UNIVERSIDAD NACIONAL AUTONOMA DE MÉXICO

ESCUELA NACIONAL DE ESTUDIOS PROFESIONALES
CAMPUS ARAGON

34

PROTOCOLOS DEL NIVEL DE APLICACIÓN DEL MODELO TCP/IP

T E S I S

QUE PARA OBTENER EL TITULO DE
INGENIERO MECANICO ELECTRICISTA

P R E S E N T A:

ALEJANDRO MARTINEZ ARAOZ

ASESOR: ING. DAVID B. ESTOPIER BERMÚDEZ

MÉXICO **2000.**

Agradezco a mis maestros, por compartir sus experiencias y conocimientos, con el fin de formar grandes personas.

Al Ing. David B. Estopier, por su tiempo y por cultivar en mi la vocación por esta profesión.

A mis dos hermanos, Pablo y Jéssica, por todo el apoyo, por cuidar siempre de mi, por los buenos consejos y por toda una vida juntos.

A ti Zamara, por ser parte de mi y por todos los momentos que hemos compartido. Tarde o temprano teníamos que encontrarnos.

A mis padres, por toda la paciencia y todo el apoyo que una persona puede recibir. Gracias por el amor que me han dado y que siempre guardaré en mi alma.

PROTOCOLOS DEL NIVEL DE APLICACION

DEL MODELO TCP/IP

INDICE

OBJETIVOS..... 1

INTRODUCCION..... 2

CAPITULO I
GENERALIDADES..... 7

1 1 Redes de datos.
1.1.1 Concepto y necesidad de una red de datos..... 8
1.1.2 Clasificación de las redes..... 9
1.2 Protocolos de comunicación.
1.2.1 Concepto de protocolos..... 14
1.2.2 Protocolos de bajo y alto nivel..... 14
1.3 Protocolo TCP/IP.
1.3.1 Evolución y objetivos del TCP/IP..... 16
1.3.2 Flexibilidad de TCP/IP..... 19
1.4 Arquitectura de red.
1.4.1 Concepto de arquitectura de red..... 20
1.5 Modelo de referencia OSI.
1.5.1 Niveles del modelo OSI..... 22
1.6 Modelo de referencia DoD
1.6.1 Niveles del modelo DoD..... 25
1.7 Modelo de referencia TCP/IP.
1.7.1 Niveles del modelo TCP/IP..... 27
1.7.2 Operación de las capas TCP/IP..... 29

CAPITULO II
PROTOCOLOS DEL NIVEL INTERNET Y TRANSPORTE DEL MODELO TCP/IP 31

2.1 Protocolos de la capa Internet
2.1.1 Protocolo Internet (IP) 33
2.1.1.1 Encabezado del Datagrama IP 35
2.1.1.2 Direcciones de red 38
2.1.2 Protocolo Internet de Control de Mensajes (ICMP) 40
2.1.2.1 Reporte de errores 42
2.1.2.2 Entrega de mensajes ICMP 43
2.1.2.3 Formato de mensajes ICMP 44
2.2 Protocolos de la capa de transporte.
2.2.1 Protocolo TCP 47
2.2.1.1 Formato del segmento TCP 48
2.2.1.2 Seguimiento de un mensaje 50
2.2.2 Protocolo UDP 52
2.2.2.1 Formato de los mensajes UDP 53

CAPITULO III
ACCESO REMOTO 55

3.1 Protocolo Telnet.
3.1.1 Descripción de Telnet 57
3.1.2 Terminal virtual de Red (NVT) 60
3.1.3 Acceso a Telnet 64
3.1.4. Puertos y sockets en Telnet 65
3.1.5 Conexión Telnet 69
3.1.6 Modo de comando Telnet 70

CAPITULO IV
TRANSFERENCIA DE ARCHIVOS 73

4.1 Protocolo de Transferencia de archivos (FTP).
4.1.1 Características del FTP 76
4.1.2 Puertos FTP 78
4.1.3 Conexión FTP 79
4.1.4 Comandos FTP 80
4.1.5 FTP Anónimo 85

4.2 Protocolo Trivial de Transferencia de archivos (TFTP).
4.2.1 Características de TFTP ... 87
4.2.2 Funcionamiento de TFTP ... 88
4.3 Sistema de Archivo de Red (NFS).
4.3.1 Descripción de NFS ... 91
4.3.2 Llamada de Procedimiento Remoto (RCP). ... 93
4.3.3 Representación de Datos Externo (XDR) ... 95
4.3.4 Funcionamiento y características de NFS ... 96

CAPITULO V
CORREO ELECTRONICO Y ADMINISTRACION DE REDES ... 99

5.1 Protocolo Simple de Transferencia de Correo (SMTP)
5.1.1 Modelo del correo electrónico. ... 101
5.1.2 Comandos y respuestas del SMTP ... 104
5.1.3 Extensiones Multipropósito de Correo en Internet (MIME) ... 109

5.2 Protocolo Simple de Administración de Red (SNMP).
5.2.1 Arquitectura y conceptos del SNMP ... 112
5.2.2 Detección de problemas con SNMP ... 114
5.2.3 Mensajes SNMP ... 116
5.2.4 Base de Información sobre la Administración (MIB) ... 118
5.2.5 Estructura de la Información sobre la Administración (SMI) ... 120

CONCLUSIONES ... 124

GLOSARIO ... 126

BIBLIOGRAFIA. ... 132

OBJETIVO

Describir la estructura de capas en la que se organiza la familia de protocolos TCP/IP (Transfer Control Protocol / Internet Protocol), para después realizar un análisis de los protocolos de nivel de aplicación más sobresalientes, de acuerdo al tipo de servicio que ofrecen al usuario: acceso remoto, transferencia y acceso de archivos, correo electrónico y administración de red.

No es objetivo de este trabajo explicar todos los protocolos que conforman la familia TCP/IP, ni explicar las operaciones que suceden en las capas inferiores, dedicadas al transporte de la información a través de los medios físicos. Se describe sólo una parte del conjunto TCP/IP, con la explicación de los principales protocolos de aplicación, al ser éstos los que comienzan y terminan los procesos en donde el usuario se comunica con la red y viceversa. Esta tesis pretende ser una guía que permita a estudiantes, administradores de sistemas y a cualquier persona interesada en aprender los mecanismos de las redes de ordenadores, tener una visión clara y resumida del funcionamiento de los protocolos más comunes que pertenecen al nivel de aplicación.

INTRODUCCION

Durante los últimos años, ha evolucionado una nueva tecnología que hace posible interconectar muchas redes físicas diferentes y hacerlas funcionar como una unidad coordinada. Esta tecnología, llamada *internetworking*, unifica diferentes tecnologías de hardware subyacentes al proporcionar un conjunto de normas de comunicación entre redes heterogéneas. La tecnología de las redes, oculta los detalles de hardware de red y permite que las computadoras se comuniquen independientemente de sus conexiones físicas.

Los protocolos son reglas acerca de la forma en que se comunican entre sí las computadoras y los dispositivos que las interconectan, y pueden incluir regulaciones concretas que recomienden u obliguen a aplicar una técnica o convenio determinado.

Se han desarrollado protocolos que permiten el intercambio de datos y transmisión de información entre diferentes redes de computadoras a través de una amplia gama de modelos hasta ahora aceptados. Actualmente, el grupo de protocolos Internet TCP/IP (Transfer Control Protocol / Internet Protocol), son los de mayor uso a nivel mundial. Este conjunto de protocolos (cuyas siglas provienen de sus dos principales estándares), puede utilizarse para establecer la comunicación a través de cualquier grupo de redes conectadas entre si. Por ejemplo, algunas empresas utilizan el TCP/IP para interconectar todas las redes dentro su corporación, aún cuando estas no tengan una conexión hacia redes externas. Otros grupos utilizan el TCP/IP para comunicarse entre sitios geográficamente alejados uno del otro.

TCP/IP demuestra su viabilidad a gran escala y forma la tecnología base para una red global que conecta hogares, escuelas, corporaciones, y laboratorios alrededor del mundo: la red Internet.

La tecnología de Internet ha sido diseñada para permitir la comunicación entre máquinas que tengan arquitecturas diferentes, para poder utilizar cualquier hardware de red y para incorporar muchos sistemas operativos.

Debido a la fuerte dependencia del trabajo en red, se ha producido una explosión de servicios que los protocolos TCP/IP pueden proporcionar. Los sistemas en red pueden hacer uso de éstos servicios para la transferencia de archivos, la entrega de correos, administración de recursos, ejecución remota y obtención de datos con nuevas aplicaciones de hipertexto.

Esto no quiere decir que todos los sistemas que utilizan TCP/IP dispongan del conjunto completo de protocolos. En la mayoría de los casos existe una lista normalizada de servicios básicos proporcionados por el software de trabajo. Las aplicaciones adicionales para el manejo del protocolo también varían, en función del sistema operativo sobre el que se ejecuta el TCP/IP.

Desde el punto de vista de un usuario, una red TCP/IP aparece como un grupo de programas de aplicación que utilizan la red para llevar a cabo tareas útiles de comunicación. Se utiliza el término interoperabilidad para describir la habilidad que tienen diversos sistemas de computación para cooperar en la resolución de problemas en redes informáticas. Los programas de aplicación de Internet muestran un alto grado de interoperabilidad. La mayoría de los usuarios que accesan a Internet lo hacen al correr programas de aplicación sin entender la tecnología TCP/IP; los usuarios confían en estos programas y en el software subyacente de la red para manejar esos detalles.

En este trabajo se analizarán los protocolos de aplicación más sobresalientes que nos ofrece la familia TCP/IP, para lo cual el primer capítulo comenzará con algunos conceptos básicos, incluyendo también el estudio de los modelos de referencia OSI (Open Systems Interconnection), DoD (Departament of Defense) y TCP/IP. Estos modelos permiten la unificación y estandarización de las diferentes tecnologías.

En el segundo capítulo se describen los protocolos del nivel Internet y de Transporte de acuerdo al modelo TCP/IP, que son los siguientes: protocolo IP (Internet Protocol), ICMP (Internet Control Message Protocol), TCP (Transfer Control Protocol) y UDP (User Datagram Protocol).

Estos dos niveles son los que anteceden al nivel de aplicación, en donde se encuentran los protocolos que permiten servicios como el acceso remoto, la transferencia y acceso de archivos, el correo electrónico y la administración de redes. Estos servicios de aplicación de Internet son los más populares y su descripción es la siguiente:

- *Acceso remoto.* Este servicio permite que el usuario que esté frente a una computadora se conecte a una máquina remota y establezca una sesión interactiva. Cuando el usuario establece la conexión, aparece una ventana en su pantalla, desde donde es posible obtener información o alterar la configuración de la terminal remota. Cuando termina la sesión, la aplicación regresa al usuario a su sistema local. Este servicio lo proporciona el protocolo TELNET (Telecommunications Network).

- *Transferencia de archivos.* Aunque los usuarios algunas veces transfieren archivos por medio del correo electrónico, el correo está diseñado principalmente para mensajes cortos de texto. Los protocolos TCP/IP incluyen un programa de aplicación para

transferencia de archivos, el cual permite que los usuarios envíen o reciban archivos arbitrariamente grandes de programas o de datos. Como el correo, la transferencia de archivos a través de una red de redes TCP/IP es confiable debido a que las dos máquinas comprendidas se comunican de manera directa, sin tener que confiar en estaciones intermedias para hacer copias del archivo a lo largo del camino. Los protocolos FTP (File Transfer Protocol) y TFTP (Trivial Transfer Protocol), son los encargados de ofrecer esta aplicación al usuario.

- *Acceso de Archivos.* Este servicio proporciona un acceso de archivos compartidos en línea que es transparente e integrado; muchas de las redes de ordenadores utilizan este servicio permitiendo que se creen enlaces a directorios de otras computadoras en sus propias terminales y usen estos archivos remotos como si fueran locales. El protocolo relacionado a este servicio es el NFS (Network File Systems).

- *Correo electrónico.* El correo electrónico permite que un usuario componga mensajes y los envíe a individuos o grupos. Otra parte de la aplicación de correo permite que un usuario lea los mensajes que ha recibido. Aunque existen muchos sistemas de correo electrónico, al utilizar el TCP/IP se logra que la entrega sea más confiable debido a que no se basa en computadoras intermedias para distribuir los mensajes de correo: Un sistema de entrega de correo TCP/IP opera al hacer que la máquina del transmisor contacte directamente la máquina del receptor. Por lo tanto, el transmisor sabe que, una vez que el mensaje salga de su máquina local, se habrá recibido de manera exitosa en el sitio de destino. El protocolo SMTP (Simple Mail Transfer Protocol) es el responsable de esta aplicación.

- *Administración de redes* Este servicio permite que una estación de administración controle y reciba mensajes (alarmas) del estado de los componentes que conforman una red. Esta aplicación es controlada por el protocolo SNMP (Simple Network Management Protocol).

Estos servicios se explicarán en los últimos capítulos de esta tesis, buscando así, un análisis ordenado de los protocolos de aplicación siguiendo el modelo TCP/IP.

CAPITULO I

GENERALIDADES

Las redes de computadoras representan actualmente una necesidad para el intercambio de datos entre corporaciones, dando lugar a una gran variedad de tipos de redes para las que existen diferentes tecnologías que anteriormente ocasionaban una incompatibilidad operativa al interconectarse dos redes diferentes entre ellas.

Una solución a este problema fue el desarrollo de modelos de referencia como el OSI (Open Systems Interconnection) o el TCP/IP, los cuales buscan la estandarización de los servicios físicos y de aplicación para lograr una interconexión universal entre redes.

Estos modelos de referencia, proponen el uso de protocolos para regular cada proceso involucrado en la comunicación de dos o más computadoras. El modelo TCP/IP, es el más utilizado a nivel mundial debido a su fácil operación, y a la manera en que hace que los procesos de comunicación sean invisibles para el usuario.

1.1 REDES DE DATOS.

1.1.1 CONCEPTO Y NECESIDAD DE UNA RED DE DATOS.

Una red se considera una colección de dispositivos conectados en ambiente común con el fin de compartir información entre el conjunto de máquinas que constituyen la red.

Existió una época en que comunicarse con una máquina era muy difícil para el operador o más aún para el usuario común. Entre más sistemas aparecían con un mayor número de terminales, más datos eran requeridos por más usuarios para la toma de decisiones gerenciales y para incrementar la producción diaria. Con el tiempo, todo esto hizo que el sistema no tuviera la capacidad para soportar las terminales que eran agregadas a él.

Por tal motivo, las redes de computadoras se desarrollan a partir de la necesidad de optimizar el uso de los recursos y equipo periférico o para compartir los programas almacenados en una máquina principal que distribuye los servicios para los que estos programas fueron diseñados, entre las computadoras o estaciones de trabajo conectadas a la red y que lo soliciten Esta computadora se conoce como servidor de archivos o de aplicaciones.

Dado el gran desarrollo que han tenido las computadoras y las redes, cada vez son más las computadoras que se interconectan para satisfacer las necesidades de compartir información de una organización, sin importar la localización geográfica donde se encuentren los usuarios o los recursos.

Cabe señalar que el intercambio de información no es solamente de documentos, con frecuencia cada vez mayor, se envían o reciben imágenes digitalizadas, voz, datos de

todos tipos para aplicaciones diversas, y todo esto se hace utilizando la infraestructura de comunicaciones y redes que se han desarrollado en los últimos años enormemente.

Las organizaciones modernas suelen estar bastante dispersas, y a veces incluyen empresas distribuidas en varios puntos de un país o extendida por todo el mundo. Muchas terminales situadas en distintos lugares y es necesario que los datos e información estén al alcance de todos los miembros de la organización. La sociedad de nuestros días emplea la información para reducir costos de producción de los bienes que consumimos, y en general para mejorar nuestra calidad de vida.

Es así como el concepto de red al momento de surgir dio respuesta a las necesidades de los usuarios en cuanto a la disponibilidad de la información y la optimización de recursos, esto mediante la compartición de los mismos.

1.1.2 CLASIFICACION DE LAS REDES.

Los primeros sistemas de comunicaciones utilizaban conexiones permanentes entre cada par de terminales. Un arreglo más económico y flexible de interconexión es facilitar la comunicación temporal entre dos estaciones que deseen comunicarse entre sí. A este proceso se le llama conmutación.

Este método elimina la necesidad de comunicación por alambrado directo entre todos los pares de estaciones de red. De esta manera, las redes se clasifican de acuerdo en su servicio en:

- Redes No Conmutadas.
- Redes Conmutadas.

REDES NO CONMUTADAS.

También conocidas como redes permanentes, son aquellas cuyo enlace es de uso exclusivo de los usuarios, por lo que ellos tienen el control permanente del mismo. Este servicio permite interconectar equipos terminales de datos y los equipos terminales de comunicaciones en distintos puntos. En una red no conmutada los caminos de la información se mantienen constantes.

REDES CONMUTADAS.

Son conexiones físicas que se establecen de manera dinámica, a solicitud y necesidad del usuario. La conexión permanece mientras dura el enlace, y se realiza mediante el intercambio de señales entre el usuario y la red. Por la forma en que se realizan los servicios de conmutación pueden ser:

- Servicios orientados a conexión.
- Servicios no orientados a conexión.

- ***Servicios orientados a conexión***: En términos generales, requieren que exista una conexión previa a la transferencia de datos. En dicha conexión se intercambian las direcciones fuente y destino, estableciéndose así las tablas de enrutamiento en los nodos dentro de la red y definiendo el circuito virtual. También es necesario asignar un número de referencia a dicha transferencia con el fin de identificar toda la información relativa a la misma transacción y hacerla viajar por la misma ruta.

- ***Servicios no orientados a conexión***: No requieren del establecimiento previo de una conexión o definición de un circuito virtual debido a que toda la información contiene

siempre datos acerca de la dirección fuente destino. Un paquete por lo general contiene sólo unos cuantos cientos de octetos y transporta información de identificación que permite al hardware de la red saber cómo enviar el paquete hacia un destino especifico. Por ejemplo, un archivo grande que será transmitido entre dos máquinas debe ser fragmentado en muchos paquetes que serán enviados a través de la red en un momento dado El hardware de la red envía los paquetes al destino especificado donde el software los reensamblará de nuevo en un sólo archivo

Los motivos para adoptar la conmutación de paquetes son el costo y el desempeño. Dado que múltiples máquinas pueden compartir el hardware de red, se requiere de pocas conexiones y el costo se reduce.

Siempre que se vayan a transmitir datos de una red con muchas ubicaciones de terminales se deberá emplear un arreglo que permita que las distintas terminales se comuniquen entre sí. Es por esta razón que es necesaria una clasificación de las redes, de acuerdo a su servicio o a su cobertura, en donde tenemos la siguiente división:

- Redes de Area Local LAN (Local Area Network).
- Redes de Area Amplia WAN (Wide Area Network).

REDES DE AREA LOCAL (LAN).

Las redes de área local cubren un área geográfica limitada (como un edificio o un pequeño campus), donde todo nodo de la red puede comunicarse con todos los demás. Por lo general, las LAN tienen varias características particulares que se mencionan a continuación:

a) Un campo de acción cuyo tamaño puede variar de 2 a 22 kilómetros aproximadamente.

b) Los canales emplean líneas de muy alta velocidad.

c) La red pertenece a una sola organización.

d) Debido a que la tecnología LAN cubre distancias cortas, ofrecen tiempo de retraso mínimo (milisegundos).

e) En la tecnología LAN, cada computadora por lo general contiene un dispositivo de interfaz de red que conecta la máquina directamente con el medio de la red.

REDES DE AREA AMPLIA (WAN).

La tecnología WAN proporciona comunicación que cubre grandes distancias, por ejemplo, una WAN puede recorrer un continente o unir computadoras a través de un océano. Por lo común las WAN son más lentas que las LAN y tienen tiempos de retraso mucho mayores entre las conexiones.

En la tecnología WAN, una red por lo común consiste en una serie de computadoras complejas, llamadas computadoras de paquetes, interconectadas por líneas de comunicación y módems. El tamaño de la red puede extenderse si se le añade un nuevo conmutador y otras líneas de comunicación.

Este tipo de redes poseen las siguientes características:

a) Los canales suelen proporcionarlos las compañías telefónicas, con un determinado costo mensual si las líneas son alquiladas, y un costo proporcional si son líneas normales conmutadas.

b) En general, los ETD (Estaciones Transmisoras de Datos) y los ECD (Equipos de Conmutación de Datos) están separados por distancias que varían desde algunos kilómetros hasta cientos de kilómetros.

c) Las líneas son relativamente propensas a errores (si se utilizan circuitos telefónicos convencionales).

Algunos conceptos que se manejan dentro de una red de telecomunicaciones LAN o WAN son los siguientes:

Host (Anfitrión).- Es una computadora que ejecuta aplicaciones y tiene uno o más usuarios. Un host puede funcionar como extremo en una comunicación.

Servidor.- Se trata de un software instalado en una computadora, que le permite ofrecer un servicio a otra computadora llamada local. El computador local contacta con el servidor remoto gracias a otro software llamado cliente. Los servidores administran los recursos de la red y proporcionan aplicaciones útiles a los usuarios.

Tarjeta de red.- Es una interfaz que permite conectar a un host con la red. Envía señales sobre un medio físico de red y recibe las señales entrantes.

Hub (Concentrador).- Dispositivo al que se conectan varias computadoras. Simula una red que interconecta a las computadoras.

Puente.- Dispositivo que conecta dos o más segmentos físicos de una LAN y retransmite las tramas cuyas direcciones de origen y destino se encuentran en distintos segmentos.

Ruteador.- Es un dispositivo que encamina datos por una red. También se les conoce como gateways (Pasarelas), y conectan a dos o más redes enviando información de una red a otra.

1.2 PROTOCOLOS DE COMUNICACION.

1.2.1 CONCEPTO DE PROTOCOLOS.

Un protocolo es un conjunto de reglas que gobiernan el formato y el significado de las tramas, paquetes o mensajes que son intercambiados por las entidades corresponsales dentro de una capa. Las entidades utilizan protocolos para realizar sus definiciones de servicio, teniendo libertad para cambiar el protocolo, pero asegurándose de no modificar el servicio visible a los usuarios, es por eso importante saber elegir el protocolo adecuado para interconectar redes de acuerdo al sistema operativo que se tenga.

En cierto sentido, los protocolos son para las comunicaciones lo que los algoritmos para la computación. Un algoritmo permite especificar o entender un cómputo aunque no se conozcan los detalles de un juego de instrucciones de CPU. De manera similar, un protocolo de comunicaciones permite especificar o entender la comunicación de datos sin depender de un conocimiento detallado de una marca en particular de hardware de red.

Los sistemas complejos de comunicación de datos no utilizan un solo protocolo para manejar todas las tareas de transmisión, sino que requieren de un conjunto de protocolos cooperativos, a veces llamados familia de protocolos o conjunto de protocolos.

1.2.2 PROTOCOLOS DE BAJO Y ALTO NIVEL.

Cada interfaz de una red se responsabiliza de llevar a cabo el protocolo de acceso al medio que controla las comunicaciones a través del medio, el protocolo del enlace que regula una comunicación entre interfaces y el protocolo de acceso a la red que especifica y

supervisa las interacciones entre una interfaz y su usuario. Estos protocolos, son llamados protocolos de bajo nivel.

Además, y encima de los protocolos de bajo nivel, existe otro conjunto de protocolos, llamados de alto nivel, quienes definen y supervisan una comunicación entre usuarios (o sus procesos). Tienen significado límite a límite, es decir, se aplican a la comunicación entre usuarios (o sea, entre los puntos " finales " de la comunicación). En la figura 1 1 se muestra el área de operación de los protocolos de bajo y alto nivel.

