105

UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

FACULTAD DE INGENIERIA

Seguridad en Redes TCP/IP (dentro de la RedUNAM)

Tesis que para obtener el título de:

INGENIERO EN COMPUTACIÓN

PRESENTAN:

Silvia Iliana Ramírez Ramírez

Fabiola Malinali Sánchez Morgan

Director de Tesis:

Ing. Michael De Leo Gayol

México D.F. 1996

TESIS

COMPLETA

Gracias a Dios y a la vida, por más que una oportunidad.

A mi papá, por los buenos momentos por su apoyo y cariño, Mamita gracias por ser lo mejor que le puede pasar a cualquier hijo en el mundo esto es muy poco de lo que tú mereces.

A mi hermano Mario, y a mis hermanitas Ivet y Flor por ser amigas y cómplices.

Por enseñarme sobre el amor y el cariño sin interés, a mi mamá Flor y a mi papá Lucío que sé que estará muy orgulloso desde donde me vea.

Esta tesis es especialmente suya, mi segunda familia y familia de tantos jóvenes mexicanos, ustedes son una segunda mano para ayudarnos a quitar las piedras (a veces rocas) del camino, por todo su amor, gracias FLAG (Fundación Lorena Alejandra Gallardo).

A Mike que nos tuvo mucha paciencia, gracias por compartir con nosotros de lo mucho que tú sabes. Gracias también por ayudarnos a dar un gran paso hacia la madurez.

Gracias, podría ser por... ¡Por tantas cosas!.. pero será simplemente por amarme: Manuelito esto también es para tí.

A mis sobrinos y a los hijos que tal vez algún día tenga.

A mis tios y primos.

A la Universidad y a la mejor escuela de Ingeniería: la Facultad de Ingeniería.

Al Banco de México.

Con mucho cariño y nostalgia, por todo lo que de tí recibí: amigos, cariño, y no una llave , sino un llavero para abrirme puertas, gracias CECAFI.

Por que no sólo con ustedes compartí la mejor de las juventudes, sino por que me enseñan lo que significa contar con alguien, mis amigos de la Facultad (burbles). No pongo nombres por que hay que economizar papel.

A mis amigos del CCH, en especial a Alfonso y a Mónica.

A mis amigochas Cris, Yolis, Pamela y Judith.

A mis buenos amigos del Banco de México, qué suerte que estén ahí.

A todas las personas que de una u otra manera colaboraron en este trabajo, en especial a Eduardo Jallath.

A los que me faltaron, que no fue por menos cariño o agradecimiento que no los pusiera sino por que las dedicatorias es lo que uno escribe un día antes de mandar a imprimir la tesis.

Iliana

El presente trabajo representa un esfuerzo constante, el cual finaliza una etapa como estudiante e inicia otra como profesionista. Durante este camino, muchas personas han sido parte fundamental en mi vida, por lo que me permito nombrar a algunas de ellas:

En primer lugar al Ing. Michael de Leo Gayol, quien me abrió la puerta hacia el campo profesional, me guió, y enseño gran parte de los conocimientos que hoy aplico; también por el apoyo en la dirección de este trabajo,

A Iliana, por su amistad, apoyo; empeño y conocimientos reflejados en el trabajo que hoy terminamos,

Especialmente a mi Padre, quien ha sido el más claro ejemplo de trabajo, rectitud, amor y dedicación, a quien dedico esta tesis, ya que sin su apoyo no hubiera podido lograr ninguna de las metas que he cumplido.

A mi Madre, por su amor, comprensión, ternura y paciencia.

A mis hermanos: Paty, Alicia Adolfo y Arturo, por su amor, comprensión y por todo lo que me han permitido disfrutar y compartir con ellos.

A mi tío Lorenzo, quien me brindo su paciencia y conocimiento en el difícil camino de la Ciencia.

A mamá Lupita, mis tíos, tías y primos, por su amor y apoyo.

A Adrián, por su amor, apoyo, y el tiempo que hemos compartido juntos.

A mis compañeros y amigos de la Facultad de Ingeniería, especialmente a Yuri, Hildia, Nadia, Caty, Consuelo, Paty, Samara, FernandoV., Pedro, Diego y Miguel.

A Eduardo P., Oscar V. y Gabriel, por su cariño y apoyo.

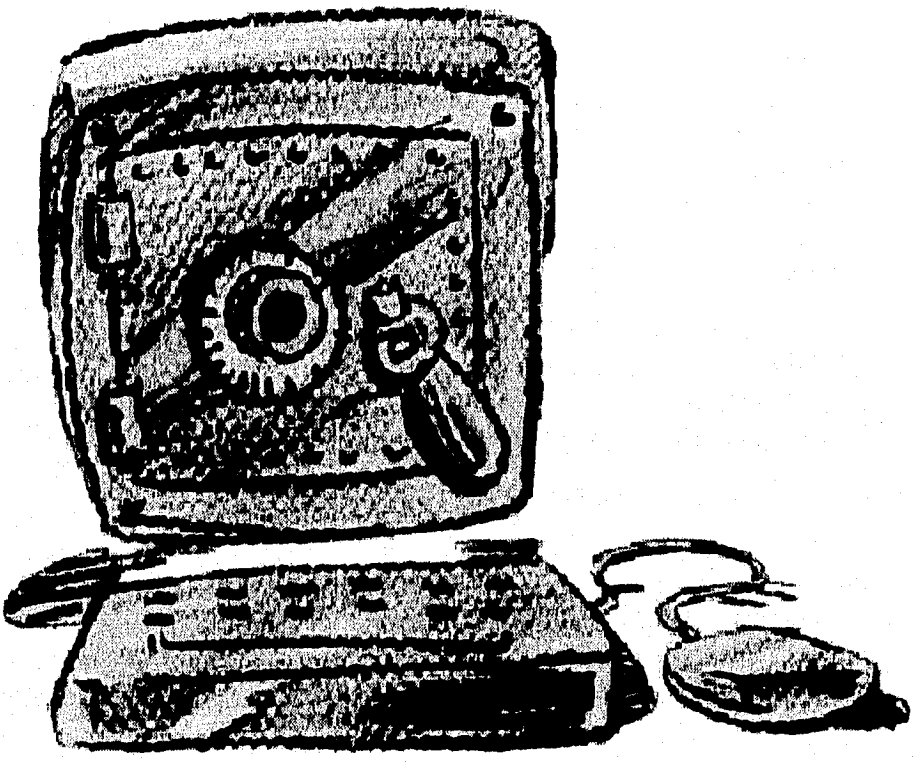
A Erick H., por su tiempo y paciencia,

A todos y cada uno de ellos: GRACIAS, LOS QUIERO.

Fabiola Malinali Sánchez Morgan.

Índice

Capítulo 1



Introducción

En 1982[1], el TCP/IP de la Internet incluía unas cuantas computadoras en dos docenas de sitios concentrados principalmente en Estados Unidos. Para el verano de 1992, arriba de 700,000 sistemas de computadoras se ligaban a la Internet en 39 países a

[1]Internetworking with TCP/IP Volume III. *Comer, Douglas E. and Stevens, David L.* Prentice Hall, New Jersey, 1993. p.1-2.

través de siete continentes, y su tamaño se duplica cada diez meses. Aproximadamente un tercio de las 4500 redes que comprendían la Internet en 1992 se localizaban fuera de los E.U.

Además, las más grandes corporaciones han elegido a los protocolos TCP/IP para sus redes privadas, muchas de las cuáles son tan grandes como lo fué hace diez años la Internet. Noventa por ciento de todos los sistemas UNIX usan TCP/IP como protocolo de transporte nativo.

Detrás del crecimiento cuantitativo, la pasada década ha sido testigo de un importante cambio en los lugares que emplean TCP/IP. Al principio su uso de enfocaba en unos cuantos servicios como el correo electrónico, la transferencia de archivos y la conexión remota. Actualmente más usuarios están diseñando protocolos de aplicación y construyendo su propio software de aplicación. De hecho, un quinto del tráfico en la Internet está formado por aplicaciones privadas. Ellas enriquecen la funcionalidad de la Internet y han habilitado a nuevos grupos de usuarios para beneficiarse de la conectividad.

Los principales productores[ii] de sistemas como IBM, Digital Equipment Corp. y Hewlett Packard empezaron a instalar TCP/IP en sus sistemas hace ya varios años como alternativas a los protocolos de red propios. Estas empresas también dependen de la rutina TCP/IP para enlazar sus operaciones mundiales en redes multiprotocolos.

TCP/IP es sinónimo de conectividad universal.

La variedad de aplicaciones usando TCP/IP está metiéndose a todas partes: incluyendo sistemas de reservación de hoteles, aplicaciones que monitorean y controlan plataformas petroleras, diferentes sistemas de control, sistemas o distribución de información económica, aplicaciones que permiten distribuir geográficamente archivos, compartir imágenes, transferirlas, imprimirlas, así como sistemas de teleconferencia y multimedia.

Este es un tiempo de increíble excitación. En unas cuantas décadas la computación ha crecido aceleradamente. Descendiendo los costos.

Sin embargo, detrás de todos estos avances aparece una terrible sombra: la existencia de personas que buscan debilidades en la seguridad de las cuales sacar provecho, o dañar sólo por diversión.

Esto nos lleva a pensar en la necesidad de seguridad, de resguardar nuestros recursos, y sobre todo nuestro más importante valor: *la información.*

Existe una gran cantidad de sucesos que nos recuerdan que es real el ataque a los sistemas, como el famoso gusano que, en 1988, invadió la red del Departamento de Comunicaciones de la Defensa de los Estados Unidos, el cual se multiplicaba así mismo

[ii]TCP/IP está derribando fronteras entre usuarios PC Journal, México, 1992 p.19-20

en cientos de computadoras. Cuántas historias reales habremos escuchado de personas que, abusando de su conocimiento dentro de sus empresas, abusaron para obtener provecho interceptando y modificando la información (generalmente en entidades bancarias), o de jóvenes que en su afán de investigación se filtran en lugares prohibidos, como el caso de 1983[iii]en el que unos jóvenes entraron a bases de datos de un hospital en Estados Unidos. Nuestra Universidad desde luego que no es la excepción, ya se han dado casos de jóvenes que logran burlar los sistemas de diferentes instituciones, incluso la de la supercomputadora CRAY, afortunadamente fue detectado pero tuvo la oportunidad de hacer grandes daños.

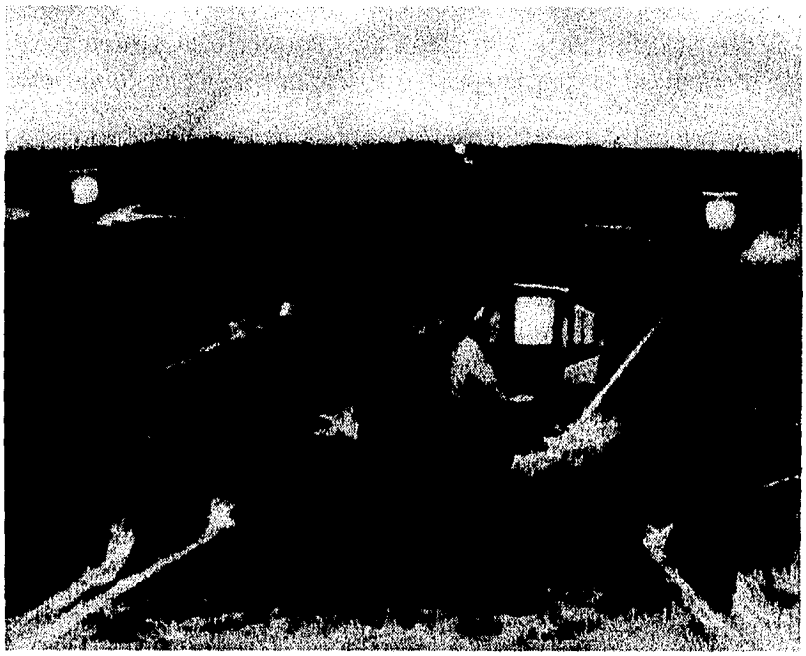
De aquí la necesidad de plantearnos con qué y cómo defendernos. Cuáles son las posibilidades con qué contamos actualmente. Qué tipo de seguridad requiere nuestro sistema y en base a qué podemos plantear un esquema de seguridad.

Existen diferentes niveles de seguridad, dependiendo de los recursos, herramientas y cómo son aplicados.

Esperamos que el siguiente trabajo contribuya proporcionando los más importantes conceptos de seguridad manejados en una red que utiliza TCP/IP, así como una visión general de las posibles soluciones y recursos que actualmente se encuentran disponibles, para que, tomándolos como base, aquél que lo requiera, determine su conjunto de posibilidades, profundice en la que considere y la aplique.

[iii]Computer Security Handbook. *Richard Baker* Mc Graw Hill, Second Edition 1991. p. xvii

CAPÍTULO 2



Teoría de Seguridad en Redes

2.1. ¿Qué es Seguridad en Cómputo?

La anterior es la primer pregunta que debemos responder para el desarrollo de esta tesis, si bien un término como Seguridad tiene más de un significado -aún los profesionales que trabajan en el área de Seguridad en cómputo no se han puesto de acuerdo en lo que este término significa- por lo que se intentará dar un concepto de acuerdo a lo que para nuestras necesidades es útil.

Una versión acerca de un sistema completamente seguro se la atribuye a Gene Spafford, podría juzgarse de cómica pero nos refleja cuán difícil puede ser mantener un sistema confiable y seguro:[iv]

"El único sistema que es completamente seguro es aquel que está apagado, desconectado, guarnecido en una caja fuerte de titanio, enterrado en una caja de concreto, rodeado de gas irritante y por unos guardias altamente armados (aún así yo no arriesgaría mi vida en él)".

Como una definición más cercana a nuestras necesidades, encontramos que según Garfinkel la seguridad en cómputo puede definirse en términos de lo siguiente:[v]

"Una computadora es segura si uno puede depender de ésta y su software, y si éstos funcionan como uno espera que lo hagan".

Si uno espera que los datos que hoy dejó en la máquina, en unas semanas sigan ahí sin que alguien más los haya leído, entonces la máquina es segura, a este concepto también se le conoce como "confiabilidad" (trust).

Esto nos lleva a identificar a la seguridad como protección, ahora nos falta específicar qué es lo que queremos proteger y de qué. Son tres las áreas de protección que nos interesan: software, hardware y los datos, de qué los queremos proteger es[vi]:

2.1.1 Hardware

- Protección de destrucción de hardware valioso

2.1.2 Software

- Protección de destrucción de programas valiosos

2.1.3 Datos

- Protección de destrucción de datos valiosos

Y para las tres áreas (Hardware, Software y Datos):

- Protección a cambios no autorizados
- Protección a uso no autorizado

[iv]Garfinkel, Simson and Spafford, Gene. Practical UNIX Security. Ed. O'Reilly & Associates, Inc. USA, 1994.

[v]Ibid, p.4

[vi]N. Derek Arnold. *UNIX Security a practical Tutorial.* McGrawHill 1993 págs.10 y 11

El siguiente paso es definir los diferenes tipos de seguridad, qué puede afectarla, los criterios para evaluar un sistema, así como otros conceptos que nos servirán para tener las bases para los siguientes capítulos.

2.2. Definiciones de seguridad.[vii]

2.2.1 Seguridad en la Información (INFOSEC).

La protección en el proceso de la información se requiere por que la información puede ser comprometida por ignorancia, inadvertencia, accidentalmente o por malicia.

2.2.2 Seguridad en Cómputo (COMPUSEC).

El sentido general de "seguridad en cómputo" (COMPUSEC), en este contexto será expuesto como el estado de certeza de que los datos computarizados y los archivos de programas no pueden ser accesados, obtenidos o alterados por personas no autorizadas.

2.2.3 Seguridad en los datos.

Consiste de procedimientos y acciones diseñados para prevenir la revelación no autorizada, transferencia, modificación o destrucción, accidental o intencional de los datos. Hoy en día el término de datos debe ser cambiado por información, simplemente por que cada vez más el tráfico de la red consiste de información en general, más que datos (incluyendo imágenes, FAX, video, etc.).

2.2.4 Seguridad en Comunicaciones (COMSEC).

En los años recientes, el tema ha tomado la dirección de que la información "segura" fluye en redes y líneas de comunicación "seguras". La seguridad en comunicaciones -COMSEC es, por lo tanto, la protección como resultado de la aplicación de "criptoseguridad", seguridad en la transmisión, la emisión de medidas de seguridad a telecomunicaciones, y de la aplicación de medidas de seguridad física a la información que se transmite por los distintos medios de comunicación.

Como se puede notar existen algunos conceptos que hacen falta aclarar para completar la explicación:

[vii] Madron, Thomas W. *Network Security in the 90's.* Issues and Solutions for Managers. Ed. Wiley Professional Computing. U.S.A. 1992. p.3-24.

2.2.4.1 Criptoseguridad

Es el componente de seguridad en comunicaciones -COMSEC- que resulta de la aplicación de criptosistemas (métodos mediante los cuáles la información se vuelve indescifrable sin una llave), técnicamente hablando, y su uso.

2.2.4.2 Seguridad en la transmisión (TRANSEC).

Es el componente de COMSEC que resulta de todas las medidas destinadas a proteger los transmisores de intercepciones y exploraciones no autorizadas.

