FACULTAD DE INGENIERIA U.N.A.M.
DIVISION DE EDUCACION CONTINUA

TCP/IP ARQUITECTURA, PROTOCOLOS E IMPLEMENTACION MODULO A

OPTATIVO, DIPLOMADO EN REDES

ABRIL 29 A 03 MAYO 1996

* * * *

Palacio de Mineria Calle de Tacuba 5 Primer piso Deleg Cuauhtemoc 06000 México, D F APDO Postal M-2285
Teléfonos 512-8955 512-5121 521-7335 521-1987 Fax 510-0573 521-4020 AL 26

CURSO: TCP/IP ARQUITECTURA, PROTOCOLOS E IMPLEMENTACIÓN

PRESENTACIÓN

TCP/IP (Transmitssion Control Protocol /Internet Protocol), es una familia de protocolos elaborado por la Universidad de Michigan, E.U. en 1982, como evolución del proyecto ARPANET del Departamento de la Defensa de los Estados Unidos y por la necesidad de interconectar las computadoras de diferentes características, existentes en las distintas universidades y centros de investigación relacionados con el Pentágono. Vale la pena señalar que ARPANET posteriormente dio origen a la red globalizada tan popular en estos tiempos, conocida como INTERNET.

La evolución de TCP/IP y su desarrollo actual está a cargo del IBA, comité formado por científicos altamente calificados, organismo que trimestralmente publica las actualizaciones, revisiones y las especificaciones de nuevos protocolos. Desde la planeación original de este grupo de protocolos, se consideró que éstos actuaran independientes del medio físico de enlace, característica que permite una conectividad de gran potencial. Actúan a partir del nivel 3 del modelo OSI, los ambientes que emplean se basan en que cada elemento de una red, tenga su propia dirección IP, y esto garantiza que se identifique y/o direccione cada nodo en una red, sea LAN o WAN, sea una o varias redes enlazadas entre sí, etc. permiten además que el ruteo sea eficiente, en síntesis, TCP/IP permite la conectividad global. Su arquitectura permite asegurar que la información en cada extremo se entregue precisa, en secuencia, completa y por la ruta más eficiente.

El conocimiento de este recurso de conectividad requiere de mucho tiempo y dedicación, este breve resumen sólo da una idea de lo que representa esta herramienta tan útil y necesaria en el campo de las redes de computadoras, herramienta que en el curso se verá a buen nivel, para que sirva de apoyo a quienes estén involucrados con las redes. Es tan importante TCP/IP que sin ellos no existiría INTERNET ni nada parecido. Terminaremos señalando que este módulo es opcional y como parte del Diplomado de Redes.

OBJETIVOS

Lograr que los participantes entiendan qué son y qué hacen la familia de protocolos de TCP/IP.

Conocer la arquitectura de los protocolos para que puedan diseñar e implementar eficientemente una red.

Implementar en las prácticas una red dónde los participantes comprueben el potencial de estos protocolos.

A QUIEN VA DIRIGIDO

A todos aquellos profesionales y profesionistas que por sus necesidades laborales, estén involucrados con las Redes de Cómputo y requieran actualizarse en las Redes de Alto Desempeño, y a los Ejecutivos que necesiten bases técnicas en su responsabilidad de toma de decisiones.

REQUISITOS

Los participantes deben tener conocimientos en Redes (LAN) de Cómputo (sin ser limitante) y de preferencia también, conocimientos de Comunicaciones Digitales.

TEMARIO TCP/IP

1.- INTRODUCCION

- Terminología
- Modelo de referencia ISO-OSI

2.- ARQUITECTURA TCP/IP

- Protocolos
- Topologías
- Arquitectura IP
- Arquitectura TCP
- Arquitectura UDP

3.- NOMBRES Y DIRECCIONES

- Nombres y Dominios
- Ejemplos de Nombres de Inter-Red
- Formateo de Direcciones
- Direcciones Clase A, Clase B, y Clase C
- Sub-Redes
  - Máscaras de Sub-Red
  - Direcciones Especiales
  - Identificación de Redes
  - Mensajes a Redes
  - Mensajes a Sub-Redes
  - Direcciones de Regreso
- Domain Name System
- Address Resolution Protocol (ARP)

4.- INTERNET PROTOCOL

- Funciones
- Mecanismos
- Proceso de Datagramas
- Relación al Modelo OSI.

5.- TRANSMISSION CONTROL PROTOCOL

- Conceptos de TCP
- Mecanismos de TCP
- Cabecera de TCP
- Rendimiento
- Relación al Modelo OSI

6.- TRABAJANDO CON TCP/IP

- FTP (File Transfer Protocol)
- Comandos de TCP
- Telnet
- NFS, RPC Y NIS
- Correo Electrónico
- SMNP (Simple Management Network Protocol)

TCP / IP
ARQUITECTURA, PROTOCOLOS E IMPLEMENTACION

1.- INTRODUCCION

Mayo de 1996.

1.- INTRODUCCION

Enlaces TCP/IP

TCP/IP (**T**ransmission **C**ontrol **P**rotocol / **I**nternet **P**rotocol) es una familia de protocolos para interconectar computadoras de diversas naturalezas. Lo que se ha venido observando al paso de los años es que TCP/IP es un protocolo fuerte que no se ha visto desplazado por otros protocolos como se pensaba. Originalmente TCP/IP se creó por pedido del Pentágono y se usó en su principio para la red ARPA que interconectaba a varias universidades y centros de investigación relacionados con el Gobierno de los Estados Unidos.

Es interesante hacer notar que ARPA después derivó a ser **Internet**, la red más grande del mundo, **Internet**, que cuenta con millones de nodos.

La evolución de TCP/IP se remonta a los primeros años de la década de los 80 y según fué desarrollándose, se fué estandarizando.

La forma en que se desarrolla hoy en día, es por medio de un Comité llamado IAB, que está formado por personas altamente calificadas, así se publican trimestralmente las especificaciones de los protocolos o sus revisiones.

Existe una diferencia primordial en estos estándares y es que, para que un protocolo reciba el nombre de estándar, debe haberse probado exitosamente en redes reales durante varios meses, lo que garantiza la funcionalidad del mismo.

Desde su planeación, **TCP/IP se pensó para ser independiente del medio físico de enlace**, es esto precisamente lo que ha hecho que sea un protocolo ampliamente usado en enlaces de redes locales entre si, o bien, con redes amplias WAN.

Los ambientes que usan TCP/IP se basan en que cada elemento de la red tenga su **dirección IP**. El propósito de lo anterior es identificar de forma única a cada elemento del conjunto, para IP cada uno de los nodos de la red.

A los nodos que son computadoras se les denomina *hosts*, bajo la terminología de TCP/IP, y los Gateways son el equipo que tiene realmente funciones de ruteador, es importante notar que la connotación de estos términos bajo TCP/IP es diferente a la que normalmente nos hemos referido.

Las direcciones de IP tienen como objetivo:

1. Identificar de manera única cada nodo de una red o un grupo de redes.

2. Identificar también a miembros de la misma red.

3. Direccionar información entre un nodo y otro, aún cuando ambos estén en distintas redes.

4. Direccionar información a todos los miembros de una red o grupo de redes.

IP hace el trabajo de llevar y traer paquetes entre todas las redes que estén unidas y usando este protocolo, pero no nos garantiza que éstos lleguen a su destino. Para remediar esto, está TCP tampoco nos regula el flujo de paquetes.

TCP tiene funciones importantes, las que se mencionan a continuación:

1. - Secuenciamiento y reconocimiento de paquetes.

2. - Control del flujo de la información.

TCP partirá en paquetes la información y la enviará. A cada paquete se le asigna un número. El reconocimiento significa que cuando un nodo recibe varios paquetes, debe informar al que los está enviando que efectivamente los está recibiendo, de esta manera se logra un cierto control sobre la información que se está transmitiendo.

El hecho de poder enviar los paquetes significa que antes de poder establecer comunicación entre dos nodos, es necesario un *handshake* que es el momento en que el receptor y el transmisor se ponen de acuerdo para poder establecer la comunicación.

Existe una serie de tareas que TCP/IP realiza y que son de suma utilidad, tales como la emulación de terminales, para poder entrar a una diversidad de equipos, así como la transferencia de archivos entre computadoras.

Dentro de las aplicaciones cliente-servidor, una de las que mayor auge ha tenido ha sido la de bases de datos, teniendo por un lado el equipo corriendo al manejador de bases de datos, y por otro, a muchas PC's conectándose a él a través de diversas herramientas e interactuando con la información.

Es importante recordar que las aplicaciones que corren en las PC's se denominan clientes y el equipo que tiene la base de datos se denomina servidor o *motor* de base de datos.

Como se desea poder realizar esa conexión entre clientes y servidores no importando si éstos están en la misma red o en redes distantes, la solución más sencilla es que ambos: clientes y servidores, se comuniquen usando TCP/IP, de hecho es la forma en que se ha comercializado. Oracle, Sybase, Gupta, Informix y varios más, usan TCP/IP como su forma de transporte de datos y comandos entre clientes y servidores.

Terminología

Como en la mayoría de las disciplinas técnicas, en el terreno de las comunicaciones se cuenta también con un lenguaje propio.

Bytes y Octetos

En el medio de la computación es muy comúnmente utilizada la palabra *byte* para referirse a una cantidad de *8 bits*. Sin embargo, esta palabra también se utiliza para definir a la unidad más pequeña direccionable en una computadora. Una solución a este problema es el empleo de la palabra *octeto* para denotar una cantidad de *8 bits*.

Big Endians y Little Endians

La característica de almacenamiento de datos en una computadora se puede clasificar en dos ramas, *Big Endians* cuando la computadora almacena los datos de tal forma que siempre queda al inicio el *byte más significativo*; y *Little Endians* en el caso en que queda al principio el *byte menos significativo.*

Protocolos, Pilas y Conjuntos

Un ***Protocolo*** es un conjunto de reglas que gobiernan las acciones de comunicación.

Una ***Pila de Protocolos*** es un conjunto subdividido de protocolos que interactúan con el fin de proveer comunicación entre diversas aplicaciones.

Un ***Conjunto de Protocolos*** es una familia de protocolos que opera de manera conjunta a efecto de crear una plataforma consistente.

Host, Ruteador y Otros conceptos

Un ***Host*** es una computadora central que puede tener uno o más usuarios, un Host con capacidad de soporte a TCP/IP puede fungir como último punto de una comunicación.

Un ***Ruteador*** especifica los caminos que deben seguir los datos a través de una red. Anteriormente se adoptaba el término *Gateway* para definir lo que hoy se conoce comercialmente como *Ruteador,* término que hoy en día se emplea para hacer referencia a un sistema que efectúa cierta clase de traducción de protocolos.

Un ***Nodo o Elemento de Red***, es toda aquella entidad en la red, sin importar si se trata de un Host, Ruteador o algún otro dispositivo.

MODELO DE REFERENCIA ISO-OSI

Las tecnologías que el hombre ha inventando, para comunicarse, siempre han seguido ciertas normas o reglas para su aceptación en un grupo social que puede ir desde una pequeña comunidad hasta todo una gran sociedad. En la época moderna los normas que rigen a las comunicaciones deben tener carácter universal. Hablando de comunicaciones digitales las normas o reglas universales están representadas por el modelo **ISO-OSI.**[1]

El modelo OSI estructura en siete niveles o capas, el fenómeno global de la comunicación, es un marco hoy en día obligado y universalmente aceptado.

Las normalizaciones en redes locales tratan de encuadrarse dentro de este modelo. Además, las redes locales deberán acoplarse a las redes públicas de área extendida, actualmente existentes y en permanente expansión.

El modelo para la interconexión de sistemas abiertos, **ISA**[2] u **OSI**[3] se ha convertido en una referencia obligada para todo lo relacionado con la intercomunicación de computadoras.

Frecuentemente, en artículos o descripciones relacionadas con este tema, se encuentra un dibujo de la "torre" de siete niveles y un enunciado somero y habitualmente poco claro, de las funciones y cometidos de cada uno de ellos.

---

1. International Standar Organization - Open System Interconection
2. Siglas en español
3. Siglas en inglés, Open System Interconection

La estructura jerarquizada de este modelo se explica a continuación:

Por ejemplo, si se analiza una estructura humana de comunicación de mensajes, se puede describir ésta mediante un determinado número de niveles de abstracción de los distintos fenómenos y tareas que se producen.

Imagínese una comunicación donde el mensaje emitido tiene un nivel cognoscitivo relacionado con cualquier materia o asunto, de manera que, para que el receptor pueda entenderlo debe estar al corriente de la materia de que se trate. (Figura 2-1).

Este mensaje ha de ser codificado en un lenguaje natural concreto, por ejemplo inglés o español.

Además para poder transferir el mensaje al receptor, será necesario utilizar algún medio físico concreto (ondas sonoras, papel, etc.) y elegir un método acorde con este medio.

En el lugar del receptor el proceso sería el mismo, pero en orden inverso.

En cada estación debe haber una comunicación interna entre niveles, de arriba a abajo en el emisor y de abajo hacia arriba en el receptor, lo que obliga a la existencia de una interface adecuada entre niveles consecutivos.

Por ejemplo:

Si para N1 se elige el método escrito en un determinado alfabeto será necesario en el emisor, alguien que sea capaz de escribirlo y en receptor alguien que sea capaz de interpretarlo.

La idea que se pretende hacer quedar clara es que, tiene que haber una coherencia entre cada par de niveles. Por lo tanto, si el lenguaje elegido es el castellano, éste debe ser el mismo en ambas estaciones.

Esto significa que existen entre niveles homólogos unos *protocolos de pares*, es decir, un conjunto de reglas que permiten relacionar horizontalmente a dos entidades de comunicación.

A nivel cognoscitivo, de nada sirve al oyente de un mensaje en castellano, tener un magnífico oído y un buen conocimiento de la lengua si no entiende el tema del que se está hablando.

En una comunicación estratificada en niveles, la comunicación real se hace en niveles consecutivos dentro de una misma estación y solamente a través del medio físico en la comunicación entre dos estaciones; aunque desde el punto de vista lógico es más interesante hablar de la comunicación entre niveles homólogos mediante protocolos de pares.

Estructura General del Modelo

Desde el punto de vista de ISO, un sistema abierto es el conjunto de una o más computadoras con su software, periféricos y terminales, capaces de procesar y transmitir información.

Es un modelo que está relacionado con las funciones que tienen que ser desarrolladas por el hardware y el software para obtener una comunicación fiable e independiente de las características específicas de la máquina. Es decir, está pensada para la interconexión de sistemas heterogéneos.

El sistema está compuesto por siete niveles, mediante los cuales dos sistemas informáticos se comunican entre si.

Con frecuencia, quienes inician el estudio del modelo se preguntan la razón de que sean siete niveles en la arquitectura y no un número mayor o menor.

Si se volviera al ejemplo anterior (de la comunicación humana), se vería que los tres niveles mediante los que se describe, podrían ser ampliados pensando por ejemplo, en la naturaleza del medio de comunicación, si se han elegido tres es porque así queda suficientemente bien dividido y descrito el problema.

De la misma manera, el grupo de estudio que elaboró el modelo OSI pensó que la división en siete niveles era una buena propuesta, pero eso no significa que tenga que ser necesariamente así.

No obstante, este modelo ha sido plenamente aceptado tanto por fabricantes como por usuarios.
Las características del modelo podrían resumirse de la siguiente forma:

- Cada nivel está representado por una entidad de nivel. Los niveles equivalentes en dos sistemas diferentes se comunican de acuerdo con unas reglas y convenios denominados *protocolos de nivel o protocolos de pares.*

- Cada nivel proporciona un conjunto definido de servicios al nivel superior y a su vez utiliza los servicios que le proporciona el nivel inmediatamente inferior.

- La comunicación se realiza a través de los niveles inferiores, siendo el protocolo de pares una abstracción lógica de relación entre las dos entidades comunicantes.

- Si un nivel N desea transmitir una unidad de datos a otro nivel N homólogo en otro sistema informático, se la pasará al nivel inmediatamente inferior, el cual le añadirá información delimitadora propia y a su vez pasará está información a su nivel inmediatamente inferior.

En el sistema receptor cada nivel separará la parte del mensaje que le corresponde y pasará el resto a su nivel inmediatamente superior, que hará lo propio. Así el mensaje del nivel N es como si viajara horizontalmente hasta su nivel homólogo en recepción.

Los Siete Niveles

Los tres primeros niveles tratan los protocolos asociados con la red de conmutación de paquetes utilizada para la conexión y pueden agruparse dentro del llamado bloque de transmisión.

El nivel cuatro enmascara a los niveles superiores los detalles de trabajo de los niveles inferiores dependientes de la red, y junto con ellos forma el bloque de transporte.

Los tres niveles superiores, del quinto al séptimo, son los usuarios del bloque de transporte y aislan la comunicación de las características específicas del sistema informático.

A continuación se analizan uno por uno los diferentes niveles, estudiando sus funciones y características.

- EL NIVEL SIETE: APLICACION

Este nivel se preocupa de proporcionar un conjunto de servicios distribuidos a los procesos de aplicación de los usuarios. El usuario se comunicará directamente con este nivel a través de la correspondiente interface o agente de usuario.

Actualmente se están desarrollando una serie de normas y recomendaciones tendientes a tipificar cada uno de estos servicios o aplicaciones distribuidas.

Entre los más conocidos podemos citar:

- ◊ Servicio de mensajería (correo electrónico), servicio de almacenamiento y recuperación de documentos, servicio de directorio, etc.

↳ EL NIVEL SEIS : PRESENTACION.

Este nivel se ocupa de la representación de los datos usados por los procesos de aplicación del nivel siete. Por lo tanto, si es necesario, realizará la transformación de los datos que reciba de o para el nivel de aplicación. Esto en el caso de que el proceso originador y el receptor tuvieran versiones de datos sintácticamente diferentes, pero también puede darse el caso de que, para una determinada aplicación distribuida exista un conjunto de caracteres normalizados diferentes de los del originador y el receptor, en cuyo caso, los niveles de presentación respectivos deberían de hacer las transformaciones necesarias.

Otra función que se puede encargar al nivel seis, es la de velar por la seguridad de los datos, siendo responsable de la encriptación de mensajes confidenciales antes de su transmisión. La función inversa será realizada por el nivel de presentación del sistema receptor.

↳ NIVEL CINCO: SESION:

Su función es establecer y gestionar un camino de comunicación entre dos procesos del nivel de aplicación. Este nivel establece una sesión y se encarga de controlar la comunicación y sincronizar el diálogo.

La información que se envía se fracciona en pedazos y se generan unos puntos de sincronización. En caso de interrumpirse la sesión por alguna falla en la comunicación, los datos pueden ser recuperados y se conoce con precisión por ambos interlocutores hasta qué punto de sincronización la comunicación fue correcta.

Al reanudarse la sesión no será necesario transmitir de nuevo toda la información, sino solamente a partir del punto donde se quedó el último paquete de información válido.

En una sesión hay un diálogo entre máquinas, entre procesos y el protocolo debe regular quién "habla", cuándo y por cuánto tiempo.

Estas reglas necesitan ser acordadas cuando la sesión comienza. Este nivel también es responsable de dirigir el diálogo entre las entidades de nivel de presentación.

Para ello, cuando se establece una conexión de sesión, es necesario que ambos niveles cinco se pongan de acuerdo sobre el papel a desempeñar por cada uno de ellos en la comunicación.

↳ NIVEL CUATRO: TRANSPORTE.

Este nivel es responsable de una transferencia de datos transparente entre dos entidades del nivel de sesión, liberando a dichas entidades de todo lo referente a la forma de llevar a cabo dicho transporte.

Los protocolos que maneja este nivel suelen llamarse *protocolos end-to-end*, o protocolos entre puntos finales, debido a que este nivel se encarga de realizar una conexión lógica entre dos estaciones de transporte de los sistemas informáticos que quieren comunicarse, independientemente de donde se encuentren éstos.

Este nivel puede multiplexar varias conexiones de transporte dentro de una única conexión de red, o puede por el contrario, repartir una conexión de transporte entre varias conexiones de red.

↳ NIVEL TRES: RED.

Este nivel enmascara todas las particularidades del medio real de transferencia. Es el responsable del encaminamiento de los paquetes de datos a través de la red. Cada vez que un paquete llega a un nodo, el nivel tres de ese nodo deberá seleccionar el mejor enlace de datos por el que envíe la información.

Las unidades de datos de este nivel son los paquetes de datos que deberán ir provistos de la dirección de destino. Por lo tanto, entre las funciones fundamentales del nivel de red se encuentran las de establecer, mantener y liberar las conexiones necesarias para la transferencia de los paquetes de datos.

Además, son funciones de este nivel la definición de la estructura de datos de los paquetes, las técnicas de corrección de errores, la entrega en secuencia correcta al nivel de transporte de los paquetes recibidos, así como otras de reiniciación y control de flujo.

Para las redes públicas de transmisión de datos la CCITT ha definido la norma X.25 que describe los protocolos de comunicación para los niveles uno, dos y tres del modelo de referencia de ISO.

↳ NIVEL DOS : ENLACE.

Un enlace de datos se establece siempre entre dos puntos físicos de conexión del sistema. En el caso de una red de datos de conmutación de paquetes, el nivel de enlace es responsable de la transferencia fiable de cada paquete al nivel de red.

La CCITT ha definido dentro de la recomendación X.25 un subconjunto del protocolo **HDLC**[4] como protocolo del nivel de enlace.

☜ NIVEL UNO: FISICO.

Este nivel engloba los medios mecánicos, eléctricos, funcionales y de procedimiento para acceder al medio físico. Es el encargado de la activación y desactivación física de la conexión. Ciertos protocolos estándar clásicos como el X.21 y V.24 son utilizados en el nivel físico.

Es muy importante recalcar que el modelo ISO-OSI es un estándar universal, pero mas que un estándar tecnológico, representa un marco de referencia. Esto es, la mayoría de los fabricantes de hardware y Software sus productos no cumplen con las funciones y límites de cada nivel, pero compararán sus productos con los niveles del modelo, argumentando sus ventajas y funciones respecto al modelo.

EL modelo ISO-OSI, proporciona un lenguaje universal entre los especialistas del medio de la interconexión de equipo de cómputo, para que hablen un "mismo idioma" y puedan comparar cualquier producto o tecnología respecto a dicho modelo.

---

[4] High Level Data Link Control

También es saludable mencionar que los grandes centros de investigación de la industria están trabajando fuertemente para lograr una tecnología comercial que se apegue estrictamente al modelo, dicha tecnología es reconocida como OSI, pero en la actualidad no deja de ser un interesante proyecto, ya que la parte comercial tiene sus ojos puestos en tecnologías ya ampliamente probadas como TCP-IP y las nuevas tecnologías que manejan un gran ancho de banda como ATM, Frame-Relay, etc.

Con el marco de referencia anterior, es importante hacer un nuevo análisis de los tres estándares que dominan en las interfaces de red.

TCP/IP

TRANSMISSION

CONTROL

PROTOCOL

INTERNET

PROTOCOL

Notas:

TCP/IP

OBJETIVO:

INTEGRACION

DE

AMBIENTES HETEROGENEOS

Notas:

TCP/IP

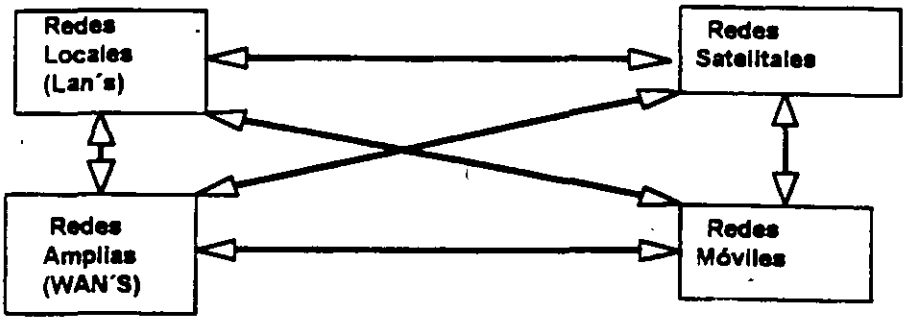
TERMINOLOGIA

- Bytes, Octetos.
- Big Endians y Little Endians
- Protocolo
- Pila y Conjunto de Protocolos
- Host
- Ruteadores
- Gateway

Notas:

TCP/IP

OBJETIVO:



Notas:

1-1

TCP/IP

HISTORIA Y GENERALIDADES

1969 Empieza el trabajo con ARPANET.

1972 Primera demostración de ARPANET.

1976 Empieza la implementación de TCP/IP.

1980 Se libera TCP/IP con Unix 4.1 BSD (Berkley).

1982 TCP/IP reemplaza a NCP en ARPANET.

1988 Se publica TCP/IP con especificaciones Militares Estándares.

1984 Se separa Milnet de ARPANET.

1989-90 Más de 200 proveedores soportan TCP/IP, más de 600,000 sistemas.

Notas:

TCP/IP

HISTORIA Y GENERALIDADES

¿Por qué TCP/IP?

- Aceptado ampliamente por los centros de investigación y desarrollo en. todo el mundo.
- Desde 1984 fue requerido por el gobierno y la defensa de E.U.A.
- Los sistemas basados en Berkley-Unix lo provee.
- SUN (SUN Microsystem) le da a TCP/IP un posicionamiento comercial.
- Los ambientes más técnicos adoptan TCP/IP.
- Son los únicos protocolos realmente abiertos y estándares disponibles actualmente.
- Predecesores de los protocolos ISO.

Notas:

TCP/IP

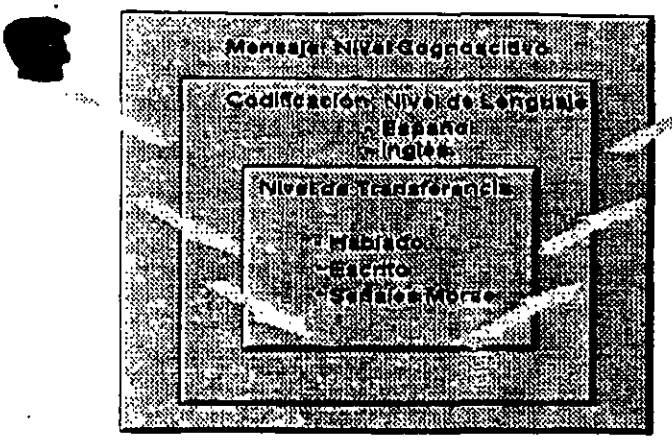
MODELO OSI

CAPA	NOMBRE
7	Aplicación.
6	Presentación.
5	Sesión.
4	Transporte.
3	Red.
2	Enlace.
1	Físico.

