UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

FACULTAD DE ESTUDIOS SUPERIORES
CUAUTITLAN

REDES DE COMPUTADORAS

TCP/IP: CONFIGURACION DE LOS SERVICIOS BASICOS EN EL SISTEMA SOLARIS 2.5.1

TRABAJO DE SEMINARIO

QUE PARA OBTENER EL TITULO DE:
LICENCIADO EN INFORMATICA
P R E S E N T A :

JOSE LUIS GARZA RIVERA

ASESOR: LIC. CARLOS PINEDA MUÑOZ

CUAUTITLAN IZCALLI, EDO. DE MEX. 1997

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE ESTUDIOS SUPERIORES CUAUTITLAN
UNIDAD DE LA ADMINISTRACION ESCOLAR
DEPARTAMENTO DE EXAMENES PROFESIONALES

U. N. A. M.
FACULTAD DE ESTUDIOS SUPERIORES-CUAUTITLAN

DEPARTAMENTO DE EXAMENES PROFESIONALES

DR. JAIME KELLER TORRES
DIRECTOR DE LA FES-CUAUTITLAN
P R E S E N T E .

AT'N: ING. RAFAEL RODRIGUEZ CEBALLOS
Jefe del Departamento de Exámenes
Profesionales de la FES-C.

Con base en el art. 51 del Reglamento de Exámenes Profesionales de la FES-Cuautitlán, nos permitimos comunicar a usted que revisamos el Trabajo de Seminario:

Redes de Computadoras. IGRP/IP: Configuración de los servicios básicos en el sistema Solaris 2.5.1.

que presenta el pasante: José Luis Gorda Rivera

con número de cuenta: 9361586-8 **para obtener el Título de:**

Licenciado en Informática.

Considerando que dicho trabajo reúne los requisitos necesarios para ser discutido en el EXAMEN PROFESIONAL correspondiente, otorgamos nuestro VISTO BUENO.

A T E N T A M E N T E .
"POR MI RAZA HABLARA EL ESPIRITU"
Cuautitlán Izcalli, Edo. de México, a 10 **de** Octubre **de 19**97

MODULO:	PROFESOR:	FIRMA:
Modulo I	Lic. Carlos Pinedo Muñoz	
Módulo III	M. en I. Gloria Ponce Venegas	
Módulo III	Ing. Jesús Moisés Hernández Duarte	

DEP/VOBOSEM

"Detrás de un gran hombre siempre hay una gran mujer"

Mi compromiso con la vida es un múltiplo de cuatro. A ellas dedico esta obra :

Mi Madre
Mamá Jessica
Mamá Rosy
Angélica mi compañera

AGRADECIMIENTOS

- A Dios por todas sus bendiciones.
- A mi Madre por lo que soy.
- A la UNAM y en especial a la Facultad de Estudios Superiores Cuautitlán por mi formación.
- A mi tía Arminda por su ayuda.
- A todo el personal administrativo y docente de la carrera por su trabajo.
- A mis compañeros de la Biblioteca de la Facultad por su apoyo.
- A Angélica por su amor.
- A la SPARCserver5 de la investigación por sobrevivir valientemente todos los ataques de mi inexperiencia.

INDICE

INTRODUCCION

TCP/IP se constituye como un conjunto de protocolos que permiten conectar redes diferentes, sin importar sus equipos o sistemas e integrar redes locales a la red global Internet, dando de esta manera acceso a acervos inmensos de información y permitiendo multiplicar el alcance de la organizaciones en proporciones geométricas. Pero para hacer uso de estas bondades es necesario contar con los conocimientos necesarios no solamente de administración de redes sino también de la manera en que funciona Internet, sus protocolos y configuración de sus servicios para accesar a ellos.

Hoy en día, la mayoria de los servidores de red desde su configuración hasta su administración y uso cotidiano ofrecen interfaces gráficas para dicho propósito; lo que, permite un manejo fácil e intuitivo de los mismos ; pero en algunas ocasiones no se cuenta con tales dispositivos gráficos o se requiere trabajar desde terminales no gráficas por ello es necesario realizar estas tareas de manera más directa mediante interfaces de modo texto, ya sea por comandos y/o bien mediante la edición de archivos.

El presente trabajo de seminario pretende ser una guía para la correcta configuración de los servicios básicos de TCP/IP sobre un servidor con sistema operativo UNIX especificamente la versión Solaris 2.5.1 de SUN Microsystems. Entendiendo por configuración de los servicios básicos de TCP/IP los pasos o paramentos indispensables para que el servidor pueda integrarse a Internet y utilizar servicios como son:

- Acceso a servidores remotos.
- FTP
- Servicios de terminal remota.
- DNS
- Finger.

La investigación se divide en cuatro capítulos de la siguiente manera:

El primer capítulo ofrece un panorama general de la evolución de Internet , TCP/IP,UNIX y Solaris, así como los nombres y funciones de los principales organismos dentro de Internet.

El segundo capítulo está enfocado a los conceptos generales de redes y su clasificación, definición y tipos de protocolos de comunicación, los métodos de conmutación de datos y el funcionamiento de manera general de TCP/IP.

Nuestro tercer capítulo conceptualiza más a detalle la transmisión de los datos en TCP/IP, la arquitectura de ruteo de IP y los servicios de TCP/IP en Solaris.

El capítulo final representa la parte práctica de como configurar los servicios de TCP/IP.

OBJETIVOS

General

Describir los elementos fundamentales para la configuración de los servicios TCP/IP en Solaris 2.5.1"

Específico

Configurar los servicios de TCP/IP mediante la edición directa de los archivos correspondientes utilizando editores de modo texto y línea de comandos.

Capítulo 1. Generalidades

Capítulo 1. Generalidades

1.1 Desarrollo de TCP/IP e Internet.

Hoy en día las telecomunicaciones tienen un lugar fundamental en el desarrollo de la ciencia, tecnología y economía a nivel mundial; es así, que la información constituye un recurso compartido a todo lo largo y ancho del globo mediante redes de computadoras. Un ejemplo claro es Internet, la mayor de todas las redes, también llamada la "red de redes" o "supercarretera" de la información.

Internet se expandió de 6,000 computadoras a finales de 1986, llegando a integrar mas de 600,000 equipos para 1991[1] manteniendo un crecimiento exponencial alcanzando varios millones de computadoras y más de 80 millones de usuarios actualmente. Este desarrollo se debe en gran parte a que Internet permite comunicar equipos con sistemas y características distintas; lo cual, es posible gracias a la existencia de ciertos estándares de comunicación, específicamente TCP/IP (Transmition Control Protocol / Internet Protocol), el cual se conforma de una serie de normas de comunicación llamadas protocolos.

Los primeros trabajos para llegar a lo que hoy en día es TCP/IP e Internet datan desde finales de los 60, época en la cual, se comenzaba a extender el uso de las redes de ordenadores, tendiéndose como principales limitaciones las siguientes:

- El software de comunicaciones era totalmente enfocado al hardware, variando enormemente entre fabricante y fabricante.
- El número de redes que se podían interconectar era muy limitado.
- En muchos casos no era posible la conexión de equipos recientes con modelos anteriores.
- La instalación, configuración y uso de los modelos de red existentes era sumamente compleja.
- Equipos de diferentes fabricantes no eran compatibles entre sí.

A finales de los 60's, la agencia de proyectos avanzados del departamento de defensa de los Estados Unidos (DARPA) junto con universidades y otros grupos de investigación de dicho país comienza a realizar trabajos sobre las nuevas tecnologías de comunicación en busca de una solución a la problemática de conectividad. En 1969 surge ARPANET como resultado del proyecto comenzando sus operaciones

[1] RFC 1296, Internet Growth (1981-1991). M. Lottor, SRI Internetional.

con éxito con un total de cuatro nodos, el primero de ellos se instala en la UCLA (Universidad de California en Los Angeles). A pesar de todos los esfuerzos los protocolos de ARPANET eran lentos y producían constantes colapsos al sistema.

En 1974 dos miembros de la IEEE Vinton G. Cerf y Robert E. Kahn publican un trabajo que propone un nuevo conjunto de protocolos "A protocol for Packet Network Information" . Los trabajos de Cerf y Kahn constituyen la base del actual TCP/IP. Desde ese año se comienza a integrar TCP/IP en ARPANET. Para 1981 surge la red BITNET "Because Its Time NETwork" , esta red proporciona correo electrónico y servidores de listas de discusión para el intercambio de información. En 1983 TCP/IP se establece como protocolo oficial de ARPANET, la cual, para ese entonces contaba con 300 hosts, entendiendo por host una computadora con uno o mas usuarios. Al ser TCP/IP protocolo oficial de la red gubernamental, particulares e instituciones de diversa índole que sostenían relaciones con el gobierno buscaron también apegarse a TCP/IP.

Desde sus inicios TCP/IP fue enfocado a satisfacer las siguientes necesidades básicas:

- Acceso de terminales a cualquier host.
- Copia de archivos entre hosts.
- Servicios de correo electrónico a lo largo toda la red.

Las políticas abiertas del ARPANET y las facilidades de TCP/IP permitieron que otras redes se integraran a ARPANET formando en su conjunto lo que desde entonces se comenzó a llamar Internet, apareciendo la primera definición de Internet como un "conjunto de computadoras interconectadas mediante TCP/IP" en 1982. Durante la década de los 80's ARPANET fue la base central o *backbone* de Internet integrándose a ella redes militares, académicas, de investigación, gubernamentales y comerciales de diversa índole. En 1983 se divide ARPANET en dos grandes ramas MILNET dedicada exclusivamente a la milicia y ARPANET enfocada a la investigación y el desarrollo. Para finales de los 80's una nueva red backbone se incorpora a Internet NSFNET (National Sciencie Foundation Network) proporcionando canales de mayor velocidad y enlazando a los 5 principales centros de supercomputo de los Estados Unidos(fig.1). ARPANET deja formalmente de existir en 1990.

AÑO	HOSTS [2]
1969	4
1972	40
1974	100
1983	300
1984	1,000
1986	6,000
1987	10,000
1989	100,000
1991	600,000

figura 1. Internet en Cifras

El desarrollo actual de Internet toma dimensiones inmensurables y el calculo del número de sus usuarios se realiza en base a estadísticas de los accesos a los principales hosts dentro de la red.

Los siguientes datos fueron obtenidos por una compañía llamada Network Wizards, la cual realiza estadísticas semestrales a través de un rastreo en el sistema de servidores de nombres (Domain Name System. DNS) mediante un robot llamado ZONE, este robot estima el número de hosts en Internet, con base a los hosts registrados en el DNS(fig. 2 y 3).

CRECIMIENTO DE INTERNET

FECHA		HOSTS	DOMINIOS
ENERO	1993	1,300,000	21,000
JULIO	1993	1,800,000	26,000
ENERO	1994	2,200,000	30,000
JULIO	1994	3,200,000	46,000
ENERO	1995	4,900,000	71,000
JULIO	1995	6,600,000	120,000
ENERO	1996	9,500,000	240,000
JULIO	1996	12,881,000	488,000
ENERO	1997	16,146,000	828,000
JULIO	1997	19,540,000	1,301,000

figura 2 Datos obtenidos por Network Wizards.[3]

[2] A partir de 1974 los datos son aproximados.
[3] http://www.mit.edu:8001/people/mkgray/net/internet-growth-raw-data.html. 10/09/97

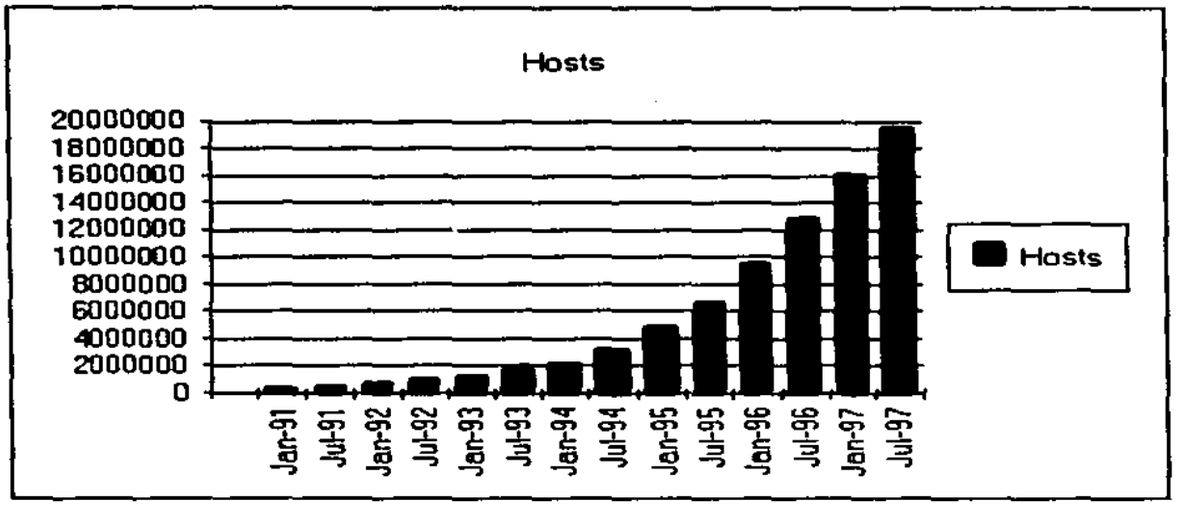


figura 3. Crecimiento de Internet.

Una de las características que dan una gran fuerza a Internet es el hecho de permitir conexiones desde computadoras personales por medio de líneas telefónicas, lo cual extiende el número de usuarios actuales y potenciales, poniendo a Internet fuera del alcance de los posibles mecanismos de medición de su crecimiento.

1.1.1 Internet en México

En México a diferencia de los Estados Unidos, el desarrollo de Internet no fue impulsado por una mancuerna entre milicia y universidades, sino solamente por las instituciones educativas, entre las que destacan el Instituto de Tecnológico de Estudios Superiores Monterrey (ITESM) y la Universidad Nacional Autónoma de México (UNAM).

Nuestro país se enlaza en 1986 al sistema BITNET, a través de una línea propiedad del ITESM, esta línea conectaba a la UTSA (Univesity of Texas in San Antonio) con el ITESM campus Monterrey. Para 1989 , mediante esta misma línea se establece el primer contacto de México con lo que hoy en día conocemos como Internet, el canal utilizado fue una línea analógica privada de 4 hilos a 9600 bps. La computadora conectada en México fue una Microvax-II y su dirección IP numérica era la 131.178.1.1, dicho equipo se encuentra fuera de servicio desde septiembre de 1993.

También en 1989 se estableció el segundo nodo de Internet en México en la Universidad Nacional Autónoma de México, instalado en el Instituto de Astronomía.

La conexión se realizó con el Centro Nacional de Investigación Atmosférica de Boulder, Colorado, en los Estados Unidos (NCAR) utilizando una transmisión digital vía satélite a una velocidad de 56 Kbps .

Posteriormente se unen al ITESM universidades como la Universidad de las Américas (UDLA) en Cholula, Puebla y el Instituto Tecnológico de Estudios Superiores de Occidente (ITESO) en Guadalajara, Jalisco; estas dos instituciones se unen a Internet mediante las líneas del ITESM. Aunque estas primeras conexiones eran de baja velocidad (9600 bps), eran más que suficientes para proveer a las instituciones enlazadas servicios básicos como acceso remoto, transferencia de archivos y correo electrónico.

Entre otras organizaciones que comenzaron a integrarse a Internet en aquel entonces tenemos:

- Colegio de Postgraduados de la Universidad de Chapingo. Estado de México. (COLPOS)
- Centro de Investigación en Química Aplicada. Saltillo, Coahuila.
- Laboratorio Nacional de Informática Avanzada. Xalapa, Veracruz.
- Universidad de Guanajuato. Salamanca, Guanajuato.
- Instituto Tecnológico de Méxicali. Mexicali, Baja California.

Para 1993 en México existían una serie de Redes ya establecidas entre ellas:

MexNET
Red UNAM
Red ITESM
RUT y C
BAJANet
Red Total CONACYT

Internet fue abierto a nivel comercial hasta 1994, con la formación de la Red Tecnológica Nacional (RTN), integrada por MEXnet y CONACYT, esta nueva red proporciona un enlace a 2 Mbps (E1) y se crea la primer red comercial PIXELnet.

Durante 1994 y 1995 se consolidaron redes como RTN creando un backbone nacional, agrupando un gran número de instituciones educativas y comerciales en toda la República. Se calcula que para abril de 1997 existían mas de 150 proveedores de acceso a Internet.

1.2 Algunas organizaciones dentro de Internet

Tanto TCP/IP como Internet se encuentran en un proceso de constante desarrollo en investigación buscando tomar la mayor ventaja posible de las nuevas tecnologías de transferencia de datos para ajustarse a las crecientes necesidades y volúmenes de información, así como el notable número de equipos y usuarios que se integran diariamente a la red . Para dicho propósito existen una serie de organismos dentro de Internet, estos organismos se integran por grupos de investigadores voluntarios de diversas áreas como son gobierno, universidades, empresas, etc. Entre las principales organizaciones encontramos:

1.2.1 Internet Architecture Board (IAB)

Conocida anteriormente como la "Internet Activities Board", esta organización se origina en el año de 1983 con el propósito de vigilar el desarrollo de Internet, aunque hoy en día sus funciones son el desarrollo de nuevos protocolos de TCP/IP, así como el mantenimiento de los actuales .

La IAB divide sus funciones en el año de 1989 formándose dos importantes organizaciones que son la Internet Research Task Force (IRTF) y la Internet Engineering Task Force (IETF),esta última es coordinada por la Internet Steering Group (IESG).figura 4

IRTF se encarga de los proyectos de investigación de largo plazo, mientras que IETF se enfoca a los proyectos de necesidades inmediatas. Ambas utilizan una metodología iterativa de diseño, implementación, experimentación y revisión.