2.2.4.3 Emisión de la Seguridad (EMSEC).

El componente de COMSEC que resulta de todas las medidas tomadas para negar el acceso a personas no autorizadas el acceso a información valiosa, que podría ser obtenida de interceptar las emanaciones de equipo de encriptamiento y sistemas de telecomunicaciones.

2.2.4.4 Seguridad Física.

El componente de COMSEC que resulta de todas las medidas físicas necesarias para salvaguardar equipo clasificado, material y documentos de acceso u observación por personas no autorizadas.

2.2.4.5 Sistemas de seguridad

Consisten de la combinación de subsistemas de hardware y software. Seguridad en equipo de comunicaciones. Por ejemplo, equipo diseñado para proveer seguridad a telecomunicaciones, para convertir la información a una forma ininteligible a un interceptor no autorizado, y posteriormente reconvertir ésta información a su forma original para los receptores autorizados; tanto como equipo diseñado específicamente para obtener ayuda o como sólo un elemento en el equipo de conversión.

2.2.5 Seguridad a través de la oscuridad.[viii]

El concepto de seguridad, derivado en gran medida de la inteligencia militar, se basa en la "necesidad de saber" (need to know). La información es particionada y se da a conocer tanto como sea necesario para realizar el trabajo.

En ambientes de trabajo donde puntos específicos de la información son susceptibles, o donde la seguridad inferencial es importante, esta política se vuelve consideradamente importante. Si tres piezas de información juntas pueden tornarse un punto vulnerable del sistema y uno no tiene acceso a más de dos, se puede asegurar la información.

[viii] Garfinkel, Simson and Spafford, Gene. Practical UNIX Security. Ed. O'Reilly & Associates, Inc. USA, 1994. p.15-17

En ambientes de operación de las computadoras, aplicar el mismo concepto, no es usualmente apropiado especialmente si se basa la seguridad en el hecho de que algo es desconocido a los atacantes. Este concepto puede mas que preservar, hacer daño a la seguridad.

Considerar un ambiente donde los administradores deciden mantener lejos de los usuarios los manuales para impedir que aprendan acerca de los comandos y las opciones del sistema es absurdo. Bajo estas circunstancias el administrador podría pensar que esta incrementando el nivel de seguridad en el sistema, pero probablemente no. Para muchos posibles atacantes al sistema, puede ser relativamente fácil conseguir este tipo de información en otras partes. Muchos vendedores sacan copias a su documentación sin requerir una licencia de ejecución. Usualmente para lograr esto sólo se requiere visitar una universidad, localizar el material y fotocopiarlo.

Mientras tanto los usuarios locales se vuelven menos eficientes para manipular la máquina, porque no les es permitido aprender sobre los comandos que les permitirían incrementar su nivel de eficiencia. Este tipo de personas son comúnmente quienes tienen una pobre actitud, porque el mensaje implícito del administrador es: "no estamos completamente seguros de que seas un usuario responsable". Además si un usuario no está abusando de los comandos y de las características del sistema, esto implica que el administrador no tiene la suficiente capacidad para tratar el problema.

Mantener algoritmos en secreto, tanto como desarrollar algoritmos de ciframiento, son valores cuestionables. Al menos que sea un experto en criptografía, es poco probable que se analice la consistencia del algoritmo. El resultado puede ser un mecanismo que tiene "un hoyo muy abierto", en cuanto a seguridad se refiere. Un algoritmo que se mantiene en secreto no es estudiado por otros, y de esta forma paradójicamente si alguien descubre el hoyo en el sistema, tiene libre acceso a los datos sin el conocimiento de ello a los administradores o al dueño de la información.

De la misma forma, mantener el código fuente del sistema operativo en secreto no es una garantía de seguridad, quienes están predispuestos a entrar y romper el sistema, encontrarán hoyos de seguridad -con o sin el código fuente. Pero sin el código fuente no es posible realizar una examinación sistemática del programa para encontrar y solucionar problemas.

2.3. Agujeros en la seguridad[ix]

Los *"agujeros"* en la seguridad (puntos vulnerables) se manifiestan de cuatro formas:

1. *Agujeros* de Seguridad en la parte física.

[ix] N. Derek Arnold. *UNIX Security a practical Tutorial.* McGrawHill 1993 págs.10 y 11

En estos el problema básicamente consiste en dar acceso a la parte física de la máquina a personas no autorizadas. Ejemplos de esto son aquellos centros donde se tienen estaciones de trabajo y para un usuario puede resultar trivial reinicializar una máquina en modo monousuario (single user) y alterar el almacenamiento de archivos, o bien no restringir el acceso a respaldos de cintas confidenciales, que pueden ser leídas por cualquier usuario con acceso al manejador de cintas.

2. *Agujeros* de Seguridad en Software.

Estos son básicamente software mal desarrollado, que debido a esto tiene altos privilegios que permiten acceder información confidencial.

3. *Agujeros* de Seguridad por incompatibilidad de uso.

En ocasiones debido a la falta de experiencia del administrador del sistema, ensambla una combinación de hardware con software que son útiles, pero desde el punto de vista de seguridad presentan un agujero, esto es conectar dos cosas incompatibles que aunque funcionen, son vulnerables en seguridad.

4. En no elegir una idónea filosofía de seguridad y mantenerla.

El cuarto tipo es percepción y entendimiento. Un software perfecto, hardware protegido, y compatibilidad de los componentes no trabajan a menos que se seleccione y aplique una política de seguridad adecuada. Tener el mejor mecanismo del mundo para asignar contraseñas (passwords), está por demás si los usuarios asignan el nombre de su clave como contraseña.

La seguridad es relativa a la política, o conjunto de políticas y a la función del sistema conforme a ese conjunto de políticas.

2.3.1 Los atacantes

Este es uno de los puntos más importantes, identificar quiénes pueden ser nuestros "enemigos" y sus causas.

A diferencia de lo que muchos pueden creer el mayor número de atacantes en las diferentes redes han sido personas de la misma empresa: administradores resentidos, vengativos, empleados que buscan algún beneficio propio ($), y los menos -pero no por eso menos peligrosos- son aquéllos que por curiosidad o para probar sus capacidades y la de su objetivo se entrometen en nuestro sistema, éstos últimos son los más famosos y existen diferentes acepciones para ellos, a veces llamados hackers y en otras crackers, aunque difieren las definiciones que podemos hallar a continuación pondremos las más difundidas:

2.3.1.1 ¿Qué es un Hacker?[x]

1.- Una persona que aprende los detalles de los sistemas de cómputo y como extender sus capacidades -opuesto a la mayoría de los usuarios, quienes prefieren aprender el mínimo necesario.

2.- Alguien que programa entusiasmadamente o quien se divierte programando, en lugar de revisar la teoría acerca de programar.

2.3.1.2 Hacker (según James Arlin Cooper)[xi]

"Individuo que persistentemente explora computadoras y redes para aprender como pueden ser utilizadas. Actualmente el término se aplica a aquellos que tratan de burlar las barreras de seguridad de la computadora y de la red, la mayoría de las veces como un desafío."

La diferencia básica que hay entre un hacker y un cracker es la intención para el primero de aprender y para el segundo de dañar, no por esto vamos a justificar al hacker que de cualquier forma está violando nuestra seguridad.

Aunque éstos atacantes son muy peligrosos, no debemos olvidar a los primeramente mencionados y más dañinos por su propia naturaleza. La mayoría de las perdidas ocurridas en empresas u organizaciones cada año, es el resultado de errores humanos, accidentes y omisiones, y de esto se deriva que las más de las veces éstos son ocasionados por personal de la empresa o gente que colabora en la organización, y una minoría de estas pérdidas es ocasionada por personas ajenas al lugar donde suceden las mismas. Una persona que tiene por su labor acceso a nuestro sistema, un usuario, no tiene que evadir las barreras que un hacker ni las lógicas ni las físicas, así que le es más fácil destruir u obtener provecho de la información o del equipo, existen cientos de casos de gentes que han violado la seguridad, realizando fraudes, haciéndose ricos, obteniendo información clasificada o vengándose de alguna actitud tomada hacia él, y estas personas estaban o habían laborado en la empresa.

2.4. Servicios de Seguridad.[xii]

Existen diferentes tipos de servicios sobre seguridad en cómputo. Es importante que ambos administradores y usuarios, conozcan los servicios de seguridad que los profesionales en la materia manejan (y que los usuarios esperan).

---

[x] Guy, L. Steele, *The Hacker Dictionary.*

[xi] Cooper, James Arlin. *Computer & Communications Security.* McGraw Hill Publishing N.Y., 1989 pág. 375

[xii] Lynch, Daniel C. and Rose, Marshall T. *Internet System Handbook.* Addison Wesley Publishing Company, INC., 1993

2.4.1 Privacidad o Confidencialidad.

Es proteger la información para que ésta no sea leída por alguien que no ha sido autorizado explícitamente. Esta protección no sólo abarca información conjunta (total), si no fragmentos de la misma que pueden parecer inofensivas por sí mismas, pero pueden ayudar a inducir a información confidencial.

2.4.2 Autenticación

En este servicio debe considerarse una doble autenticación: el origen de los datos y la entidad en comunicación. La autenticación del origen puede definirse como: "la corroboración de que el originador de los datos recibidos es quien pretende ser" y la autenticación de la entidad en comunicación es verificar que el otro extremo en la comunicación es el que se esperaba.

2.4.3 Integridad de los datos.

Protección a la información (incluyendo programas) de ser borrada o alterada, sin autorización del dueño de la misma. La información a proteger también incluye registros de contabilidad, respaldo en cintas, tiempo de creación de los archivos y documentación.

2.4.4 Consistencia.

Asegurarse de que el sistema funciona tal como los usuarios esperan que se comporte. No existe si el hardware o el software repentinamente empieza a comportarse de forma totalmente diferente a como solía comportarse, especialmente después de una actualización al sistema o de revisión de una falla.

2.4.5 Aislamiento o Control de Acceso.

Regularización del acceso al sistema. Controlar el acceso al sistema de individuos no autorizados o no conocidos; se deben determinar los siguientes puntos: cómo logro entrar, qué ha hecho y quién, o qué más tiene acceso al sistema. No debe confundirse el controlar el acceso con autenticar ya que se puede autenticar a un usuario para accesar el sistema y mediante el aislamiento se definen los permisos que tiene.

2.4.6 Revisión o No Repudio.

Así como los usuarios no autorizados deben ser controlados, los usuarios autorizados también cometen errores y aún más actos maliciosos. En este caso se debe determinar qué fue hecho, por quién y qué fue afectado. La única forma de obtener esta información es por medio de un registro incorruptible que registre todas las actividades del sistema y que sea capaz de identificar el actor y las acciones involucradas, de esta manera se evita que quien(es) esté(n) involucrados en la comunicación nieguen haber participado.

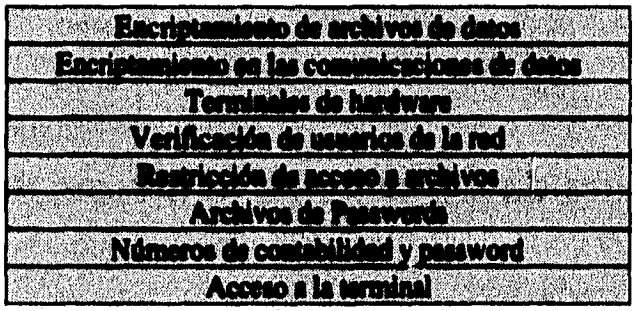
2.5. Un modelo de Seguridad en la Red.

A continuación se presenta un primer modelo de seguridad que provee una descripción de los puntos que se deben tratar en cuanto a seguridad en redes e integridad de los datos se refiere.

Davidson y White, sugieren un sistema de seguridad "como una serie de círculos concéntricos formando capas alrededor de los datos de la computadora. Fuera de estos círculos se representa el nivel mínimo de seguridad, y hacia dentro de los círculos se representa el máximo nivel de seguridad.

El problema de este modelo es que es demasiado simplista a la luz de la nueva tecnología. En vez de círculos se presenta a continuación un modelo integrado por capas, donde la capa superior representa el máximo nivel de seguridad, y la más baja el mínimo nivel de seguridad.

2.5.1 Mayor Seguridad



Menor Seguridad

Como se puede observar, hasta cierto punto este modelo está incompleto, por lo que a continuación se muestra una tabla que resume las partes en que se puede dividir el proceso de análisis para obtener un modelo de seguridad[xiii], cabe resaltar que no todas las partes caen en el contexto de esta tesis pero se muestran para describir el problema de forma integral. Los renglones de la tabla representan los pasos que se deben dar tomar en cuenta en la seguridad y las columnas los diferentes ambientes que se deben cuidar y que corresponden a cada paso:

[xiii] Cooper, James Arlin. *Computer & Communications Security* McGraw Hill Publishing N.Y., 1989 pág. 20

	Físico	Personal	Regulaciones	Hardware	Software	Redes
Entidades a vigilar	Hardware Software Redes Personal	Personal	Hardware Software Redes Personal	Computadoras Periféricos	Programas Datos	Redes
Amenazas	Del personal Naturales Accidentes	Del personal Naturales Accidentes	Del personal	Del personal Naturales Accidentes	Del personal Naturales Accidentes	Del personal Naturales Accidentes
Vulnerabilidades	Entradas Ambiente (físico) Fuentes de energía	Debilidades Ambiente de trabajo	Debilidades Incompletas	Ambiente Márgenes de diseño Desastres ambientales	Ambientales Control de acceso Cuidado de respaldos	Ambientales Control de acceso Cuidado de respaldos
Riesgos	Acceso Daños Destrucción	Venganza Corrupción	Multas Sanciones	Falla Daño Destrucción	Pérdida Alteración Divulgación	Falla Daño Destrucción
Medidas de protección	Candados Vigilancia Ambiente	Investigación Buen ambiente Vigilancia	Leyes Reforzamiento	Redundancia Margen de diseño Equipo para desastres	Control de acceso estructurado Encriptamiento	Control de acceso Encriptamiento

2.6. Estructura de los criterios (o niveles de seguridad)[xiv]

Para poder definir que tan confiable es un sistema, el Departamento de la Defensa de los Estados Unidos en su *Orange Book* indica los diferentes niveles de seguridad que se han establecido. Se enuncian las características de implementación que deben cumplir los sistemas para pertenecer a una o a otra división.

2.6.1 Introducción

Los criterios están clasificadosen cuatro divisiones D, C, B y A ordenados de manera jerárquica, estando la división más alta (A) reservada para sistemas que proveen mayor seguridad. Cada división representa una mejor implementación que se traduce en mayor confiabilidad en un sistema en cuanto a protección de información sensible (llamaremos información sensible a aquella que es factible a ser dañada, y que además tiene nuestra atención por su importancia). Dentro de las divisiones C y B existen subdivisiones llamadas clases, que a su vez son ordenadas de manera jerárquica (C2 a mejor implementación C1 a menor).

[xiv] Departament of Defense *Trusted Computer System Evaluation Criteria.* DOD 5200.28-STD 1985

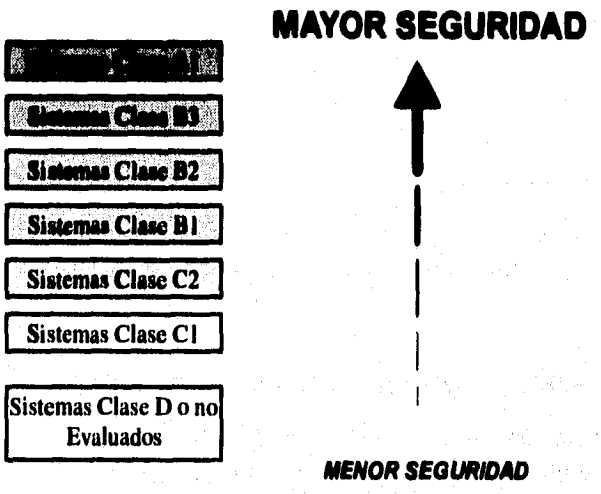


FIG.2.1 Niveles de Seguridad según el Orange Book.

Para garantizar el diseño e implementación de los sistemas se realizan pruebas sobre porciones relevantes para la seguridad del mismo sistema. Dichas porciones son referidas como Trusted Computing base (TCB). O como lo define James Arlin Cooper[xv]:

> *"La totalidad de mecanismos de seguridad de hardware y software en un sistema computacional."*

Esto es, un TCB es el conjunto de mecanismos de protección dentro de un sistema computacional -incluyendo hardware, firmware y software-, combinación responsable de reforzar las políticas de seguridad sobre un producto o sistema. La habilidad de un TCB para reforzar correctamente una política de seguridad depende únicamente de los mecanismos dentro del TCB y de las entradas correctas de parámetros introducidas por el personal administrador del sistema en relación con las políticas de seguridad.

A partir de pruebas al TCB se declara un sistema dentro de una clase u otra, estas pruebas principalmente verifican atributos de seguridad, especialmente su estructura de diseño e implementación.

Para cada criterio existe un grupo de requerimientos. Estas agrupaciones fueron desarrolladas para asegurar que los tres objetivos de control básicos sobre seguridad computacional sean satisfechos y no pasados por alto, entendiendo como objetivos de control a métodos que expresan metas de seguridad. Estos objetivos cubren:

[xv]Cooper, James Arlin. *Computer & communications Security.Strategies for the 1990's.* McGrawHill Communications Series, New York, 1989 pág. 386

- Políticas de Seguridad
- Contabilización
- Confianza

Los métodos deben proveer un esquema que permita desarrollar una estrategia que llene completamente un conjunto de requerimientos de seguridad en cualquier sistema dado.