Notas:

TCP/IP

NORMALIZACION



Notas:

TCP/IP

ESTRUCTURA GENERAL DEL MODELO OSI

7	Aplicación
6	Presentación
5	Sesión
4	Transporte
3	Red
2	Data Link
1	Físico

Notas:

TCP/IP

MODELO OSI
NIVEL 1

NIVEL FISICO
Define cómo será transmitida la información binaria

- Niveles de Voltaje
- Modulación
- Velocidad de Transmisión

Notas:

TCP/IP

MODELO OSI
NIVEL 2

NIVEL DE DATA LINK

Checa errores de transmisión a nivel de FRAMES y presenta al nivel tres una línea libre de errores.

Define métodos de acceso al medio físico

Notas:

TCP/IP

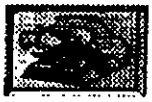
MODELO OSI
NIVEL 3

NIVEL DE RED

Agrupa en paquetes y define qué camino toma cada paquete (enrrutamiento)

Notas:

TCP/IP



MODELO OSI
NIVEL 4

NIVEL DE TRANSPORTE

Verifica que los paquetes lleguen en el orden requerido (secuencial).

Notas:

TCP/IP

MODELO OSI
NIVEL 5

NIVEL DE SESION

- Define el procedimiento para iniciar la comunicacion entre dos procesos a nivel de presentacion.
- Usualmente este nivel es la interface del usuario (y del software), de la RED.

Notas:

TCP/IP

MODELO OSI
NIVEL 6

NIVEL DE PRESENTACION

Realiza transformaciones en la información

- Conversión de Código
- Compresión
- Encripción
- Conversión de Formatos de Archivo

Notas:

TCP/IP

MODELO OSI
NIVEL 7

NIVEL DE APLICACION

Provee servicios a los usuarios de la RED

- Correo Electronico
- Transferencia de Archivos
- Emulacion de Terminales

Notas:

105

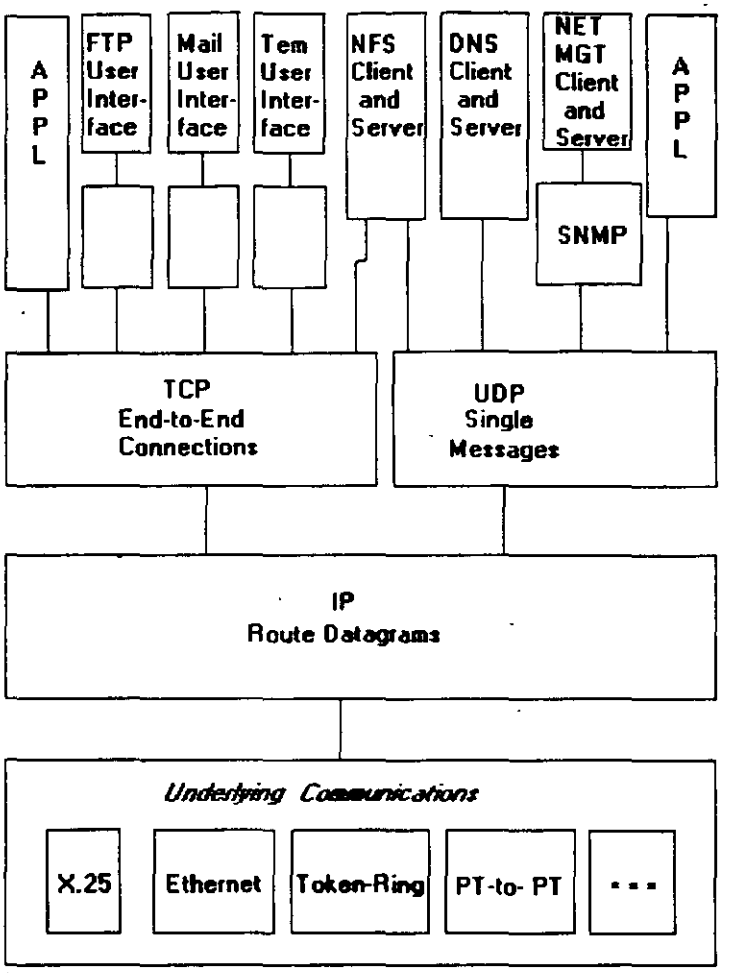
TCP / IP
ARQUITECTURA, PROTOCOLOS E IMPLEMENTACION

2.- ARQUITECTURA TCP/IP

Mayo de 1996.

2.- ARQUITECTURA TCP/IP

Protocolos



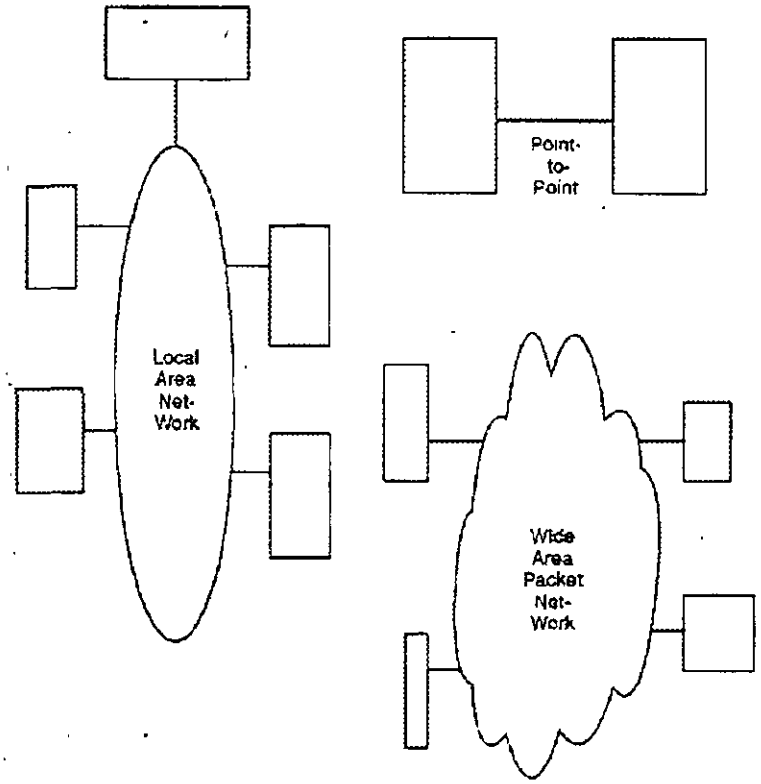
La figura **(Fig 2.1)** muestra la manera en que se complementan las partes del Conjunto de Protocolos TCP/IP. A pesar de que las interfaces al usuario para las aplicaciones *FTP, Telnet* y *DNS* han sido estandarizadas de manera formal, la mayoría de los proveedores ofrecen una colección de comandos que se encargan de copiar las interfaces al usuario de *UNIX Berkeley Software Distribution*.

Los módulos *FTP, SMTP,* y *Telnet* se comunican con sus clientes mediante conexiones TCP confiables. La mayoría de los Servidores NFS intercambian mensajes de *UDP* con sus clientes, a pesar de la escasa existencia de implementaciones *NFS* creadas especificamente para TCP.

Los protocolos *DNS* proporcionan servicios de directorio en redes TCP/IP. Los servidores *DNS* excluyen a la mayoría de las transacciones por medio de mensajes de *UDP*, pero ocasionalmente cambian a TCP cuando es necesario mover una mayor cantidad de datos.

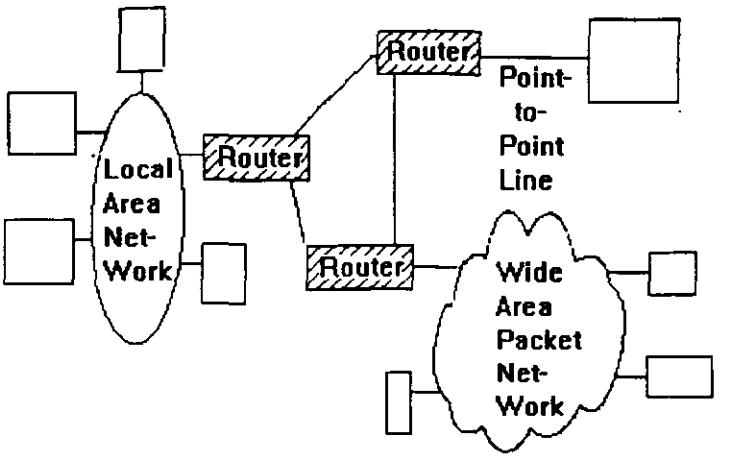
Topologías

El Conjunto de Protocolos de TCP/IP puede emplearse en standalone tanto en redes LAN como en redes WAN, así como en Inter-Redes complejas creadas a base de la unión de redes sencillas.



La figura **(Fig 2.2)** muestra las redes en standalone. Cualquier Host equipado con TCP/IP es capaz de comunicarse con otro mediante una línea *punto a punto* que puede ir a una red LAN o WAN.

En una Inter-Red, las redes se unen haciendo uso de un ruteador IP. La figura **(Fig 2.3)** muestra una Inter-Red implementada utilizando ruteadores IP para enlazar a una LAN, a una WAN y a un Host Remoto.



Además de ejecutar software IP, los ruteadores emplean típicamente un segundo protocolo para intercambiar información con otro, acerca de la situación actual de la Inter-Red a la que pertenecen.

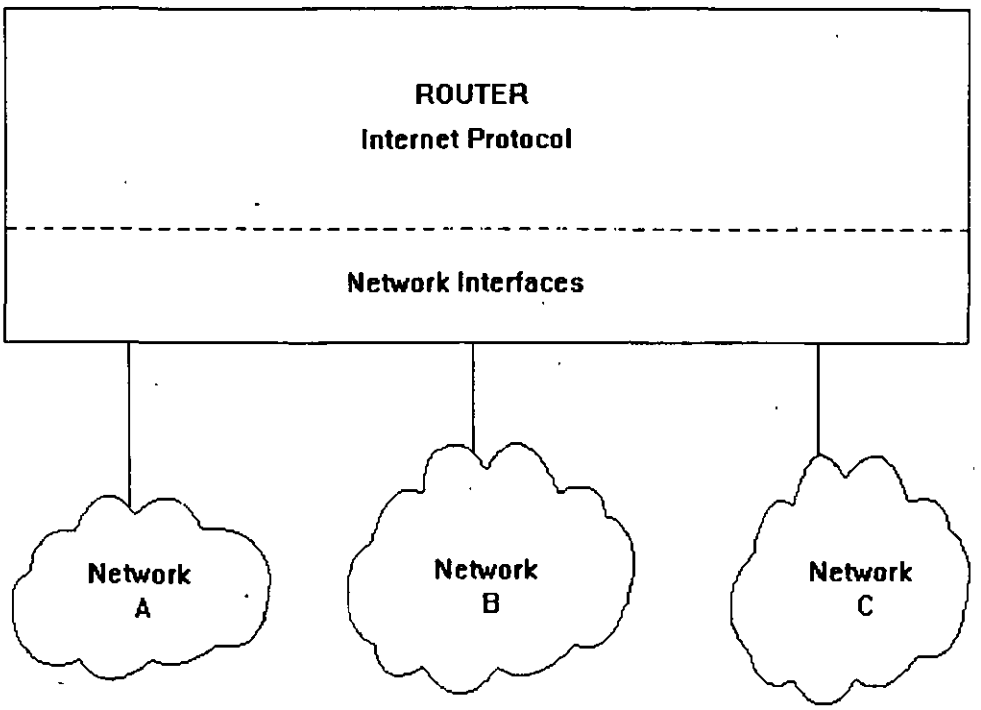
El amplio y competitivo mercado de ruteadores IP ha sido de gran utilidad para promover la arquitectura TCP/IP. Los proveedores de ruteadores están a la expectativa en la implementación de nuevas tecnologías LAN y WAN, ampliando las opciones de conectividad de sus clientes. La relación precio-desempeño de los ruteadores, ha disminuido de manera insistente en los últimos años.

En teoría, las Inter-Redes pueden tener topologías arbitrariamente mezcladas sin embargo, cuando la Inter-Red tiene una estructura coherente, resulta más fácil para los ruteadores el llevar a cabo su trabajo de manera óptima, y reaccionar rápidamente a una falla en alguna parte de la red, alterando las rutas de tal manera que los datagramas ev :en un *trouble-spot*.

Un diseño lógico y fácil de entender resulta de gran utilidad para los administradores de red en lo referente al diagnóstico, localización y reparación de fallas.

Arquitectura IP

El Software de Protocolo Inter-Red (IP) opera tanto en Host como en Ruteadores IP. En general, el Software IP permite a la computadora que lo ejecuta, funcionar como un Host IP, como un Ruteador IP, o como ambos a la vez. La mayoría de las compañías prefieren utilizar equipo especializado para ruteo en la unión de sus redes. Sin embargo, es conveniente tener la posibilidad de utilizar una computadora que regularmente no se utiliza, para ponerla en servicio como un ruteador.



La figura **(Fig 2.4)**, ilustra la arquitectura de protocolo de un *ruteador dedicado*. Debe observarse que no existe la necesidad de TCP debido a que las conexiones de las aplicaciones no inician ni terminan en el ruteador. Es evidente que un ruteador debe estar conectado al menos a dos redes.

Los productos modernos de ruteo están equipados con diversas interfaces de red que pueden ser configuradas con la combinación de conexiones que el cliente desee: Ethernet, Token Ring, conexión síncrona punto a punto, fibra óptica, etc.

Acciones de IP

Si el destino de un Datagrama no se encuentra en la misma red comó el Host fuente, el IP del Host direcciona el datagrama al ruteador local. Si éste no está conectado a la red destino, entonces el datagrama debe ser enviado a otro ruteador. Esta secuencia de operaciones continúa hasta que el datagrama llega a la red destino.

El IP decide el ruteo de la información mediante la detección de un destino remoto en una tabla de ruteo. El IP busca una entrada en la tabla de ruteo que corresponda al destino con la identidad del siguiente ruteador al cual se le relevará el tráfico de datagramas.

Información de la Tabla de Ruteo

En una Inter-Red pequeña y fija, las tablas de ruteo pueden ser introducidas y tener un mantenimiento en forma manual. En Inter-Redes más grandes, los ruteadores mantienen sus tablas actualizadas mediante el intercambio de información con los demás. Los ruteadores tienen la capacidad de descubrir dinámicamente hechos tales como:

- La conexión de una nueva red a la Inter-Red.
- La inhabilitación de un camino hacia una red destino
- La conexión de un nuevo ruteador a la Inter-Red, mismo que determina la ruta más corta hacia ciertos destinos.

No existe un estándar para el intercambio de información entre ruteador y ruteador.

Los ruteadores que están bajo el control de una organización se denominan *Sistemas Autónomos*. La organización tiene la opción de elegir cualquier protocolo para el intercambio de información que desee en su propio Sistema Autónomo. El protocolo de intercambio de información en ruteadores que se utiliza en un Sistema Autónomo, se conoce como ***Interior Gateway Protocol (IGP).***

El Protocolo de Información de Ruteo (RIP) es un IGP muy popular, debido a que es muy fácil de encontrar. Sin embargo, el nuevo protocolo *Open Shortest Path First (OSPF)* cuenta con un buen número de herramientas útiles. La disponibilidad y la popularidad de este protocolo está creciendo de manera insistente.

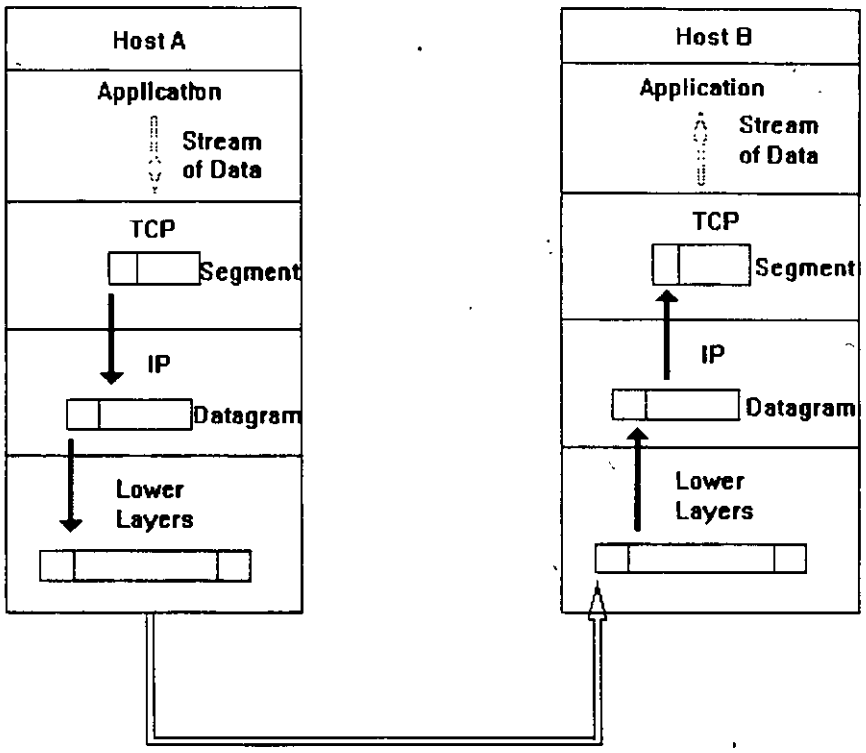
Algunos proveedores de ruteadores dan sus propios protocolos para el intercambio de información de ruteador a ruteador, así como soporte para protocolos estandarizados. Algunos proveedores tienen la habilidad de ejecutar diversos protocolos a la vez, de esta manera, sus ruteadores pueden intercambiar información con los demás con cualquiera de esos protocolos.

Arquitectura TCP

El TCP se implementa en Hosts. La Entidad de TCP en cada extremo de una conexión debe asegurar que los datos que se entreguen a su aplicación local lleguen:

- Precisos
- En secuencia
- Completos
- Sin datos duplicados.

El envío de una aplicación pasa una trama de bytes al TCP. Este se encarga de disgregar la trama en secciones y añadirle a cada sección una cabecera, formando *segmentos*. Posteriormente el TCP pasa cada segmento al IP para ser transmitido en un Datagrama **(Fig 2.5).**

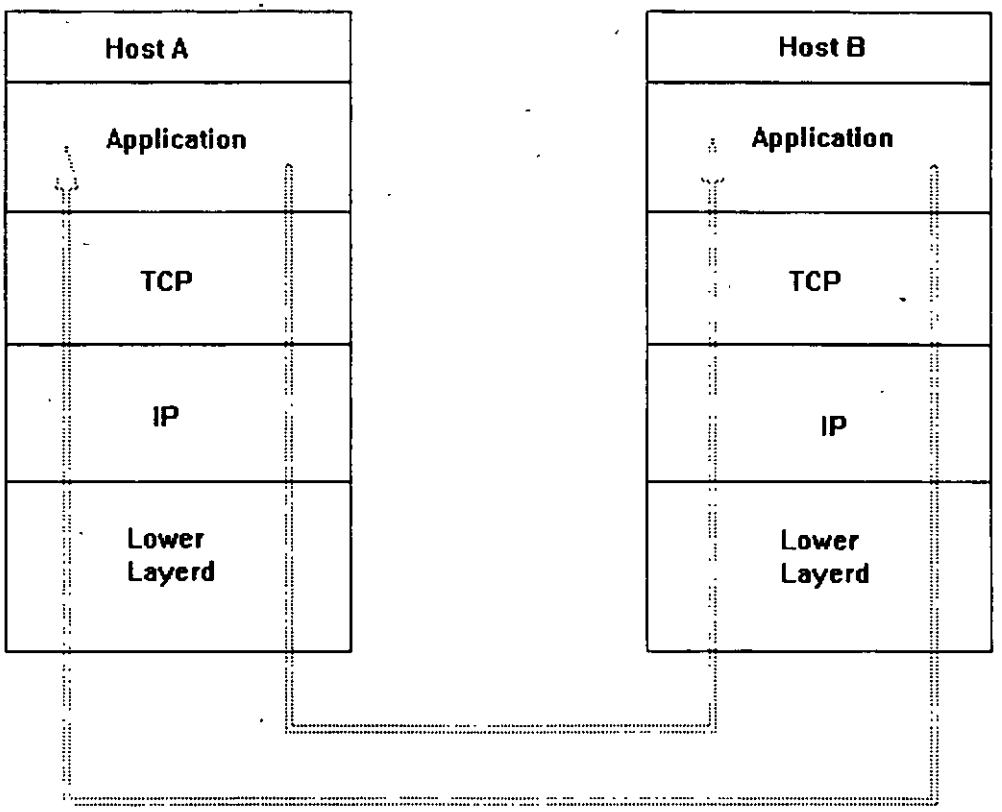


Un TCP receptor debe mantener informado al emisor acerca de la cantidad de información correcta que le ha llegado, mediante señales de reconocimiento (*AKCs*). Si el AKC de un segmento no llega en un intervalo de tiempo determinado, el TCP emisor vuelve a enviar ese segmento. A esta estrategia se le conoce con el nombre de *Retransmisión con Reconocimiento Positivo.*

Ocasionalmente una retransmisión provocará una reproducción en los segmentos entregados al TCP receptor.

El TCP receptor debe arreglar los segmentos que va recibiendo, en forma correcta, descartando todos aquellos que estén duplicados. De esta manera, el TCP entrega los datos a su aplicación de manera íntegra.

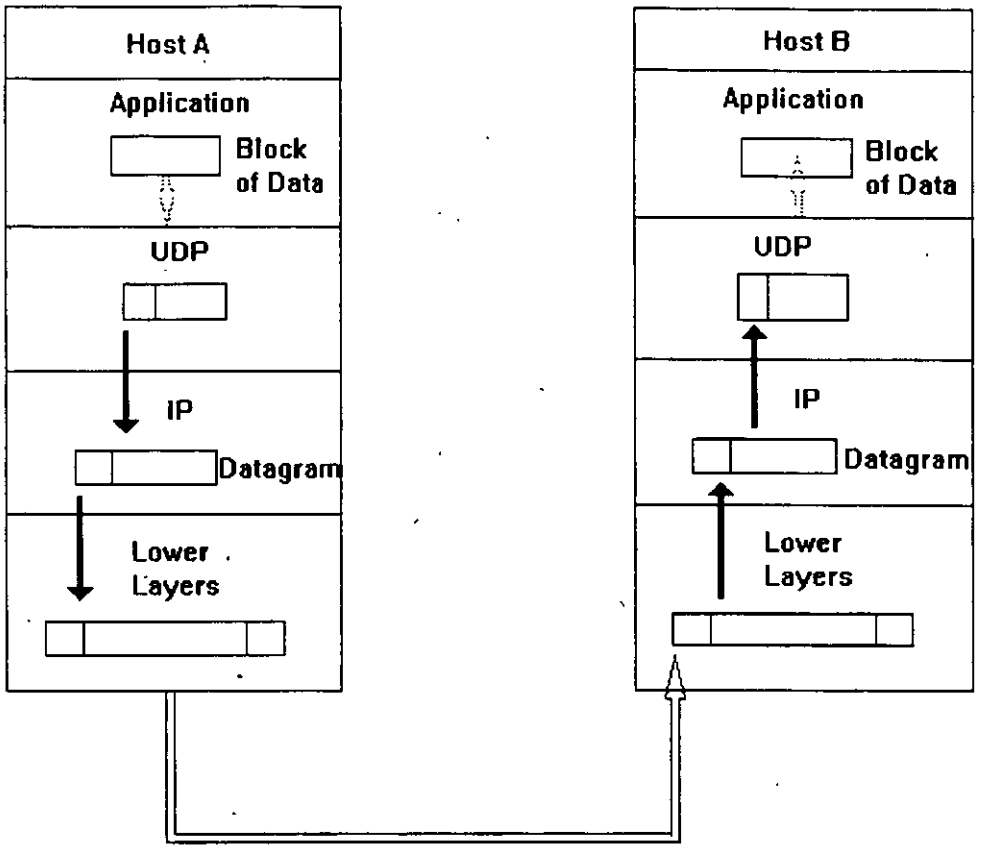
TCP es un protocolo completamente bilateral, es decir; los dos extremos de la conexión pueden enviar y recibir información al mismo tiempo, por lo que, de hecho se transmiten dos tramas de bytes. **(fig 2.6).**



Arquitectura UDP

El nivel UDP se implementa en Hosts finales. El UDP no garantiza una entrega íntegra, solo se limita a intercambiar información que confirme que los datos que se enviaron llegaron de una manera segura.

Una aplicación que se desee enviar vía UDP, tiene que pasar un *bloque* de datos al UDP, donde se le agrega una cabecera, formando así el ***Datagrama del Usuario (UD).*** Posteriormente el datagrama de usuario pasa al IP y se compacta en un datagrama IP.

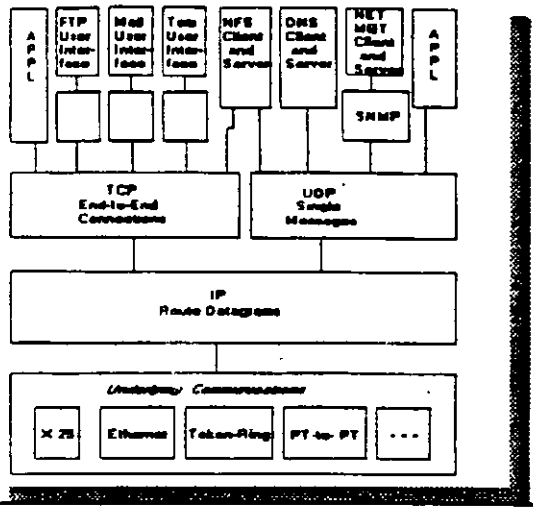


La figura **(Fig 2.7)** muestra como un bloque de datos se compacta y envía por UDP. Obviamente los mensajes de UDP deben ser enviados tanto por el emisor como por el receptor y el Host B puede estar concurrentemente en proceso de preparación de un bloque para enviar al Host A.