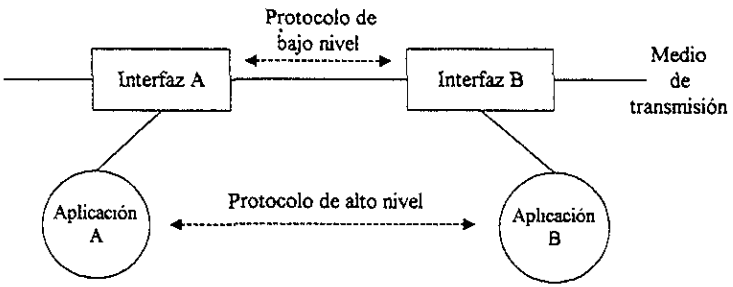

Figura1.1. Nivel de operación de los protocolos de alto y bajo nivel.

Los protocolos pueden describir detalles de bajo nivel de las interfaces de máquina a máquina (por ejemplo, el orden en que los bits de un octeto se envían a través de un cable) o del intercambio entre programas de aplicación (por ejemplo, la forma en que un programa transfiere un archivo a través de una red). La mayor parte de los protocolos incluye descripciones intuitivas de las interacciones esperadas así como especificaciones

más formales. Por lo tanto, es importante elegir el protocolo para interconectar redes de acuerdo al sistema operativo que se tenga en la red. Entre los protocolos más completos y utilizados por diferentes empresas se encuentran: Ethernet, Token Ring, Frame Relay, y TCP/IP.

1.3 PROTOCOLO TCP/IP.

1.3.1 EVOLUCION Y OBJETIVOS DEL TCP/IP.

La familia de protocolos TCP/IP surge a mediados de los 70's para poder establecer comunicación entre equipos independientemente de la tecnología y arquitectura de la red.

En 1971 el ejército de los Estados Unidos, a través de DARPA (Defense Advanced Research Project Agency), empieza la búsqueda de nuevas tecnologías para interconectar redes. DARPA contaba con una red de paquetes conmutada llamada ARPANET. Esta red estaba formada inicialmente por cuatro nodos de conmutación llamados Procesadores de Mensajes Internet o IMP's (Internet Messages Processors). Estos nodos se localizaban en la Universidad de California en Los Angeles, la Universidad de California en Santa Bárbara, el Instituto de investigación de Stanford y la Universidad de Utha.

Inicialmente ARPANET utilizaba el protocolo de comunicación "1822", llamado así porque ese era el número del documento técnico que describía el sistema, pero a medida que se desarrollaban las funciones que desempeñaba el sistema, como la necesidad de transferir archivos de una máquina a otra, así como la capacidad de aceptar registros de

entrada remotos, el protocolo que se usaba ya no era capaz de manejar estas funciones, por lo que se desarrolló uno nuevo que ya implementaba nuevas funciones. El protocolo desarrollado fue el NCP (Network Control Program), pero aparecía también una nueva aplicación que fue el correo electrónico. De esta manera en 1973 el NCP ya no era capaz de manejar el volumen de tráfico y la nueva funcionalidad propuesta.

Así es como se inició un nuevo proyecto, con el objeto de desarrollar un nuevo protocolo; en 1974 se propone por primera vez el TCP/IP, que se describía como un sistema que incluía un protocolo de aplicación estandarizado, que también utilizaba confirmaciones de extremo a extremo, pero lo más importante era que se sugería que el nuevo protocolo fuera independiente de la red y del hardware de computación. También se proponía la conectividad universal a través de la red. Estas dos ideas permitían que cualquier tipo de plataforma participara en la red.

En 1982 el TCP/IP sustituye al NPC como protocolo dominante en la creciente red. A partir del constante desarrollo de ARPANET, DARPA decide dividir su red en dos: la ARPANET y MILNET. ARPANET se ocupaba de los fines científicos y MILNET se dedicaba a fines militares; ambas redes contaban con la misma arquitectura, sólo que en MILNET se empleaba mayor seguridad.

En 1983 DARPA decide difundir el TCP/IP mediante un convenio que realiza con la BSD (Berkley Software Distribution) de la Universidad de Berkley, la cual era el centro de desarrollo UNIX. Así es como BSD emite una versión de UNIX que incorporaba a TCP/IP como elemento integral, quedando esta versión disponible para el mundo como software de dominio público.

El TCP/IP se difundió a través de UNIX, porque precisamente en esa época este sistema era el más usual en las escuelas de informática y de computación.

Dado el creciente interés de las universidades y centros de investigación, la NSF (National Science Foundation) se da cuenta de la importancia de las redes de datos para las aplicaciones científicas nacionales y en 1985 aprueba el desarrollo de un amplio acceso para supercomputadoras y crea la NSFNET, conectando sus seis centros nacionales de supercómputo.

En 1986 la NSF difunde el desarrollo de su red otorgando dinero para la integración de las universidades nacionales e internacionales, además de apoyar a países externos a los Estados Unidos, financiando parte de los gastos de conexión; de esta manera, la NSFNET empieza a formar la espina dorsal de lo que se llama ahora Internet y debido a este desarrollo, en 1990, DARPA declara ARPANET obsoleta y la desmantela, quedando así Internet, que ha logrado un crecimiento exponencial, en gran parte gracias al desarrollo de la familia de protocolos TCP/IP.

Desde sus inicios TCP/IP ha tenido los siguientes objetivos:

1) Independencia de la tecnología de conexión a bajo nivel y de la arquitectura de la computadora.
2) Conectividad universal a través de la red.
3) Reconocimiento de extremo a extremo.
4) Protocolos de aplicación estandarizados.

1.3.2 FLEXIBILIDAD DE TCP/IP.

La característica más significativa del TCP/IP es la modularidad del conjunto de protocolos, ya que para interconectarse se puede utilizar cable de fibra óptica con Ethernet, o cable par trenzado con la interfaz de datos distribuidos, o pueden utilizarse módems como protocolo punto a punto. Esta independencia significa que puede elegirse el mejor método para cada parte de la red, y es posible casi cualquier combinación.

TCP/IP existe para la mayoría de los sistemas operativos, aunque presenta algunas limitaciones en algunas plataformas. TCP/IP puede coexistir en el mismo medio con otros protocolos de red y pueden encapsular otros protocolos o ser encapsulado.

Con el gran desarrollo de TCP/IP, la disparidad entre equipos está disminuyendo, ya que permite que redes de diferentes arquitecuras puedan conectarse entre sí. Esto hace que muchos de los usuarios puedan tener acceso a una gran cantidad de servicios, haciendo de esta manera, un uso eficiente del equipo mediante la óptima administración de los recursos de la red. La interconexión a todos los servicios de la red debe ser transparente para el usuario, de manera que simule como si se estuviera dentro de la misma, independientemente de los dispositivos que se encuentren en la misma habitación, dispersos en un edificio o separados por muchos kilómetros mediante el uso de líneas telefónicas dedicadas, microondas o sistemas satelitales.

Una realidad es la conexión con Internet, ya que puede dar más visibilidad a una gran organización y permitir el intercambio de correo electrónico con cerca de 25 millones de personas y proporcionar acceso a los inmensos recursos de datos de la propia Internet.

1.4 ARQUITECTURA DE RED.

1.4.1 CONCEPTO DE ARQUITECTURA DE RED.

La mayoría de las redes se organiza en una serie de capas o niveles, con el objeto de reducir la complejidad de su diseño. Cada una de ellas se construye sobre su predecesora. El número de capas, el nombre, el contenido y función de cada una varía de una red a otra. Sin embargo, en cualquier red, el propósito de cada capa es ofrecer ciertos servicios a las capas superiores, liberándolas del conocimiento detallado sobre cómo se realizan dichos servicios.

La capa "n" en una máquina conversa con la capa "n" de otra máquina. Las reglas y convenciones utilizadas en esta conversación se le conoce como protocolo de la capa "n", como se muestra en la figura 1.2, para el caso de una red de 5 capas.

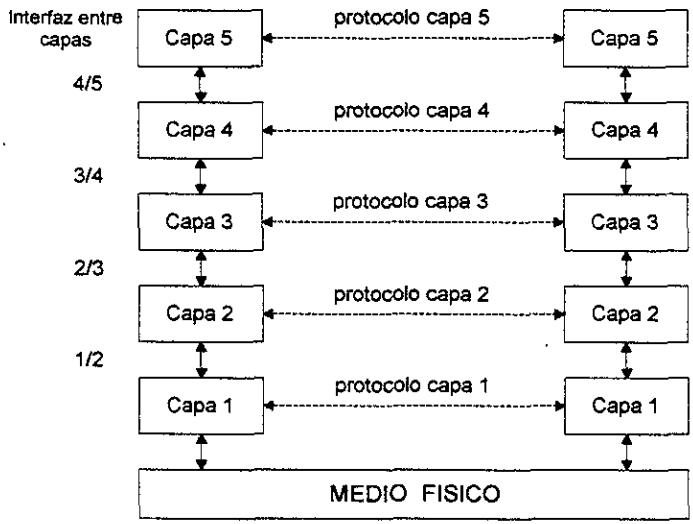

Figura 1 2. Arquitectura de una red de 5 capas.

En realidad no existe una transferencia directa de datos desde la capa "n" de una máquina hasta al capa "n" de otra; más bien, cada capa pasa la información de datos y control a la capa inmediatamente inferior, y así sucesivamente hasta que alcanza la capa localizada en la parte más baja de la estructura. Debajo de la capa 1 está el medio físico, a través del cual se realiza la comunicación real.

Entre cada par de capas adyacentes hay una interfaz, la cual define los servicios y operaciones que la capa inferior ofrece a la superior. Una de las consideraciones más importantes a tomar cuando los diseñadores de redes deciden el número de capas por incluir en una red, así como lo que cada una de ellas deberá hacer, consiste en definir claramente la interfaz entre capas, ya que el diseño claro de una interfaz, además de minimizar la cantidad de información que debe pasar entre capas, hace más simple la sustitución de la realización de una capa por otra diferente. De esta manera, todo lo que se necesita de la nueva capa, es que ofrezca exactamente el mismo conjunto de servicios a la capa superior, tal como lo hacía la anterior.

Al conjunto de capas y protocolos se le denomina arquitectura de red. Las especificaciones de esta, deberán contener la información suficiente que permita al diseñador escribir un programa o construir el hardware correspondiente a cada capa, y siga la forma correcta del protocolo apropiado.

El concepto de arquitectura de red es la base que se utiliza en los diferentes modelos de referencia para la interconexión de sistemas abiertos, es decir, sistemas que pueden interconectarse con diferentes tecnologías. Por ejemplo, los modelos de referencia OSI, DoD y TCP/IP, que se estudiarán a continuación.

1.5 MODELO DE REFERENCIA OSI.

1.5.1 NIVELES DEL MODELO OSI.

La Organización Internacional de Normas (ISO), como primer paso hacia la normalización internacional de protocolos, desarrolló lo que se conoce como Modelo de Referencia OSI (Open Systems Interconnection), el cual se refiere a la conexión de sistemas heterogéneos. El modelo OSI pone atención al intercambio de información entre sistemas y no al funcionamiento interno de cada sistema en particular, en otras palabras, el modelo de referencia OSI constituye el marco de trabajo para el desarrollo de protocolos estándares para la comunicación entre dos o más equipos. El objetivo a largo plazo de OSI es desarrollar una compatibilidad total entre sistemas y productos ofrecidos por distintos proveedores de redes al rededor del mundo. En la figura 1.2 se ilustran las 7 capas del modelo OSI.

CAPA	FUNCION
7	Aplicación
6	Presentación
5	Sesión
4	Transporte
3	Red
2	Enlace
1	Física

Fig. 1.3. Modelo de referencia OSI.

CAPA DE APLICACIÓN.

Este nivel interactúa con las aplicaciones de los usuarios y con los sistemas operativos locales, ya que consiste en una serie de programas de aplicación que se utilizan en la red, y es aquí en donde se definen las reglas para entrar a los sistemas de comunicaciones

CAPA DE PRESENTACION.

Es esta capa tenemos la presentación, la conversión de códigos, terminales virtuales, transferencia de archivos, codificación de la información y traducción. Su tarea es definir la forma en que los datos son presentados, de tal manera que puedan ser intercambiados entre sistemas con diferentes representaciones internas. Un servicio que ofrece es suministrar la sintaxis para la conversión de datos.

CAPA DE SESION.

Es la capa encargada de la organización del diálogo, sincronización y administración de las entidades que se comunican, es decir, se ocupa de la comunicación de las aplicaciones, ya que realiza la función lógica para el intercambio ordenado de datos (por ejemplo, el reinicio de una sesión después de una interrupción) y es la primera capa que ofrece servicios orientados al usuario, entre los cuales tenemos, los servicios para el establecimiento de una conexión (login remoto).

CAPA DE TRANSPORTE.

Este nivel maneja los ruteos entre nodos múltiples y rutas alternas, transferencia de datos nodo a nodo así como la detección y corrección de errores punto a punto. Una

funcion primordial de esta capa es que ofrece la multiplexación que permite realizar varias conexiones de la capa o nivel de transporte.

CAPA DE RED.

Es aquí donde se lleva el control de la red, la coordinación entre los nodos adyacentes, el ruteo, manejo/conmutación de paquetes y la administración de las conexiones a través de la red para las capas superiores. En esta capa se efectúa la ruta entre las estaciones emisoras y receptoras, a través del establecimiento de trayectorias, resolviendo problemas de ruteo.

CAPA DE ENLACE.

Esta capa tiene el control del enlace lógico, direccionamiento, sincronización punto a punto, detección y corrección de errores. Ofrece la entrega confiable de datos a través del nivel físico y se da mediante la transmisión de tramas de información. Los dispositivos empleados en esta capa son las diferentes tarjetas de red.

CAPA FISICA.

Se le considera como la interfaz del medio, la interconexión física y de transmisión de bits. Esta capa define las características físicas, mecánicas y eléctricas del medio de comunicación. Controla el intercambio de información a nivel de bits, lo relacionado con la velocidad de información y la conexión física. Esta capa ofrece los mecanismos necesarios para la activación, mantenimiento y desactivación de las conexiones físicas.

1.6 MODELO DE REFERENCIA DoD.

1.6.1 NIVELES DEL MODELO DoD.

Este modelo de referencia surgió a finales de los años 70's bajo el patrocinio del Departamento de Defensa de los Estados Unidos (DoD), el cual es un modelo pensado por y para un usuario, y no desarrollado por un fabricante El concepto fundamental de este modelo es la comunicación entre los procesos.

CAPA	FUNCION
4	Nivel de proceso / aplicación
3	Nivel Transporte
2	Nivel Internet
1	Nivel de acceso a red

Fig. 1.4. Modelo de referencia DoD.

NIVEL DE PROCESO / APLICACION.

Contiene los protocolos que realizan funciones específicas de compartición de recursos y de acceso remoto, que permiten al usuario transferir archivos e intercambiar mensajes.

NIVEL DE TRANSPORTE

Esta capa realiza los procesos de comunicación entre diferentes terminales. Otros servicios que puede ofrecer esta capa son la detección de errores y el control del flujo de datos.

NIVEL INTERNET

En esta capa se realizan operaciones para que la información atraviese las múltiples redes entre terminales Los protocolos están definidos entre las terminales y los ruteadores. Otra característica es el uso de direcciones para permitir la comunicación con procesos en terminales correspondientes a otras redes. También se incluyen funciones como la identificación de trama del mensaje, segmentación y reensamblado, así como el manejo de errores.

NIVEL DE ACCESO A RED.

En este nivel existen protocolos que comunican a las terminales y a los Procesadores de Comunicación de Subred CSNP (Communications Subnet Processor), para permitir el acceso a red. Permite a un host enviar datos a su correspondiente CSNP (más una indicación de dónde debe ser enviado), recibir datos y regular el flujo de datos en el enlace host – CSNP.

1.7 MODELO DE REFERENCIA TCP/IP.

1.7.1 NIVELES DEL MODELO TCP/IP.

Este modelo fue diseñado para poder intercambiar las partes de cada nivel según el tipo de conexión que se utilizara. Como se concibió con varias tecnologías distintas, las partes tenían que estar estructuradas de forma que sólo fuera necesario cambiar una pieza para utilizarlo en un medio diferente. En la figura 1.5 se muestra el modelo TCP/IP y el conjunto de protocolos en cada nivel.

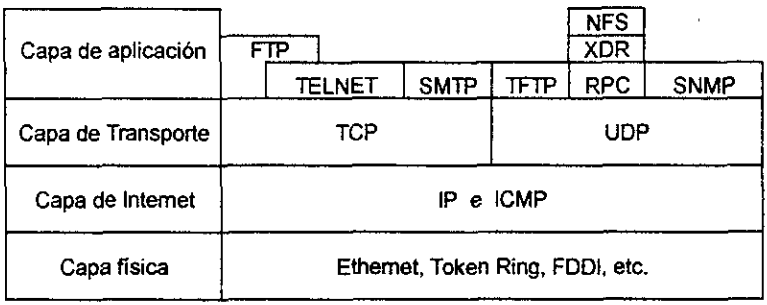

Fig. 1.5. Modelo de referencia TCP/IP.

NIVEL DE APLICACION.

Es la capa que contiene las aplicaciones de Internet y de red en general. Entre los ejemplos de aplicaciones de Internet se incluyen los programas para la comunicación personal, tales como el correo electrónico, acceso a terminales remotas, ejecución de trabajos remotos, boletines, simulación de terminal virtual, transferencia de archivos etc.

NIVEL DE TRANSPORTE

Esta capa es responsable de la comunicación terminal a terminal. Es aqui en donde los programas y los procesamientos en diferentes computadoras finalmente llegan a conectarse y comunicarse una con otra. Si la conexión es orientada, proporciona mecanismos de seguridad y realiza operaciones que evitan el tráfico para ofrecer fluidez a través de Internet.

En este nivel se definen dos protocolos: TCP y UDP El primero, proporciona una conexión mediante el modelo de circuito virtual orientado. UDP proporciona una conexión deliberada que en esencia es proporcionada por la capa inferior (Protocolo IP) y es el servicio de datagrama (paquete de datos).

NIVEL INTERNET.

Esta capa proporciona las funciones necesarias para conectar redes y ruteadores dentro de un sistema coherente, esto es, hace que muchas redes se conecten a la capa de Internet. El protocolo definido para esta operación es el IP, el cual es responsable de entregar los datos desde la fuente hasta su destino final (la capa superior). IP proporciona el servicio de datagrama (paquete de datos) a las capas superiores, también proporciona el servicio de ruteo (vía de acceso más rápida) para las computadoras conectadas de red. En esta capa se encuentra también el Protocolo ICMP.

NIVEL FISICO.

Para referirnos de manera general, la capa inferior contiene a las subredes y a la interfaz de subred, que se ocupan de introducir datos en cada red desde el protocolo de

paquetes X.25 hasta las especificaciones de la IEEE 802 para redes de área local (la red puede ser Ethernet, Token Ring etc.). Esto es, proporciona acceso la medio con las ventajas de proveer un control de flujo así como la detección y corrección de errores.

1.7.2 OPERACIÓN DE LAS CAPAS TCP/IP.

El funcionamiento de las capas de TCP/IP se puede ejemplificar a través de la comunicación de dos subredes, como puede apreciarse en la figura 1.6. Aquí la aplicación del usuario de la terminal A envía una solicitud al protocolo de la capa de aplicación en la terminal B, por ejemplo, para realizar una transferencia de archivo. El software de transferencia de archivos ejecuta una variedad de funciones y adiciona un encabezado de transferencia de archivo al usuario de datos. Como se indica el sentido del flujo de datos mediante las flechas, esta unidad de datos es enviada a la capa inferior que corresponde al protocolo de transporte. En esta capa se realizan una serie de funciones y se agrega otro encabezado a la aplicación cuando pasa por ésta. La unidad de datos se llama ahora segmento.

El siguiente paso es llevar este segmento a la capa de red o Internet a través del protocolo IP; esta unidad es ahora llamada datagrama, y se pasa a la siguiente capa inferior. Aquí la capa de enlace de datos agrega nuevamente otro encabezado y esta nueva unidad de datos (conocida ahora como frame o trama) es enviada hacia el interior de la red por medio de la capa física.

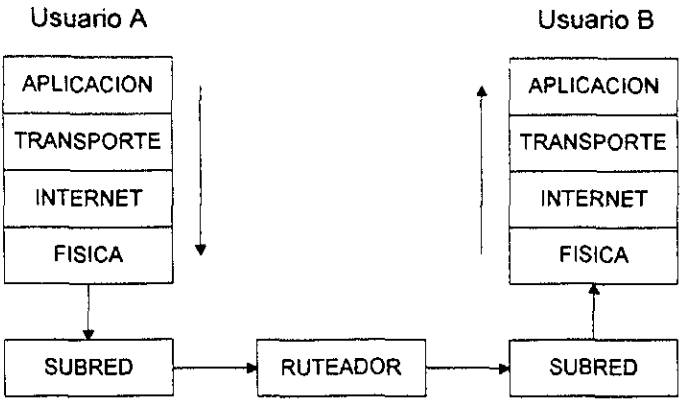

Figura 1.6. Operación de las capas TCP/IP.

El protocolo IP desconoce lo que sucede en el interior de la red, debido a que el administrador de red es libre para manejar la aplicación en la forma en que sea necesaria. Las subredes en ocasiones se auxilian de un ruteador, donde procesan el frame; este ruteador se enlaza con otra red hasta llegar a las capas inferiores y pasan a la capa de IP (capa de red) del terminal B. Aquí el proceso se revierte al desglosar los encabezados correspondientes a las capas.

En este trabajo se estudiarán las características de la capa de aplicación según el modelo TCP/IP, para lo cual el siguiente capítulo explicará como antecedente la capa Internet y la capa de transporte, que son la base para los protocolos de aplicación.

CAPITULO II

PROTOCOLOS DEL NIVEL INTERNET Y TRANSPORTE DEL MODELO TCP/IP.

En la capa Internet operan principalmente dos protocolos: el Protocolo IP, y el Protocolo ICMP. El primero se encarga de fragmentar la información en datagramas, y enrutarla a través de los medios físicos de la red; mientras que ICMP, se encarga de la notificación de algunos errores durante la transmisión de los datagramas.

Dentro de esta capa existe también el Protocolo de Resolución de Dirección (ARP) y el Protocolo de Resolución de Dirección Inverso (RARP), que tienen la función de buscar la dirección física a la cual se transmitirá la información; sin embargo, debido a que estos protocolos se refieren más al nivel físico, no serán incluidos en el presente trabajo, pues este se enfoca a las funciones de la capa de aplicación.

En la capa de transporte existen dos opciones separadas, UDP y TCP. El uso de estos protocolos depende del tipo de servicio que sea requerido por la aplicación del usuario, aunque ambos tienen por objetivo el proporcionar la comunicación entre un programa y otro. Estos protocolos son llevados en el campo de datos del datagrama IP.

Algunas de las principales responsabilidades de la capa de transporte son el proporcionar un servicio de flujo de datos libres de errores, el flujo controlado, el acceso a datos a la aplicación correcta y el multiplexaje de datos de muchas aplicaciones. Para hacer esto, el software del protocolo envía acuses de recibo de retorno y retransmite los paquetes perdidos.

El software de transporte divide el flujo de datos que se está enviando en pequeños fragmentos (paquetes) y pasa cada uno con una dirección destino, hacia la siguiente capa de transmisión.

La capa de transporte también debe aceptar datos desde varios programas de usuario y enviarlos a la capa del siguiente nivel. Para hacer esto, se añade información adicional a cada paquete, incluyendo códigos que identifican qué programa de aplicación envía y qué programa de aplicación debe recibir, así como una suma de verificación para asegurar que el paquete ha llegado intacto.

2.1 PROTOCOLOS DE LA CAPA INTERNET.

2.1.1 PROTOCOLO INTERNET (IP).

El IP es uno de los protocolos del nivel de red y provee envío de mensajes en el modo de servicio sin conexiones, añadiendo el formato de todos los datos enviados a la red. IP no garantiza la entrega de datos, no informa al transmisor ni al receptor el estado del paquete. Esto significa que si ocurre una falla en el enlace de datos o si ocurre algún error recuperable, el nivel IP no conforma las partes involucradas. Es responsabilidad de las capas superiores ejecutar procedimientos de recuperación de errores.