2.6.2 Definiciones (tomadas del Orange Book)

A continuación explicaremos el significado de los objetivos y daremos algunas definiciones que pueden resultar útiles para aclararlos y para comprender la explicación sobre las clases y las subdivisiones que aparece posteriormente. El orden en que se presentan las definiciones va de acuerdo a su funcionalidad y tomando en cuenta que para entender las últimas hay que conocer el significado de las anteriores.

2.6.2.1 Integridad de los datos

Estado que existe cuando el dato computabilizado es el mismo que en el documento original y no ha sido expuesto a alteraciones accidentales o maliciosas o a la destrucción.

2.6.2.2 Objeto

Entidad pasiva que contiene o recibe información. El acceso a un objeto potencialmente implica acceso a la información que contiene. Son ejemplos de objetos: bloques, páginas, segmentos, archivos, directorios, árboles de directorios y programas, así como bits, bytes, palabras, campos, procesadores, monitores, teclados, impresoras, nodos de red, etc.

2.6.2.3 Sujeto

Entidad activa, generalmente en forma de persona, proceso o dispositivo que causa que la información fluya de entre los objetos o que cambie el estado del sistema (el iniciador de una acción).

2.6.2.4 Dominio

Conjunto de objetos que puede accesar un sujeto.

2.6.2.5 Información sensible

Información que, determinada por una autoridad competente, debe ser protegida porque su divulgación no autorizada, alteración, pérdida o destrucción, al menos, causará un daño perceptible a algo o alguien.

2.6.2.6 Pruebas formales

Es un argumento matemático completo y convincente, presentando la justificación lógica completa para cada paso de la prueba, de la veracidad de un teorema o conjunto de teoremas. El proceso de verificación formal usa pruebas formales para mostrar la veracidad de ciertas propiedades y para mostrar que los programas satisfacen sus especificaciones.

2.6.2.7 Etiqueta sensible

Es una pieza de información que representa el nivel de seguridad de un objeto y que describe la sensibilidad (por ejemplo: clasificación) de los datos en el objeto. Las etiquetas sensibles son empleadas por el TCB como las bases para las decisiones del control de acceso obligatorio.

2.6.2.8 Políticas de Seguridad

Se conoce como conjunto de políticas de seguridad al conjunto de reglas, leyes y controles que regulan el cómo una organización maneja, proteje y distribuye su información sensible. Dicho conjunto debe ser definido con precisión e implementado para cada sistema que se emplee para procesar información.

2.6.2.9 Modelo de políticas de seguridad

Presentación informal de un modelo formal de políticas de seguridad.

2.6.2.10 Modelo formal de políticas de seguridad

Informe matemático preciso de una política de seguridad. Para ser adecuadamente preciso, debe representar el estado inicial del sistema, la forma en que éste progresa de un estado a otro, y la definición de un estado seguro en el sistema. Para ser aceptado como base para el TCB el modelo debe soportar una prueba formal de tal forma que si el estado inicial del sistema satisface la definición de "estado seguro" y si todas las suposiciones requeridas por el modelo se mantienen, entonces todos los estados futuros del sistema serán seguros. Algunas técnicas de modelado formal incluyen: modelos de transición de estados, modelos lógicos temporales, modelos de especificación algebraica, etc.

2.6.2.11 Control de acceso discreto

Es un medio para restringir el acceso a objetos basado en la identidad de sujetos y/o grupos a los que pertenecen. Los controles son discretos en el sentido en el que un sujeto con un cierto permiso de acceso es capaz de pasar ese permiso (tal vez

indirectamente) a cualquier otro sujeto a menos de que esté restringido por el control de acceso obligatorio.

2.6.2.12 Control de acceso obligatorio

Medio de restringir acceso a objetos basado en la sensibilidad (representada como una etiqueta) de la información contenida en los objetos y la autorización formal de sujetos para accesar información de cierta sensibilidad.

2.6.2.13 Reutilización de objetos

Es reasignar a algún sujeto un medio (sector de disco, cinta magnética) que contiene uno o más objetos. Para una reasignación segura dicho medio no debe contener residuos de datos de los objetos previamente contenidos.

2.6.2.14 Dispositivo de un sólo nivel

Dispositivo que se emplea para procesar datos de un sólo nivel de seguridad en cualquier tiempo. A partir de que el dispositivo no se le confieren datos de diferentes niveles de seguridad, la etiquetas sensibles no necesitan estar almacenadas con lo datos que están siendo procesados.

2.6.2.15 Dispositivo Multinivel

Dispositivo que es usado de forma que permite procesar datos simultáneamente de dos o mas niveles de seguridad sin riesgo en las transacciones. Para complementar esto, las etiquetas sensibles son normalmente almacenadas en el mismo medio físico y en la misma forma en que los datos son procesados (por ejemplo leíble por la máquina o leíble por personas).

2.6.2.16 Contabilización

Este objetivo de control básico alude a uno de los principios fundamentales de seguridad: la contabilización, por ejemplo la individual. La contabilización individual es la llave para asegurar y controlar cualquier sistema que procesa información para individuos o grupos de individuos. Deben de cumplirse una serie de requerimientos para satisfacer este objetivo:

- El primer requerimiento es para identificación individual de usuarios.
- Segundo, existe la necesidad de autentificación y de identificación. La identificación es funcionalmente dependiente de la autentificación. Sin una identidad creíble, ni las políticas de seguridad obligatorias ni las discretas pueden ser invocadas propiamente por que no hay confianza de que las autorizaciones se hayan hecho adecuadamente.
- El tercer requerimiento es sobre capacidades de intervención. Esto es, un TCB debe proveer a personal autorizado la habilidad de intervenir cualquier acción que

pueda potencialmente causar acceso, generación de, o efecto de liberación de información sensible o clasificada.

2.6.2.17 Confianza

Este objetivo de control básico es concierniente a garantizar o proveer confianza de que las políticas de seguridad han sido implementadas correctamente y sin distorsión alguna. El ciclo de vida de la confianza se refiere a los pasos tomados por una organización para asegurarse de que el sistema está diseñado, desarrollado y mantenido usando controles y estándares rigurosos.

2.6.2.18 Ruta de comunicación confiable

Mecanismo mediante el cual una persona en una terminal puede comunicarse directamente con el TCB. Este mecanismo únicamente puede ser activado por el personal o el TCB y no puede ser imitado por software que no sea parte del TCB.

2.6.2.19 Canal Oculto

Canal de comunicación que permite a un proceso transferir información de forma que viola la(s) política(s) de seguridad del sistema. Por ejemplo el canal oculto de almacenamiento que consiste de un canal que compromete la escritura directa o indirecta a alguna localidad de almacenamiento por un proceso y la lectura directa o indirecta de la localidad de almacenamiento por otro proceso; los canales de almacenamiento ocultos típicamente comprometen un recurso finito (por ejemplo sectores de disco) que es compartido por dos sujetos en diferentes niveles de seguridad.

2.6.2.20 Facilidades

Se refiere a las funciones del operador y del administrador. Cuando se habla de manejo confiable de las facilidades el significado corresponde a que el TCB debe soportar funciones separadas de administrador y de operador y que áquellas funciones de administrador que puedan resultar "no muy seguras" se asignen estrictamente a una persona confiable para ello.

2.6.2.21 Manejo de la Configuración

Durante el desarrollo y mantenimiento del TCB debe tener lugar un manejo de la configuración para todo el hardware, software y firmware relevante en cuanto a seguridad que mantenga un control sobre los cambios de las especificaciones, diseño de datos, documentación, código fuente, pruebas, etcétera. Debe protegerse de modificaciones o destrucciones la copia maestra de todo el material empleado para generar el TCB.

2.6.2.22 Monitor de referencia

Concepto de control de acceso que se refiere a una técnica de mediación de todos los accesos a objetos por sujetos. Esto es, el monitor de referencia valida cada referencia a

datos o programas por cualquier usuario (programa) contra una lista de autorizaciones para ese usuario. Los tres requerimientos de diseño para el monitor de referencia son:

a) El mecanismo de validación de referencia debe ser a prueba de intromisiones.

b) El mecanismo de validación de referencia debe ser invocado siempre.

c) El mecanismo de validación de referencia debe ser suficientemente corto para ser sujeto de análisis y pruebas, y completo para ser seguro.

2.6.3 Clases y divisiones

Como se mencionó algunas de las divisiones (C y B) contienen subdivisiones llamadas clases. A continuación se resumen las características a cumplir por un sistema para encontrarse dentro de una división y, según corresponda, a una clase.

2.6.3.1 DIVISIÓN D: PROTECCIÓN MÍNIMA

Esta división contiene sólo una clase. Está reservada para aquellos sistemas que han sido evaluados pero que fallan en alcanzar requerimientos de una clase mayor. La mayoría de los sistemas de PC puede pertenecer a esta clase, pero no tiene sentido evaluar para caer en esta división, que es por omisión.

2.6.3.2 DIVISIÓN C: PROTECCIÓN DISCRETA

Las clases en esta división proveen protección discreta e informes de sujetos y las acciones que ellos realizan a través de la inclusión de capacidades auditoras y contabilizaciones de sujetos y las acciones que éstos inician.

A) Clase C1: Protección de seguridad discreta

El TCB de un sistema de clase C1 proporciona los elementos básicos para satisfacer los requerimientos mínimos de seguridad discreta proveyendo separación de los usuarios y los datos. Incorpora algunos controles capaces de reforzar las limitaciones de acceso en una base individual, por ejemplo, permitir a los usuarios el proteger o privatizar información no dejando a otros usuarios leer "accidentalmente" o destruir sus datos. Se supone que el ambiente de una clase C1 sea de usuarios procesando datos al mismo nivel de sensibilidad de manera cooperativa.

B) Clase C2: Protección de acceso controlada

Los sistemas en esta clase refuerzan de una manera más granular el control de acceso discreto que los sistemas de la clase C1, haciendo a los usuarios responsables de manera individual de sus acciones a través de sus procedimientos de *login*, auditando los eventos de seguridad relevante, y del aislamiento de recursos.

El sistema operativo Windows NT cae dentro de la clase C2.

2.6.3.3 DIVISIÓN B: PROTECCIÓN OBLIGATORIA

Un requerimiento básico en esta división es la noción de un TCB que se encargue de preservar la integridad de las etiquetas y que utilice éstas para reforzar un conjunto de reglas de control de acceso obligatorias. Los sistemas en esta división deben manejar las etiquetas con estructuras de datos en el sistema. El desarrollador del sistema también proveerá el modelo de políticas de seguridad en el que se basa el TCB. Se debe de proveer también la evidencia necesaria que demuestre que el concepto de monitor de referencia se ha implementado.

A) Clase B1: Protección de seguridad mediante etiquetas

Los sistemas de la clase B1 requieren todas las características de la clase C2, además de un modelo de seguridad informal, etiquetamiento de datos, y control de acceso obligatorio sobre objetos y sujetos. Cualquier "flujo"[9] identificado por pruebas debe ser removido.

Sybase Secure SQL Server ha sido evaluado para esta clase.

B) Clase B2: Protección Estructurada

En los sistemas de la clase B2, el TCB se basa en un modelo formal de políticas de seguridad claramente definido y documentado que requiere del reforzamiento del control de acceso obligatorio y discreto encontrados en la clase B1 extendiéndose a todos los sujetos y objetos en el sistema. Además se direccionan canales ocultos. La interfase TCB está mejor definida y su diseño e implementación se someten a mayores pruebas y más complejas revisiones. Se refuerzan mecanismos de autentificación. Se agregan elementos de confiabilidad para el operador y administrador del sistema. El sistema es relativamente resistente a la penetración.

C) Clase B3: Seguridad de dominios

El TCB de la clase B3 debe satisfacer los requerimientos del monitor de referencia, intervenir todos los accesos de sujetos y objetos, ser a prueba de intromisiones, y lo suficientemente pequeño para ser sujeto de análisis y pruebas. Por esto, el TCB está estructurado para excluir código no esencial para reforzar las políticas de seguridad. Soporta un administrador de seguridad, los mecanismos de auditoría se expanden para señalar eventos de relevancia en seguridad, además se requieren procedimientos de recuperación del sistema. El sistema es altamente resistente a la penetración.

Las siguientes compañías han sometido sistemas para caer dentro de alguno de los niveles mencionados[xvi] :

---

[9] *flujo: Error por descompostura, omisión o descuido en un sistema que permite eludir los mecanismos de protección*

[xvi] N. Derek Arnold. *UNIX Security A practical tutorial.* McGrawHill, 1993 pág. 6

SecureWare, Inc. of Atlanta, Georgia ha sometido sistemas para alcanzar evaluaciones B1, B2 y C2.

Sun Federal, Inc. (subsidiaria de Sun Microsystems Inc.) mercantiliza una variación llamada Seguridad Multinivel (MLS) que AT&T ha incorporado en su versión para el gobierno de los E.U. del sistema operativo UNIX, ha sido evaluada para B1.

Addamax Corporation of Champaign, Illinois, fue evaluada para B1.

2.6.3.4 DIVISIÓN A: PROTECCIÓN VERIFICADA

Esta división se caracteriza por el uso de métodos de verificación formal de la seguridad para asegurarse de que los controles de seguridad obligatorios y discretos empleados en el sistema puedan efectivamente proteger la información clasificada o sensible almacenada o procesada por el sistema. Se requiere una extensiva documentación para demostrar que el TCB cumple con los requerimientos de seguridad en todos los aspectos de diseño, desarrollo e implementación.

A) Clase A1: Diseño verificado

Los sistemas en la clase A1 son equivalentes funcionalmente a los de la clase B3, no se agregan características en la arquitectura o requerimientos de políticas. Los rasgos distintivos de los sistemas en esta clase es el análisis derivado de la especificación formal del diseño, las técnicas de verificación y el resultante alto grado de confianza de que el TCB está correctamente implementado. Esta seguridad proviene del desarrollo, desde el modelo formal de las políticas de seguridad hasta las especificaciones de diseño formales del más alto nivel. Independientemente del sistema o lenguaje de verificación empleado, existen criterios importantes para la verificación del diseño, entre ellos destaca que el modelo formal de políticas de seguridad debe estar claramente identificado y documentado, incluyendo pruebas matemáticas de la consistencia del modelo con sus axiomas respectivos. En concordancia con el exhaustivo diseño y análisis de desarrollo del TCB requerido de sistemas en la clase A1, se requiere un más riguroso manejo de la configuración y se establecen procedimientos para una segura distribución del sistema a los diferentes sitios. Es soportado un administrador de seguridad.

Consideramos importante mencionar las áreas que destacan en una clase A1:

Arquitectura del sistema

Debe proporcionarse una demostración formal que muestre los requerimientos de auto-protección y completitud para los monitores de referencia que hallan sido implementados en el TCB.

Pruebas de seguridad

Se debe aspirar que a un futuro se implemente algún tipo de auto-prueba automática para las especificaciones formales.

Especificaciones y verificaciones formales

El TCB debe ser verificado a nivel de código fuente, utilizando métodos de verificación formal donde sea posible (se ha probado que éste tipo de verificaciones sobre un sistema operativo son realmente difíciles).

Ambiente de diseño confiable

El TCB debe ser diseñado bajo condiciones confiables, con características de confiabilidad y sólo con personal confiable.

Hasta la fecha el único que ha alcanzado la clasificación A1[xvii] ha sido el sistema Honeywell SCOMP.

En la siguientes imágenes se tienen los criterios de evaluación de un sistema computacional confiable divididos por grupos y aplicables a cada división (renglones). Cada columna indica un criterio y la tonalidad del cuadro es un indicativo de aplicación de dicho criterio en cierta división.