Una Aplicación participando en comunicaciones UDP debe enviar mensajes de recepción UD en cualquier momento. Solo depende de los clientes y de los servidores el conservar un registro de todas las relaciones de UD que se estén intercambiando.

TCP/IP

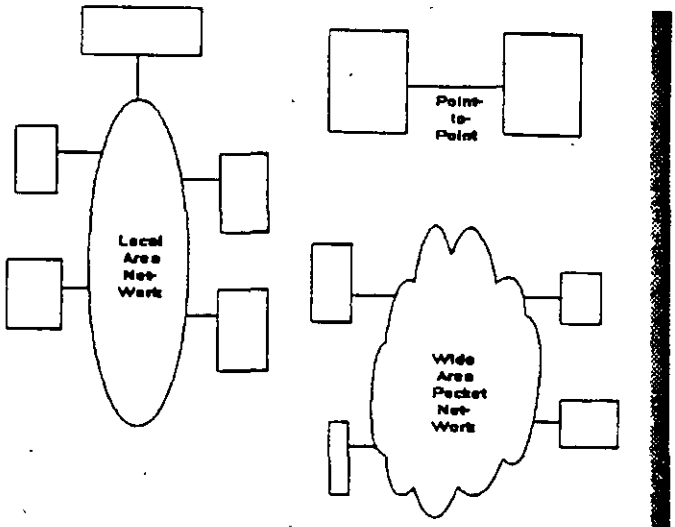
ARQUITECTURA
Protocolos:



Notas:

TCP/IP

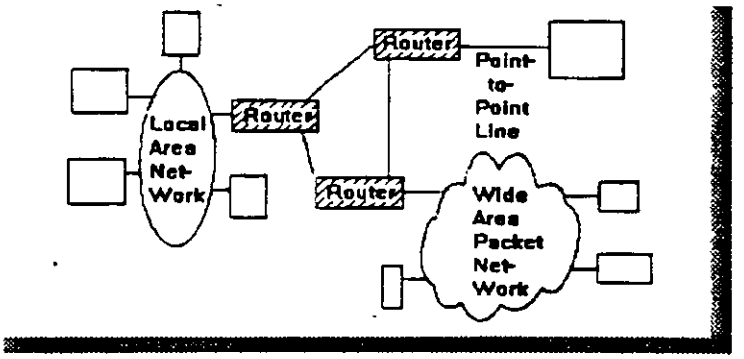
ARQUITECTURA
Topologías:



Notas:

TCP/IP

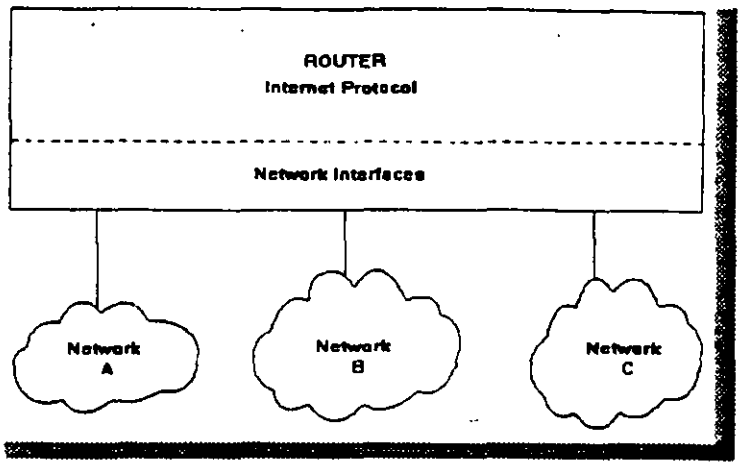
ARQUITECTURA
Topologías:



Notas:

TCP/IP

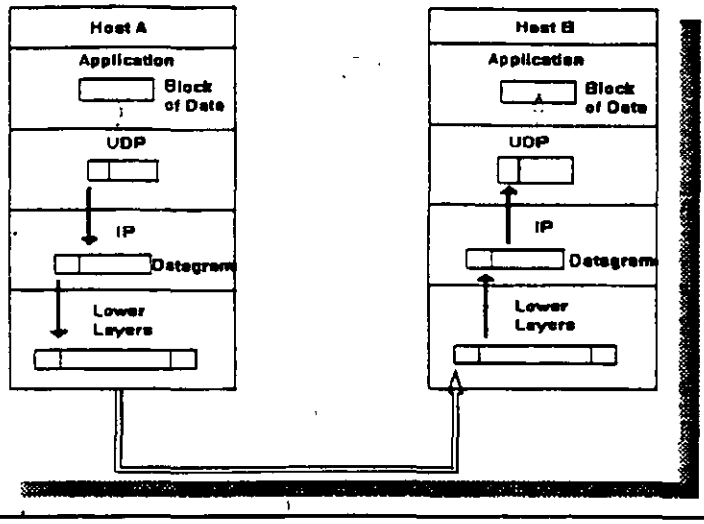
RUTEADOR:



Notas:

TCP/IP

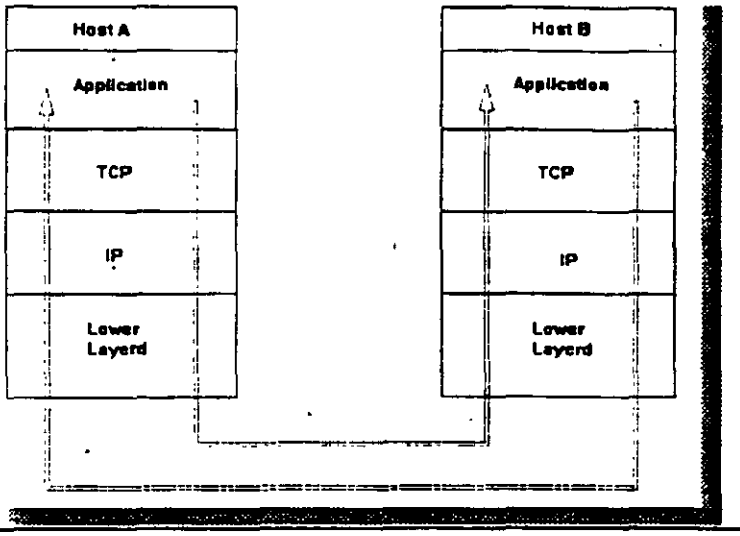
OPERACION TCP



Notas:

TCP/IP

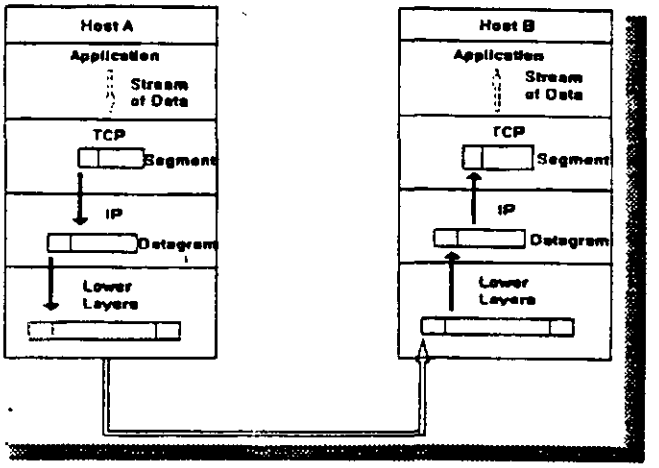
TCP/IP Protocolos Bilateral



Notas:

TCP/IP

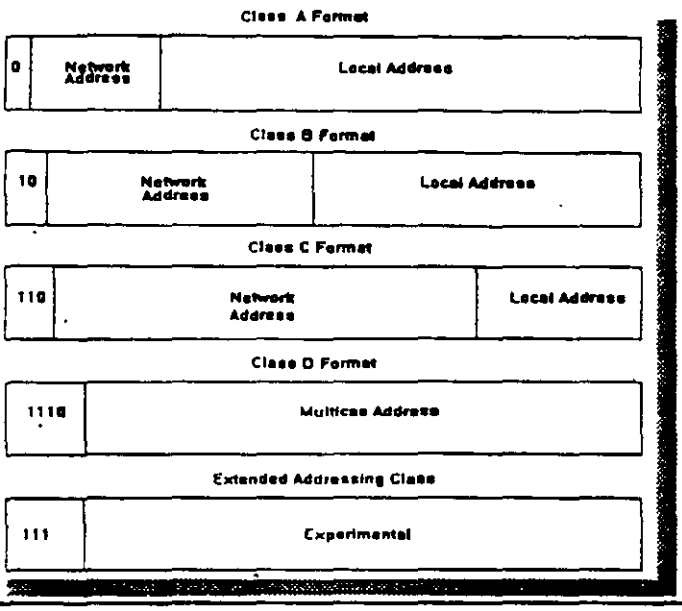
OPERACION UDP



Notas:

TCP/IP

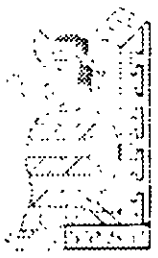
CLASE



Notas:

TCP / IP
ARQUITECTURA, PROTOCOLOS E IMPLEMENTACION

3.- NOMBRES Y DIRECCIONES



Mayo de 1996.

3.- NOMBRES Y DIRECCIONES

Nombres y Dominios

Tanto los nombres de la estructura de una Inter-Red como los de un sistema administrativo, son jerárquicos. Una Inter-Red está dividida en partes llamadas *Dominios*.

La responsabilidad de asignar nombres dentro de un dominio es tarea del administrador designado de ese dominio. Este administrador puede crear subdominios y delegar la autoridad de nombramiento a otro individuo de cada subdominio.

Ejemplos de Nombres de Inter-Red

Un nombre de Inter-Red puede describir a un sistema de una manera muy apropiada ya que su estructura se basa en la concatenación de etiquetas que hacen referencia a cada subdominio. El nombre de una Inter-Red puede ser escrito en mayúsculas o en minúsculas indistintamente:

TALLER.DIPLOM.DECFI.UNAM
unix.diplom.decfi.unam
Parte2.Diplom.Decfi.Unam
INTRO.DIPLOM.DECFI.UNAM

Es fácil entender la estructura jerárquica de estos nombres. Todas las divisiones de la Universidad se encuentran en el dominio UNAM de la Inter-Red. DECFI es el dominio de segundo nivel justo abajo del nivel UNAM. DIPLOM hace referencia a los diplomados impartidos por la DECFI de la UNAM y se encuentra como dominio de tercer nivel bajo DECFI. Finalmente el nombre del Host que identifica un sistema individual, inicia la cadena que define el nombre. Las partes adyacentes del nombre se separan por medio de puntos (.).

El tamaño límite de cada etiqueta es de 63 caracteres, pero el número máximo de caracteres por nombre es de 255 incluyendo los puntos separadores.

Formatos de Direcciones

El IP utiliza direcciones para identificar a los Host y para enviarles información. Cada Host debe tener asignada una dirección IP que pueda utilizarse en comunicaciones reales. El nombre de un Host es traducido a su dirección IP mediante la tabla de relación de Nombres y Direcciones.

Una dirección IP es un valor binario de 32 bits que define el espacio total de direcciones que es un conjunto de numero de direcciones. El conjunto total de direcciones IP contiene 2^{32} números.

La notación *punto* es la forma más popular de expresar una dirección IP de tal forma que los usuarios finales pueden leerlas y escribirlas fácilmente. Cada octeto de las direcciones se convierte en un número decimal y cada número se separa por un punto (.). Por ejemplo, la dirección de TALLER.DIPLOM.DECFI.UNAM en notación de 32 bit binarios será:

10000010 10000100 00001011 00011111
130.132.11.31

Cabe hacer notar que el número más grande que puede aparecer en una notación separada por puntos es 255, que corresponde al número binario 11111111.

Una dirección IP se constituye de dos partes:

- Dirección de Red
- Dirección Local

La Dirección de Red identifica la Red a la cual está conectado ese nodo, la Dirección Local a su vez, identifica al nodo de manera individual.

Dirección de Red	Dirección Local

Direcciones Clase A, Clase B y Clase C

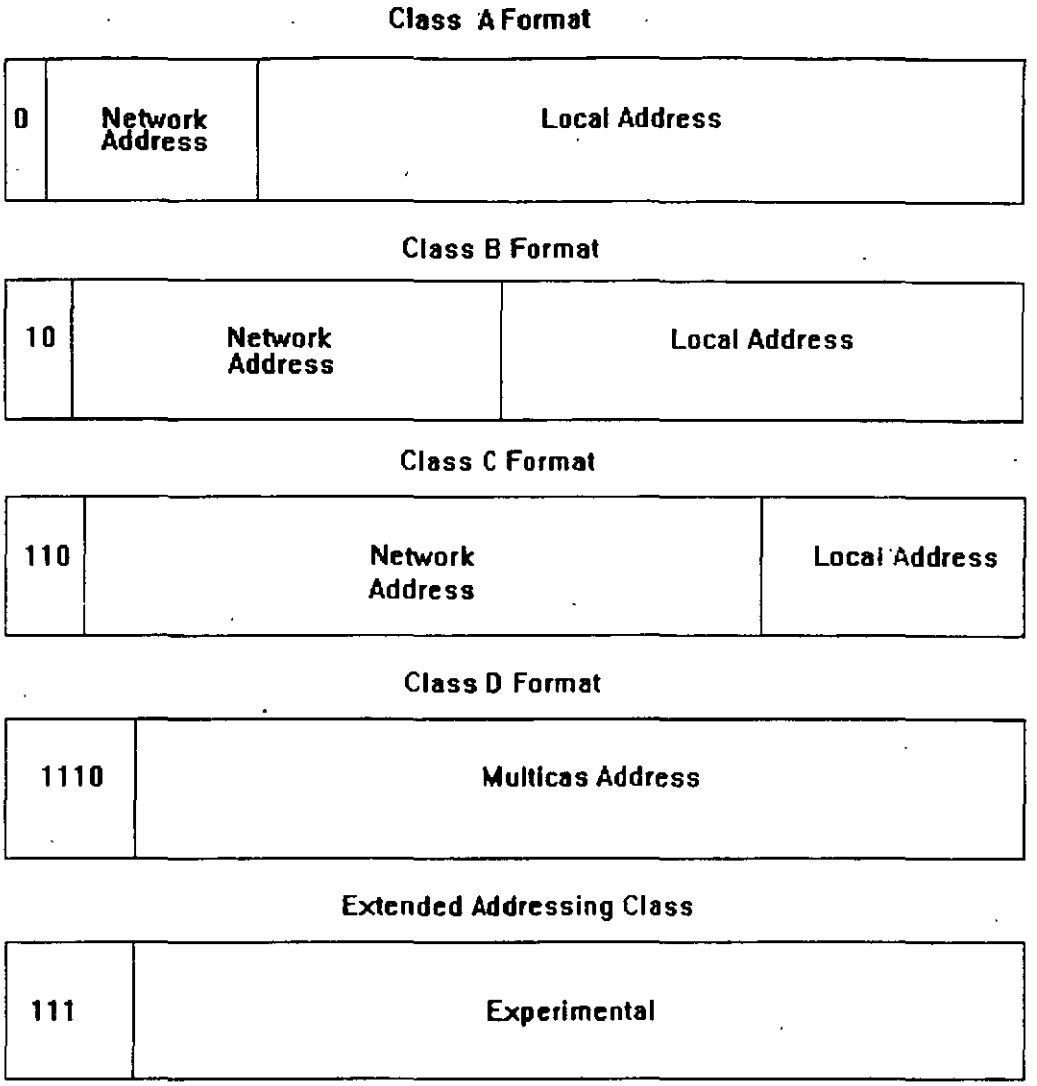
Las redes varían en tamaño. Existen tres formatos de direcciones diferentes para Inter-Redes que definen el uso dependiendo de su tamaño:

- Clase A para redes grandes
- Clase B para redes medianas
- Clase C para redes pequeñas.

Además de las clases A, B y C existen dos formatos de direcciones especiales, esto son: Clase D y Clase E. Los formatos de Clase D se utilizan para un *Multicasting* de IP que se emplea para distribuir un mensaje a un grupo de sistemas dispersos a través de la Inter-Red. La Clase E reserva su formato de direcciones para uso experimental exclusivamente.

Los primeros cuatro bits de cada dirección determinan su clase:

BITS INICIALES	CLASE
0xxx	A
10xx	B
110x	C
1110	D
1111	E



(Fig. 3.1)

Sub-Redes

Un administrador que desarrolla una implementación que cuenta con una dirección de Red Clase A o Clase B entiende la implicación de una complicada interconexión de Redes LAN y WAN. Es por eso que resulta práctico dividir en partes el espacio de direcciones de tal forma que corresponda a la estructura de la Red como una familia de Sub-Redes. Para llevar a cabo esto, es necesario descomponer la parte local de la dirección de la siguiente manera

Dirección de Red Dirección de Sub-Red Dirección de Host

La asignación de la dirección de Sub-Red frecuentemente se hace en un byte límite, un administrador que implementa direcciones Clase B como 156.33 debe utilizar su tercer byte para identificar las Sub-Redes, por ejemplo:

156.33.1
156.33.2
156.33.3

El cuarto byte será utilizado para identificar a los Hosts de manera individual dentro de una Sub-Red. Por otro lado, un administrador que implementa direcciones Clase C sólo tiene un espacio de dirección de un byte y deberá utilizar cuatro bits para las direcciones de los Host.

Máscaras de Sub-Red

Una máscara de Sub-Red es una secuencia de 32 bits que cubre con unos (1s) las zonas correspondientes a la red y a la Sub-Red, y cubre con ceros (0s) la zona que le corresponde a la dirección del Host.

El tráfico de información se rutea hacia un Host, considerando las partes de Red y Sub-Red de su dirección IP. Es sencillo decir que tanto de una dirección corresponde a la dirección de red debido a los formatos estrictamente definidos para Clase A, Clase B y Clase C.

A efecto de reconocer cualquier tipo de campo, con un tamaño arbitrariamente elegido para la Sub-Red, se creó un parámetro de configuración denominado *Máscara de Sub-Red*. Consta de una secuencia de 32 bits. Los bits que incluyen a las direcciones de Red y de Sub-Red, se restablecen con 1.

Por ejemplo, un administrador de una Red Clase B con dirección 156.33 ha elegido hacer uso del tercer byte de todas las direcciones a fin de identificar las Sub-Redes, por lo tanto, la Mascara de Sub-Redes será:

11111111 11111111 11111111 00000000

La máscara de Sub-Red puede ser expresada de las siguientes maneras:
En notación de unos y ceros (1s y 0s):

1111111 11111111 11111111 00000000

Se puede expresar en notación *hexadecimal* como:

ffffff00

o alternativamente, en notación *punto* como:

255.255.255.0

Los Ruteadores que están conectados directamente a una Sub-Red se configuran con la máscara para la Sub-Red. Es común el uso de una sola máscara de Sub-Red a través de toda una Internet de una Corporación.

Si una Red contiene muchas líneas *punto a punto*, los números de Sub-Red se estarían desperdiciando debido a que sólo existen dos sistemas en cada Red *punto a punto*. El administrador debe optar por hacer uso de máscaras de 14 bits (255.255.255.255) para sus líneas *punto a punto*.

La máscara de Sub-Red para una red usualmente es sólo conocida por los ruteadores que se encuentran conectados directamente a la Red. Cuando se ejecutan protocolos de ruteo tradicionales, es imposible "ver desde afuera" de que manera se encuentra subdividida la Red.

■ Direcciones Especiales

■ *Identificación de Redes*

Es muy recomendable conocer la forma en que se debe utilizar la notación *punto* para la dirección de IP, a fin de hacer referencia a la Red. Por convención, esto se hace llenando con ceros la parte correspondiente a la dirección local de la dirección IP. Por ejemplo, 5.0.0.0 identifica una Red Clase A, 131,18,0,0 identifica a una Red Clase B y 201.49.16.0 identifica a una Red Clase C. La misma convención se sigue para la identificación de Sub-Redes con la desventaja de que nunca deben asignarse direcciones de este tipo a Host o a Ruteadores debido a que, por la notación empleada, es muy factible caer en una confusión.

■ *Mensajes a Redes*

La dirección de IP 255.255.255.255 tiene un propósito especial. Se emplea para enviar mensajes a todos los Host de la Red Local, aunque también es posible enviar un mensaje a cualquier Host de una Red Remota que se elija.

Esto se consigue llenando con unos (1) parte de Dirección Local de la Dirección de IP. Un mensaje se utiliza frecuentemente cuando un Host requiere la localización de un Servidor. Por ejemplo: suponiendo que un usuario desea enviar un mensaje a todos los nodos de una Red Ethernet Clase C con dirección 201.49.16.0, La dirección que deberá utilizar será:

201.49.16.255

El resultado de enviar un *datagrama de IP* en esta dirección será que dicho datagrama será turnado al ruteador que esté conectado a la red 201.49.16.0, entonces éste hará un *MAC layer broadcast* para entregar el mensaje a todos los Host de la Red. Es importante hacer notar que ningún Host debe tener asignada la dirección 201.49.16.255.

🖫 *Mensajes a Sub-Redes*

Un mensaje también puede ser enviado a una Sub-Red específica. Por ejemplo: Si la dirección 131.18.7.0 identifica a una Sub-Red de una Red Clase B, entonces la dirección que deberá emplearse para enviar un mensaje a todos los nodos de esta Sub-Red será 131.18.7.255.

La dirección 131.18.255.255 se puede seguir utilizando para enviar mensajes a todos los nodos de la Red Clase B completa. Los ruteadores de la configuración deberán ser lo suficientemente inteligentes para distribuir el mensaje enviado a cada Sub-Red. Si se le ha asignado el número 255 a alguna de las Sub-Redes se presentará un problema, debido a que no estará claro si el mensaje enviado en la dirección 131.18.255.255, iba dirigido a toda la Red Clase B, ó únicamente a la Sub-Red 255. La única forma de evitar este tipo de percances es asignar a las Sub-Redes números diferentes de 255.

🖫 *Direcciones de Regreso*

Así como existen mensajes que se envían a Redes o Sub-Redes específicas, también existen aquellas que nunca dejan el Host local. A efecto de hacer una prueba del software de Red, es muy útil contar con una dirección de regreso que define *"quien es el nodo emisor"*, mismo que funciona como receptor.

Para este efecto, se utiliza por convención cualquier dirección que comience con 127, por ejemplo:

127.0.0.1

Existen otros formatos de direcciones especiales que se emplean solo durante la inicialización del sistema. Estos formatos están reservados y no se pueden utilizar para identificar destinos. Por convención, la dirección 0.0.0.0 definirá a un Host específico de una Red específica, los demás Host de la misma red se definirán cambiando la parte que corresponde al Host en la dirección; por ejemplo: 0.0.0.5 identifica al Host 5 de una Red en específico.

💻 Domain Name System

A efecto de establecer una comunicación con un Host, es necesario conocer en que dirección se encuentra. Por lo regular, el usuario final conoce el nombre del Host con el que desea comunicarse, pero no así su dirección. En este caso ya sea el usuario final o la aplicación que éste haya invocado, tienen la necesidad de visualizar estas direcciones.

En Redes pequeñas y aisladas, se puede hacer frente a este problema teniendo una tabla central de mantenimiento en la que se establezca la relación nombre-dirección de Host, de esta forma, los Hosts individuales se mantendrán "al día" copiando esta tabla periódicamente.

El *Domain Name System (Sistema de Nombre del Dominio)* se implementó con el fin de brindar un mejor método para relacionar los nombres y direcciones en una Inter-Red. Los nombres y direcciones se guardan en *name servers* distribuidos a través de toda la Inter-Red.

Estos *name servers* se actualizan en forma local, asi, la conexión, desconexión y/o el movimiento de un nodo se registra rápidamente y con precisión en un *primary authoritative server*. Debido a que la conversión nombre-dirección no es tan importante, la información es copiada a uno o más *secondary authoritative servers*.

Muchos proveedores ofrecen software que permiten una función de sistema como *name server*. Regularmente el software es una adaptación del **D**ominio de **I**nter-Red **B**erkeley *(BIND)*. una corporación puede hacer uso de este software para ejecutar su servicio propio de *name servers* y opcionalmente, puede conectar su servicio de nombres al *Internet Domain Name System (Sistema de Nombres de Dominio de Inter-Red)*.

Un producto capaz de llevar a cabo visualizaciones de DNS es una parte estándar de productos de TCP/IP y recibe el nombre de *resolver*.

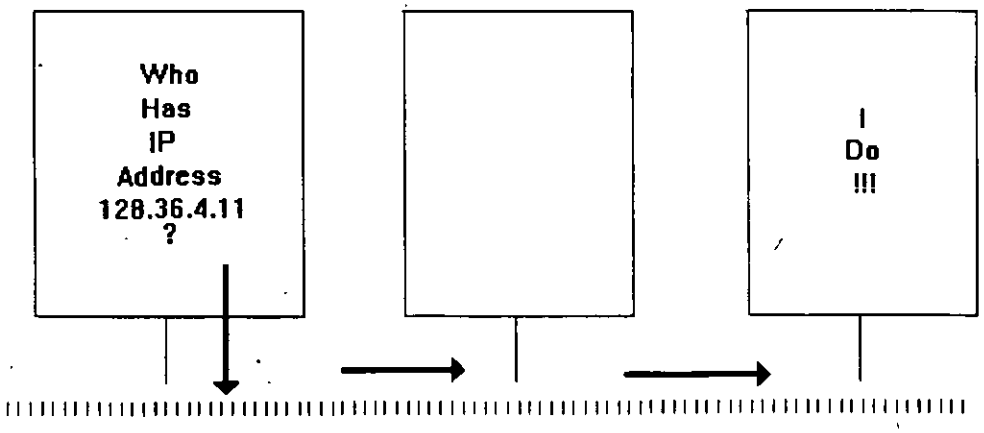
Address Resolution Protocol (ARP)

En una comunicación es necesario convertir los nombres de los nodos en sus direcciones de IP, antes de que la información pueda ser enviada de una estación a otra en una Red LAN, se debe llevar a cabo una segunda conversión ya que debe conocerse la dirección física del nodo destino. Para lograr esto se conocen tres métodos:

- Configurar una tabla de valores directamente en cada nodo
- Configurar una tabla de valores en un servidor al cual puedan consultar los nodos.
- Conocer otros valores mediante el envío de una consulta en la Red LAN.

