junto con la dirección, algunos parámetros adicionales: pasarela (*gateway*) por defecto, servidor WINS, servidor DNS, etc. Lo que DHCP consigue es que la asignación y liberación de las direcciones IP en una red sea dinámica y automática; se evita las duplicidades y se optimiza el consumo de direcciones. La intervención del administrador de redes, aun en grandes configuraciones, es mínima.

El protocolo de configuración de *host* dinámica (DHCP) proporciona un espacio de trabajo para pasar información de configuración a los *host*s sobre una red TCP/IP. DHCP se basa en el protocolo BOOTP, añadiendo la capacidad de localización automática de direcciones de red reutilizables y opciones de configuración adicionales.

4.1.2 Componentes

DHCP consiste en dos componentes:

1. Un protocolo que transporta los parámetros de configuración específicos de *host* de un servidor DHCP a un *host*.

2. Un mecanismo para la localización de direcciones de red a *host*s.

IP requiere la configuración de muchos parámetros del *software* de implementación del protocolo. Como IP se puede usar en muchos tipos distintos de *hardware* de red, los valores para esos parámetros no se pueden adivinar o asumir que son correctos por defecto. El uso de un esquema de localización de direcciones distribuidas basada en un mecanismo sondeo/defensa, para descubrir direcciones de red que ya están en uso, no puede garantizar direcciones únicas de red porque los *hosts* no pueden ser capaces de defender sus direcciones de red.

4.1.3 Mecanismos para localizar direcciones IP

DHCP soporta tres mecanismos para localizar direcciones IP:

4.1.3.1 Localización automática

DHCP asigna una dirección IP permanente al *host*.

4.1.3.2 Localización dinámica

DHCP asigna una dirección IP para un periodo de tiempo limitado. Tal dirección de red se llama una *lease*. Este es el único mecanismo que permite la reutilización automática de direcciones que ya no necesita el *host* a las que fue asignado.

4.1.3.3 Localización manual

El administrador de red asigna la dirección del *host*.

4.1.4 Formato de un DHCP

En la figura 16 se observa el formato de un mensaje DHCP.

Figura 16. Formato de un DHCP

4.1.4.1 Código

Indica petición (1) o respuesta (2).

4.1.4.2 TipoHW

El tipo de *hardware*, por ejemplo: *Ethernet* o Redes IEEE 802.

4.1.4.3 Longitud

Longitud de la dirección *hardware* en *bytes. Ethernet* y *token-ring* usan 6, por ejemplo.

4.1.4.4 Saltos

El cliente lo pone a 0. Lo incrementa el *router* que transmite la petición a otro servidor y se usa para identificar bucles.

4.1.4.5 ID de transacción

Se utiliza un número aleatorio para emparejar esta petición de arranque con la respuesta generada.

4.1.4.6 Segundos

Lo fija el cliente. Es el tiempo transcurrido en segundos desde que el cliente comenzó su proceso de arranque.

4.1.4.7 Campo *Flags*

El *bit* más significativo del campo *Flags* se usa como flag de *broadcast*. El resto de los *bits* debe ponerse a cero, y están reservados para uso futuro. Normalmente, los servidores DHCP intentan transportar mensajes DHCP directamente a un cliente usando transporte *unicast*. La dirección de destino en la cabecera IP se pone a la dirección IP DHCP y la dirección MAC se pone a la dirección *hardware* del cliente DHCP. Si un *host* es incapaz de recibir un datagrama IP *unicast* hasta que sepa su dirección IP, entonces este *bit* de *broadcast* se debe activar para indicar al servidor que la respuesta DHCP se debe enviar como un *broadcast* IP y MAC. En otro caso, este *bit* se debe poner a cero.

4.1.4.8 Dirección IP del cliente

Lo fija el cliente. Es una dirección IP conocida ó 0.0.0.0.

4.1.4.9 Tu dirección IP

Lo fija el servidor si el campo de la dirección IP del cliente era 0.0.0.0.

4.1.4.10 Dirección IP del servidor

Lo fija el servidor.

4.1.4.11 Dirección IP del *router*

Lo fija el *router* expedidor si se está usando BOOTP.

4.1.4.12 Direción *hardware* del cliente

Lo fija el cliente. DHCP define una opción de identificador de cliente que se usa para la identificación del cliente. Si no se usa esta opción, el cliente se identifica por su dirección MAC.

4.1.4.13 Nombre del *host* servidor

Nombre del *host* servidor opcional terminado en X'00'.

4.1.4.14 Nombre del fichero de arranque

El cliente lo deja nulo o especifica un nombre genérico, como el *router* indicando el tipo de fichero de arranque a utilizar. En una petición DHCPDISCOVER, éste campo se pone a nulo. El servidor devuelve un nombre de ruta de directorio cualificada completa en una petición DHCPOFFER. El valor termina en X'00'.

4.1.5 El proceso DHCP

DHCP automatiza la distribución y asignación de direcciones IP, además configura la información acerca de las estaciones de trabajo en la red. Sin DHCP sería necesario ir a cada estación de trabajo y manualmente configurarla con una única dirección IP acompañada de la información de la configuración. Para automatizar la configuración de una estación de trabajo, DHCP sigue los siguientes procesos:

a. Una estación de trabajo hace una solicitud acerca de la información de la configuración. Para solicitar la información DHCP, la estación de trabajo envía un paquete para que cualquier servidor DHCP escuche la solicitud.

El servidor DHCP responde con información acerca de la dirección IP. El servidor DHCP examina el paquete solicitante para determinar desde qué segmento de red fue enviada la solicitud. Si el servidor DHCP no contiene información acerca de la configuración de ese segmento de red no responderá la solicitud. En caso contrario, él contestará con la información de configuración solicitada del segmento de red que hizo la solicitud.

4.1.6 Agentes *RELAY*

Las solicitudes DHCP son difundidas, y los paquetes difundidos son dirigidos por enrutadores. Por consiguiente, si un servidor DHCP no reside físicamente en el mismo segmento desde el cual una solicitud DHCP es hecha, el servidor DHCP nunca responderá la llamada dirigidas al servidor por el enrutador que se encuentra dentro de los dos segmentos. Para hacer que un servidor DHCP responda a una solicitud DHCP desde un segmento remoto es necesario utilizar un agente *Relay*. Este es un *software* enrutador el cual dirige las solicitudes DHCP hacia el servidor DHCP y entonces retorna las respuestas a la estación de trabajo.

4.1.7 Localización de una nueva dirección de red

Esta sección describe la interacción cliente /servidor si el cliente no sabe su dirección de red. Asumir que el servidor DHCP tiene un bloque de direcciones de red de las que puede satisfactoriamente pedir para nuevas direcciones. Cada servidor mantiene también una base de datos de direcciones localizadas y alquiladas en almacenamiento local permanente.

1. El cliente emite un mensaje DHCPDISCOVER en su subred física local. El mensaje DHCPDISCOVER puede incluir algunas opciones como sugerencia de dirección de red o duración del alquiler, etc.

2. Cada servidor puede responder con un mensaje DHCPOFFER que incluye una dirección de red disponible y otras opciones de configuración.

3. El cliente recibe uno o más mensajes DHCPOFFER de uno o más servidores. El cliente escoge uno basado en los parámetros de configuración ofrecidos y emite un mensaje DHCPREQUEST que incluye la opción identificador de servidor para indicar que mensaje ha seleccionado.

4. Los servidores reciben la emisión de DHCPREQUEST de los clientes. Aquellos servidores que no haya seleccionado el mensaje DHCPREQUEST usan el mensaje como notificación de que el cliente ha rechazado esa oferta del servidor. El servidor seleccionado en el mensaje DHCPREQUEST almacena permanentemente el enlace con el cliente y responde con un mensaje DHCPACK que contiene los parámetros de configuración para la petición del cliente. La combinación del *hardware* del cliente y las direcciones de red asignadas constituye un identificador único para el alquiler del cliente y lo usan el cliente y el servidor para identificar un alquiler enviado en cualquier mensaje DHCP. El campo tu dirección IP en los mensajes DHCPACK se rellenan con las direcciones de red seleccionadas.