[xvii] Idem

Sumario de Criterios de Evaluación para un Sistema Computacional Confiable (TCB)

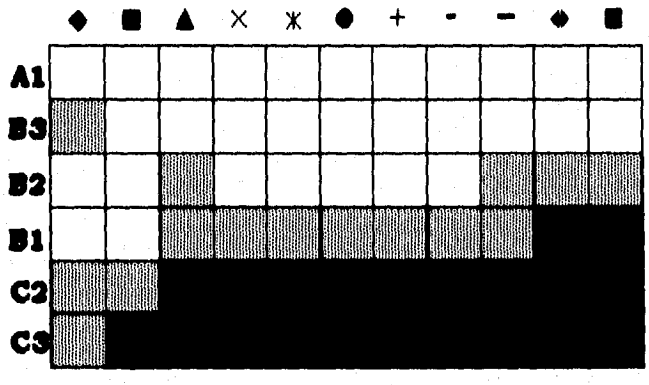
Significado de las tonalidades

Requerimiento igual al de la clase inferior

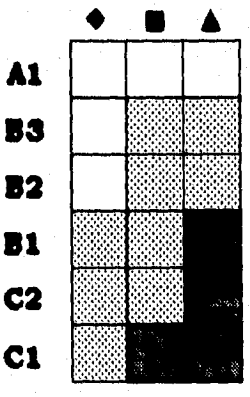
Requerimiento nuevo o ampliado sobre la clase inferior

No es requerimiento para esta clase

Políticas de Seguridad



- ◆ Control De Acceso discreto
- ■ Reutilización de objetos
- ▲ Etiquetas
- × Integridad de las etiquetas
- ✳ Exportación de información etiquetada
- ● Exportación de dispositivos multinivel
- \+ Exportación de dispositivos de un solo nivel
- \- Etiquetamiento de salida leíble para el usuario
- — Control de Acceso obligatorio
- ◆ Etiquetas sensitivas del sujeto
- ■ Etiquetas de dispositivo

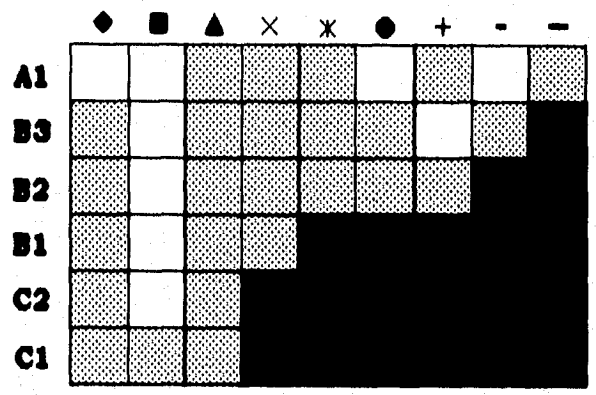


Contabilización

- ◆ Identificación y Autentificación
- ■ Auditorías
- ▲ Ruta confiable

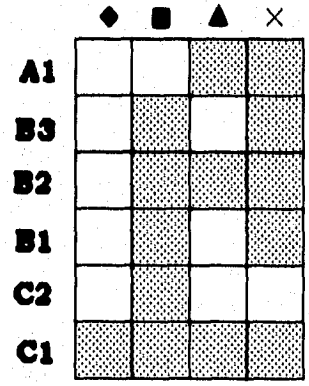
- ◆ Arquitectura del Sistema
- ■ Integridad del Sistema
- ▲ Pruebas de Seguridad
- × Especificación y verificación del diseño
- ✳ Análisis de canales ocultos
- ● Manejo confiable de las facilidades
- \+ Manejo de la configuración
- \- Recuperación confiable
- — Distribución confiable

Confiabilidad

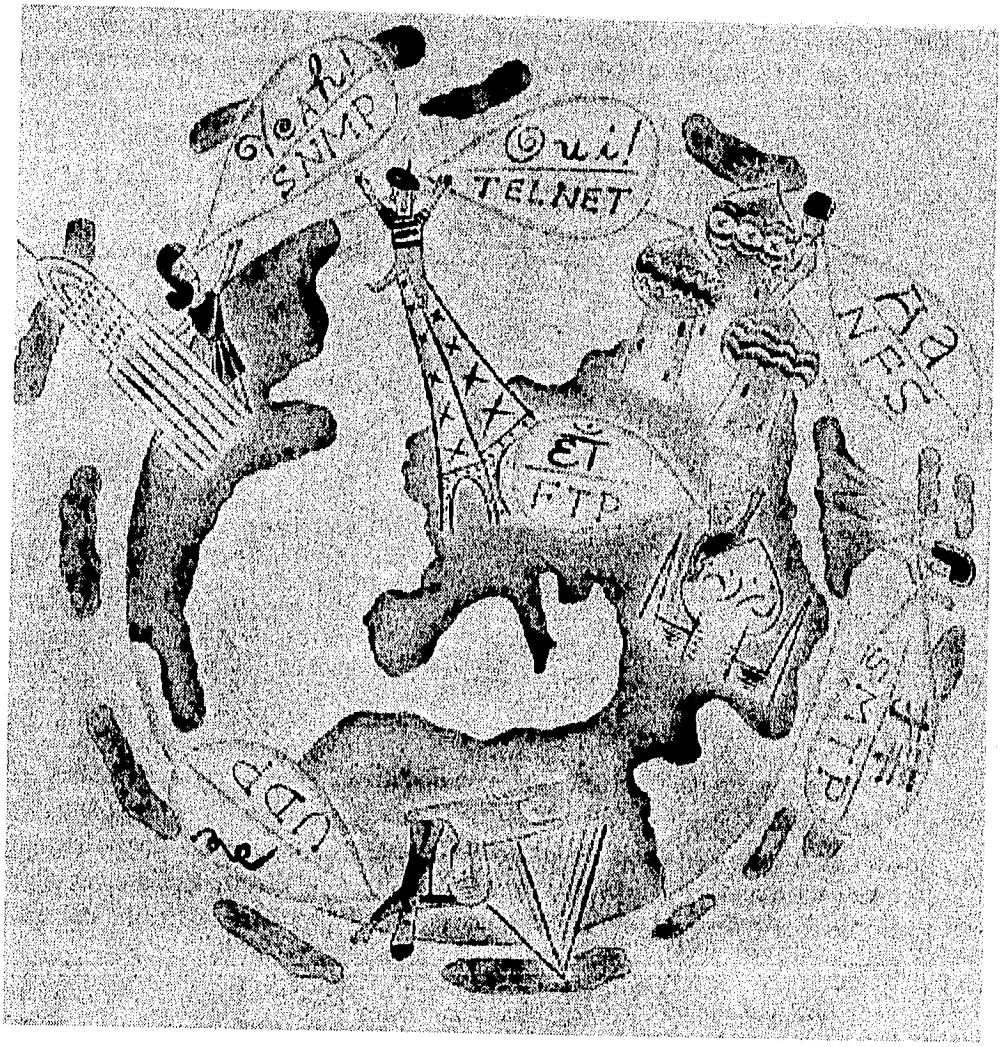


Documentación

- ◆ Manual de Usuario de las implementaciones de Seguridad
- ■ Manual de seguridad sobre implementaciones para el Administrador
- ▲ Documentación de las pruebas
- × Documentación del diseño



CAPÍTULO 3



Descripción de la una red TCP/IP

Este capítulo tratará los dos puntos que conforman una red típica TCP/IP conectada a la Internet, que es también la típica dentro de la redUNAM. Estos puntos son la topología física y la topología lógica. En la primera parte se enunciarán las partes que conforman la topología física de la red, sus características principales, sus ventajas y desventajas. En la segunda parte se habla sobre la parte lógica (protocolos) que hacen funcionar a la red y se describe cómo se desempeñan estos.

La topología física de una red está definida por el método de enlazar el cable entre dos nodos; la red física es lo que se puede ver y tocar. La topología lógica está definida por las características eléctricas de la red, no se puede ver o tocar pero define la forma en que la red funciona.

3.1. Topología Física

Podemos agrupar los componentes de la topología física de la siguiente manera:

- Cableado
- Repetidores
- Puentes
- Enrutadores
- Enlaces de radio frecuencia

En seguida se dan las descripciones generales.

3.1.1 Cableado

3.1.1.1 Cable Par Trenzado

Como su nombre lo indica consiste de dos alambres de cobre aislados y trenzados uno en otro con el fin de reducir la interferencia eléctrica. Es el más antiguo de los medios de transmisión y uno de los más difundidos, se le reconoce por su amplio uso en las redes telefónicas. Puede utilizarse tanto para transmisión analógica como digital y su ancho de banda depende de la distancia que tiene que cubrir así como del calibre del alambre.

Existen dos tipos de cables de par trenzado:

- UTP (Unshielded Twisted Pair) Par trenzado sin blindaje

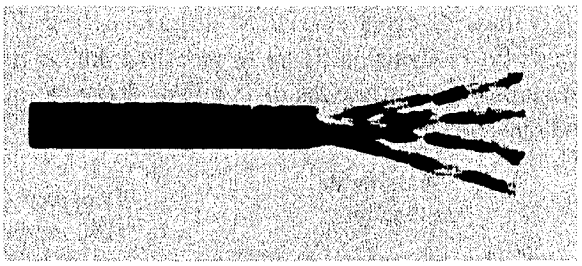


Fig. 3.1 Par Trenzado sin blindaje

- **STP (Shielded Twisted Pair) Par trenzado blindado**

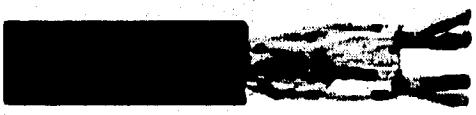


Fig 3.2 Par trenzado blindado

La diferencia principal radica en que el par trenzado blindado tiene una cubierta de aluminio entre los cables y el forro plástico cuyo propósito es reducir la absorción de ruido.

El UTP es más fácil de instalar pero es bastante más susceptible al ruido, es más barato no sólo que el STP sino que los otros tipos de cables. Aunque el STP es algo más seguro podemos decir que estos dos tipos de cables no ofrecen gran seguridad para una red por su facilidad a ser intervenidos.

3.1.1.2 Cable coaxial(coax)

Consiste de un conductor de cobre en la parte como núcleo rodeado por un aislante que a su vez está rodeado por un conductor cilíndrico en forma de malla. La naturaleza del cable lo hace altamente resistente a la interferencia y capaz de soportar un gran ancho de banda.

El coax es frecuentemente empleado en las redes que conectan redes locales, porque dichas redes deben soportar los niveles de tráfico de todas las redes pequeñas que implica un mayor ancho de banda.

La impedancia estándar para redes Ethernet de cable grueso es de 50 ohms. Una de sus características es que al final del cable debe terminar con una resistencia de la misma impedancia que el promedio del cable, un terminador es el conector que incorpora el resistor apropiado. Esta resistencia balancea las características eléctricas del cable y absorbe señales de donde termina el cable de modo que éstas no pueden rebotar hacia el cable y causar interferencia.

Existen dos tipos de cable coaxial: el delgado o "thin Ethernet" y el grueso o "standard Ethernet". En la siguiente figura se muestran ambos tipos de cables (tanto el delgado como el grueso son construidos de la misma forma, lo que difiere es el tamaño del conductor y el diámetro del cable).

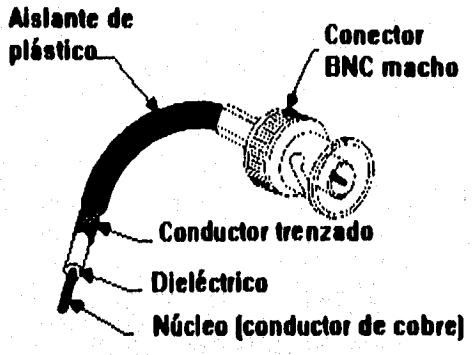


Fig. 3.3 Cable Coaxial

Generalmente, los cables coaxiales más gruesos pueden llevar señales más lejos y tienen un mayor ancho de banda. Segmentos de Ethernet estándar pueden servir a aproximadamente 100 nodos sobre una distancia total de 500 metros.

En la siguiente figura se muestra la topología de un segmento simple del cable coaxial Ethernet estándar.

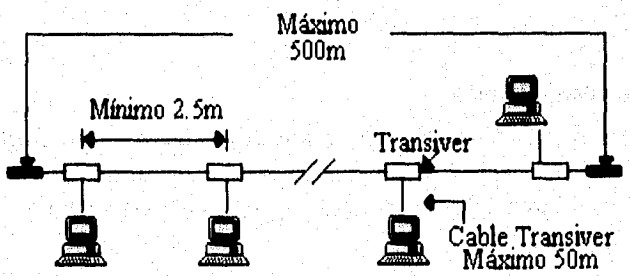


Fig. 3.4 Cable coaxial grueso

Las reglas de configuración básicas son las siguientes:

- Un segmento de cable no puede ser mayor a 500 metros.
- Cada segmento debe conectarse a tierra en un sólo punto de la tierra del sistema del edificio.

- En los extremos de cada segmento un terminador de 50 ohms.
- Un máximo de 100 transceptores por segmento de cable
- La distancia mínima entre transceptores es de 2.5 metros.
- El largo máximo del cable transceptor es de 50 metros.

En la siguiente figura se muestra la topología de un segmento simple del cable coaxial delgado.

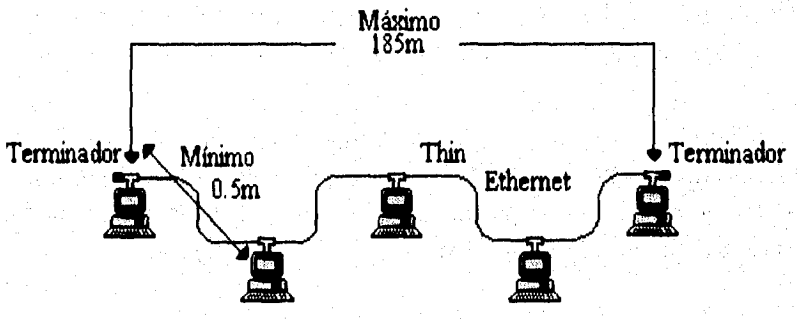


Fig 3.5 Cable coaxial delgado

Las reglas básicas de configuración son:

- El largo máximo de segmento de cable puede ser hasta de 185 metros
- En los extremos debe haber un terminador de 50 ohms.
- Las estaciones pueden estar directamente conectadas al cable usando conectores BNC T o transceptores.
- Debe haber al menos 0.5 metros entre cada estación.
- El máximo número de estaciones es de 30.

En cuanto a la instalación del cable de cobre en general deben tomarse consideraciones importantes. Las dos reglas para conectar cable de cobre son emplear los mejores conectores y obtener las herramientas adecuadas. Ahorrar unos pesos en una herramienta prácticamente garantiza que la red tendrá fallas en los medios.

3.1.1.3 Cable de Fibra óptica

La fibra óptica supera con mucho al cable de cobre, pero es más cara y menos flexible en su manejo. Su potencialidad radica en que puede trasmitir datos mucho más rápido, a mayores distancias y no es sensible a interferencias eléctricas, otras de sus ventajas son:

- Reciclable para futuros estándares
- Puede ser empleado para voz, datos y video
- No depende de tierras ni de diferencias de potenciales

Para la transmisión de los datos el cable de fibra óptica usa pulsos de luz en vez de pulsos de electricidad. La luz viaja a través de un tubo de vidrio delgado en un ángulo que forza a las ondas de luz a refractarse en el tubo en vez de fugarse, evitando pérdidas.

El cable de fibra de vidrio es el más usado. Los cables de fibra plástica han sido introducidos recientemente y prometen reducir el costo del cableado con fibra óptica. La fibra de plástico es menos transparente y tiene menor velocidad de transmisión que la de vidrio.

El cable de fibra óptica consiste de una delgada fibra como núcleo (usualmente fabricada de silicón altamente purificado) cubierta por una funda delgada que absorbe la luz de plástico o de vidrio. Esta funda está a su vez forrada por un plástico grueso. La construcción de la fibra óptica se observa en la siguiente figura:

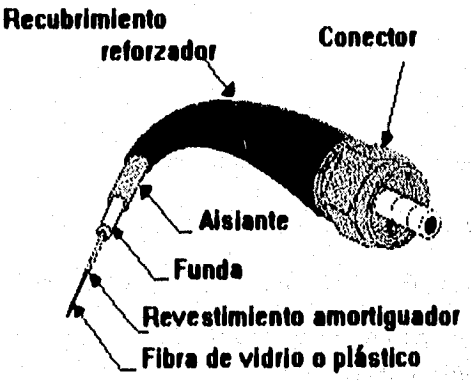


Fig. 3.6 Construcción de la Fibra óptica

Una fuente de luz al final del cable introduce en la fibra pulsos de luz codificados. Dicha fuente puede ser un diodo láser o un LED (light emitting diode). Un LED es menos caro, pero el diodo láser es más eficiente y puede alcanzar velocidades de envío de datos mayores. Los pulsos de luz son transmitidos a través de la fibra hacia un fotodiodo en el extremo opuesto, que los convierte en señales eléctricas.

Las velocidades de transmisión actuales están limitadas por el tipo de red. La de una red Ethernet es de 10 Mbps. Existen versiones de Token Ring de 4 y 16 Mbps. FDDI tiene un límite teórico de 100 Mbps por segundo. Es posible combinar múltiples señales de red en una sola fibra empleando una técnica conocida como multiplexión. Mediante la multiplexión una sola fibra puede llevar señales para 10 Ethernets entre dos edificios

separados ampliamente, a pesar de esto, cada Ethernet continua operando a una velocidad efectiva de 10 Mbps.

La fibra óptica tiene tres modos de transmisión:

- Multimodo
- Multimodo de índice graduado
- Monomodo

En la *fibra multimodo* la luz entra en una variedad de ángulos, en algunos la luz escapa de la fibra y es absorbida por la funda, en otros la luz es refractada continuamente a lo largo de la fibra; las refracciones en este modo son ligeramente imperfectas, las refracciones espóreas limitan el ancho de banda que la fibra puede soportar. La fibra multimodo tiene el ancho más bajo de los tres tipos y soporta anchos de banda arriba de 200 MHz/km.

La *fibra multimodo de índice graduado* usa fibra de vidrio en la que el índice de refracción de la fibra es variado. En vez de refractar señales éstas fibras gradualmente encorvan las señales en dirección de la funda. La luz permanece coherente y el ancho de banda crece. Este tipo de fibra es tal vez el más usado en redes de área local. Las fibras multimodo de índice graduado soportan anchos de banda cuyos valores están entre los otros dos modos, y es generalmente empleado para soportar anchos de banda de 100MHz/km a 3GHz/km.