El ARP define un método basado en mensajes para una conversión dinámica entre direcciones de IP y direcciones físicas. ARP permite al administrador de la Red añadir nodos a una Red local o cambiar una interface de red de un nodo en especial, sin necesidad de actualizar manualmente las tablas de conversión de direcciones.

Los sistemas en la Red Local pueden hacer uso de ARP para encontrar información de las direcciones físicas para sí mismos. Cuando un Host desea establecer una comunicación con otro local, visualiza la dirección de IP de éste en su tabla de ARP. Si no encuentra esa dirección, el Host envía una petición ARP que contenga la dirección de IP destino. **(fig. 3.2).**



El Host destino reconoce su dirección de IP y lee la petición. Primeramente actualizará su propia tabla de conversión de direcciones con la dirección de IP y la dirección física del Host emisor. Entonces el Host receptor envía la dirección de su propia interface de red. Cuando el Host emisor recibe esta dirección, actualiza su tabla de ARP y queda listo para una nueva transmisión a través de la Red.

NIVEL 3. PROTOCOLO DE RED

El nivel 3 provee un fuerte poder de transmisión y otros servicios.

- En nivel de paquetes punto a punto.
- Amplio direccionamiento.
- Identificación a varios niveles.
- Fragmentación.
- Datagramas de mayor envergadura.
- Uso de redes con ancho de banda limitado.
- Permite operación Inter-Red.

Notas:

TCP/IP

ARQUITECTURA

Protocolo a nivel de RED

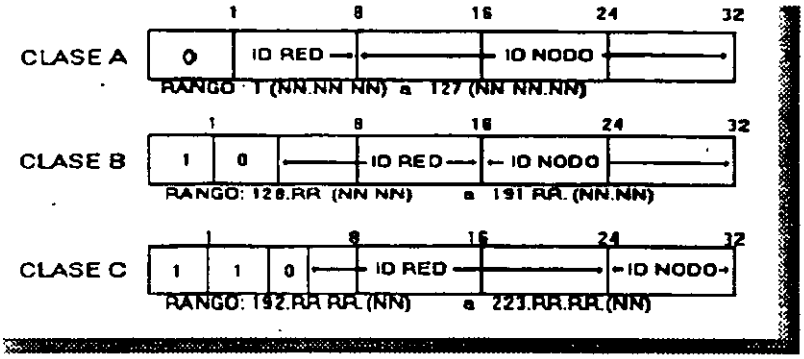
IP	Internet Protocol
ICMP	Internet Control Message Protocol .
ARP	Address Resolution Protocol .
RARP	Reverse Address Resolution Protocol .
RIP	Routing Information Protocol .
EGP	External Gateway Protocol .
OSPF	Open Shortest First .

Notas:

TCP/IP

IP: INTERNET PROTOCOL

Formato de las direcciones IP



Notas:

IP: INTERNET PROTOCOL

Brinda dos servicios básicos.

- Enrrutamiento
- Fragmentación/Re-ensambiaje

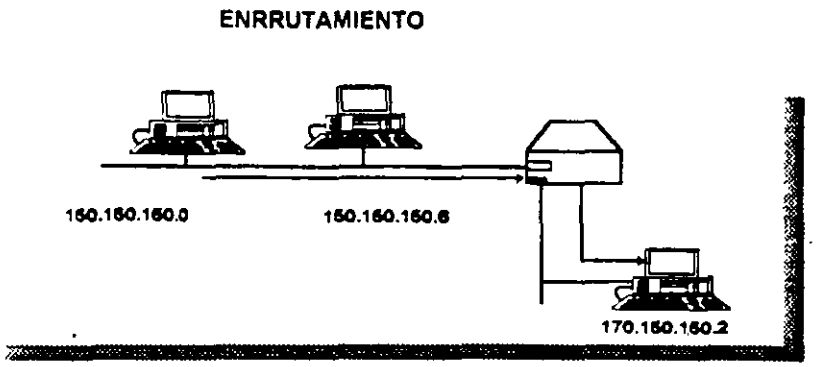
Utiliza direcciones IP para decidir el ruteo

Aísla los protocolos superiores de las características específicas de la Red.

Notas:

TCP/IP

IP: INTERNET PROTOCOL

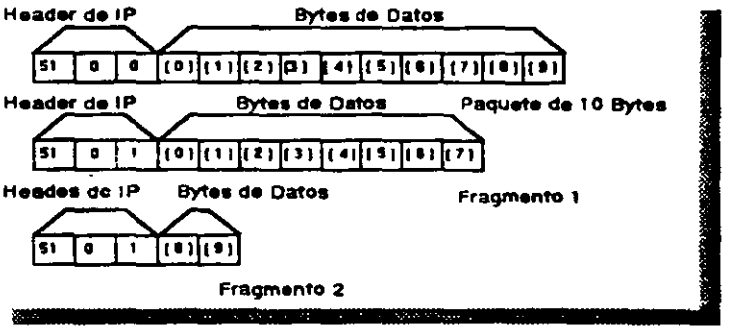


Notas:

TCP/IP

IP: INTERNET PROTOCOL

FRAGMENTACION



Notas:

TCP/IP

RUTEO DE IP

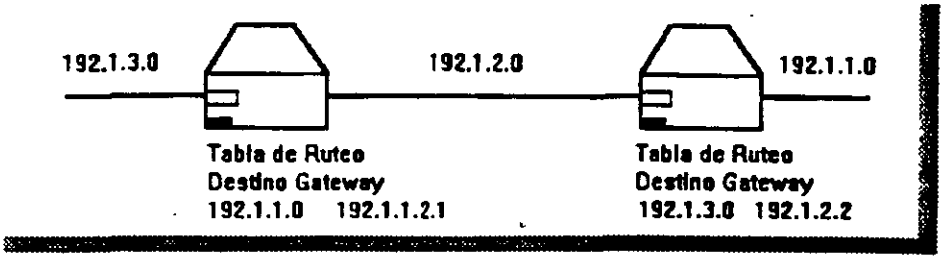
Protocolo de Ruteo

- Intercambiar entre dispositivos (generalmente routers) información acerca de las Redes que conectan.
  - Protocolos Intradominio
  - Protocolos Interdominio

Notas:

TCP/IP

RUTEO DE IP



Notas:

IP: INTERNET PROTOCOL

Nodo que envía

- Toma datos de TCP.
- Pone los datos en un paquete (datagrama).
- Decide si es necesaria la fragmentación.
- Determina ruta de acceso.
- Envía el datagrama.
- Local: Consigue la dirección física y envía.
- Remoto envía a un ruteador.

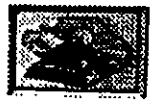
Notas:

IP: INTERNET PROTOCOL

Nodo que recibe

- Toma el paquete del nivel de enlace.
- Determina si el paquete ha sido fragmentado.
- Si está fragmentado la re-ensambla.
- Pasa el datagrama a TCP.

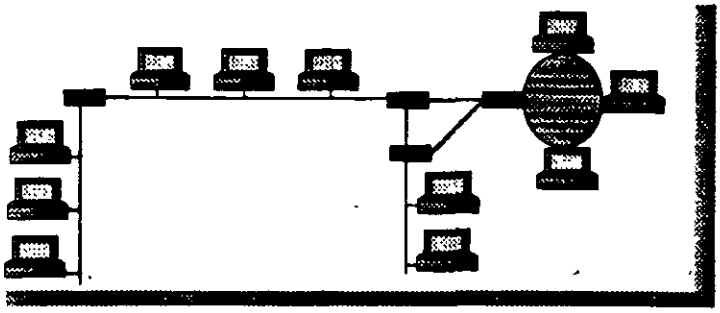
Notas:



IP: INTERNET PROTOCOL

Direccionamiento Inter-Red

- **Una Inter- Red (internet) está formada por una colección de Redes individuales, unidas por ruteadores, a veces llamados Gateways.**



Notas:

TCP/IP

I C M P

Internet **C** ontrol **M** essage **P** rotocol

- **Brinda información de ruteo y detección de errores**
- **Utilizado para informar al módulo de IP acerca de:**
  - **Paquetes que no alcanzan su destino.**
  - **Ruteadores incapaces de enviar los paquetes.**
  - **Ruteadores que pueden enviar los paquetes por rutas más cortas.**

Notas:

RARP

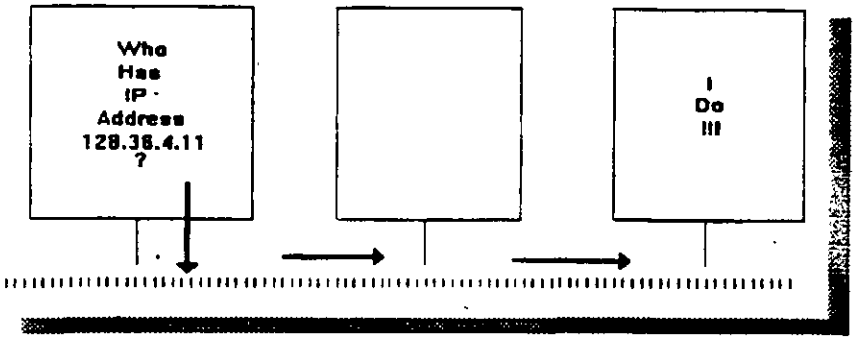
Reverse **A**ddress **R**esolution **Protocol**

- Encuentra la dirección IP para una determinada dirección Ethernet (MAC).
  - La operación se realiza a través de un broadcast
  - Se requiere de un proceso de servidor RARP por cada Red.

Notas:

TCP/IP

A R P



Notas:

TCP/IP

RARP

Reverse **A**ddress **R**esolution **P**rotocol

- Encuentra la dirección IP para una determinada dirección Ethernet (MAC).
  - La operación se realiza a través de un broadcast
  - Se requiere de un proceso de servidor RARP por cada Red.

Notas:

PROTOCOLO R I P

Routing I nformation Protocol

- Es un protocolo de ruteo simple.
- Cada ruteador transmite el costo y la dirección destino a sus vecinos.
- Existen dos tipos de paquetes:
  - Solicitud (Request).
  - Respuesta (Response).

Notas:

PROTOCOLO R I P

FORMATO DE PAQUETES

Comando

Versión

Reservado

Identificación de dirección de familia

Dirección

Métrica

Notas:

PROTOCOLO E G P

External Gateway Protocol

- Fue uno de los primeros protocolos.
- Tres funciones o aspectos.
  - Adquisición de Vecinos
  - Confirmación de Vecinos
  - Información de Ruteo.
- No funciona con enlaces redundantes.
- No se incluye métrica en la información.
- Permite que en una Lan un sólo router transmita la información a el/los routers de otros dominios.

Notas:

PROTOCOLO O S P F

O pen S hortest P ath F irst

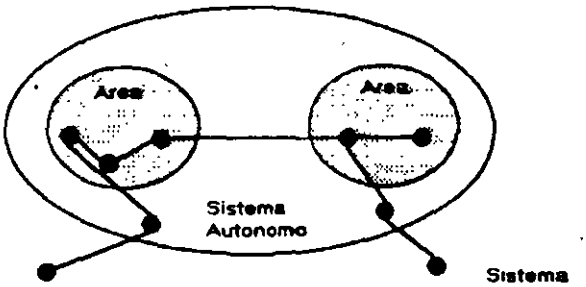
- Protocolo no propietario
- Divide la Redes en:
  - Área
  - Sistema Autónomo
  - Sistema Global
- Sistema de seguridad para la propagación de las rutas disponibles.
- Uso de diferentes tipos de métricas.

Notas:

TCP/IP

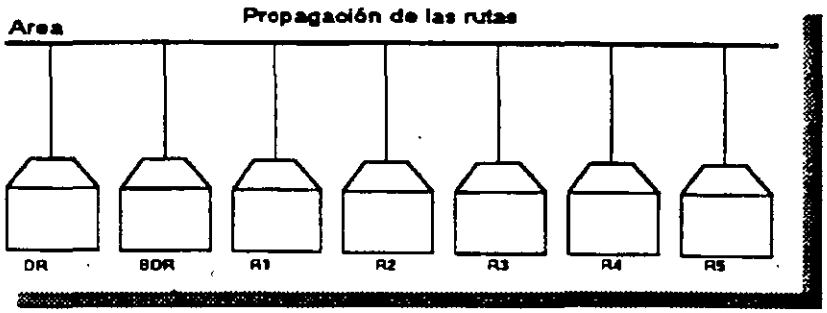
PROTOCOLO O S P F

Divisiones o Particiones



Notas:

PROTOCOLO O S P F



Notas:

PROTOCOLO O S P F

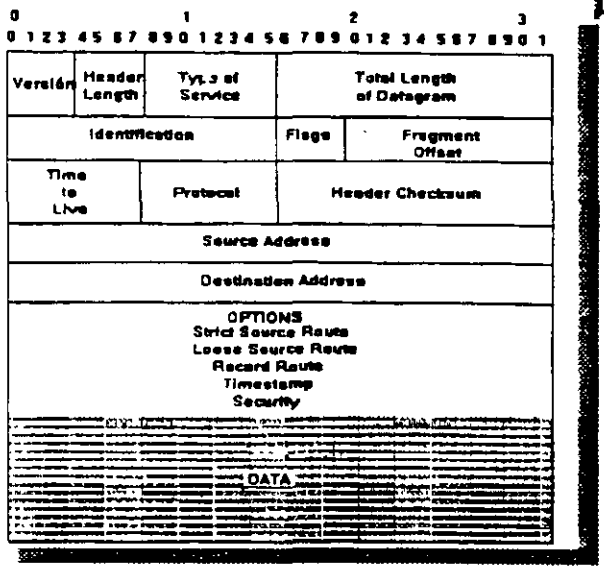
Tipos de paquetes propagación de rutas

- Tipo 1: Routing Link Advertisement.
- Tipo 2: Network Links Advertisement.
- Tipo 3: Network Summary Link Advertisement.
- Tipo 4: AS Boundary Routerssumary Link.
- Tipo 5: AS External Link Advertisement.

Notas:

TCP/IP

IP .- Cabecera

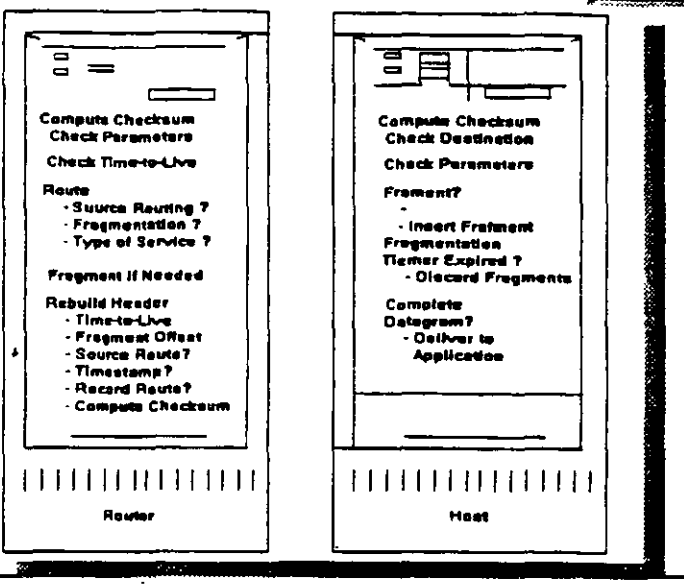


Notas:

TCP/IP

FUNCIONES

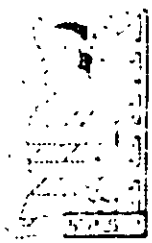
Host- Ruteador



Notas:

TCP / IP
ARQUITECTURA, PROTOCOLOS E IMPLEMENTACION

4.- INTERNET PROTOCOL



Mayo de 1996.

4.- INTERNET PROTOCOL

Funciones

Función Primaria

La función primordial de IP es aceptar datos de TCP o de UDP en un Host, crear un datagrama, y rutearlo a través de la Red. En el Host destino, IP entrega los datos a un receptor TCP o UDP. El Software IP rutea la información a su destino empleando los siguientes mecanismos:

- Máscara de Sub-Red
- Tabla de Ruteo

Máscara de Sub-Red

La máscara de Sub-Red se utiliza de tal forma que si, por ejemplo, un Host con dirección 128.36.12.27 desea enviar información a otro con dirección 128.36.12.14, se detecta que tanto el Host emisor como el receptor pertenecen a la misma Sub-Red 128.36.12, ambos pueden establecer comunicación directamente a través de la Sub-Red.

Antes de que un Host envíe información, la dirección de IP del Host receptor debe ser convertida a dirección física. Si el Host emisor no encuentra en su tabla de conversión de ARP la dirección de IP del Host destino, entonces el Host emisor utilizará el protocolo ARP para enviar una petición de información.

Tabla de Ruteo

En esta tabla se encuentran las direcciones de IP que identifican al Host como emisor y receptor, así como a los demás Host receptores. El primer destino de la tabla de ruteo es una *dirección de regreso* que autoidentifica al Host, los demás destinos se listan apareciendo primero los que pertenecen a la Sub-Red en la que se encuentra el Host. El último destino está etiquetado como *default* y es el de mayor importancia, ya que cualquier información que no haya sido específicamente ruteada, se envía en esta dirección.

En otra columna existe un indicador que permite saber cuando está activo un ruteador y si el destino es un Host o un Ruteador.

La Siguiente Columna muestra cual de los ruteadores esta listo para trabajar, y si el destino es un Host u otro ruteador.

La columna consecutiva, está etiquetada como REFcnt y cuenta el número de usos activos de una ruta. La siguiente columna está etiquetada como Use y lleva la cuenta de el número de paquetes que se le envían al ruteador. Se utiliza la interface lógica *1o0* para la prueba de *loopback*. Todo el tráfico externo pasa a través de una interface simple *1e0 Ethernet*.

Cuando el IP verifica la dirección de un Host destino, primero hace una búsqueda en la tabla para cerciorarse de que exista una entrada para la dirección de IP completa, si esto ocurre, esa entrada se utiliza para el ruteo de tráfico de información. Si no existiera la entrada, entonces el IP busca una entrada que corresponda a la Red destino. Si ni siquiera esa entrada se llegara a encontrar, entonces se utilizará la entrada por *default*.

Protocolos de Ruteo

Ordinariamente, el ruteo de datagramas tiene características de adaptabilidad, es decir; se lleva a cabo la mejor selección de ruteo mediante la verificación de la tabla que se encuentra tanto en el Host emisor como en todos los demás que están en la red. Cabe hacer notar que cualquier cambio en la topología de la red solo hará que los datagramas se ruteen nuevamente en forma automática.

Tablas de los Ruteadores

Inicialmente se configuran manualmente pero en algunas aplicaciones es necesario un cambio dinámico de las tablas de ruteo durante la operación de la red. Este cambio se automatiza mediante el ***Protocolo de Ruteo de Información (RIP)***.

No es necesario un protocolo específico para el ruteo de la información, la filosofía de Inter-Red ofrece la libertad de elegir cualquier protocolo de ruteo interno que se desee.

Tablas de Hosts

Un ruteador tiene la capacidad de notificar automáticamente a un Host si es que existe otro ruteador que le brinde un camino más eficiente para el tráfico de información a destinos particulares.

Un Host debe ser lo suficientemente "inteligente" para registrar cuando ha dejado de operar un ruteador local, determinar nuevas rutas o marcar todas aquellas que se relacionen con el ruteador que dejó de operar.

Fragmentación

En una Inter-Red grande, un Host inicio puede no saber todo acerca de los límites del tamaño de los datagramas que encontrará en su camino. Cuando un datagrama que se envía es demasiado grande para alguna red intermedia, al llegar a un ruteador que está conectado a esa red intermedia, IP fragmenta el datagrama original en muchos datagramas más pequeños. Sólo dependerá del Host destino la recepción de los fragmentos y reconstruir el datagrama original.

La fragmentación se lleva a cabo generalmente en un solo ruteador, no obstante, una aplicación UDP deberá enviar un mensaje grande que ocasione que el Host emisor fragmente ese datagrama.

Tipo de Servicio

Las aplicaciones pueden seleccionar diversas características del tipo de servicio como *nivel de prioridad, confiabilidad de entrega, necesidad de retardo, etc.*

Tiempo de Vida

La ruta de un datagrama puede cambiar debido a una falla en el ruteador o una congestión en el tráfico de información. Un datagrama puede ser enviado en un camino indirecto y llegar muy tarde, o incluso puede caer en un lazo repetitivo. El mecanismo denominado *Tiempo de Vida* asegura que los datagramas que no se enviaron recientemente sean removidos de la red. El tiempo de vida máximo se incluye como cabecera de un datagrama y se va decrementando a medida que pasa por diferentes ruteadores. Cuando este valor llega a cero, el datagrama se remueve de la Red.

Checksum

La cabecera de IP lleva consigo un campo de *checksum.* el checksum se calcula de los campos de las cabeceras que quedan. Podría parecer innecesario calcular otro checksum cuando un *Frame de Secuencia de Verificación* se calcula para cada datagrama.

Sin embargo, se recomienda calcular otro checksum debido a la presencia de los errores del bus y las fallas de la tarjeta de interface. El checksum de la cabecera de IP debe recalcularse cada que el datagrama llega a un ruteador debido a que la cabecera se actualiza constantemente.

Opciones diversas

IP ofrece un gran número de opciones de servicio que pueden ser invocados por una aplicación de comunicaciones.

La opción de *Strict Source Route* obliga a que una ruta completa se siga en una sola dirección, es decir; de fuente a destino. Esta opción puede ser usada como parte de un programa de seguridad, o ser invocado por un programa de mantenimiento que verifique cual de todas las rutas está disponible.

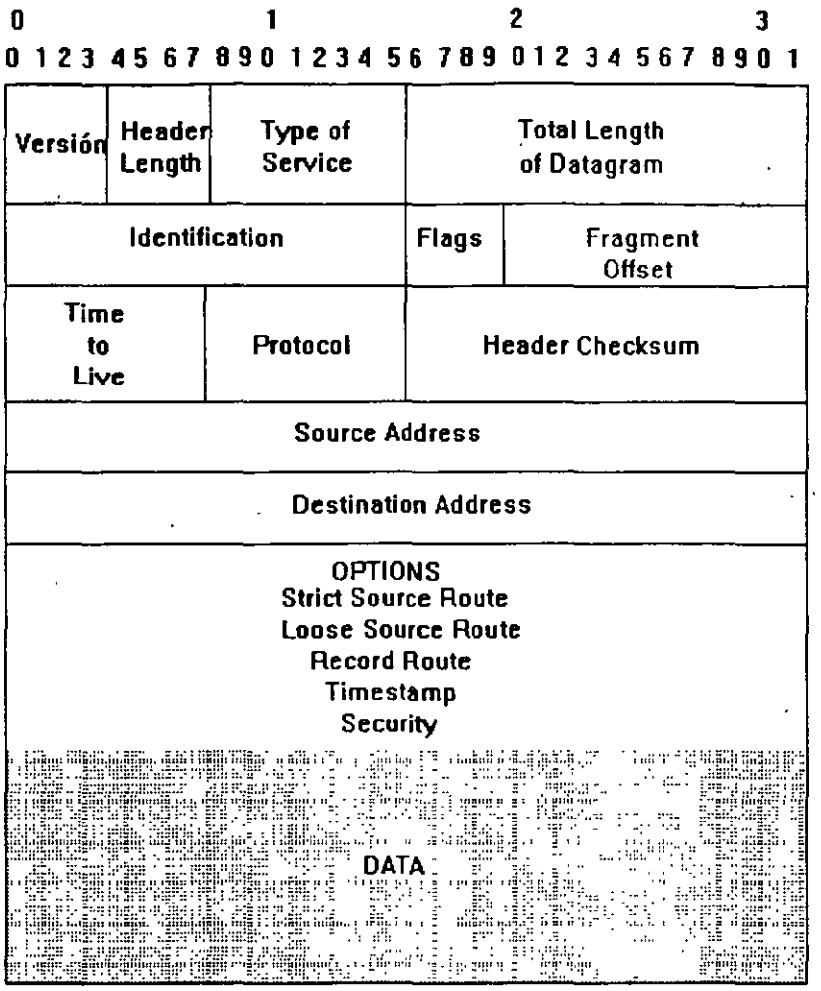
La opción de *Loose Source Route* determina las direcciones *landmark* a lo largo de una ruta. El datagrama visitará los *landmarks*, pero probablemente haya pasos intermedios entre cada *landmark*. Esta opción puede ser muy útil cuando se intenta rutear a partes remotas de una Inter-Red.

Las opciones de *Record Route* y *Timestamp* son muy útiles en el manejo de redes. La dirección de cada ruteador visitado se suma un campo de *registro de rutas*. El campo de *Timestamp* registra una secuencia de *sellos de tiempo* de 32 bits y opcionalmente la ruta.

Mecanismos

Cabecera de Datagrama

Se compone de 5 o más palabras de 32 bits. el tamaño máximo de la cabecera es de 15 palabras o 60 bytes, pero en la práctica, la mayoría de las cabeceras de datagrama tienen una longitud mínima de 5 palabras, o 20 bytes. **(fig 4.1).**



La primera palabra en la cabecera del datagrama contiene:

- Versión
- Longitud de Cabecera
- Tipo de Servicio
- Longitud Total.

La última versión de IP es la 4. Esta versión no interactúa con versiones anteriores, si se recibe en esta versión un datagrama de una versión anterior, éste será descartado por el software actual.

La longitud de la cabecera se mide en palabras de 32 bits. Si no lleva opciones, la longitud de la cabecera es de 5 palabras. si se incluyen una o más opciones, será necesario llenar con ceros (0) la cabecera a efecto de completar el límite de los 32 bits.