IP acepta datos del nivel de transporte (UDP ó TCP), los encapsula (formatea) para transmitirlos al nivel de enlace de datos (nivel físico); De manera inversa, IP toma datos del nivel de enlace de datos y los entrega a las capas superiores.

La unidad de información que transfiere IP es conocida como datagrama, la cual se compone de un área llamada encabezado, que lleva información de control, y un área de datos. El software IP en un nodo crea un datagrama que adapta dentro del marco físico de la red. Sin embargo, en el viaje a su destino, un datagrama puede tener que pasar a través de muchos tipos de redes.

Para manejar esta faceta de la transmisión, el IP especifica un método de rompimiento de datagramas en fragmentos en cualquier nodo que deba transmitir el datagrama, y un método de ensamblaje de fragmentos en el nodo destino. Por lo tanto, un ruteador que recibe paquetes de una red con un tamaño definido puede necesitar fragmentar los paquetes IP recibidos para transmitirlos a otra red, si la segunda red tiene un marco de

red más pequeño. Una vez que se fragmentaron los paquetes, no serán reensamblados hasta que alcancen su destino final

Las tareas principales de este protocolo son:

- Direccionamiento de los datagramas entre computadoras, determinando a dónde se enviarán, así como las rutas alternas en caso de problemas.
- Administración del proceso de fragmentación de dichos datagramas.

El protocolo tiene una definición formal del diseño del datagrama y de la información de un encabezado, compuesto por la información a la que se refiere o maneja el datagrama. IP no tiene nada que ver con la confiabilidad en la entrega de la información y no posee la capacidad para checar el estado de la información, pues no contiene suma de verificación para el contenido de datos de un datagrama, solo para la información del encabezado. Las tareas de verificación y control de flujo se dejan a otros componentes del modelo en capas.

Un datagrama se puede destruir en el camino debido a:

- Errores de bits durante su transmisión en el medio. La información se puede retrasar, mutilar al dividir y reensamblar los datagramas, así como también enrutar de manera incorrecta
- Un ruteador congestionado descartó el datagrama debido a la falta de espacio en su área de almacenamiento o buffer.
- Temporalmente, no había camino hacia el destino.

El protocolo puede dividir en forma automática un datagrama de información en datagramas más pequeños si es necesario, esto es, en caso que se necesite enviar un datagrama de una longitud mayor a la permitida en el siguiente enlace. Cuando el primer

datagrama de un mensaje que se dividió llega a su destino, da inicio un sincronizador de reensamble. Si no se han recibido todas las piezas de un datagrama completo cuando el sincronizador llega a un valor predeterminado, todos los datagramas que se han recibido serán descartados. Por un campo del encabezado IP la máquina receptora sabe el orden en que se deben reensamblar las piezas.

IP se preocupa de los nodos a través de los cuales pasa un datagrama a lo largo de la ruta, e incluso en qué máquinas empieza y termina el datagrama. También se ocupa del direccionamiento del datagrama mediante la dirección completa Internet de 32 bits.

2.1.1.1 ENCABEZADO DEL DATAGRAMA DEL PROTOCOLO INTERNET.

El datagrama es la unidad de transferencia que IP utiliza. El encabezado del Protocolo Internet emplea 6 palabras de 32 bits (24 bytes) cuando se incluyen todos los campos opcionales. A continuación se muestra el esquema del encabezado de IP, en donde el encabezado termina hasta el área de datos, y cada renglón representa una palabra:

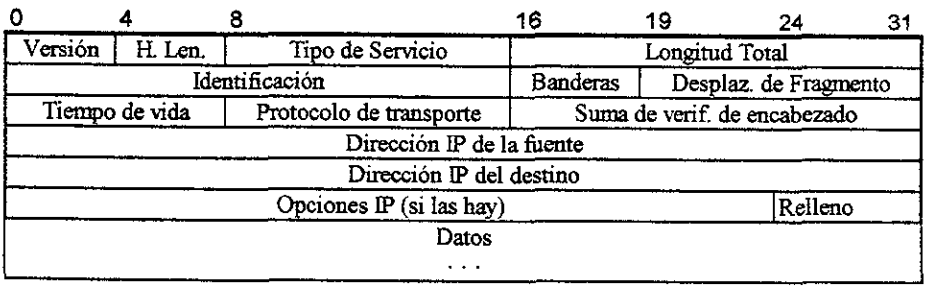

Figura 2.1. Formato de un datagrama Internet.

La explicación para cada campo del datagrama IP es la siguiente:

- **Número de Versión**.- Este es un campo de 4 bits que contiene el número de versión IP que soporta el software del protocolo. Se requiere el número de versión para que el software receptor sepa cómo descifrar el resto del encabezado. La versión actual de IP es la 4. La versión de la siguiente generación es la 6.
- **Longitud del Encabezado (H.Len.)**.- Este campo de 4 bits refleja la longitud total del encabezado IP dado en palabras de 32 bits. Para descifrar correctamente el encabezado, IP debe saber cuando termina el encabezado y empiezan los datos, razón por la cual se incluye este campo.
- **Tipo de Servicio**.- Este campo está formado por 8 bits y da las instrucciones a IP para procesar correctamente el datagrama, los 3 primeros bits indican la procedencia del datagrama; los siguientes 3 bits controlan el retraso, el rendimiento y la contabilidad; los 2 bits finales no se emplean.
- **Longitud del Datagrama**.- Este campo da la longitud total del datagrama, incluyendo el encabezado, en bytes. El tamaño del campo de longitud máxima es de 65 535 bytes para un datagrama.
- **Identificación.**- Este campo contiene un número que es un identificador único creado por el nodo emisor. Este nodo se requiere al volver a ensamblar los mensajes para asegurar que en la fragmentación no se mezclen los mensajes con los de otros nodos.
- **Banderas**.- El campo de banderas es de 3 bits, el primero de los cuales no se emplea. Los 2 bits restantes se dedican a banderas conocidas como DF (*no fragmentar*) y MF (*más fragmentos*), las cuales controlan el manejo de los datagramas cuando la fragmentación resulta confiable.

- **Desplazamiento de los Fragmentos** - Si se encuentra establecido el bit de bandera MF, el desplazamiento de fragmento contiene la localización del submensaje en el mensaje completo. Esto permite que IP reensamble los paquetes fragmentados en orden apropiado.
- **Tiempo de Vida (TTL).**- Este campo proporciona la cantidad de tiempo en segundos que un datagrama puede permanecer en la red antes de descartarse. El nodo emisor establece este tiempo (normalmente entre 15 y 30 segundos).
- **Protocolo.**- Este campo contiene el número de identificación del protocolo con el cual se ha manejado el paquete. Al protocolo TCP le corresponde el número 6, al UDP el 17 y al ICMP el número 1.
- **Suma de Verificación del Encabezado.**- Número entero calculado a partir de los bits empleados en la cabecera del datagrama. Se utiliza para detectar errores durante la transmisión del mismo. El receptor del datagrama recalcula la suma de verificación y lo compara con el dato obtenido de la transmisión.
- **Direcciones de la Fuente y del Destino.**- Estos campos contienen las direcciones IP de 32 bits de los dispositivos emisor y destino. Estos campos se establecen al crearse el datagrama, y no se alteran durante el enrutamiento.
- **Opciones.**- Este campo es optativo y está compuesto de varios códigos distintos de longitud variable. Si en el datagrama se emplean más de una opción, éstas aparecen en el encabezado IP en forma consecutiva. La clase y el número de opción indican el tipo de opción y su valor en particular. Las de mayor importancia son las opciones que permiten el enrutamiento y que las marcas de tiempo sean registradas. Estas se emplean

para obtener un registro del paso del datagrama a traves de la red, lo que resulta útil para efectos de diagnóstico

Hay dos tipos de enrutamiento indicados en los campos de opciones: libre y estricto. El enrutamiento libre proporciona una serie de direcciones IP por las cuales debe pasar el datagrama, pero permite cualquier ruta para llegar a cada una de esas direcciones. El enrutamiento estricto no permite ninguna desviación de la ruta específica. Si la ruta no se puede seguir, el datagrama se abandona.

- **Relleno**.- El contenido del área depende de las opciones seleccionadas. Por lo general el relleno se utiliza para asegurar que el encabezado del datagrama corresponde al número redondeado de bytes.

2.1.1.2 DIRECCIONES DE RED.

Las direcciones de la red son análogas a las direcciones de correos. Tres términos se emplean comúnmente y estos son: el nombre, la dirección y la ruta.

Un nombre es la identificación específica de una máquina, un usuario o una aplicación. Generalmente es único y proporciona un objetivo absoluto para el datagrama.

Una dirección típicamente identifica la localización del objetivo, por lo general su localización lógica o física.

Una ruta le dice al sistema cómo hacer llegar el datagrama a la dirección. Un paquete de software de red, conocido como servidor de nombres, tratará de descifrar la dirección y la ruta a partir del nombre. El uso de un servidor de nombres tiene como ventaja primordial además de hacer que el direccionamiento y el enrutamiento resulten irrelevantes

para el usuario final. Las reglas convencionales para identificar por nombres difieren dependiendo de la plataforma de la red.

En una red, es necesario definir dos tipos de direcciones para asegurar la entrega correcta de los datos: la *dirección física* y la *dirección IP*.

- **Dirección Física**.

Cada dispositivo de una red que se comunica con otros dispositivos tiene una *dirección física* única, también conocida como *dirección de hardware*. Para el hardware, las direcciones están normalmente cifradas en la tarjeta de interfaz de red, establecidas ya sea mediante interruptores o software.

En la capa física se realiza el análisis de cada datagrama que se recibe. Si la dirección del receptor coincide con la dirección física del dispositivo, el datagrama puede pasarse hacia arriba por las capas. La longitud de la dirección física varía dependiendo del sistema de la red, pero Ethernet y algunos más utilizan para cada dirección 48 bits. Por lo que se requieren dos direcciones para entablar una comunicación: una para cada uno de los dispositivos, el emisor y el receptor.

- **Dirección IP.**

Las direcciones IP sirven para identificar el host y encaminar los datos hacia ellos. Todos los host deben tener una dirección IP única para las comunicaciones. El nombre de host se traduce a su dirección IP consultando el nombre en una base de datos de pares nombre-dirección. Una dirección IP es un número binario de 32 bits separados por un punto cada octeto (es decir, cada 8 bits), para después transformar cada octeto a su valor decimal.

Por ejemplo, la dirección del host *blintz.med.yale.edu* de la Universidad de Yale, es un número binario de 32 bits cuya notación es:

10000010.10000100.00010011.00011111

130.132.19.31

Se puede deducir entonces, que el mayor número que puede aparecer en una posición dada es 255, que corresponde al número binario 11111111.

Una dirección IP tiene un formato de dos partes que son la dirección de red y la dirección local. La dirección de red identifica la red a la que está conectado el nodo. La dirección local identifica a un nodo particular dentro de la red de una organización.

Todas las computadoras deben tener una dirección IP única en el rango de sistemas con los que se comunican. Para lograr esto, toda organización que planee conectarse a Internet debe conseguir un bloque de direcciones IP únicas. Las direcciones se consiguen de la autoridad de registro de direcciones IP, la InterNIC (Internet Network Information Center), quien delega grandes bloques de su espacio de direcciones IP a los proveedores locales de conexión a Internet, que a su vez lo asignan a sus usuarios.

2.1.2 PROTOCOLO INTERNET DE CONTROL DE MENSAJES (ICMP).

En el sistema sin conexión los ruteadores operan de manera autónoma, ruteando o entregando los datagramas que llegan sin coordinarse con el transmisor original. El sistema trabaja bien si todas las máquinas trabajan de manera correcta y si están de acuerdo respecto a las rutas. Por desgracia ningún sistema opera de manera correcta. Además de las fallas en las líneas de comunicación y en los procesadores, el IP tiene fallas en la entrega de datagramas cuando la máquina de destino está desconectada temporalmente o

permanentemente, cuando el contador de tiempo expira o cuando los ruteadores intermedios se congestionan.

Para permitir que los ruteadores en una red reporten los errores o proporcionen información sobre circunstancias inesperadas, los diseñadores agregaron a los protocolos TCP/IP un mecanismo de mensajes de propósito especial. El protocolo ICMP (Internet Control Message Protocol), se considera parte obligatoria de IP y se debe incluir en todas las implementaciones de IP. Al igual que el resto de tráfico, los mensajes ICMP viajan a través de la red en la porción de datos de los datagramas IP. Sin embargo el destino final de un mensaje ICMP no es un programa de aplicación ni un usuario en la máquina destino, sino el software del Protocolo Internet en dicha máquina. Esto es, cuando llega un mensaje de error ICMP, el módulo de software ICMP lo maneja. Si ICMP determina que un protocolo de un nivel más alto o un programa de aplicación causaron un problema, notificará al módulo apropiado.

En un principio fue creado para ser implantado en ruteadores, aunque también puede ser instalado en máquinas para permitir que estas envíen mensajes ICMP a otras máquinas.

Entre las principales funciones de este protocolo se encuentran las siguientes:

- Informar de los errores ocurridos en el procesamiento de los datagramas.
- Proporcionar algunos mensajes de administración y estatus.

ICMP se utiliza entre computadoras y ruteadores por diversas razones, entre ellas:

- Cuando no se puedan enviar los datagramas.
- Para verificar la existencia de trayectorias hacia alguna red y el estado de la misma.
- Para reportar destinos inalcanzables.

- Cuando un ruteador no dispone de suficiente capacidad de almacenamiento (buffer) para retener y enviar unidades de datos.

2.1.2.1 REPORTE DE ERRORES.

ICMP es un mecanismo de reporte de errores, proporciona una forma para que los ruteadores que encuentren un error lo reporten a la fuente original. Aunque en esencia ICMP resalta algunas acciones deseables a tomar, es necesario aclarar que este protocolo no especifica las acciones que hay que tomar para cada falla.

La mayor parte de los errores provienen de la fuente original, pero otros no, sin embargo, debido a que el ICMP reporta los problemas a la fuente original, no se puede utilizar para informar los problemas a los ruteadores intermedios.

Un mensaje de error ICMP podría presentarse, por ejemplo, si un usuario intenta conectarse a una dirección de red inalcanzable por medio del protocolo Telnet, en donde sucedería lo siguiente:

```
>telnet 150.100.1.1
Trying 150.100.1.1 ...
telnet connect: Host is unreachable
```

Se ha enviado un mensaje de error generado por ICMP, informando al usuario que el host al que se intenta conectar es inalcanzable.

Algunos mensajes de error ICMP que pueden presentarse son:

- Destination Unreachable (Destino inalcanzable).- Un datagrama no puede llegar a su host, utilidad o aplicación de destino.
- Redirect (Redirigir).- Un host ha enviado un datagrama al ruteador equivocado.

- Time Exceeded (Plazo superado).- El tiempo de vida de un datagrama ha terminado en un ruteador o el plazo de reensamblado en un host de destino ha expirado.
- Parameter Problem (Problema de los parámetros) - Existe un parámetro erróneo en el encabezado IP.
- Source Quench (Disminución de origen).- Un ruteador o un destino está congestionado. Se recomienda que los sistemas no envíen estos mensajes.

2.1.2.2 ENTREGA DE MENSAJES ICMP.

Cada mensaje ICMP viaja a través de la red en porción de datos de un datagrama IP, el cual viaja a través de cada red física en la porción de datos de una trama. Los datagramas que llevan mensajes ICMP se rutean exactamente como los que llevan información de usuario y no existe ni una confiabilidad ni una prioridad adicional. Por lo tanto los mensajes de error se pueden perder o descartar; y pueden causar congestionamiento pero ello no quiere decir que estos mensajes causen a su vez otros mensajes de error ICMP.

Estos mensajes requieren dos niveles de encapsulación, como se muestra:

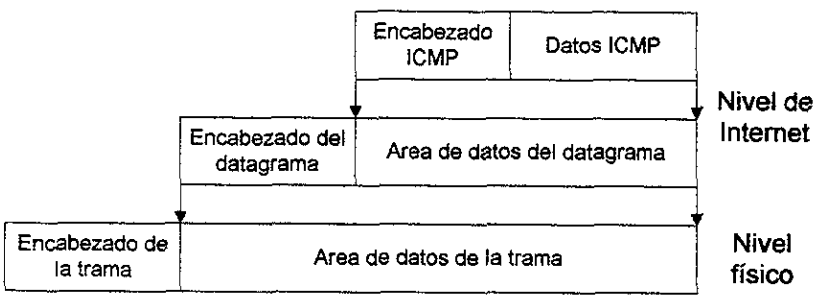

Figura 2.2. Encapsulación del mensaje ICMP.

Es importante tener en mente que aunque los mensajes ICMP se encapsulan y envían mediante el IP, el ICMP no se considera como un protocolo de nivel más alto sino como una parte obligatoria del IP La razón de utilizar IP para entregar mensajes ICMP es que quizás necesiten viajar a través de muchas redes físicas para alcanzar su destino final.

Uno de los usos más comunes de ICMP es el programa PING, el cual es un mensaje ICMP que intenta localizar otras estaciones en el Internet, y determina si están activas o para ver si una ruta física está levantada. Otro uso del Ping es checar los retardos en una ruta. La respuesta a un ping solicitante reporta el retardo en la respuesta, medido en milisegundos. Es ampliamente utilizado por administradores de red, para validar que los enlaces estén bien.

2.1.2.3 FORMATO DE MENSAJES ICMP.

La figura 2 3 muestra el formato de los mensajes ICMP. Como se mencionó anteriormente, esos mensajes se sitúan en la parte de datos del datagrama IP. El campo de protocolo del encabezado IP toma un valor de 1 para indicar que se está utilizando ICMP.

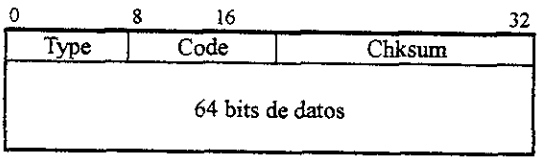

Figura 2.3. Formato de un mensaje ICMP.

Todos los mensajes ICMP contienen tres campos:

- **Tipo**.- Define el tipo de mensaje.
- **Código**.- Define el tipo de error o información de estatus.

- **Suma de verificación**.- Realiza una suma de verificacion para determinar errores en el encabezado.

En el área de datos de un mensaje ICMP se envían los primeros 64 bits de datos del datagrama que causó el problema.

Dentro del campo de código, se definen algunos de los servicios de información de errores y estatus, los cuales se resumen a continuación:

CODIGO	TIPO DE MENSAJE ICMP
0	Respuesta de Eco
3	Destino inaccesible
4	Disminución de origen
5	Redireccionar
8	Solicitud de Eco
11	Tiempo excedido
12	Parámetro no comprensible
13	Solicitud de marca temporal
14	Respuesta de marca temporal
15	Solicitud de información
16	Respuesta de información

Figura 2.4. Tipo de mensajes ICMP.

- ECO Y RESPUESTA DE ECO (PING)- El eco se puede enviar a cualquier dirección IP, por ejemplo, un ruteador, el cual debe devolver una respuesta de eco al solicitante. De esta forma los administradores de red pueden saber cuál es el estado de los recursos de la red.

- DESTINO INALCANZABLE Un ruteador lo invoca si encuentra problemas para alcanzar la red de destino especificada.
- DISMINUCION DE ORIGEN: Este servicio es utilizado por una máquina cuando no tiene suficiente espacio de almacenamiento para poner en cola los datagramas que van llegando.
- REDIRECCION: Este servicio sirve para suministrar a una computadora información de gestión de encaminamiento y es invocado por un ruteador. La redirección indica que hay disponible una mejor ruta.
- TIEMPO EXCEDIDO DE VIDA DEL DATAGRAMA: Servicio ejecutado por un ruteador en caso de que el tiempo de vida del datagrama IP haya expirado.
- PARAMETRO NO COMPRENSIBLE. Este servicio se activa si una computadora o ruteador encuentran problemas al procesar cualquier parte de una cabecera IP.
- MARCA TEMPORAL Y RESPUESTA DE MARCA TEMPORAL. Este servicio es usado por los ruteadores y computadoras para determinar el retardo empleado en el envío de tráfico por la red o redes.

2.2 PROTOCOLOS DE LA CAPA DE TRANSPORTE.

2.2.1 PROTOCOLO DE CONTROL DE TRANSFERENCIA (TCP).

El Protocolo TCP (Transfer Control Protocol), proporciona un número considerable de servicios a la capa IP y a las capas superiores. Pero aún con mayor importancia, proporciona a las capas superiores un protocolo orientado a conexión, que permite a una aplicación asegurarse que un datagrama se reciba totalmente, por lo que TCP opera como un protocolo de validación de mensajes proporcionando comunicaciones confiables Si un datagrama se corrompe o se pierde, por lo general es TCP el que maneja la retransmisión. Para garantizar la seguridad, se añaden una cantidad significante de encabezados que son utilizados para manejar los reconocimientos, flujo de control, temporizadores y facilidad para el manejo de conexión.

TCP maneja el flujo de datagramas provenientes de las capas superiores, así como los datagramas de llegada provenientes de la capa IP. Este protocolo debe ser capaz de manejar la terminación de una aplicación en una capa superior, que estaba esperando la llegada de datagramas, así como fallas en capas inferiores. TCP también debe mantener una tabla de estado de todos los flujos de datos hacia dentro y fuera de la capa TCP. Sin esta capa, cada aplicación tendría que implementar estos servicios por sí misma, lo que resultaría un desperdicio de recursos.

Este protocolo reside en la capa de transporte, colocado encima de IP, pero debajo de las capas superiores de aplicación. TCP no reside en dispositivos que solamente enruten datagramas, por lo que en un ruteador no hay capa TCP, ya que aquí el datagrama no tiene necesidad de ir más arriba de la capa IP.

Los dos principales servicios que proporciona el protocolo TCP son

- Multiplexaje: Soporta múltiples conexiones por el uso de puertos, los cuales sirven para identificar a cada aplicación.
- Reporte de errores: Servicio de transporte de fallas.

Debido a que TCP es un protocolo orientado a conexión, por lo general se utiliza el término de circuito virtual para referirse al saludo existente entre dos máquinas terminales, la mayor parte de los cuales son simples mensajes de acuse de recibo y números de secuencia del datagrama.

2.2.1.1 FORMATO DEL SEGMENTO TCP.

La figura 2.3 muestra como es el formato de un segmento TCP.

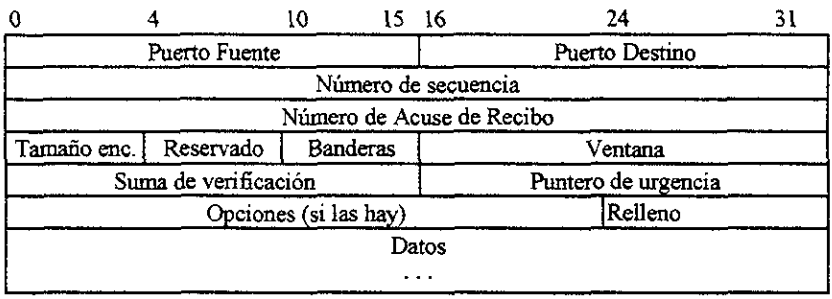

Figura 2.5. Formato de TCP.

Las partes que integran al formato TCP se describen a continuación:

- **Puerto Fuente y Destino**.- Contiene el número de puerto TCP que consiste en un número que identifica los programas de aplicación en los extremos de la conexión.

- **Número de secuencia**.- Identifica el número de secuencia del primer byte de la cadena de datos del segmento que se va a enviar.
- **Número de acuse de recibo**.- Identifica el número de secuencia del próximo segmento que el emisor espera recibir.
- **Tamaño del encabezado**.- Indica la longitud del encabezado en múltiplos de 32 bits. Es necesario porque el campo de opciones varía su tamaño.
- **Reservado**.- Apartado para usos futuros.
- **Banderas**.- Determina el propósito y contenido del segmento. Existen 6 banderas:

URG: Indica que los datos son urgentes. Se utiliza por ejemplo, cuando durante una transferencia de archivos, el usuario ha pulsado una tecla de aviso o interrupción, que permite saltar los bytes intermedios y trasladar esta información a la aplicación de destino.