Ya que las señales de luz pueden seguir diferentes rutas a través de los cables multimodo de índice graduado, las señales pueden tomar levemente diferentes tiempos para alcancar al receptor al final del cable. Esta característica limita al ancho de banda.

La *fibra monomodo* tiene el mayor ancho de banda. En este tipo de cable el diámetro de la fibra de vidrio es reducido de modo que sólo una señal puede ser transmitida. Esto elimina las refracciones y encorvamientos de la fibra multimodo e incrementa dramáticamente el ancho de banda, puede proveer anchos de banda arriba de los 50GHz/km.

Los diámetros de las fibras ópticas se especifican en términos de micras (millonésimas de metro). Esta medida también se conoce como la *apertura* porque representa el ángulo en el que el cable puede aceptar luz. Diamétros comunes son 50, 62.5 (el más común y estándar de FDDI), y 100 micras.

Estos diámetros extremadamente pequeños determinan una de las mayores dificultades al instalar sistemas de fibra óptica: al alinear dos fibras, las terminaciones de la fibra deben ser muy cuidadosamente alineadas. Los conectores, herramientas y habilidades de instalación requeridas para lograr esta precisión son mas caros de obtener que en la instalación del cable de cobre.

También se requiere un equipo para prueba de fibra. Una mala instalación o fibra dañada puede introducir pérdidas que afecten la operación de la red. Los examinadores de cable deberían ser considerados como herramientas obligatorias para el soporte de cualquier red grande, ya sea que use cable de cobre o fibra óptica.

3.1.1.4 Cable transceptor

Un cable transceptor conecta el transceptor a una tarjeta de interfase en la terminal. La máxima longitud del cable es de 50 metros. Está formado por cinco pares de cable trenzados e individualmente aislados. Dos pares para los datos de entrada y salida respectivamente, otros dos para señales de control de entrada y salida y un quinto para permitir que la terminal alimente a los circuitos eléctronicos del transceptor.

El cable transceptor termina en la tarjeta de interfase dentro de la estación. A su vez esta tarjeta contiene un chip controlador que transmite y recibe tramas hacia y desde el transceptor, respectivamente.

En la siguiente tabla se enuncian las diferentes ventajas y desventajas de los cables mencionados, así como algunas comparaciones entre ellos.

	Ventajas	Desventajas
Par Trenzado	• Ampliamente Difundido, Fácil instalación • El más económico	• Susceptible al ruido
Cable coaxial	• Resulta fácil trabajar con él • Resistente a la interferencia • Alto ancho de banda • Tecnología bien establecida • Algunos edificios ya están cableados con cable coaxial	• Más caro que el par trenzado. • No soportado por algunos estándares de red como Token Ring

Cable de Fibra óptica	• Insensible a la interferencia • Alto ancho de banda • La tecnología tiene un alto potencial de vida • Más ligero y pequeño que el de cobre	• Gasto inicial alto • Mayor dificultad y tiempo al instalar que el empleado en el de cobre
Cable coaxial delgado y grueso	El cable coaxial delgado es más fácil de manejar y de menor costo que el cable grueso. El problema es que cada segmento de cable coaxial delgado en la red tiene un largo máximo (útil) de 185 metros y soporta un máximo de 30 nodos (que para muchos sitios es suficiente).	
Fibra óptica contra cable coaxial	• El cable de fibra óptica es más ligero y de menores dimensiones • Es completamente insensible a la interferencia eléctrica • Alcanza mayores anchos de banda, esto implica que transmite mayor número de datos a mayores distancias • Es considerablemente más caro que el cable coaxial • Su instalación es más complicada, pero existen grandes avances en la tecnología para los conectores de instalación	

3.1.1.5 Conectores

A) Conectores comunes

El conector más común en redes de cable coaxial es el conector BNC. Los conectores para cable coaxial deben mantener la conexión entre el centro del conductor y el blindaje, y todos tienen un pin en el centro que se conecta al conductor. El BNC (conector de bayoneta) es un conector que se enrosca en su posición, es empleado para Ethernet delgado, ARCnet y otras redes. En la siguiente figura se muestra un BNC con sus respectivas partes, estos conectores son fáciles de instalar; la parte más importante es una laminilla diseñada para el cable en específico y para los requerimientos de la marca de conector en particular que será usado.

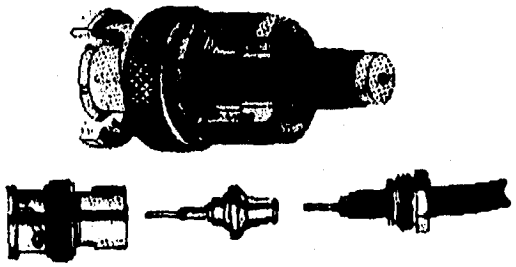


Fig. 3.7 Conector BNC

Otro conector común es el N empleado para coaxial grueso.

Cuando segmentos de coaxial deben de conectarse se emplea un conector de cañón o tonel que permite a los conectores en los extremos de los dos cables estar en línea.

Los dispositivos usualmente se conectan al cable coaxial delgado empleando un conector T. Para instalar un conector T, el coaxial principal debe cortarse y conectarse a la barra del conector T con conectores BNC. El dispositivo a conectarse se liga directamente a la base del T.

Los conectores son generalmente el punto débil de una red, y el conector T tiene muchas oportunidades de fallar. La mayoría de los conectores y cables introducen muchísima tensión mecánica. Es recomendable utilizar sólo conectores y cables de alta calidad para reducir la probabilidad de falla.

Actualmente existe una alternativa para estos problemas, un innovador sistema de conectores Ethernet de AMP, Inc. puede eliminar los vulnerables conectores T. El AMP LAN-LINE Thinnet Tap System consiste de placas, cables y cajas en forma de pastilla que hacen a los sistemas de cableado Ethernet más fáciles de reconfigurar y menos susceptibles a fallas.

B) Terminadores

Los terminadores deben instalarse en los extremos de la mayoría de tipos de cables coaxiales. Un terminador es un conector que incluye una resistencia que balancea las características eléctricas del cable. Los terminadores se requieren normalmente para topologías físicas de bus únicamente, de las que Ethernet es un ejemplo. Los cables en otras topologías se conectan generalmente directamente entre dos dispositivos y son terminados por los mismos dispositivos. Este es el caso para token Ring.

Un terminador BNC se muestra en la siguiente figura:

Fig. 3.8 Terminador BNC

C) Transceptores (Transceivers)

Los transceptores son dispositivos que proveen la conexión con el cable coaxial y contiene la eléctronica necesaria para enviar, recibir y manejar las señales codificadas. El transceptor puede configurarse como un toma tipo T, con dos conectores para el cable coaxial principal. El transceptor tiene también un conector D que se emplea para conectar el dispositivo de red a través del cable transceptor.

Un transceptor puede manejar la detección por portadora y de colisión. Cuando se detecta una colisión, el transceptor también coloca una señal especial de invalidación en el cable, para asegurar que todos los demás transceptores tengan conocimiento de la colisión.

Algunos transceptores pueden tener conectados hasta 8 estaciones periféricas conectadas con objeto de reducir el número de transceptores necesarios.

3.1.2 Repetidores

Los repetidores se emplean para extender el largo de un cable físico, o el número de estaciones de trabajo permitidas por segmento. Por ejemplo, una diminuta red Ethernet/IEEE 802.3 puede soportar 30 enlaces por segmento de 185 metros. Si se excede ya sea el número de enlaces o el largo, se puede instalar un repetidor entre dos segmentos.

Como su nombre lo indica desempeña la función de amplificar señales. Esto le permite a la señal original ir más lejos de los limites que la atenuación del medio permitiría.

La elección de repetidor depende de dos variables: la arquitectura de red y el tipo de medio de transmisión usado.

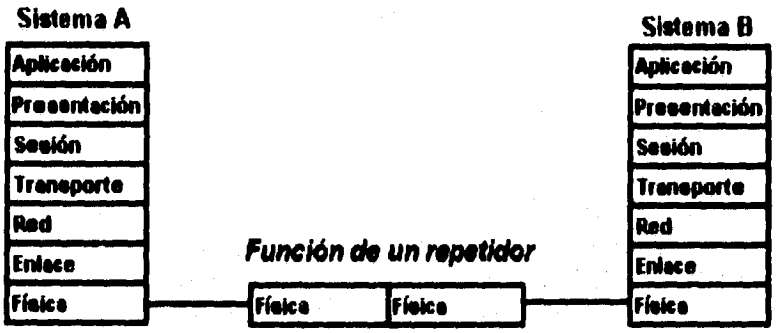


Fig. 3.9 Función de un repetidor.

3.1.3 Puente (Bridge)

Cuando el repetir una señal es insuficiente para la red, un bridge añade la funcionalidad de la capa de red. El bridge separa lógicamente dos segmentos de red operando sobre la dirección dentro de la capa de red (o frame IEEE Medium Access Control [MAC]).

La información que está almacenada en el bridge o dentro del frame ayuda al bridge a tomar una decisión: pasar el frame al siguiente segmento (conocido como forwarding) o no pasar el frame (conocido como filtering). Los puentes operan en redes cuyas capas de enlace tienen esquemas de direccionamiento compatibles (como IEEE 802.3 a 802.3 o 802.3 a 802.5) pero son transparentes a los protocolos de la capa de Red y superiores.

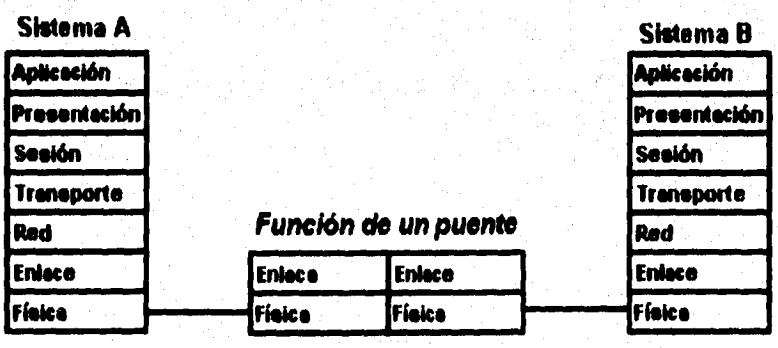


Fig. 3.10 Función de un puente.

3.1.4 Enrutadores (Routers)

Los enrutadores operan en la capa de red y pueden interpretar uno o más protocolos en dicha capa. Conviene recalcar que la capa de red hace una selección entre las rutas disponibles dentro de una subred de comunicaciones, eventualmente conectando los hosts destino y origen. Un enrutador lee la información referente a la dirección destino y pasa el paquete a la apropiada red destino (los puentes simplemente toman una decisión binaria pasar o no pasar el frame después de examinar la direccíon de la capa de ligado de datos). Los enrutadores pueden operar en un protocolo de capa de red como DoD Internet Protocol (IP), o múltiples protocolos como IP, DECnet, e IPX de Novell.

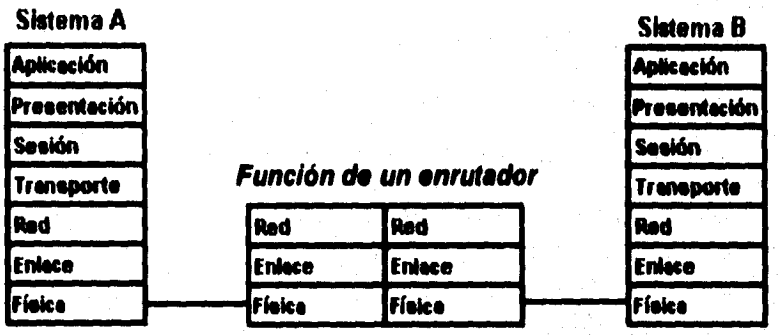


Fig. 3.11 Función de un enrutador.

3.1.5 Gateways

Los Gateways pueden operar en las siete capas de OSI. Interconectan aplicaciones de diferentes conjuntos de protocolos. Son orientados a aplicación y pueden ser responsables de conectar sistemas electrónicos de correo incompatibles, convertir y transferir archivos de un sistema a otro, o habilitar la interoperabilidad entre diferentes sistemas operativos.

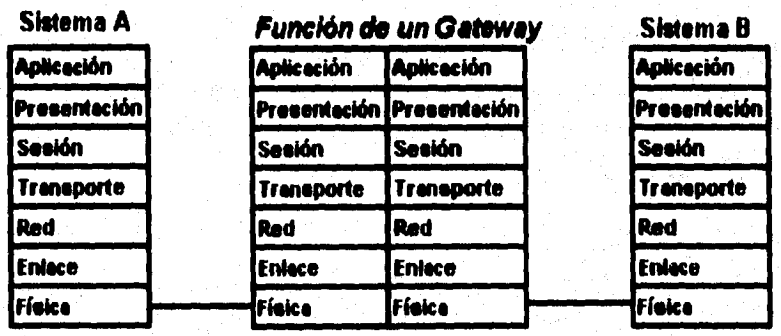


Fig. 3.12 Función de un Gateway.

La siguiente tabla resume las funciones, objetivos y aplicaciones de dispositivos de hardware para *internetworking*.

3.1.5.1 REPETIDOR Opera en la capa física de OSI Regenera o repite señales físicas Usado para rangos extensos LAN
3.1.5.2 BRIDGE Opera en la capa de Enlace de OSI Separa lógicamente segmentos de red Independiente de protolocolos de capas más altas Empleado para manejo de tráfico en LAN
3.1.5.3 ENRUTADOR Opera en la capa Red de OSI Separa subredes lógicamente Dependiente del protocolo de la capa Internet Debe obtener datos de la topología de red Empleado para la comunicación entre redes
3.1.5.4 GATEWAY Opera en las capas mas altas de OSI (de sesión a aplicación) Dependiente de las aplicaciones del usuario Empleado para la comunicación entre aplicación y aplicación

3.1.6 Enlaces de radio

3.1.6.1 Microondas

La tecnología de microondas es una tecnología de interconexión usada para puentes LAN entre edificios o en un mismo lugar. Las señales son transportadas por microondas que son alta frecuencia, ondas de radio de onda corta.

Una conexión de microondas requiere antenas parabólicas en ambos extremos de ésta. Las antenas deben de estar en línea (verse una a la otra) para la transmisión y recolección de señales.

La distancia de transmisión depende de la altura de la torre desde la que se transmite, mientras mayor sea mayor será la distancia.

Esta tecnología es una excelente opción cuando existen dificultades para abrir zanjas e instalar el cableado, por lo que resulta más barato y se necesita menos mantenimiento (por ejemplo en el cableado se necesitan repetidores).

La transimisión empleando microondas se lleva a cabo en una escala de frecuencia que va desde 2 a 40ghz. Estas frecuencias se han dividido en bandas y asignado a diferentes aplicaciones (telefónicas, gubernamentales, militares, etc.).

Uno de los problemas en la transmisión con microondas es la interferencia debido a lluvias intensas, aunque hay quienes opinan que esto es un mito y que la lluvia no es un gran problema.

Las principales ventajas de la transmisión con microondas son:

- Es privada y bajo control del usuario
- Normalmente es menos costosa que contratar una línea

Las desventajas serían:

- Gran inversión al principio
- Puede dañar los ojos
- Problemas con las leyes de algunas ciudades (asignaciones de bandas).[12]

3.1.6.2 Comunicaciones satelitales

La transmisión satelital es usada para transmitir datos entre puntos separados por grandes distancias, en lugares donde el terreno o la infraestructura no permiten cableado o resultaría, sino imposible, si muy difícil de instalarse una red.

Las comunicaciones satelitales pueden transportar todo tipo de datos: voz, imágenes, datos, etc.

En este tipo de comunicaciones los satélites generalmente están en una órbita geosíncrona al ecuador de tal forma que parecen estar estáticos ya que se mueven a la misma velocidad que la tierra. Cada satélite tiene cierto número de transmisores-

[1] *Connectivity:Local Area Networks* pag. 114

[2] *Redes de Ordenadores* pág 76 y 77

receptores que reciben las señales de comunicación, toma la señal, la limpia, amplifica y la transmite de regreso a la Tierra, pudiendo abarcar grandes radios, continentes enteros dependiendo de las antenas empleadas tanto en el satélite como en los extremos de las ligas en la Tierra.

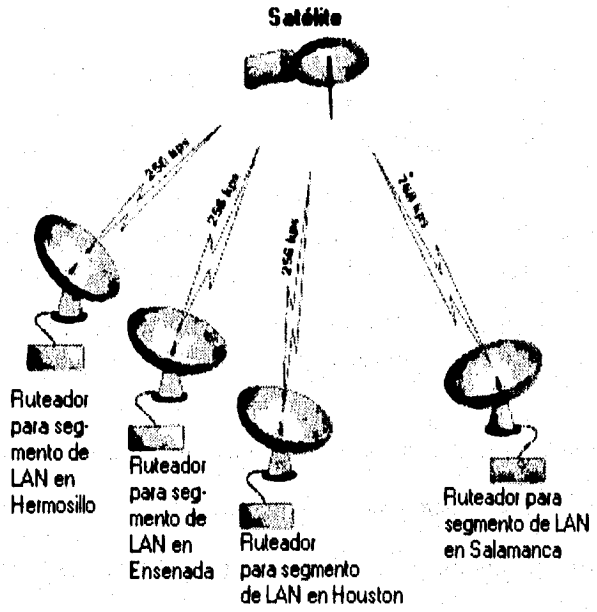


Fig. 3.13 Comunicaciones satelitales.