Tipo de Servicio

Este campo consta de 8 bits:

BITS	DESCRIPCION		
0-2	Prioridad o precedencia. De 0 a 7 donde 7 es la mayor		
3	Índice de Retardo	0 = Normal	1 = Bajo
4	Índice de Troughput	0 = Normal	1 = Alto
5	Índice de Confiabilidad	0 = Normal	1 = Alto
6-7	Reservado		

El estándar IP no gobierna las aciones específicas que ocasionan los valores del campo Tipo de Servicio. Originalmente se intentó que IP mapeará el status de Tipo de Servicio en opciones similares para la Sub-Red subsecuente para el siguiente ruteo. Por ejemplo, el acceso a una red Token-Ring se puede programar en base a un nivel de prioridad. IP puede mapear el nivel de prioridad hacia otro nivel de prioridad Token-Ring correspondiente.

Algunos Host y ruteadores ignoran por completo el campo Tipo de Servicio, mientras otros hacen uso de éste tomando decisiones de ruteo o decidiendo que tráfico debe estar protegido en caso de estar en posibilidades de ser descartado por escasez de memoria.

Campo de Longitud Total

Este campo contiene la longitud del datagrama medida en octetos, incluyendo tanto a la cabecera como a los fragmentos del datagrama. Este campo de 16 bits puede expresar valores de hasta 64KBytes. El estándar IP requiere que todos los Host tengan la capacidad de aceptar datagramas de hasta 576 bytes.

Campos de Fragmentación

Los campos de Identificación, Semaforeo y Fragment Offset, son primordiales en la Fragmentación y el Reensamble.

- El campo de *Identificación* contiene un número de 16 bits que ayuda al host receptor a reconocer los fragmentos de datagramas que deben ir unidos.

- El campo de *Semaforeo* tiene la siguiente estructura:

BITS	0	1	2
	0	0 = Puede Fragmentarse	0 = Ultimo Fragmento
	0	1 = No Fragmentar	1 = Más Fragmentos

Cuando un ruteador fragmenta un datagrama, cada porción debe alinearse en un límite de 8 bytes. A esta unidad se le conoce como *Fragment Blocks.*

- El campo de Offset contiene la distancia medida en *Fragment Blocks* del dato del fragmento actual al comienzo del datagrama original. Este campo tiene una longitud de 13 bits de tal forma que los offsets pueden ir de 0 hasta 8192 *Fragment Blocks* correspondientes a 64KBytes.

Reconstrucción de un Datagrama Fragmentado

Esta tarea se lleva a cabo en el Host receptor. Las partes del datagrama fragmentado pueden llegar en desorden, cuando la primera parte llega al Host receptor, IP localiza los recursos de reensamble. El campo *Offset* le indicará el byte extremo en el cual deben ser puestos los datos de ese fragmento.

Fragmentos que correspondan a los campos *Identificación, Fuente, Destino y Protocolo* deben ir juntos y se van uniendo conforme van llegando. Una omisión inconveniente en el Protocolo IP es que el Host receptor no tiene medios para conocer la longitud total del datagrama hasta llegar el último fragmento. El campo *Longitud Total* en un fragmento indica únicamente la longitud de ese fragmento de datagrama.

Esto implica que el sistema receptor tiene que "adivinar" cuanto espacio de buffer deberá reservar para la llegada de un datagrama.

Tiempo de Vida

Este campo contiene el tiempo máximo, expresado en segundos, que puede permanecer un datagrama en una Inter-Red antes de que llegue a su destino.

Este campo lo fija el Host emisor y se va decrementando a medida que pasa por un ruteador que manipule el datagrama.

Protocolo

Este campo contiene un número de 8 bits que identifica al protocolo de capa superior que se encarga de recibir la porción de datagrama. El identificador para TCP es 6 y para UDP es 17.

Checksum de Cabecera

Este campo de 16 bits contiene un *checksum* que se calcula en los campos de la cabecera de IP. El *checksum* debe ser actualizado a medida que el datagrama avanza ya que el campo *Tiempo de Vida* se decrementa en cada ruteador.

Direcciones Fuente y Destino

Es un valor de 32 bits que indica la dirección de IP.

Opciones adicionales

Estas se incluyen en el datagrama y son elegidas por las aplicaciones de comunicación, las opciones que actualmente se utilizan son:

- Strict Source Route
- Loose Source Route
- Record Route
- Timestamp
- Department of Defense Basic Security
- Department of Defense Extended Security
- No Operation
- End of Option List

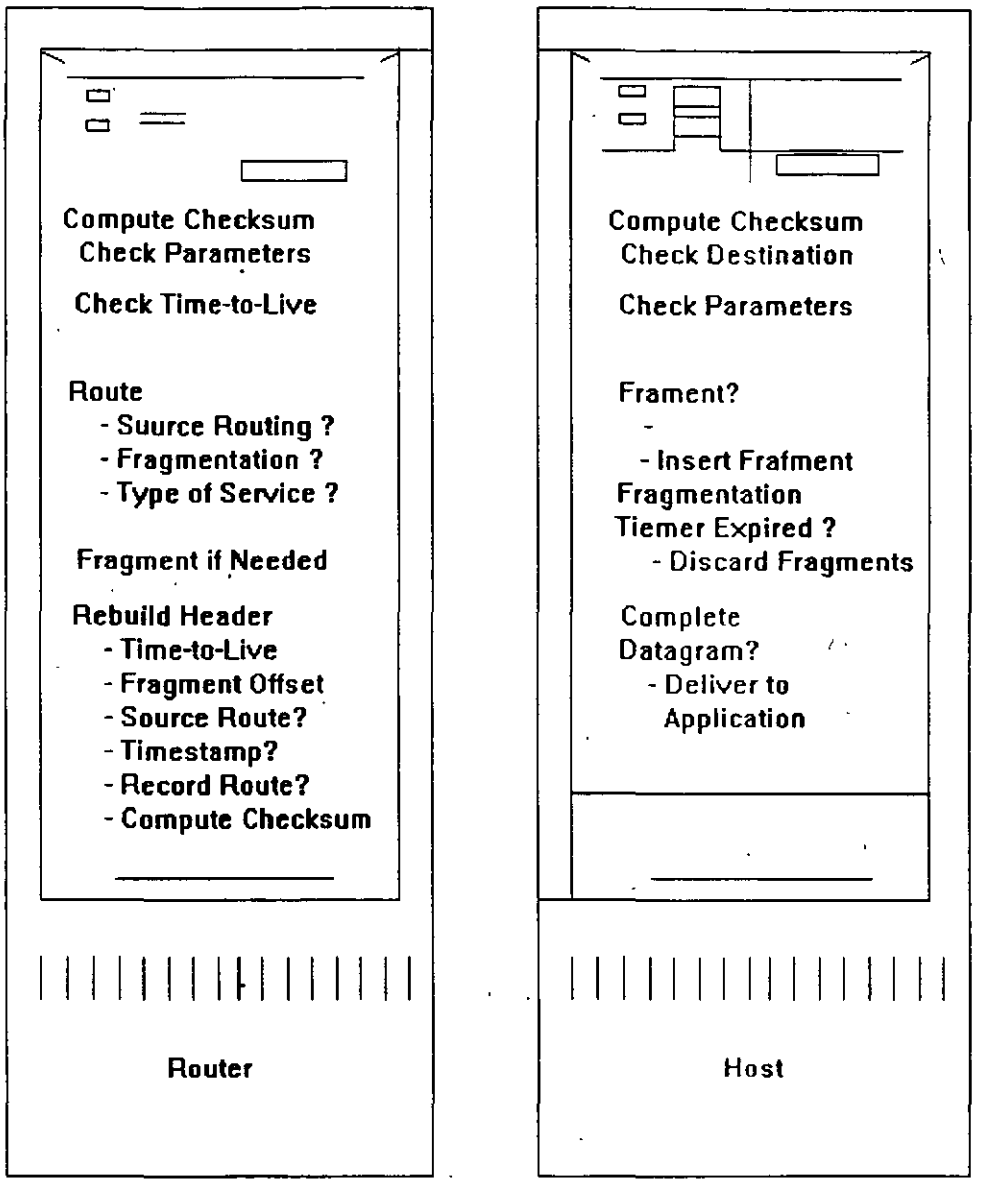
Las opciones de Ruteo y de *Timestamp* ya han sido explicadas con anterioridad.

Las dos opciones de *Department of Security* solo están disponibles para organismos como la CIA, la Agencia Nacional de Seguridad de E.U. y el Departamento de Energía de E.U.

La opción *No Operation* se usa como "relleno" entre opción y opción; por último, la opción *End of Option List* se utiliza para indicar en donde finaliza la lista de opciones dentro del límite de los 32 bits.

Proceso de Datagramas

A efecto de entender mejor la filosofía de IP en la siguiente figura **(fig. 4.2)** se resumen los mecanismos y funciones principales de IP.



Procesos de los Ruteadores

Cuando un ruteador recibe un datagrama, este pasa primero por una serie de verificaciones para saber si el datagrama debe ser o no descartado. la cabecera de *checksukm* se recalcula y se compara con el campo *checksum*. Los campos:

- Versión
- Longitud de Cabecera
- Longitud Total
- Protocolo

se despliegan en pantalla a efecto de verificar que concuerden. El campo *Tiempo de Vida* se decrementa.

Un error de *checksum* de parámetros o un valor de cero en el campo *Tiempo de Vida* causará que el datagrama se descarte. Después, el siguiente ruteador ejecuta su procedimiento de ruteo consultando primeramente el campo de *Strict Routing* si existe. Posteriormente se verifican los campos de *Tipo de Servicio* y se checa también la permisibilidad de fragmentación.

Si es necesario se fragmenta el datagrama para luego reconstruir una nueva cabecera para cada datagrama o fragmento de datagrama. Se incluye ahora un nuevo valor para el campo *Tiempo de Vida* en la cabecera y de la misma manera se actualizan los demás campos.

Finalmente se recalcula un nuevo *checksum* de cabecera y el datagrama se envía a su siguiente destino.

Procesos de los Hosts Destino

En el Host destino se calcula un *checksum* y se compara con el campo *checksum* de cabecera. Se despliegan en pantalla los campos:

- Versión
- Longitud de Cabecera
- Longitud Total
- Protocolo

A efecto de verificar que concuerden. El datagrama se descartará en caso de que los campos anteriores no concuerden o si el Host no tiene suficiente espacio en buffer disponible para procesar el datagrama.

Si el datagrama viene fragmentado el Host verifica los campos:

- Identificación
- Dirección Fuente
- Dirección Destino
- Protocolo.

Posteriormente el Host une los fragmentos con valores idénticos y después utiliza el Offset del Fragmento para darle al fragmento de datagrama la posición correcta.

Un Host no puede esperar indefinidamente para completar el reensamble de un datagrama. Cuando el fragmento llega el fragmento inicial se configura localmente un *timer* entre uno y dos minutos, así cualquier fragmento se descartará cuando el tiempo se acabe.

Relación al Modelo OSI

IP corresponde al ***Connectionless Network Layer Protocol (CNLP)*** de OSI. De la misma manera que IP, CNLP de OSI se basa en datagramas y su arquitectura es muy similar a la de IP.

OSI utiliza diferente terminología para diversos conceptos, por ejemplo:

IP	OSI
Datagrama	Network Layer Protocol Data Unit
Fragmento	Segment
Tipo de Servicio	Quality of Service

La Unidad de Datos del Protocolo de Capas de Red de OSI lleva el mismo tipo de información de la cabecera de un datagrama de IP, sin embargo, existen algunas diferencias **(fig 4.3)**:

Octets	
1	Network Layer Protocol Identifier
1	Header Lenght
1	Version
1	Lifetime (Time-to-Live)
1	Flags
2	Segment (Fragment) Length
2	Checksum
1	Destination Address Lenght
Varies	Destination Address
	Source Address Lenght
Varies	Source Address
2	Identification
2	Segment (Fragment) Offset
2	PDU Total Lenght
Varies	OPTIONS Padding Security Source Routing Recording of Route Quality (Type) of Service Priority
Varies	DATA

TCP/IP

NIVEL 4 PROTOCOLOS DE TRANSPORTE

El nivel de transporte provee a una máquina con conexiones punto a punto independiente de la subred y servicios de transacción.

- Provee enlaces confiables y eficientes entre procesos
- Forma en conjunto con los nieveles inferiores una robusta plataforma de comunicaciones.
- Realiza los enlaces virtuales.
- Tiene dos protocolos principales.
  - TCP
  - UDP

Notas:

TCP/IP

ARQUITECTURA

Protocolo a nivel de Transporte

TCP	**Transmission Control Protocol**
UDP	**User Datagrama Protocol .**
NVP	**Network Voice Protocol .**

Notas:

PROTOCOLO TCP

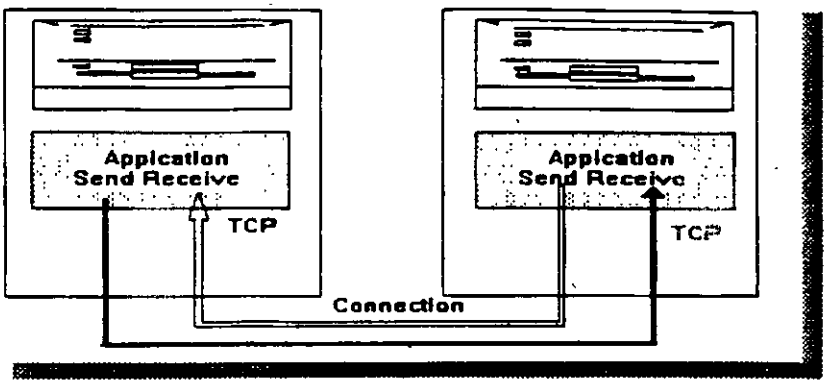
Transmission Control Protocol

- Asignación de números de puerto para transmisión de datos.
- Reconocimiento de datos recibidos.
- Regulación de flujo de datos
- División de los mensajes de datagramas.
- Verificación de los datagramas.
- Administración
  - Establecimiento
  - Mantenimiento
  - Terminación

Notas:

TCP/IP

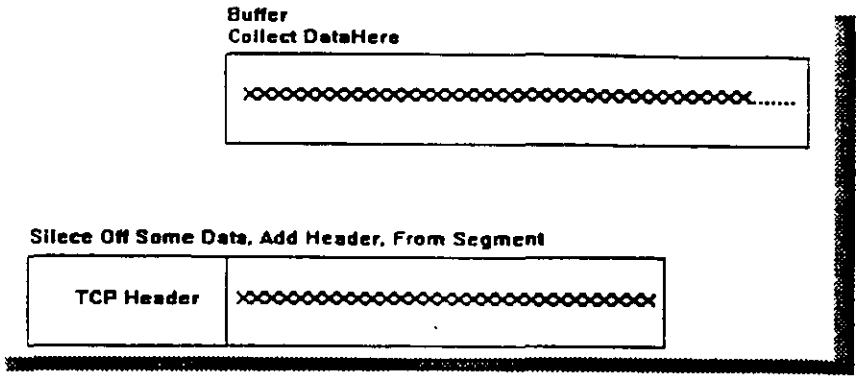
Entradas y Salidas de Tramas de Datos



Notas:

TCP/IP

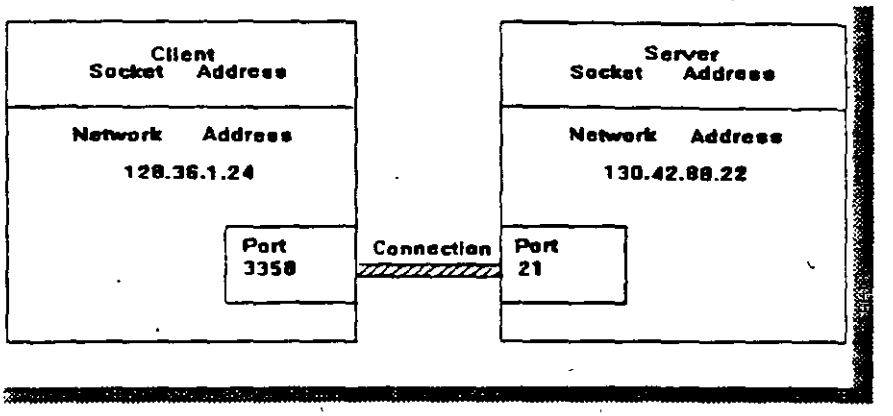
DATAGRAMAS



Notas:

TCP/IP

PUERTOS



Notas:

TCP/IP

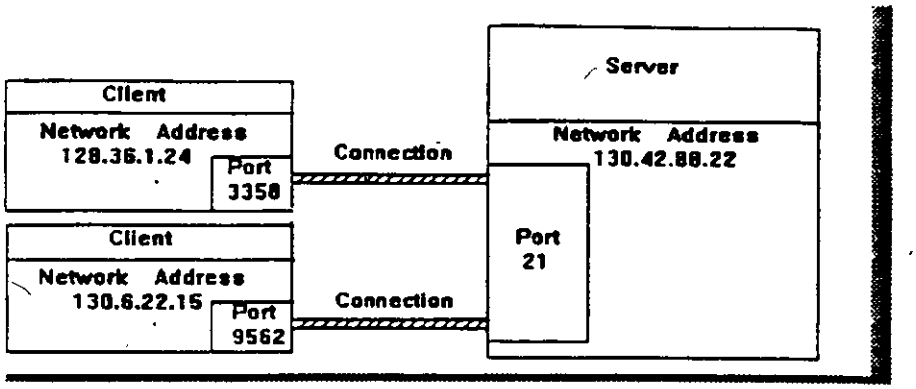
PUERTOS

Port	Application	Description
9	Discard	Discard all incoming data
19	Chargen	Exchange streams of characters
20	FTP-Data	File Transfer data transfer port
21	FTP	File Transfer dialogue port
23	TELNET	Telnet remote login port
25	SMTP	Simple Mail Transfer Protocol port
103	X400	Used for X400 mail service
110	POP3	Used for PC mail service

Notas:

TCP/IP

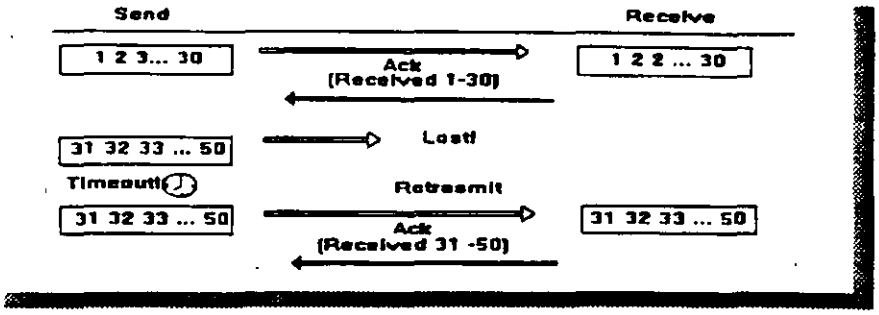
PUERTOS



Notas:

TCP/IP

NUMERACION Y RECONOCIMIENTO

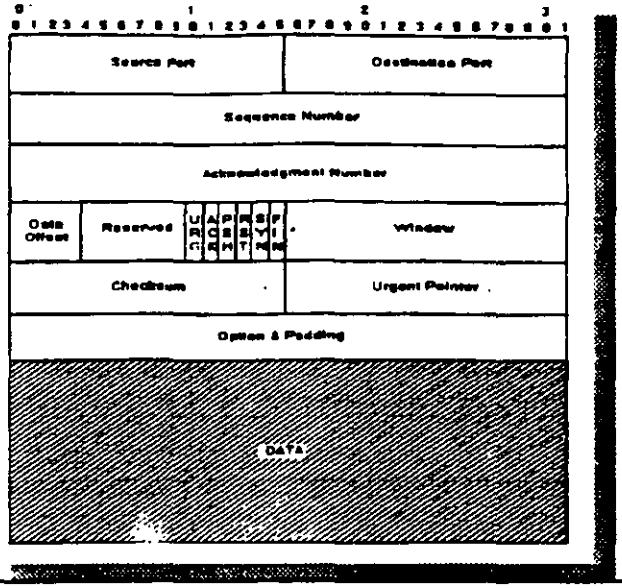


Notas:

TCP/IP

CABECERA

TCP



Notas:

PROTOCOLO U D P

U ser D atagrama P rotocol

- UDP brinda servicio de datagramas a los programas del usuario.
- No garantiza una transferencia confiable de los datos.
- Envía/Recibe datos sin capacidad de retransmisión.
- Supone que la aplicación de más alto nivel realiza la validación.
- Utilizado por:
  - NFS (N etwork F ile S ystem)
  - SNMP
  - FTP

Notas:

PROTOCOLO NVP

Network Voice P rotocol

- Servicio para transporte de voz digitalizada.
- Protocolo de transacción de tiempo real.
- Utiliza IP para transmitir información.
- Emplea algoritmos de compresión.
- Es coneccition less.

Notas:

TCP / IP
ARQUITECTURA, PROTOCOLOS E IMPLEMENTACION

5.- TRANSMISSION CONTROL PROTOCOL

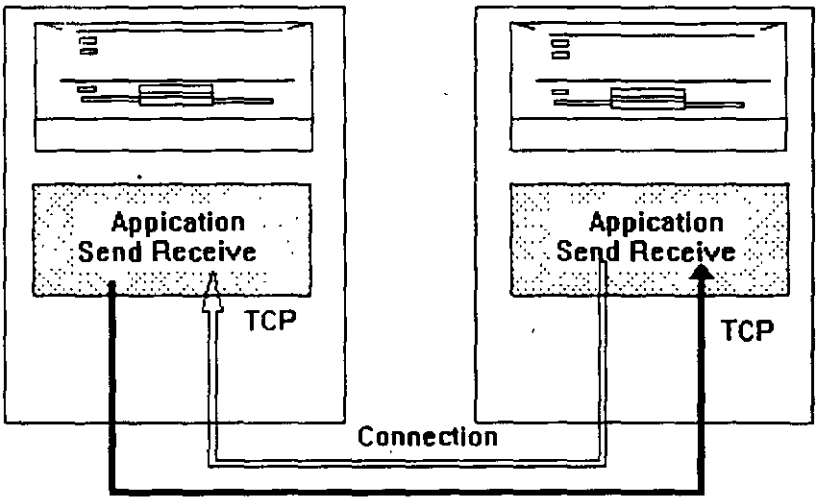
Mayo de 1996.

5.- TRANSMISSION CONTROL PROTOCOL

Conceptos de TCP

Entrada y Salida de Tramas de Datos

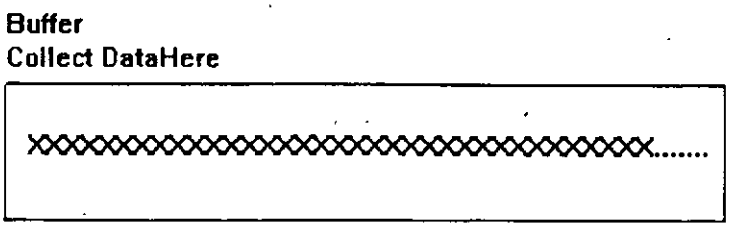
Durante una conexión TCP, las aplicaciones envían una trama de datos hacia una aplicación cliente. Al mismo tiempo, ésta recibe una trama de datos de su propio cliente. TCP ofrece un servicio completamente bidireccional que maneja simultáneamente dos tramas de datos **(fig. 5.1)**. Esto significa que TCP actúa como emisor y receptor concurrentemente.

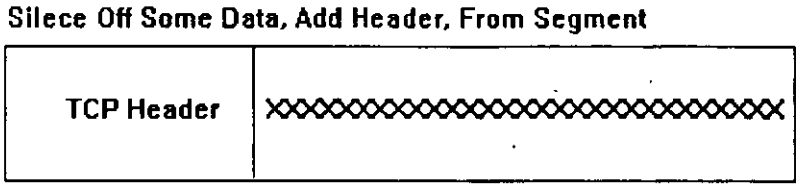


Segmentos

TCP debe convertir una trama de datos que va de salida, a una forma que pueda ser entregada en datagramas. Esto se lleva a cabo de la siguiente manera:

La aplicación turna los datos a TCP que se encarga de acumular estos datos en un buffer de envío. TCP toma una parte de los datos periódicamente y le adiciona una cabecera, formando así un *segmento*. **(fig 5.2)**.





TCP turna el segmento a IP para la entrega como un datagrama. El envío de datos en tramas de tamaño óptimo, hace más eficientes las facilidades de transmisión, por lo que TCP "espera" un tiempo específico hasta que la trama de datos sea del tamaño óptimo, antes de crear un *segmento*.

Presión y Datos Urgentes

Algunas veces no resulta conveniente para el usuario de TCP, el manejo de tramas de datos de tamaño óptimo. Por ejemplo, suponiendo que un usuario final ha iniciado una conexión remota y teclea un comando finalizando con un *Return*.

El usuario desea que estos datos sean envíados al Host remoto en una forma inmediata. Existe una función TCP de *Presión* que logra que esto suceda. El software local indica una *presión* después de que el usuario oprime la tecla *Return*. TCP transmite los datos del usuario al cliente TCP que, en su momento, entregará esos datos a su aplicación.

Otra función de TCP muy útil se encarga de marcar alguna información como *urgente*.

Tipo de Servicio y Seguridad

Como se había mencionado anteriormente, el Tipo de Servicio de los datagramas identifica su prioridad de entrega, así como su retardo y su nivel de confiabilidad. La cabecera de IP contiene el campo *Tipo de Servicio* y opcionalmente puede contener un campo de *Seguridad*. Las decisiones acerca del *Tipo de Servicio* y la *Seguridad* apropiados para una conexión, las toman las aplicaciones que la utilizan.

Relación con IP

TCP e IP se soportan entre sí en la confiabilidad de la información. IP crea las cabeceras de sus datagramas de salida en base a la información que viene de TCP. Cuando llega un datagrama de entrada, IP reporta información de la cabecera de ese datagrama como puede ser la *Dirección Fuente, el Tipo de Servicio, Longitud de los Datos* y las diferentes opciones para TCP.