ACK : Es un acuse de recibo, que muestra que un extremo de la conexión ha recibido con éxito la información que le fue enviada.

PSH: Indica si los datos deben enviarse inmediatamente, por ejemplo, cuando durante una sesión el usuario ejecuta un comando del programa de aplicación.

RST: Sirve para abortar una sesión. Se utiliza si TCP ha detectado un problema serio en la comunicación que no se puede resolver.

SYN: Señala el establecimiento de la conexión.

FIN: Indica la terminación correcta de la conexión.

- **Ventana**.- Informa cuantos datos esta dispuesto a aceptar el software TCP cada vez que envía un segmento.
- **Suma de Verificación**.- Verifica la integridad de los datos enviados. Esta suma es obligatoria para TCP, y se aplica sobre el segmento completo e incluye información del

datagrama IP. La información de IP que se toma en cuenta son los campos que contienen las direcciones IP origen y destino y el campo del número de protocolo cuyo valor es 6 para TCP. Para cada segmento entrante se calcula esta suma y se compara con el valor del campo de suma de verificación del encabezado TCP. Si los valores no coinciden, el segmento se descarta.

- **Puntero de urgencia**.- Cuando la bandera *URG* está activada, este puntero especifica la posición dentro del segmento en la que terminan los datos urgentes.
- **Opciones**.- Especifica el tamaño máximo de los segmentos que se transferirán.
- **Relleno**.- Se utiliza para asegurar que el encabezado termina en un múltiplo de 32 octetos.
- **Datos**.- Campo asignado para la información.

2.2.1.2 SEGUIMIENTO DE UN MENSAJE.

La aplicación traslada los datos a TCP, el cual sitúa estos datos en un buffer de envío. Toma un trozo de los datos y le añade un encabezado, creando un segmento. TCP traslada el segmento a IP para que lo entregue como un único datagrama. El empaquetado de datos en trozos del tamaño adecuado permite usar de manera eficiente los servicios de transmisión, por lo que TCP espera a recoger una cantidad razonable de datos antes de crear un segmento. La longitud del segmento por lo general es determinada por TCP o por un valor de sistema establecido por el administrador del sistema.

Además del buffer de envío, cada parte llama a una subrutina que crea un bloque de memoria para almacenar los parámetros de TCP y de IP durante la conexión, como los sockets (que consiste en los números de puerto y las direcciones IP del emisor y receptor),

los números de secuencia, el valor de IP para el tiempo de vida y otros La aplicacion servidora espera a los clientes. Un cliente que desee acceder al servidor lanza una solicitud de conexión mediante los sockets.

TCP emplea la numeración y la confirmación para la transferencia confiable de datos. La numeración consiste en que cada segmento tiene un número de secuencia en caso de que el mensaje completo sea de más de un segmento. La confirmación la llevan a cabo mensajes de acuse de recibo, llamados *ACK*. Se espera que el receptor confirme la recepción de los datos Si no llega un mensaje *ACK* en un plazo dado se retransmiten los datos. Esta estrategia se llama confirmación positiva con retransmisión.

El TCP receptor va observando la secuencia de números que llegan para mantener los datos en orden y para asegurarse de que no se pierdan datos. Como a veces se pierden algunos *ACK*, o llegan tarde, pueden llegar segmentos duplicados al receptor. Los números de secuencia indican cuáles son los datos duplicados que, por lo tanto, se pueden descartar.

Después de establecer el circuito virtual, TCP envía el segmento al software IP, que a su vez envía el mensaje a la red como un datagrama. Una vez que haya establecido el camino a través de la red, el IP de la máquina receptora pasa el segmento recibido a la capa de la misma máquina donde se procesa y pasa a las aplicaciones superiores mediante el uso de un protocolo de capa superior. Si el mensaje tenía más de un segmento de largo, el software TCP lo ensambla utilizando los números de secuencia del encabezado. Si algún segmento es erróneo o esta corrompido, TCP devuelve un mensaje con el número de secuencia defectuoso.

Al igual que la mayor parte de los protocolos basados en conexión, los temporizadores son un aspecto importante de TCP, ya que estos aseguran que no se espere más tiempo del necesario un acuse de recibo o un mensaje de error. Si el temporizador

termina se supone una transmisión incompleta. Por lo que un temporizador que termina antes de un envio de mensajes de acuse causará la retransmisión del datagrama.

2.2.2 PROTOCOLO DE DATAGRAMA DE USUARIO (UDP).

El Protocolo UDP (User Datagram Protocol), reside en la capa de transporte, arriba de la capa del protocolo Internet y bajo la capa de aplicación.

UDP proporciona un servicio de entrega de paquetes sin conexión y no confiable, es decir, estos mensajes se pueden perder, duplicar, retrasar o entregar en desorden. Los datagramas son enviados a un nodo remoto sin algún requerimiento de respuesta que indique si el datagrama ha llegado. Los servicios de aplicación, tales como TFTP y NFS utilizan este tipo de transporte.

Este protocolo utiliza el IP para entregar datagramas. La diferencia importante entre los datagramas UDP y los IP, es que el UDP incluye un número de puerto de protocolo, lo que permite al emisor distinguir entre varios programas de aplicación en una máquina remota dada. En la práctica el UDP también incluye una suma de verificación opcional en el datagrama que se está enviando.

Por encima del trabajo efectuado por IP, UDP brinda únicamente un número de puerto y una verificación. A diferencia de TCP, en este caso, no hay acuses de recibo de transporte u otros mecanismos de confiabilidad, sin embargo, a la falta de estas adiciones hace que UDP sea particularmente eficiente, y por esta razón, es adecuado para aplicaciones que demandan una alta velocidad (por ejemplo, sistemas de archivos distribuidos NFS, etc.) lo cual es de alguna manera, sólo adecuado para instalación sobre un

medio de transporte rápido y con probabilidades de falla muy bajo, por ejemplo en redes de área local.

Algunos productos permiten configurar el tamaño máximo de un datagrama UDP, lo cual sería necesario si algunos componentes del sistema no estuvieran listos para recibir datagramas grandes. Reduciendo el Tamaño Máximo del Datagrama (MDS) disminuye la cantidad de memoria requerida en la terminal, pero también reduce el desempeño de los servicios que utilizan este protocolo. Por esta razón no es difícil que los valores ya predefinidos se cambien.

2.2.2.1 FORMATO DE LOS MENSAJES UDP.

Cada mensaje UDP se conoce como datagrama de usuario. Un datagrama de usuario consta de dos partes: un encabezado UDP y un área de datos UDP. Como se muestra en la figura el encabezado del protocolo UDP se divide en cuatro campos de 16 bits, que especifican el puerto desde que se envió el mensaje, el puerto para el que se destina el mensaje, la longitud del mensaje y una suma de verificación UDP.

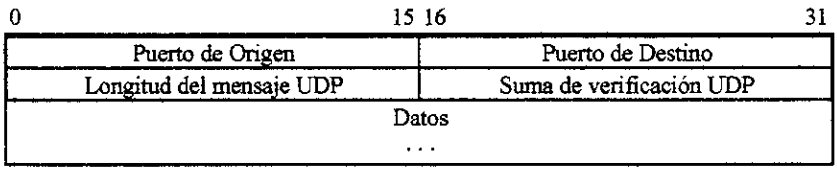

Figura 2.6. Formato de los campos en un datagrama UDP.

Los significados de los campos del encabezado UDP son los siguientes:

Puerto UDP de Origen y Destino.- Contienen los números de puerto del protocolo UDP utilizados para el demultiplexado de datagramas entre los procesos que esperan recibir. El puerto de origen es opcional; cuando se utiliza, especifica la parte a la que se deben enviar las respuestas, de lo contrario, puede tener valor de cero.

Longitud del Mensaje UDP.- El campo de longitud contiene un conteo de los octetos en el datagrama UDP, incluyendo el encabezado y los datos del usuario UDP. Por lo tanto, el valor mínimo para el campo de longitud es de 8 bytes, que es la longitud del encabezado.

Suma de Verificación UDP.- La suma de verificación UDP es opcional. Un valor de "0" en el campo suma de verificación significa que la suma no se calculó. Como en el caso de TCP, la suma de verificación incluye información del datagrama IP (direcciones IP origen y destino y el número de protocolo que es 17 para UDP). El IP no computa una suma de verificación de la porción de datos de un datagrama IP (sólo se calcula la del encabezado). Así que, la suma de verificación UDP proporciona la única manera de garantizar que los datos lleguen intactos, por lo que se debe utilizar.

Con esto concluye este capítulo en el que se han analizado cuatro protocolos que explican el manejo de la información que debe ser transmitida a través de la red, hasta llegar a los programas de aplicación, en donde el usuario recibe los servicios de la misma. En los siguientes capítulos se examinarán los protocolos de alto nivel, calsificándolos de acuerdo al servicio que proporcionan en: acceso remoto, transferencia y acceso de archivos, correo electrónico y administración de red.

CAPITULO III

ACCESO REMOTO

En este capítulo comienza la descripción de uno de los servicios más utilizados en Internet: el acceso remoto.

Este servicio permite al usuario accesar a los servicios de una computadora distante (llamada servidor), de manera que la conexión se realice como si el teclado y el monitor del usuario fueran los mismos de la máquina remota.

El protocolo Telnet, como parte de la familia TCP/IP, ofrece un programa mediante el cual es posible presentar al usuario de manera transparente el servicio de acceso remoto, mediante una conexión TCP. Telnet especifica también la manera en que se establecerá esta conexión y especifica los comandos necesarios para que el usuario pueda personalizar la conexión. Este protocolo se basa en un modelo cliente/servidor, por lo que un usuario debe tener en su computadora un programa cliente Telnet, si es que desea utilizar los recursos de una terminal remota.

La necesidad de usar el protocolo Telnet surge por las incompatibilidades que pudieran existir entre los diferentes equipos, porque como se recordará TCP/IP está

diseñado para permitir la comunicación de diferentes equipos sin importar la arquitectura de éstos. Telnet se encarga de convertir la información que se envia desde la computadora local a un formato que sea reconocido por el sistema remoto, y ya en éste la información es nuevamente convertida al formato propio del sistema.

El acceso remoto es útil cuando un usuario está enfrente a una máquina de poca potencia y desea utilizar las capacidades de procesamiento de otra máquina, o si otra máquina tiene alguna herramienta en particular que el usuario no desea cargar en su máquina local.

Además, la aplicación actual más importante que se le da a Telnet es la administración de redes WAN, al ser una herramienta que permite accesar a ruteadores para cambiar su configuración y optimizar su funcionamiento. Desde el teclado del usuario se puede controlar el equipo remoto, así como ver el resultado de sus acciones en la pantalla.

3.1 PROTOCOLO TELNET.

3.1.1 DESCRIPCION DE TELNET.

El protocolo Telnet (Telecommunications Network) permite accesar y utilizar los recursos de un sistema remoto como si estuvieran conectados al sistema local.

Telnet pretende proporcionar conexión remota o capacidad de terminal virtual a través de una red. En otras palabras, un usuario de la máquina A debería ser capaz de registrarse en la máquina B desde cualquier parte de la red, y por lo que respecta al usuario, aparecer como si estuviera sentado frente a la máquina B.

Telnet se encarga de mandar la información generada desde un dispositivo local (por ejemplo el teclado) hacia la terminal remota, donde la información es procesada y posteriormente, el resultado es enviado de regreso a la computadora local, de forma que la información es desplegada en su dispositivo de salida (por ejemplo el monitor).

Telnet se creó debido a que en un tiempo el único método para permitir que una máquina tuviera acceso a otra era estableciendo un enlace mediante dispositivos de comunicaciones como módems o redes en puertos dedicados. El problema de lograr este tipo de enlace se debe a la amplia variedad de terminales y computadoras existentes, cada una con sus propios códigos de control y características de terminal. Cuando se está conectado directamente a un servidor, la unidad central de procesamiento (CPU) de éste, debe administrar la conversión de los códigos de terminal entre ambas, lo que impone una severa carga en el CPU del servidor. Con varias conexiones remotas activas, el CPU del servidor puede gastar una cantidad considerable de tiempo administrando las conversiones.

Telnet aligera este problema incrustando las secuencias caracteristicas de terminal dentro del protocolo Telnet Cuando dos máquinas se comunican mediante Telnet, durante la fase de conexión Telnet mismo determina y establece los parámetros de comunicación y de terminal para la sesión, e incluye capacidad de no aceptar un servicio que uno de los extremos no pueda administrar. Cuando se establece una comunicación mediante Telnet, ambos extremos acuerdan un método para el intercambio de información entre las dos máquinas, descargando al CPU del servidor de un porcentaje considerable de este trabajo.

Este protocolo cumple con dos funciones básicas y tres principales servicios. Las funciones de Telnet son:

- En la computadora local actúa como cliente, esto es porque el protocolo solicita un servicio a una computadora remota.
- En la terminal remota, Telnet actúa como servidor, porque recibe solicitudes de algún otro sistema.

Los tres servicios principales son:

- Define una terminal virtual de red, la cual provee de una interfaz estándar con el sistema remoto.
- Provee un grupo de opciones que pueden ser negociadas entre ambos sistemas.
- Trata ambos extremos de la conexión de manera simétrica, porque cualquier extremo puede solicitar una opción, siempre y cuando el otro extremo tenga la posibilidad de manejar esta opción.

Telnet se basa en un modelo cliente/servidor, lo que significa que el usuario ejecuta un programa (el cliente) para utilizar los recursos de una computadora remota (el servidor).

Este servidor, permite que muchos clientes distintos accedan a sus recursos simultáneamente, es decir, no está dedicado a un único usuario.

La figura 3.1 permite ver cómo los programas de aplicación usan Telnet para establecer la comunicación. Cuando algún programa en el sistema local requiere de un servicio remoto, hace uso de Telnet. El programa de aplicación en el sistema local se convierte en cliente, estableciendo una conexión TCP con el sistema remoto. El sistema local recibe la información desde el dispositivo de entrada, y por medio de la red, la información es mandada al servidor donde es procesada. Finalmente, se manda la salida de regreso al sistema local. Todo lo anterior tomando en cuenta que el sistema operativo sirve de interfaz entre la red y Telnet.

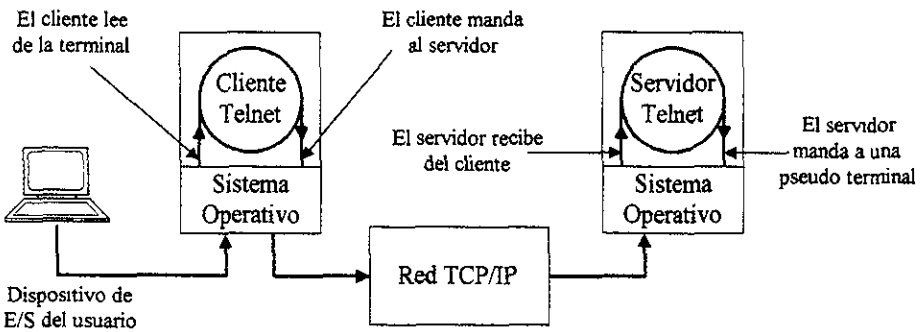

Figura 3.1. Trayectoria de los datos en una sesión Telnet.

En la figura se muestra al servidor dando servicio a un solo cliente, o sea realizando una sola tarea. En realidad el servidor puede procesar varias tareas para poder dar servicio a más de un solo cliente, en este caso existe un servidor maestro el cual se encarga de crear

varios servidores esclavos y cada uno se encarga de controlar una sola tarea. El Telnet servidor que se muestra en la figura representa uno de esos servidores esclavos que pueden existir en el servidor. Este manda caracteres a una entrada del sistema operativo, en la figura marcada como pseudo-terminal, usada por el Telnet servidor para transferir caracteres al sistema operativo como si hubieran sido introducidos por el teclado propio del sistema. También se debe mencionar que cada Telnet esclavo tiene su propia pseudo-terminal.

3.1.2 TERMINAL VIRTUAL DE RED (NVT).

Para hacer que Telnet interopere entre tantos sistemas como sea posible, debe adaptar los detalles de los diferentes tipos de computadoras y sistemas operativos. Algunos sistemas permiten que el usuario pulse una tecla que interrumpe un programa que se está corriendo. Sin embargo, el pulso de teclado empleado para interrumpir un programa varía de sistema a sistema (por ejemplo, algunos sistemas utilizan *Ctrl C*, mientras que otros se valen de *ESC*).

Para adaptar la heterogeneidad, Telnet define cómo deben mandarse las secuencias de datos y comandos a través de Internet. Esta definición se conoce como Terminal Virtual de Red NVT (Network Virtual Terminal). Como se muestra en la figura 3.2, el software cliente traduce las pulsaciones del teclado de la terminal del usuario a formato NVT y las envía al servidor. El software del servidor traduce los datos que acaban de llegar de formato NVT al formato que el sistema requiera. Para devolver los datos, el servidor traduce del formato de una máquina remota a NVT y el cliente local traduce éste formato al formato de la máquina local.

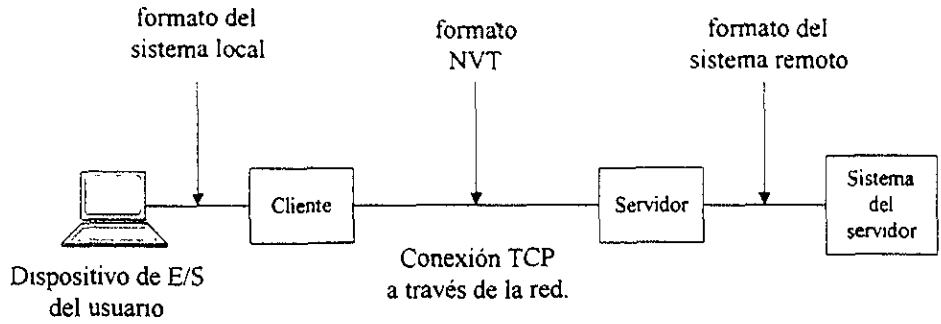

Figura 3.2. Cambio de formato de la información para la comunicación entre el cliente y el servidor.

El servidor puede utilizar un pequeño repertorio de caracteres ASCII de control para manipular la pantalla del cliente. Se muestran en la tabla siguiente los códigos ASCII que se han traducido a números decimales.

Código de control	Significado	Valor decimal	Descripción
NUL	Nulo	0	No hay operación
BEL	Timbre	7	Sonido audible
BS	Retroceso	8	Movimiento a la izquierda de un carácter
HT	Tabulador horizontal	9	Movimiento a la derecha al siguiente tab
LF	Salto de línea	10	Movimiento hacia abajo a la siguiente línea
VT	Tabulador vertical	11	Movimiento hacia abajo al siguiente tab
FF	Salto de página	12	Movimiento a la página siguiente
CR	Retorno de carro	13	Movimiento hacia la izquierda en la línea actual

Figura 3.3. Interpretación NVT para Telnet de los caracteres de control ASCII.

Además de la interpretación de caracteres de control, NVT define la terminación de línea estándar como una secuencia de dos caracteres: *CR-LF*. Cuando un usuario pulsa la tecla que corresponde a fin de línea en la terminal local (por ejemplo *ENTER*), el cliente Telnet debe transformarla en *CR-LF* para su transmisión. El servidor Telnet traduce a *CR-LF* en la secuencia de caracteres apropiada de fin de línea para la máquina remota.

NVT posee las siguientes características:

- Los datos NVT se componen de caracteres ASCII de 7 bits aumentados a 8 bits por medio de un 0 inicial.
- Los datos se envían línea a línea.
- Los bytes cuyo bit inicial (más significativo) es 1 se usan para códigos de comandos.
- NVT es semidúplex. Después de enviar una línea, el cliente espera hasta recibir una línea del servidor. El servidor envía sus datos y, a continuación, un comando "Adelante" (*Go Ahead*), indicando al cliente que ya puede enviar otra línea.

El concepto de Terminal Virtual de Red se utiliza para definir ambos extremos de una conexión Telnet. Cada extremo de la conexión tiene un teclado y una impresora lógicos. La impresora lógica (pantalla del usuario) puede desplegar caracteres, y el teclado lógico puede generar caracteres. La impresora de red es por lo general una pantalla de terminal, en tanto que el teclado lógico es por lo general el teclado del usuario, aunque puede ser algún archivo o cualquier otro flujo de entrada. Estos términos se utilizan también en el Protocolo de Transferencia de Archivos (FTP) y en el Protocolo Simple de Transferencia de Correo (SMTP).

El protocolo Telnet trata ambos extremos de la conexión como si fueran terminales virtuales de red. El concepto de terminales virtuales permite a Telnet interconectarse con cualquier tipo de dispositivo, siempre y cuando haya mapeo disponible de los códigos virtuales al dispositivo físico. Una ventaja de este enfoque es que algunos dispositivos físicos no pueden aceptar todas las operaciones, por lo que la terminal virtual no tendrá dichos códigos. Cuando los dos extremos estén estableciendo la conexión, la carencia de estos códigos se hará notoria y se ignorarán las secuencias que las utilizarían. Este proceso es sencillo: un extremo pregunta si se acepta la función y el otro contesta positiva o negativamente. Si se acepta, se envían los códigos necesarios.

El NVT se encarga de negociar el tipo de terminal a emular para asegurarse de que, básicamente, el teclado y monitor funcionan como el host espera que funcionen. La emulación de terminal más habitual es la VT-100. De modo que, si se emplean programas Telnet, ésta es la emulación más común y segura que puede utilizarse, aunque existen otros emuladores de terminal como el IBM3270 o el HP2648.

La emulación de terminal determina de qué manera el teclado transmite información al computador remoto acerca de cómo aparecerán los datos en la pantalla, además de indicar, entre otras cosas, el comportamiento de determinadas teclas como la de retroceso.

3.1.3 ACCESO A TELNET.

Hay software cliente de Telnet para los principales sistemas operativos, incluidos UNIX, Macintosh y todas las versiones de Windows. Habitualmente se emplea un cliente Telnet sólo con teclear la palabra Telnet, seguida de la dirección Internet del equipo al que se desea acceder. Solamente se puede utilizar el nombre si el sistema tiene algún método para convertir el nombre a su dirección IP. También se puede utilizar un nombre de puerto para conectarse a un servicio. Si no se especifica nombre, dirección o puerto, Telnet introducirá su modo de comando y esperará instrucciones específicas (como se verá más adelante).

Por ejemplo, si se quisiera acceder a una computadora llamada Fed World, que permite acceder a una gran cantidad de información gubernamental de los Estados Unidos, debería escribir: *telnet fedworld.gov*. Un software cliente de Telnet preparado para Windows o Macintosh, recuerda por el usuario los nombres de los host. Además, es posible mantener una agenda de nombres de host, con lo cual es posible visitarlos de nuevo.

Cuando se establezca la conexión, se solicitará identificación de usuario y contraseña. Es posible registrarse con cualquier identificación de usuario que resulte válida para el sistema remoto. A menudo, para registrarse es posible emplear el nombre "*guest*". Algunos sistemas precisan que el usuario proporcione información sobre él, como su nombre y dirección. Otros pueden llegar a instar al usuario a que seleccione un nombre de usuario y una contraseña, que utilizará la siguiente vez que se registre.

3.1.4 PUERTOS Y SOCKETS EN TELNET.

Las computadoras que hacen las funciones de servidor, están preparadas para tener instaladas en ellas más de un software servidor. Para poder diferenciar qué software servidor de los instalados tiene que ejecutarse para cada acceso remoto, a cada software se le asigna un número de puerto diferente El número de puerto no se refiere a ninguna asignación de hardware ni a ningún enlace físico con un puerto de comunicaciones hardware, sino que es un número que identifica a cada una de las aplicaciones software instaladas como servidor en el ordenador.

Esto quiere decir que cuando se accede a una terminal mediante Telnet, no solo habrá que especificar la dirección del ordenador, sino también el número de puerto asignado al software servidor al que pretendemos acceder. Sin embargo, cada aplicación de Internet tiene asignada un número de puerto estándar. Típicamente los números de puerto mayores a 255 se utilizan para el uso privado de la máquina local, pero los números inferiores a 255 se utilizan para procesos de uso frecuente.