El cobro por uso de satélite no depende de la distancia sino del ancho de banda que es una medida del promedio de datos que se transmiten.

En cuando a costos si hablamos de una distancia menor a 500 millas los circuitos satelitales son extremadamente más caros que los enlaces terrestres, pero superando esta distancia son muy competitivos. Si se desea instalar temporalmente un sistema el gasto de sátelite sería menor que tener que pagar todos los gastos de instalación de líneas terrestres.

Las comunicaciones mediante satélite son confiables aún cuando ocurran desastres naturales, esto significa que en tanto resista la estación en la Tierra el desastre no romperá los circuitos.

Las principales desventajas del enlace satelital son:

1. Un desempeño relativamente lento causando mayores gastos a los servicios que requieren mayor rapidez que 19.2 kilobits por segundo (transmisión típica del más bajo costo), puesto que para alcanzar la rapidez necesaria se necesitan mayores y más caras antenas.

2. Retardo de satélite, debido a la distancia entre la Tierra y el satélite, a la señal le toma 0.27 segundos hacer el viaje ida y vuelta. Este retardo del satélite puede ser significativo para ciertas aplicaciones.

Los circuitos satelitales no son la solución perfecta para interconectar redes para toda organización pero ofrecen ciertas características únicas y flexibilidad geográfica que ningún otro servicio puede dar.

3.1.6.3 Módem[3]

Los datos pueden ser transmitidos mediante dos métodos: digital o analógico.

Las comunicaciones digitales son discretas por naturaleza lo que significa que están limitadas a un número de representaciones. Los valores representados analógicamente son infinitamente variables; ejemplos de estos valores son: el patrón de ondas de la voz, la transmisión eléctrica sobre una línea de potencia, etc. Los teléfonos convencionales transmiten señales de manera analógica.

El ancho de banda de una línea telefónica no es suficientemente grande para transmisitir datos a alta velocidad. Se necesitan algunas formas de convertir o cambiar la señal transmitida para poder transmitir a mayores velocidades. Este paso a mayores velocidades se realiza modulando la señal de diferentes maneras.

Modulación es la capacidad del módem de cambiar la frecuencia, amplitud o fase de una onda transmitida, que hace que se alcancen mayores velocidades de transmisión. De tal manera que la velocidad de los módems varía debido a que emplean diferentes tipos de modulación.

Un término muy empleado en este aspecto es *"baud rate"* que se refiere al número de señales que tienen lugar durante un especificado período de tiempo. También tiene que ver con la forma en que el módem modula la señal. Un módem operando a 2400 bauds cambia la señal 2400 veces por segundo. El número de cambios, en la señal actual, que el módem realiza no corresponde directamente con el número de bits transmitiéndose (bps) por que algunas técnicas de modulación permiten que más de un bit sea transmitido en un cambio de señal.

Otro término empleado es *"duplexaje"(duplexing). Half-duplexing* significa que el módem puede comunicarse en una sola dirección en un tiempo dado. Esencialmente significa que el módem debe enviar una transmisión, esperar por el fin del otro extremo y "voltear la línea".

Full-duplexing significa que el módem puede comunicarse en ambas direcciones al mismo tiempo. La comunicación resultante es mucho más rápida.

---

[3] *Connectivity: Local Area Networks* p 297 a 302

A) Tipos de Módems

Existe una gran variedad de módems disponibles. Hace unos años algunos vendedores como Hayes, Telebit y Microcom trataron de establecer estándares *de-facto* para altas velocidades de módems (9600 baud). Como resultado, mucha gente compró sus módems y pudo comunicarse a alta velocidad únicamente con otro módem del mismo fabricante.

Algunos estándares creados por la CCITT habilitan módems de diferentes fabricantes a comunicarse unos con otros, si ellos siguen el mismo estándar.

Uno de los primeros conjuntos de estándares para módems fue el Bell 103/V.21. Este módem es capaz de trasmitir datos a 300 bps o 300 baud, que para los estándares actuales, es extremamente lento. El Bell 103 es obsoleto ahora. Comparable con este fue el Bell 212A que es capaz de transmitir datos a 1200 bps. El 212A puede estar bien para sistemas que no necesitan manejar altas velocidades de tráfico.

El estándar V.32 establecido por la CCITT describe los métodos mediante los cuales los módems pueden operar a 4800 bps y 9600 bps unos con otros. Siguiendo este estándar se asegura que los otros módems puedan comunicarse apropiadamente a alta velocidad, independiente de quien lo manufacture. V.32 usa un método de modulación llamado *"trellis encoding"* que agrega un bit verificador para mejorar el desempeño en una red pública de circuitos. No incluye ningún método de corrección de errores. En el V.32bis se implementa la corrección, también se le agregan velocidades de 7200 bps, 12000 bps y 14400 pbs.

Estos estándares se basan en la forma en que el módem maneja las transmisiones eléctricas para alcanzar mayores velocidades. Los estándares nuevos incluyen la capacidad del módem de comprimir datos, que se refleja en aún mayores velocidades de transmisión.

El CCITT desarrolló un nuevo estándar, el V.42 que usa el procedimiento de Ligado de Acceso para Módems (LAPM) para hacer corrección de errores en hardware. El estándar V.42 bis tuvo además capacidad de compresión de datos, usa las técnicas de compresión Lempel-Zip que compactan los datos antes de ser transmitidos sobre la línea. Los módems que siguen el estándar V.42 pueden transmitir datos a una velocidad de hasta 38400 bps.

Más o menos al mismo tiempo en que la CCITT estaba desarrollando estándares para alcanzar mayores velocidades, Microcom estaba desarrollando sus estándares propietarios para módems, llamados MNP (Microcom Network Protocol). MNP tiene diferentes niveles, algunos comparables con los estándares de la CCITT.

Los diferentes nivles de MNP tiene que ver con corrección de errores y compresión de datos. MNP Nivel 4 es una técnica de corrección de errores que puede darle a un módem una mayor velocidad de transmisión.

MNP nivel 5 es un estándar de compresión de datos, en el que el MNP diseña árboles para comprimir los datos antes de que se transfieran. Es comparable con el estándar V.42bis de la CCITT.

El número de estándares y tipos de módems confunden a mucha gente. Esencialmente, para tener módems de alta velocidad trabajando, se necesita tener el mismo estándar implementado en cada módem en los extremos de la conexión. Cuando los módems empiezan a "hablar", descubren en que nivel se pueden comunicar ambos.

Supongamos, por ejemplo, que nuestro módem sigue el estándar V.42bis. Si conectamos otro módem que siga el mismo estándar, ellos usarán ese estándar para comunicarse. Si no, el módem tratará de detectar el nivel al que el otro módem es capaz de comunicarse y descenderá a ese nivel. Si el otro módem es más simple y soporta 2400 bps, nuestro módem tendrá que bajarse a 2400 bps y apagar la detección de errores o el esquema de compresión de datos, si el otro no lo soporta.

3.2. Topología Lógica

Cómo se menciona al inicio del capítulo la topología lógica está constituída por los protocolos que hacen funcionar a la red. En el caso específico de la RedUNAM el principal protocolo y más difundido es el TCP/IP, y en segundo término en menor grado de difusión IPX.

3.2.1 TCP/IP

En este punto proporcionamos un resumen del conjunto de protocolos TCP/IP. La red Internet , así como la RedUNAM tienen como columna vertebral dicho conjunto de protocolos, de ahí su importancia, así que cuando hablamos de redes TCP/IP nos estamos refiriendo a un gran conjunto de redes si no es que a la mayoría de las que conforman nuestro contexto.

3.2.1.1 LAS DIFERENTES CAPAS[4]

La frase TCP/IP se refiere a una colección de protocolos de comunicación. Fue desarrollado bajo auspicios de la DARPA (U.S. Defense Advanced Research Projects Agency) y fue empleado por primera vez en la antigua ARPANET en 1983.

TCP/IP está organizado en cuatro capas, que están contempladas sobre una capa cinco- el hardware de la red. Las especificaciones de hardware no están construidas dentro del modelo TCP/IP.

[4]Cheswick, William R and Bellovin, Steven M. *Firewalls and Internet Security. Repelling de Wily Hacker.* Addison-Wesley Professional Computing Series, U.S:A.,1994. p.19-48

El siguiente diagrama (fig. 3.10) muestra las capas del modelo y varios de los protocolos que componen las diferentes capas.

Aplicación. La capa de aplicación es la capa de más alto nivel. Esta capa provee servicios a usuarios finales, tales como transferencia de archivos, correo electrónico y acceso a terminales remotas. Los programas de aplicación (algunos tienen definido su propio protocolo) interactúan con la capa de transporte. Estos tienen una alternativa entre varios protocolos de transporte, dependiendo del tipo de transporte que se requiera.

Transporte. El principal objetivo de la capa de transporte es proveer una comunicación punto-a-punto entre aplicaciones. Los protocolos de transporte usan el servicio de liberación de paquetes que provee la capa de Internet.

Internet. La capa de internet provee servicio de liberación de paquetes desde una máquina a otra. La rehabilitación de los paquetes no se lleva a cabo en esta capa, de esto se encargan los protocolos de más alto nivel (transporte o aplicación).

Interfase de Red. Esta capa (algunas veces referida como "enlace de datos") acepta datagramas desde la capa Internet y los envía físicamente. Un módulo de interfase de red es a menudo un manejador de un dispositivo de hardware específico. La capa de red consta de varios módulos.

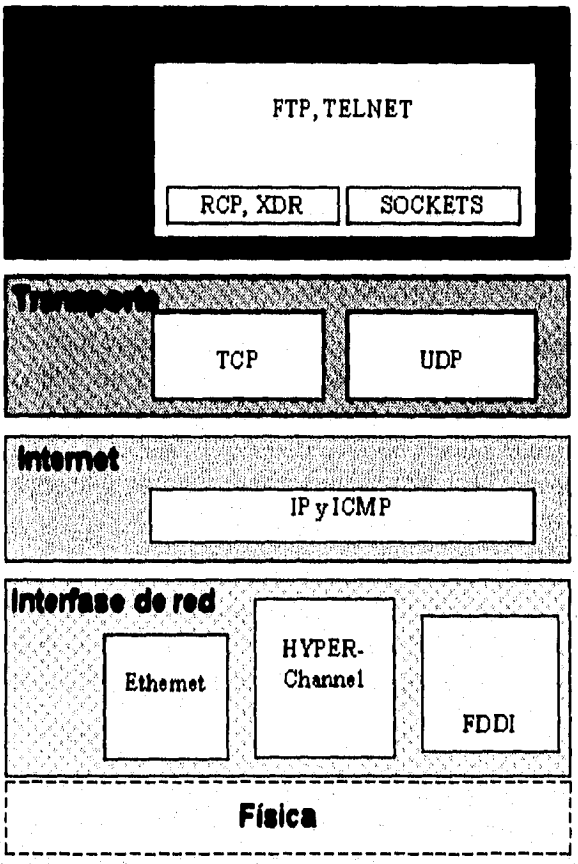


Fig. 3.14 Capas del Protocolo TCP/IP

El flujo de los datos se lleva a cabo entre capas adyacentes, esto es, una capa alta sólo puede comunicarse con la capa inmediata inferior, nunca con una de un menor nivel. Cada renglón es una capa de protocolo diferente. La capa más alta contiene las aplicaciones: transmisión de correo, conexión, servidores de video, etc. Ellos llaman a las capas más bajas para recoger y entregar los datos.

En la mitad de la telaraña está el protocolo Internet (IP)[5]. IP es un paquete multiplexor. Los mensajes de protocolos de más alto nivel tienen un encabezado IP pegados a ellos. Ellos son enviados al manejador de dispositivo apropiado para la transmisión. Examinemos elementos de las capas.

A) IP

Los paquetes IP son los bloques de datos que constituyen la parte fundamental del conjunto de protocolos TCP/IP. Cada paquete lleva una dirección destino y origen de 32

[5]Postel, Jon. *Internet Protocol. RFC 791*, Septiembre 1981.

bits, algunos son opcionales; un encabezado de control (checksum) y la carga de datos. Un paquete IP típico es de unos cuantos cientos de bits de largo. Estos paquetes fluyen por billones a través de las redes.

No existe la noción de circuito virtual o "llamada telefónica" al nivel IP: cada paquete es independiente. IP es un servicio no confiable de *datagramas*. No existen garantías de que el paquete será liberado, ni sólo una vez, o liberado en un orden en particular. No existe ninguna verificación de correctitud del paquete. El checksum del encabezado IP cubre sólo al encabezado.

De hecho, no hay garantía de que el paquete haya sido enviado realmente por la dirección origen dada. En teoría, cualquier host puede transmitir un paquete con cualquier dirección origen. Aunque varios sistemas operativos controlan este campo y aseguran que lleva el valor correcto, no se puede confiar en la validez de la dirección origen, excepto bajo ciertas circunstancias controladas. La autenticación y la seguridad en general, deben ser mecanismos en capas más altas del protocolo.

Un paquete viajando largas distancias puede dar varios saltos. Cada salto termina en un host o en un enrutador, que pasa el paquete al siguiente salto. Un paquete muy largo es fragmentado. Los fragmentos hacen su propio camino separados unos de otros hasta su destino. Cuando las piezas arriban a la máquina destino, son reensamblados.

Direcciones IP

Las direcciones IP tienen 32 bits de largo y se dividen en dos partes, una porción de red y otra de host. Los límites exactos dependen de los primeros bits de la dirección. Las direcciones cuyas porciones son todas ceros o todas unos están reservadas.

Generalmente, la porción de host de las direcciones es dividida en subredes y direcciones de hosts. Las subredes se emplean para ruteo dentro de una organización.

Etiquetas de Seguridad IP

IP tiene un número de campos opcionales que pueden aparecer, pero que no son usados comúnmente. Para nuestros propósitos, los importantes son las etiquetas de seguridad y las fuentes de ruteo estrictas y relajada.

La opción de seguridad IP[6] [7]es empleada principalmente en sitios militares, aunque existe una tendencia tratando de definir una variante comercial. Cada paquete es etiquetado con la sensibilidad de la información que contiene. Las etiquetas incluyen dos componentes: uno jerárquico: Secreto, Ultra Secreto, etc.; y una categoría opcional: criptografía, armas nucleares, etc.

---

[6]Housley, Russell. *Security Label framework for the Internet. RFC 1457.* Mayo 1993

[7]Stephen, Kent. *Security Options for the Internet Protocol. RFC 1108.* Nov. 1991

Las etiquetas indican el nivel de seguridad tanto del proceso que envía como del que recibe. Un proceso no puede escribir en un medio de menor nivel de seguridad porque permitiría la divulgación de información confidencial. Por razones obvias no debería leer de un medio que contenga información más altamente clasificada. La combinación de estas dos restricciones nos obliga a que los dos procesos a los extremos de una conexión sean del mismo nivel[8].

B) ARP

Los paquetes IP normalmente son enviados sobre Ethernets. Los dispositivos Ethernet no entienden las direcciones de 32 bits IP. Por lo que, un dispositivo IP debe traducir la dirección destino IP a una dirección destino del medio de acceso. Ya sea que existan algunos mapeos estáticos o algorítmicos entre estos dos tipos de direcciones, generalmente se emplea una tabla de búsqueda. El protocolo de resolución de direcciones (Address Resolution Protocol) se usa para determinar estos mapeos.

El ARP trabaja enviando un paquete Ethernet conteniendo la dirección deseada IP. El host destino, o cualquier otro sistema actuando en su nombre, responde con un paquete conteniendo su correspondiente dirección Ethernet. Esta es atrapada por el originador para reducir tráfico innecesario.

C) TCP

El protocolo de Control de Transporte (Transport Control Protocol)[9] provee circuitos virtuales confiables para los procesos del usuario. Los paquetes dañados o perdidos son retransmitidos; los paquetes en desorden son mezclados, si es necesario, para volver al orden original de la transmisión.

El orden es mantenido por números de secuencia en cada paquete. Cada byte enviado, así como las peticiones de inicio y fin (open, close), son ordenados individualmente. Todos los paquetes, excepto por el primer paquete TCP enviado durante la conversación contiene un número de aceptación que da el número de secuencia del último byte secuencial recibido exitosamente.

[8]Amoroso, E. *Fundamentals of Computer Security Technology.* Prentice Hall, Englewood Cliffs, NJ, 1994.

[9]Postel, Jon. *Transmision Protocol. RFC 793.* Sept. 1981

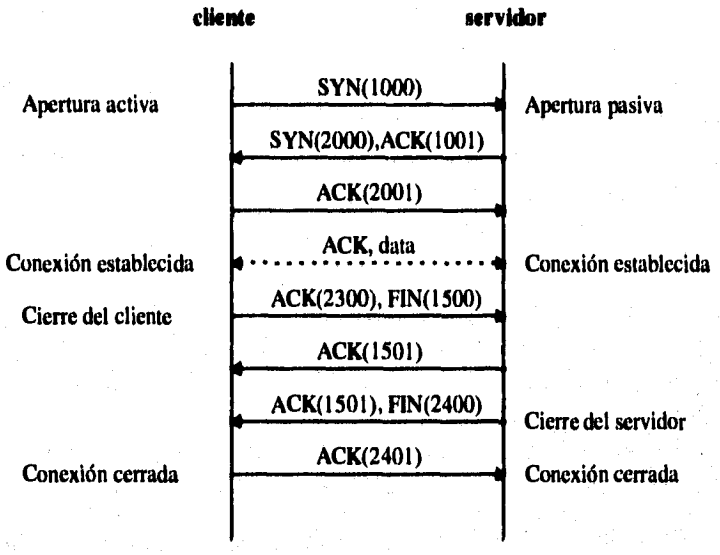


Fig. 3.15 ejemplo de una sesión TCP.