Puertos

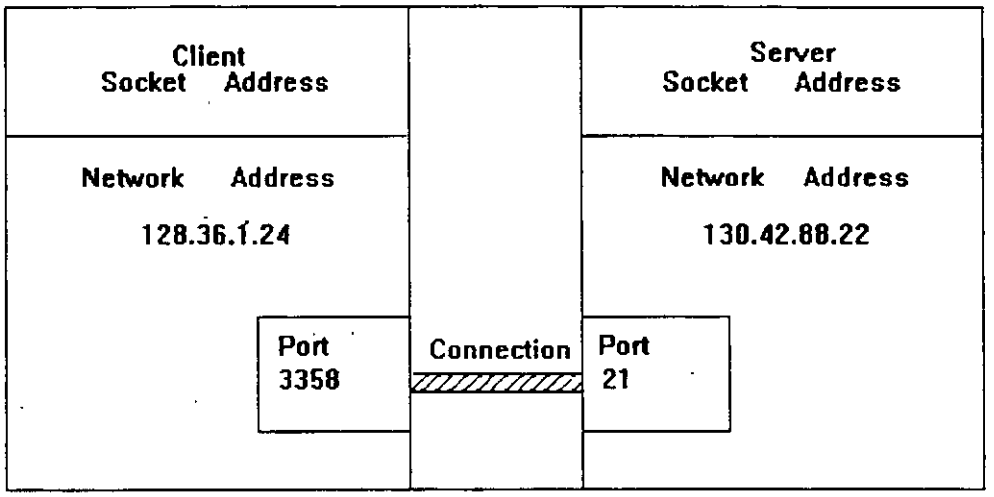
Un cliente debe identificar al servidor que quiere llegar. Esto se consigue especificando la dirección de IP del Host del servidor y su *Número de Puerto TCP*. Estos números de puerto se encuentran en el mismo rango que los números de UDP: de 0 a 2^{16} -1.

Los puertos en el rango de 0 a 1023 son los puertos más conocidos y se utilizan para accesar servicios estandarizados.(**Tabla 5.1**).

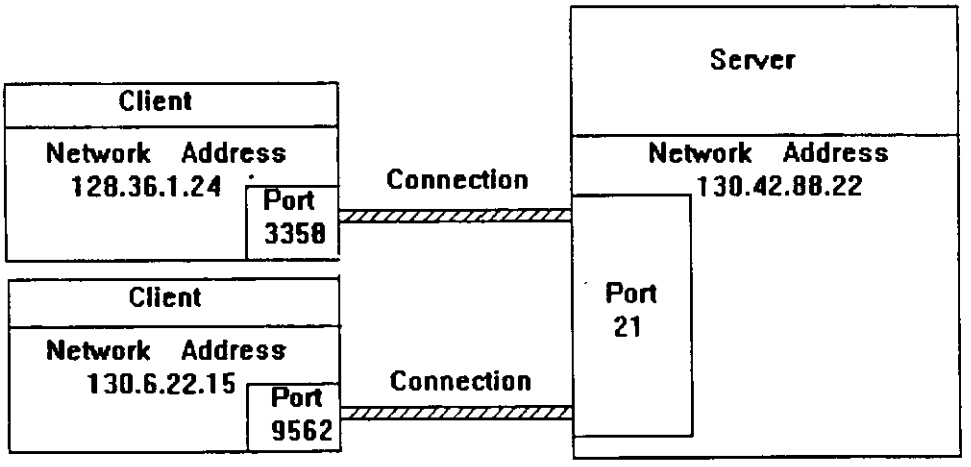
Port	Application	Description
9	Discard	Discard all incoming data
19	Chargen	Exchange streams of characters
20	FTP-Data	File Transfer data transfer port
21	FTP	File Transfer dialogue port
23	TELNET	Telnet remote login port
25	SMTP	Simple Mail Transfer Protocol port
103	X400	Used for X400 mail service
110	POP3	Used for PC mail service

Dirección de Socket

Se le denomina *Dirección de Socket* a la combinación de las direcciones de IP y el puerto utilizado para la comunicación. Una conexión TCP queda determinada en su totalidad por las *Direcciones de Socket* de sus dos extremos. **(fig. 5.3).**



Como los datos para una conexión particular de TCP siempre queda identificada por sus direcciones de IP y sus números de puertos, para el servidor es muy sencillo seguir el rastro de las conexiones de muchos clientes. **(fig. 5.4).**



Mecanismos de TCP

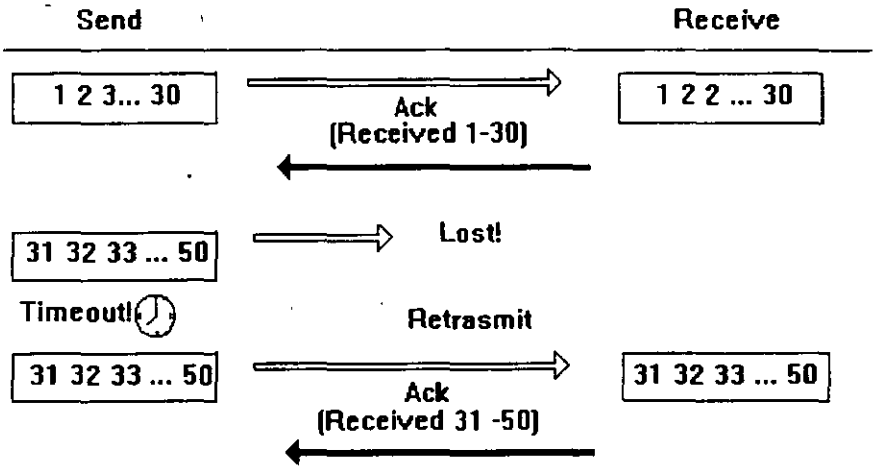
Numeración y Reconocimiento

Para la transferencia de datos de manera confiable, TCP emplea un esquema de *Numeración y Reconocimiento.*

Un TCP receptor debe mantener informado al emisor acerca de la cantidad de información correcta que le ha llegado, mediante señales de reconocimiento (*AKCs*). Si el AKC de un segmento no llega en un intervalo de tiempo determinado, el TCP emisor vuelve a enviar ese segmento.

A esta estrategia se le conoce con el nombre de *Retransmisión con Reconocimiento Positivo*. Ocasionalmente una retransmisión provocará una reproducción en los segmentos entregados al TCP receptor.

El TCP receptor debe arreglar los segmentos que va recibiendo, en forma correcta, descartando todos aquellos que estén duplicados. De esta manera, el TCP entrega los datos a su aplicación de manera íntegra. **(fig. 5.5)**.



Campos de Cabeceras TCP para Puertos, Secuencias y ACKs.

El emisor completa la secuencia numérica del primer byte de datos incluidos, así como las direcciones fuente y destino de la comunicación.

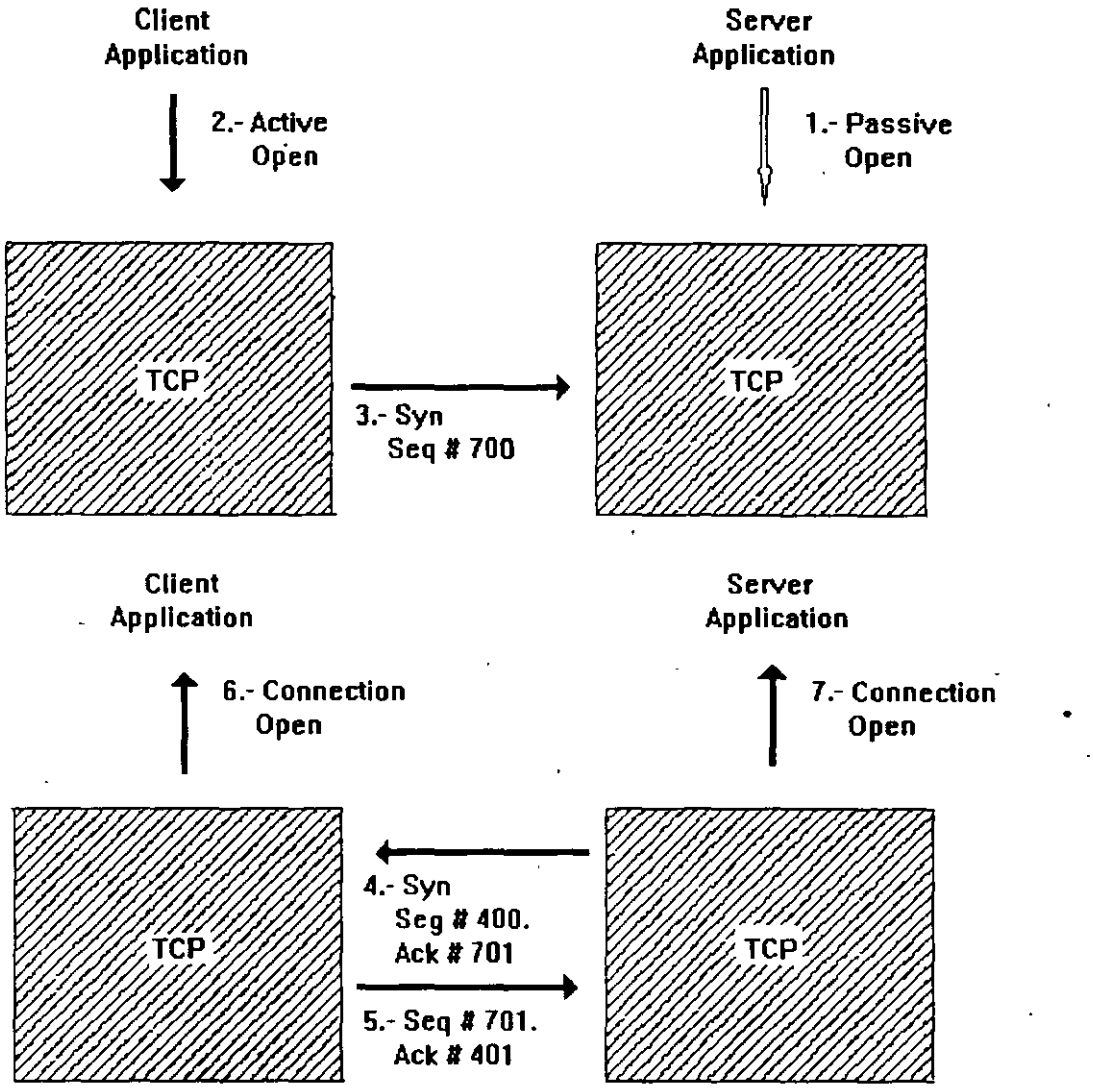
El emisor también completa el campo de reconocimiento que contiene el número del siguiente byte esperado del otro extremo. Es decir, si se han recibido hasta 30 bytes, se escribirán entonces 31 en el campo de reconocimiento del segmento que se transmitirá posteriormente.

Establecimiento de una Conexión

El patrón más utilizado para establecer una conexión es el siguiente:

Un proceso es un servidor que opera en un puerto reservado. El servidor envía una apertura pasiva que le muestra a su TCP como esperar comandos de sus clientes. Después el cliente envía una apertura activa a su TCP solicitando ser conectado al puerto y a la dirección de red del servidor. Existe un punto técnico: Más que comenzar su numeración de bytes, cada lado toma un número inicial de la secuencia de un reloj interno de 32 bits.

Esquema de Conexión



(fig. 5.6)

El procedimiento de conexión se conoce también como *"Tres maneras de Saludarse"* ya que se intercambian tres mensajes para establecer la comunicación (SYN, SYN y ACK).

El procedimiento se lleva acabo de la siguiente manera:

- El servidor envía un comando de apertura pasiva que le indica a TCP que está listo para aceptar conexiones de los clientes.

- El Cliente envía un comando de apertura activa que le indica a TCP que desea comenzar una conexión con un servidor a una dirección de IP y un puerto específicos.

- El Cliente TCP toma un número de secuencia inicial y envía un *segmento sincronizado (SYN)* que lleva ese número de secuencia.

- Cuando llega el *SYN* el servidor TCP toma su propio número de secuencia inicial y envía un *SYN* con ese número y una señal de reconocimiento *ACK* que contiene el número que debe tener el primer byte de los datos que debe enviar el cliente.

- Cuando el cliente TCP recibe los mensajes *SYN* y *ACK* del servidor, el cliente regresa un *ACK* con el número que deberá tener el primer byte de los datos que debe enviar el servidor.

- El Cliente TCP notifica a los procesos de su capa superior que se ha iniciado una conexión.

- Cuando el servidor TCP recibe el mensaje *ACK* del cliente TCP, notifica a los procesos de su capa superior que se ha iniciado una conexión.

Hasta este punto tanto el cliente como el servidor han sincronizado sus números de secuencia y se encuentran listos para el intercambio de datos.

Soporte de Tipo de Servicio

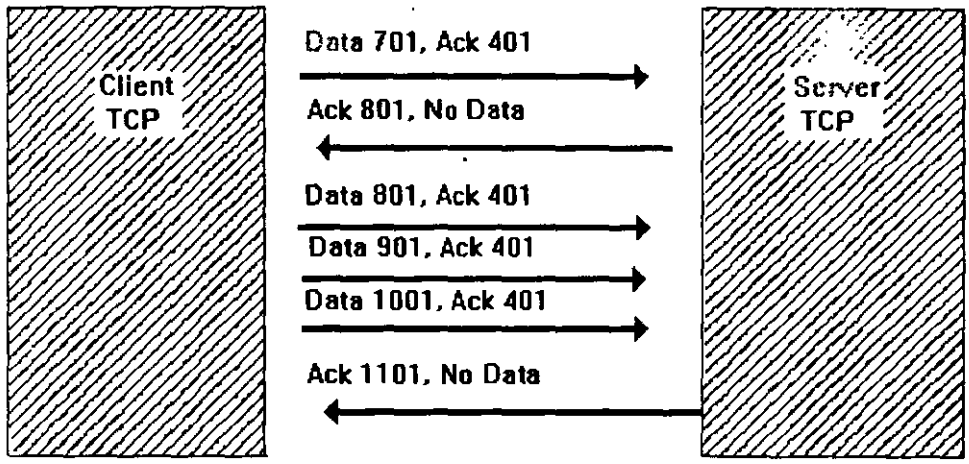
Una aplicación informa a TCP el tipo de servicio que requiere para establecer la conexión. TCP notifica al IP el tipo de servicio que se eligió, IP debe utilizar ese mismo para el envío de datagramas de salida. Cada extremo de la conexión elige su propio tipo de servicio de manera independiente.

Soporte de Seguridad

Si una aplicación está haciendo uso de la opción de seguridad de IP los dos extremos de la conexión deben concordar con el mismo nivel de seguridad. TCP le informa al IP el nivel de seguridad que se solicitó. IP se encarga de incluir la opción de seguridad en la cabecera de IP para el mensaje *SYN*.

Transferencia de Datos

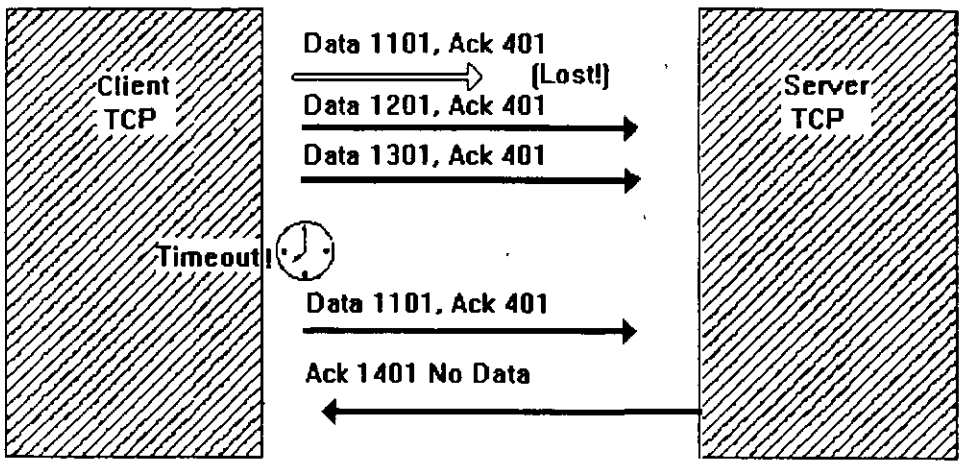
La Transferencia de Datos comienza después de haberse completado las *"tres formas de saludarse"*. La **(fig. 5.7)** muestra una transferencia de datos en un solo sentido.



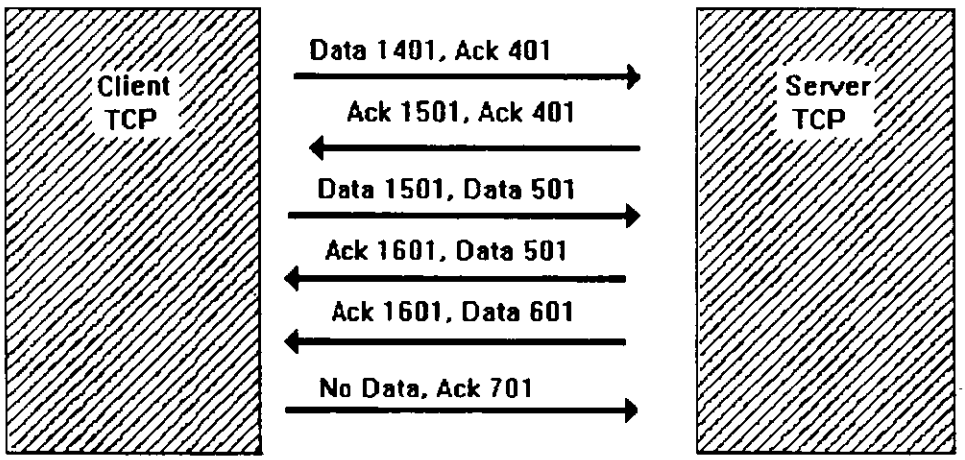
A efecto de manejar un ejemplo sencillo, se envían 100 bytes en cada mensaje. Cada una de las cabeceras de los segmentos incluyen un *ACK* que identifica el número de secuencia que que debe traer el siguiente byte esperado del Host emisor.

El primer segmento enviado por el Cliente contiene los bytes desde el 701 hasta el 800. El campo *ACK* informa que el siguiente número de secuencia para el byte esperado es 401. El servidor responde con un *ACK* que indica que los bytes 701 al 800 ya han sido recibidos, por lo que el número de secuencia correspondiente al siguiente byte esperado del cliente será 801. Cabe hacer notar que el emisor no necesita esperar un *ACK* para poder enviar más datos.

En el diálogo el emisor envía segmentos rápida y sucesivamente, que comienzan con los bytes 801, 901 y 1001. El *ACK* de respuesta del servidor indica que estos segmentos han sido recibidos de manera satisfactoria.



En la **(fig. 5.8)**, se muestra una transferencia de datos en la que se pierde el primer segmento. Después de un cierto intervalo de tiempo el segmento es transmitido nuevamente. Debe observarse que, una vez que el segmento perdido llega a su destino, el receptor envía un *ACK* que solo confirmará que los tres segmentos llegaron a salvo.



El diálogo continúa en la **(fig. 5.9)** en la que se muestra un intercambio de datos en ambas direcciones. Cada extremo se encarga de numerar sus propios datos y enviar un *ACK* por cada dato que ha recibido. Por simplicidad se seguirán utilizando segmentos de 100 bytes.

El primer segmento contiene los bytes 1401 hasta el 1500. El cliente aún espera recibir el byte 401 del servidor. Este responde con un segmento que contiene los bytes 401 al 500. De esta manera, tanto el cliente como el servidor continúan transmitiéndose datos uno a otro.

Cierre de una Conexión

La finalización normal de una conexión se lleva a cabo mediante *"tres maneras de despedirse"* , de manera similar a la apertura de la conexión.

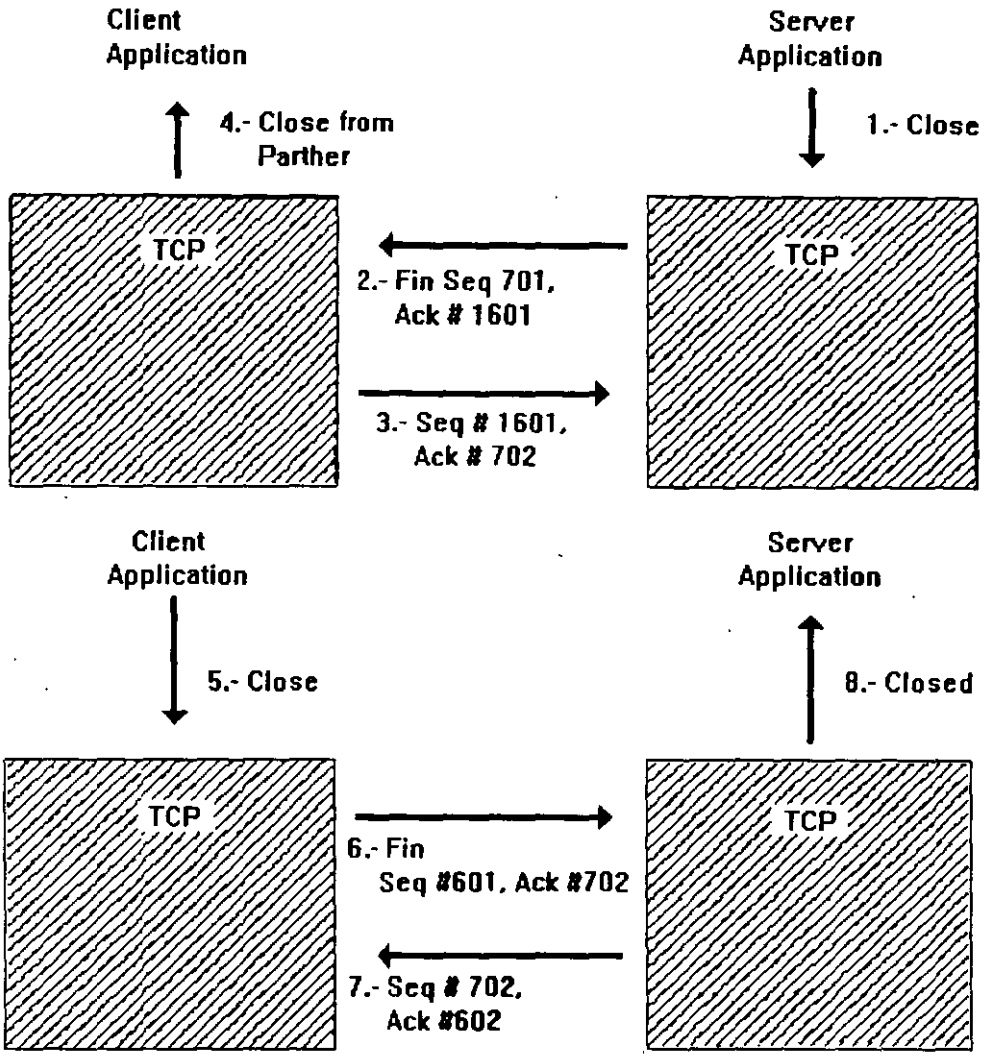
Cualquier extremo de la conexión puede terminar la conexión, siguiendo un patrón determinado análogo a una despedida:

A: Ya terminé. No tengo más datos por enviar.
B: OK
B: Yo también ya terminé
A: OK

O en otro caso:

A: Ya terminé. No tengo más datos por enviar.
B: Yo aún tengo datos por enviar
B: Ya terminé yo también
A: OK

(Fig 5.10).



En el ejemplo es el servidor quien inicia el cierre de conexión apegandose al esquema siguiente:

- El servidor finaliza su sesión y le pide a TCP cerrar la conexión.
- El servidor TCP envía un segmento FIN, informando al otro extremo de la conexión que ya no enviará más información.
- El cliente TCP acusa de recibo el segmento FIN.

- El cliente TCP notifica a sus aplicaciones que el servidor desea cerrar la conexión.
- El cliente solicita a TCP el cierre de conexión.
- El cliente TCP envía un mensaje FIN.
- El servidor TCP recibe el mensaje FIN del cliente y responde con un ACK.
- El servidor TCP notifica a sus aplicaciones que la conexión se ha terminado.

Cierre Repentino

Cualquiera de los extremos de una conexión puede solicitar un cierre repentino de conexión. Esto puede hacerse cuando una aplicación requiera abortar la conexión o cuando TCP detecta un problema de comunicación grave que no puede resolver. Un cierre repentino se consigue enviando al otro extremo de la comunicación un *"Reset"* mediante un indicador en la cabecera de TCP.

Máximo Tamaño del Segmento

Los segmentos grandes tienen un mejor desempeño durante la transferencia de bloques de datos, debido a que las cabeceras hacen un uso más reducido de memoria y del porcentaje de ancho de banda. Por ejemplo, si un datagrama tiene una cabecera de IP de 20 bytes, una cabecera de TCP de 20 bytes y 60 bytes de datos, se consumirán un poco más del 40% de los recursos.

No obstante, no todas las computadoras tienen la capacidad de manejar segmentos muy grandes. Una pequeña computadora de escritorio puede tener la capacidad de procesar segmentos de tamaño no mayor a 1 Kbyte.