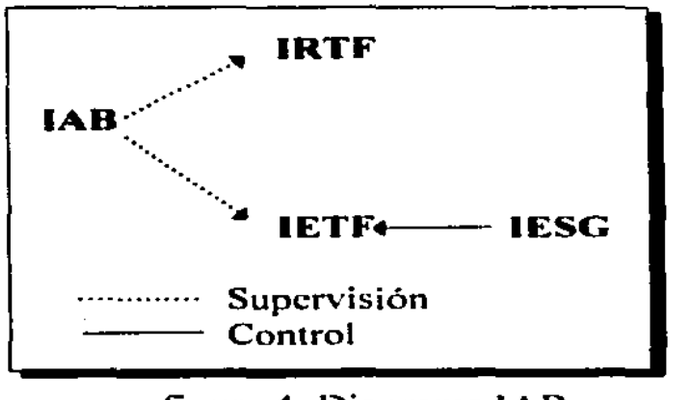


figura 4: Diagrama IAB

En 1991 la IAB es reestructurada y pasa a formar parte de la recién fundada Sociedad Internet (ISOC).

Entre las responsabilidades de la IAB tenemos las siguientes:

- Seleccionar a los miembros de IESG y IETF de una lista propuesta por el comité de nominación de IETF.
- Revisar y vigilar la arquitectura de los protocolos utilizados en Internet.
- Una vez que un protocolo ha sido probado y puede ser implementado, IAB se encarga de circular un documento llamado RFC (Request For Comments), copias del mismo son colocadas en servidores de acceso público en todo el mundo. Su distribución y uso es totalmente libre y gratuita. Los RFCs no solamente abordan información sobre protocolos, sino también, otros tópicos relacionados con Internet y la comunidad que la integra.
- Administrar los números asignados a Internet mediante un organismo dependiente llamado IANA (Internet Assigned Numbers Authority).
- Actuar como el representante de los intereses de la Sociedad Internet ante otras organizaciones en lo concerniente a estándares y otros asuntos técnicos y organizacionales referentes a Internet.
- Asesorar y guiar a la Sociedad Internet y cuerpos locales en lo concerniente a aspectos técnicos, estándares, arquitectura y procedimientos referentes a Internet y sus tecnologías.

1.2.2 InterNIC

InterNIC fue establecida en enero de 1993 como un proyecto entre AT & T, General Atomics y Network Solutions, Inc. con el apoyo de la Fundación Nacional de Ciencias de los Estados Unidos (NSF). General Atomics se separa del proyecto en Febrero de 1995.

Los servicios ofrecidos por InterNIC se dividen en dos grandes rubros:

- Servicios de bases de datos en línea (Info Scout).
- La asignación de direcciones IP y registro de dominios para las diferentes redes. Esta función era realizada anteriormente por el Centro de Información de la Red perteneciente a la red del departamento de defensa de los Estados Unidos. (DDN NIC).

1.2.3 Internet Society

Su creación fue anunciada en una conferencia de interconectividad en el año de 1991 en Copenhague, comenzando sus operaciones formalmente en enero de 1992. La Sociedad Internet (Internet Society) es una organización internacional para la cooperación y la coordinación de Internet, sus tecnologías y aplicaciones[4]. Está formada por miembros de la comunidad de Internet entre los que se encuentran individuos, corporaciones, organizaciones no lucrativas y agencias gubernamentales.

El propósito principal de ISOC (Internet Society) es mantener y extender el desarrollo así como la disponibilidad de Internet, sus tecnologías y aplicaciones.

Entre sus consignas se encuentran:

- Desarrollo, mantenimiento, evolución y diseminación de estándares de Internet , sus tecnologías y aplicaciones.
- Crecimiento y evolución de la arquitectura de Internet.
- Propuesta de procesos administrativos necesarios para la operación de Internet e Intranets.
- Educación e investigación relacionada a Internet e interconectividad.
- Recopilación y difusión de información relacionada con Internet e interconectividad.

La Sociedad Internet de México A.C. es el capítulo en México de la Internet Society (ISOC).La ISOCMex se funda en 1996 como una iniciativa de un grupo de personas interesadas (Inicialmente en su mayoría universitarios) en contar con un foro nacional sobre Internet donde los usuarios y todo tipo de proveedores de servicios o infraestructura del Internet puedan reunirse y expresar sus ideas respecto

[4] http://www.isoc.org/whatis/what-is-isoc.html 12/09/97

al desarrollo, servicios, seguridad y alcance de esta red que ya alcanza todos los Estados de la República Mexicana.[5]

1.3 Reseña del desarrollo de UNIX.

Un sistema operativo es un programa que maneja los recursos de la computadora, brindando una interfaz entre las instrucciones que pueden ser manejadas por el usuario y las que son entendidas por la computadora, permitiendo así controlar y dirigir la operación de la misma.

El sistema operativo Unix e Internet comparten el mismo momento histórico, en muchas ocasiones los mismos centros de desarrollo y hasta los mismos patrocinadores. Es un hecho conocido que el Departamento de Defensa de los Estados Unidos proporcionó muchos de los fondos necesarios no solamente para el desarrollo de Internet (en aquel entonces ARPANET), sino también para la investigación en UNIX. Gran parte del desarrollo de Unix se llevo acabo en la Universidad de California, siendo esta la que tuvo el primer nodo de ARPANET en la UCLA. Tanto la primera versión de UNIX como ARPANET hacen su aparición en 1969 y ambas buscaban la interconectividad. El hecho de que los servicios de TCP/IP estén implementados como parte natural de UNIX, dan mucha fuerza tanto a UNIX como a Internet.

Al igual que ARPANET la primera versión de UNIX aparece en el año de 1969, desarrollada por Ken Thompson en Bell Laboratories en Murray Hill, New Jersey. La primera versión de UNIX llamada unics, se ejecutaba en una computadora Digital Equipment PDP-7. En 1970, junto con Denis Ritchie, Thompson lo transportó a una PDP-11/20. Ritchie diseño y escribió además el primer compilador de C con el objeto de ofrecer un lenguaje que pudiera usarse para escribir una versión transportable del sistema.

En 1974 aparece una versión llamada quinta edición y se entrega a las universidades con fines educativos (punto clave para su difusión). La sexta edición conocida como v6 fue liberada en 1976. En 1978 se libera la séptima edición la cual junto con la sexta dieron lugar a varios caminos en el desarrollo de UNIX.

Las dos principales ramificaciones de UNIX son las versiones comerciales desarrolladas por AT & T conocidas como sistema V y los diversos sistemas BSD (Berkeley Software Distribution). El trabajo de BSD fue apoyado en gran parte por

[5] ftp://udgftp.cencar.udg.mx/pub/incoming/ISOC 12/09/97

contribuciones de DARPA. En 1983 DARPA funda la comisión Bolt, Benarek y Newman (BBN) para implementar TCP/IP en UNIX BSD, comenzando así la unión UNIX-TCP/IP.

TCP/IP se constituye como parte de UNIX a partir de las versiones 4.2 de BSD y del Sistema V de AT & T dándose este cambio en 1983. Las versiones subsecuentes de BSD y Sistema V han tenido una gran influencia mutua a través de los años. (fig. 6)

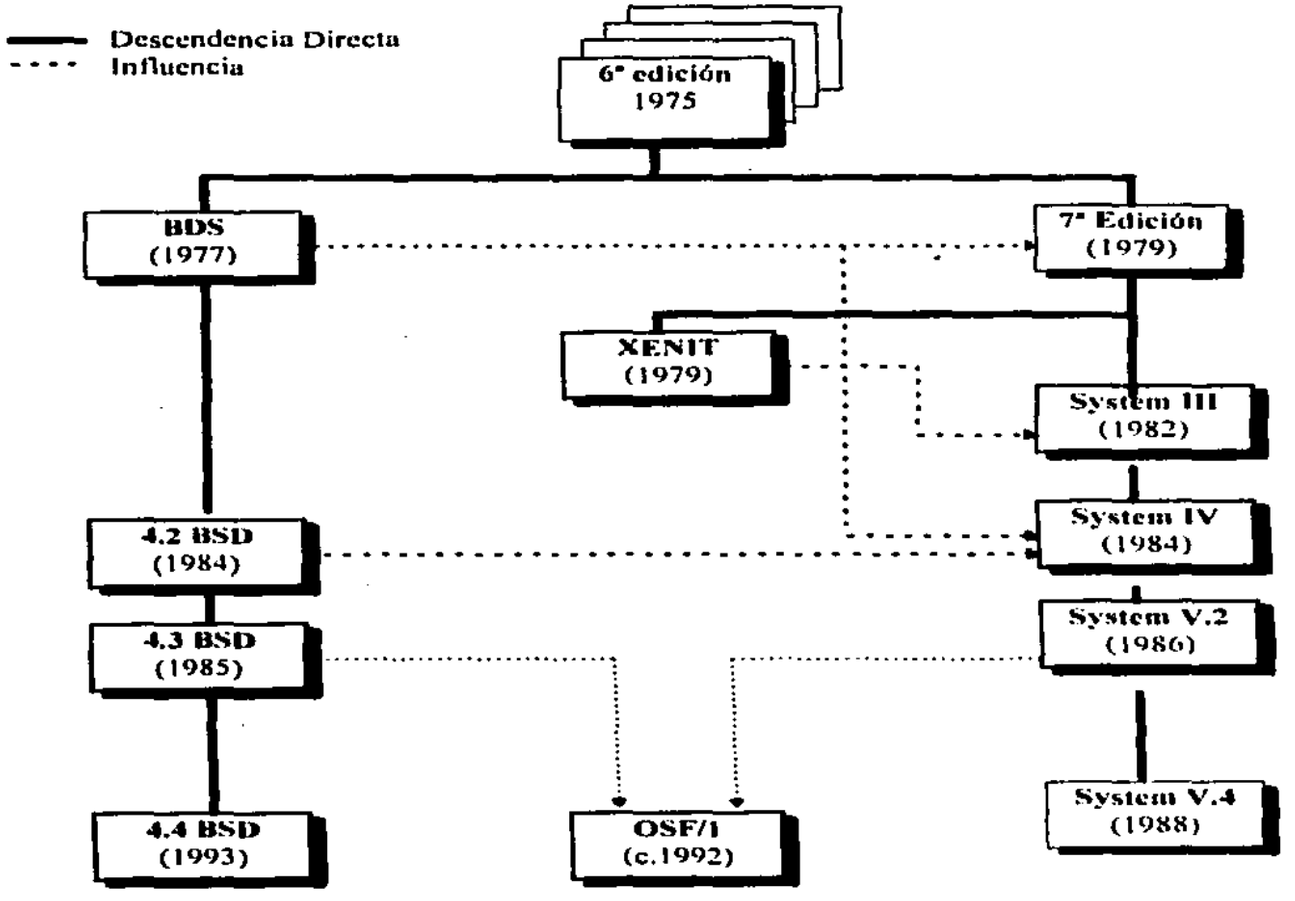


figura 6. Evolución de UNIX

Diversas compañías de hardware han desarrollado sus propias implementaciones de UNIX basándose en BSD , Sistema V o ambas. (fig. 7)

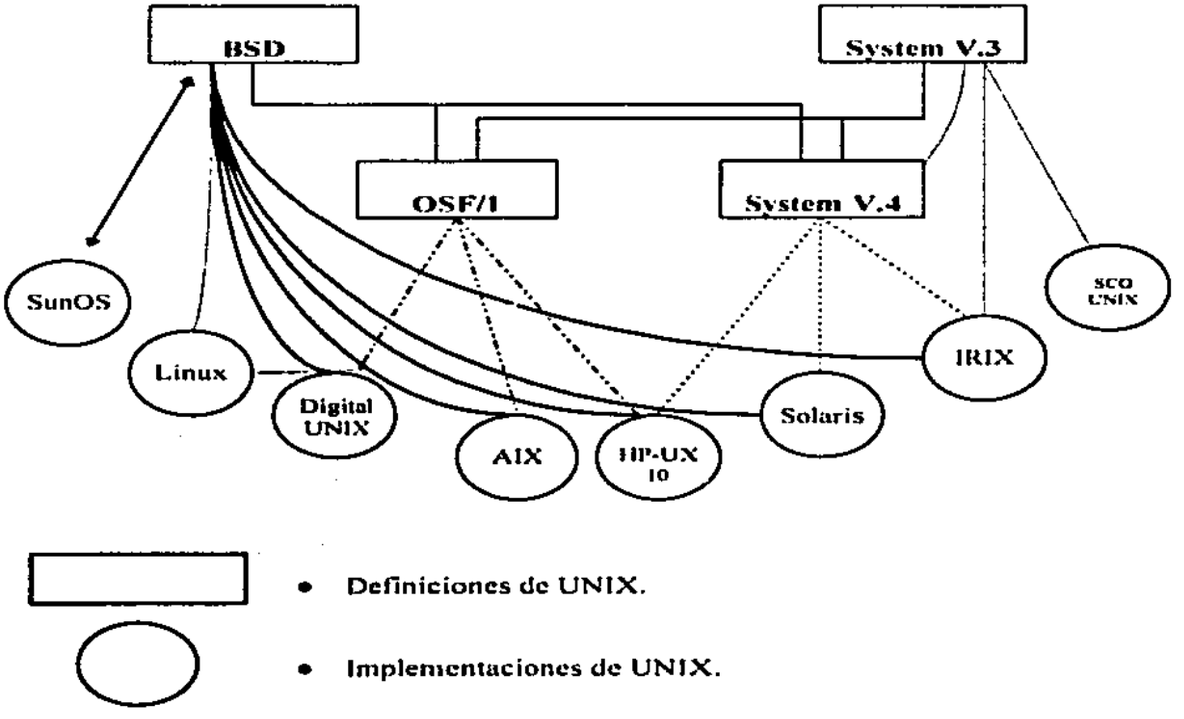


figura 7. Implementaciones de UNIX.

La presente investigación se desarrolla sobre la implementación de UNIX Solaris. Solaris es una de las versiones de UNIX de SUN Microsystems empresa fundada en el año de 1982 por Bill Joy (creador del programa vi de UNIX BSD). Solaris está basado principalmente en el Sistema V versión 4.

Capítulo 2. Redes, Protocolos y TCP/IP

Capítulo 2. Redes, Protocolos y TCP/IP

2.1 Nociones generales sobre redes.

2.1.1 Definición

"Una red de área local (LAN), es una combinación de dos o más computadoras que están física y lógicamente conectadas entre sí. Las redes de área local pueden estar interconectadas con otras redes en alguna otra localización, esto se conoce como red de área amplia (WAN), y se realiza usualmente a través de líneas públicas".[6]

2.1.2 Topologías

A la forma en que se interconectan los nodos de una red se le conoce como topología física, y al modo en que interactuan se le denomina topología lógica. Entre las principales topologías se encuentra:

2.1.2.1 Topología de Bus

En esta topología los equipos se conectan mediante un cable que actúa como camino de la información (bus). La principal ventaja de esta topología es su sencillez y bajo costo. La desventaja es que si ocurre una ruptura en el bus se inhabilita toda la red, además de que las redes de bus son más propensas a colisiones. Una colisión se produce cuando 2 nodos tratan de transmitir al mismo tiempo, dando como resultado que la información se tenga que retransmitir. (fig. 8)

---

[6]Norton, Peter. Periféricos y accesorios para la IBM, PC,PS/2 y compatibles. México: Prentice Hall. 1994. p.208.

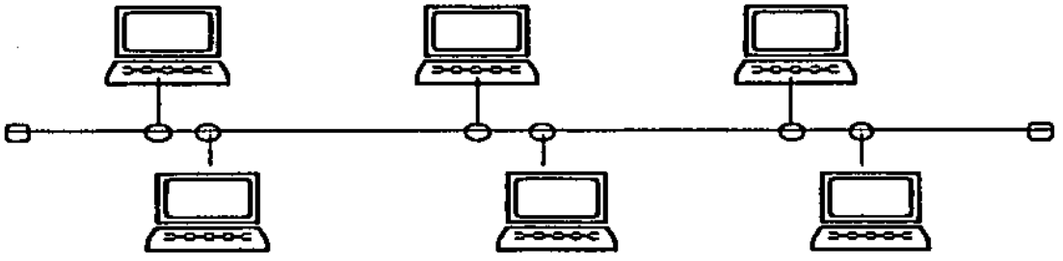


figura 8. Topología de Bus

2.1.2.2 Topología de Anillo

En la topología de anillo las estaciones de trabajo se interconectan entre si formando un circulo, es decir la última estación de la línea se conecta con la primera. Bajo esta topología cada nodo espera un turno para transmitir, por lo tanto el riesgo de colisión es prácticamente nulo. Su desventaja es que en caso de que un nodo no funcionara correctamente o se rompiera la conexión, toda la red quedaría inhabilitada. (fig. 9)

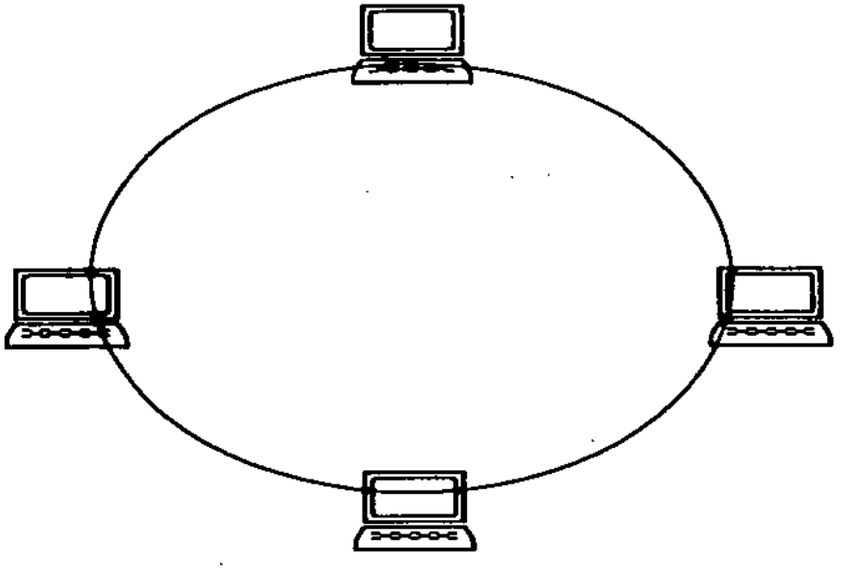


figura 9. Topología de Anillo.