5. El cliente recibe el mensaje DHCPACK con parámetros de configuración. Realiza una comprobación final de los parámetros, por ejemplo con ARP para direcciones de red localizadas, y anota la duración del alquiler y el *cookie* de identificación de alquiler especificado en el mensaje DHCPACK. Llegados a este punto se configura el cliente. Detecta un problema con los parámetros en el mensaje DHCPACK, el cliente envía un mensaje DHCPDECLINE al servidor y reinicia el proceso de configuración. El cliente debería esperar un mínimo de diez segundos antes de reiniciar el proceso de configuración para evitar el tráfico excesivo de red en caso de bucle.

Si el cliente recibe un mensaje DHCPNAK, el cliente reinicia el proceso de configuración.

6. El cliente puede elegir abandonar su alquiler de una dirección de enviando un mensaje DHCPRELEASE al servidor. El cliente identifica el alquiler a liberar incluyendo su dirección de red y su dirección *hardware*.

4.1.8 Reutilización de una dirección de red localizada previamente

Si el cliente recuerda y desea volver a usar una dirección de red localizada previamente, entones se procesan los siguientes pasos:

1. El cliente emite un mensaje DHCPREQUEST en su subred local. El mensaje DHCPREQUEST incluye la dirección de red del cliente.

2. Los servidores con conocimiento de los parámetros de configuración del cliente responden al cliente con un mensaje DHCPACK.

3. El cliente recibe el mensaje DHCPACK con los parámetros de configuración. El cliente realiza una comprobación final de los parámetros y anota la duración del alquiler y el *cookie* de identificación del alquiler especificado en el mensaje DHCPACK. Llegados a este punto se configura el cliente.

Si el cliente detecta un problema en los parámetros en el mensaje DHCPACK, envía un mensaje DHCPDECLINE al servidor y reinicia el proceso de configuración pidiendo una nueva dirección de red. Si recibe un mensaje DHCPAK, no puede reutilizar su dirección de red memorizada. El cliente puede elegir abandonar su alquiler de una dirección de red enviando un mensaje DHCPRELEASE al servidor. El cliente identifica el alquiler que se libera con el *cookie* de identificación de alquiler.

4.2 DNS (*Domain Name System*)

4.2.1 Introducción

Los usuarios de las redes prefieren utilizar nombres pronunciables, más fáciles de recordar, en vez de la dirección IP de las máquinas conectadas a la red.

Inicialmente, en Internet el sistema de nombres escogido era una secuencia de caracteres arbitraria, administrada por el NIC *(Network Information Center)*, que comprobaba la no existencia de otra máquina con ese mismo nombre. Como el número de usuarios se incrementó demasiado, el tener una única autoridad de asignación de nombres no era nada práctico, debido al enorme trabajo administrativo que era mantenerla al día.

La solución hallada, que aún se utiliza, fue el descentralizar el mecanismo de asignación de nombres, delegando la autoridad, en parte del espacio de nombres y distribuyendo la responsabilidad de asignar la relación entre nombres y direcciones.

La definición de asignación entre nombres y direcciones debe estar definida orientada a la traducción eficiente y que garantice el control autónomo de la asignación de nombres. Por ejemplo:

nombre_local.nombre_general

Donde nombre_local sería el nombre administrado por una localización en concreto y nombre_general administrado por una autoridad general (nótese que ambos nombres están separados por un punto). Si aparece una nueva localización, la autoridad central incluirá su nombre en la lista de localizaciones válidas y le daría capacidad para administrar todos los grupos de nombres que antecedan al nombre de esa nueva localización (separada por puntos).

Los nombres se componen de combinaciones de los 26 caracteres anglosajones (A-Z y a-z), los dígitos (0-9) y el carácter "-". La longitud máxima de nombres de dominios o subdominios es de 63 caracteres y del nombre completo de 255 caracteres. Así, llegamos al punto de tener una estructura jerárquica, subdividiendo el espacio de nombres hasta que este sea manejable, esto es:

disc.eps.ua.es

Donde "disc" sería el Departamento de Ingeniería de Sistemas y Comunicaciones, de la Escuela Politécnica Superior de Alicante "eps" de la Universidad de Alicante "ua", de España "es". Podríamos caer en la falacia de que los nombres asignados están relacionados (necesariamente) con la topología de la red o la estructura de las interconexiones físicas.

El mecanismo que implementa una jerarquía de nombres de máquinas en las redes se llama sistema de dominio de nombres *(Domain Name System,* DNS; a partir de ahora utilizaremos las siglas anglosajonas para referirnos a este sistema).

El DNS especifica la sintaxis de los nombres, y las reglas para delegar autoridad sobre los nombres, además especifica la implementación de un sistema distribuido que relaciona eficientemente nombres con direcciones.

4.2.2 Sintaxis de nombres

La sintaxis de los nombres se compone de nombres de dominios separados por puntos. El nivel más bajo se sitúa a la izquierda (esto facilita comprimir mensajes con múltiples nombres de dominios).

En Internet, la máxima autoridad para asignar las direcciones IP y los DNS es el IANA *(Internet Assigned Numbers Authority)*, también es la encargada de delegar el segundo nivel de DNS a la organización IR *(Internet Registry)* o a registros regionales.

El sistema principal *(root)* no tiene nombre y es el que en un sólo fichero (*hosts.txt*) tiene los nombres de los *host* y sus direcciones.

El "*Top-level domain names" (*TLDs*,* nivel superior del dominio de nombres) se divide en los siguientes DNS:

Tabla VII. Nombres de dominio del nivel superior

NOMBRE	SIGNIFICADO
Com	Organizaciones comerciales, se van a establecer subdominios
Edu	Instituciones educativas (registro de 2 a 4 años)
Gov	Instituciones gubernativas de EE.UU.
Mil	Grupos militares de EE.UU.
Net	Principales centros de soporte de red (NICs,NOCs ...)
Org	Otras organizaciones (diferentes a las anteriores)
arpa	Dominio ARPANET temporal (obsoleto)
Int	Organizaciones internacionales
Código de país	Cada país (esquema geográfico)

En la tabla VIII, se ejemplifica los códigos de países, según la norma ISO-3166 :

Tabla VIII. Códigos de país

NOMBRE	SIGNIFICADO
Es	España
Uk	Inglaterra (*United Kingdom*)
De	Alemania (*Deuschland*)
Us	EE.UU. (*United States of América*)
Fr	Francia
Gt	Guatemala
…	…

Conceptualmente, se permiten dos tipos diferentes de jerarquías: geográfica y organizacional. Cada organismo solicita con qué tipo de esquema desea tener su nombre (en Internet el esquema geográfico es administrado por organismos generalmente públicos, en el caso particular de España, actualmente, este organismo sólo permite que utilicen el nombre de dominio "es" las empresas S.A. y S.L.).

Ejemplos de ambos tipos de jerarquías:

ozu.com ozu.es (dos empresas distintas, nacidas de la separación de Ozu)

La configuración de los *host* locales pasa por unas especificaciones del administrador principal, este le provee de:

- La definición de su zona de actuación.

- El fichero maestro de datos.

- Le actualiza el fichero maestro.

Y el sistema de dominio proporciona los métodos estándar de:

- Formatos de recursos de datos.

- Métodos de búsqueda en las BD.

- Métodos NS para actualizar los datos locales sobre servidores de nombres.

En algunos países, el segundo nivel de la jerarquía esta definida por categorías (AC, CO, GO, RE ...), en otras por políticas geográficas, por ejemplo en EE.UU. es de la forma:

nombre-entidad.localidad.estado.us

IBM.Armonk.NY.US

En EE.UU. existe un sub-dominio en el segundo nivel (aparte de los estados):

Tabla IX. Códigos de nombres de un sub dominio

NOMBRE	SIGNIFICADO
k12	Escuelas
Cc	Colegios
Tec	escuelas técnicas
State	agencias estatales de gobierno
Cog	Ayuntamientos
Lib	Bibliotecas
Mus	Museos

El IR se encarga de seleccionar y designar la administración diaria del DNS.

Destaquemos que usando sólo la sintaxis de dominio de nombres no se puede distinguir los nombres de sub-dominios de máquinas individuales.