En el caso de Telnet, su número estándar es el 23, y por lo tanto, cuando el usuario no le especifica a Telnet su número de puerto, el programa automáticamente se conecta al software servidor identificado con el número de puerto 23.

La forma de indicar con Telnet el número de puerto del software servidor con el cual el usuario de quiere conectar, es escribiendo este número de puerto a continuación de la identificación del ordenador remoto. Es decir:

Nombre del ordenador remoto + puerto

Con este formato podemos decir que con los dos comandos siguientes se obtienen idénticos resultados.

wugate.wustl.edu

wugate.wustl.edu 23

Existen ordenadores que tienen instalado un software servidor para Telnet con un número de puerto distinto a 23. Esto es así cuando un mismo ordenador pretende ofrecer servicios distintos e independientes en cada número de puerto. Por ejemplo, el ordenador culine.colorado.edu ofrece información sobre la liga de basket ball en el puerto 863 y sobre la liga de base ball en el 862.

Así como existe un número de puerto para cada aplicación, cada circuito de comunicación dentro y fuera de la capa TCP se identifica en forma única mediante la combinación de dos números, los cuales en conjunto se conocen como socket. El socket se compone de la dirección IP de la máquina y del número de puerto utilizado por el software TCP. Hay un socket tanto en la máquina emisora como en la receptora.

Debido a que la dirección IP es única a través de toda la red interna y los números de puerto serán únicos para la máquina individual, los números de socket también resultarían únicos en toda la interred. Esto permite que un proceso se comunique con otro a través de la red basándose enteramente en el número de socket.

Si el software TCP desea establecer una sesión Telnet desde su puerto 350, el número de socket se compondrá de la dirección IP de la máquina fuente y del número de puerto, y el mensaje tendrá un número de puerto destino de 23 (número de puerto de Telnet). El TCP receptor tendrá un puerto fuente 23 y el número de puerto destino 350 (el puerto de la máquina emisora). En la siguiente figura se muestra el establecimiento de un circuito virtual mediante números de socket.

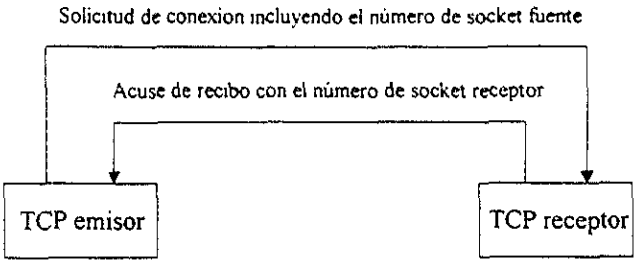

Figura 3.4. Establecimiento de comunicación con números de socket.

Las máquinas emisora y receptora mantienen una tabla de puertos, misma que lista todos los números de puertos activos. Las dos máquinas involucradas tendrán entradas invertidas para cada sesión entre las dos. Esto se conoce como ligadura y se muestra en la figura 3.4. En la tabla de puertos los números fuente y destino están sencillamente invertidos para cada conexión. Naturalmente, las direcciones IP, y por tanto el número de socket serán distintos.

Si la máquina emisora solicita más de una conexión, los números de puerto fuente serán diferentes, aún cuando los números de puerto destino pueden ser iguales. Por ejemplo, si la máquina emisora intentara establecer simultáneamente tres sesiones Telnet, los números de puerto de la máquina fuente serían, por ejemplo, 350, 351 y 352, en tanto que los números de puerto destino serían 23 todos. Lo anterior, se muestra en la figura siguiente:

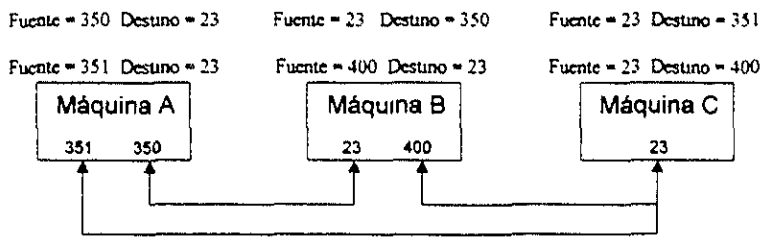

Figura 3.5. Entradas ligadas en las tablas de puerto.

Es posible que más de una máquina compartan el mismo socket destino (proceso conocido como multiplexar). En la figura 3.6, tres máquinas están estableciendo sesiones Telnet con un destino. Todas utilizan el puerto destino 23, por lo que es un puerto multiplexado. Debido a que los datagramas que emergen del puerto tienen información completa del socket (con direcciones IP únicas), no existe confusión respecto a cuál máquina se destina un datagrama.

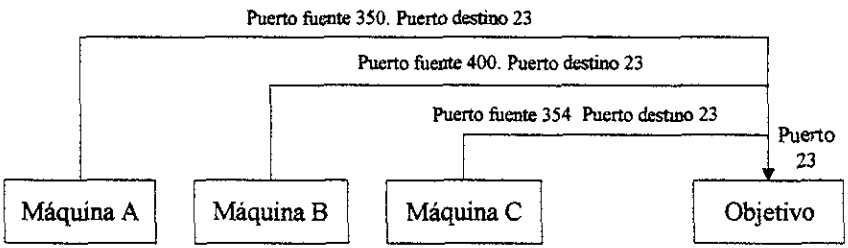

Figura 3.6. Multiplexación de un puerto destino.

Cuando se establecen varios sockets, es concebible que más de una máquina envíe una solicitud de conexión con los mismos puertos fuente y destino, sin embargo, las direcciones IP correspondientes a ambas máquinas serán distintas, por lo que aun así los sockets se identificarán en forma única, a pesar de tener números de puerto fuente y destino idénticos.

3.1.5 CONEXION TELNET.

Una vez que el usuario se conecta al host, una de las primeras cosas que hacen la terminal remota y la computadora del primero es negociar la forma como se comunicarán entre sí y además, deciden el tipo de emulación de terminal que se utilizará.

Durante una sesión Telnet, a medida que se escribe el texto, éste se acumula en un buffer de la computadora del usuario. En el momento que una línea de datos completa está preparada para transmitirse, o cuando se ejecuta un comando para transmitir los datos (como pulsar la tecla *Enter*), los datos se envían por la red desde el teclado NVT. Junto con los datos se envía la dirección IP del host, lo que asegura que el paquete se transmite a la ubicación correcta.

También se envía la dirección IP del usuario, de modo que pueda dirigírsele la información de vuelta. Asimismo, se envían comandos específicos de Telnet para que el otro NVT pueda decidir lo que debe hacer con los datos o cómo responderlos. Por ejemplo, cuando se transmiten datos de un NVT a otro y es necesario enviar determinada información de vuelta al NVT de origen para que pueda continuar un proceso, se ejecuta el comando *Go Ahead* (GA).

El host Telnet recibe los datos enviados por el usuario Luego los procesa y transmite de nuevo a la pantalla del usuario (su "impresora" NVT) los resultados del uso de datos o de ejecutar el comando en una computadora remota. Así, por ejemplo, si se teclean las letras *dir* y se pulsa *<Intro>*, la computadora remota envía de vuelta a la pantalla propia el comando *dir*, así como el resultado de ejecutarlo en la computadora remota.

Debido a que los paquetes deben atravesar un gran número de ruteadores de Internet en cada dirección entre la computadora del usuario y el host, quizá haya un cierto retraso entre el momento que se envía el comando y cuando se ven los resultados en la pantalla.

Cabe resaltar la gran importancia y uso de una conexión Telnet, ya que con este protocolo es posible administrar redes WAN, en donde el administrador es capaz de accesar desde su terminal a cualquier ruteador que conforme su red y cambiar los parámetros de configuración para agregar recursos o solucionar problemas. Una conexión Telnet proporciona un método para validar que el enlace de comunicación entre dos terminales esta trabajando correctamente. Podría pensarse que el programa Ping realiza esta función, sin embargo, este último sólo verifica la accesibilidad a nivel físico del enlace, mientras que al realizar una conexión Telnet estamos validando además que a nivel aplicativo las dos terminales conectadas trabajan correctamente.

3.1.6 MODO DE COMANDO TELNET.

La aplicación Telnet dispone de una serie de comandos que pueden ser utilizados por el usuario, y que en determinados casos le permite particularizar la conexión.

Si al entrar en modo de comando está conectado en una sesión activa, Telnet esperará a que el usuario emita un comando, lo ejecutará, y a continuación retornará

automaticamente a la sesión. El modo de .comando permite introducir comandos relacionados con el cliente (la máquina enfrente de la cual está el usuario que solicitó el servicio) en vez del servidor.

Una vez que se establezca con éxito la conexión, su sesión se comportará como si el usuario estuviera en la máquina remota, con todos los comandos válidos de dicho sistema operativo. Todas las instrucciones estarán relativas al servidor, por lo que un comando de directorio mostrará el directorio de trabajo del servidor, no del cliente.

Para terminar la sesión remota, simplemente debe emitirse el comando de registro de salida.

Para entrar al modo de comando de Telnet, se debe teclear *TELNET*, sin ningún nombre de ordenador remoto a continuación. También se puede acceder al modo de comando si una vez establecida la conexión se pulsa la tecla correspondiente al comando escape. Este comando suele ser generalmente la combinación de teclas *Ctrl]*.

Independientemente de cómo lleguemos al modo de comando, el indicativo que veremos será *TELNET>*, y una vez en este punto el usuario podrá utilizar cualquiera de los comandos siguientes:

- **TELNET** [dirección del servidor remoto].- Conecta al usuario con la terminal remota cuyo nombre le especifique. Si tenemos establecida una conexión con otro ordenador, antes de iniciar la nueva sesión, se debe cerrar la actual con el comando CLOSE.
- **OPEN**.- Permite al usuario conectarse a una computadora en caso de que al ejecutar el comando TELNET, no se haya escrito la dirección del servidor remoto.
- **CLOSE**.- Da por terminada la conexión activa con el ordenador remoto.

CAPITULO IV

TRANSFERENCIA DE ARCHIVOS.

Una de las funciones con las que debe cumplir todo tipo de red es la de proporcionar medios para compartir archivos y para ello se han desarrollado varios métodos. Entre ellos, están los de utilizar un computador central, llamado servidor de archivos, en donde son almacenados todos los archivos. A esta computadora, se le pueden conectar terminales (incluso aquellas que no disponen de medios de almacenamiento como un disco duro), y por medio de estas terminales se puede acceder a la información almacenada en el servidor. También existe la opción de tener computadoras con sistema de almacenamiento local, en este caso periódicamente se mandan los archivos a un sistema de almacenamiento central, para que los archivos sigan disponibles para los demás usuarios.

Podemos decir entonces que es posible compartir archivos en dos formas distintas: *copiado de archivo completo* y *acceso en línea.* El copiado de archivo completo significa que, cada vez que un programa quiera acceder a un archivo, éste obtendrá una copia local. El copiado a menudo se utiliza para datos de sólo lectura, pero si el archivo debe modificarse, el programa hace los cambios en la copia local y transfiere de regreso el

archivo modificado a la localidad original. El acceso en línea significa que se permite a varios programas acceder de manera concurrente a un solo archivo. Los cambios que se realizan al archivo se efectúan inmediatamente y están disponibles a todos los programas que accedan al archivo.

En ambos casos, el sistema operativo proporciona acceso a archivos remotos compartidos de la misma manera que proporciona acceso a archivos locales. Se dice que el archivo remoto está integrado con los archivos locales y que todo el sistema de archivos proporciona acceso transparente a los archivos compartidos.

La familia de protocolos TCP/IP ofrece entre sus aplicaciones el protocolo FTP para la transferencia de archivos, mediante el método de copiado de archivo completo. Este protocolo realiza conexiones TCP y permite que los usuarios listen directorios en la máquina remota, así como transferir archivos en cualquier dirección (del servidor de archivos al usuario y viceversa). Además incluye un control de autenticación, para autorizar la transferencia de archivos únicamente a uno o varios grupos de usuarios mediante una clave de acceso.

Otra alternativa es utilizar un protocolo más simple: el TFTP. Este realiza conexiones mediante el protocolo UDP, no proporciona el servicio de autenticación y se emplea en aquellas aplicaciones que sólo necesitan de la transferencia de archivos en forma unidireccional (del servidor de archivos al usuario).

Una tercera opción es el protocolo NFS, que a diferencia del FTP y TFTP proporciona acceso en línea a archivos compartidos. Utiliza el UDP para el transporte de información y es utilizado para acceder a archivos ya sea del sistema local o de algún sistema remoto, sin que el usuario se entere en dónde está realmente el archivo. Cuando un programa requiere abrir un archivo, el sistema transfiere el control ya sea hacia el sistema

de archivos local o hacia el cliente NFS, dependiendo si el archivo se encuentra en el disco local o en un servidor de archivos, haciendo el servicio completamente transparente.

4.1 PROTOCOLO DE TRANSFERENCIA DE ARCHIVOS (FTP).

4.1.1 CARACTERISTICAS DEL FTP.

El protocolo FTP (File Transport Protocol) provee los elementos básicos de la capa de aplicación para la transferencia de archivos en una red de trabajo mediante el modelo cliente–servidor. Utilizando FTP, un usuario de la red puede cargar archivos (*upload*) de su computador a otro, o descargar archivos (*download*) de un determinado computador al suyo.

Este protocolo es una utilería para la administración de archivos a través de máquinas, sin tener que establecer una sesión remota mediante Telnet.

Es posible administrar un sistema de archivos dentro de una red mediante FTP, mediante una computadora central llamado servidor FTP, al cual se le pueden conectar terminales, de manera que todos los usuarios pueden tener acceso a él, y los cambios que sean realizados a un archivo quedan registrados inmediatamente, aunque únicamente un usuario puede utilizar este archivo, y estará disponible para los demás usuarios cuando sea liberado. Otra opción consiste en mandar una copia de un archivo al usuario que lo solicite. En este caso es posible consultar y editar el archivo (si está autorizado para ello), y en caso de realizar modificaciones el usuario debe regresar el archivo al servidor FTP, permitiendo así que varios usuarios utilicen el mismo archivo simultáneamente.

FTP puede hacer una copia de un archivo, pero no puede mover un archivo de un sistema a otro. Esto es, después de terminar una transferencia de FTP, existen ahora dos copias exactas del archivo, una de las cuales permanece en la fuente del sistema y la otra

en el sistema destino. El usuario de FTP debe borrar el archivo en la fuente del sistema después de la transferencia, pero esto requiere de pasos adicionales y de permisos.

Con este protocolo se puede copiar un archivo completo. No puede copiarse sólo una parte de un archivo, así que si el archivo es grande, y el usuario sólo desea copiar una pequeña parte del archivo, FTP no se usaría en este caso.

FTP utiliza el protocolo TCP, eliminando así la necesidad de preocuparse sobre la confiabilidad o la administración de la conexión, ya que TCP realiza adecuadamente esas funciones. Sin embargo, los detalles de autorización, el nombre y la representación entre diferentes máquinas hace que el protocolo sea más complejo. Además el FTP ofrece muchas facilidades que van más allá de la función de transferencia misma. Los servicios que ofrece FTP son:

- **Acceso interactivo**. Aunque el FTP está diseñado para usarse mediante programas, la mayor parte de las implantaciones proporciona una interfaz interactiva que permite a las personas interactuar fácilmente con los servidores remotos.
- **Especificación de formato** (representación). El FTP permite al cliente especificar el tipo y formato de los datos almacenados. Por ejemplo, el usuario puede especificar si un archivo contiene información de texto o binaria, así como, los archivos de texto utilizan los conjuntos de caracteres ASCII o EBCDIC.
- **Control de autenticación**. El FTP requiere que los clientes se autoricen a sí mismos con el envío de un nombre de conexión y una clave de acceso al servidor antes de pedir la transferencia de archivos. El servidor rechaza el acceso a clientes que no puedan abastecer una conexión o una clave de acceso válida.

4.1.2 PUERTOS FTP.

Para establecer la comunicación entre cliente y servidor, este protocolo utiliza dos canales TCP. El puerto 20 TCP, es el canal de datos, en tanto que el puerto 21 es el canal de comandos. FTP difiere de la mayor parte de otros programas de transferencia de archivos en que utiliza dos canales, permitiendo transferencias simultáneas de comandos FTP y de datos. Los dos canales que existen entre ambas máquinas se conocen como intérprete de protocolo PI (Protocol Interpreter) y el proceso de transferencia de datos DTP (Data Transfer Process). PI transfiere las instrucciones entre las dos implementaciones mediante el canal de datos 21 de TCP, y DTP transfiere archivos sobre el canal de datos 20 de TCP. La figura 4.1 muestra las conexiones de estos dos canales.

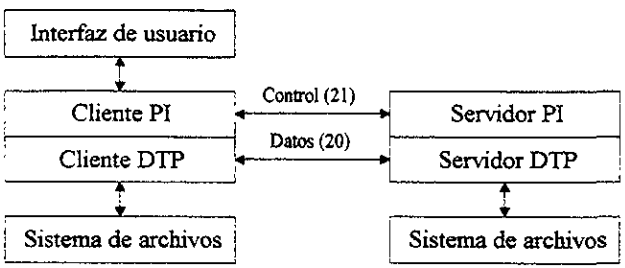

Figura 4.1. Conexión de los canales FTP.

Su funcionamiento es muy simple, el "Cliente PI" inicia el control de la conexión, que sigue el protocolo Telnet. El usuario a través de una interfaz transmitirá sus peticiones al cliente PI que las convertirá en comandos FTP estándar y se los enviará al "Servidor PI". Asimismo, las respuestas serán enviadas por el servidor PI al cliente PI que se las mostrará finalmente al usuario.

Los comandos FTP especifican los parámetros para la conexión de datos como. puerto de datos, modo de transferencia, tipo de representación y la naturaleza de las operaciones sobre el sistema de ficheros (como almacenar o borrar).

El FTP usa el protocolo Telnet en la conexión de control, es decir, que el servidor y el cliente PI estarán implementados siguiendo las reglas de dicho protocolo, es por eso que en la figura 1.5, el protocolo FTP se representa como si dependiera de Telnet. Esto puede ser aplicado de dos maneras: primero, el cliente o servidor PI, pueden adoptar directamente las características del protocolo Telnet en sus procedimientos, logrando en este caso eficiencia e independencia. Segundo, el cliente o servidor PI pueden hacer uso del módulo Telnet existente en el sistema, obteniendo así facilidad de implementación y programación modular.

4.1.3 CONEXION FTP.

Por lo general FTP se inicia con el nombre o con la dirección de la máquina destino, una identificación log-in o ID asignada por el administrador del sistema y un password. Una vez conectado al servidor FTP remoto, el usuario puede transferir archivos como texto en ASCII o archivos binarios. Igual que para Telnet, deberá ser posible convertir el nombre de un host en una dirección IP para que tenga éxito el comando. La máquina destino también se puede especificar desde la línea de comandos de FTP.

Después de registrarse, en realidad el usuario no está en la máquina remota. Lógicamente sigue en el cliente, por lo que todas las instrucciones referentes a transferencias de archivos y a movimientos de directorios deben relacionarse con su

máquina local y no con la remota. Nótese que esto es lo opuesto a Telnet (una diferencia que provoca confusión a los nuevos usuarios de FTP y Telnet)

El proceso que sigue FTP cuando se establece una conexión es el siguiente:

1. - **Registro de entrada**: Verifica identificación y contraseña del usuario.
2. - **Define directorio**: Identifica el directorio de inicio
3. - **Define el modo de transferencia de archivo:** Define el tipo de transferencia.
4. - **Inicia transferencia de datos**: Habilita los comandos de usuario.
5. - **Detiene la transferencia de datos**: Cierra la conexión.

Estos pasos se ejecutan en orden para cada conexión. Para controlar FTP, un usuario tiene cierto número de comandos a su disposición; los comandos utilizados con más frecuencia se resumen en el siguiente tema.

4.1.4 COMANDOS FTP.

Actualmente muchas personas disponen de Interfaces gráficas de usuario GUI (Graphical User Interfaces) para la transferencia de archivos, los cuales facilitan y transparentan los procedimientos de FTP. Sin embargo, es necesario conocer los comandos FTP porque de no contar con una GUI, el usuario puede invocar a FTP desde una interfaz de texto (como puede ser una ventana DOS en Windows), desde la cual es posible explotar los recursos de este protocolo con la ayuda de comandos. La conexión FTP se realiza tecleando ftp seguido del nombre o la dirección IP del host del que se quiere hacer una transferencia. Una vez hecha la conexión se pueden utilizar estos comandos:

- !.- Nos permite dejar momentáneamente el prompt del programa FTP, para poder realizar alguna otra operación.
- **cd.**- Cambia de directorio.
- **bye.**- Cuando la sesión de FTP ha llegado a su fin, es decir hemos transferido todos los archivos que necesitamos, se da el comando *bye* con el cual se cierra la conexión y nos saca del programa.
- **close.**- Cierra la conexión con ese servidor pero no nos saca del prompt del ftp, permitiéndonos volver a abrir otra conexión si así lo deseamos.
- **get.**- Con este comando, seguido del nombre del archivo que deseamos obtener del servidor FTP, lo transferimos a nuestro equipo.
- **put.**- Permite copiar un archivo de una computadora al servidor.
- **hash.**- Si deseamos ver el avance en la transferencia debemos poner este comando, lo cual mandara un símbolo # por cada 8kb transferidos.
- **ls.**- De igual manera que en UNIX, este comando despliega el contenido del directorio del servidor de FTP.
- **mget.**- Cuando se requiere transferir muchos archivos, del servidor a nuestra maquina, podemos hacer uso del comando *mget* y de la utilización de los comodines, * y ?, funcionan de igual manera que en DOS.
- **mput.**- Si deseamos transferir muchos archivos de una máquina al servidor, utilizamos el comando *mput* y los comodines, de igual manera que en el comando *mget*.
- **open.**- Cuando se cierra la conexión con el comando *close*, y se desea abrir una nueva a otro lugar o incluso al mismo, lo que se hace es dar el comando *open* seguido por la dirección del servidor ya sea con dirección IP o por nombre.

- **pwd**.- Este comando despliega la ruta donde nos encontramos en el servidor de FTP.

FTP debe hacer compatibles estos comandos y permitir la manipulación de archivos en diferentes sistemas operativos. Un ejemplo de esto es el siguiente. Si un usuario se conecta a una computadora remota, su primera petición sería pedir una lista de los archivos que contiene el directorio al cual se ha accedido. En la siguiente tabla se muestran los diferentes comandos para solicitar un directorio, dependiendo el sistema operativo.

SISTEMA OPERATIVO	COMANDO DIRECTORIO
DOS	dir
Windows 3.x	Click sobre el folder
UNIX	ls

Figura 4.2. Diferencia del comando directorio según el sistema operativo.

Cuando se tiene una comunicación entre computadoras y sistemas operativos diferentes, el comando de listado de directorio requiere de una traducción, así como también otras funciones como cambio de directorio, acceso, creación de directorio, eliminación de archivo, etc.

Para poder garantizar la transferencia de archivos, FTP utiliza un conjunto de comandos que se ejecutan entre máquinas, logrando así, que sin tomar en consideración el sistema operativo, el comando que es transmitido sobre la red es siempre el mismo.

Los comandos del protocolo FTP son secuencias ASCII de cuatro caracteres, terminados por un carácter de nueva linea. Algunos de los códigos requieren parámetros después de ellos. Una ventaja primordial en el uso de los caracteres ASCII para comandos, es que un usuario puede observar el flujo de comandos y comprenderlos fácilmente. Estos comandos se pueden dividir en tres grupos:

1. - Comandos de Control de Acceso.
2. - Comandos de Parámetros de transferencia.
3. - Comandos de Servicio FTP.