2.1.2.3 Topología de Estrella

Todos los nodos en esta topología están conectados a un núcleo central que se encarga de distribuir la información. En caso de que se rompiera alguna de las líneas que conectan a un nodo con el núcleo, solamente se afectaría al nodo en cuestión. Su inconvenientes son el costo de los equipos centrales y que requiere mayor cantidad de cable. (fig. 10)

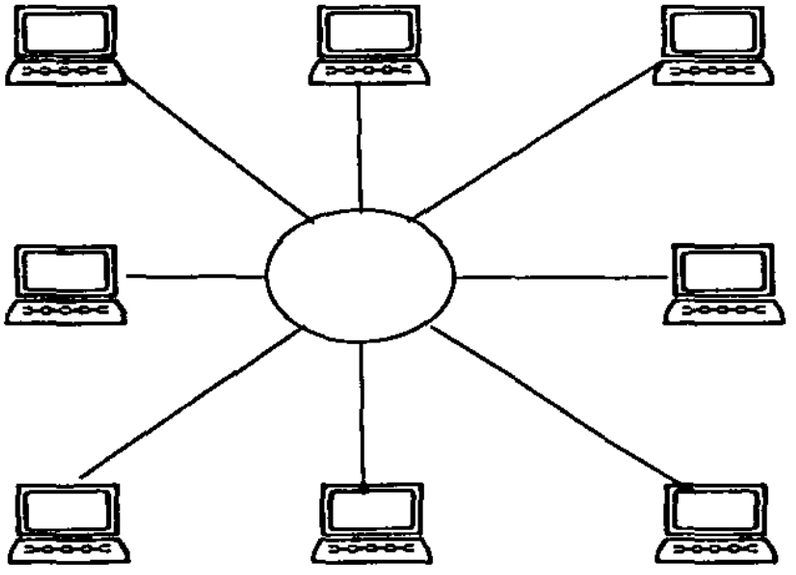


figura 10. Topología de Estrella

2.1.3 Componentes

Los componentes que integran una red son:

- Equipos o computadoras.
- Medios físicos de conexión
- Tarjetas de red
- Otros dispositivos

2.1.3.1 Equipos o computadoras

Estos equipos van desde supercomputadoras, minicomputadoras, equipos personales, hasta terminales sin capacidad de procesamiento llamadas terminales tontas.

2.1.3.2 Medios físicos de conexión

El medio físico más común lo constituyen los cables entre los que destacan:

- Cable Ethernet Grueso (10Base5)
  Consiste en un cable coaxial de 0.4 pulgadas de grosor. Cada extremo del cable termina con un conector tipo D. Su desventaja es su peso, volumen y la dificultad de doblado en las esquinas. Es capaz de transportar señales hasta 500 metros (2,500 pies). El alcance puede ser extendido añadiendo repetidores a lo largo del mismo. Maneja una resistencia de 50 ohms.

- Cable Ethernet Delgado (10Base2)
  Tiene un grosor de 0.2 pulgadas, es más flexible y ligero que el Ethernet Grueso. En cada extremo del cable tiene un conector BNC. Tiene un alcance de 185 metros por segmento.

- Cable par trenzado (10BaseT)
  Consiste de una serie de pares de cables trenzados entre sí (de 2 a 4) recubiertos por un forro. Es similar al cable telefónico. Existen 2 variedades, el cable de par trenzado sin blindaje (UTP) y el cable de par trenzado con blindaje (STP). El más común es UTP y puede transmitir en segmentos de hasta 100 metros, el cable STP llega a alcanzar distancias mayores.

- Cable de fibras ópticas (FDDI)
  Este cable permite transferir información a velocidades de hasta 100 Mbps. A diferencia de los anteriores es inmune a interferencias eléctricas y magnéticas. Permite segmentos de hasta 2 Km. Sus desventajas son el alto costo, la dificultad para doblarse en las esquinas y la precisión requerida en los cortes, ya que es muy difícil realizar enlaces entre segmentos de cable.

2.1.3.3 Tarjetas de red (nic: Network Inferface Cards)

Permiten la conexión al medio de transmisión de datos, existen varios tipos de acuerdo a diferentes medios y estándares como son:

- Ethernet
  Maneja redes con topología físicas de bus o estrella y lógicas de bus. La velocidad de transmisión de datos es de 10 Mbps.

  Ethernet utiliza un método de transmisión denominado CSMA/CD (Carrier Sense Multiple Acces /Collision Detection). Mediante CSMA/CD cada nodo que requiere transmitir datos verifica el medio físico para determinar si alguna información está siendo transmitida, en caso de que la línea esté libre el nodo transmitirá la información. Cuando dos o más nodos intentan transmitir al mismo tiempo se produce una colisión, y cada uno de los nodos esperará un tiempo aleatorio para reintentar la transmisión. En el caso de colisiones recurrentes el tiempo de espera se aumenta al doble en cada intento. Entre más nodos tenga una red Ethernet, más propensa a colisiones se encontrará.
  Dentro de Ethernet existen 3 estándares: 10Base5, 10Base2 y 10BaseT.

- ARCNET
  Esta topología funciona desde 1977. Trabaja en una topología lógica de bus y física de estrella; tiene una velocidad de transmisión de 2.5 Mbps. Sus ventajas son su relativamente bajo precio y facilidad de instalación.

- Token Ring
  Utilizada para redes IBM, maneja una topología lógica de anillo y física de estrella; el envío de la información se basa en paquetes que son transmitidos de nodo a nodo a través del anillo. A estos paquetes se les llama fichas (tokens). Las velocidades de transferencia de datos son de 4 Mbps y 16 Mbps.

2.1.3.4 Otros dispositivos

En una red pueden figurar uno, algunos o ninguno de los siguientes componentes, dependiendo de su tamaño, alcance y tipo:

- *Ruteadores (Routers)*. Dispositivos que operan en la capa de red, permiten interconectar redes que utilizan el mismo protocolo de red (por ejemplo IP), aunque tengan topologías distintas. Dirigen y encaminan la

información entre redes basándose en algoritmos para encontrar rutas óptimas.

- *Compuertas (Gateways)*. Permiten conectar redes que manejan protocolos distintos realizando conversiones entre protocolos. En algunos textos el termino gateway y router se usa de manera indistinta.

- *Puentes (Bridges)*. Conectan dos redes que utilizan el mismo protocolo de acceso al medio. Los puentes tienen como principal función disminuir el tráfico al examinar las direcciones físicas hacia donde se envían los datos que llegan al puente, filtrando el paso a través de él.

- *Repetidores*. Al ser enviada una señal por un medio físico, esta se atenúa o se debilita conforme la distancia aumenta. Un repetidor es un dispositivo que toma una señal, elimina el ruido, la amplifica y la retransmite.

- *Concentradores o hubs*. Dispositivos utilizados para conectar varios nodos de una red vía una caja central o núcleo (topología de estrella).

2.2 Protocolos, stacks y suites de protocolos.

Un *protocolo* es un conjunto de normas que regulan una función de comunicación. Podemos mencionar como ejemplos al protocolo IP, este consiste en una serie de reglas para dirigir los datos a través de la red; otro ejemplo es TCP que se encarga de que la transmisión de datos se realice de manera confiable.

Una pila o *stack* es un conjunto de protocolos individuales que trabajan juntos de manera estratificada. Por ejemplo TCP, IP y Ethernet constituyen una pila de protocolos.

Un juego o *suite* de protocolos es la forma de agrupar o hacer referencia a un grupo de protocolos representándolos como uno solo. En el caso de TCP/IP, este nombre se refiere a la serie de protocolos de comunicación dentro de Internet, TCP e IP son solo dos de estos protocolos, pero dan el nombre a la suite debido a su importancia.

Para efectos del presente trabajo se hará referencia a TCP/IP como todo el conjunto de protocolos de Internet y a TCP e IP separadamente como protocolos individuales.

2.3 Arquitectura de TCP/IP

La arquitectura de TCP/IP se encuentra dividida en cuatro capas. La estratificación se establece en base a la forma en que los datos originados por una aplicación son procesados y enviados a través de la red, así como el proceso necesario para su recepción y recuperación. Al pasar los datos por cada capa esta agrega información de control a los datos para asegurar su adecuada recepción. A la información agregada se le llama *encabezado*. Como ya se menciono cada una de las capas agrega un encabezado a la información que recibe de la capa inmediata superior y la entrega a la capa inmediata inferior. Al proceso de colocar encabezados se le llama *encapsulamiento* de los datos. Cuando los datos encapsulados llegan al equipo receptor se realiza el proceso inverso eliminando encabezados y transmitiendo el remanente a la capa inmediata superior.

Las capas que constituyen a TCP/IP son:

CAPA DE APLICACIÓN: Consiste de aplicaciones y procesos que utiliza la red.
CAPA DE TRANSORTE: Provee servicios de entrega de datos.
CAPA DE INTERNET: Define datagramas y maneja el ruteo de los datos.
CAPA DE ACCESO A LA RED. Consiste de rutinas para accesar los dispositivos físicos de la red.

Existen dos formas de transferencia a través de TCP/IP, utilizando el protocolo TCP ó UDP en la capa de transporte. Mediante el uso de TCP se realiza una verificación de la integridad de los datos, y con el segundo simplemente se envían sin verificar su correcta recepción, la ventaja de este último es mayor velocidad.

Utilizando TCP una aplicación simplemente manda un flujo de datos (*stream*) a TCP, este se encarga de dividirlo en porciones y agregarle un encabezado dirigido a la capa de TCP de equipo receptor, a cada una de las porciones de datos son llamas *segmentos;* TCP transfiere cada uno de estos segmentos a la capa de Internet. En esta fase se les coloca otro encabezado formando *datagramas*, los datagramas son enviados a la capa de acceso a la red. Finalmente se le agrega un encabezado más

formando *frames* los cuales son transmitidos por la red hasta la computadora destino realizando esta todo el proceso de manera inversa desde la capa de acceso a la red , concluyendo con la entrega de los datos ya revisados y en orden a la aplicación receptora.

Mediante UDP el proceso es diferente en las dos capas más altas, una aplicación manda los datos en forma de secciones llamadas *mensajes* a UDP. UDP le agrega un encabezado a cada mensaje formando *paquetes*, los paquetes se transfieren a la capa de Internet la cual le agrega un encabezado y forma *datagramas* , estos se envían a la capa de acceso a la red. El proceso restante es igual al empleado utilizando TCP. Cabe resaltar que mediante este método la responsabilidad de la integridad de los datos recae sobre las aplicaciones implicadas (capa de aplicación) y no sobre la capa de transporte(fig. 11).

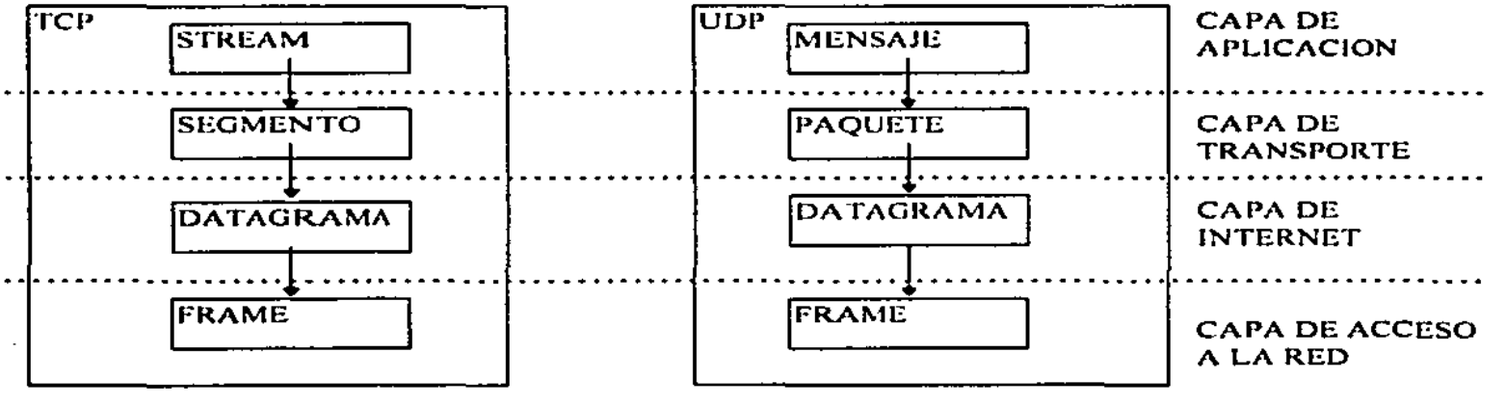


figura 11. Capas de TCP/IP.

A continuación se detalla cada una de las capas:

2.3.1 Capa de acceso a la red.

Los protocolos que integran esta capa proveen los medios para transmitir datos a otros dispositivos u otras redes interconectadas de manera directa o indirecta con la computadora origen.

A diferencia de capas de nivel más alto, la capa de acceso a la red tiene que tratar con aspectos físicos de la red para poder realizar la transferencia a través del medio físico. Es así que cada vez que una nueva tecnología de hardware de comunicaciones aparece es necesario desarrollar nuevos protocolos de acceso a la red para que redes con TCP/IP puedan utilizar estos nuevos dispositivos.

Entre las funciones que se encuentran en este nivel están las de encapsulación de datagramas en frames y la conversión de direcciones IP en direcciones físicas utilizadas por los dispositivos de la red.

2.3.2 Capa de Internet

El principal protocolo no solamente de la capa de Internet, sino de todo TCP/IP es IP. Todos los protocolos en las capas por debajo y arriba de IP, utilizan IP para enviar y recibir datos, es decir todos los datos que son enviados o recibidos tienen que pasar por IP.

IP es un protocolo no orientado a conexión, sin detección y recuperación de errores. La conexión y detección de errores son manejadas por otras capas de TCP/IP.

Dentro de las funciones de IP tenemos las siguientes :

2.3.2.1 Definición de datagramas

Los protocolos de Internet fueron diseñados para trabajar sobre ARPANET, dicha red trabajaba por conmutación de paquetes. IP funciona bajo este principio, es decir, la información para ser enviada se divide en paquetes, cada uno de ellos contiene información de su origen y su destino viajando de manera independiente a través de la red.

El formato de paquetes definido en IP se llama datagrama. Las primeras cinco o seis palabras (tomando como palabra 32 bits consecutivos) de un datagrama constituyen información de control llamada encabezado (header). Por omisión un encabezado mide 5 palabras siendo la sexta palabra opcional. Debido a esta característica el encabezado contiene un campo llamado Internet Header Length (IHL) que indica el tamaño del encabezado. El encabezado contiene la información necesaria para la entrega del datagrama.

2.3.2.2 Ruteo de Datagramas

IP entrega el datagrama revisando la dirección destino en la quinta palabra del encabezado, si la dirección es un host de la red local, lo entrega directamente. Cuando un datagrama se envía fuera de la red local, IP lo entrega al routeador local, si el routeador local no tiene información sobre la red destino el datagrama es enviado a otro ruteador y así hasta llegar a su destino.

Cada routeador contiene una tabla de ruteo donde se especifican los host que pertenecen a su red local, así como otros ruteadores disponibles. En redes pequeñas la tabla se puede actualizar manualmente, pero en redes mayores los ruteadores pueden actualizar sus tablas intercambiando datos entre ellos.

2.3.2.3 Fragmentación de Datagramas

Cuando se conectan redes heterogéneas algunas veces es necesario dividir un datagrama en piezas más pequeñas. Un datagrama recibido de una red puede ser demasiado largo para ser retransmitido en otra. Cada tipo de red tiene una unidad máxima de transmisión (MTU), la cual indica el tamaño del paquete más grande que puede ser transferido. Si un datagrama recibido es mayor que la MTU de la red receptora es necesario dividir el datagrama, a este proceso se le llama *fragmentación*. El datagrama resultante contiene en su encabezado la información necesaria para ser ensamblado nuevamente.

2.3.2.4 Paso de datagramas a la capa de transporte

Cuando IP recibe un datagrama dirigido al host local, debe pasar la porción de datos del datagrama al protocolo correcto de la capa de transporte. Esto lo realiza en base al campo número de protocolo de la palabra 3 del encabezado del datagrama.

2.3.3 Capa de transporte

Los dos principales protocolos dentro de este estrato son TCP (Transmission Control Protocol) y UDP (User Datagram Protocol) . Ambos transfieren datos entre las capas de Aplicación e Internet. La entrega de datos a la aplicación correcta se realiza mediante el número de puerto destino. Es decisión de los programadores de las aplicaciones decidir que utilizar UDP ó TCP.

2.3.3.1 UDP

Este protocolo se encarga de la entrega de datagramas, en un modo sin conexión, control ni recuperación de errores. Es más sencillo que TCP y más rápido, ya que la cantidad de datos a transmitir es menor debido a que cada paquete tiene un encabezado que mide solo 64 bits (2 palabras).