4.2.3 Normas para nombrar DNS

El registro de un nuevo nombre no implica derechos de marca (C, R, TM, LTD...), y la responsabilidad es de cada uno al elegir su nombre, asegurándose que no es una TM (marca registrada).

IANA no se encarga de decidir que es o no un país, estado, etc., por ello la codificación de países la hace mediante la utilización del ISO 3166.

4.2.4 Dominio de nombres para correo electrónico

Los mensajes de correo son de la forma:

nombre_usuario@nombre_parte_de_dominio

por ejemplo:

a00444@eps.ua.es

Donde a00444 sería el nombre del usuario (en este caso, algún alumno de la promoción 92-97 de la carrera Ingeniería Informática impartida en la Universidad de Alicante), este nombre es configurado por el administrador de la subred, la red local o incluso por un usuario de una máquina conectada directamente a Internet. @ nos indicaría que es una dirección de correo (el sistema de correo utiliza el DNS MX). Y eps.ua.es sería el nombre del dominio (Escuela E.P.S.A. de la Universidad de Alicante, España).

4.2.5 Resolución de nombres en direcciones

El esquema de dominio de nombres incluye un sistema eficiente, seguro, de propósito general y distribuido para relacionar nombres con direcciones. Este sistema está compuesto por una serie de sistemas independientes, pero cooperativos denominados servidores de nombres; cada uno de ellos es un programa que funciona en un servidor y que soporta traducciones de direcciones IP a nombres y viceversa.

El programa cliente, denominado resolutor de nombres, necesitara usar uno o más servidores al traducir un nombre.

Existen dos tipos de peticiones de resolución:

- Recurrida: donde el servidor de nombres contará con otros servidores hasta hallar la respuesta a la petición y la enviará al remitente.

- Iterativa: en la que el servidor, en el caso de que no pueda resolver la dirección por sí mismo, mandará un mensaje al remitente diciéndole que no puede resolverla e indicándole la dirección del servidor de nombres al que debe dirigirse para hacerlo.

Para optimizar la búsqueda de nombres en servidores remotos, y para reducir el tráfico en la red, los servidores utilizan la técnica *caching*, que consiste en:

1.- Cuando se recibe respuesta de un servidor remoto de una petición de resolución, se le añade un tiempo de vida (*Time To Live*, TTL).

2.- Se mantiene durante un cierto tiempo en la memoria del servidor de nombres aquellas parejas nombre-dirección que hayan sido resueltas a petición de algún usuario de la red, junto con su tiempo de vida (TTL).

3.- Antes de enviar una petición a un servidor remoto, buscará en memoria si ya tiene esa dirección resuelta. Si existe en memoria, se la enviará al remitente informándole que pudiera no estar actualizada; enviará también la dirección al servidor de nombres remoto, por si le interesa garantizar la veracidad de la información.

4.2.6 La transmisión de mensajes

La transmisión se produce con octetos. Cada octeto numeraremos los *bits* de izquierda a derecha, empezando por 0, siendo éste el de mayor peso. En cuanto a los octetos, enviaremos en orden de significación. También podemos hacerlos en ASCII con paridad cero. Se utiliza TCP/IP sobre el puerto 53 (decimal). Retransmisión después de 2-5 segundos.

4.2.7 Formato del mensaje de dominio de nombres

Este mensaje es usado por la aplicación, que debe comunicarse con una máquina y necesita resolver el nombre (que le ha introducido el usuario) para hallar la dirección equivalente, la máquina lo mandará a un servidor de nombres local y este le contestará con otro mensaje (menor de 512 caracteres):

Tabla X. Formato del mensaje de dominio de nombres

IDENTIFICACIÓN (16 *bits*)	PARÁMETRO (16 *bits*)
NÚMERO DE PREGUNTA	NÚMERO DE RESPUESTAS
NÚMERO DE AUTORIDAD	NÚMERO DE AÑADIDOS
SECCIÓN DE PREGUNTA ... SECCIÓN DE RESPUESTA ... SECCIÓN DE AUTORIDAD ... SECCIÓN DE AÑADIDOS ...	

4.2.7.1 Identificación

Usado por el remitente para comparar respuestas y preguntas.

4.2.7.2 Parámetro

Especifica la operación pedida y el código de respuesta (ordenados los *bits* de izquierda a derecha):

Tabla XI. Especificación de los códigos del parámetro

Bit		**SIGNIFICADO**
0	Operación:	0 Pregunta 1 Respuesta
1-4	Tipo de Pregunta:	0 Estándar 1 Inversa 2 Obsoleta (Terminación 1) 3 Obsoleta (Terminación 2)
5		1 Pregunta de autoridad
6		1 Mensaje Truncado
7		1 Se desea recursión
8		1 Recursión disponible
9-11		Reservado

12-15	Tipo de respuesta:	0 Sin error 1 error de formato en pregunta 2 Fallo de servidor 3 Nombre no existe

4.2.7.3 Número de pregunta

Número de entradas en la sección pregunta.

4.2.7.4 Número de respuestas

Número de entradas en la sección respuestas.

4.2.7.5 Número de autoridad

Número de entradas en la sección autoridad.

4.2.7.6 Número de añadidos

Número de entradas en la sección añadidos.

4.2.7.7 Sección de pregunta

Preguntas sobre las que se solicita respuesta. En la tabla XII se especifica el formato de cada pregunta.

Tabla XII. Formato de cada pregunta

DOMINIO DE NOMBRES DE LA PREGUNTA (32 *bits*)	
TIPO PREGUNTA (16 *bits*)	CLASE PREGUNTA (16 *bits*)

4.2.7.7.1 Dominio de nombres de la pregunta

Contiene el nombre solicitado. El primer octeto indica la longitud de cada etiqueta (en octetos). La ultima etiqueta es de longitud 0 para indicar el fin del nombre.

4.2.7.7.2 Tipo pregunta

¿Es una máquina?, ¿es una dirección de correo?...

4.2.7.7.3 Clase pregunta

Nos permite usar este mensaje para otras direcciones que no sean Internet.

4.2.7.8 Sección de respuesta

El servidor responderá a cada pregunta de la sección anterior, con una respuesta en esta sección. El formato de las secciones respuesta, autoridad y añadidos se muestra en la Tabla XIII.

Tabla XIII. Formato de las secciones de respuesta

PETICIÓN DOMINIO DE NOMBRE (32 *bits*)	
TIPO (16 *bits*)	CLASE (16 *bits*)
TTL	LONGITUD PETICIÓN
DATOS PETICION …	

4.2.7.8.1 Petición dominio de nombre

Nombre propio (del nodo que pide la resolución).

4.2.7.8.2 Tipo

El formato del tipo de petición disponible, se describe en la Tabla XIV:

Tabla XIV. Formato del tipo de petición disponible

Tipo	Valor	Significado y contenido
A	1	Dirección de *Host*: Dirección IP 32 *bits*.
NS	2	Servidor de nombres autorizado para el dominio.
MD	3	Obsoleto (destino de correo).
MF	4	Obsoleto (fuente de correo).
CNAME	5	Nombre canónico de un dominio.
SOA	6	Inicio de autoridad: Especifica que parte de la jerarquía de nombres está implementada por un servidor de nombres.
MB	7	Experimental: MDN (*Mailbox domain name*).
MG	8	Experimental: Miembro de grupo de correo.
MR	9	Experimental: Renombre del MDN.
NULL	10	Experimental: Nulo.
WKS	11	Descripción de servicio conocido bueno.
PTR	12	Nombre del dominio como puntero.
HINFO	13	Nombre de la CPU y del S.O.
MINFO	14	Información de un buzón o lista de correo.
MX	15	16 *bits* prioritarios y nombre del *host* que actúa como central de correo para ese dominio.
TXT	16	Texto ar*bit*rario: cadena ASCII sin interpretación.
AAAA	28	Dirección de *Host*: Dirección IPV6 128 *bits*.
AXFR	252	Petición de transferencia de una zona.
MAILB	253	Petición de campos de correo (MB,MG,MR)
MAILA	254	Obsoleto: Petición resolución de correo.
*	255	Petición de todos los registros.