Comandos de Control de Acceso.- Estos comandos definen el acceso de un usuario de un host remoto. Los más importantes se muestran en la siguiente tabla:

COMANDO	CODIGO	FUNCION
Nombre de usuario	USER	Especifica al usuario que solicita el servicio FTP.
Password	PASS	Especifica el password del usuario
Cuenta	ACCT	Identifica la cuenta del usuario
Código de directorio de trabajo	CWD	Cambia de directorio de trabajo.
Reinicialización	REIN	Termina y reinicia una conexión
Salida	QUIT	Termina una conexión cerrando el control de la misma.

Figura 4.3. Comandos de Control de Acceso.

Comandos de Parámetro de Transferencia.- Estos comandos de parámetros de transferencia de datos tienen valores preasignados, de tal forma que estos comandos solamente son requeridos cuando este valor preasignado es cambiado. Estos comandos establecen el formato de los datos, la estructura de archivos y el modo de transmisión que se utilizará al copiar archivos, por ejemplo, en UNIX, para poder transmitir un archivo binario (archivos de imagen o ejecutables) es necesario cambiar el modo de transferencia a este para que los datos no se corrompan. y se especifican en la tabla:

COMANDO	CODIGO	FUNCION
Puerto de datos	PORT	Especifica el puerto de datos que ha de ser utilizado para la conexión de datos. Este como otros comandos tiene un valor preasignado
Modo de transferencia de archivo	MODE	Especifica el modo de transferencia de datos mediante un caracter Telnet: S - stream (serie) B - block (bloque) C - compressed (comprimida)

Figura 4.4. Comandos de Parámetros de Transferencia.

Comandos de Servicio.- Los comandos de servicio FTP definen la transferencia y administración de archivos requeridos por el usuario. Los comandos pueden ir en cualquier orden pero hay algunos que deben seguir ciertas normas por ejemplo, el comando *RNFR* debe ir seguido por el comando *RNTO*. A continuación se muestran en la tabla algunos de estos comandos.

COMANDO	CODIGO	FUNCION
Extraer	RETR	Transfiere una copia del archivo especificando al servidor localizado al final de la conexion de datos
Almacenar	STOR	Transfiere datos del cliente a través de la conexión y los almacena como un archivo en el servidor
Añadir	APPE	Provoca que el cliente acepte que los datos sean almacenados en un archivo del servidor. Si este archivo ya existe en el servidor, los datos serán añadidos en ese archivo
Abortar	ABOR	Cancela el comando de servicio FTP previo y cualquier transferencia de datos asociada.
Renombrar de	RNFR	Especifica el nombre del archivo que será renombrado.
Renombrar a	RNTO	Especifica el nuevo nombre del archivo señalado en RNFR
Borrar	DELE	Provoca que el archivo especificado sea borrado.
Listado	LIST	Envía un listado del servidor al cliente
Sistema	SYST	Identifica el sistema operativo del servidor.
Ayuda	HELP	Envía información de ayuda en alguna implementación.

Figura 4.5. Comandos de Servicio.

4.1.5 FTP ANONIMO.

Un problema que tiene el protocolo FTP es que no se puede acceder a la computadora remota a no ser que se tenga la autorización necesaria, es decir, que no se pueden copiar archivos si no se está dado de alta como usuario de dicha máquina.

Este problema se soluciona con FTP Anónimo, que permite establecer una sesión FTP sin necesidad de estar registrado como usuario.

El acceso a FTP Anónimo significa que el cliente no necesita una cuenta, sino especificar una clave de acceso de invitado. El servidor permite que el usuario anónimo se conecte pero restringe su acceso únicamente a los archivos públicos disponibles.

Para poder dar este servicio, el administrador del sistema configura una cuenta especial denominada *anonymous*, que puede utilizar cualquier usuario de Internet, y define una serie de directorios de acceso público. Para evitar el acceso indebido a los archivos de uso local de un computador remoto, los archivos disponibles vía FTP Anónimo son almacenados en áreas separadas de aquellos. De esta forma, es completamente seguro para la organización proporcionar acceso público a los usuarios externos. Otra medida de seguridad es que sólo se permite la copia de archivos del servidor al cliente y no en caso contrario.

Para un FTP Anónimo se introduce como nombre de usuario la palabra *anonymous* y el password es generalmente la dirección de correo electrónico del usuario, aunque algunas veces también se permite un password vacío. Una vez iniciada la sesión es posible utilizar los comandos de FTP descritos anteriormente.

4.2 PROTOCOLO TRIVIAL DE TRANSFERENCIA DE ARCHIVOS (TFTP).

4.2.1 CARACTERISTICAS DE TFTP.

El conjunto TCP/IP contiene un segundo protocolo de transferencia de archivos que proporciona un servicio económico y poco sofisticado, utilizado por ejemplo para enviar los archivos de configuración al arrancar un ruteador o una estación de trabajo sin disco. Este servicio lo ofrece el Protocolo TFTP (Trivial File Transfer Protocol) y se diseñó para aplicaciones que no necesitan interacciones complejas entre cliente y servidor. Consiste fundamentalmente en la lectura o escritura por parte de un cliente a un archivo de un servidor. Como es más restrictivo el software TFTP resulta más pequeño que el FTP.

El tamaño reducido es importante para muchas aplicaciones. Por ejemplo, los fabricantes de dispositivos sin disco pueden codificar al TFTP en la memoria de sólo lectura (ROM) y usarlo para obtener una imagen de memoria inicial cuando se encienda la máquina. La ventaja de utilizar el TFTP es que permite al código de arranque emplear los mismos protocolos TCP/IP subyacentes que el sistema operativo utiliza una vez que empieza la ejecución. De este modo es posible para una computadora arrancar desde un servidor en otra red física

Este protocolo difiere de FTP en dos formas primordiales: no hay registro de entrada en la máquina remota y utiliza el protocolo UDP para encapsular su información, el cual es un protocolo de transporte sin conexión, en vez de TCP. Al utilizar el UDP, TFTP no vigila el progreso de la transferencia de archivos, aunque para asegurar la integridad adecuada de los datos tiene que emplea ciertos algoritmos.

TFTP tiene estas características:

- Envía bloques de 512 octetos (excepto el último).
- Añade un encabezado de 4 octetos a cada bloque de datos.
- Numera los bloques empezando por 1.
- Admite transferencias de octetos ASCII o binarios.
- Se puede utilizar para leer o escribir un archivo remoto.
- No contempla la autenticación del usuario.
- Utiliza los mismos comandos que FTP.

4.2.2 FUNCIONAMIENTO DE TFTP.

Este protocolo utiliza los campos de puerto fuente y destino de UDP para establecer los dos extremos de la conexión. Esto lo realiza mediante identificadores de transferencia TFTP (TID's), creados por TFTP y pasados a UDP, el cual los coloca en los encabezados.

Igual que Telnet y FTP, TFTP utiliza la ligadura de puertos, que es cuando la máquina emisora selecciona un TID y la remota se establece al número de socket 69 (el número de puerto de TFTP). La máquina remota responderá con un acuse de recibo a la solicitud de conexión, un socket fuente 69, y el TID destino enviado a la solicitud.

Las reglas TFTP sin sencillas. El primer paquete enviado solicita una transferencia de archivo y establece la interacción entre cliente y servidor; el paquete especifica el nombre de archivo y si el archivo se leerá (transferido al cliente) o escrito (transfendo al servidor). Los bloques del archivo están numerados en forma consecutiva comenzando con 1. Cada paquete de datos contiene un encabezado que especifica el número de bloque que

se transporta Y cada acuse de recibo contiene el número de bloque el que se está recibiendo. Un bloque de menos de 512 octetos señala el final del archivo.

Es posible enviar un mensaje de error en lugar de los datos o del acuse de recibo. Los errores también terminan con la transferencia. Un error se indicará enviando un paquete de error. Este paquete ni se asiente ni se retransmite, así que el otro extremo de la comunicación puede no recibirlo nunca.

A continuación se muestra el formato de los cinco tipos de paquetes TFTP:

CODIGO	DESCRIPCION
RRQ	Solicitud de lectura
WRQ	Solicitud de escritura
DATA	Enviar datos
ACK	Acuse de recibo
ERROR	Error

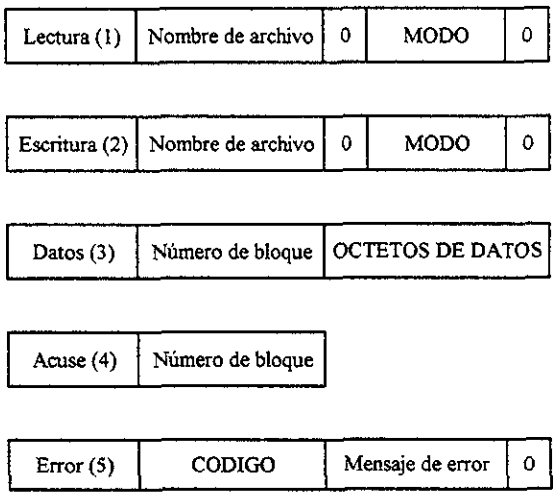

Figura 4.6. Diseños de paquetes TFTP. Los campos no se muestran a escala porque algunos son de longitud variable.

El paquete inicial se envía al socket 69 del servidor y debe utilizar códigos de operación 1 ó 2, dependiendo si se trata de una petición de lectura (Read Request) o una

petición de escritura (Write Request). Una vez que se ha hecho una petición de escritura o lectura, el servidor selecciona un socket para el resto de la transferencia de archivos.

Los mensajes de error que pueden presentarse se listan a continuación:

CODIGO	DESCRIPCION
0	No definido
1	No se encuentra el archivo.
2	Los permisos impiden el acceso.
3	Disco lleno o límite de asignación excedido
4	Operación TFTP ilegal solicitada.
5	Número de transferencia desconocido.

Figura 4.7 Mensajes y códigos de error TFTP.

En todos los paquetes el último bloque contiene entre 0 y 511 bytes de datos, que se han mandado como 0 en la figura 4.6. Esto sirve de relleno para que el bloque de datos llegue a 512 bytes.

Debido a que no existe conexión directa entre cliente y servidor, el cliente establece un temporizador y espera respuesta del servidor. Si ésta no llega antes de que el temporizador expire, se envía otra solicitud. Después de recibir un ACK, se transmite un paquete DATA, respecto al cual se recibe otro ACK o un ERROR. Si hay varios paquetes por transferir, se organizan de forma que tengan una longitud de 512 bytes y un número de secuencia creciente. El proceso termina cuando el servidor recibe un paquete DATA con una longitud menor de 512 bytes.

4.3 SISTEMA DE ARCHIVOS DE RED (NFS)

4.3.1 DESCRIPCION DE NFS.

El Sistema de Archivos en Red NFS (Network File System) es un sistema de operación de red que provee conexiones transparentes entre redes y computadoras, archivos y directorios de computadoras en forma local. Es decir, NFS permite que se creen enlaces a directorios remotos en los sistemas de archivos locales y se usen los archivos remotos como si estuviesen locales. Existen procedimientos NFS que permiten que un cliente acceda, lea y escriba en archivos. El cliente puede hacer averiguaciones sobre la organización y la capacidad del sistema remoto de archivos y puede pedir que le muestren los atributos de archivos individuales.

El objetivo de este protocolo es integrar a las estaciones de trabajo en las redes de área local, así como simplificar el acceso a archivos remotos y verificar el comportamiento de periféricos.

NFS permite que una aplicación lea y escriba archivos que residan en servidores que contengan este protocolo, siendo el acceso al servidor totalmente transparente para la aplicación y para el usuario. Este acceso transparente a la estructura de archivo de otras máquinas se consigue al agregar lógicamente al cliente, el servidor NFS. A este proceso se le conoce como montaje. El directorio en el cual ocurre el montaje se le conoce como punto de monta.

Algunos fabricantes incluyen software de cliente o de servidor de NFS en sus productos para TCP/IP, mientras que otros venden NFS como una opción disponible por una cantidad adicional, incluyendo una biblioteca para programación de RPC (sobre la cual

se construye NFS). Los productos de TCP/IP para Windows ofrecen una opción que permite a un sistema Windows ser cliente o servidor de NFS, en donde un directorio remoto NFS es montado como un dispositivo local, por ejemplo una unidad de disco E: .

Este protocolo se construyó sobre la Llamada a Procedimiento Remoto RPC (Remote Procedure Call), y la Representación Externa de Datos XDR (eXternal Data Representation), como puede verse en la figura siguiente:

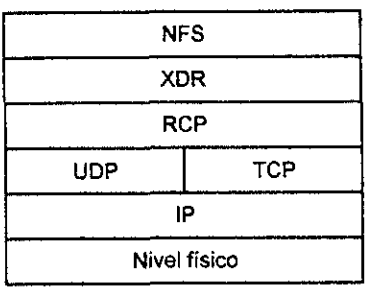

Figura 4.8. Arquitectura del protocolo NFS.

- **RPC**.- La Llamada de Procedimiento Remoto es una estructura diseñada para permitir el desarrollo de aplicaciones generales cliente / servidor.
- **XDR**.- La Representación de Datos Externa consiste en un lenguaje de definición de tipos de datos de codificación de los mismos con un formato normalizado. Ello permite que tipos diferentes de computadoras, como un host Unix, PC, Machintosh y grandes computadoras de IBM puedan intercambiar datos
- **NFS**.- El Sistema de Archivo de Red es una interfaz de la capa de aplicación para la transferencia de archivos, acceso y administración.

4.3.2 LLAMADA DE PROCEDIMIENTO REMOTO (RPC).

Algunas aplicaciones cliente/servidor se construyen sobre RPC Todo comienza con una petición de llamada del cliente al servidor; quien después de recibir el mensaje y extraer la solicitud, el servidor ejecuta el procedimiento y ensambla el mensaje de respuesta con los resultados. Al recibir respuesta, el cliente desensambla el mensaje y continúa con su aplicación normal.

Los componentes de la estructura de RPC se pueden describir como sigue:

1.- Uno o varios programas que se ejecutan en un servidor implementan un servicio de RPC. Por ejemplo, hay diferentes programas para implementar los servicios de acceso a archivos y bloqueo de archivos.

2.- Cada programa puede ejecutar varios procedimientos. La idea es que un procedimiento debe realizar una sola función bien definida. Por ejemplo, existen procedimientos diferentes de acceso a archivos de NFS para leer, escribir, cambiar el nombre y borrar.

3.- Cada programa tiene un identificador numérico asignado.

4.- Cada procedimiento de un programa también tiene un identificador numérico asignado.

Los ID numéricos de los procedimientos los asignan los diseñadores de los programas. Por ejemplo, *read* (leer) es el procedimiento 6 y *rename* (cambiar nombre) es el procedimiento 11 de NFS. Con el tiempo los procedimientos cambian, ya que son mejorados y se añaden otros nuevos. Por este motivo, una llamada RPC debe identificar la versión del programa. No es inusual encontrar varias versiones de un programa de RPC ejecutándose en un host remoto.

Los formatos de las solicitudes y respuestas de las llamadas RPC las podemos ver a continuación:

Número ID de la transacción
Indicador de dirección de envio. Llamada (0)
Número de versión de RPC y del programa
Número de programa
Número de procedimiento
Información de la autorización
Verificación de la autorización
Parámetros de la llamada de procedimiento

A

Número ID de la transaccion
Indicador de dirección de envio Respuesta (1)
Repite el Status
Verificación de la autorización
Aceptación de estatus
Resultados específicos del procedimiento

B

Figura 4.9. (A) Encabezado de solicitud RCP. (B) Encabezado de respuesta RCP.

Para hacer corresponder cada respuesta con su llamada se necesita un identificador de transacción. El indicador de dirección muestra si el mensaje se originó en el cliente o en el servidor. El número de programa identifica el tipo de servicio y el número de procedimiento identifica el procedimiento particular en el servicio.

El cliente puede necesitar algún método para identificarse a sí mismo por medio de alguna credencial que demuestre su derecho a invocar el servicio. Por lo que el encabezado RPC utiliza el campo de información de autorización. Cabe mencionar, que no existe un estándar único para la autenticación. Cada diseñador debe decidir las necesidades de sus programas. Sin embargo, existe un esfuerzo creciente para conseguir estándares en esta área.

Además, la llamada del cliente llevará los parámetros de entrada. Por ejemplo, una llamada *read* de NFS indica el archivo y el número de bytes que se quieren leer.

Por último, el servidor debe informar al cliente el resultado de las llamadas que tuvieron éxito, y dar a conocer cuáles de sus peticiones fueron rechazadas y el motivo, mediante una respuesta RCP. Se puede rechazar una solicitud debido a que las versiones no se corresponden o a fallos en la autenticación del cliente. El servidor tiene que informar de los errores debidos a un parámetro incorrecto o a fallos como "*imposible encontrar el archivo*".

4.3.3 REPRESENTACION DE DATOS EXTERNOS (XDR).

XDR es un método con el cual se codifican los datos de un mensaje RCP. No existe un encabezado formal para XDR. Define información en múltiplos de cuatro bytes (32 bits). Si el valor a ser representado no se adapta exactamente en un múltiplo de cuatro bytes, el valor se rellena con ceros.

XDR se emplea para asegurar la compatibilidad de los datos entre un sistema y otro. Habilita ambos extremos para que conviertan su representación de datos local en un formato común, eliminando cualquier duda sobre el significado de datos.

El formato XDR utiliza bits secuenciales dentro de un buffer, entregándoles un formato en un mensaje por medio del cual se envían a las capas superiores. A fin de simplificar el manejo de los datos de formato XDR se ha desarrollado un lenguaje especial, similar a *C*, llamado XDR, el cual se utiliza dentro de otros lenguajes de programación.

XDR es una definición de cómo la información debería estar formada antes de que se transmita por la red. Un formato de información de clientes NFS basado en XDR,

traducirá la información en forma entendible a los sistemas operativos de la computadora local.

4.3.4 FUNCIONAMIENTO Y CARACTERISTICAS DE NFS.

La siguiente figura muestra el orden en el que se realiza una petición NFS a un servidor y la forma en que éste responde.

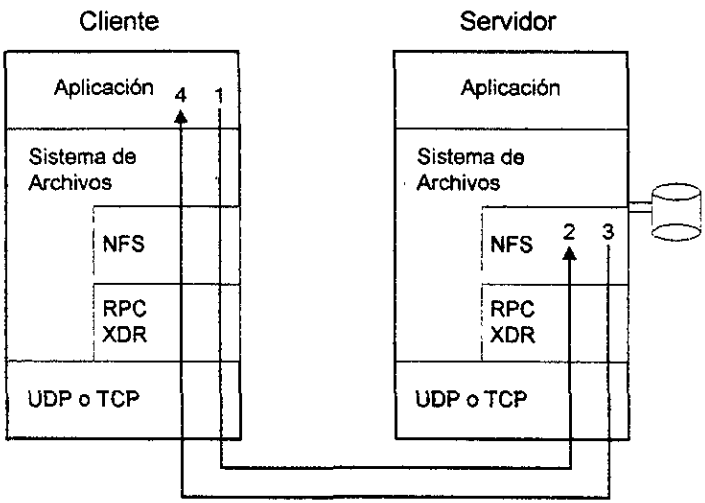

Figura 4.10. Funcionamiento de NFS.

Por ejemplo, en el punto 1 el usuario realiza una petición para leer un archivo remoto desde su aplicación, la cual es procesada por RPC y XDR en el lado del cliente

indicando la acción que se desea hacer con el archivo remoto En le paso 2 el servidor recibe la petición verificando antes que el cliente esté autorizado para realizar esta operación. Una vez aprobada la operación el servidor envía la información solicitada (paso 3), a la aplicación del cliente que lo solicitó (punto 4). En ambos lados la información debe ser procesada por XDR y RPC para garantizar que la información sea comprensible en las dos terminales, tomando en cuenta que los datos serán transmitidos a través de diferente hardware, sistemas operativos, protocolos de transporte y tecnologías de red

Un host cliente se prepara para usar NFS montando un subárbol de directorios remoto en su propio sistema de archivos. El cliente lo hace enviando una petición de RPC al programa *mount* del servidor. El programa mount tiene muchos parámetros opcionales. Los más importantes deciden:

- Si el directorio se debe montar sólo de lectura o de escritura.
- Si se debe intentar de nuevo los montajes fallidos y el límite en el número de intentos.
- Si el usuario puede interrumpir una llamada que tarda mucho en completarse.

Algunos archivos los comparten muchos usuarios. Un cliente que necesite actualizar un archivo compartido querrá obtener acceso exclusivo al archivo, es decir, bloquear el archivo durante el proceso de actualización.

El bloqueo de archivos en un entorno NFS se lleva acabo por medio de dos servicios RCP: el *lock manager* (administrador de bloqueos) y el programa *status*. El administrador de bloqueos maneja las peticiones de bloqueo de los clientes. El monitor de

status del servidor intenta seguir el rastro de los host cliente que realizan bloqueos en un momento dado.

El usuario final o la aplicación no tienen conciencia de la existencia de NFS Cuando se hace una llamada para realizar una operación con una archivo, como leer, escribir, copiar, cambiar el nombre, etc., que se encuentra en una computadora remota, el sistema operativo redirige la petición a NFS.

Por convención, NFS toma el puerto 2049 al inicializarse. Normalmente se implementa sobre un transporte UDP, pero en algunas ocasiones también se emplean conexiones TCP. UDP trabaja bien cuando el cliente y el servidor residen en la misma red LAN. Se debería utilizar TCP en comunicaciones a través de una red WAN, donde es necesario realizar cálculos de plazos de retransmisión y de recuperación de congestiones.

NFS se diseñó con el sistema de archivos Unix en mente, por lo que deben hacerse ciertas consideraciones al utilizar DOS. Por ejemplo, si un cliente de DOS quiere leer una archivo de texto creado por un usuario UNIX, aparecen algunos problemas. En primer lugar, los nombres de archivos de DOS se componen de ocho caracteres seguidos, opcionalmente, por un punto y hasta tres caracteres más. Cuando un usuario de DOS teclea un nombre de archivo, todas las letras se convierten a mayúsculas. Por ejemplo, *COMMAND.COM* es un nombre de archivo DOS. Los nombres de Unix pueden ser mucho más largos y contener una mezcla de letras mayúsculas y minúsculas. Por ejemplo, *unNombreLargo.deArchivo* es un nombre válido para Unix.

Para solucionar este problema, los fabricantes realizan traducciones automáticas de nombres e incluyen también una utilidad para ver el nombre del archivo original.

CAPITULO V

CORREO ELECTRONICO Y ADMINISTRACION DE REDES.

El correo electrónico es el servicio de la capa de aplicación más utilizado en redes de computadoras, incluso muchos usuarios comienzan a usar las redes al emplear este servicio, ya que prefieren usarlo para transferir archivos en lugar de los protocolos de transferencia de archivos.

Para el envío y recepción de correo es necesario definir direcciones que identifiquen al emisor y al receptor del mensaje. Estas direcciones de correo electrónico tienen en general el formato siguiente:

Parte-local@nombre-dominio

Donde la parte-local es la dirección del buzón al cual se desea enviar el mensaje. El nombre-dominio es la nomenclatura de dominio del destino de correo (del programa de intercambio de correo, no de la computadora precisamente) donde debe ser entregado el correo.

El SMTP no especifica de qué manera el sistema acepta los mensajes o cómo se presenta al usuario el correo entrante. Se enfoca específicamente en cómo se transfieren los

mensajes de correo a través de un enlace de una máquina a otra, definiendo los formatos para texto sencillo mensajes y para mensajes multimedia, con la ayuda de los estándares MIME.

Por otro lado, se explica en este capítulo el protocolo para la administración de redes de TCP/IP, el SNMP. La administración de redes implica dos tareas diferentes: *Vigilar* significa observar el comportamiento de la red para asegurarse que esté funcionando sin problemas y observar los puntos de conflicto potenciales. *Controlar* significa cambiar la red mientras está corriendo, alterando de cierta forma la configuración de los equipos, con el fin de mejorar el desempeño o corregir las partes que no estén funcionando correctamente. De esta forma, tenemos que los objetivos del SNMP son:

- Vigilar y controlar las funciones de red.
- Minimizar el número y complejidad de las funciones de administración.
- Ser independiente de la arquitectura de red y de los diferentes tipos y marcas de equipos.