En dicho encabezado solamente se indican los puertos de origen y destino, la cantidad de datos de el paquete y un campo de verificación. (fig. 12)

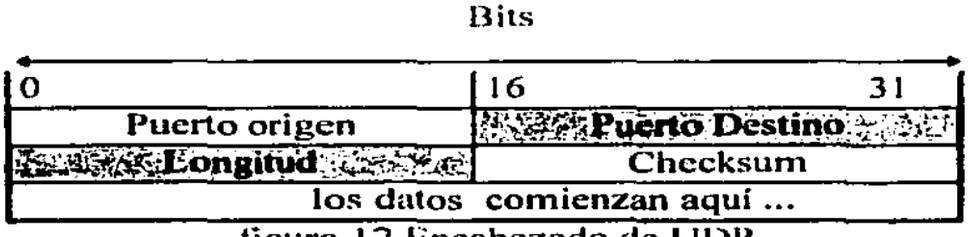


figura 12.Encabezado de UDP.

Este protocolo es ideal para aplicaciones de tipo pregunta-respuesta , donde una pregunta o una respuesta se puede interpretar como una recepción correcta de datos.

2.3.3.2 TCP

Es más complejo que UDP, provee un servicio de entrega de datos con detección y corrección de errores. Se caracteriza por ser más confiable que UDP y por que a diferencia de este último es un protocolo con conexión (fig. 13).

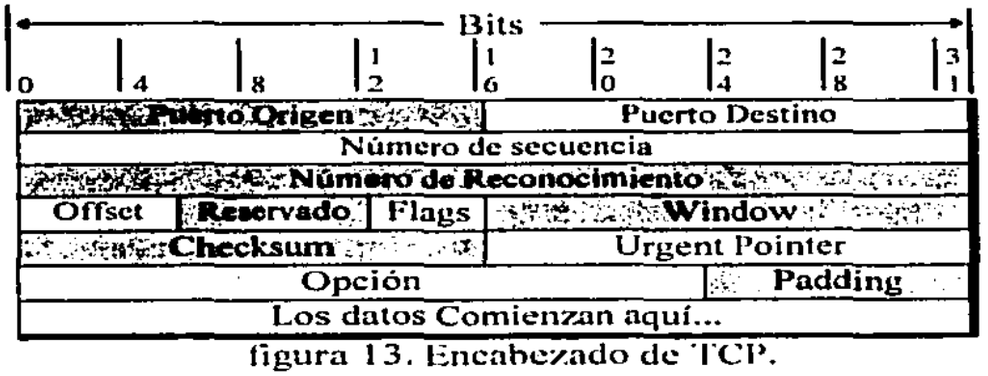


figura 13. Encabezado de TCP.

La confiabilidad de TCP se logra mediante un mecanismo llamado reconocimiento positivo de re-transmisión . Dicho mecanismo consiste en que a través de TCP el host emisor reenviara un segmento de datos previamente enviado a menos que reciba una señal de recepción correcta por parte del host receptor. Cada segmento transmitido, contiene un dato de verificación que el receptor utiliza para asegurarse de la integridad del segmento. En caso de que el segmento sea recibido en buen estado, el host destino transmite un reconocimiento al emisor, de lo contrario el segmento es desechado. Después de un periodo de tiempo preestablecido el emisor reenviará cualquier segmento del cual no haya recibido un reconocimiento de segmento. Es responsabilidad de TCP acomodar los segmentos de manera secuencial y descartar los segmentos dañados o duplicados antes de que el flujo de datos sea entregado a la aplicación.

La información contendida en el reconocimiento de segmento (ACK), además de permitir verificar la integridad, número y orden de los segmentos permite establecer el control de flujo, ya que en un ACK no solamente permite saber que información se ha recibido correctamente, sino que también cuantos datos más pueden recibirse en el host destino, evitando con esto congestionamientos.

El que TCP sea un protocolo con conexión se refiere a que antes de comenzar a transmitir segmentos al destinatario, verifica la posibilidad de la conexión por medio del intercambio de información de control con el host destino (handshake). El handshake realizado por TCP se denomina de tres vías ya que tres segmentos son intercambiados antes y al final de una transmisión de datos de la capa de aplicación de la siguiente manera:

1. El host origen envía un segmento con datos de sincronización (SYN) al destino, mediante los cuales le informa que requiere hacer una conexión para transferencia así como el número para el primer segmento de datos, cada paquete subsecuente contendrá un número secuencial que posteriormente le permite al host destino reensamblar los datos.

2. El host destino responde al host origen con un segmento que contiene la aceptación y sus números de sincronización

3. Finalmente el host origen envía un segmento de reconocimiento de la transmisión del host destino y transfiere el primer segmento con datos de la capa de aplicación.

Después de este proceso, los datos son enviados, al concluir la transmisión de datos se realiza nuevamente un handshake de tres vías con segmentos indicando que no hay más datos por transmitir, cerrando así la transmisión.

2.3.4 Capa de aplicación

El la capa más alta de TCP/IP. Dentro de este estrato existen muchos protocolos, la mayoría de ellos proveen servicios al usuario.

Entre los más comunes tenemos:

Telnet. Acceso remoto.
FTP. Transferencia de archivos.
SMTP. Correo electrónico.
DNS. Conversión de direcciones IP por dominio a numéricas.
RIP. Servicios de ruteo de información
NFS. Uso compartido de archivos.

Algunos de estos protocolos se ejecutan a voluntad del usuario como es el caso de FTP y Telnet, pero otros como RIP se llevan acabo aun sin que el usuario conozca su existencia.

Capítulo 3. Transmisión de datos y servicios en TCP/IP

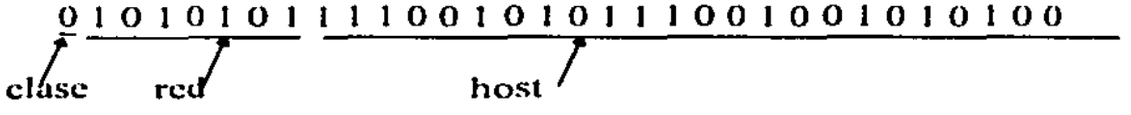
Capítulo 3. Transmisión de datos y servicios en TCP/IP

3.1 Direcciones IP numéricas y por nombre.

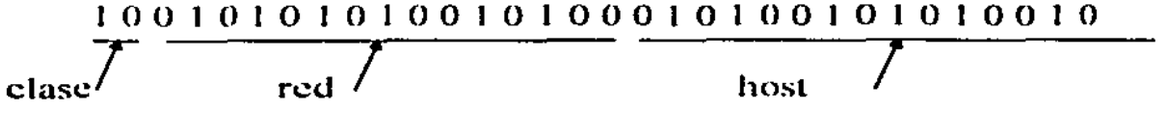
IP intercambia datos entre hosts en forma de datagramas. Cada datagrama es entregado a la dirección contenida en el campo de dirección destino del encabezado del datagrama (palabra 5). La dirección destino es una dirección IP estándar de 32 bits, la cual tiene la información suficiente para identificar de manera única una red y un host dentro de la misma.

Una dirección IP esta compuesta de 2 partes, la parte que indica la red y la que hace mención al host dentro de esta red. El número de bits utilizados para identificar una red o un host varia dependiendo de la clase de la red. Las clases de red son A, B y C. Examinando los primeros bits de una dirección IP se pude determinar su clase de red. Los criterios para la determinación de la clase son los siguientes:

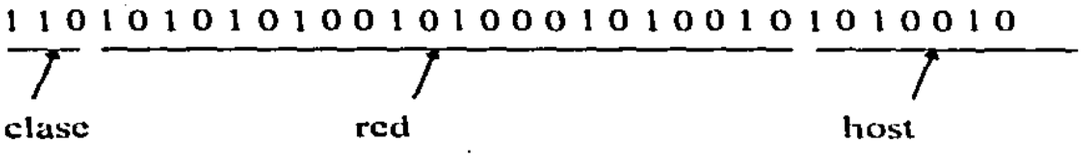
Redes Clase A: El primer bit de estas direcciones es 0, los siguientes 7 bits identifican a la red y los 24 restantes al host. Esto nos da la posibilidad de menos de 128 redes clase A en todo Internet, pero cada una de estas puede contener millones de hosts, ej.



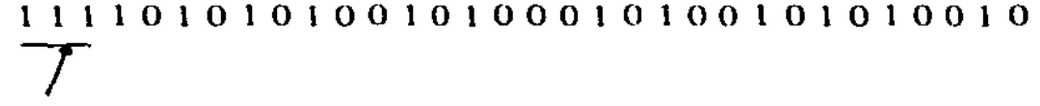
Redes Clase B: Los primeros 2 bits de ese tipo de red son 1 0. Tenemos entonces que los primeros 2 bits identifican la clase de la red, los siguientes catorce bits pertenecen a la red y los 16 restantes identifican al host. Pueden existir miles de redes clase B conteniendo miles de hosts. ej.



Redes Clase C: Los primeros 3 bits de las direcciones de esta clasificación son 110. Como se observa los primeros 3 bits determinan el tipo de red, los siguientes 21 la red y los 8 restantes identifican el host. Pueden existir millones de redes clase C pero cada una de estas tiene menos de 254 host.



Direcciones Clase D: Los primeros 3 bits de estas direcciones son 111, dichas direcciones no son utilizadas para referir a redes especificas ya que son direcciones reservadas. Las direcciones asignadas dentro de esta clasificación son utilizadas como direcciones multicast y para usos experimentales. Las direcciones multicast son usadas para direccionar grupos de computadoras al mismo tiempo e identifican a un grupo de computadoras que comparten un protocolo común, pero que no se encuentran dentro de la misma red.



El manejo de las direcciones mediante números binarios es muy complicado y se da solo a nivel de máquina. Generalmente las direcciones IP se representan con cuatro números decimales separados por puntos. Cada uno de los cuatro números se encuentra dentro de un rango de 0 a 255 (los valores decimales posibles para un byte). Tomando esta notación podemos identificar a las direcciones de la siguiente manera:

Direcciones clase A. El primer número es menor de 128 e indica el número de la red, los 3 números siguientes hacen referencia al host.

Direcciones clase B. El primer número esta en un rango de 128 a 191. Los primeros dos números indican la red y los dos restantes el host.

Direcciones clase C. El primer número esta en un rango de 192 a 223. Los primeros tres números indican la red y el restante el host.

Direcciones clase D. El primer número es mayor a 223, como ya se menciono estas son direcciones reservadas.

La siguiente figura muestra la notación decimal.

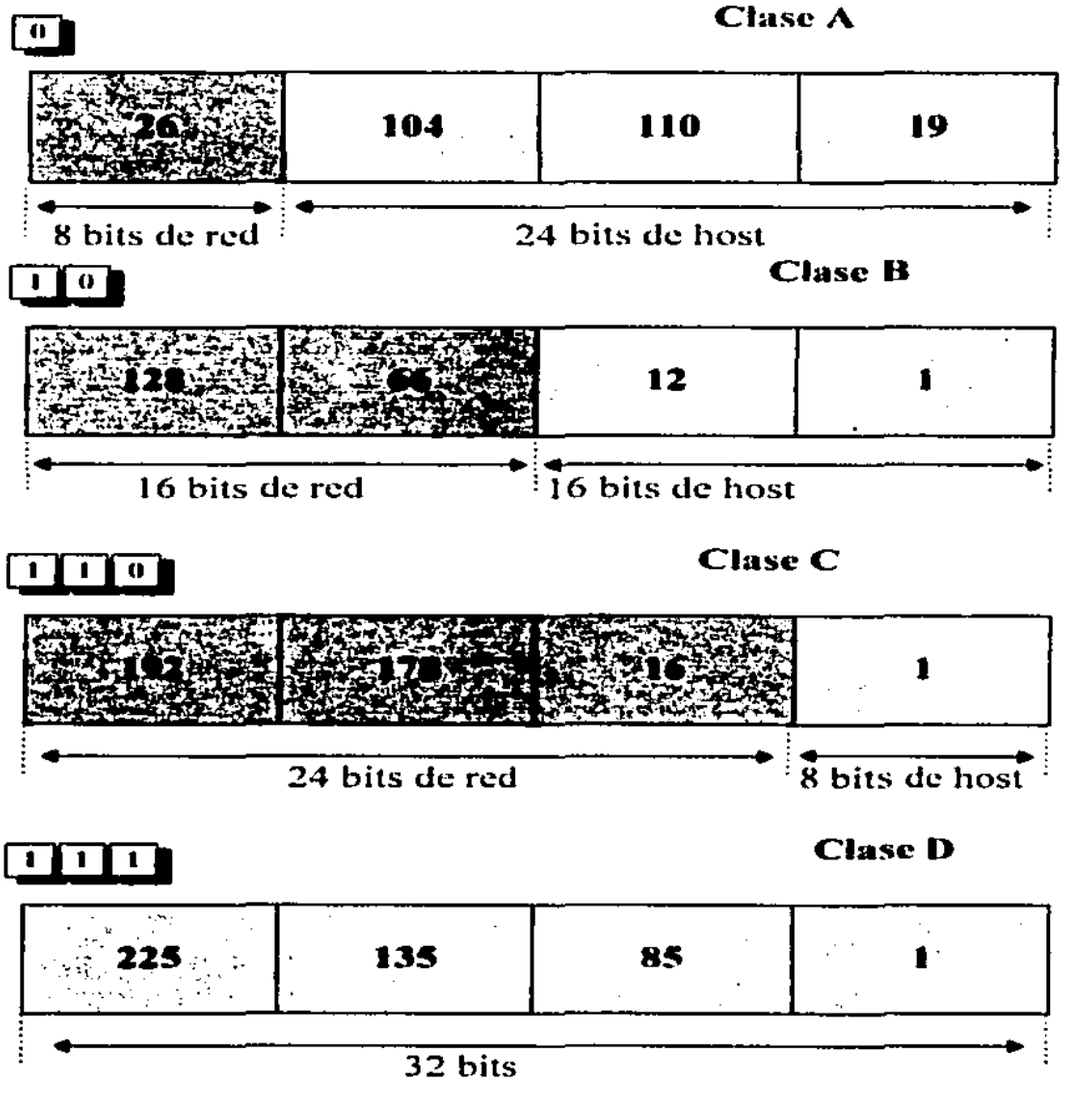


figura 14. Notación Decimal

Aun de esta manera es complicado manejar las direcciones de forma numérica. Por tal motivo se implemento la representación de las direcciones por nombre, los nombres son más fáciles de recordar.

Una dirección IP por nombre esta integrada por un nombre de host más una serie de campos alfanuméricos llamados dominios, separados entre sí por puntos. La

dirección IP de un host refleja la estructura jerárquica de la organización a la que pertenece. Los dominios de una dirección van de lo general a lo particular de derecha a izquierda.

En la mayoría de los casos los nombres y las direcciones numéricas pueden ser utilizadas indistintamente por ejemplo:

La dirección IP 132.248.10.1 tiene asignada la dirección por nombre redvax1.dgsca.unam.mx. en caso de querer utilizar el servicio telnet con este host se puede hacer de dos maneras.

telnet 132.248.10.1

telnet redvax1.dgsca.unam.mx

Una dirección por nombre puede contener desde 2 hasta 5 campos, cada uno con hasta 63 caracteres, teniendo como restricción que la dirección completa no debe exceder de 255 caracteres.

Se pueden tener por ejemplo las siguientes direcciones:

office.microsoft.com
bellcore.com
redvax1.dgsca.unam.mx
autos.produccion.nissan.com.mx
dgb1.cuautitlan2.unam.mx

La responsabilidad de asignar nombres de host o subdominios dentro de un dominio es delegada a una organización designada como administrador de dominio. El administrador de un dominio puede crear subdominos y delegar autoridad a alguna organización más dentro de cada dominio, y así sucesivamente. (fig. 15)

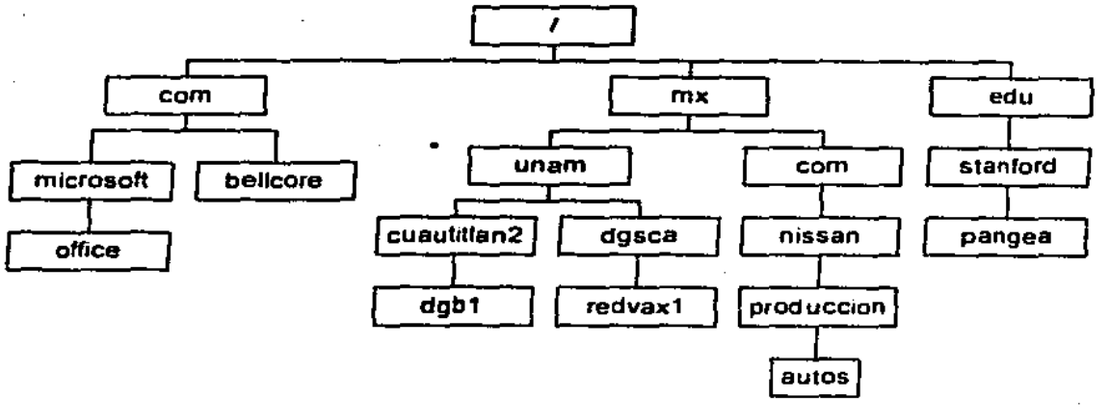


figura 15. Delegación de Dominios.

Los dominios de más alto nivel son administrados por InterNIC. Entre estos dominios encontramos:

Dominio	Designado para:
COM	Organizaciones comercial
EDU	Instituciones educativas
GOV	Cuerpos gubernamentales.
MIL	Organismos Militares
ORG	Organizaciones no lucrativas.
INT	Grupos Internacionales.