Los más utilizados son A y MX.

A continuación se describen los tipos de campos de recursos más usados.

4.2.7.8.2.1 SOA (Inicio de autoridad)

Cada zona contiene un SOA al inicio del archivo de zona. Almacena información específica de la zona, dicha información es:

4.2.7.8.2.1.1 Servidor de autoridad

Indica el servidor DNS primario que tiene autoridad sobre la zona.

4.2.7.8.2.1.2 Persona responsable

El correo electrónico de la persona responsable de la zona. Usualmente se utiliza un punto(.) en lugar del símbolo ``arroba"(@).

4.2.7.8.2.1.3 Número de serie

Usado para indicar la copia de la zona más actual. Cuando el servidor secundario contacta al servidor primario para ver si es necesario actualizar su zona, verifica su propio número de serie con el DNS primario y si es menor se realiza una actualización de la zona en el servidor DNS secundario. El formato a usarse es AAAAMMDDVV para el año (AAAA), mes (MM), día (DD) y Versión (VV).

4.2.7.8.2.1.4 Refrescar

Representa el tiempo (en segundos) en el que el servidor DNS secundario debe esperar para conectarse al servidor DNS primario y verificar si es necesario realizar una transferencia de la zona.

4.2.7.8.2.1.5 Reintento

Indica cuánto tiempo (en segundos) el servidor DNS secundario debe esperar a que el servidor DNS primario le conteste antes de realizar un reintento.

4.2.7.8.2.1.6 Expiración

Si no se puede transferir una zona a un servidor DNS secundario en este lapso (en segundos) se considera que la información del DNS secundario es incorrecta o no actualizada.

4.2.7.8.2.1.7 TTL mínimo

Es el Tiempo de vida (en segundos) aplicado a todos los campos de registro donde no fue especificado. Este tiempo indica los segundos en que un servidor de nombres externo puede mantener almacenado en su caché después de que se obtuvo una respuesta del DNS primario.

4.2.7.8.2.2 NS (Servidor de nombres)

Contiene los servidores primario y secundario con autoridad sobre la zona o zonas delegadas. Cada zona debe contener al menos un campo NS.

4.2.7.8.2.3 A (Dirección IP)

Mantiene una dirección IP específica para un *host*.

4.2.7.8.2.4 MR (Registro de recursos con el nombre del buzón cambiado)

Especifica un nombre de buzón del dominio en buzón con otro nombre, el cambio de nombre apropiado de un buzón existente especificado en el campo propietario. Se utiliza a menudo un registro de recursos MR como entrada de reenvío para un usuario que se ha trasladado a un buzón diferente. Los registros MR no provocan procesamiento de secciones adicionales.

4.2.7.8.2.5 MX (Intercambio de Correo)

Especifica el *host* que recibirá el correo del dominio en el que se encuentra, considerado como el servidor de correo. Existe un número llamado prioridad o número de preferencia. El correo será tratado de ser enviado primero al que contiene el número menor (mayor preferencia) y si no hay respuesta se tratará con el que tiene el siguiente número menor. Los números son asignados arbitrariamente.

4.2.7.8.2.6 CNAME (Nombre Canónico)

Define un alias o nombre sinónimo de un *host*, el cual debe tener un campo de tipo A.

4.2.7.8.3 Clase

En la Tabla XV se define el formato de la clase de petición del dominio de nombres.

Tabla XV. Formato de la clase de petición del dominio de nombres

TIPO	VALOR	SIGNIFICADO Y CONTENIDO
IN	1	Internet.
CS	2	Obsoleta: CSNET.
CH	3	CHAOS.
HS	4	Hesiod.
	255	Ninguna clase.

4.2.7.8.4 TTL

Tiempo que debe mantenerse en la memoria del servidor de nombres local, número entero positivo del tipo de con signo de 32 *bits*.

4.2.7.8.5 Longitud de datos recurso

Número de octetos en la sección datos recurso, 16 *bits integer*.

4.2.7.8.6 Datos recurso

Aquí se haya la respuesta a la pregunta solicitada.

5. GUÍA

5.1 Introducción

El protocolo de IPV6 fue diseñado con importantes características que permitirán tener más y mejores redes superando las limitaciones del protocolo IPV4 usado actualmente.

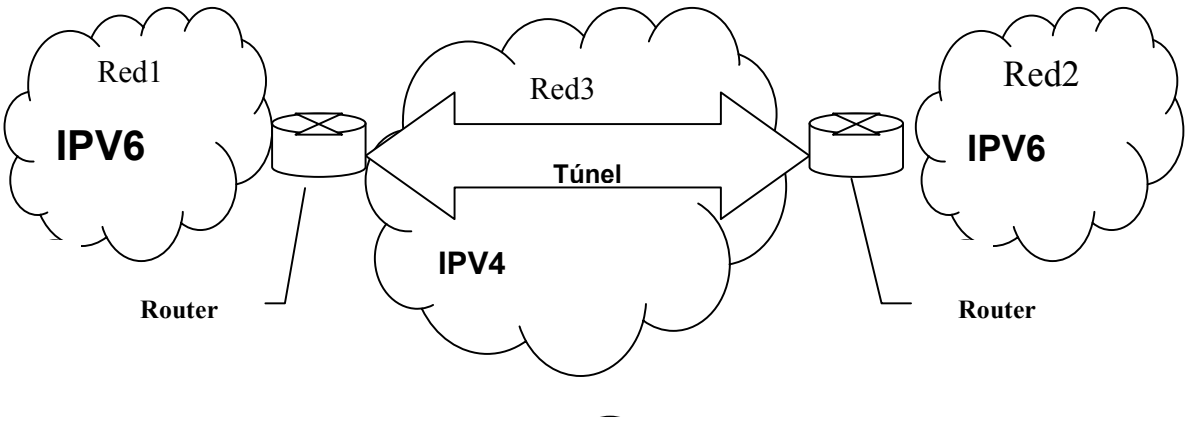
Entre las características más importantes destacan las de espacio de direcciones prácticamente infinito, autoconfiguración de computadoras y ruteadores, computación móvil, mayor soporte para seguridad, herramientas de calidad de servicio, manejo de tráfico multimedia en tiempo real, aplicaciones *multicast* y mecanismos para la transición gradual de IPV4 a IPV6; características que harán posible, por ejemplo, la coexistencia de la telefonía, las comunicaciones móviles e inalámbricas y los medios audiovisuales en redes más grandes, eficientes y seguras.

A pesar de que las espectativas son muy promisorias, pasar de un protocolo a otro es bastante complicado, no sólo el costo que llevaría transformar toda una red a otra si se hace radicalmente el cambio, sino también que el tiempo que conlleva cada prueba que se realiza para que el protocolo IPV4 soporte al nuevo protocolo IPV6.

Actualmente, IPV6 *Backbone (6Bone)* es una red mundial usada para probar los conceptos e implementaciones del IPV6. Esta red esta compuesta por islas que soportan IPV6, unidas por enlaces punto a punto llamados túneles.

En éste capítulo se hará un análisis del proceso de transición de los protocolos IPV4 e IPV6 utilizando los servicios DNS y DHCP.

Figura 17. Estructura del Internet con los protocolos IPV4 e IPV6



En la figura 17 se observa un pequeño ejemplo de la estructura de Internet utilizando los dos protocolos *(*IPV4 – IPV6*)*. Los grupos de redes como por ejemplo red1 y red2 se les denominan **islas** ya que en comparación a la utilización del protocolo IPV4, es poca la gente que tiene montada una red con el protocolo IPV6.