La figura es un ejemplo de una sesión TCP. El paquete inicial, con bit de sincronización encendido (SYN o petición de conversación), transmite el número de secuencia inicial para su lado de la conexión. El número de secuencia inicial es aleatorio. Todos los subsecuentes paquetes tienen el bit de reconocimiento encendido (ACK o acknowledge).

Cada mensaje TCP es marcado como proveniente de un host y de un número de puerto en particular, así como un puerto y host destinos; entonces el siguiente conjunto identifica de manera única a un circuito:

host-local, puerto-local, host-remoto, puerto-remoto

Los Servidores -procesos que ofrecen alguna aplicación vía TCP- escuchan a un puerto en particular. Por convención los puertos están numerados con los números más bajos. Los números de puertos de todos los servicios estándares son asumidos como conocidos por el que hace las peticiones.

Los Clientes hacen uso de los servicios ofrecidos. Un proceso cliente raramente pregunta por un número de puerto en particular en su host local, aunque le es permitido hacer eso. Normalmente le es adignado cualquier número de puerto libre que el sistema operativo le asigna.

Muchas versiones de TCP y UDP para UNIX refuerzan la regla de que sólo el super usuario (root) puede crear puertos con un valor menor al 1024. Estos son puertos privilegiados. La intención es de que los sistemas remotos puedan confiar en la autenticidad de la información escrita en dichos puertos. La restricción es sólo por convención y no es un requerimiento del protocolo.

D) UDP

Este protocolo (User Datagram Protocol)[10] se extiende a programas del mismo nivel de servicio empleado por IP. Hace la liberación en base del mejor esfuerzo, no existe corrección de errores, retransmisión o bien detección de pérdida, duplicamiento ni reordenamiento de los paquetes. Aunque existe opcionalmente la posibilidad de detección de errores.

Para compensar estas desventajas, existe mucho menos sobrecarga. En particular, no existe establecimiento de conexión. Cuando es empleado para transmisiones largas tiende a comportarse mal en la red.

UDP usa el mismo puerto y convenciones que TCP, pero en direcciones separadas. De manera similar, por lo regular, los servidores ocupan los puertos menormente numerados. No existe noción de circuito. Todos los paquetes destinados a un puerto dado son enviados al mismo proceso, independientemente de la dirección origen o el número de puerto.

E) ICMP

El Protocolo de Control de Mensajes Internet[11] es el mecanismo de más bajo nivel empleado para influenciar el comportamiento de las conexiones IP. Puede ser usado para informar a los hosts de una mejor ruta a un cierto destino, reportar problemas con respecto a una ruta o terminar una conexión por problemas en la red. También soporta la herramienta más importante de monitoreo a bajo nivel para los administradores del sistema y la red: el programa *ping*.

3.2.1.2 ENRUTADORES Y PROTOCOLOS DE RUTEO

Los protocolos de ruteo son mecanismos para encontrar dinámicamente las rutas apropiadas a través de la Internet. Ellos son fundamentales para la operación de TCP/IP. La información de ruteo establece dos rutas: de la máquina que llama a la de destino y de regreso, usualmente la segunda ruta es la inversa de la primera. Desde un punto de vista de seguridad la segunda ruta es la más importante. Cuando una máquina es atacada, ¿qué ruta toman los paquetes de vuelta, en teoría, para la máquina atacada?

3.2.1.3 SERVICIOS ESTÁNDARES

A) EL SISTEMA DE DOMINIO DE NOMBRES

El DNS (Domain Name System) es un sistema de bases de datos distribuidas que se usa para mapear nombres de hosts a direcciones IP y viceversa. En su forma normal de operar, los hosts envían solicitudes UDP a servidores DNS. Los servidores responden con

[10]Postel, Jon. *User Datagram Protocol. RFC 768.* 28 agosto 1980.

[11]Postel, Jon. *Internet control message protocol. RFC 792.* Sept. 1981

la respuesta correspondiente o bien con información acerca de servidores más inteligentes. Las solicitudes también pueden hacerse vía TCP, pero la operación TCP está normalmente reservada para la *zona de transferencias*. Dicha zona es usada por los servidores de respaldo para obtener una copia total de su porción del espacio de nombres (también es usada por los hackers para obtener rápidamente una lista de objetivos).

El espacio de nombres DNS es estructurado en forma de árbol. Por facilidades de operación, pueden delegarse subárboles a otros servidores.

B) SMTP

Cuando se le pregunta a cualquier compañía que aún no tiene una red instalada qué beneficios espera de la conexión con Internet, el correo electrónico encabeza la lista. Si hablamos de transporte de correo en la Internet, usualmente estamos hablando del SMTP (Simple Mail Transport Protocol)[12] [13].

SMTP transporta caracteres de texto ASCII de 7 bits empleando un protocolo simple y débil. No podemos estar seguros de quién envía el correo basado en SMTP.

Su implementación más común se encuentra en el *sendmail* [14]. Este programa está incluido en la mayoría de las distribuciones de software de UNIX. Consiste de miles de líneas de C y frecuentemente se ejecuta como *root*. (Un demonio* SMTP no necesita correr como *root* !!!)

C) Telnet

Telnet provee acceso simple de una terminal a una máquina. El protocolo incluye provisiones para manejar varios ambientes de terminal. Como regla, un demonio de *telnet* llama a *login* para autenticar e inicializar una sesión. El solicitante provee un nombre de cuenta y usualmente un contraseña para conectarse.

Una sesión de telnet puede ocurrir entre dos máquinas confiables. En este caso, un telnet seguro[15] [16]puede emplearse para encriptar la sesión completa, protegiendo la contraseña y el contenido de la sesión.

---

[12]Postel, Jon. *Simple mail transfer protocol RFC 821*. Agosto 1982

[13]Braden, Robert,editor. *Requirements for Internet hosts -application and support. RFC 1123*. Oct. 1989

[14]Costales, Bryan et al. *sendmail* O'Reilly and Associates, Sebastopol, CA, 1993.

* De la palabra inglesa *daemon* que inicialmente fue asociada a su significado mitológico y después racionalizada con el acrónimo "Disk And Execution MONitor" . Es un programa que es invocado explícitamente, pero se mantiene dormitando esperando que alguna(s) condición(es) ocurra(n). La idea es que el perpetrador de la condición no necesita estar al pendiente de que el demonio se encuentre acechando (aunque frecuentemente un progama ejecutará una acción sólo porque sabe que ésta invoca de manera explícita a un demonio).

Tomado de: The Free Dictionary of Computing en el URL: http://wombat.doc.ic.ac.uk/

[15]Borman, David, editor. *Telnet authentication option. RFC 1416*. Feb. 1993

D) NTP (Network Time Protocol)

Este protocolo[17] es un ayudante valioso para los gateways. Como su nombre lo indica, es empleado para sincronizar el reloj de una máquina con el mundo exterior. NTP cree en la noción de tiempo absoluto correcto, que es difundido a la red por máquinas con relojes atómicos o radio-relojes sintonizados con los servicios de sincronización del tiempo nacional. Cada máquina habla con una o más de sus vecinas, las comparaciones entre múltiples fuentes de información del tiempo permiten a los servidores de NTP descartar entradas erróneas; esto provee un alto grado de protección contra subversión deliberada.

E) Buscando gente

Dos protocolos estándares: *finger*[18]y *whois* [19]son comúnmente usados para buscar información sobre individuos.

El *finger* puede emplearse para obtener información ya sea de un usuario en particular o de usuarios conectados al sistema. Provee información personal, sobre cuándo fue la última vez que se uso la cuenta, cuándo fue la última vez que se conectó el usuario, el nombre y la dirección electrónica del usuario.

El protocolo *whois* provee información para contactar.

3.2.1.4 PROTOCOLOS BASADOS EN RPC

A) RPC y el mapeador de puerto

El protocolo RPC de Sun (Remote Procedure Call), es la base de muchos de los servicios más recientes. Desafortunadamente, muchos de estos servicios representan un problema potencial de seguridad.

El concepto básico de RPC es algo sencillo. Una persona creando un servicio de red emplea un lenguaje especial para especificar los nombres de los puntos de entrada externos y sus parámetros. Un lenguaje precompilador convierte estas especificaciones en bloques o rutinas unidas para módulos del cliente y de servidor. Con la ayuda de esta unión el cliente puede hacer llamadas a rutinas de manera "ordinaria" a un servidor remoto. La mayoría de las dificultades de programar en red son enmascaradas por la capa RPC.

---

[16]Safford, David R. et al.*Secure RPC authentication (SRA) for TELNET and FTP.* In Proceedings of the Fourth Usenix UNIX Security Simposium. p 63-67, Santa Clara CA. Oct. 1993

[17]Mills, David. *Network time protocol (version 3) specification, implementation and analysis. RFC 1305.* Marzo 1992

[18]Harrenstien, Ken. *NAME/FINGER protocol. RFC 742.* Dec 30, 1977

[19]Harrenstien Ken and White,Vic. *NICNAME/WHOIS. RFC 812* Marzo 1, 1982

Los mensajes RPC comienzan con su propio encabezado. Incluyen el número de programa, el número de procedimiento denotando el punto de entrada dentro del procedimiento y algunos números de versión. Cualquier intento de filtrar los mensajes RPC debe hacerse en estos campos. El encabezado también incluye un número de secuencia que se emplea para hacer coincidir las peticiones con las respuestas.

El mapeador de puertos (*portmapper*), que en sí mismo emplea al protocolo RPC para comunicación, actúa como un intermediario entre los servidores y los clientes RPC. Para contactar a un servidor, el cliente pregunta primero al mapeador en el host servidor sobre el número de puerto y el protocolo (UDP o TCP) del servicio. Esta información es entonces usada por la actual llamada RPC.

B) NIS

Una de las aplicaciones más peligrosas de RPC es el Servicio de Información de la Red (Network Information Service), normalmente conocida como YP (Yellow Pages). NIS es usado para distribuir una variedad de bases de datos importantes de un servidor central a sus clientes. Estas incluyen el archivo de contraseñas, la tabla de direcciones de hosts, y las llaves privada y pública de las bases de datos empleadas por el RPC seguro. El acceso puede llevarse a cabo por una clave de búsqueda o transferirse el archivo completo.

C) NFS

El Sistema de Archivos de Red (Network File System)[20][21], fue originalmente desarrollado por Sun Microsystems y actualmente es soportado en muchas computadoras. Es un componente vital de la mayoría de las estaciones de trabajo, permite compartir disco y no se visualiza que pronto vaya a desaparecer.

Por robustez, NFS está basado en RPC, UDP y otros servicios locales. Esto es, el servidor NFS -el host que generalmente tiene el disco de almacenamiento real- maneja cada solicitud independientemente. Por lo mismo, todas las operaciones deben ser autenticadas individualmente.

La herramienta básica es el manejador de archivo, una cadena única que identifica cada archivo o directorio en el disco. Todas las peticiones NFS son especificadas en términos de un manejador de archivo, una operación, y cualesquiera parámetros que sean necesarios para esa operación. Las peticiones que otorgan acceso a nuevos archivos, como *open*, devuelven un nuevo manejador al proceso cliente. Los manejadores de archivos no son interpretados por el cliente. El servidor los crea con la suficiente estructura para sus propias necesidades; muchos de los manejadores de archivo incluyen un componente aleatorio también.

---

[20]Sun Microsystems, *NFS: Network file system protocol specification. RFC 1094.* Marzo 1989.

[21]Sun Microsystems. *Network iInterfaces Programmer's Gulde.* Mountain View, CA, Marzo 1990. SunOS 4.1

El manejador inicial para el directorio de root de un sistema de archivos es obtenido en tiempo de montaje. El demonio de montaje, un servicio basado en RPC, verifica el nombre de host del cliente, del usuario y el archivo de sistema solicitado contra una lista de administración, y verifica el modo de operación (sólo lectura contra lectura/escritura). Si todo está bien, el manejador de archivo para el directorio raíz del sistema de archivos es enviado al cliente.

Los manejadores de archivo son asignados normalmente en tiempo de creación del sistema de archivos, vía un generador de números seudo aleatorios. Los nuevos manejadores sólo pueden ser escritos a un sistema de archivos no montado. Antes de hacer esto, cualquier cliente que tiene el sistema de archivos montado debe desmontarlo.

Normalmente los servidores NFS viven en el puerto 2049. La elección del número de puerto es problemática, puesto que se encuentra en un rango "no privilegiado", y por consiguiente está en el rango de asignación a procesos ordinarios. Algunas versiones de NFS viven en puertos aleatorios, con el mapeador de puertos proveyendo la información de direccionamiento.

D) Andrew

El Sistema de Archivos Andrew (AFS)[22] [23]es otro sistema de archivos de red que puede, en cierto grado, interoperar con NFS. Su principal propósito es proveer un sistema de archivos simple, escalable, independiente de su localización, a una organización, o incluso a la Internet. AFS permite a los archivos vivir en cualquier servidor dentro de la red.

3.2.1.5 PROTOCOLOS DE TRANSFERENCIA DE ARCHIVOS

A) TFTP

TFTP (Trivial File Transport Protocol) es un mecanismo simple de transferencia de archivos basado en UDP. El protocolo no tiene autenticación. Es frecuentemente empleado para inicializar estaciones de trabajo y terminales X11.

Un demonio TFTP configurado apropiadamente restringe la transferencia de archivos a uno o dos directorios (típicamente /usr/local/boot) y la librería de fonts de X11.

[22]Howard, John H. *An overview of the Andrew File System*. In USENIX Conference Preceedings, p 23-26, Dallas TX, 1988.

[23]Kazar, Michael Leon. *Syncronization and caching issues in the andrew file system*. In USENIX conference Proceedings, p 27-36, Dallas TX, 1988

B) FTP

El protocolo de transferencia de archivos (File Transfer Protocol)[24] soporta la transmisión de texto en caracteres y de archivos binarios.

El servidor usa el puerto 20 para este fin. Por omisión el cliente usa el mismo número de puerto que es empleado para el canal de control. La especificación del protocolo FTP sugiere que se cree sólo un canal y se mantenga abierto durante toda la sesión de transferencia de archivos. Además, debido a una de las más oscuras propiedades de TCP (el estado *TIMEWAIT*, para quien desee investigar más a profundidad), un diferente número de puerto debe ser empleado cada vez. Normalmente el cliente escucha en un número de puerto aleatorio, y le informa al servidor vía el comando *PORT*. En su turno, el servidor hace la llamada a dicho puerto.

Por omisión las transferencias se hacen en modo ASCII, pero con el comando **BINARY** se puede hacer la transferencia de archivos binarios.

El programa *anonymous ftp* es el mecanismo de distribución de datos más importante. Permite a cualquier usuario externo tomar archivos de un área restringida del sistema sin una autorización. Por convención los usuarios se conectan con *anonymous* como usuario. Algunos lugares solicitan que el usuario introduzca su dirección de correo electrónico como contraseña, otros sólo *guest*.

C) FSP

FSP, el Protocolo de Transporte de Archivos (File Transport Protocol), que por cierto sus siglas nada tienen en relación con su nombre, es otro protocolo de transferencia de archivos. Emplea el puerto UDP (frecuentemente el puerto privilegiado 21) para implementar un servicio similar a FTP. No es un protocolo oficial y casi se encuentra en desuso.

3.2.1.6 LOS COMANDOS "R"

Los comandos "r" confían en el mecanismo de autenticación de BSD (Berkeley Software/Standard Distribution). Uno puede hacer un *rlogin* a una máquina remota sin introducir un contraseña si se encuentra el criterio de autenticación.

La llamada debe originarse de un puerto TCP privilegiado. En otros sistemas (como las Pcs) no hay dicha restricción. Un corolario de esto es que tanto las llamadas a *rlogin* como *rsh* (abre un shell remoto) deben permitirse sólo de máquinas donde se refuerce esta restricción.

El usuario solicitante así como su máquina deben de encontrarse en la lista de compañeros confiables de la máquina destino (típicamente /etc/hosts.equiv).

[24]Postel, Jon and Reynolds,Joyce. *File Transfer Protocol. RFC 959*. Oct. 1985

El nombre del solicitante debe corresponder con su dirección IP.

3.2.1.7 SERVICIOS DE INFORMACIÓN

A) World Wide Web

El crecimiento de lo que sería el mejor término para los llamados *protocolos de información* ha sido explosivo. Estos incluyen al *gopher*[25], *Wide Area Information Services* (WAIS), y otros asociados en ocasiones bajo el rubro de *World Wide Web* (WWW). Aunque ellos difieren en detalle grandemente, existen puntos esenciales de similitud en cuanto a su operación.