Existen también dominios por paises :

Dominio	Designado para :
aq	Antártida
ar	Argentina
at	Austria
au	Australia
be	Bélgica
bg	Bulgaria
ca	Canadá
ch	Suiza (Cantones Helvéticos)
cl	Chile
cn	China

cs	República Checa y Eslovaca
de	Alemania (Deutschland)
dk	Dinamarca (Denmark)
ec	Ecuador
ee	Estonia
eg	Egipto
es	España
fi	Finlandia
fr	Francia
gb	Gran Bretaña
gr	Grecia
hk	Hong Kong
hr	Croacia
hu	Hungría
ie	República de Irlanda
il	Israel
in	India
is	Islandia
it	Italia
jp	Japón
kr	Corea del Sur
kw	Kuwait
li	Liechtenstein
lt	Lituania
lu	Luxemburgo
lv	Latvia
mx	México
my	Malasia (Malaysia)
nl	Holanda (Netherlands)
no	Noruega
nz	Nueva Zelanda
pl	Polonia
pr	Puerto Rico
pt	Portugal
re	Reunión
se	Suecia
sg	Singapur
si	Eslovenia (Slovenia)
su	Unión Soviética (?)

th	Tailandia (Thailand)
tn	Túnez
tw	Taiwan
uk	Reino Unido (United Kingdom)
us	Estados Unidos (United States)
ve	Venezuela
yu	Yugoslavia
za	Sudáfrica

3.1.1 Máscaras de subred

Los tipos de redes definidos anteriormente (A, B y C), albergan a un gran número de hosts. Generalmente las redes clase A y B , están formadas a su vez por subredes.

Entre las razones para establecer subredes se encuentran:

- Interconexión de redes con topología física distinta mediante ruteadores IP. Por ejemplo redes Ethernet, Token Ring, FDDI.
- Disminución de tráfico, ya que el tráfico local permanece en la subred.
- Simplificación de la administración de la red, delegando funciones administrativas a diferentes organizaciones o individuos por cada subred o grupo de las mismas.
- Aislamiento y control de acceso, permitiendo el uso de ciertos recursos internos de la subred solamente a los hosts conectados directamente a ella.
- Aislamiento de problemas, detectando deficiencias a nivel de subred.

Para poder rutear los datos entre redes y subredes, se extiende la parte de red de una dirección IP, asignando un parte para identificar a la subred tomándola de la parte del host. (fig. 16)

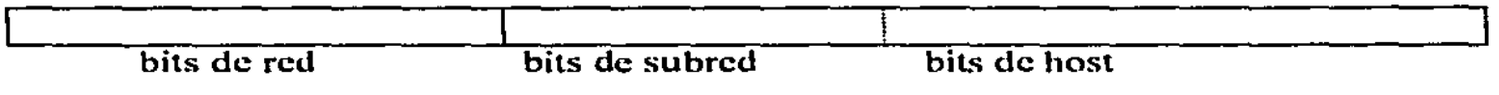


figura 16. Dirección IP de un host dentro de una subred

Por ejemplo si se tiene la siguiente dirección clase A.

26.104.10.19

La organización que tiene asignada la red a la que pertenece este host tendrá la posibilidad de establecer aproximadamente 16 millones de hosts. Este es un número muy grande para una sola red, y se decide tomar el segundo byte para definir un número de subred. De esta manera, se puede seccionar a la red clase A en cerca de 256 subredes, conteniendo unos 65 mil hosts en cada una de ellas.

Cada organización elige subdividir ó no su red, y el número de bits utilizados para tal efecto. Para subdividir una red, se utiliza el valor de configuración denominado *máscara de subred,* el cual consiste en una secuencia de 32 bits. Los bits de la mascara que corresponden a la parte de red y subred de la dirección IP tendrán un valor de uno.

Retomando el ejemplo anterior de una red clase A con 8 bits de subred, la mascara aplicada sería:

11111111111111110000000000000000

De igual forma que las direcciones IP, las máscaras se manejan mediante cuatro valores decimales separados por un punto, por lo tanto, la representación de la máscara anterior es:

255.255.0.0

Los ruteadores conectados a una red, deberán también ser configurados con la máscara de subred.

3.1.2 Direcciones especiales

3.1.2.1 Idetificación de redes y direcciones de difusión

La identificación de las redes y subredes, se hace mediante una dirección IP, en la cual se coloca el número de la red, la parte de la subred y se deja en ceros la parte del host.

Por ejemplo si se tiene a una red clase B con 8 bits de subred, su representación se hace colocando los valores de red en los primeros 2 octetos, el valor de subred en el tercero y el cuarto octeto se deja en cero.

132.248.0.0	Representación de la red clase B.
132.248.10.0	Representación de la subred 10 dentro de 132.248.0.0

De manera similar para representar a todos los hosts de una red o subred, los bits de host de una dirección se fijan en uno. A estas direcciones se les denomina direcciones de difusión (broadcasts) y se utilizan para enviar mensajes a todos los hosts de una red o subred. Retomando el ejemplo anterior las direcciones de difusión correspondientes son:

132.248.255.255	Todos los hosts de la red 132.248.0.0
132.248.10.255	Todos los hosts de la subred 132.248.10.0

Por ende los valores 0 y 255 no son permitidos en ningún octeto de la dirección IP de un host, reduciendo el número de direcciones posibles.

3.1.2.2 Dirección loopback

Todas las direcciones que comiencen con el valor 127 en el primer octeto, son direcciones reservadas y se utilizan para realizar pruebas en el software de red. La dirección *loopback* designada para pruebas internas dentro de un host es:

127.0.0.1

Esta media descarta 2^{24} direcciones que pudieran ser asignadas a hosts

3.2 Arquitectura de ruteo IP

Existen dos modelos básicos para la arquitectura de ruteo de Internet:

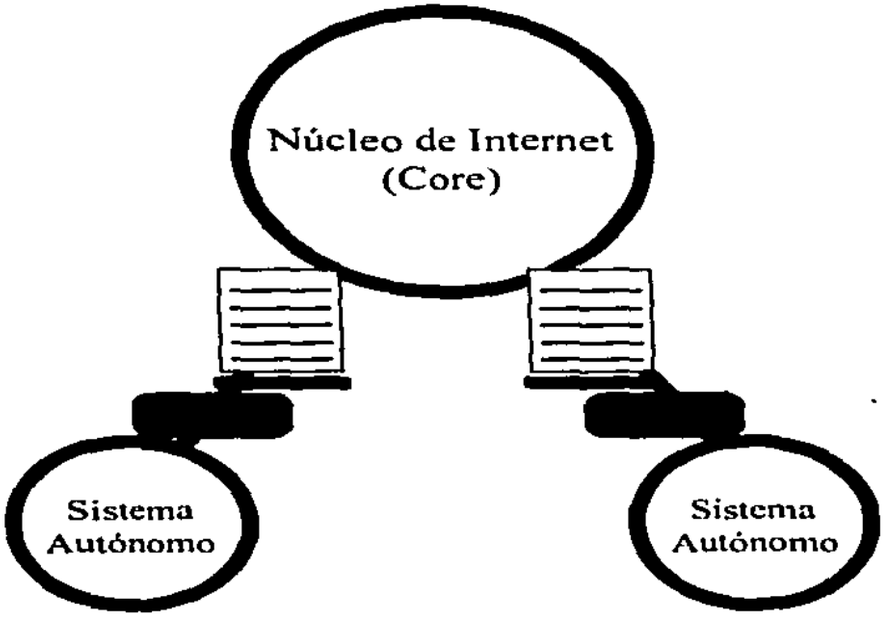
- El modelo jerárquico.
- El modelo de dominios de ruteo.

3.2.1 Modelo jerárquico

En sus primeros años de funcionamiento, la arquitectura de ruteo de Internet estaba basada en un sistema jerárquico de ruteadores. En los inicios ARPANET era el backbone de la red, constituyendo el sistema central llamado *core* (núcleo) y los ruteadores pertenecientes a ARPANET eran denominados *core gateways*.

Dentro de este sistema jerárquico toda la información de ruteo de las redes era pasada a través de los core gateways, los cuales la procesaban e intercambiaban entre ellos mediante un protocolo llamado *Gateway to Gateway Protocol* (GGP). Fuera del núcleo de Internet (ARPANET), existían grupos de redes llamados *sistemas autónomos* (AS). Un sistema autónomo es una colección de redes y ruteadores con sus propios mecanismos internos de ruteo. En los SA la información de ruteo se pasaba de unos a otros mediante el uso de un protocolo denominado Exterior Gateway Protocol (EGP). La principal desventaja del modelo jerárquico es que toda la información tiene que fluir de un sistema autónomo a otro pasando por el core (fig. 17).

Este modelo continua empleándose en las redes del departamento de defensa de los Estados Unidos (DDN), los cuales constituyen una porción de Internet.



– Gateway de Core

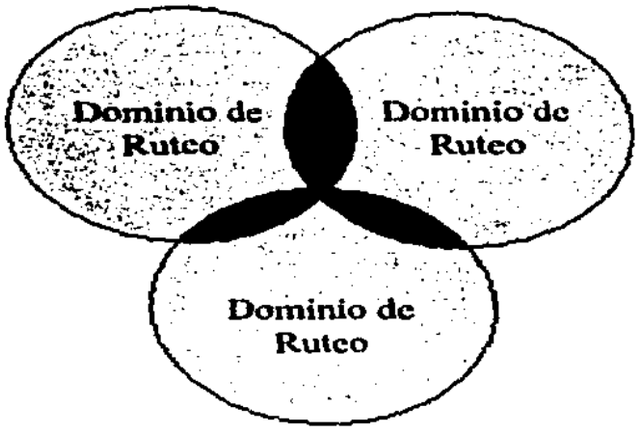
– Gateway externo

figura 17. Modelo Jerárquico.

3.2.2 Modelo de dominios de ruteo

El crecimiento exponencial de Internet hizo inoperable el funcionamiento de la red mediante el empleo del modelo de ruteo jerárquico, razón por la que surge el modelo de dominios de ruteo (routing domains).

El nuevo modelo está basado en colecciones de sistemas autónomos de igual jerarquía llamados dominios de ruteo. Los dominios de ruteo intercambian información con otros dominios utilizando los protocolos Border Gateway Protocol (BGP) ó Exterior Gateway Protocol (EGP). Cada dominio de ruteo procesa información que recibe de otros dominios. La figura 18 representa este modelo con tres círculos intersectados. Cada círculo constituye un dominio de ruteo. Las áreas sombreadas es donde se intercambia la información. Ambos modelos funcionan mediante tablas de ruteo.



Areas en donde los datos de Ruteo son intercambiados

figura 18. Modelo de dominios de ruteo

3.2.3 Tablas de Ruteo

No solamente los ruteadores IP tienen que tomar decisiones de ruteo, sino que todos los dispositivos de la red incluyendo los ruteadores y los hosts deben realizar tales decisiones.

En la mayoría de los hosts las decisiones de ruteo se dan de la siguiente manera.

- Si el host destino se encuentra en la red local se entrega la información directamente.
- Si el host destino esta en una red remota, los datos son entregados al ruteador IP local.

En IP las decisiones de ruteo se basan en la parte de red de una dirección. IP analiza los primeros bits de la dirección, determinando así la clase de la red y la porción de la dirección que identifica la red destino. Si la red destino es la red local, entonces se aplica la mascara de subred local para determinar la porción de subred y la parte que corresponde al host.

Después de determinar la red y en su caso la subred destino, IP busca en la tabla local de ruteo. La información es entregada según lo dispuesto de dicha tabla. Un tabla de ruteo puede ser construida por un administrador de redes (tablas estáticas) o bien de manera dinámica por un protocolo de ruteo (tablas dinámicas).

Las tablas de ruteo estáticas se manejan en sistemas con un solo ruteador local o cuando el número de ruteadores es reducido. Las tablas de ruteo dinámicas son utilizadas en sistemas multiruteadores, son esenciales cuando más de un ruteador puede ser utilizado para entregar datos a un mismo destino.

En una tabla de ruteo se tiene la información de mediante que hosts o ruteadores se puede accesar a una red destino, además de una entrada por omisión que indique hacia donde se dirigirán todas las transmisiones para redes destino no enlistadas en la tabla. La siguiente es una tabla de ruteo:

Destination	Gateway	Flags	Ref	Use	Interface
127.0.0.1	127.0.0.1	UH		0	192 lo0
132.248.102.0	132.248.102.95	U		3	132 le0
224.0.0.0	132.248.102.95	U		3	0 le0
default	132.248.102.254	UG		0	90

Una tabla de ruteo no contiene listas de enlaces punto a punto, sino que una ruta únicamente apunta hacia el siguiente ruteador (next hop). En el caso de redes remotas el host origen confía en el ruteador local para que entregue los datos, y este último confía en otros ruteadores. Los datagramas serán enviados de un ruteador a otro, hasta que finalmente lleguen a un ruteador conectado directamente a la red destino. Este último ruteador será el encargado de entregar los datos al host destino.

3.3 Sockets, Puertos y Daemons

Una vez que los datos han sido entregados al host destino, estos deben llegar al usuario o proceso correcto . Conforme los datos se mueven de una capa a otra de TCP/IP es necesario que sean entregados a los protocolos correctos en cada capa, para tal efecto IP utiliza *números de protocolos* para identificar los protocolos en la capa de transporte a los cuales deberá entregar los datos (TCP,UDP); los protocolos en la capa de transporte utilizan *números de puerto*, para de igual manera identificar los protocolos o aplicaciones de la capa de aplicación a los cuales deberán transferir los datos.

A la combinación de un número de protocolo y uno de puerto se le denomina *servicio*. Algunas combinaciones de números de protocolos y puertos son reservadas para servicios predeterminados como son telnet y FTP. En Unix los servicios predeterminados se especifican en el archivo /etc/services. Dentro de Unix una aplicación se puede ejecutar al mismo tiempo desde diversos equipos, para diferenciar los datos de un usuario-aplicación y otro se establecen números de puerto dinámicos. La generación de tales números permite que dos procesos simultáneos no tengan asignados los mismos números de puerto. En el caso de servicios predeterminados el puerto destino se asigna tomando el valor preestablecido y el puerto origen se asigna dinámicamente asegurándose que no sea un número de puerto origen en uso.

Para el correcto funcionamiento de los servicios de TCP/IP es necesario la carga de ciertos programas llamados *daemons*. Un daemon es un programa que se

ejecuta en segundo plano y sirve para proveer directamente de servicios o para manejar tablas que son utilizadas por otros programas dentro de la red. Generalmente los daemons son cargados en el proceso de arranque del host. Entre los principales daemons tenemos los siguientes:

Nombre	**Función**
inetd	Es el responsable de la mayoría de las operaciones de red TCP/IP en la capa de aplicación.
named	**Realiza operaciones de traducción de direcciones.**
routed y gated	Generación de tablas de ruteo dinámicas.
timed	**Sincronización de relojes de los diferentes sistemas de la red.**

El número de protocolo es un byte en la tercera palabra del encabezado de un datagrama. Los números de protocolo en Unix se encuentran asentados en el archivo /etc/protocols. Este archivo contiene una tabla con el nombre del protocolo y su número asociado. Cuando un datagrama llega al host, IP decide a que protocolo en la capa de transporte entregarlo comparando el número de protocolo contenido en el encabezado del datagrama contra los datos en la tabla.

Una vez que los datos pasan a la capa de transporte, esta última los entrega a la aplicación o proceso correcto con base a los números de puerto contenidos al inicio del encabezado del segmento TCP ó paquete UDP. El *número de puerto origen* indica el proceso o aplicación que envía los datos y el *número de puerto destino* especifica el proceso o aplicación al cual se tiene que entregar los mismos.

A la combinación de una dirección IP y el puerto destino se conoce como *socket.*

3.4 Servicios de TCP/IP en Solaris 2.X

Desde su fundación en 1982, la arquitectura del sistema de red de los equipos SUN esta basada en TCP/IP. (fig. 19)

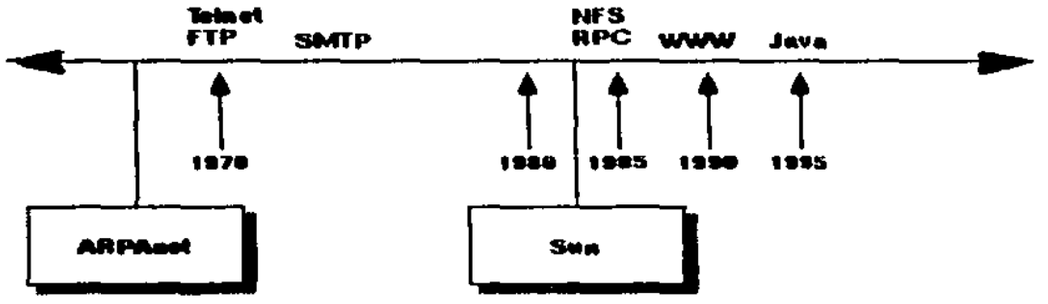


figura 19.Internet y Sun.

Entre los principales servicios aportados por SUN Microsystems a TCP/IP se encuentran:

- NSF (Network File System).- Permite que sistemas de archivos que físicamente residen en una computadora sean utilizados por otros equipos en la red, dando la impresión a los usuarios que el sistema remoto de archivos es parte de su sistema de archivos local.

- NIS (Network Information System).- Servicio de bases de datos distribuidas que permiten mantener la configuración y administración de una red completa de manera centralizada mediante un solo conjunto de archivos de configuración. Es de utilidad en el caso de administradores de red que manejan múltiples servidores. Cabe destacar que NIS fue diseñado para ambientes de red abiertos, donde el acceso libre entre los host del sistema es deseado. En el caso de redes conectadas a Internet por cuestiones de seguridad no es recomendable el uso de NIS.

- RPC (Remote Procedure Call).- Este servicio permite la ejecución de procedimientos en servidores remotos de manera transparente dando la impresión de que se ejecutan en el sistema local, igual que NIS tiene implicaciones de seguridad.

Solaris soporta también los protocolos estándar de Internet como son:

- FTP (File Transfer Protocol).- Permite copiar archivos entre computadoras, independientemente de su tipo y formato.

- Telnet.- Acceso mediante terminales remotas.

- SMTP (Simple Mail Transport Protocol).- Maneja el intercambio de correo electrónico.