SNMP sigue el modelo de una base de datos. Todos los sistemas de la red contienen información sobre la configuración, estado, error y rendimiento a la que los administradores les gustaría acceder. Esta información se ve desde una base de datos llamada MIB, la cual puede ser manipulada por el administrador de red.

5.1 PROTOCOLO DE TRANSFERENCIA DE CORREO SIMPLE (SMTP).

5.1.1 MODELO DEL CORREO ELECTRONICO.

El objetivo del Protocolo SMTP (Simple Mail Transfer Protocol) es transferir el correo confiable y eficientemente Para ello, se utiliza un programa de correo de usuario final, llamado Agente de Usuario de Correo MUA (Mail User Agent), cuyos objetivos principales son:

- Mostrar información de los mensajes de correo.
- Guardar los mensajes de entrada o de salida en carpetas o archivos locales.
- Disponer de un buen editor para componer el texto de los mensajes.

El diseño del correo electrónico está basado en el siguiente modelo de comunicación: como resultado de un usuario que demanda correo, el emisor SMTP establece una conexión de dos sentidos con el receptor SMTP utilizando el número de puerto de TCP. El receptor puede ser el destino final o un destinatario intermedio, que puede ser un servidor de correo para una red LAN, el cual reenvía el correo a otro servidor hasta encontrar su destino. Los sistemas que utilizan reenvío se denominan sistemas de almacenamiento y envío (*store and forward*).

Los comandos SMTP se generan por el emisor y se envían al receptor; las respuestas SMTP a estos comandos recibidos se envían del receptor al emisor.

La entrega de correo es un nuevo concepto porque difiere fundamentalmente de otros usos de las redes. Los protocolos de red envían paquetes directamente a sus destinos, utilizando un límite de tiempo y retransmisión si no se devuelve un acuse de recibo. En un sistema de correo electrónico el emisor no desea esperar a que la máquina remota esté disponible para seguir trabajando, ni el usuario quiere que se aborte la transmisión sólo por que las comunicaciones con la máquina remota no están disponibles temporalmente.

Para manejar este tipo de entregas con retraso, el sistema de correo utiliza una técnica conocida como *spooling*. Cuando el usuario envía un mensaje de correo, el sistema coloca una copia en su área de almacenamiento privado (*spool*) junto con la identificación del emisor, máquina de destino y hora de depósito. El sistema, entonces, inicia la transferencia hacia la máquina remota como una actividad subordinada o secundaria, permitiendo al emisor que continúe con otras actividades computacionales. La figura 5.1 muestra este proceso.

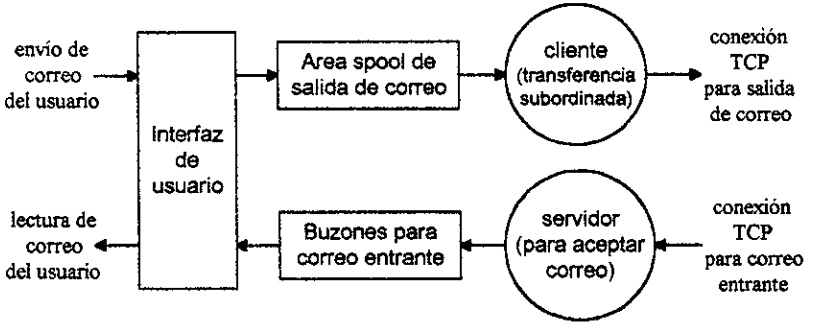

Figura 5.1 Componentes conceptuales de un sistema de correo electrónico.

El proceso subordinado de transferencia de correo se establece como un cliente El proceso primero utiliza el sistema de nombres de dominio para transformar el nombre de la máquina destino en una dirección IP y luego trata de establecer una conexión TCP hacia el servidor de correo en la máquina destino. Si tiene éxito, el proceso de transferencia envía una copia del mensaje al servidor remoto, el cual almacena la copia en el área de proceso *spool* del sistema remoto. Una vez que el cliente y el servidor acuerdan que la copia ha sido aceptada y almacenada, el cliente desecha la copia local. Si no se puede establecer una conexión TCP o si la conexión falla, el proceso de transferencia registra la hora en que se intentó la entrega y termina el proceso. El proceso de transferencia subordinado realiza de manera periódica un barrido a través del área *spool*, por lo general, una vez cada 30 minutos, en busca de correo no enviado. Cada vez que se encuentra un mensaje, el proceso subordinado intenta entregarlo de nuevo. Si encuentra que el mensaje de correo no se puede entregar después de un tiempo prolongado (por ejemplo, un día), el software de correo devuelve el mensaje al emisor.

Se ha diseñado un conjunto adicional de normas adaptadas a la forma de trabajo actual de mucha gente que trabaja en una red LAN que utiliza su propio servidor de correo. El Protocolo de Oficina de Correo POP (Post Office Protocol) permite a un cliente transferir sus mensajes desde el servidor de correo hasta su máquina. Como alternativa, el Protocolo de Acceso a Correo de Internet IMAP (Internet Message Access Protocol), que permite que un usuario lea, copie o borre mensajes almacenados en un servidor, pero el servidor es el depositario autorizado de los mensajes. Esto resulta útil para los usuarios que quieran beneficiarse de los servicios administrativos, como el realizar una copia de seguridad diaria, evitar el uso de espacio en el disco duro local o tener acceso a su correo cuando están de viaje.

5.1.2 COMANDOS Y RESPUESTAS DEL SMTP.

Las órdenes SMTP definen la transferencia de correo o el funcionamiento del sistema requerido por el usuario. Esta es la lista de comandos que indican una operación:

- **HELO** (Hello). Esta sentencia se usa para identificar al emisor SMTP. Este campo contiene el nombre del host del emisor. El receptor SMTP se identifica al emisor en la conexión, respondiendo a este comando. Esta orden y una respuesta OK confirman que tanto el emisor como el receptor están en estado inicial, esto es, no hay transacción en progreso y todas las tablas de estado y buffers están limpios. Este sería el primer comando enviado por el emisor cuando se establece una nueva transmisión SMTP.
- **MAIL** *"dirección de correo del emisor"*. Esta orden se usa para iniciar una transacción de correo en la cual los datos son entregados a uno o varios buzones.
- **RCPT** *" dirección de correo del recipiente (es) "* (Recipient). Esta orden se usa para identificar un usuario individual de un correo: Este comando se repite para identificar múltiples destinatarios. Si es posible, el receptor comprueba la validez del nombre del destinatario e indica e l resultado en el mensaje de la respuesta. En un host de reenvío no es común una comprobación inmediata. Si más adelante descubre que algún destinatario no era válido, envía un mensaje de correo breve al origen, informando el error.
- **DATA**. Si el receptor acepta el este comando, envía una respuesta al emisor, indicándole que está listo para recibir el mensaje. A partir de entonces, cualquier información enviada por el emisor es considerada parte del mensaje. El correo se termina mediante una línea que contiene un solo punto, y se envía una respuesta afirmativa al emisor.

- **RSET** (Reset). Termina el proceso de correo que se esté ejecutando Toda la información de la transacción se desecha, incluyendo los nombres del emisor y del receptor así como los datos del correo. Este comando es seguido por una respuesta OK del receptor SMTP.
- **NOOP.** Este comando no afecta ningún parámetro o comando, sólo especifica que el receptor envió una respuesta OK.
- **QUIT.** Indica al otro extremo que se desea terminar la transmisión SMTP. Debe enviarse un OK antes de cerrar el canal de transmisión.

Estos comandos requieren de una serie de respuestas que indiquen el estado de la transmisión. Estas respuestas constan de un código de tres dígitos que van seguidos de un texto informativo para el usuario, pero el sistema únicamente analiza los tres números. A continuación se explica el significado de estos tres dígitos:

- **PRIMER DIGITO.**

Las respuestas SMTP se categorizan en cinco clases o series, cada una se identifica por el primer dígito del código, de manera que una respuesta 211 pertenece a la serie 200 o una respuesta 554 corresponde a la serie 500. Estas series representan lo siguiente:

MENSAJES DE LA SERIE 100. Indican que un comando enviado por el emisor SMTP se valida pero este comando no ha terminado. Uno de estos mensajes podría ocurrir si un mensaje del correo fue enviado a través de otra red antes de alcanzar su destino final. En este caso el receptor SMTP sería solamente una destinación intermedia. Una respuesta sería

enviada indicando que el mensaje del correo fue recibido en ese punto, pero no se sabe si el mensaje fue entregado al destino final.

MENSAJES DE LA SERIE 200. Informan que el comando enviado por el emisor SMTP fue ejecutado con éxito.

MENSAJES DE LA SERIE 300. Indican que el comando fue aceptado, pero se requiere más información para ejecutar el comando.

MENSAJES DE LA SERIE 400. Estos mensajes representan un error temporal. En este caso el comando no fue aceptado y la acción solicitada no ocurrió. Esto podría ocurrir cuando el sistema de ficheros del receptor SMTP está lleno y no puede recibir más correo. Aunque los mensajes señalan un error, el sistema está comunicando que la acción solicitada puede intentarse otra vez posteriormente.

MENSAJES DE LA SERIE 500. Estos mensajes informan que un error permanente ha ocurrido. En este caso el comando no fue aceptado y la acción solicitada no se debe intentar posteriormente. Un ejemplo de este tipo de mensaje de error sería cuando una petición de correo fue enviada a un usuario que no existe.

- **SEGUNDO DIGITO.**

Este dígito indica categorías de error. Por ejemplo, un 0 representa un error de sintaxis o un 2 se refiere al status de la transmisión o al canal de comunicación.

- **TERCER DIGITO.**

Es usado para diferenciar el texto descriptivo en una serie específica de códigos de respuesta.

En la siguiente tabla se muestran los comandos de respuesta más usuales en un envío de correo.

CODIGO DE RESPUESTA	DESCRIPCION
211	Estado del sistema.
214	Mensaje de ayuda.
220	Servicio listo.
221	Servicio cierra el canal de transmisión.
250	OK. Acción de correo solicitada correcta o terminada.
251	Usuario no es local.
354	Empezar entrada de correo.
421	Servicio no disponible, se cierra el canal de transmisión.
450	Buzón no disponible.
451	Se abortó la solicitud; error en el procesamiento.
452	Almacenamiento del sistema insuficiente.
500	Error de sintaxis, comando no reconocido.
502	Comando no implementado.
503	Mala secuencia de comandos.
550	No existe nombre de usuario.
553	Nombre de buzón no permitido.
554	Falló la transacción.

Figura 5.2 Ejemplos de códigos de respuesta SMTP.

El diálogo que ocurre entre el cliente y el servidor consiste en texto ASCII que es posible leer. Inicialmente, el cliente establece una conexión de flujo confiable con el servidor y espera que el servidor envíe un mensaje *220 READY FOR MAIL* (si el servidor está sobrecargado deberá retardar el envío del mensaje *220* temporalmente). Al recibir este mensaje, el cliente envía un comando *HELO*, a lo cual el servidor responde identificándose.

Una vez que el canal de transmisión se ha establecido, el emisor envía un comando *MAIL* indicando el nombre del emisor del correo. Si el receptor acepta el correo, envía la

respuesta *250 OK*. Entonces, se manda un comando *RCPT*, para identifican al receptor. Se envía un mensaje de este tipo por cada receptor de correo a quienes se desee enviar el mensaje. Los receptores deben enviar un acuse de recibo enviando un *250 OK* o el mensaje de error *550 No such user here*. Después de que todos los comandos *RCPT* han sido reconocidos, el emisor emite un comando *DATA*. Esta orden informa al receptor que el emisor está listo para transferir un mensaje de correo. El receptor responde con el mensaje *354 Start mail input* y especifica la secuencia de caracteres utilizada para terminar el mensaje de correo.

El usuario no verá normalmente los comandos aplicados a este protocolo, sino que operará con los comandos suministrados por el fabricante para hacer más fácil el servicio a usar. Utilizando tales métodos el programa agente de usuario generará los comandos del cliente y manejará respuesta del servidor.

A continuación se muestra un ejemplo de cómo se envía el correo a varios recipientes, en donde se usan las letras *C* y *S* para identificar los mensajes enviados por el cliente y el servidor, respectivamente:

```
S: 220 Beta.GOV Simple Mail Transfer Service Ready
C: HELO Alpha.EDU
S: 250 Beta.GOV
C: MAIL FROM :<usuario1@Alpha.EDU>
S: 250 OK
C: RCPT TO: <usuario2@Beta.GOV>
S: 250 OK
C: RCPT TO :<usuario3@Beta.GOV>
S: 550 No such user here
C: DATA
S: 354 Start mail input; end with <CR><LF>.<CR><LF>
C: . . . cuerpo del mensaje de correo . . .
C: . . . continua hasta las líneas que el mensaje contenga . . .
C: <CR><LF>.<CR><LF>
S: 250 OK
C: QUIT
S. 221 Beta.GOV Service closing transmission channel
```

En este ejemplo el usuario 1 en el servidor *Alpha.EDU* manda un mensaje a los usuarios 2 y 3 en el servidor *Beta.GOV*. Al enviar un mensaje, el servidor rechaza al recipiente del usuario 3 porque no reconoce el nombre como un destino de correo válido (es decir, que no es un usuario ni una lista de correos). Sin embargo, el protocolo SMTP no especifica los detalles de cómo maneja un cliente estos errores; el cliente debe decidir. En general los clientes continúan con la entrega hacia los recipientes válidos y luego, reportan los problemas al emisor mediante un correo electrónico. El mensaje de error contiene un resumen de los errores así como el encabezado del mensaje de correo que ha ocasionado el problema.

Una vez terminado con el envío del correo se solicita terminar la conexión por medio del comando *QUIT* a lo que el servidor responde con un mensaje *221* para cerrar la conexión TCP.

5.1.3 EXTENSIONES MULTIPROPOSITO DE CORREO EN INTERNET (MIME).

Para permitir la transmisión de datos no ASCII a través del correo electrónico, se han definido a MIME (Multipurpose Internet Mail Extensions). Este estándar aumenta la capacidad del correo electrónico, lo que permite incluir en un mensaje audio, video, gráficos y, sobre todo, texto en cualquier conjunto de caracteres. De esta forma se solucionan los problemas generados por aquellos idiomas que, como el castellano, tienen caracteres (*eñe*, letras acentuadas) que no se pueden representar con los siete bits utilizados por el protocolo SMTP y que llegaban al destinatario convertidos en otros caracteres. Hay que señalar, no obstante, que la utilización de un cliente que siga el estándar MIME no es

suficiente para garantizar que el mensaje llegue inalterado, ya que puede haber nodos intermedios (MTA = Mail Transfer Agents o "estafetas") que no obedezcan dicho estándar.

La información MIME reside en el encabezado de correo, en donde se especifican tres cosas:

1.- La versión de MIME utilizada.

2.- Una declaración *Content – Transfer – Encoding*, que muestra la codificación empleada para convertir los datos en ASCII

3.- Una declaración *Content – Type*, la cual debe contener dos identificadores: uno con el contenido o tipo de datos que se envían y un subtipo, separados por una diagonal. Lo anterior se muestra en el ejemplo, en donde *image* es el tipo de contenido y *gif* es el subtipo:

```
From : alejandro@umbral.com
To : zamy75@hotmail.com
MIME - Version : 1.0
Content - Type: image / gif
Content - Transfer - Encoding: base64
  . . .  datos de la imagen  . . .
```

MIME define siete tipos de contenidos básicos, los subtipos válidos para cada uno y las codificaciones de transferencia. Por ejemplo, aun cuando una imagen puede tener los subtipos *jpeg* o *gif*, el texto no puede utilizar ningún subtipo. La figura 5.3 lista los siete tipos de contenidos básicos:

TIPO DE CONTENIDO	DATOS EN EL MENSAJE
Text	Texto (por ejemplo un documento)
Image	Imagen
Audio	Grabaciones de sonido
Video	Grabaciones de video que incluyen movimiento
Application	Datos para un programa
Multipart	Mensajes múltiples de los que cada uno tiene una codificación y un tipo de contenido diferentes
Message	Mensajes de correo electrónico

Figura 5.3 Tipos básicos que pueden aparecer en una declaración *Content-Type* de MIME.

Dentro del tipo de contenido *multipart* se definen cuatro posibles subtipos. El subtipo *mixed* permite que un solo mensaje contenga submensajes independientes, de los que cada uno tiene un tipo independiente y una codificación diferente. Con esta característica, es posible incluir textos, gráficos y audio en un solo mensaje. El subtipo *alternative* permite que un solo mensaje incluya varias representaciones de los mismos datos. Estos mensajes son útiles cuando se envia un mensaje a muchos recipientes de los que no todos utilizan el mismo hardware y software de sistema. El subtipo *paralell* permite que un solo mensaje incluya subpartes que deben ser vistas juntas (por ejemplo, subpartes de audio y video que deben presentarse de manera simultánea). Por último, el subtipo *digest* permite que un solo mensaje contenga un conjunto de otros mensajes (por ejemplo, la colección de mensajes de correo de una discusión).

5.2 PROTOCOLO SIMPLE PARA LA ADMINISTRACION DE RED (SNMP).

5.2.1 ARQUITECTURA Y CONCEPTOS DEL SNMP.

Dentro de la arquitectura del Protocolo Simple para Administración de Red SNMP (Simple Network Management Protocol) existen los siguientes elementos:

Estaciones de administración de red NMS (Network Management Stations). Ejecutan aplicaciones de administración que monitorizan y controlan los elementos de red mediante un software servidor SNMP. Se pueden mencionar los siguientes ejemplos de software para la administración de redes: el Open View de Hewlett Packard, el Manage Wise de Novell o el UniCenter TNG de Computer Associates. Este software permite a la estación de administración transferir o solicitar diferentes tipos de información. Por ejemplo, el NMS solicita las estadísticas de los dispositivos, incluyendo el número de paquetes que se manejan, el estado del dispositivo, las condiciones especiales que están asociadas con el tipo de dispositivo (como las indicaciones de que se terminó el papel en una impresora o la pérdida de conexión en un módem) y la carga del procesador.

Nodos administrables. Son dispositivos como hosts, impresoras, módems, ruteadores y parecidos, que están equipados con un software llamado *agente SNMP* (comúnmente de 64 Kb) y una base de datos llamada Base de Información sobre la Administración MIB, los cuales pueden ser manejados o consultados desde una estación de administración. El agente responde a solicitudes de información y de acción que provienen de la estación de gestión,

y puede regresar información importante a esta estación que no ha sido solicitada (alarmas). Los periféricos que tienen integradas las capacidades para su gestión, corren el software de agente SNMP, cargado como parte de un ciclo de arranque del dispositivo.

Entidades de doble función. Se ha dicho que las estaciones de administración sólo interactúan con los nodos sin embargo, es necesario apreciar que el software de cada estación de administración puede realizar tanto la función de administrador como la de agente. Teniendo esto en cuenta, se pueden construir relaciones jerárquicas entre las estaciones de administración. Por ejemplo, se puede diseñar un sistema de administración en donde cada segmento de una LAN tenga una aplicación de administración que controle el estado de los dispositivos de ese segmento; estas aplicaciones deberían informar a aplicaciones de estaciones de administración regionales, las cuales deberían informar a estaciones de administración entre empresas. En este ejemplo, el software de cada estación realiza un papel de administrador al monitorizar y controlar dispositivos que dependen de él jerárquicamente, y un papel de agente al informar y actuar según los comandos proporcionados por un superior jerárquico.

El SNMP es el protocolo encargado de comunicar los dispositivos administrados y las estaciones de administración, para lo cual utiliza el esquema de cliente/servidor, considerando a las Estación de Administración de Red como cliente y al agente del nodo administrado como servidor. Este protocolo es asíncrono, es decir, no necesita esperar por una respuesta después de enviar un mensaje en particular.

Este protocolo considera los aspectos que deben considerarse en la administración de redes. Para empezar, se debe conocer la diversidad de dispositivos existentes y proporcionar un entorno apropiado. Existe un axioma fundamental para la administración de redes, el cual dice lo siguiente:

El impacto de añadir una administración de red a un nodo debe ser mínimo.

También es importante la elección de un servicio de transporte por parte del protocolo de administración, ya que de estos mecanismos internos depende la efectividad del protocolo, y de acuerdo con el Axioma Fundamental, hay que elegir la forma de comunicación menos impactante en el proceso. Esto conduce a elegir un servicio de transporte no orientado a conexión por medio del protocolo UDP.

Además de estas características, el SNMP requiere de otros dos componentes (que se verán más adelante) con los que forma un equipo:

- **Base de Información sobre la Administración (MIB).** - Una base de datos que contiene información del estado de los nodos administrados.
- **Estructura de Información sobre la Administración (SMI).** - Una especificación que define las entradas en una MIB.

5.2.2 DETECCION DE PROBLEMAS CON SNMP

En el entorno de administración, cada nodo tiene una serie de variables, de modo que leyendo estas variables se monitoriza el nodo, y cambiándoles el valor se controla. Además de estas operaciones de lectura y escritura, existen otras dos:

- **Una operación de examen** o de sondeo, que permite determinar a la estación de administración qué variables soporta un nodo. Es decir, como cada nodo realiza distintas funciones dentro de la red, también contiene diferentes variables de administración. Las estaciones deben determinar qué variables se soportan. Sin embargo, el protocolo debe proporcionar un significado para examinar la lista de variables soportadas por un nodo.
- **Una operación de interrupción**, que permite a los nodos informar a la estación de administración de un evento extraordinario.

Con el método basado en interrupciones, tenemos las siguientes ventajas: Cuando ocurre un evento extraordinario, el nodo envía una interrupción a la estación de administración adecuada (suponiendo que el dispositivo no ha caído y se puede alcanzar la estación). Por tanto tenemos la ventaja de una notificación inmediata.

Las desventajas del método basado en interrupciones son: Requiere recursos para generar la interrupción ya que si la interrupción debe contener mucha información, el nodo perderá tiempo en prepararla y no se dedicará a cosas útiles. Por supuesto, cuando se genera una interrupción, el agente asume que la aplicación de administración está preparada para recibir la información. Por tanto hay que usar un diseño cuidadoso para que las interrupciones sean recibidas sólo por aquellas estaciones interesadas. Más aún, si ocurre un evento extraordinario, las interrupciones pueden ocupar un gran ancho de banda, lo cual es poco deseable si se está informando de un problema de congestión de la red. Por eso se mejora el método de las interrupciones de modo que un nodo sólo informa cuando la ocurrencia de un evento sobrepasa un determinado umbral, pero esto implica que el agente debe gastar tiempo para determinar cuándo debe generar una interrupción. Es decir, el

método basado en interrupciones tiene un fuerte impacto en el agente, en la red o en ambos. En conclusión, como los nodos tienen una pequeña visión de toda la red, es conveniente que las aplicaciones de administración detecten el problema de alguna otra forma.

Con el método basado en sondeo, una aplicación de administración pregunta periódicamente al nodo cómo van las cosas, así el control lo tiene la aplicación, la cual sí tiene una visión global de la red.

La desventaja del método de sondeo es que la aplicación de administración no sabe qué elementos sondear ni con qué frecuencia: Si el intervalo de frecuencia es breve, se ocupa mucho ancho de banda, y si es muy largo, la respuesta a eventos catastróficos es demasiado lenta. Otra desventaja es el tráfico que se introduce en la red, por lo que la aplicación de administración debe tener recursos de almacenamiento adicionales para ello.

En el entorno de administración SNMP se usa el modelo *interrupción-sondeo directo* (*trap-directed polling*). Cuando ocurre un evento extraordinario, el nodo manda una interrupción simple a la aplicación. La aplicación es entonces la responsable de iniciar conversaciones con el nodo para determinar la naturaleza y la extensión del problema. Esto es muy efectivo ya que el impacto creado en los nodos es pequeño, en el ancho de banda es mínimo y los problemas se resuelven en el momento oportuno. Por tanto, las interrupciones actúan como una alarma previa, y se usa un sondeo de baja frecuencia.