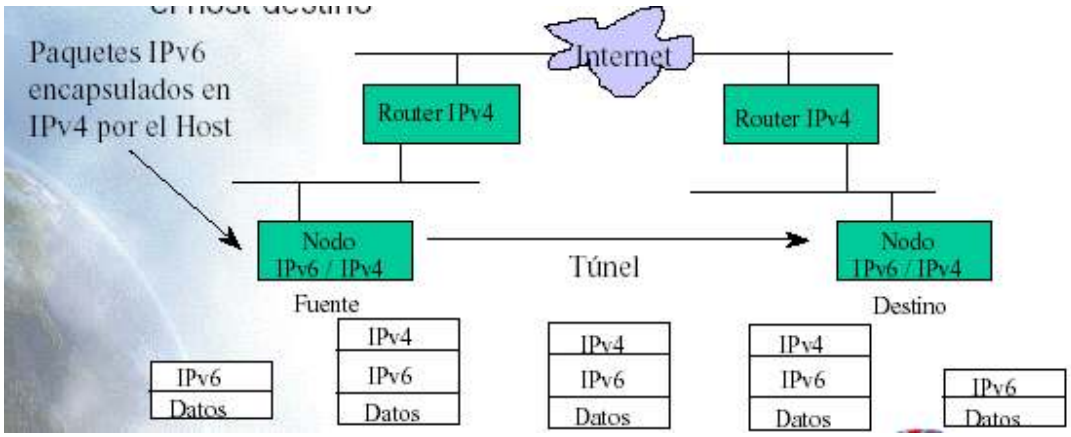
5.2 Analizando el traslado utilizando el servicio DNS

Si usted posee un esquema tradicional basado en IPV4 y desea que su DNS funcione con el protocolo IPV6, tome en cuenta lo siguiente:

1. No habría forma de comunicación entre una Red1 que utiliza el protocolo de IPV6 y una Red3 que utiliza el protocolo IPV4 (como se observa en la figura 17), porque el tamaño de las direcciones es diferente, es decir, el protocolo IPV6 utiliza direcciones de 128 *bits* en cuanto al protocolo de IPV4 utiliza direcciones de 32 *bits*. Por ejemplo, si la Red1 se intenta comunicar con la Red3, el programa de la Red1 le proporcionará un nombre de dominio de IPV6, y el DNS le devolverá si es que llegase a encontrar una dirección IP de 128 *bits*. Lo que se debe de hacer es actualizar el *software* tanto para los *host* como para los *routers* que detecten la transición simple del Internet, es decir:

   i. *Simple Internet Transitions (*SIT*)*:
      a. *Stack* IP doble en *host*s y *routers*.
      b. Tablas dobles IPV4 e IPV6

Figura 18. ***Simple Internet Transitions*** **(SIT)**



Fuente: IPV6 la siguiente Generación (IPng), Foro IPV6

2. Para tener direcciones IPV6 compatible con IPV4 se tiene que emplear lo siguiente:

   i. Un prefijo nulo antecediéndole a la dirección IPV4

      a. Como notación abreviada se emplea ::a.b.c.d

   ii. Total interconectividad con los *routers* IPV4, pero no se crea una nueva tabla de enrutado, por lo que no se aprovecha las características de direccionamiento jerarquizado.

   iii. También se emplean los túneles automáticos.

3. Una vez que el *software* del establecimiento de una red esté en su lugar, las aplicaciones pueden comenzar a emplear las características IPV6 y sus alcances para la comunicación con otros nodos IPV6, si se asume que existe un camino directo vía conexiones capaces de soportar IPV6.

4. En el caso de las redes 1 y 2 sí existe comunicación directa, ya que ambas redes poseen el mismo protocolo *(*IPV6*)*. Para realizar la conexión de un extremo a otro se utiliza túneles montados sobre la Internet IPV4 actual (los datagramas IPV6 se encapsulan como carga en datagramas IPV4)

i. Túneles (IPV6 → IPV4 → IPV6) :

a. Configurados (configurados estáticamente, por lo general entre *routers*)

i. Uso de paquetes IPV6 en modo nativo.

ii. Paquetes encapsulados en un túnel IPV4 definido por una entrada manual en la configuración del *routers*.

b. Automáticos (configurados automáticamente)

i. La tabla de *routing* revela direcciones IPV4 compatibles con la IPV6 y encapsula los paquetes.

ii. La dirección IPV4 especifica el final del túnel, que ha de ser el *host* destino

5. El único requisito que debe cumplirse para poder conectarse a las redes que soportan protocolo IPV6 es disponer de un equipo con capacidad de ruteo IPV6 y que tenga asignada una dirección IPV4 estática.

6. Otra forma es por medio de traductores llamados NAT los cuales permiten la migración de un protocolo a otro utilizando asociación de direcciones tanto IPV4 como para IPV6. Existen 4 tipos de traductores categorizados de la siguiente manera:

i. Traductor A

Se utiliza en el primer tiempo de la transición para establecer una conexión de un ordenador principal IPV6 en una isla IPV6 a un ordenador principal IPV4 en el océano IPV4.

ii. Traductor B

Se utiliza en el primer tiempo de la transición para establecer una conexión de un ordenador principal IPV4 en el océano IPV4 a un ordenador principal IPV6 en una isla IPV6.

iii. Traductor C

Se utiliza en la última etapa de la transición para establecer una conexión de un ordenador principal IPV4 en una isla IPV4 a un ordenador principal IPV6 en el océano IPV6.

iv. Traductor D

Se utiliza en la última etapa de la transición para establecer una conexión de un ordenador principal IPV6 en el océano IPV6 a un ordenador principal IPV4 en una isla IPV4.

5.2.1 Ventajas de la transición

1. El *software* es simple de instalar y de entender.

2. No rompe el extremo para terminar el concepto de conectividad.

3. Utiliza las funciones IPV6 al comunicarse con otros nodos IPV6.

4. Un nodo dual de la pila puede comunicarse con IPV4 / IPV6, IPV4 solamente y solamente los nodos IPV6.

5.2.2 Desventajas de la transición

1. Carencia de escalabilidad. Utilizar pila dual a través de la red implica que todos los nodos requieren un direccionamiento IPV6 e IPV4. Esto no ayuda con las ediciones de la dirección pues el direccionamiento IPV6 y un direccionamiento único IPV4 todavía serán requeridos para cada nodo, pues los direccionamientos IPV4 llegan a ser más escasos este acercamiento no escalará simplemente.

2. Complejidad de la dirección. Es bastante compleja sin asociar dos direccionamientos del IP para un solo nodo y por lo tanto podría hacer la administración de este tipo de transición extremadamente difícil.

3. Por lo descrito en el inciso anterior, los direccionamientos IPV4 se afectan un aparato solamente cuando un nodo necesita comunicarse con un nodo IPV4 o viceversa. Por lo tanto cada nodo no requiere su propio *IP ADDRESS*.

4. Vectores de encaminamiento crecientes. Los *routers* también necesitarán utilizar los dos protocolos del IP adentro afecta con dos vectores de encaminamiento separados.

5. Ninguna ayuda para la comunicación entre solamente nodos IPV6 e IPV4.

5.3 El mecanismo que se emplea para la configuración de direcciones al servicio DNS

Actualmente las implementaciones de DNS disponibles corren sobre IPV4, y el sistema de DNS que soporta IPV6 está enlazado a la información del IPV4. No obstante existen algunas implementaciones DNS que comienzan a soportar IPV6 nativo. El DNS actual no es fácilmente extendible para soportar las direcciones de 128 *bits* de IPV6, ya que al hacer una consulta, las aplicaciones asumen que se les devolverá una dirección de 32 *bits* (IPV4). Para resolver esto, se definieron las siguientes extensiones:

- Un nuevo tipo de registro de recurso, el registro AAAA. Actualmente el registro AAAA se usa para almacenar una dirección IPV6 porque las extensiones están diseñadas para ser compatibles con implementaciones de DNS existentes.

- Un nuevo dominio para soportar búsquedas basadas en direcciones. Este dominio es IP6.INT.

- Redefinición de las consultas existentes que localizan direcciones IPV4, para que puedan también procesar direcciones IPV6.

Además para soportar las direcciones IPV6, la renumeración y el *multi-homing*, se incluyó un nuevo tipo de registro de recurso, A6 para almacenar las direcciones IPV6 de forma que se agilice la renumeración de la red.

5.3.1 *Software*

En la actualidad, los sistemas operativos se están modificando para agregarles los nuevos campos para soportar el nuevo protocolo IPV6. Para la configuración de nombres en el protocolo de IPV4 es necesario instalar el paquete BIND *(Berkeley Internet Name Daemon)* actualmente las versiones para este paquete se encuentra entre 9.x.