- DNS (Domain Name System).- Convierte direcciones IP de nombre a direcciones numéricas.

- TFTP (Trivial File Transfer Protocol).- Protocolo residente en memoria de solo lectura (ROM), es utilizado para inicializar terminales sin unidades de disco.

- ICMP (Internet Control Message Protocol).- Genera paquetes conteniendo mensajes de error.

- ARP (Addres Resolution Protocol).- Convierte direcciones IP a direcciones Ethernet.

Entre las características agregadas en Solaris 2.X al TCP/IP estándar se encuentran las siguientes:

- Algoritmos para el establecimiento de unidades máximas de transmisión (MTU) de una ruta, lo que permite transportar datagramas de un modo más eficiente evitando su fragmentación al pasar por enlaces de red con MTUs diferentes.

- IP multicasting que permite utilizar más eficientemente los servicios de ancho de banda mediante la transmisión de un solo datagrama multicast a diferentes sitios predeterminados en lugar de enviar un datagrama por cada sitio. Esta mejora es de especial utilidad en los servicios de vídeoconferencia o cualquier otro uso que requiera de conexiones entre aplicaciones o usuarios uno a varios ó varios a varios.

El funcionamiento de multicasting se representa en el siguiente diagrama (fig. 20):

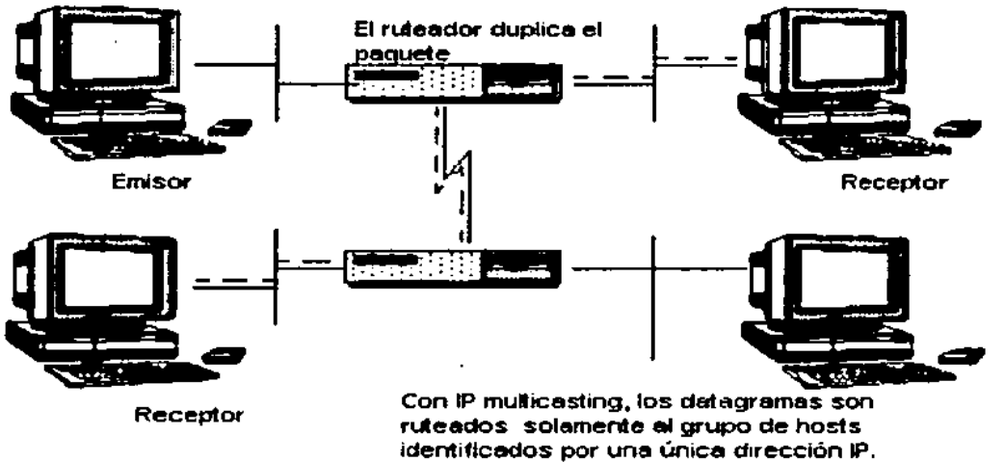


figura 20. Direccionamiento Multicasting,

- Mecanismos de descubrimiento de ruteadores que permiten configurar los servicios de ruteo de IP de manera dinámica, permitiendo establecer rutas alternativas para la entrega de datagramas.

Capítulo 4. Caso práctico: Configuración de TCP/IP en Solaris 2.5.1

Capítulo 4. Caso práctico: Configuración de TCP/IP en Solaris 2.5.1

4.1 Preparativos

Antes de comenzar la configuración de los servicios de TCP/IP en Solaris, será necesario que previamente se tenga lo siguiente:

- Equipo SUN instalado, con todos sus dispositivos funcionando y conectado a la red.
- Sistema operativo Solaris 2.5.1 cargado y funcionando.
- Clave de acceso al sistema como superusuario (root).

La configuración se TCP/IP se realizará mediante la edición directa de archivos del sistema y el uso de comandos. La estructura básica de archivos de Solaris es la siguiente:

Directorio	Tipo de archivos contenidos
/	Es la base del sistema de archivos, todos los archivos y directorios sin importar su ubicación física, están contenidos de forma lógica en esta directorio llamado raíz o root.
/bin y /etc	Contienen los archivos de configuración y archivos ejecutables del sistema.
/kernel	Núcleo de Unix y sus manejadores relacionados.
/var	Archivos y directorios de administración del sistema y los correos electrónicos.
/opt	Aplicaciones de software.
/proc	Lista de procesos activos del sistema.
/tmp	Memoria Temporal.
/export	Sistema de archivos compartidos.
/usr	Generalmente maneja las cuentas de usuarios.
/dev	Referencia hacia dispositivos.

Los archivos a trabajar se encuentran en el directorio /etc.

Una vez instalado el equipo y con el sistema operativo funcionando será necesaria la siguiente información:

- Dirección IP del ruteador local por omisión.
- Dirección IP de los servidores de nombres.
- Dirección IP que se asignará al equipo a configurar.
- Nombre de host.
- Nombre de dominio.
- Mascara de subred.
- Dirección Broadcast.

4.1.1 Dirección IP del ruteador local por omisión.

Debemos conocer de antemano la dirección IP del ruteador que utilice la red local a la que este conectada el servidor a configurar. En el caso de existir más de un ruteador local, adicionalmente se deberá conocer el protocolo de ruteo utilizado en la red. Si solamente se cuenta con un ruteador no será necesario ejecutar ningún protocolo de ruteo, ya que se utilizarán tablas de ruteo estáticas.

La dirección IP del ruteador debe ser proporcionada por el administrador de la red local o proveedor de conexión.

4.1.2 Dirección IP de los servidores de nombres (NS)

Estas direcciones especifican los host que se encargarán de realizar el servicio de traducción de direcciones por nombre a direcciones IP numéricas. Existen muchas direcciones de servidores de nombres distribuidos por toda Internet, se recomienda tomar más de una dirección de servidores de nombres, incluyendo algunos que se encuentren geográficamente cerca o pertenezcan a la red local.

Los valores de NS deben ser proporcionados por el administrador de red a la que se esté conectado o en su defecto por el proveedor de la conexión a Internet.

4.1.3 Dirección IP que se asignará al equipo a configurar.

En el caso de que el equipo sea un nodo más de una red ya existente este número será proporcionado por el administrador de la misma, en caso de que esté montando una nueva red y esta no sea subred de alguna existente, se deberá recurrir a la división pertinente de InterNIC, para solicitar el rango de direcciones IP a utilizar en la nueva red. La dirección IP permitirá identificar el host de manera única dentro de Internet.

4.1.4 Selección de un nombre de host.

Para el más fácil manejo de las direcciones IP, estas se asocian con un nombre. Se recomienda que el nombre del host sea relacionado con la función o ubicación del host al cual será asignada. Una vez que el nombre del host ha sido establecido se anexa al dominio que le corresponde y se registra ante el administrador de dominio correspondiente. El proceso de registro ante InterNIC puede ser realizado directamente por el administrador del host o de la red local.

4.1.5 Nombre de dominio

Constituye la parte final de una dirección IP por nombre y corresponde a la parte de red. El nombre de dominio es proporcionado por la división correspondiente de InterNIC en el caso de una nueva red, tratándose subredes o la instalación de un nuevo host a una red local, los valores son proporcionados por el administrador de la red local. El nombre de dominio permite el acceso a servidores dentro del mismo dominio con solamente proporcionar el nombre del host.

4.1.6 Máscara de subred

La máscara de subred es aplicada en los casos en que una red se tiene que dividir en subredes. La máscara de subred permite extender la parte de red de una dirección IP.

4.1.7 Dirección Broadcast

La dirección Broadcast por omisión es una dirección en la que todos los bits de la parte de host tiene el valor de uno. Por ejemplo, si tenemos la subred 132.248.102.0 la dirección broadcast correspondiente será 132.248.102.254, la

dirección broadcast se puede modificar pero se recomienda que se utilice la convencional, a menos de que el servidor se instale en una red o subred con número de broadcast diferente al convencional.

Los datos del host que será utilizado en el presente trabajo son los siguientes:

Dirección IP del ruteador local por omisión.	132.248.102.254
Dirección IP de los servidores de nombres	132.248.10.2 132.248.1.3
Dirección IP :	132.248.102.95
Nombre de host.	dgb1
Nombre de dominio.	Cuautitlan2.unam.mx
Mascara de subred.	255.255.255.0
Dirección Broadcast.	132.248.102.254

Cabe destacar también lo siguiente:

- Forma parte de una subred que depende de una red de clase B.
- La interfaz que se utiliza es una tarjeta Ethernet.
- Utiliza un solo ruteador local por lo que aplica tablas de ruteo estáticas.

4.1.8 Recomendaciones

Durante del proceso de instalación se recomienda lo siguiente:

1. Respaldar el sistema completo por lo menos antes y al terminar la configuración.
2. Procurar realizar los cambios en la configuración de manera progresiva, de ser posible realizar pruebas entre un cambio y otro.
3. Evitar apagar la máquina sin ejecutar antes el procedimiento de cierre del sistema (shutdown). Algunas de las maneras seguras de hacerlo son tecleando lo siguiente:

```
shutdown now
```

o bien

sync
init 0

4. Llevar cuenta de los cambios realizados procurando tomar nota de configuraciones originales y nuevas.

5. Antes de modificar un archivo de configuración, realizar una copia del mismo, si por ejemplo, se procediera a modificar el archivo /etc/hosts la copia se realizaría de la siguiente manera:

   cp /etc/hosts /etc/hosts.respaldo

   En este caso se decidió utilizar la extensión .respaldo para identificar a la copia del archivo, pero se podría utilizar cualquier otro nombre.

   De tener la necesidad de restaurar el archivo /etc/hosts a su condición original simplemente se ejecutaría:

   cp /etc/hosts.respaldo /etc/hosts

6. Tomar nota de los mensajes de error, desde el primer momento en que aparecen.

7. Mantener la clave de root de manera confidencial, procurando cambiarla de manera periódica, cualquier persona con la clave de root no solamente puede desconfigurar los servicios de TCP/IP, sino que también puede eliminar o dar un mal uso a toda la información del sistema, aun sin utilizar la consola del servidor.

Estas recomendaciones no son indispensables para la configuración de los servicios de TCP/IP, pero en caso de presentarse algún problema podrían ahorrar horas o días de trabajo innecesario y la pérdida parcial o total de información.

4.1.9 Carga de Inetd

El primer paso en la configuración de TCP/IP es la carga de **inetd.** Este daemon es el soporte principal para los servicios de Internet, por lo que debe ser incluido en los archivos de arranque del host. Por omisión se encuentra especificado en el archivo /etc/init.d/inetsvc.

Cuando es inicializado, **inetd** toma los valores de configuración del archivo /etc/inetd.conf . Este archivo contiene los nombres de los servicios que **inetd** inicializa. Los servicios se pueden deshabilitar o agregar realizando cambios en el archivo inetd.conf.

En inetd.conf se especifica un servicio por entrada. El carácter # sirve como marcador de comentario. Un ejemplo de entrada de inetd.conf es la siguiente línea:

ftp stream tcp nowait root /usr/etc/in.ftpd in.ftpd

Los campos contenidos en una entrada de intetd.conf, de izquierda a derecha son los siguientes:

Campo	Descripción
Nombre	Indica el nombre del servicio, debe coincidir con el nombre agregado en el archivo /etc/services. El valor del ejemplo es ftp.
Tipo	Tipo de flujo de datos utilizado. Los tipos más comunes son: stream — Servicio de entrega de flujo de datos. Es proveído por TCP. dgram — Servicio de entrega de datagramas. Es proveído por UDP. raw — Servicio de entrega de datagramas directamente a IP sin pasar por TCP o UDP.
Protocolo	Nombre del protocolo, debe coincidir con alguno de los valores contenidos en el archivo /etc/protocols
Wait-status	Los valores posibles son "wait" ó "nowait". Generalmente los servicios de datagramas requieren un valor "wait" y los servicios de "stream" requieren un valor "nowait"
uid	Nombre de usuario bajo el cual se ejecutará el servicio. Se puede utilizar cualquier nombre de usuario válido, generalmente *root*. En el caso del servicio **finger** por razones de seguridad frecuentemente se establece el usuario *nobody* ó *daemon*.

Servicio	Ruta completa del programa que ejecuta el servicio. En el caso de servicios propios de **inetd** se coloca el valor "internal" en lugar de una ruta.
Argumentos	Argumentos que se requiera pasar al programa que ejecuta el servicio.

Por lo regular no es necesario modificar el archivo que viene con la instalación de Solaris, salvo en los casos que por razones de seguridad se decida agregar o deshabilitar un servicio.

Para deshabilitar un servicio se antepone un signo # a su entrada. Para reestablecer el servicio bastará con borrar el carácter agregado.

El archivo /etc/inetd.conf del servidor de la investigación contiene las siguientes entradas:

```
ftp     stream  tcp     nowait  root    /usr/sbin/in.ftpd       in.ftpd
telnet  stream  tcp     nowait  root    /usr/sbin/in.telnetd    in.telnetd
name    dgram   udp     wait    root    /usr/sbin/in.tnamed     in.tnamed
shell   stream  tcp     nowait  root    /usr/sbin/in.rshd       in.rshd
login   stream  tcp     nowait  root    /usr/sbin/in.rlogind    in.rlogind
exec    stream  tcp     nowait  root    /usr/sbin/in.rexecd     in.rexecd
comsat  dgram   udp     wait    root    /usr/sbin/in.comsat     in.comsat
talk    dgram   udp     wait    root    /usr/sbin/in.talkd      in.talkd
uucp    stream  tcp     nowait  root    /usr/sbin/in.uucpd      in.uucpd
#tftp   dgram   udp     wait    root    /usr/sbin/in.tftpd      in.tftpd -s /tftpboot
finger  stream  tcp     nowait  nobody  /usr/sbin/in.fingerd    in.fingerd
#systat stream  tcp     nowait  root    /usr/bin/ps             ps -ef
#netstatstream  tcp     nowait  root    /usr/bin/netstat  netstat -f inet
time    stream  tcp     nowait  root    internal
time    dgram   udp     wait    root    internal
echo    stream  tcp     nowait  root    internal
echo    dgram   udp     wait    root    internal
discard stream  tcp     nowait  root    internal
discard dgram   udp     wait    root    internal
daytime         stream  tcp     nowait  root    internal
daytime         dgram   udp     wait    root    internal
chargenstream   tcp     nowait  root    internal
chargendgram    udp     wait    root    internal
fs      stream  tcp     wait nobody /usr/openwin/lib/fs.auto    fs
```

4.2 Configuración de Interfaces.

El siguiente paso en la configuración de TCP/IP es la configuración de la interface, los principales valores que se darán de alta son los siguientes:

- Dirección IP del host.
- Máscara de subred.
- Dirección Broadcast

La configuración de interfaces dentro de Solaris se realiza mediante el comando **ifconfig** (interface configure) . El comando **ifconfig** permite establecer y verificar los valores asignados a las interfaces de red.

La configuración de la interface puede efectuarse de dos modos:

4.2.1 Modo centralizado

En el modo centralizado se utiliza el comando **ifconfig**, el cual, se carga en uno de los archivos de inicialización de Solaris. Aplicando este modo simplemente se introduce la línea en archivo de arranque /etc/rcs.d/s30rootusr.sh. En el caso del servidor de la investigación, la forma de declararlo es tecleando lo siguiente:

```
ifconfig lo0 127.0.0.1
ifconfig le0 132.248.102.95 netmask 255.255.255.0 broadcast 132.248.102.255
```

Los argumentos utilizados en esta instrucción son los siguientes:

Argumento	Función
Interface	Nombre de la interface de red. El nombre de la interface de red Ethernet en Solaris es le0. Otra interface que se crea por omisión es lo0 la cual representa a la dirección loopback.
Dirección	Es la dirección IP asignada a la inteface. Se puede utilizar la dirección en forma numérica o por nombre, pero se recomienda la forma numérica. Al usar una dirección por nombre, esta debe de todas formas ser traducida por el host en una numérica para su asignación.
Máscara de red	En el caso de no utilizar subredes se puede omitir este campo. El valor asignado en el caso de investigación es 255.255.255.0 ya que el host se encuentra dentro de una red clase B subdividida con 8 bits de subred.

Dirección broadcast	Especifica la dirección broadcast empleada por la red o subred.

La ventaja de este tipo de carga en el archivo de arranque, es que toda la información de cada interface se encuentra en una linea, y se modifica solamente un archivo. El inconveniente es que si al modificar un archivo de arranque se realiza un cambio indebido, el equipo podría bloquearse al encendido.

4.2.2 Modo distribuido

Los archivos de arranque contenidos en Solaris 2.5.1. vienen preconfigurados, para tomar los valores de **ifconfig** de los siguientes archivos.

/etc/netmasks
/etc/hosts
/etc/networks

Retomando el mismo ejemplo los archivos quedan de la siguiente manera:

```
/etc/netmasks
132.248.0.0   255.255.255.0

/etc/hosts
127.0.0.1       localhost
132.248.102.95  DGB1    loghost

/etc/networks
loopback        127
#
# Internet networks
#
arpanet                 10              arpa    # Historical
```

La ventaja de este modo es que no es necesario modificar archivos de arranque, ya que por omisión la instrucción ifconfig viene en estos. El inconveniente es que de haber algún problema sería necesaria la revisión de varios archivos, incluyendo los de arranque.

Una vez configurada la interface se pueden verificar sus parámetros tecleando la siguiente instrucción:

ifconfig -a

El argumento -a le especifica a ifconfig que muestre la configuración de todas las interfaces. El sistema dará el siguiente resultado:

```
lo0: flags=849<UP,LOOPBACK,RUNNING,MULTICAST> mtu 8232
      inet 127.0.0.1 netmask ff000000
le0: flags=863<UP,BROADCAST,NOTRAILERS,RUNNING,MULTICAST> mtu 1500
      inet 132.248.102.95 netmask ffffff00 broadcast 132.248.102.255
      ether 8:0:20:72:f0:98
```

En este caso la interfaz lo0 se refiere a la dirección loopback, le0 es el nombre de dispositivo utilizado por Solaris para designar a la tarjeta Ethernet.