5.2.3 MENSAJES SNMP.

Existen cinco tipos de mensajes disponibles en SNMP los cuales informan al NMS sobre el estado de los nodos administrados, con la ayuda de la MIB. Estos mensajes se colocan dentro de un programa UDP y se enrutan vía IP.

- **Get request** (obtener solicitud).- Utilizado para consultar una MIB de un nodo administrado. Contiene valores para proporcionar el status del nodo.

- **Get next request** (obtener la siguiente solicitud).- Utilizado para leer secuencialmente las filas de una tabla MIB, es decir, solo hace una visualización de ésta.

- **Set request** (fijar solicitud).- Se utiliza para escribir una acción que se llevará al cabo en un elemento. Permite al administrador cambiar valores en la MIB.

- **Get bulk**. (obtener volumen). – Solicita al agente obtener varios valores con una sola petición. Regresa tanta información de la solicitada como pueda.

- **Response** (respuesta) - Devuelve el resultado de una operación *get*, *get-next*, *get-bulk* o *set*.

- **Trap** (trampa).- Utilizado para reportar eventos extraordinarios, pudiendo ser por cambios en la configuración o falla en un agente.

Los mensajes SNMP se encierran en un datagrama de UDP, y son enviadas como se muestra en la figura, en donde se aprecia que las solicitudes se envían desde cualquier socket conveniente (llamado en la figura socket X), al puerto 161. Las respuestas se envían de vuelta al socket solicitante. Las trampas se envían desde cualquier socket conveniente (llamado en la figura socket Y), al socket 162.

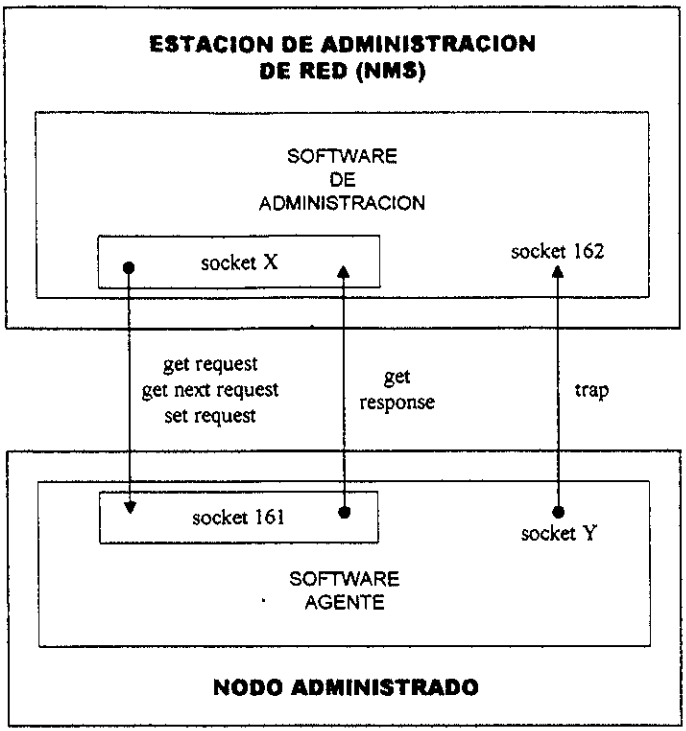

Figura 5.4 Mensajes del protocolo SNMP.

5.2.4 BASE DE INFORMACION SOBRE LA ADMINISTRACION (MIB).

Cada dispositivo administrado por SNMP, mantiene una base de datos que contiene estadísticas y otro tipo de información. Esta base de datos se llama MIB (Management Information Base), la cual especifica los elementos de los datos que un nodo administrado debe conservar y las operaciones permitidas en cada uno. Por ejemplo, un ruteador

mantiene estadísticas del estado de sus interfaces de red, el tráfico que entra y sale, de los datagramas eliminados y de los mensajes de error generados.

La MIB tiene información sobre el estado y desempeño del dispositivo, sus conexiones hacia los diferentes componentes y su configuración. La mayor parte de los administradores SNMP consultan a los agentes en un intervalo regular, 15 minutos por ejemplo, a menos que el usuario indique otra cosa.

El MIB divide la información de la administración en 8 categorías, como se muestra a continuación:

CATEGORIA MIB	INCLUYE INFORMACION SOBRE:
system	Sistema operativo del host o ruteador
interfaces	Interfaz de red individual
addr. trans.	Dirección de traducción (por ejemplo, transformación ARP)
ip	Software de Protocolo Internet
icmp	Software de Protocolo de Mensajes de Control Internet
tcp	Software de Protocolo de Transmisión de Internet
udp	Software de Protocolo de Datagrama de Usuario
egp	Software de Protocolo de Compuerta Exterior

Figura 5.5. Categorías de información del MIB.

Cada una de estas categorías tienen una lista de variables que son las que guardan información específica de los dispositivos administrados. La siguiente figura lista ejemplos de variables MIB junto con sus respectivas categorías:

VARIABLE MIB	CATEGORIA	DESCRIPCION
sysUpTime	system	Tiempo desde el ultimo arranque
ifNumber	interfaces	Número de interfaz de red
ifMtu	interfaces	MTU para una interfaz en particular
ipDefaultTTL	ip	Valor IP utilizado en el campo de tiempo limite
ipInReceives	ip	Número de datagramas recibidos
ipForwDatargams	ip	Número de datagramas enviados
ipOutNoRoutes	ip	Número de fallas de ruteo
ipReasmOKs	ip	Número de datagramas reensmblados
ipFragsOKs	ip	Número de datagramas fragmentados
ipRoutingTable	ip	Tabla de ruteo IP
icmpInEchos	icmp	Número de Solicitudes de Eco ICMP recibidas (ping)
tcpRtoMin	*tcp*	*Tiempo de transmisión mínimo TCP permitido*
tcpMaxConn	tcp	Conexión TCP máxima permitida
tcpInSegs	tcp	Número de segmentos que TCP ha recibido
udpInDatagrams	udp	Número de datagramas UDP recibidos

Figura 5.6. Ejemplos de variables MIB.

5.2.5 ESTRUCTURA DE LA INFORMACIÓN SOBRE LA ADMINISTRACIÓN (SMI).

Además del estándar MIB, el cual especifica variables de administración de red y sus significados, un estándar separado especifica un conjunto de reglas utilizadas para identificar las variables MIB. Estas reglas se conocen como SMI (Structure of Management Information).

El estándar SMI especifica que todas las variables del MIB deben definirse y ser referidas por medio de ASN.1 (Abstract Syntax Notation 1), que es un lenguaje formal que tiene dos características principales: una notación utilizada en documentos que los usuarios

pueden leer y una representación codificada compacta de la misma información empleada en los protocolos de comunicación. En ambos casos, la notación formal precisa cómo codificar los nombres y los datos en un mensaje para que la forma y el contenido de las variables se mantenga libre de ambigüedades.

Para esta asignación de nombres a las variables de la MIB se creó una estructura en árbol SMI, la cual proporciona un identificador de objeto (object identifier) a cada variable de administración. Los objetos o nodos superiores del árbol representaban a las autoridades administrativas responsables de las partes inferiores del árbol, como se aprecia en la siguiente figura.

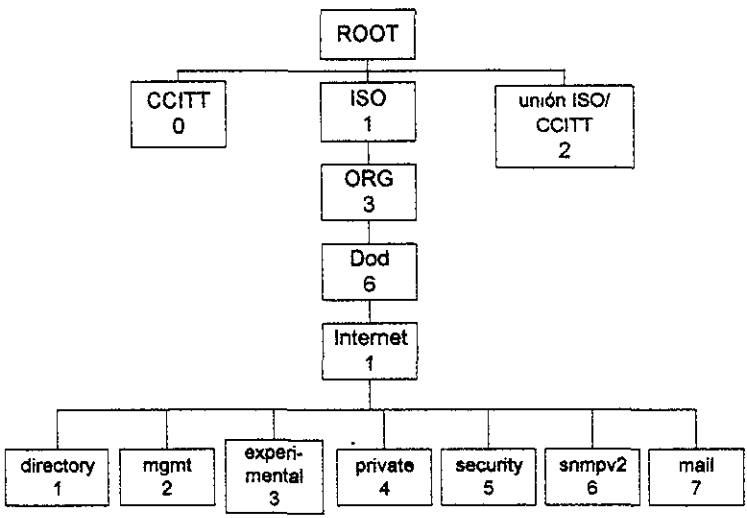

Figura 5.7. Arbol Administrativo de nombres SMI.

Administrativamente, el árbol es obsoleto, ya que fue diseñado para coordinar el SNMP con las normas ISO, además el Departamento de Defensa (DoD) ya no administra

Internet. Sin embargo, el árbol sigue sirviendo para su función principal de definir los nombres de las variables MIB. Actualmente, cuando se añade una nueva tecnología, a un entorno de administración, se crea un comité y se asigna un nuevo nodo a árbol. Después, el comité crea las variables que sean necesarias en su propio subárbol.

El nombre de objeto en la jerarquía es la secuencia de etiquetas numéricas de los nodos a lo largo de la trayectoria desde la raíz hacia el objeto. La secuencia está escrita con puntos que separan a los componentes individuales. Por ejemplo, el nombre *1.3.6.1.4* denota el nombre *private.*

Como se aprecia en la siguiente ilustración, el MIB ha sido asignado a un nodo bajo el subgrupo *internet mgmt* con el nombre de *mib* y con el valor numérico *1*. Debido a que todas las variables MIB quedan bajo este nodo, todas tienen nombres que comienzan con el prefijo *1.3.6.1.2.1.*

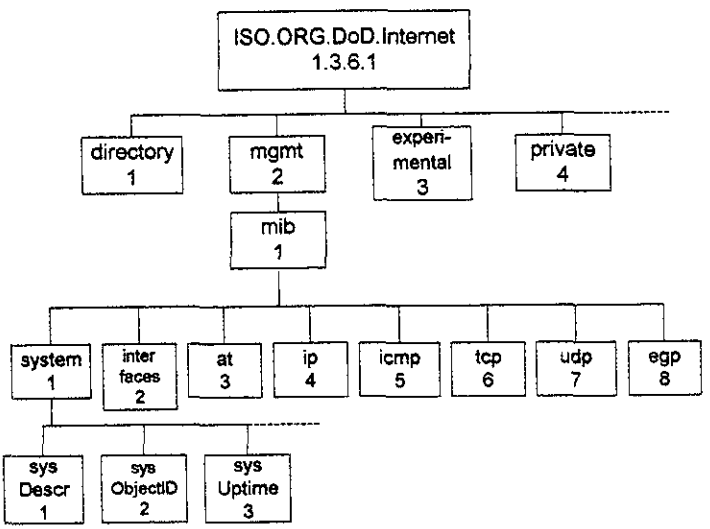

Figura 5.8. Arbol de nombres para los objetos de la MIB.

Como se puede ver, las ocho categorías de la MIB que se mostraron en la figura 5 5, corresponden a los subárboles del nodo *mib*. Cada una de estas categorías contiene las variables MIB, como las que se ejemplificaron en la figura 5.6.

Por ejemplo, de acuerdo al árbol, el identificador de objeto para la variable *sysUptime* de la categoría *system*, sería el siguiente: *1.3.6.1.2.1.1.3*. De esta forma, es posible identificar cada variable MIB con un identificador de objeto único, las cuales pueden ser llamadas para su consulta o para modificar sus parámetros mediante el software de administración de red.

CONCLUSIONES

Los protocolos TCP/IP se diseñaron para ser independientes del hardware de los equipos que conforman una red o de su sistema operativo, así como de las tecnologías de los medios y los enlaces de datos. De aquí su importancia, ya que han permitido el crecimiento explosivo de la red Internet, además de permitir a las redes locales una interconectividad que se transforma en eficiencia para la empresa que las utiliza.

El éxito alcanzado por TCP/IP se debe principalmente a que comprende una amplia variedad de protocolos distribuidos por capas, los cuales determinan la forma en que los datos son transferidos desde que el usuario solicita a un servicio, hasta la forma en que son procesados y enviados por los medios físicos. Esto permite la compatibilidad entre distintos tipos de redes y marcas de equipos, además de proporcionar un modelo estructurado que permite a quienes desarrollan nuevos protocolos y aplicaciones para el trabajo en red, olvidarse de los procesos que suceden en capas vecinas.

Los servicios que ofrece TCP/IP fueron planeados desde sus inicios, y continúan operando con grandes resultados, únicamente se han hecho mejoras a los protocolos con el paso del tiempo, pero su funcionamiento es básicamente el mismo, por lo que es fácil deducir que el futuro de TCP/IP consiste en la aparición de nuevas versiones de sus protocolos, para adaptarlos mejor a la creciente demanda de usuarios y hacer más eficiente el transporte de aplicaciones multimedia.

Esta tesis se ha enfocado al estudio de los principales protocolos del nivel de aplicación, tomando como antecedente la explicación de los protocolos de la capa internet

(IP e ICMP) y los protocolos de transporte (TCP y UDP), quienes proporcionan las reglas para la comunicación.

Este grupo de protocolos, contienen los detalles de los formatos de los mensajes, describen cómo responde una computadora cuando llega un mensaje y especifican de qué manera se maneja un error.

En la capa superior, encontramos a los protocolos de aplicación, cuyos servicios fueron clasificados para su estudio en este trabajo en el acceso remoto, la transferencia de archivos, el correo electrónico y la administración de redes. La importancia los protocolos de aplicación radica en la función que estos cumplen, al ofrecer a cualquier usuario, herramientas útiles para trabajar en la red.

Al explicarse los protocolos de la capa de aplicación de TCP/IP, esta tesis representa una guía importante, para cualquier usuario que trabaje dentro de una red LAN o alguien que desde su casa tenga acceso a Internet, ya que los servicios descritos son los de mayor demanda. La información presentada en este trabajo, es especialmente de gran utilidad para quienes desarrollan su trabajo en área de administración de redes o soporte técnico, para conocer más a detalle los procesos y características de estas aplicaciones, y saber aún más que un usuario común.

GLOSARIO

Agente.- En el protocolo SNMP es el proceso de un dispositivo que responde y hace peticiones y envía mensajes de excepción.

ANSI (American National Standards Institute).- Organización responsable en Estados Unidos de la coordinación de las actividades de estandarización. ANSI es miembro de ISO.

ASN.1.- Protocolo estándar de presentación de ISO utilizado por SNMP para representar mensajes.

ARP (Protocolo de Resolución de Dirección).- Protocolo que asigna una dirección IP de alto nivel a una dirección de física de bajo nivel.

ARPANET.- La primera red de conmutación de paquetes, que durante mucho tiempo funcionó como núcleo de Internet.

ASCII.- Código estándar de la ANSI para el intercambio de información. Para definir un carácter ASCII se requieren de siete de los ocho bits de un octeto.

Asíncrono.- Es un tipo de comunicación que envía datos usando el control del flujo sin necesidad de sincronización entre un terminal origen y un terminal destino.

Autenticación.- Verificación de la identidad del otro extremo de una comunicación.

Buffer.- Area de memoria utilizada para almacenar temporalmente los datos de entrada o de salida.

BW (Ancho de Banda).- Cantidad de datos que se puede enviar por un enlace, normalmente medida en bits por segundo.

Circuito Virtual.- Término que proviene de las redes de conmutación de paquetes. Un circuito virtual se crea usando medios que comparten muchos usuarios, aunque cada circuito aparezca ante el usuario como una conexión punto a punto dedicada.

CCITT (Comité Consultivo Internacional de Telefonía y Telegrafía).- Organización creada para facilitar la conexión de sistemas de comunicación a las redes internacionales. Actualmente se le conoce como la ITU-T.

Cliente.- Es un software que trabaja en la computadora para hacer uso de algún servicio de una computadora remota.

Datagrama.- Unidad de datos encaminados por el protocolo IP.

Dirección IP.- Número de 32 bits que identifica una interfaz de red.

Dirección de subred.- Un determinado número de bits de la parte local de una dirección IP que se utiliza para identificar a los sistemas conectados a una red común.

DNS (Sistema de Nombres de Dominios).- Conjunto de bases de datos distribuidas con información como la traducción entre nombres de sistemas y sus direcciones IP.

DTE (Equipo Terminal de Datos).- Origen o destino de los datos. A menudo se usa para referirse a las computadoras conectadas a una red.

EBCDIC.- Codificación utilizada por los host de IBM para sus archivos de texto.

EGP (Protocolo de Gateway Exterior).- Protocolo utilizado por un ruteador para anunciar al conjunto de redes a las que puede alcanzar un sistema autónomo. Un sistema autónomo es una colección de ruteadores bajo el control de una única autoridad administrativa.

Ethernet.- Es un tipo de red de área local. En este tipo de red las computadoras pueden utilizar el protocolo TCP/IP, por lo que muchas computadoras acceden a Internet a través de una LAN Ethernet.

FTP.- Protocolo de TCP/IP para la copia de archivos entre sistemas, con funciones de administración de archivos, como el renombrado o el borrado de los mismos.

Gateway.- Un ruteador IP.

Host (Anfitrión).- Cualquier sistema de computadora de usuario final que se conecta a una red.

Hub (Concentrador).- Dispositivo al que se conectan varias computadoras, por lo general mediante un cable de par trenzado.

ICMP.- Protocolo necesario para la implementación con IP. ICMP especifica los mensajes de error que hay que enviar cuando se descarga un datagrama o el sistema se congestiona.

Internet.- La mayor red del mundo. Internet utiliza el conjunto de protocolos TCP/IP.

IP.- Protocolo de la capa 3 de TCP/IP responsable del transporte de datagramas por Internet.

ISO (Organización Internacional de Normalización).- Organización fundada para promover el comercio Internacional y el progreso de la ciencia y tecnología.

ITU-T (Grupo de Normalización de Telecomunicaciones de la Unión Internacional de Telecomunicaciones).- Organización que preside grupos de estudio y escribe recomendaciones para los estándares internacionales. Anteriormente CCITT.

MIB (Base de Información sobre la Administración).- Conjunto de todas las definiciones de objetos de red que se pueden administrar. También es la configuración, estado e información que se puede obtener de un dispositivo de red.

MIME (Extensiones Multipropósito de Correo en Internet).- Extensiones del correo electrónico que permiten el envío de mensajes en una o varias partes, cada una de las cuales con distintos tipo de contenidos, como texto, imágenes, sonido o datos de aplicaciones.

MTA (Agente de Transferencia de Mensajes).- Entidad que mueve los mensajes de correo electrónico entre computadoras.

MTU (Unidad Máxima de Transferencia).- Es la mayor cantidad de datos que se puede transferir a través de una red física. El MTU lo determina el hardware de la red.

NFS.- permite que se creen enlaces a directorios remotos en sus sistemas de archivos locales y usen los archivos remotos como si fueran locales.

NVT (Terminal Virtual de Red).- Conjunto de reglas que definen una interacción simple de terminal virtual. NVT se usa al comienzo de una sesión Telnet.

OSI (Interconexión de Sistemas Abiertos).- Conjunto de estándares de ISO relativas a la interconexión de datos.

Paquete.- En una red los datos transmitidos por una computadora son divididos en conjuntos de caracteres independientes. Cada paquete viaja por la red independientemente de los demás hasta llegar a su destino.

PDU (Unidad de Datos del Protocolo).- Termino genérico para las unidades de datos de un protocolo, en cualquier capa.

Ping (Packet InterNet Groper).- Programa utilizado para comprobar al accesibilidad física de un destino, enviando una solicitud de eco ICMP y esperando una respuesta.

Protocolo.- Es un conjunto de normas que indican cómo deben actuar los elementos de una red para comunicarse entre sí.

POP (Protocolo de Oficina de Correos).- Protocolo que se usa para descargar el correo electrónico de un servidor a un cliente.

Puerto.- Número binario de dos octetos que identifica a una aplicación de alto nivel de TCP o UDP.

RARP (Protocolo de Resolución de Dirección Inverso).- Protocolo que una máquina sin disco duro utiliza al arrancar para encontrar su dirección IP.

Ruteador.- Sistema que reenvía el tráfico que no es para sí mismo. Los ruteadores se usan para conectar LAN y WAN separadas en una red, y para dirigir el tráfico entre las redes que la forman.

RFC (Petición de Comentario).- Nombre de una serie de notas que contienen estudios, mediciones, técnicas y observaciones, así como estándares de los protocolos TCP/IP aceptados. Los RFC están disponibles por Internet.

RPC (Llamada a Procedimiento Remoto).- Protocolo que permite que una aplicación invoque a una rutina que se ejecuta en un servidor. El servidor devuelve variables de salida y un código de retorno de vuelta.

Servidor.- Se trata de un software instalado en una computadora, que le permite ofrecer un servicio a otra computadora llamada local. El computador local contacta con el servidor remoto gracias a otro software llamado cliente.

Servidor Proxy.- Es un programa que se ejecuta en un servidor situado entre una red privada e internet. Se utiliza para filtrar todas las conexiones exteriores con el fin de que aparezcan que son de la misma máquina y, al mismo tiempo, evitar el acceso a la red interna de intrusos.

Síncrono.- Es un método de comunicación a través de una conexión controlada por un temporizador que requiere que cada participante esté sincronizada con el resto.

SMTP.- Protocolo de TCP/IP usado para la transferencia de correo entre sistemas.

SNMP.- Protocolo que permite que una estación de administración controle sistemas de red y reciba mensajes de avisos (alarmas) de los sistemas de red.

Socket.- Entero que usa una aplicación para identificar una conexión. Un socket está formado por una dirección de red de 32 bits y un número de puerto de 16 bits.

TCP.- Ofrece una transmisión de datos orientada a conexión, fiable, entre un par de aplicaciones.

Telnet.- Protocolo de aplicación de TCP/IP que permite a un terminal conectado a un host acceder a otros host e interactuar con sus aplicaciones.

TFTP-. Protocolo para la carga y descarga de archivos. El uso típico suele ser la inicialización de estaciones de trabajo sin disco.

Token Ring.- Es un tipo particular de red de área local. Estas redes utilizan frecuentemente el protocolo TCP/IP.

Trama (frame).- Unidad de datos del protocolo de la capa de enlace. Se refiere a la unidad que se envía fuera de la estación origen en una red física.

UA (Agente de Usuario).- Aplicación de correo electrónico que ayuda a un usuario final a preparar, guardar y enviar mensajes, y a ver, almacenar y responder a los mensajes entrantes.

UDP.- Protocolo básico que permite que una aplicación envíe mensajes individuales a otras aplicaciones. No garantiza la entrega y los mensajes no tienen porqué entregarse en el mismo orden en que se enviaron. Se utiliza cuando se necesita transmitir voz o video y resulta más importante transmitir con velocidad que garantizar el hecho de que lleguen todos los bytes.

UNIX.- Es un sistema operativo multitarea y multiusuario, sobre el cual se desarrollaron los protocolos TCP/IP.

X.25.- Es un protocolo de transmisión de paquetes ISO utilizado en redes de área amplia. Forma parte del modelo OSI.

XDR (Representación Externa de Datos).- Estándar que define los tipos de datos usados como parámetros y la codificación de dichos parámetros para su transmisión.

BIBLIOGRAFIA

- Aprendiendo TCP/IP en 14 días.
  Thimothy Parker
  Ed. Prentice Hall

- Internetworking with TCP/IP.
  Principles, Protocols and Architecture.
  Vol. I
  Douglas E. Comer
  Ed. Prentice Hall

- Internetworking with TCP/IP.
  Principles, Protocols and Architecture.
  Vol. II
  Douglas E. Comer
  Ed. Prentice Hall

- Redes Globales de Información con Internet y TCP/IP.
  Principios básicos, protocolos y arquitectura.
  Tercera Edición
  Douglas E. Comer
  Ed. Prentice Hall

- Redes Locales y TCP/IP.
  José Luis Raya
  Ed. Ra-Ma

- TCP/IP.
  Arquitectura, protocolos e implementación con Ipv6 y seguridad de IP.
  Primera Edición
  Dr. Sidnie Feit
  Ed. McGraw Hill

- Tecnología de Interconectividad de redes.
  Steve Spanier
  Tim Stevenson
  Ed. Prentice Hall