5.3.2 Configuración de BIND (*Berkeley Internet Name Daemon*)

Después de instalar el paquete *bind* se debe de crear el archivo *named.conf*, en el directorio donde se configuraran las zonas utilizando la opción *directory* para guardar los ficheros de configuración, como también con la opción *allo-query* el cual se utiliza para restringir el acceso a los dominios de los rangos de IP's.

Una edición del archivo de configuración, *named.conf*, también será necesaria para una zona de resolución inverso.

5.3.3 Resolución de nombres

Mapeo de nombres /IP apenas requeridas para incluir dos registros de tipo AAAA a un archivo de zona ya existente del dominio, junto con los direccionamientos de IPV4.

Ejemplo:

```
sp      IN      AAAA   2001:04A0:0:1::D
rj      IN      AAAA   2001:04A0:0:1::9
rs      IN      AAAA   2001:04A0:0:1::B
rn      IN      AAAA   2001:04A0:0:1::A
```

Donde sp, rj, rs, rn son registro que especifican los nombres de los ordenadores los cuales son del tipo AAAA y cada uno posee una dirección de la máquina correspondiente.

5.3.4 Resolución inversa

Para la resolución inversa de direcciones se debe editar un archivo de configuración named.conf e incluir una nueva zona inversa.

En este caso un bloque "2001:04A0:0001::/48" fue utilizado para ejemplificar la resolución inversa de direcciones. Así mismo, debe proceder de la siguiente forma la configuración:

Archivo: ***named.conf***

```
zone "1.0.0.0.0.a.4.0.1.0.0.2.IP6.int"
{       type master;
        file "2001:4a0:0001.IP6.int";
};
zone "1.0.0.0.0.a.4.0.1.0.0.2.IP6.arpa"
{       type master;
        file "2001:4a0:0001.IP6.arpa";
};
```

Definición de zona inversas

Pasamos, entonces, a responder el direccionamiento inverso del bloque que fue cedido. Tal prefijo ya debe haber sido delegado por el proveedor de direcciones IPV6, de la misma forma como ocurre con el direccionamiento de los subdominios.

Una forma de como se declara un registro inverso y semejante a IPV4, se hace comenzando por el último dígito hexadecimal de dirección, denominado *nibble format*.

Archivo: 2001:4a0:0001.IP6.int

```
$TTL 86400
@       IN      SOA     ns.pop-xx.rnp.br.    root.pop-xx.rnp.br. (
                                    2001112201;serial
                                    3H      ; refresh
                                    15M     ; retry
                                    1W      ; expiry
                                    1D )    ; minimum
        IN      NS      ns.pop-xx.rnp.br.

1.b.0.5.6.e.e.f.f.f.c.a.4.0.2.0.0.0.0.0  IN      PTR     x.pop-xx.rnp.br
b.1.e.c.2.6.e.f.f.f.9.2.6.0.2.0.0.0.0.0  IN      PTR     y.pop-xx.rnp.br
6.1.1.7.6.e.e.f.f.f.c.a.4.0.2.0.0.0.0.0  IN      PTR     z.pop.xx.rnp.br
```

Archivo: 2001:4a0:0001:IP6.arpa

```
$TTL 86400
@       IN      SOA     ns.pop-xx.rnp.br.    root.pop-xx.rnp.br. (
                                    2001112201;serial
                                    3H      ; refresh
                                    15M     ; retry
                                    1W      ; expiry
                                    1D )    ; minimum
        IN      NS      ns.pop-xx.rnp.br.

1.b.0.5.6.e.e.f.f.f.c.a.4.0.2.0.0.0.0.0.0  IN      PTR     x.pop-xx.rnp.br
b.1.e.c.2.6.e.f.f.f.9.2.6.0.2.0.0.0.0.0.0  IN      PTR     y.pop-xx.rnp.br
6.1.1.7.6.e.e.f.f.f.c.a.4.0.2.0.0.0.0.0.0  IN      PTR     z.pop.xx.rnp.br
```

5.3.5 Delegación de prefijos

Es una forma de cómo se delega bloques de direcciones usando un registro de tipo NS. Supongamos que atribuimos un bloque a un nuevo cliente que queramos escoger y sea apto para resolver direcciones para nombres.

Nuevo bloque: 2001:04A0:0001::/48

Bloque atribuido: 2001:04A0:0001:0100::/56

Así mismo, se tiene que adicionar como siguiente información un nuevo archivo **2001:4a0:0001.IP6.int**

$ORIGIN 0.0.1.0.1.0.0.0.0.A.4.0.1.0.0.2.IP6.int.

@	IN	NS	xpto.dominio.br.
	IN	NS	xyz.dominio.br.

El nuevo archivo **2001:4a0:0001.IP6.arpa**:

$ORIGIN 0.0.1.0.1.0.0.0.0.A.4.0.1.0.0.2.IP6.arpa.

@	IN	NS	xpto.dominio.br.
	IN	NS	xyz.dominio.br.

5.3.5 Nuevos registros

Los nuevos registros para el mantenimiento de las zonas de DNS serán introducidos los siguientes:

A6 – substituido por AAAA

DNAME – Delegación de resolución inversa

Un nuevo formato para una representación de registros inversos por ser también introducido, tomando un lugar de *nibble format*. Estamos hablando del uso de *Bitstring label.*

5.4 Analizando el servicio DHCPV6

Antes de detallar la transición del servicio DHCPV6, se da a conocer los siguientes conceptos que serán los utilizados para la manipulación del nuevo protocolo IPV6. El protocolo IPV6 utilizará la autoconfiguración.

5.4.1 ¿Qué es la autoconfiguración?

- Conjunto de pasos por los cuales un *host* decide como autoconfigurar sus interfaces en IPV6, también se puede decir que es un mecanismo que nos permite afirmar que IPV6 es *Plug & Play*.

- El proceso de la autoconfiguración incluye la creación de una dirección de enlace local, verificación de que no está duplicada en dicho enlace y determinación de la información que ha de ser autoconfigurada (direcciones y otra información).

Para la utilización del servicio DHCPV6 se introducen dos nuevos conceptos denominados *Stateful o* configuración predeterminada, el cual permite obtener direcciones de forma manual, y *Stateless* o descubrimiento automático, sin intervención, el cual permite la configuración automática.

El nuevo protocolo IPV6 define el proceso de generar una dirección de enlace local, direcciones globales y locales de sitio, mediante el procedimiento automático *(stateless)* y el mecanismo para detectar direcciones duplicadas.

La autoconfiguración *Stateless* (sin intervención), no requiere ninguna configuración manual del *host*, configuración mínima (o ninguna) de *routers*, y no precisa servidores adicionales. Permite a un *host* generar su propia dirección mediante una combinación de información disponible local e información anunciada por los *routers*.

Los *routers* anuncian los prefijos que identifican la subred (o subredes) asociadas con el enlace, mientras el *host* genera un identificador de interfaz, que identifica de forma única la interfaz en la subred. La dirección se compone por la combinación de ambos campos. En ausencia de *router*, el *host* sólo puede generar la dirección de enlace local, aunque esto es suficiente para permitir la comunicación entre nodos conectados al mismo enlace.

En la autoconfiguración *Stateful* (predeterminada), el *host* obtiene la dirección de la interfaz y/o la información y parámetros de configuración desde un servidor. Los servidores mantienen una base de datos con las direcciones que han sido asignadas a cada *host*.

5.4.2 Complementariedad en base a la autoconfiguración

Ambos tipos de autoconfiguración (*stateless y stateful*), se complementan. Un *host* puede usar autoconfiguración sin intervención (*stateless*), para generar su propia dirección , y obtener el resto de parámetros mediante autoconfiguración predeterminada (*stateful*).

El mecanismo de autoconfiguración sin intervención se emplea cuando no importa la dirección exacta que se asigna a un *host*, sino tan sólo asegurarse que es única y correctamente enrutable, al igual que IPV4.

El mecanismo de autoconfiguración predeterminada, por el contrario, nos asegura que cada *host* tiene una determinada dirección, asignada manualmente.