4.3 Servicios de Ruteo

El ruteo permite comunicar equipos dentro y fuera de la red. El ruteo consiste en el envío de los datagramas basándose en la información contenida en las tablas de ruteo. Los protocolos de ruteo son programas que intercambian información para construir tablas de ruteo de manera dinámica.

La configuración de ruteo no siempre implica el uso de protocolos de ruteo, en los casos en que se utiliza un solo ruteador o se configura una red aislada de otras redes, el uso de protocolos de ruteo no es necesario.

Las tres configuraciones de ruteo son las siguientes:

Configuración	Uso
Configuración de ruteo mínima	En el caso de redes aisladas y que no contienen subredes. La tabla de ruteo que se utiliza es generada en el momento de ejecutar el comando **ifconfig**.

Configuración de ruteo estática	Se implementa en redes con un solo ruteador o un número limitado de los mismos. Las tablas de ruteo estáticas son construidas mediante el uso del comando **route.**
Configuración de ruteo dinámica	Utilizada en redes con múltiples ruteadores. Manejan algoritmos para utilización de rutas óptimas y actualizan los datos de las tablas dinámicamente con base a cambios en la red.

En el caso del servidor a configurar en la investigación, se cuenta con un solo ruteador, por lo que se aplicará una configuración de ruteo estática.

La configuración de ruteo estática se basa en la configuración de ruteo mínima. Al ejecutar el comando **ifconfig**, este crea la tabla de ruteo mínima habilitando así la dirección loopback y la comunicación con otros hosts dentro de la misma red ó subred en el caso de manejo de subredes.

Para poder accesar otros hosts en otras redes ó subredes, será necesario agregar la entrada de los ruteadores a utilizar manualmente. El comando **route** permite agregar o eliminar entradas de la tabla de ruteo.

La sintaxis para agregar el ruteador al servidor del presente trabajo es la siguiente:

route -n add default 132.248.102.254

A lo cual es sistema deberá responder:

add net default:gateway 132.248.102.254

Los argumentos utilizados son los siguientes:

Argumento	Función
Opciones	La opción -n previene al comando **route** de tratar de utilizar servicios de conversión de nombres que podrían no estar instalados o aun no hayan sido inicializados.
Acción	**add** - Ingresa una entrada a la tabla de ruteo. **delete** - Elimina una entrada de la tabla de ruteo.

Destino	Especifica la red a la cual se quiere accesar, el valor *default* indica que la entrada a utilizar en todos los casos no declarados en la tabla.
Ruteador	Indica el ruteador o host utilizado para comunicarse con la red especificada en campo destino de la misma entrada.

Las tablas de ruteo se almacenan en la memoria de la computadora, por lo que es necesario asignarlas cada vez que se enciende el equipo.

Para asignar la ruta estática desde el arranque será necesario incluir la dirección IP del ruteador por omisión en el archivo /etc/defaultrouter (en caso de no existir crearlo). El contenido de dicho archivo en este caso será:

132.248.102.254

Al reiniciar el equipo **ifconfig** configurado anteriormente cargará la tabla de ruteo mínima y el archivo de arranque /etc/init.d/inetinit buscará el archivo defaultrouter, de existir tomará su contenido y lo asignará como ruteador por omisión, completando con esto la tabla de ruteo estática de nuestro ejemplo.

Una vez reiniciado el equipo la tabla de ruteo se puede verificar con el comando **netstat** de esta manera:

netstat -nr

Donde los argumentos -nr indican :

-n Muestra las direcciones IP en forma numérica.
-r Despliega la tabla de ruteo.

El resultado es :

Routing Table:

Destination	Gateway	Flags	Ref	Use	Interface
127.0.0.1	127.0.0.1	UH	0	192	lo0
132.248.102.0	132.248.102.95	U	3	132	le0
224.0.0.0	132.248.102.95	U	3	0	le0
default	132.248.102.254	UG	0	90	

El comando **ping** permite verificar la comunicación con otros hosts. La forma de utilizar **ping** para tal efecto es tecleando el comando ping seguido de una dirección IP conocida:

ping 132.248.102.36

Las posibles respuestas son:
"132.248.102.36 is alive" ó bien "no answer from 132.248.102.36"

En el caso de la primer respuesta la comunicación se ha establecido con éxito. La segunda respuesta implica imposibilidad en la comunicación la cual puede estar dada por alguna de las siguientes causas.

- La dirección IP se tecleo incorrectamente.
- La dirección IP solicitada no corresponde a un host existente.
- El equipo remoto está apagado o no se encuentra disponible.
- La red a la que pertenece el equipo remoto no esta disponible.
- La tabla de ruteo de nuestro servidor está mal configurada.
- El ruteador utilizado no funciona correctamente

Se sugiere que se utilicen direcciones conocidas, tanto de la misma red como externas.

4.4 Servicios de DNS.

DNS (Domain Name System) consiste en un sistema de bases de datos distribuidas, con el objeto principal de determinar direcciones IP a partir de nombres de hosts. A este proceso se le llama resolución (resolving). DNS es implementado mediante un sistema cliente/servidor. A la parte del cliente que es la que solicita la resolución de nombres se le llama agente de resolución (resolver). La parte de DNS que responde a solicitudes de resolución se le denomina servidor de nombres y funciona mediante la carga de el daemon **named**.

Al arrancar el sistema, el archivo /etc/init.d/inetsvc busca la existencia del archivo /etc/named.boot (principal archivo de configuración de **named**), de encontrarlo carga el daemon.

En el caso de la investigación, el host utilizará un servidor de nombres externo, por lo que el servicio de DNS se configurará solamente como agente de resolución, por tanto, la carga de **named** no será necesaria.

La configuración del agente de resolución, está basada en dos archivos de configuración:

/etc/nsswitch.conf
/etc/resolv.conf

El archivo /etc/nsswitch.conf determina los servicios a utilizar por el agente de resolución y el orden de los mismos, mediante la entrada:

hosts:

Los valores posibles para esta entrada son:

Valor	Función
files	Al solicitar una aplicación el servicio de resolución, se buscará la dirección IP correspondiente en el archivo /etc/hosts.
Dns	Se utilizarán servidores de nombres designados en el archivo /etc/resolv.conf.
nis	El sistema de información de red (NIS), se empleara para las resoluciones.

Las opciones pueden ser utilizadas de manera combinada, pero no es recomendable usar **dns** y **nis** simultáneamente, ya que se pueden producir algunos conflictos. Cada opción se separa de la siguiente por un espacio en blanco. El archivo /etc/nsswitch.conf de la investigación quedará de la siguiente manera:

```
passwd:     files
group:      files
hosts:      files dns
networks:   files
protocols:  files
rpc:        files
ethers:     files
netmasks:   files
```

```
bootparams: files
publickey:  files
# At present there isn't a 'files' backend for netgroup;  the system will
#   figure it out pretty quickly, and won't use netgroups at all.
netgroup:   files
automount:  files
aliases:    files
services:   files
sendmailvars:   files
```

En este caso se emplearán las opciones **files** y **dns.** Esto implica que las búsquedas se harán primero de manera local y de no ser resueltas, se recurrirá a un servidor de nombres.

Cada entrada del archivo /etc/hosts asocia una dirección IP a su respectivo nombre de host y un alias o nombre alternativo, separados entre si por espacios en blanco.

El contenido del archivo /etc/hosts del host del presente trabajo será:

```
127.0.0.1               localhost
132.248.102.95          DGB1                        loghost
132.248.10.1 servidor.unam.mx           servidor
132.248.10.3 condor.dgsca.unam.mx       condor
```

La primera entrada del archivo establece la dirección loopback con el nombre de localhost. La segunda entrada hace referencia a la dirección del host, asignándole el alias loghost. Las últimas entradas especifican otros servidores fuera de la red local.

El uso del archivo /etc/hosts, permite ahorrar tiempo y conexiones al establecer en él los nombres de hosts de uso más frecuente.

El archivo /etc/resolv.conf controla la forma en que el resolver utiliza DNS. Los datos contenidos en este archivo especifican los servidores de nombres y el dominio por omisión a utilizar. El archivo /etc/resolv.conf en este caso es:

```
# Dominio local
domain      cuautitlan2.unam.mx
nameserver  132.248.10.2
nameserver  132.248.1.3
```

Las entradas **nameserver** especifican las direcciones IP de los servidores de nombres a utilizar. Los servidores son consultados en el orden en que se establecen en el archivo. En caso de que el archivo /etc/resolv.conf no existiera o no tuviera datos, todas las peticiones de resolución serían enviadas al host mismo.

La entrada **domain** define el dominio por omisión, esto implica que a cualquier petición de resolución de nombre de host que no contenga un punto, el agente de resolución agregará el punto y el dominio por omisión declarado, antes de realizar la petición de resolución.

Una vez configurado DNS puede ser verificado su correcto funcionamiento mediante el comando **nslookup**, si se introduce sin argumentos, este deberá devolver el nombre y la dirección IP del primer servidor de nombres disponible de acuerdo a la lista proporcionada /etc/resolv.conf. Ejemplo:

```
# nslookup
Server: ns.dgsca.unam.mx
Address: 128.66.12.2

>
```

Al aparecer el símbolo mayor que (>), podemos introducir otros nombres de hosts, y nslookup devolverá su dirección IP. Ejemplo:

```
>servidor.unam.mx
Name: servidor.unam.mx
Address: 132.248.10.4
```

Para terminar la ejecución de **nslookup**, se introduce la palabra **exit** en el prompt (>).

```
>exit
```

Habiendo confirmado el funcionamiento de **nslookup**, se procede a ejecutar una aplicación utilizando una dirección por nombre que no este asignada en el

archivo /etc/hosts, finalizando con esto la verificación de los servicios de DNS. Por ejemplo:

telnet asteroide.acatlan.unam.mx.

A lo que el sistema deberá responder con la solicitud de nombre de usuario y contraseña del host asteroide.acatlan.unam.mx

4.4 Configuración de los servicios básicos

Los servicios de Internet suministrados por Solaris funcionan mediante programas llamados *servidores* (servers). Para que un servidor funcione necesita tener asignado un protocolo (generalmente TCP ó UDP), un número de puerto y que el programa que lo activa sea cargado en memoria.. La carga de los servidores se realiza mediante daemons, siendo el principal **inetd**.

Los servicios se encuentran declarados en el archivo /etc/services, donde cada línea de este archivo consta del nombre del servicio, un número de puerto, un nombre de protocolo y una lista de alias para el servicio. El archivo /etc/services es copiado al disco duro del equipo al instalar Solaris. Los servicios comunes de Internet, incluso algunos servicios de poco uso se encuentran ya habilitados en este archivo.

Para agregar un servicio será necesario anexar su entrada a /etc/services y cargar su daemon correspondiente. En caso de requerir deshabilitar un servicio bastará con borrar su entrada ó anteponer a esta el símbolo # en /etc/services , en el archivo que invoca a su daemon y reinicializar el equipo.

En el caso de los servicios inicializados por el daemon **inetd** , los datos del nombre de servicio y protocolo del archivo /etc/services deberán coincidir con los de entrada correspondiente de /etc/inetd.conf. Por ejemplo si se tiene la entrada en /etc/inetd.conf:

telnet stream tcp nowait root /usr/etc/in.telnetd in.telnetd

La entrada correspondiente en /etc/services será:

telnet 23/tcp

Por razones de seguridad o necesidades específicas de los usuarios de la red será necesario deshabilitar o agregar servicios.

Entre los principales servicios de TCP/IP en Solaris tenemos los siguientes:

4.4.1 Systat (TCP puerto 11)

Diseñado para proveer información del estado del host a otros equipos. Un ejemplo claro del uso de este servicio es mediante el ejecución del comando **who** de manera remota. Se puede verificar si dicho servicio se encuentra activo mediante la instrucción desde otra computadora de la red:

telnet 132.248.102.63 11

Los argumentos utilizados en **telnet** son:

- La dirección IP del equipo a verificar.
- El número de puerto, específicamente el 11.

En caso de obtener una respuesta de rechazo de conexión significa que el servicio no está activo.

4.4.2 (FTP) File Transfer Protocol (TCP puertos 20 y 21)

FTP permite la transferencia de archivos entre sistemas. El puerto 21 es utilizado para la transferencia de comandos y el puerto 20 ocasionalmente se emplea para transmitir datos, aunque es más común que el cliente y el servidor negocien el número de puerto a utilizar tomando por lo general una valor mayor a 1024.

Cuando un usuario contacta a una computadora remota mediante FTP requiere un nombre de usuario (login) y contraseña. Al ingresar una clave de solicitud de FTP, esta es enviada a través de la red sin ser encriptada, por esta razón

se recomienda no utilizar la clave de root para accesar FTP al servidor desde una terminal.

Una manera de limitar el acceso a FTP es mediante la configuración del archivo /etc/ftpusers. Este archivo contiene una lista de cuentas a las que no les es permitido utilizar FTP. El archivo /etc/ftpusers deberá contener por lo menos las cuentas que no estén asignadas a usuarios reales, un ejemplo del contenido de /etc/ftpusers es el siguiente:

```
root
uucp
news
bin
ingres
nobody
daemon
```

Donde cada línea representa un usuario previamente establecido en el archivo /etc/passwd.

4.4.2.1 Establecimiento de un FTP anónimo

Un servicio de FTP anónimo es aquel al que se tiene acceso con un nombre de usuario y clave de acceso de dominio público. Por convención se ha establecido en FTP anónimo la palabra *anonymous* como nombre de usuario y una dirección electrónica como clave de acceso.

La creación de un FTP anónimo es una cuestión de seguridad muy delicada, ya que permite el acceso a toda la comunidad de Internet al host. Un FTP anónimo se debe configurar asegurándose que los usuarios puedan obtener información de ciertos directorios, pero no colocar información o modificar la estructura del sistema de archivos.

Los pasos para el establecimiento de un FTP anónimo son los siguientes:

1. Ingrese desde la consola con el usuario root.

2. Agregar el usuario ftp al archivo /etc/passwd, y una nueva entrada al archivo /etc/group para el grupo de usuarios de ftp anónimo.

3. Crear un directorio ftp en el directorio designado para el establecimiento del ftp anónimo (ej. /export), darle la propiedad del mismo al usuario ftp y que no pueda ser utilizado para escritura por ningún otro usuario.

4. Agregar un directorio bin con propiedad de root dentro del directorio ftp, que no tenga atributos de escritura para ningún otro usuario.

5. Colocar el programa ls en este directorio con atributos solo de ejecución.

6. Elaborar el directorio etc en el directorio ftp, el directorio etc recién creado debe ser propiedad de root, y ningún otro usuario tendrá derecho de escritura sobre él.

7. Crear los archivos passwd y group en ftp/etc, el archivo passwd contendrá la entrada adicionada a /etc/passwd; el archivo group tendrá la entrada adicionada al archivo /etc/group. Después de su creación, copiar los archivos /etc/nsswitch.conf, /etc/netconfig a el directiorio ftp/etc y cambiar los cuatro archivos a modo de sólo lectura.

8. Establecer un directorio pub en el directorio ftp cuya propiedad sea del usuario ftp. Si se desea que los usuarios de ftp puedan colocar archivos establecer un modo de lectura, escritura, y ejecución (no recomendable). Si se desea que los usuarios de ftp anónimo solamente puedan obtener archivos, pero no colocar, establezca el modo de sólo lectura para este directorio.

9. Dentro de Solaris se requiere de la copia de ciertos archivos contenidos en el directorio /usr/lib hacia el subdirectorio /usr/lib que debe crearse en el directorio donde se establezca el ftp anónimo. Estos archivos son necesarios para el funcionamiento del servicio de ftp anónimo.

10. Crear el directorio /dev dentro del directorio creado para montar el servicio de ftp anónimo, el cual contendrá el dispositivo *zero* creado mediante la instrucción **mknod.**

11. Colocar los archivos a disposición en el directorio /etc/ftp/pub. Para prevenir que estos archivos puedan ser sobre escritos se debe establecer el

modo 644 a todos los archivos del directorio y cambiar la propiedad de los mismos a otro usuario distinto a ftp.