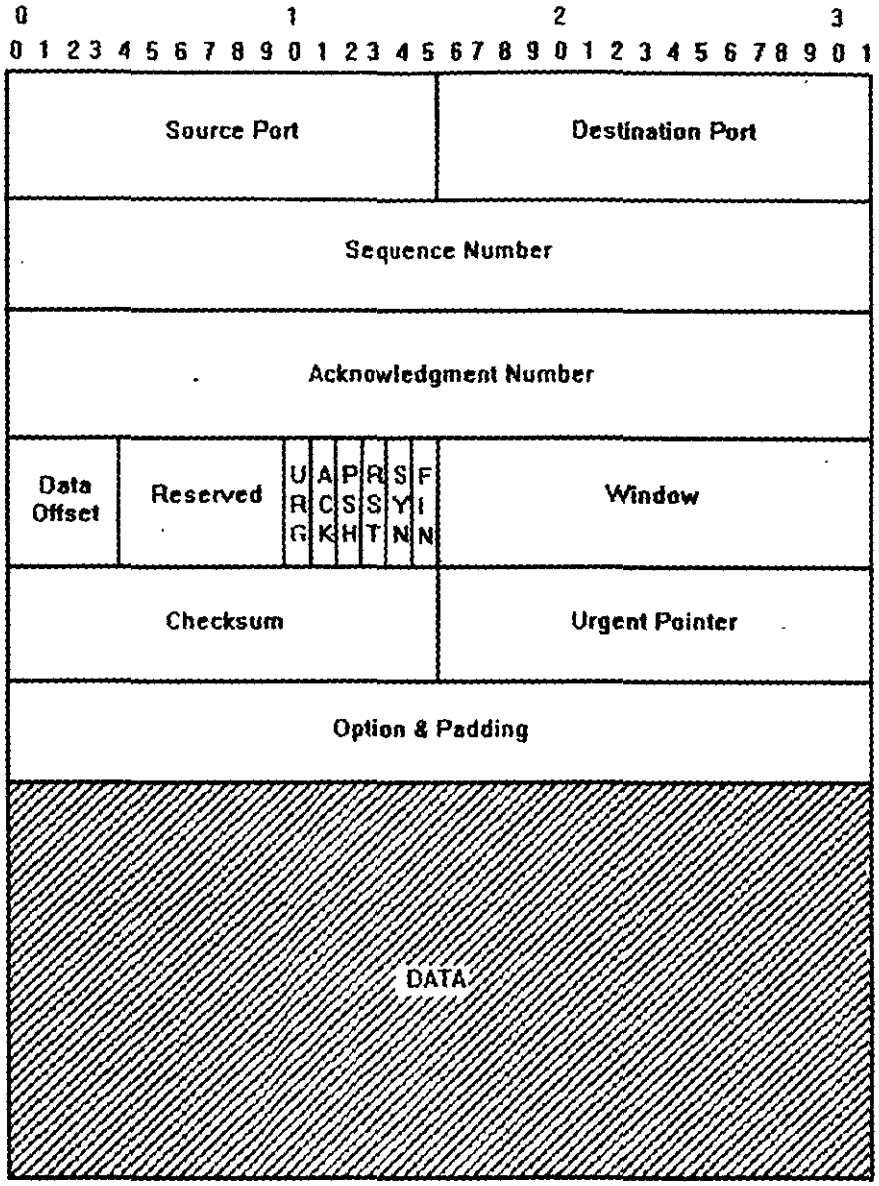
Una computadora mediana puede manejar segmentos de hasta 4 Kbytes. Por su parte, una supercomputadora tiene la capacidad de manejar segmentos de hasta 16 Kbytes.

Es necesario considerar los límites de la transmisión máxima de datagramas en la elección de los tamaños de los segmentos. Por ejemplo, Ethernet establece un tamaño máximo de datagrama del orden de 1.5 Kbytes. Debe recordarse que un segmento debe tener cabida dentro de un solo datagrama.

Durante el establecimiento de la conexión, cada extremo puede declarar el *máximo tamaño de segmento (MMS)* que espera recibir, es decir; cada extremo de la conexión declara la cantidad máxima de datos que puede ser llevada en un segmento. El tamaño de la cabecera de TCP no está incluido en el valor del MMS.

Existe un pequeño problema en la elección del MMS, ya que una cabecera de IP puede variar entre 20 y 60 bytes. Actualmente, las cabeceras de TCP solo llevan consigo una opción durante el establecimiento de la conexión, por lo que es razonable asumir que consistirán de 20 bytes. La situación más común es que las cabeceras de IP también se conforman de 20 bytes. Para el destino en una red local, el MMS se fija frecuentemente en 40 bytes menos que la máxima unidad de transmisión para la interface del sistema de la red. Un MMS estimado de 576 - 40 = 536 se utiliza para destinos remotos.

Cabecera de TCP



La **(fig. 5.11)** muestra un segmento, es decir; cabecera de TCP y sus datos. La cabecera de TCP comienza con los identificadores de puerto origen y destino. El campo Número de Secuencia identifica la posición que guardan los datos en una trama de datos de envío en ese segmento. El campo *ACK* identifica el lugar donde se espera se localizará el siguiente segmento de llegada en la trama de datos a recibir.

Uso de los Campos de la Cabecera durante el establecimiento de la Conexión.

El primer segmento que se envía para comenzar una conexión lleva el campo *SYN* fijo en uno (1) y el campo *ACK* fijo en cero (0). El Campo *Número de Secuencia* contiene el *número que inicia la secuencia.* El campo *Ventana* contiene el *tamaño inicial de la Ventana de Recepción.* El campo de *Opciones* puede contener el tamaño máximo de segmento (*Maximum Segment Size MMS),* que espera el iniciador de la conexión. Esta es la única opción de TCP y se utiliza prácticamente en todas las implementaciones.

En respuesta de aceptación de una conexión, los campos *SYN* y *ACK* se restablecen a uno (1). El número inicial de secuencia de quien responde se encuentra en el campo *Número de Secuencia* y el tamaño de la ventana de recepción en el campo *Ventana.* El campo *Opciones* puede contener el tamaño máximo de segmento que espera recibir quien responde.

Un intento de conexión puede ser rechazado enviando una respuesta que restablezca el campo *RST* con uno (1).

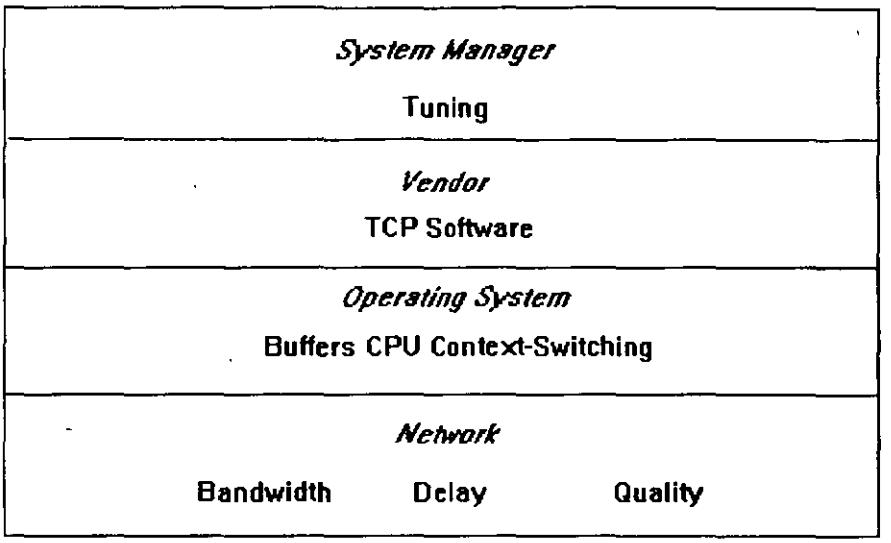
El campo de *Desplazamiento de Datos* contiene la longitud de la cabecera de TCP, medida en palabras de 32 bits. El tamaño máximo de segmento actual se codifica con un introductor de 2 bytes seguido de un valor de 2 bytes, por lo que el tamaño más grande posible será de 65,535 bytes. Cuando no se incluye un valor máximo de tamaño de segmento, se utiliza el valor de 536 bytes por omisión.

Rendimiento

Existen diversos factores que afectan en el rendimiento, los más comunes son los recursos como memoria ancho de banda. Una transmisión de baja calidad ocasiona que muchos datagramas se descarten, al ocurrir esto, se provocan retransmisiones y por ende, el ancho de banda efectivo se recorta. Por ejemplo si el ancho de banda de transmisión esta operando a 3 mbits/s. y es necesario retransmitir la mitad de los datos, entonces el ancho de banda efectivo de transmisión operará unicamente a 2 mbits/s.

Otro factor muy importante en el rendimiento de transmisión, es la capacidad que tenga el host para reaccionar a eventos de mayor prioridad y conmutar rápidamente, es decir; dejar una tarea para atender otra de mayor prioridad.

- Se necesitan también recursos capaces del CPU para deshacerse rápidamente de la carga de trabajo que representa el proceso de cabecera de TCP. Un CPU que no pueda efectuar un checksum rápidamente, puede entorpecer la velocidad de transmisión de datos. En la **(Fig. 5.12)** se explica a nivel bloques la importancia de los diversos factores en el rendimiento de la transmisión.



Relación al Modelo OSI

El *Protocolo de Transporte Clase 4 de OSI (OSI TP4)* ha tomado varias ideas de TCP, no obstante, las diferencias más importantes que se presentan entre ellos son:

- TCP utiliza un solo formato de cabecera. OSI TP4 tiene cabeceras separadas que constituyen un arreglo completo de unidades de datos de protocolo. Éstas ofrecen compatibilidad con diferentes unidades de datos de protocolos que se utilizan en la capa 3 de X.25

- TCP numera bytes de datos. OSI numera unidades de datos de protocolo en secuencia además de que esta es mayor en velocidades altas.

- El tamaño máximo de ventana de recepción de TCP es de 65,535 bytes, mientras que OSI soporta solamente 32,767 unidades de datos.Para soportar anchos de banda de transmisión muy altos, es necesario tener un buffer muy grande.

TCP/IP

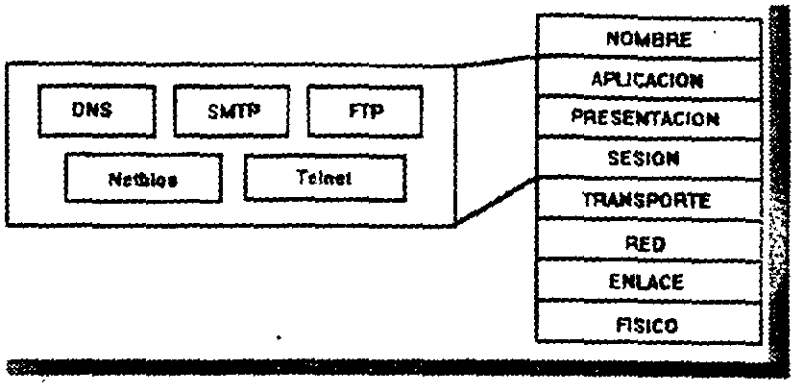
NIVEL 5-7 - APLICACION

- Nivel de Sesión
- Nivel de Presentación.
- Nivel de Aplicación

Notas:

TCP/IP

NIVEL 5-7 SESION - APLICACION



Notas:

TCP/IP

ARQUITECTURA

Protocolo a nivel de Sesión

SMTP	Simple Mail Transfer Protocol.
FTP	File Transfer Protocol.
TELNET	Comunicación de Terminal.
DNS	Domain Name Service.
NSP	Name Service Protcol.

Notas:

TCP/IP

PROTOCOLO S M T P

Simple Mail Transfer Protocol

- Uno de los protocolos más implementados.
- Define cómo transmitir mensajes entre 2 usuarios.
- Se basa en Spooling para el envío de Mensajes.
- Se conoce como envío de mensajes punto apunto.
- Describe la estructura del mensaje y especifica el protocolo para el intercambio de correo.

Notas:

TCP/IP

PROTOCOLO F T P

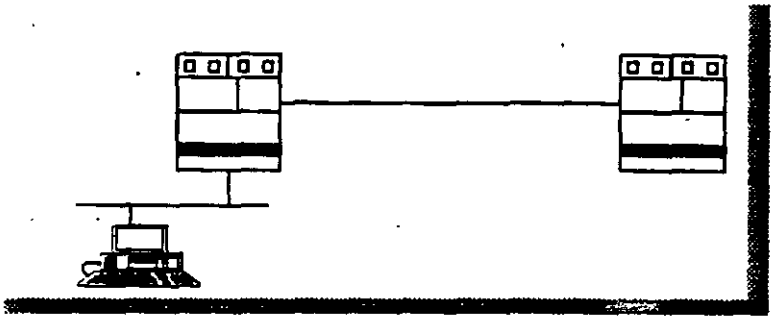
File Transfer Protocol

- FTP permite el envío y recepción de uno o más archivos en forma interactiva.
- Soporta formatos de archivo en ASCII, Binario y EBCDIC.
- Modo de transmisión " Stream ", Bloques o comprimido.
- Permite las manipulaciones sencillas dentro de los sistemas de archivos Locales y Remotos.

Notas:

TCP/IP

PROTOCOLO F T P



Notas:

PROTOCOLO F T P

Comandos	
Abort	Interrupción.
Asccii	Define modo de transferencia a asccii.
Bget	Leer un archivo en modo binario.
Bell	Alarma para indicar fin de transferencia.
Bye	Termina enlace y sale.
Case	Hace el cambio de los nombres de archivo locales a minúsculas.
Cd	Cambio de directorio remoto.

Notas:

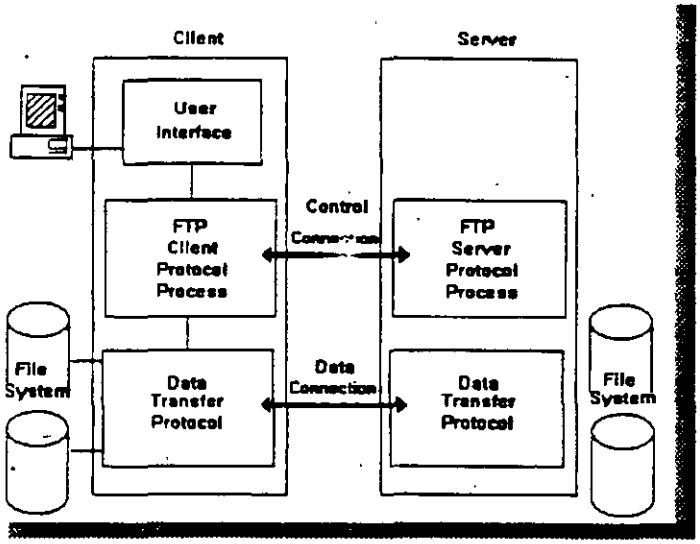
PROTOCOLO F T P

Ejemplo:

```
$ ftp
* Open vax1
<Enter PASS comand)
password:
<user logged inn, default directory)
*
* get remotefile local, file
* put local.file rnewfilename
bye
$
```

Notas:

TCP/IP


Client
Server
User Interface
FTP Client Protocol Process
Control
FTP Server Protocol Process
File System
Data Transfer Protocol
Data
Data Transfer Protocol
File System

Notas:

TCP/IP

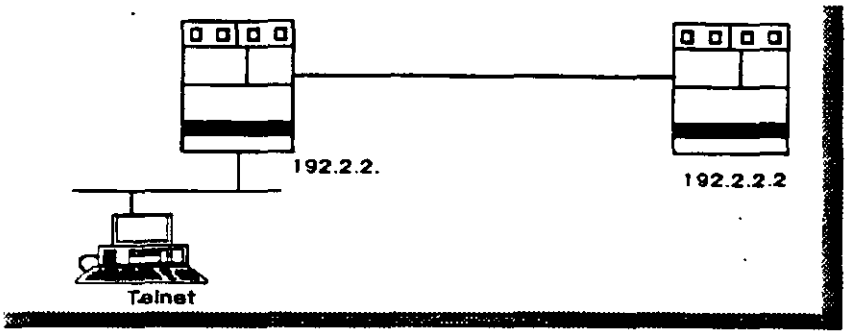
PROTOCOLO TELNET

- Protocolo de Acceso Remoto e interactivo de terminal.
- Brinda una conexión virtual a nodos remotos.
- Permite a los usuarios accesar nodos remotos como si fueran terminales "Físicamente Conectadas" al host.

Notas:

TCP/IP

PROTOCOLO TELNET



Notas:

TCP/IP

PROTOCOLO TELNET

Comandos

Open	Conectarse a un Host
Close	Cerrar sesión actual
Escape	Definir carácter de escape
Exit	Fin de telnet
Local echo	Cambio de eco (encendido, apagado)
Status	Información da cada sesión
?	Ayuda

Notas:

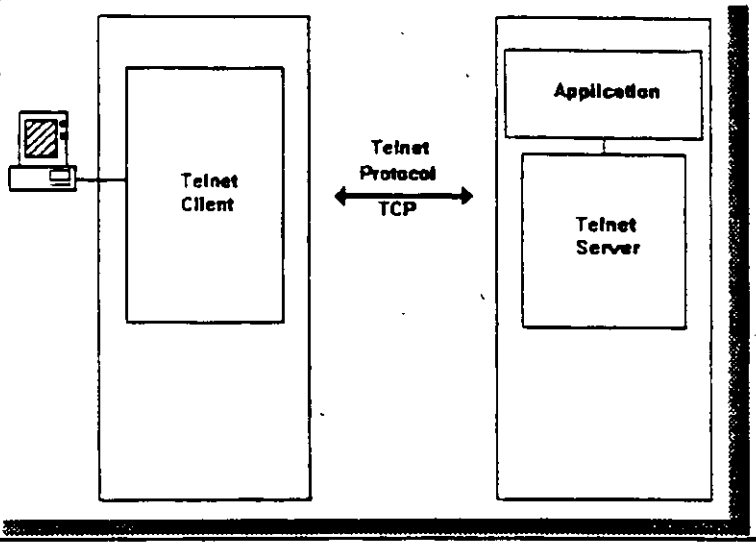
PROTOCOLO TELNET

Ejemplo:

```
Telnet > open apollo1
Trying ... open
Connected to apollo1
Escape Character is ' ^] '
    Username:
    Password:
```

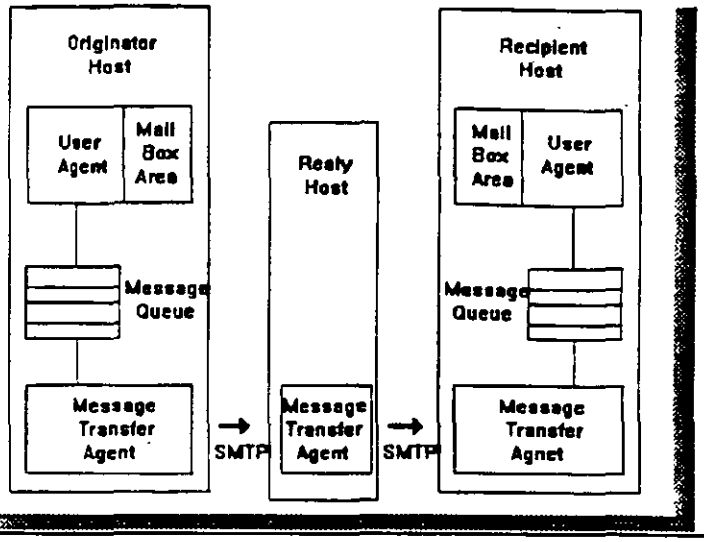
Notas:

TCP/IP


Application
Telnet
Client
Telnet
Protocol
TCP
Telnet
Server

Notas:

TCP/IP


Originator Host
User Agent
Mail Box Area
Message Queue
Message Transfer Agent
SMTP
Realy Host
Message Transfer Agent
SMTP
Recipient Host
Mail Box Area
User Agent
Message Queue
Message Transfer Agnet

Notas:

TCP/IP

D N S

Domain Name Service

- Protocolo de nombramiento.
- Brinda traducción de nombre-dirección IP.
- Dominio: Grupo de Hosts.
- " Domain Name Server ".

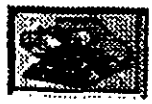
Notas:

DOMAIN NAME SERVICE

- Información del servidor.
  - Dirección Internet.
  - Tipos de Computadora.
  - Lista de servicios brindado por computadoras.
- Servidor.
  - Servidores Maestros.
  - Primario.
  - Secundario.

Notas:

TCP/IP



SERVICIO DE NOMBRAMIENTO

- Host
  - Contiene relación de nombres y direcciones IP sobre cada nodo de la red.
- Name Service
  - Un servicio central de nombramiento. El archivo de nombres en el servidor es similar al archivo "HOSTS".
- Domain Name Service
  - Sistema descentralizado de nombramiento.
  - Utiliza varios archivos para resolver las direcciones de IP.
  - Especificación RFC 1032-1034.

Notas:

SERVICIOS DE NOMBRAMIENTO

Archivo HOSTS

127.1.1.1	Localhost
128.90.1.3	Vax1
128.90.1.9	Sun
128.90.3.9	Vax3
192.1.10.25	Apollo1
192.1.10.95	Hp9000 conta1
192.1.10.93	Hp9000 conta2

Notas:

TCP/IP

PROTOCOLO N F S

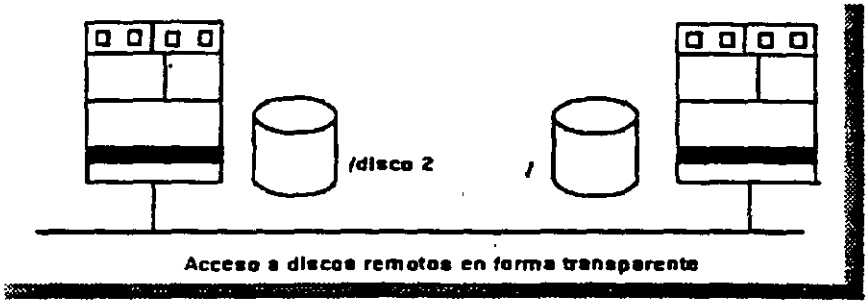
N etwork F ile S ystem

- Originado y popularizado por SUN Microsystems.
- Diseñado para ser portado fácilmente a diferentes sistemas operativos.
- Brinda acceso transparente a sistemas remotos de archivos.
- Los usuarios no necesitan saber la localidad física de los discos.

Notas:

TCP/IP

PROTOCOLO N F S



Notas:

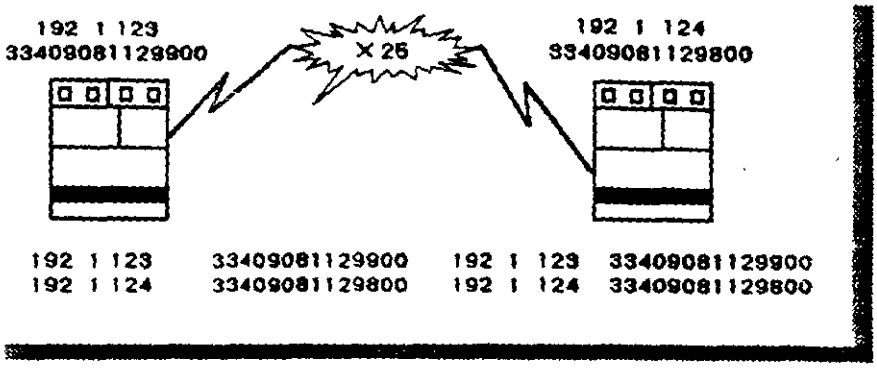
TCP/IP SOBRE X.25

- Implementación de TCP/IP para Redes de área amplia dos opciones interno o externo.
- Generalmente con conexiones dinámicas.
- En caso de no usar la línea ésta se desconecta temporalmente.
- La fragmentación la realiza X.25

Notas:

TCP/IP

TCP/IP SOBRE X.25



Notas:

TCP/IP SOBRE X.25

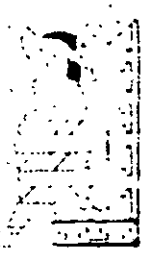
- Se requiere tablas de conversión para determinar equivalencia entre X.25 y TCP/IP.

Ejemplo:	
192.1.20.3	33409081109900
192.1.21.5	38219087113800

Notas:

TCP / IP
ARQUITECTURA, PROTOCOLOS E IMPLEMENTACION

6.-TRABAJANDO CON TCP/IP



Mayo de 1996.

6.0 Trabajando con TCP/IP

FTP (File Transfer Protocol)

FTP es un Protocolo de Transferencia de Archivos que pretende solucionar los problemas más comunes de transmisión de una manera sencilla. Está diseñado para ser operado por usuarios finales o por programas de aplicación.

Escenario de FTP

Se utiliza un grupo estándar de comandos para comunicarse con un servidor remoto FTP pero el usuario final identifica los directorios utilizando las convenciones de nombres del sistema remoto. Usualmente las computadoras no permiten que sus archivos sean manipuladas por extraños, sin embargo, hay ocasiones en las que resulta de gran utilidad crear un área de archivos públicos.

FTP ofrece dos tipos de servicio para la organización de la compartición de información pública y los sistemas de seguridad de archivos privados:

- Acceso a archivos públicos mediante entradas "Anónimas" (*anonymous)*
- Acceso a archivos privados, restringidos a usuarios con identificadores de entrada y claves de acceso.

La conexión inicial con el servidor se denomina conexión de control y se utiliza para enviar comandos al servidor y recibir respuestas de éste.

El indicador de recurso ftp> aparece siempre que la aplicación local de FTP está en espera de algún comando del usuario. Por otra parte, las líneas que comienzan con número contienen mensajes del servidor de archivos remoto.

El Protocolo de Transaferencia de Archivos tienen un estilo de operación característico. Siempre que un archivo es copiado, se establece una segunda conexión y se utiliza para la transferencia de datos. Después de comando *get* (comando para copiar archivos), el FTP local adquiere un puerto adicional y lo informa al servidor. Esto no puede ser visto, pero si se puede observar la respuesta:

200 PORT command succesful.

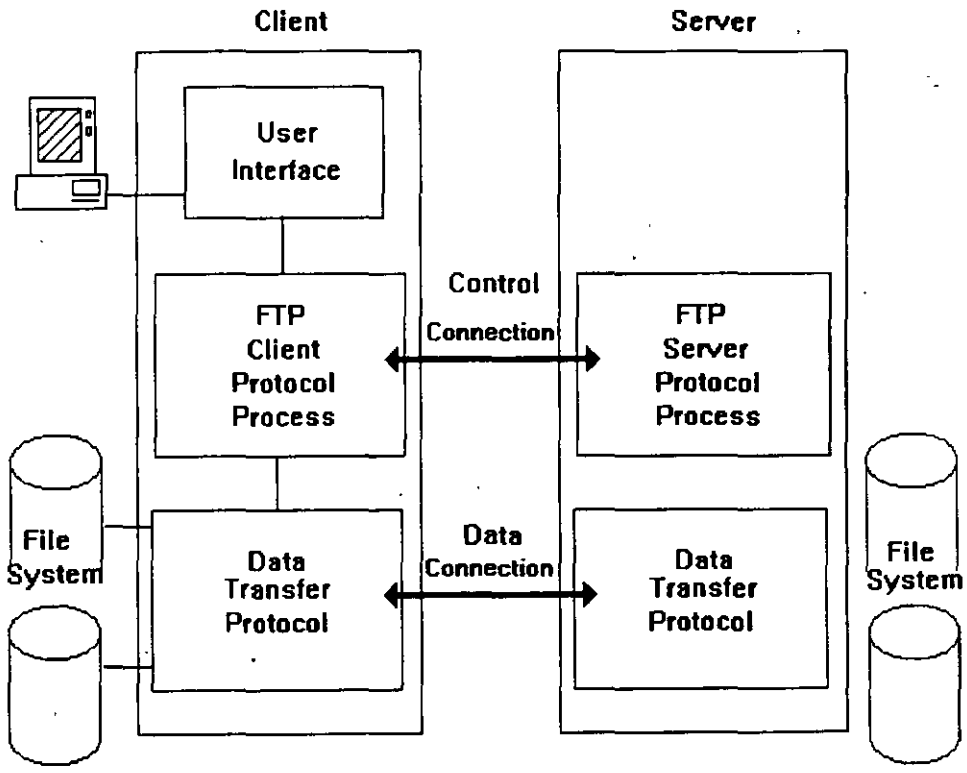
150 ASCII data conecction for diplom_1.txt (128.36.12.27,1401)

La respuesta indica que el FTP local que se encuentra en la dirección d
128.36.12.27 del usuario, ha adquirido el puerto 1401 para la conexión.