5.4.3 Proceso de caducidad de las direcciones

Cada dirección es cedida a una interfaz durante un tiempo predefinido (posiblemente infinito).

El tiempo de vida, indica durante cuanto tiempo esta vinculada dicha dirección a una determinada interfaz.

Cuando expira, la vinculación se invalida y la dirección puede ser reasignada a otra interfaz en cualquier punto de Internet.

Una dirección pasa a través de dos fases diferentes:

- Inicialmente, una dirección es *preferred* (preferida), lo que significa que su uso es arbitrario y no esta restringido.
- Posteriormente, la dirección es *deprecated* (desaprobada), en anticipación a que el vínculo con su interfaz actual va a ser anulado.

En estado “desaprobado”, su uso no es aconsejado, no prohibido. Cualquier nueva comunicación (una nueva comunicación TCP), debe usar una dirección preferida.

Una dirección desaprobada debería ser usada tan sólo por aquellas aplicaciones que ya la venían utilizando y a las que les es muy difícil cambiar a otra dirección sin interrupción del servicio.

5.4.4 Detección de direcciones duplicadas

Para asegurarse de que todas las direcciones configuradas son únicas, en un determinado enlace, los nodos ejecutan un algoritmo de detección de direcciones duplicadas, antes de asignarlas a una interfaz. Este algoritmo es ejecutado para todas las direcciones, independientemente de que hayan sido obtenidas mediante autoconfiguración *stateless o stateful.*

La autoconfiguración está diseñada para *host*s, no para *routers*, aunque ello no implica que parte de la configuración de los *routers* también puede ser realizada automáticamente (generación de direcciones de enlace local). Además, los *routers* también tienen que aprobar el algoritmo de detección de direcciones duplicadas.

5.4.5 Autoconfiguración *stateless* (DHCPV6)

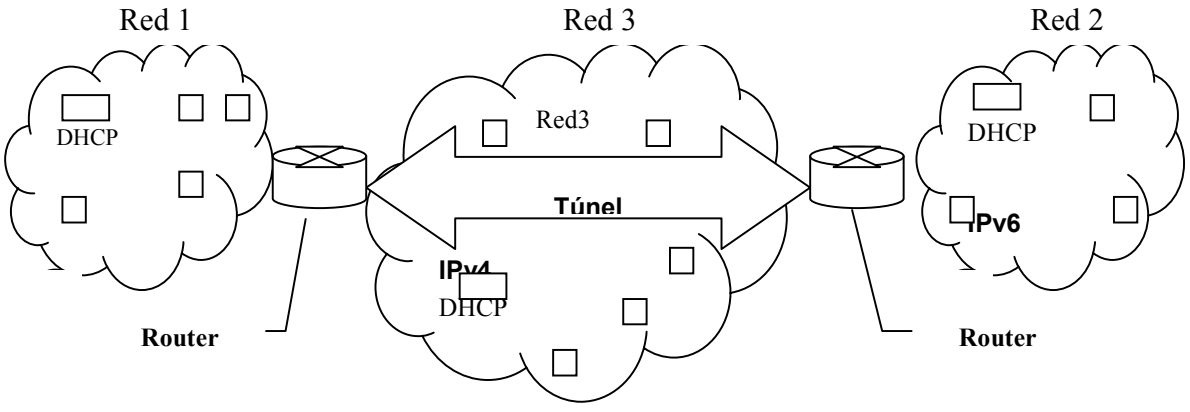
DHCPV6 es un protocolo UDP cliente/servidor, diseñado para reducir el costo de gestión de nodos IPV6 en entornos donde los administradores precisan un control sobre la asignación de los recursos de la red, superior al facilitarlos por el mecanismo de configuración *Stateless*.

Ambos mecanismos *(stateless y stateful)*, pueden usarse de forma concurrente para reducir el coste de propiedad y administración de la red.

Para lograr este objetivo, se centraliza la gestión de los recursos de la red, tales como direcciones IP, información de encaminado, información de instalación de sistemas operativos, información de servicios de directorios, sobre uno o varios servidores DHCP, en lugar de distribuir dicha información en ficheros de configuración locales en cada nodo.

Además, DHCP ha sido diseñado para ser fácilmente extensible con nuevos parámetros de configuración, a través de extensiones que incorporan esta nueva información.

Figura 19. **Estructura de redes con servidores DHCP**



5.5 Analizando el traslado utilizando el servicio DHCP

Si usted posee un esquema tradicional basado en IPV4 y desea que su DHCP funcione con el protocolo IPV6, tome en cuenta lo siguiente:

1. Para cada red se necesita poseer un servidor DHCP que pueda proporcionar direcciones a los *host* que pertenecen a dicha red. Como se observa en la figura 39, el servidor DHCP de la Red1 solamente puede proporcionar direcciones IP de 128 *bits*, al igual que el servidor DHCP que pertenece a la Red 3. En la Red 2, el servidor DHCP solamente podrá proporcionar direccionamiento IP de 32 *bits*.

2. Por lo descrito anteriormente, no existe forma alguna de comunicación entre una Red1 que utiliza el protocolo de IPV6 y una Red3 que utiliza el protocolo IPV4, porque tanto el tamaño de las direcciones es diferente, como también la configuración de cada servidor DHCP es distinta.

3. Si usted necesita tener comunicación con una red que posea el protocolo IPV6 lo que deberá de hacer es implementar una red que posea una estructura IPV6, es decir, utilizar un servidor DHCP que manipule direccionamiento IPV6, un servidor DNS que trabaje con direccionamiento IPV6.

4. Se debe de actualizar el *software* que es utilizado en los *routers* como en los *host*s. Una vez que el *software* del establecimiento de una red esté en su lugar, las aplicaciones pueden comenzar a emplear las características IPV6 y sus alcances para la comunicación con otros nodos IPV6.

5. Como sucede en el caso de las redes 1 y 2, donde sí existe comunicación directa, ya que ambas redes poseen el mismo protocolo *(*IPV6*),* para realizar la conexión de un extremo a otro se utilizan túneles montados sobre la Internet IPV4 actual.

6. Para la transición del servicio DHCP del protocolo IPV4 al protocolo IPV6 tampoco existe un proceso de transición gradual. Las implementaciones actuales de *software* se basan solamente en un direccionamiento IPV6 y no de ambas, esto se debe a que las direcciones IPV4 se están agotando que no existe mucho tiempo para una transición gradual.

5.5.1 Consideraciones para configuración del servicio DHCPV6

Para la configuración del servicio DHCPV6 se debe de tomar en cuenta los siguientes aspectos:

1. Evitar la configuración manual de dispositivos antes de su conexión a la red. Se requiere, en consecuencia, un mecanismo que permita a los *host* obtener o crear direcciones únicas para cada para cada una de sus interfaces, asumiendo que cada interfaz puede proporcionar un identificador único para sí misma (identificador de interfaz). En el caso más simple, el identificador de interfaz consiste en la dirección de la capa de enlace, de dicha interfaz. El identificador de interfaz puede ser combinado con un prefijo para formar la dirección.

2. Las pequeñas redes o sitios, con máquinas conectadas a un único enlace, no deberían requerir la presencia de un servidor *stateful* o *router,* como requisito para comunicarse. Para obtener, en este caso, características *Plug & Play*, empleamos las direcciones de enlace local, dado que tienen un prefijo perfectamente conocido que identifica el único enlace compartido, al que se conectan todos los nodos. Cada dispositivo forma su dirección de enlace local anteponiendo el prefijo de enlace local a su identificador de interfaz.

3. En el caso de redes o sitios grandes, con múltiples subredes y *routers*, tampoco se requiere la presencia de un servidor de configuración de direcciones *stateful*, ya que los *host* han de determinar, para generar sus direcciones globales o de enlace local, los prefijos que identifican las subredes a las que se conectan. Los *routers* mensajes periódicos de

anunciación, que incluyen opciones como listas de prefijos activos en los enlaces.