Para ejecutar todo el proceso anterior se puede ejecutar el siguiente *script*, donde el único dato que opcionalmente puede ser modificado es el valor de la entrada ftphome que indica el nombre del directorio donde se instalará ftp anónimo. Las entradas que inician con # son comentarios y pueden ser omitidas.

```
#!/bin/sh
ftphome="/export/ftp"
echo Fixing ${ftphome} for SunOS 5.x

grep '^ftp:' /etc/passwd >/dev/null
if [ $? != "0" ] ; then
        echo adding user ftp
        echo ftp:x:30000:30000:Anonymous FTP:$ftphome:/nosuchshell >>
/etc/passwd
        echo ftp:NP:6445:::: >> /etc/shadow
fi

mkdir $ftphome

mkdir $ftphome/pub
mkdir $ftphome/bin
mkdir $ftphome/dev
mkdir $ftphome/etc
mkdir $ftphome/usr

mkdir $ftphome/usr/lib

cp /usr/bin/ls ${ftphome}/bin
chmod 111 ${ftphome}/bin/ls

#The following are needed for basic operation
cp /usr/lib/ld.so* ${ftphome}/usr/lib
cp /usr/lib/libc.so.1 /usr/lib/libdl.so.1 ${ftphome}/usr/lib
cp /usr/lib/libintl.so.1 /usr/lib/libw.so.1 ${ftphome}/usr/lib
cp /etc/passwd /etc/group /etc/netconfig ${ftphome}/etc
```

```
        echo "You might not want the current full copy of your /etc/passwd file in
${ftphome}/etc"

        #The following are needed for 'ls' to resolve NIS names
        cp /usr/lib/nss*.so.1 ${ftphome}/usr/lib
        cp /usr/lib/libnsl.so.1 ${ftphome}/usr/lib
        cp /usr/lib/straddr.so ${ftphome}/usr/lib
        cp /etc/nsswitch.conf ${ftphome}/etc

        chmod 555 ${ftphome}/usr/lib/*
        chmod 444 ${ftphome}/etc/*

        # make device nodes. ticotsord and udp are necessary for
        # 'ls' to resolve NIS names.
        prefix="/devices/pseudo/mm@0:"

        for device in zero
        do
                line=`ls -l ${prefix}${device} | sed -e 's/,//'`
                major=`echo $line | awk '{print $5}'`
                minor=`echo $line | awk '{print $6}'`
                rm -f ${ftphome}/dev/${device}
                mknod ${ftphome}/dev/${device} c ${major} ${minor}
        done

        prefix="/devices/pseudo/tl@0:"

        for device in ticotsord
        do
                line=`ls -l ${prefix}${device} | sed -e 's/,//'`
                major=`echo $line | awk '{print $5}'`
                minor=`echo $line | awk '{print $6}'`
                rm -f ${ftphome}/dev/${device}
                mknod ${ftphome}/dev/${device} c ${major} ${minor}
        done

        prefix="/devices/pseudo/clone@0:"

        for device in tcp udp
        do
                line=`ls -l ${prefix}${device} | sed -e 's/,//'`
```

```
        major=`echo $line | awk '{print $5}'`
        minor=`echo $line | awk '{print $6}'`
        rm -f ${ftphome}/dev/${device}
        mknod ${ftphome}/dev/${device} c ${major} ${minor}
done
chmod 666 ${ftphome}/dev/*

#put chmod's at end
chmod 555 $ftphome/usr/lib
chmod 555 $ftphome/usr
chmod 555 $ftphome/bin
chmod 555 $ftphome/dev
chmod 555 $ftphome/etc

#chmod 777 $ftphome/pub
chmod 755 $ftphome/pub
chmod 555 $ftphome

# in case some of the files existed before and were not owned by root
chown -R root $ftphome

#sfb exit for now
exit

# This is for the wuarchive ftp server
echo Setting up wuarchive FTP server
cp -r /usr/local/etc/msgs ${ftphome}/etc
chmod -R a+r ${ftphome}/etc/msgs
chmod a+x ${ftphome}/etc/msgs

cp -r /usr/local/etc/msgs /etc
chmod -R a+r /etc/msgs
chmod a+x /etc/msgs

if [ ! -f /usr/sbin/in.ftpd.orig ]; then
        mv /usr/sbin/in.ftpd /usr/sbin/in.ftpd.orig
fi
cp /export/local/etc/ftpd /usr/sbin/in.ftpd
```

Este código puede ser introducido en un archivo de texto ó bien se puede obtener en la siguiente dirección.

ftp ://ftp.math.fsu.edu/pub/solaris/ftp.anon

Una vez que se tenga el archivo en el servidor, deben colocarse los atributos para ejecución del mismo mediante la instrucción.

chmod u+x ftp.anon

Habiendo cambiado el archivo a modo de ejecución, únicamente será necesario teclear su nombre y oprimir Enter.

ftp.anon

Después de la ejecución del script se procede a colocar los archivos a disposición de los usuarios del servicio en el directorio pub, verificando que la propiedad de los mismos sea del usuario root.

Es muy importante que se pruebe el servicio de ftp y se verifique su seguridad, antes de darlo a conocer.

4.4.3 Trivial File Transfer Protocol (TFTP) (UDP puerto 69)

TFTP es un programa de transferencia de archivos basado en UDP, el cual no provee seguridad. Exististe una serie de archivos que TFTP puede transmitir desde la computadora, lo único que TFTP necesita para transmitirlos es que le sean requeridos. El uso principal de TFTP es el de inicializar terminales que no cuentan con unidades de disco. Se recomienda que de no tener terminales sin unidades de disco no se habilite este servicio.

Para deshabilitarlo bastará con colocar un carácter # al inicio de su entrada en el archivo inetd.conf de la siguiente forma, y reiniciar el servidor.

```
#tftp    dgram  udp     wait     root     /usr/sbin/in.tftpd          in.tftpd -s /tftpboot
```

4.4.4 Finger (TCP puerto 79)

El servicio **finger** tiene un doble propósito:

- Si se ejecuta **finger** sin argumentos, el programa despliega el nombre de usuario, nombre completo, localización y hora de entrada de cada usuario conectado actualmente al servidor (Parte de la información es obtenida del archivo /etc/passwd)

- Si se ejcuta **finger** con un nombre como argumento, el programa busca en el archivo /etc/passwd e imprime la información de los usuarios que coincidan con el argumento en el nombre de usuario, o nombre completo.

Generalmente **finger** se ejecuta en el sistema local, pero puede ser utilizado también para buscar usuarios en sistemas remotos. Por ejemplo si se deseara saber los usuarios conectados al sistema pangea.stanford.edu se teclearía los siguiente:

finger @pangea.stanford.edu

O bien para buscar a un usuario específico llamado edgarr:

finger edgarr@pangea.stanford.edu

Como se puede observar **finger** ofrece muchas facilidades para obtener información, lo cual constituye un grave problema de seguridad. De esta forma cualquier persona podría obtener una lista de los usuarios del sistema, llenando los sistemas de correo electrónico de información publicitaria, o tal vez intentar irrumpir mediante los nombres de usuarios encontrados.

Es decisión del administrador de la red deshabilitar este servicio. Existen dos formas de realizarlo que son:

- Colocar un símbolo de comentario (#) al inicio de su entrada en el archivo inetd.conf o eliminarla. Este cambio producirá que el cualquier persona que intente pedir una conexión finger desde otro sistema obtenga un mensaje de "Connection refused error". Esta forma produce un aislamiento total, pero efectivo.

- Remplazar el programa finger con un script que imprima un mensaje con instrucciones, para que los interesados puedan contactar al administrador del sistema o a la institución. Un ejemplo de script a utilizar es el siguiente:

```
# !/bin/sh
#
/bin/cat << 'XX'

Bienvenido a la Universidad Nacional Autónoma de México
Facultad de Estudios Superiores Cuautitlán
Unidad de Servicios Médicos
Para contactar a alguno de los usuarios favor de llamar al
858-969-874 en México, D.F. ó enviar un correo electrónico a
administrador@servidor.unam.mx
```

Guardar el script en un archivo , por ejemplo /usr/sbin/informa. Y colocarle atributos de ejecución.

```
chmod u+x /usr/sbin/informa
```

Remplazar la entrada in.fingerd del archivo /etc/inetd.conf con el nombre del archivo ejecutable.

```
finger  stream  tcp     nowait  nobody /usr/sbin/informa     informa
```

Después de realizar los cambios será necesario reinicializar el servidor.

4.4.5 Telnet (TCP puerto 23)

El servicio Telnet permite la conexión de terminales remotas, también llamadas "terminales virtuales", ya que aunque físicamente no se encuentran conectadas a la red local, pueden trabajar como si lo estuvieran. Para configurar el servicio de telnet bastará con que sus entradas correspondientes sean establecidas en los archivos /etc/inetd.conf y /etc/services. (Establecidas por omisión al momento de instalar Solaris).

La entrada de /etc/inetd.conf será:

```
telnet  stream  tcp     nowait  root    /usr/sbin/in.telnetd     in.telnetd
```

La correspondiente en /etc/services es:

telnet 23/tcp

Algunas implicaciones de seguridad de telnet son:

- En redes como Ethernet, los paquetes en una transmisión son enviados hacia todos los nodos, siendo el nodo indicado el que los toma. Mediante el uso de programas llamados interceptores (sniffers), se pueden interceptar los paquetes destinados a cierta dirección, pudiendo descifrar el nombre del usuario y su clave de acceso en una transmisión, por tal motivo se recomienda no utilizar la clave de supervisor de manera remota.
- Si se tiene clave de acceso o administración en más de un equipo, se deberá utilizar claves de acceso distintas, las cuales deberán ser cambiadas constantemente para minimizar los riesgos.

Para establecer una sesión de telnet desde Solaris, simplemente se necesita teclear telnet y la dirección del equipo remoto. Al ingresar el sistema solicitara el nombre del usuario o login, así como la clave de acceso. Ejemplo (fig. 21):

```
telnet 132.250.102.6
Trying...
Conected to 132.250.102.6
Escape carácter is ' ^]'

4.3 BSD UNIX

login: lolo
passwd:
```

figura 21. Telnet

Por razones de seguridad al teclear el password, el texto no es visible.

4.4.6 Talk (UDP puerto 517)

El servicio **talk** permite establecer una comunicación bidireccional entre dos usuarios de la red.

Su entrada en /etc/inted.conf es:

```
talk    dgram  udp    wait    root    /usr/sbin/in.talkd        in.talkd
```

La correspondiente entrada en /etc/services queda de la siguiente manera:

```
talk    517/udp
```

Una vez instalado el servicio **talk** se ejecuta simplemente tecleando la palabra **talk** y la dirección electrónica del usuario a contactar (nombre del usuario@dirección IP). El usuario destino recibirá un mensaje con la dirección electrónica del emisor, solicitando establezca la conexión, a lo cual deberá responder de igual manera con el comando **talk** seguido de la dirección electrónica del usuario que solicita la conexión

Una vez establecida la comunicación, la pantalla de cada uno de los usuarios se divide en dos secciones mostrando en cada una de ellas los mensajes de cada usuario. (fig. 22)

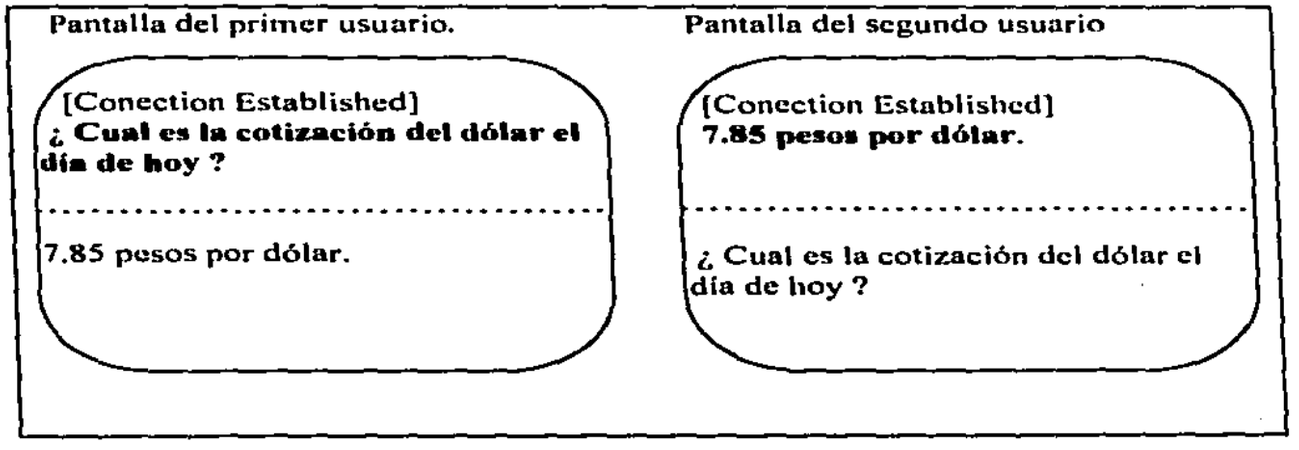


figura 22. Comunicación con talk.

La manera en que un usuario puede desactivar la recepción de solicitudes de comunicación, es mediante el comando **mesg n.** Esta instrucción puede ser activada desde el teclado o introducida directamente en el archivo .login del usuario. Para restablecer la recepción de mensajes se utiliza el comando **mesg y**. La instrucción **mesg n** no tiene efecto cuando quien envía un mensaje o solicitud de comunicación es el superusuario.

4.5 Recomendaciones finales

Hasta aquí se han dado los pasos necesarios para el establecimiento de los servicios de TCP/IP. A partir de este punto, se sugiere se tomen las siguientes consideraciones:

1. Si se monta un servidor con información con un grado alto de confidencialidad, evitar establecer servicios públicos en él.

2. Monitorear regularmente el sistema para determinar los usuarios y procesos activos (comandos **who** y **ps**).

3. Procurar no instalar software de fuentes no confiables o desconocidas en el servidor, de ser posible probarlo previamente en alguna terminal, equipo local, o bien respaldar el sistema antes de su instalación.

4. No utilizar claves de acceso de superusuario desde una terminal, es decir limitar su uso a la consola del servidor.

5. En el caso que se creen cuentas para usuarios, establecer los límites de espacio para las mismas (quotas).

6. Bloquear el acceso con una cuenta de usuario desde varios puntos al mismo tiempo.

7. Evitar el acceso a la consola a usuarios no autorizados, estableciendo políticas para su uso.

8. En el caso de los servidores de FTP anónimo, no colocar software que no sea de dominio público o sin consentimiento del los propietarios y/o autores, así como vigilar las posibles violaciones a derechos de autor.

9. Elaborar y ejecutar programas de mantenimiento para la verificación de la seguridad y rendimiento del sistema, cumpliendo por lo menos con los siguientes requerimientos:

   - Cambio de contraseña de root.
   - Revisión del espacio utilizado por los usuarios (comando **du**).
   - Eliminación de colas de impresión perdidas.
   - Depuración de los archivos de mensajes y accesos contenidos en el directorio /var/adm. Estos archivos son de texto y contienen mensajes del sistema.
   - Verificación de la bitácora de accesos (comando **last**).
   - Respaldo del sistema.
   - Borrado de archivos y aplicaciones innecesarias.
   - Revisión de la alimentación de corriente eléctrica y los cambios en la instalación.

   Estas recomendaciones son dirigidas a evitar:

   - Caídas del sistema.
   - Disminución en el rendimiento del equipo.
   - Desperdicio de recursos.
   - Acceso a usuarios no autorizados.
   - Problemas de tipo legal.

Conclusiones

Es innegable el crecimiento exponencial de los servicios de red y su importancia en la sociedad actual. Dentro de la también creciente comunidad de Internet, se manifiesta un cambio necesario donde cada vez son más los usuarios que pasan de una fase de meramente consumidores a productores de servicios de red. Merced de este cambio, Internet se vislumbra como un medio más de comunicación, un espacio abierto donde consumidores y productores se conjugan en el mismo ambiente.

Como se puede observar en el presente trabajo la configuración de un servidor tiene mayores complicaciones que la de un equipo personal. Es en este punto donde resalta la función de un administrador de red, ya que el trabajo no concluye con la carga de los servicios, se requiere de un proceso adicional de revisión y control por parte de la administración.

Si bien la conexión a red proporciona muchas facilidades abre también múltiples accesos a la comunidad hacia nuestra información constituyendo un asunto de seguridad que se debe tener en consideración.

La información presentada en la presente investigación muestra la configuración de servicios básicos y conexión dentro de Solaris, al ser este una implementación de UNIX, el proceso en otros sistemas (Linux, HP-UX, IRIX,etc.) es similar y mucho de lo aquí expuesto se puede aplicar por analogía en ellos.

Adicionalmente será necesario que antes de montar un servicio hacia Internet, se realice un análisis de los servicios a cargar y la información que contendrán, lo cual permitirá un mejor aprovechamiento de los recursos y contribuirá a la integración de una red global más robusta y útil.

Bibliografía

Cypser, R.J Comunications for cooperating systems: OSI, SNA and TCP/IP. New York: Addison Wesley. 1992. 743 p.

Feit , Sidnie. TCP/IP: architecture, protocols and implementation. New York: McGraw Hill. 1993. 466 p. (Computer Comunications)

Frisch, Aeleen. Essential System Administration. 2 ed. . Sebastopol, California: O'Reilly. 1995. 758 p.

Gardner, James. Learning UNIX. 2 ed. Indianapolis,Indiana : Sams.1994. 646 p.

Garfinkel, Simpson. Practical Unix & Internet Security. Sebastopol, California: O'Reilly. 1996. 971 p.

Gray, Matthew. Internet Growth. 1996. http://www.mit.edu :8001/people/mkgray/net/internet-growth-raw-data.html

Henry, S. Lee. Solaris 2.x system administrator's guide. New York : McGraw-Hill. 1995. sp.

Hunt, Craig. TCP/IP Network Administration. Sebastopol, California: O'Reilly. 1994. 472 p.

InterNIC. About the Internet . http ://www.internic.net/ds/about.html. 17/09/97.

Liu, Criket. et. al. Managing Internet information services. Sebastopol, California: O'Reilly. 1994. 630 p.

Norton, Peter. Periféricos y accesorios para la IBM, PC,PS/2 y compatibles. México: Prentice Hall. 1994. 552 p.

Pike, Mary Ann. Using the Internet. 2 ed. Estados Unidos: Prentice Hall. 1991. 1240 p.

Quarterman, J. What is the Internet Anyway ?. Network Working Group. Abril 1996. RFC 1935. http://www.es.net/pub/rfc/rfc1935.txt

Savetz, Kevin M Your Internet Consultant. Indianapolis,Indiana : SAMS. 1994. 550 p.

Sun Microsystems. Solaris FAQ. http://www.sun.com/smcc/solaris-migration/tools/docs/FAQ/solaris2.html. Palo Alto, California. 29/09/97

Sun Microsystems. TCP/IP : TCP/IP protocol is the vehicle of choice for transfering data between database client and servers . Palo Alto, Califonia. 22/09/97. 14 p.

Tanembau, Andrew S. Computer Networks. 2 ed. Englewood Cliffs, New Jersey : Prentice Hall . 1991. 658 p.

Winsor, Janice. Solaris advanced system administrator's guide. Emerville, California : Ziff-Davis, 1993. 447 p.