El modelo FTP

El usuario interactúa con el proceso de FTP del cliente local. El software
cliente local lleva una conversación formal con el software FTP del serv
remoto acerca del control de la conexión.

Las transferencias de datos reales se llevan a cabo en la conexión sepa
de datos que se crea exprofeso. Existe un Protocolo de Transferencia
datos que se utiliza a lo largo de toda la conexión de datos. **(Fig. 6.1).**



Comandos de FTP

Existen comandos de autentificación que permiten la identificación del usuario *(userid)*, la clave de acceso *(password)* y su cuenta en general para ser utilizada en un grupo de actividades de FTP.

Existen comandos de transferencia de archivos que permiten al usuario:

- Copiar un solo archivo entre hosts.
- Copiar varios archivos entre hosts.
- Añadir un archivo local a uno remoto.
- Copiar un archivo y añadir un número a su nombre de tal manera que el nombre sea único.

También se cuenta con comandos de manejo de archivos que permiten al usuario:

- Listar archivos en directorios.
- Identificar el directorio en el que se está operando, así como cambiar de directorio de trabajo.
- Crear y remover directorios.
- Renombrar y Borrar archivos.
- Los comandos de control permiten al usuario:
- Identificar si se van a transferir datos ASCII, EBCDIC o binarios.
- Establecer si la estructura del archivo está en series de bytes o como secuencia de registros.
- Describir de que manera se transferirán los archivos, como tramas de bytes, secuencias de bloques o en formato comprimido.
- Los comandos que se envían a través de la conexión de control, tienen un formato fijo.

La interface al usuario generalmente incluye comandos adicionales que permiten al usuario la personalización del entorno local, como:

- Pedir a TCP que emita una señal sonora al final de la transferencia.
- Pedir a TCP que imprima un *hash symbol* "#", en cada bloque de datos transferidos.
- Instalar la adaptación automática de las letras mayúsculas en un nombre de archivo, o instalar una tabla de equivalencia de caracteres para un cambio automático en los nombres de los archivos transferidos.

El juego completo de funciones soportadas por un host en particular, pueden conocerse entrando a la aplicación FTP y tecleando *help*.

Es importante recalcar el orden en el esquema de conexión para una transferencia de datos:

- El cliente local toma un puerto nuevo y hace uso de la conexión de control para indicar el número de ese puerto al servidor de FTP.
- El servidor de FTP se conecta al puerto nuevo del cliente.
- Se transfieren los datos.
- Se termina la conexión.

Se puede hacer uso de otro esquema. Si el número de puerto es enviado mediante el comando *PASV,* el servidor espera los datos de conexión provenientes del puerto del cliente. Por convención, el servidor utiliza el puerto número 20 para fines de conexión de datos.

El Protocolo de FTP

Existen muchos elementos que conforman el protocolo de transferencia de archivos :

- El formato de los datos que serán transferidos.
- Los comandos y parámetros relacionados que se envían a la conexión de control.
- Los códigos numéricos de acuse de recibo.

Comandos de TCP

El conjunto de comandos de FTP que puede ser enviado en la conexión de control se resume a continuación:

Comandos de Control de Acceso

Los comandos y parámetros que definen el acceso que tiene un usuario específico a los archivos de un host remoto, se definen a continuación:

COMANDO	DEFINICIÓN	PARÁMETROS
USER	Identifica al usuario	Userid
PASS	Da una clave de acceso	Password
ACCT	Da una cuenta a ser cargada	Account
REIN	Reinicializa	Ninguno
QUIT	Logout	Ninguno
ABORT	Aborta el comando inmediato anterior y su transferencia de datos asociada.	Ninguno

Comandos de Manejo de Archivos

Estos comandos permiten al usuario ejecutar funciones típicas de posicionamiento de directorios y manejo de archivos en un host remoto:

COMANDO	DEFINICIÓN	PARÁMETROS
CWD	Cambiar a otro directorio del servidor.	Nombre del directorio
CDUP	Cambiar al directorio padre.	Ninguno
DELE	Borrar un archivo	Nombre del archivo
LIST	Listar información de los archivos	Nombre del directorio, lista de archivos o ninguno para obtener información del directorio de trabajo.
MKD	Crear un directorio.	Nombre del directorio
NLST	Listar los archivos en un directorio.	Nombre del directorio o ninguno para el directorio de trabajo.
PWD	Imprimir el nombre del directorio de. trabajo	Ninguno
RMD	Borrar un directorio.	Nombre del directorio.
RNFR	Identificar un archivo a ser renombrado.	Nombre del archivo.
RNTO	Renombrar un archivo.	Nombre del archivo.
SMNT	Montar un sistema de archivos	Identificador.
	diferente.	
TYPE	Identificar el tipo de datos e imprimir el format (si existe) para transferencia	ASCII,EBCDIC, Imagen/Binario, No impreso, Telnet, ASA.
STRU	Organización del archivo	Archivo o Registro
MODE	Formato de Transmisión	Trama, Bloque o Comprimido.

Comandos para el Establecimiento del Formato de los Datos

Los siguientes comandos se utilizan para establecer la combinación del formato de los datos, el formato de los archivos y el formato de la transmisión que se usará cuando se copien archivos.

COMANDO	DEFINICIÓN	PARÁMETROS
TYPE	Identifica el tipo de los datos e imprime el formato (si existe) para la transferencia	ASCII, EBCDIC, Imagen/Binario, No impreso, ASA, Telnet.
STRU	Identifica la Organización del archivo	Archivo o Registro
MODE	Indica el formato de Transmisión	Trema, Bloque o Comprimido

Comandos de Transferencia de Archivos

Estos establecen los datos de conexión, los archivos a copiar y soportan la recuperación de reinicio.

COMANDO	DEFINICIÓN	PARÁMETROS
ALLO	Reservan espacio de almacenamiento suficiente para los datos consecuentes.	Un número entero de bytes.
APPE	Añade un archivo local a uno remoto.	Nombres de Archivos.
PASV	Identifica la dirección de red y puerto a ser utilizada para la conexión de datos, que será inicializada por el *cliente*.	Dirección de IP y Número de Puerto.
PORT	Identifica la dirección de red y puerto a ser utilizada para la conexión de datos, que será inicializada por el *servidor*.	Dirección de IP y Número de Puerto.
REST	Identifica un marcador de reinicio (para ser se seguido por el comando de transferencia que reiniciará).	Valor del Marcador.
RETR	Carga un archivo	Nombres de Archivos.
STOR	Guarda un Archivo.	Nombres de Archivos.
STOU	Guardar único: Crea una versión de un archivo, con un nombre único.	Nombres de Archivos.

Comandos Misceláneos

Este último bloque de comandos resulta de gran utilidad para el usuario final.

COMANDO	DEFINICIÓN	PARÁMETROS
HELP	Devuelve la información acerca de la implementación del servidor.	Ninguno.
NOOP	Le pide al servidor que le responda: "OK"	Ninguno.
SITE	Utilizado para subcomandos específicos del servidor que no son parte del estándar, pero pueden requerirse en donde se encuentra el servidor.	Ninguno.
SYST	Le pide al servidor que identifique su sistema operativo.	Ninguno
STAT	Pide parámetros de información y el status de la conexión.	Ninguno

Desempeño:

La eficiencia de las operaciones de transferencia de archivos, dependen de diversos factores:

- Eficiencia en el acceso al sistema de archivos del Host.
- Necesidad de Procesamiento para reformatear datos.
- Interacción con los servicios de TCP.

Es muy importante que una implementación incluya algún chequeo interno del status de las conexiones de FTP. Los sistemas y las conexiones de comunicación suelen presentar fallas. Un proceso de FTP puede ser abandonado y quedarse en espera por mucho tiempo, desperdiciando muchos recursos, si el servicio no verifica por si mismo las conexiones.

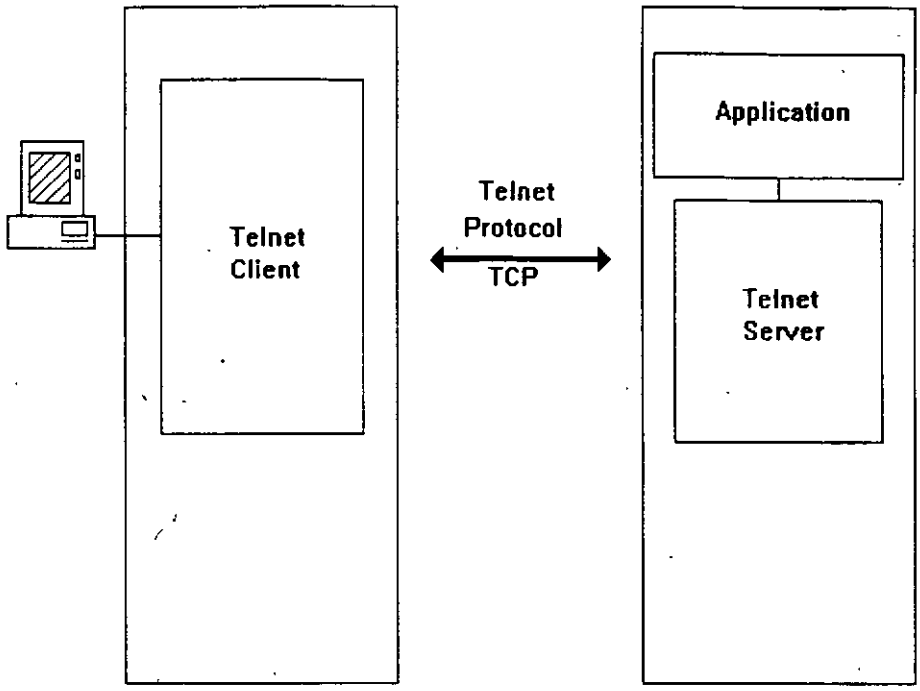
Relación con el Modelo OSI

El Estándar de OSI de Manejo, Acceso y Transferencia de Archivos (FTAM), define un nivel de funcionalidad muy alto con respecto a FTP. Las diferencias en los sistemas de archivos se superan definiendo un almacenamiento de archivos virtual y genérico. Los atributos de los archivos se definen en forma de sistema invariante. Algunos de los puntos a favor de FTAM son:

- FTAM soporta la transferencia de unidades individuales de datos.
- Las estructuras complejas de archivos, como archivos jerárquicos o archivos accesados mediante teclas definidas por el usuario.
- Existen comandos que permiten al usuario navegar en el archivo a efecto de encontrar un dato específico para lectura o para escritura.
- Se pueden definir Grandes estructuras de datos .
- Se cuenta con mecanismos que soportan el control de acceso sensitivo.
- Se pueden imponer restricciones en el acceso concurrente.

El precio que se tiene que pagar por toda esta serie de ventajas, es que FTAM es complejo y su desarrollo es difícil. Se requiere del consumo de muchos recursos.

Telnet



Como se muestra en la **(fig. 6.2)**, un usuario en una terminal interactúa con su proceso local de cliente Telnet. Este cliente intercambia datos con el servidor Telnet remoto mediante una conexión de TCP. El Servidor Telnet interactúa con las aplicaciones emulando una terminal nativa.

Protocolo de Comandos de Telnet

Los comandos de Telnet se representan por un byte *IAC (Interptret **As Command)***, seguido de uno o más bytes de código.

La siguiente tabla muestra los acrónimos con sus correspondientes núm decimales, cada uno deberá de ir precedido por 255 cuando se en tra de una conexión Telnet.

ACRONIMO	COMANDO	CODIGO
BRK	Cortar Secuencia	243
IP	Interrumpir Proceso	244
AO	Abortar Salida	245
AYT	¿ Estás Ahí ?	246
EC	Eliminar Cáracter	247
EL	Eliminar Línea	248
GA	Proseguir	249

Relación con el Modelo OSI

El Servicio de Terminal Virtual de OSI soporta el acceso remoto a terminal. Como es de esperarse, el estándar OSI es mucho más complic que Telnet, permitiendo establecer un elaborado entorno de termin virtuales. OSI define:

- Un modo A de operación que reensambla la manipulación Full-Du de un teclado y Monitor virtual.
- Un modo S de operación que modela diálogos síncro bidireccionales.

El proceso de negociación de OSI se diseñó más complejo que el de Telnet, obstante; el escenario básico de negociación de OSI de *invite/accept/rejec* semejante al *DO/DON'T/WILL/WON'T* de Telnet.

Existe un sin número de opciones que pueden ser utilizadas para describir Terminal Virtual de OSI.

OSI une grupos de opciones en arreglos predefinidos. Se puede establecer completo un Entorno de Terminales Virtuales rápidamente mediante selección de ese arreglo. Esto evita un intercambio prolongado de negocia al principio de la sesión.

NFS, RPC y NIS

El Modelo NFS

NFS como se mencionó anteriormente, es un Sistema de Archivos de Red (**N**etwork **F**ile **S**ystem) que tiene una gran influencia de UNIX. NFS trabaja mejor con una estructura jerárquica de directorios, frecuentemente los archivos de NFS se toman como tramas de bytes sin estructura.

Siguiendo las convenciones de UNIX, NFS está definido para ser un sub-árbol de directorios localizado en un solo dispositivo físico. En una computadora UNIX, los sistemas de archivos localizados en diferentes dispositivos, son unidos para formar la estructura completa del directorio de la computadora, que contiene un solo directorio raíz.

Los directorios de UNIX y sus archivos, son identificados por nombres de rutas que se forman listando los nombres a lo largo de un camino y a partir del directorio raíz, separando esos nombres con una diagonal (/).

La sintaxis que se usa en otros sistemas para escribir los nombres de las rutas puede ser diferente. NFS asume que todos los archivos pueden identificar un nombre de camino.

El Protocolo de NFS

El alma de NFS es que el servidor debe tener el mínimo de información del cliente que recordar, de manera que la recuperación de un cliente cuando algo suceda en el servidor sea menos costosa y más simple. NFS se construye frecuentemente sobre UDP, un UDP no es confiable, además las peticiones al servidor se repiten después de un período de tiempo. Por esta razón los servidores de NFS usualmente guardan un registro de las transacciones recientes de tal forma que las peticiones duplicadas puedan manejarse correctamente.

RPC (Remote Procedure Call)

Un RPC se envía de un cliente a un servidor en un mensaje con formato determinado. La interacción de RPC puede ser síncrona o asíncrona.

En una interacción síncrona el cliente espera la respuesta del s dor. otra parte, en una interacción asíncrona el programa del cliente nc era respuesta, sino que continúa con su ejecución. Por medio de a mecanismo local se le notifica al cliente la respuesta.

A RPC no le interesa que protocolo de transporte se esté utilizando para ll estos mensajes. En el mundo de TCP/IP, RPC corre tanto sobre TCP c sobre UDP. RPC puede ser implementado en otros transportes como ISO o TP4. RPC esconde el transporte actual de los niveles altos de serv como pueden ser NFS o NIS, lo que significa que las aplicaci cliente/servidor construidas sobre RPC son transportables, es decir; pue ejecutarse en donde RPC pueda ejecutarse.

NIS (Network Information Service)

El NIS es un servicio de base de datos de directorios diseñado por Microsystems. Las bases de datos de NIS se denominan Mapas y encuentran almacenadas en uno o más servidores. Un mapa de NIS cont una clave y alguna información sencilla correspondiente a ésta. Un conj de mapas NIS contiene la información de la configuración de una Red.

Los mapas de NIS se utilizan para convertir los nombres de los Ho direcciones de IP, los nombres de los programas de RPC a eros Programas, y los servicios a números de puerto.

Si un administrador desea que todas las computadoras de una red apare como una sola, todos los userid y passwords se guardarán en un mapa ce de NIS.

En un sistema que utiliza NIS siempre se encontrará una librería de sub rut a efecto de que los programas que requieran conocer información, c direcciones de IP, números de Programas de RPC, etc., hagan uso de ma de NIS para reemplazar o complementar los archivos normales configuración.

El Protocolo de NIS

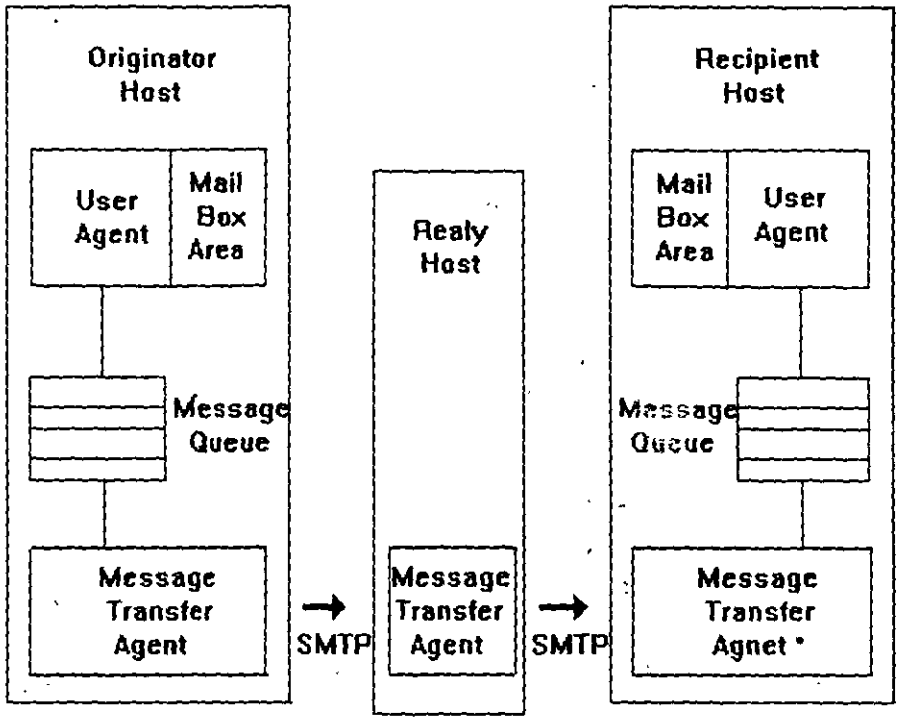
Este protocolo se construyó por encima de RPC, EDP, TCP y UDP. Los ma se accesan vía el programa 100004 RPC. Los procedimientos principales d NIS son:

- Encontrar la correspondencia de una clave con un valor.
- Leer la primera entrada en un mapa.
- Leer la siguiente entrada en un mapa.
- Cargar todas las entradas en un mapa.

UDP se utiliza para los primeros tres procedimientos. TCP se utiliza para el último debido a que requiere una gran transferencia de datos.

En un mapa de NIS se pueden guardar todos los Userids y los Passwords. Esto brinda la oportunidad de controlar los privilegios de acceso del usuario desde el servidor maestro. Existe un programa RPC llamado *yppassword* que permite a los usuarios finales actualizar sus claves personales guardadas en el servidor de NIS maestro.

Correo Electrónico



En la **(fig. 6.3)** se muestran los elementos de un sistema de corre electrónico.

User Agent es el nombre formal que recibe un programa de correo deseablemente debe ofrecer lo siguiente:

- Desplegar la información acerca de mensajes que llegan y almacenan temporalmente en el buzón del usuario.
- Guardar en archivos locales, los mensajes enviados y recibidos.
- Indicar al usuario el asunto de cada mensaje
- Proporcionar un buen editor de texto a efecto de poder es mensajes.

El correo se prepara con la asistencia de la aplicación *User Agent*. El *User Agent* pone el correo en cola de espera de una aplicación separada, denominada *Message Transfer Agent* encargada de establecer comunicación con Host remotos y transmitir el correo.

Los términos *User Agent* y *Message Transfer Agent* son utilizados en el sistema estándar de mensajes X.400 y denotan entidades que son válidas para *SMTP (Simple Mail Transfer Protocol).*

El correo puede ser enviado directamente del emisor al receptor, o utilizando *Message Transfer Agents* como intermediarios. Cuando un correo se envía a través de un *Message Transfer Agent*, el mensaje completo se envía a un Host intermedio, en donde se almacena hasta que pueda ser enviado nuevamente hasta su destino. Los sistemas de correo que hacen uso de intermediarios se conocen como sistemas *store-and-foward*.

En el Host receptor, el correo se pone en una cola de espera de entradas, y posteriormente se pasa a un área de almacenamiento del buzón del usuario. Cuando el usuario invoca un programa *User Agent*, éste le despliega generalmente un resumen del correo recibido que tiene en espera en el buzón.

Nombres y Dominios de Correo

La tarea principal del correo electrónico es entregar mensajes a los buzones receptores. Se envía un acuse de recibo al emisor del correo. En los estándares de correos de Inter-Redes se asume que los buzones son fuentes y receptores de correo.

Los receptores de correo de Inter-Redes se definen con un nombre seguido de un patrón general:

Parte Local@Nombre del Dominio

El formato de la Parte local puede variar dependiendo del dominio. Un alias es un caso especial de la forma general y sigue el siguiente patrón:

Userid@Host-Nombre del Dominio

La parte del Nombre del Dominio de un nombre de correo de una Inter-Red debe ser un nombre lógico que identifique, más que a una computadora, el dominio del correo. Un *Message Transfer Agent* verifica el nombre del dominio en una base de datos, generalmente en un *Servidor de Nombres de Dominios (DNS)*, a efecto de descubrir si existe un host de intercambio de correo al cual se deba turnar el correo.

Cuando un correo llega a un intercambiador de correos, se verificará la Parte Local en un archivo alias convertido a un Userid y nombre de Host, o se utiliza cualquier tipo de identificador de correo en la Red destino.

SMTP (Simple Mail Transfer Protocol)

Este protocolo define una manera directa de navegar a través de los Host de una Red. El protocolo SMTP juega los papeles de emisor y receptor. El emisor establece una conexión TCP con el receptor que utiliza el puerto número 25.

Durante una sesión de SMTP el emisor y el receptor intercambian una serie de comandos y respuestas. Primeramente identifican los nombres de los dominios de sus propios Host, entonces el emisor ejecuta una transacción de correo valiéndose de:

- Identificación del emisor del correo.
- Identificación de los receptores del correo.
- Transmisión de los datos del correo.
- Transmisión de un código que indicará que el correo está completo.

Al finalizar la transacción, el emisor tiene la posibilidad de:

- Iniciar otra transacción.
- Convertirse en receptor.
- Cerrar la conexión.

SMNP (Simple Management Network Protocol)

Un entorno SMNP se conforma de uno o más estaciones de manejo y un grupo de elementos de Red. Todos los elementos que participan en comunicaciones pueden ser manejados: Hosts, Gateways, Hubs, Puentes e incluso Modems, Multiplexores y Switches de Datos. Las estaciones de Monitores que, capturan información del tráfico de la Red de manera pasiva, pueden compartir sus datos con un manejador SMNP.

Un elemento manejado contiene un software agent y una base de datos especial denominada *MIB (Management Information Base)*, que contiene:

- Información del status de los sistemas y dispositivos.
- Estadísticas de Desempeño.
- Parámetros de Configuración.

Mensajes de SMNP

Los manejadores y los elementos de una Red se comunican entre sí enviándose mensajes de SMNP. Sólo existen cinco tipos de Mensajes:

- Get Request (Tomar Petición): Contiene la(s) variable(s) que el administrador desee leer del MIB.
- Get Next Request (Tomar Siguiente Petición): Proporciona una manera de leer secuencialmente a través del MIB.
- Get Response (Tomar Respuesta): Se envía a manera de acuse de recibo de un mensaje Get Request.
- Set Request (Asignar Petición): Se utiliza para asignar un valor a una o más variables.

- Trap (Bloquear): Se utiliza para reportar eventos como:
  - Autoreinicialización.
  - Fallo de enlace local.
  - Enlace funcionando nuevamente.
  - Sin Respuesta.

Es parte de la filosofía de SMNP que el número de mensajes de bloqueo que se envíen sea el menor posible. Frecuentemente, una estación de administración se asignará a la lectura de estadísticas con cierta periodicidad. Pasado ese límite de tiempo, las estadísticas se podrán restablecer en cero.

La implementación conserva su simplicidad al limitarse únicamente al intercambio de estos cinco mensajes, mientras no existan límites en la funcionalidad de SMNP.

Formato de los Mensajes de SMNP

Un mensaje de SMNP consiste en una secuencia de elementos:

- Versión del protocolo.
- Nombre de la comunidad.
- Unidad de Datos del Protocolo de Mensajes, que puede ser cualquiera de los cinco mensajes mencionados anteriormente.

Una unidad de Protocolo de Petición o respuesta se compone de :

- Una identificación de Petición (request-id), utilizada para relacionar peticiones con respuestas.
- Un campo de status de error, que será igual a cero en peticiones, y utilizado en respuestas que reporten problemas para cumplir con una petición.
- Un campo de índice de error, que será cero en peticiones, y se usará para describir problemas más detalladamente.

- Una secuencia de valores e identificadores de objetos. En un Get Request los valores son nulos. En un Set Request o un Get Response, los valores se completan.

Una Unidad de Bloqueo de Datos de Protocolo contiene:

- El tipo de elemento que genera el bloqueo.
- La dirección de IP de ese elemento.
- El tipo de bloqueo, p.e. pérdida del enlace.
- Un código de bloqueo más específico.
- Un sello de tiempo *(timestamp)*
- Parámetros relacionados con el bloqueo.