4. La configuración de direcciones debe facilitar la renumeración de los dispositivos de un sitio, por ejemplo, cuando se desea cambiar de proveedor de servicios. La renumeración se logra al permitir que una misma interfaz pueda tener varias direcciones, que recibe en préstamo. El tiempo del préstamo es el mecanismo por el que se renuevan las direcciones, al expirar los plazos para las viejas, sin que se conceda una prórroga. Al poder disponer de varias direcciones simultáneamente, permite que la transición no se distorsione, permitiendo que ambas, la vieja y la nueva dirección den continuidad a la comunicación durante el período de transición.

5. Sólo es posible utilizar este mecanismo en enlaces capaces de funciones *multicast*, y comienza, por tanto, cuando es iniciada o activada una interfaz que permite *multicast*.

6. Los administradores de sistemas necesitan la habilidad de especificar que mecanismos (*stateless o stateful*, o ambos) deben ser usados. Los mensajes de anunciación de los *routers* incluyen indicadores para esta función.

5.5.2 Asignación de direcciones al servicio DHCPV6

1. Una vez se activa la interfaz:

   - Se genera la dirección tentativa de enlace local, como se ha descrito antes.

   - Verificar que dicha dirección tentativa puede ser asignada (no esta duplicada en el mismo enlace).

   - Si está duplicada, la autoconfiguración se detiene, y requiere un procedimiento manual (por ejemplo, usando otro identificador de interfaz).

   - Si no está duplicada, la conectividad a nivel de IP se ha logrado, al asignarse definitivamente dicha dirección tentativa a la interfaz en cuestión.

   - Si se trata de un *host*, se interroga a los posibles *routers* para indicar al *host* lo que debe de hacer a continuación.

   - Si no hay *routers*, se invoca el procedimiento de autoconfiguración *stateful*.

   - Si hay *routers*, éstos contestarán indicando fundamentalmente, como obtener las direcciones si se ha de utilizar el mecanismo de *stateful*, u otra información, como tiempos de vida, etc.

2. Hay algunos detractores de este mecanismo, ya que implica que cualquier nodo puede ser identificado en una determinada red si se conoce su identificador IEEE (dirección MAC).

5.5.3 Nuevas funciones con DHCPV6

Entre las nuevas implementación que se le han hecho al servicio DHCP se encuentran las siguientes:

1. Configuración de actualizaciones dinámicas de DNS.
2. Desaprobación de direcciones, para renumeración dinámica.
3. Relés preconfigurados con direcciones de servidores, o mediante *multicast*.
4. Autenticación.
5. Los clientes pueden pedir múltiples direcciones IP.
6. Las direcciones pueden ser reclamadas mediante el mensaje de iniciar_reconfiguración.
7. Integración entre autoconfiguración de direcciones *stateless* y *statefull*.
8. Permitir *relés* para localizar servidores fuera del enlace.

5.6 Ventajas de la transición

1. La dirección de enlace local permite a un nodo tener una dirección tan pronto como arranca, lo que significa que todos los clientes tienen una dirección IP fuente para localizar un servidor o *relé* en su mismo enlace.

2. Los indicadores de compatibilidad BOOTP y *broadcast* han desaparecido.

3. El *multicast* y los ámbitos de direccionamiento permiten el diseño de paquetes de descubrimiento, que definen por si mismos su rango por la dirección *Multicast*, para la función requerida.

4. La autoconfiguración *stateful* ha de coexistir e integrarse con la *stateless*, soportando la detección de direcciones duplicadas y los dos tiempos de vida de IPV6, para facilitar la renumeración automática de direcciones y su gestión.

5. Se soportan múltiples direcciones por cada interfaz.

6. Algunas opciones DHCPV4 ya no son precisas, debido a que los parámetros de configuración se obtienen a través de ND o del protocolo de localización de servicios.

CONCLUSIONES

1. A pesar de que las ventajas del nuevo protocolo IPV6 son superiores al protocolo actual, pasar de un protocolo necesitará la inversión tanto en recurso humano como en el económico.

2. Para una transición completa del protocolo IPV4 al nuevo protocolo IPV6, se llevará poco tiempo, ya que existe mucho dinero invertido en el actual protocolo tanto en *software* como en *hardware*. Eso significaría pérdidas muy elevadas para la mayoría de empresas que poseen redes montadas sobre el protocolo IPV4.

3. Para que al protocolo IPV4 pueda comunicarse con el protocolo IPV6, se modificó la estructura del datagrama para que soportará paquetes de IPV6, y de esa manera hacer la comunicación de extremo a extremo utilizando túneles sobre el protocolo IPV4.

4. La transición al protocolo IPV6 será laborioso debido a la dificultad en modificar de los servicios actuales del DNS y DHCP para que soporten el protocolo de IPV6, ya que dicha transición lleva implícito la actualización de *software* tanto de los *host*s como de los *routers* y de esta manera tener comunicación de una red con direccionamiento IPV4 con otra red con direccionamiento IPV6.

5. Con la utilización del nuevo protocolo IPV6, se podrá asignar más de una dirección a cada usuario y con ello evitar las limitaciones de transferencia de información de una red a otra.

RECOMENDACIONES

1. Es necesario crear proyectos orientados a la difusión y formación a la transición del protocolo IPV6 para tener una base de conocimiento, de sus ventajas y desventajas.

2. Integrar grupos de trabajo con otras instituciones para poder montar una plataforma de pruebas adecuada a la transición del protocolo IPV4 al protocolo IPV6.

3. Obtener espacio de direcciones suficiente que permita asignar espacio IPV6 a las instituciones, involucradas en el proyecto de transición al protocolo IPV6.

4. Incrementar las actividades de investigación sobre el protocolo IPV6, con el fin de obtener un claro entendimiento del mismo.

5. El estudio de los proyectos pilotos para la transición del protocolo IPV6 deberá de ser a pequeña escala, por ejemplo, tomando como base una o dos instituciones con las cuales se tendrá la comunicación de direccionamiento IPV6.

BIBLIOGRAFÍA

Redes. Topología. http://vgg.sci.uma.es/redes/topo.html#sring. (Febrero 2002).

Redes. Servicios de red. http://vgg.sci.uma.es/redes/servicio.html. (Marzo 2002).

Redes. Protocolo de red. http://vgg.sci.uma.es/redes/red.html. (Marzo 2002).

Redes. Medio físico. http://vgg.sci.uma.es/redes/fisico.html. (Marzo 2002).

Redes. Equipos de red. http://vgg.sci.uma.es/redes/equipos.html. (Marzo 2002).

Redes informáticas. http://nti.educa.rcanaria.es/conocernos_mejor/apuntes/paginas/redes.htm. (Marzo 2002).

Protocolos de red. http://www.nowsmia.4t.com/proto.htm. (Marzo 2002).

Protocolos de comunicación en red. http://www.cs.buap.mx/~jpalacio/sod/dos/node13.html. (Marzo 2002).

Nuevos estándares de protocolos. http://members.tripod.com/a_pizano/impresion/Cap4.htm. (Abril 2002).

IPV6, el futuro de Internet. http://www.undersec.com. (Septiembre 2002).

DHCP. http://atenea.udistrial.edu.co/estudiantes/moreno/dhcp.html. (Agosto 2002).

Los campos de recursos (RR, *Resource Records)*. http://fismat.umich.mx/~emurguia/Mipagina/tesis/node59.html. (Agosto 2002).

Tutorial y descripción técnica de TCP/IP. http://ditec.um.es/laso/docs/tut-tcpip/3376c45.html. (Julio 2002).

Este micro-COMO trata sobre como instalar un DNS para la red local. http://teleline.terra.es/personal/garzones/dns-micro-como.html#1. (Octubre 2002).

Boletín bimestral sobre tecnología de redes, producido y publicado en RNP – *Rede Nacional de Encino e Pesquisa, Piloto de serviço IPv6: procedimento de configuração de DNS*. http://www.rnp.br/newsgen/0205/ipv6_dns.shtml. (Mayo 2002).

COMO de IPV6 y conexión al 6bone.
http:// www.sindominio.net/pipermail/madridwireless/2001-November/000193.html. (Octubre 2002).

6bone práctica de ruteo. http://www.linux.org.ar/info/articulos/JKoumian/JKoumian-1.html. (Octubre 2002).

"6bone Práctica de Ruteo"

http://www.linux.org.ar/info/articulos/JKoumian/JKoumian-1.html