Qué son los WWW, según CERN (los creadores del web)[26]:

"El World Wide Web (W3) es el universo de información accesible por la red, todo un mundo de conocimiento humano. Es una iniciativa que empezó en CERN, ahora con muchos participantes. Tiene un cuerpo de software, y un conjunto de protocolos y convenciones. W3 usa hipertexto y técnicas de multimedia para hacer a la red fácil de recorrer, buscar y contribuir."

El web está compuesto por distintos compontes interoperando: *Servidores*, en los cuales los distintos sitios de la Internet pueden exportar datos al mundo; *Clientes*, donde los usuarios pueden navegar en la red (dos navegadores populares son Mosaic y Lynx); y los *Agentes*, que son empleados para facilitar la comunicación y proveer control de acceso a los lugares que deben confiar en un host intermediario para la comunicación con la Internet, lugares como los que se encuentran detrás de un firewall.

El protocolo usado es el HTTP, o HyperText Transfer Protocol. Este define un lenguaje para accesar a los servidores de web. Otro término es el URL (Uniform Resource Locator), usado para nombrar un recurso de la red. Un URL toma la forma de "recurso://server.host.nombre[:puerto]/[ruta]", por ejemplo: http://www.tis.com

Generalmente, un host hace contacto con un servidor, envía una solicitud o un apuntador a información, y recibe una respuesta. La respuesta puede ser un archivo a desplegarse o un apuntador o un conjunto de ellos en algún otro servidor.

Algunas veces, los apuntadores enviados son una dirección de host y un puerto o un pequeño diálogo de conexión.

---

[25] Anklesaria, Farhad et al. *The Internet gopher protocol (A distributed document search and retrieval protocol) RFC 1436.* Marzo 1993

[26] Dalva, David Y. *Security and the World Wide Web.* Home page of TIS: http://www.tis.com/Home/Network Security/WWW/Article. June 1994

B) NNTP

El NNTP (Network News Transfer Protocol)[27] es usualmente conocido como *Netnews*. El diálogo es similar al usado por SMTP. Existen algunas desavenencias sobre como debe pasar NNTP a través de las paredes de fuego (firewalls).

El camino obvio es tratarlo como un simple correo. Esto es los artículos de novedades deben ser procesados y confiados al gateway. Pero existe un buen número de desventajas en esta actividad.

Primero que todo, las novedades en red son un recurso sucio. Consume grandes cantidades de espacio en disco, tiempo de CPU, etc. Implican un esfuerzo administrativo, y representan una gran cantidad de software del que debe preocuparse el administrador del gateway.

C) Multicasting y el Mbone

Multicasting es la generalización de *unicast* y *broadcast*. En vez de que un paquete sea enviado sólo a un destino o a todos los destinos de la red, un paquete *multicast* es enviado a un subconjunto de estos destinos, en el rango de un host a todos los hosts.

Debido a que muchos enrutadores comerciales no soportan todavía el *multicasting*, algunos hosts son usados para este fin. Ellos hablan un protocolo de ruteo especial, el *Distance Vector Multicast Routing Protocol* (DVMRP).

Los enrutadores *multicast* hablan entre ellos encapsulando el paquete entero, incluyendo el encabezado IP, en otro paquete IP, con la dirección destino normal. Cuando el paquete llega a la máquina destino, se desencapsula el paquete. El paquete es entonces enviado a otros enrutadores *multicast* y transmitido a sus redes locales. Los destinos finales son generalmente puertos UDP.

Un buen número de aplicaciones interesantes de la red usan el *MBone* -el soporte de multicast en la Internet- para alcanzar grandes audiencias.

Por convención, los puertos asignados dinámicamente al MBone están en el rango del 32769 al 65535.

3.2.1.8 EL SISTEMA X11

X11[28] es hoy en día el sistema de ventanas cliente-servidor. Emplea a la red para comunicaciones entre aplicaciones y los dispositivos de entrada y salida (la pantalla, el ratón, etc.), que permite a las aplicaciones residir en diferentes máquinas. Esta es la fuente de mucho del poder de X11. Aunque es también la fuente de un gran peligro.

---

[27]Kantor, Brian and Lapsley, Phil. *Network News transfer Protocol. RFC 977.* Feb. 1986

[28]Scheifler, Robert and Gettys, James. *X Window System.* Digital Prewss, Burlington, MA 3rd de, 1992.

El concepto fundamental de X11 es la noción desconcertante de que la terminal del usuario es un servidor. Esto es un poco lo contrario al patrón normal, en el que las máquinas de los usuarios son los clientes, solicitando servicios vía red a ciertos servidores. En X11, la estación local controla todo lo referente a la interacción con los dispositivos. Las aplicaciones hacen llamadas a este servidor cuando desean hablar al usuario. No importa cómo son invocadas estas aplicaciones; el sistema de ventanas no necesita meter mano en su creación. Si conocen los *tokens* mágicos -la dirección de red del servidor- ellos pueden conectarse y tomar control del servidor.

Las aplicaciones conectadas a un servidor X11 pueden hacer toda clase de cosas. Pueden detectar pulsaciones de teclas, bajar el contenido de la pantalla, generar presiones de teclas para aplicaciones que se los permitan, etc. En otras palabras, si un enemigo está conectado a nuestro teclado, podemos decirle adiós a nuestras posesiones. Sin embargo, existe la posibilidad de ponerle al teclado un seguro para evitar esta situación.

3.3. Sistemas Operativos

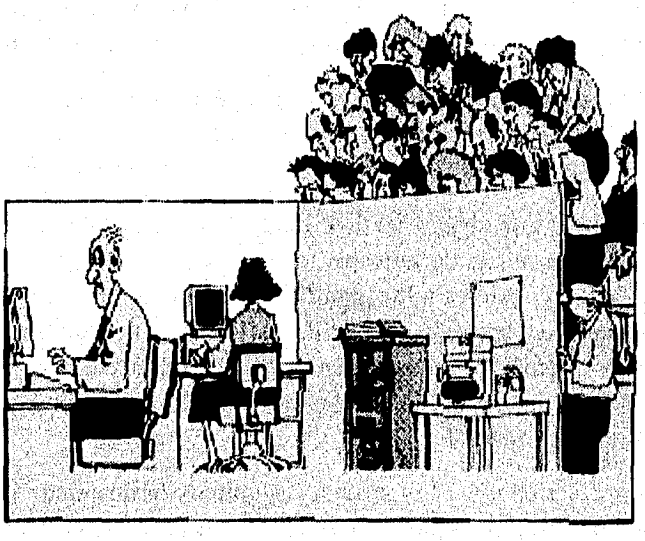
Otro factor que influencía a la seguridad es la elección de sistema operativo [29]- en particular los sistemas operativos de la red. Algunos tienen mayor seguridad que otros, pero el problema primario, desde una perspectiva de red, es la necesidad de acomodoar múltiples sistemas operativos. Esto acarrea problemas de compatibilidad y consistencia. Por ejemplo, una área emplea una versión X de Unix con un conjunto de implementaciones de seguridad y otra usa una versión segura Y, entonces cada área trabaja de manera diferente a la otra, lo que Brian Redler de la NSCC[30] recomienda es emplear paquetes de seguridad de terceros, esto puede costar un poco más pero se tiene consistencia.

Tomando en cuenta lo anterior podremos imaginar el problema que implica el manejo de la redUNAM donde no sólo se emplean diferentes sistemas operativos (UNIX, VMS, DOS, Netware) sino de diferentes proveedores de ellos y versiones distintas - escencialmente de UNIX: ULTRIX, HPUX, Solaris, etc.

[29]Russel Kay .*Distributed and Secure.BYTE*, June 1994, p.178

[30]Ibid

CAPITULO 4



Enfoque general de los problemas de seguridad y propuestas de soluciones

En este capítulo describiremos los principales problemas que se pueden presentar en una red típica TCP/IP. A continuación de cada tema se proponen soluciones que dependiendo de las características, propósitos y recursos de nuestra red pueden o no aplicarse. Al final del capítulo se mencionarán soluciones que no son defensas para un problema en particular sino para un conjunto de ellos o que colaboran como obstáculo para varios riesgos en la seguridad.

4.1. Problemas de Seguridad en el Protocolo TCP/IP[1] y soluciones

4.1.1 INTRODUCCIÓN

El protocolo TCP/IP, cuyo uso se encuentra ampliamente difundido en nuestros días, ha sido desarrollado bajo el patrocinio del Departamento de Defensa de los Estados Unidos. A pesar de eso, existe un alto número de fallas de seguridad inherentes a los protocolos, independientemente de lo correcto de cualquier implementación.

Algunas de estas fallas existen debido a que los hosts dependen de las direcciones origen IP para la autenticación. Otras se deben a los mecanismos de control de la red, y en particular a los protocolos de ruteo, cuya autenticación es mínima o no existe.

En las partes que siguen se describirán la variedad de ataques basados en esas fallas: la burla de secuencia de números, engaño en las direcciones origen y ataques de autenticación. Los enrutadores y los protocolos de enrutamiento son descritos como parte de la problamática de seguridad TCP/IP. Por último, se describen los problemas de seguridad en los protocolos que utilizan RPC y Portmapper, que aunque algunos son parte del Sistema Operativo Unix, por su estrecha relación con RPC, se decidió describirlos desde este apartado.

Cuando describimos a los ataques, asumimos básicamente que el atacante tiene más o menos un control completo sobre alguna máquina conectada a la Internet. Esto puede deberse a fallas en la protección de los mecanismos de esa máquina, o porque la máquina es una microcomputadora, y por lo tanto está desprotegida. Incluso, el atacante puede ser un administrador del sistema.

Cabe aclarar que se tratarán los problemas genéricos de los protocolos y no fallas particulares de la implementación de éstos. Como se verá posteriormente la cuidadosa implementación de técnicas pueden aliviar o prevenir algunos de éstos problemas.

4.1.2 CAPA DE RED

4.1.2.1 ARP -ADDRESS RESOLUTION PROTOCOL

ARP es un protocolo que se utiliza para obtener la resolución de direcciones (dirección MAC a dirección IP). Los paquetes ARP son enviados vía *broadcast* en la red Ethernet. Los dispositivos de la capa física y de enlaces de datos no entienden la dirección IP, solo entienden la dirección MAC. El paquete ARP contiene un requerimiento de dirección IP asociada a la dirección MAC que conoce del nodo destino. El nodo destino o cualquier otro dispositivo activo en la red capaz de desempeñar esta

[1] Bellovin, Steven.M. *Security Problems in the TCP/IP Protocol Suite*, Computer Communication Review, Vol 19 No. 2, pp. 32-48, Abril 1989 (Tomado con ftp de ftp.research.att.com en /dist/internet_security/ipext.ps.Z)

tarea (un enrutador) contesta el requerimiento completando la información para tener la pareja de direcciones asociadas al nodo: IP y MAC. Este par de direcciones son almacenadas en la memoria caché del nodo fuente para evitar sobrecarga ARP en la red.

En este sentido existe un punto vulnerable en cuanto a seguridad se refiere. Si un nodo no confiable tiene acceso con permiso de escritura en la red local a información de la inter-red este nodo puede emitir mensajes ARP falsos y desviar el tráfico hacia sí mismo, de esta forma podría imitar la conexión a él validándose como nodo confiable, de esta manera se realiza uno de los principales ataques: impersonar a otro host, o simplemente modificar, alterar o destruir las cadenas de datos mientras pasan.

El proceso de la transmisión de requerimientos ARP en una red son generalmente automáticos. En una red especial, con propósitos de seguridad, los *mapeos* de dirección MAC a dirección IP pueden ser definidos estáticamente vía *hardware* y suprimir el paso de los paquetes ARP que son generados automáticamente por el protocolo, esto con el fin de prevenir interferencias.

4.1.3 CAPA INTERED

4.1.3.1 IP

IP son un conjunto de datos que forman la base de la familia de protocolos TCP/IP. El datagrama IP, esta compuesto por varios campos, de los cuales los siguientes son los que para fines de seguridad nos importan: dirección fuente (32 bits), dirección destino (32 bits) y un campo opcional.

Etiquetas IP.

Cómo se mencionó, IP tiene un número de campos opcionales para etiquetar los paquetes de acuerdo al nivel de seguridad en que están clasificados.

El campo de seguridad de IP es actualmente usado por organizaciones militares, sobre la utilización e interpretación de este campo, han surgido variaciones que ya se emplean también en aplicaciones comerciales.

Cada paquete es etiquetado de acuerdo al nivel de sensibilidad de la información que contienen. La etiqueta describe dos componentes: un *componente de Jerarquía* (Secreto y Ultra Secreto) y una *Categoría* opcional (nuclear, armas, criptografía, etc).

Las etiquetas indican prácticamente el nivel de seguridad de los últimos procesos - emisor y receptor. Un proceso no puede escribir o comunicarse con otro proceso de un nivel de seguridad más bajo que en el que esta clasificado, por que se permitiría descubrir o acceder a información no autorizada. Por razones obvias un proceso no puede leer de un medio que contiene información de un nivel más alto de seguridad. La combinación de las dos restricciones anteriormente descritas estipulan por norma de seguridad que dos procesos que están intercambiando información deben ser exactamente del mismo nivel de seguridad.

Dentro de una red el propósito principal de las etiquetas de seguridad es "forzar" decisiones de enrutamiento. En primer término, un paquete marcado como tráfico "Ultra Secreto" no puede ser transmitido por un enlace inseguro,es decir un enlace por el que generalmente se transmite tráfico "Secreto", y como segundo propósito, se tiene que controlar el equipo de cifrado, esto es, un paquete etiquetado como tráfico "Ultra Secreto" puede ser enviado a través de un enlace inseguro si está propiamente encriptado dentro de un algoritmo.

4.1.3.2 PROTOCOLO DE CONTROL DE MENSAJES INTERNET

El ICMP (Internet Control Message Protocol)[ii] es la herramienta básica de manejo del protocolo TCP.

El ICMP puede usarse para informar a los hosts de una mejor ruta a un destino, reportar un problema con alguna ruta, o terminar una conexión por problemas en la red. También soporta la herramienta más importante a bajo nivel para monitoreo: el programa *ping*.

Debido a lo anterior podríamos pensar que es objeto de muchos abusos. Sorprendentemente los ataques ICMP son bastante difíciles, de cualquier forma existen agujeros que pueden explotarse.

El primero y el más obvio es la Redirección de mensajes ICMP que es empleado por los enrutadores para informarle a los hosts sobre mejores rutas.

Supongamos que un intruso ha penetrado en un enrutador secundario disponible para un host destino, pero no el primario (es suficiente penetrar a un host ordinario en la red local de destino y pretender ser un enrutador). Asumamos que el intruso desea establecer una ruta falsa para un host confiable T a través del ya comprometido enrutador secundario. Puede entonces seguirse la siguiente secuencia: enviar un paquete de abertura TCP al host destino, pretendiendo ser T; el destino responderá con su propio paquete de abertura, ruteándolo a través del enrutador primario seguro, y haciendo referencia a la falsa conexión; este paquete aparecerá como un mensaje de control legítimo, de esta forma el cambio de ruteo que contiene será aceptado; si el host destino hace el cambio global a sus tablas de ruteo, el intruso aprovechará la conexión por la ruta oculta para burlar al host T.

Algunos hosts no desempeñan suficientes validaciones en la redirección de mensajes ICMP. El mensaje *Redirect* debe ser obedecido únicamente por nodos no enrutadores y únicamente cuando el mensaje proviene de un enrutador en una red directamente conectada, sin embargo no todos los administradores de los enrutadores son cuidadosos en este sentido, por lo que es posible abusar de ICMP para crear nuevas rutas hacia un destino. Si eso sucede las normas de seguridad no están siendo cumplidas

[ii]Postel, J *Internet Control Message Protocol. RFC 792*, 1981

Los mensajes ICMP recibidos en un host especifican el estado actual de una conexión particular, es decir envían información de la máquina especificada por el mensaje. En estos casos, el encabezado IP y los primeros 64 bits del encabezado de la capa de transporte incluyen el mensaje ICMP. La intención de este mensaje es limitar la visión de los cambios en la topología de la red. Los mensajes *Redirect* o *Destination Unreachable* deberían en todos los casos especificar el estado de la conexión en la cual están siendo transmitidos los mensajes ICMP. Desafortunadamente los versiones anteriores de ICMP no utilizan esta información adicional. Cuando uno de estos mensajes llega, todas las conexiones entre dos host serán afectadas por la información del mensaje. Por ejemplo, si un paquete enviado a un host "X" regresa al nodo fuente con un mensaje *Destination Unreachable*, todas las conexiones habilitadas al host "X" serán dadas de baja. Este mensaje tendrá la misma validez aún si el mensaje es regresado por un filtro "firewall", por lo tanto es considerado como política de firewalls contener mensajes ICMP que permitan derribar llamadas legítimas originadas desde la misma máquina.

4.1.4 CAPA DE TRANSPORTE

4.1.4.1 TCP -TRANSPORT CONTROL PROTOCOL

Predicción de los Números de Secuencia TCP

TCP (Transport Control Protocol) provee circuitos virtuales confiables para los procesos del usuario[iii]. Los paquetes perdidos o dañados son retransmitidos; los paquetes que van arribando se reacomodan, si es necesario, hasta coincidir el orden de transmisión original.

El orden se mantiene mediante números de secuencia en cada paquete. Cada byte enviado, así como las peticiones para abrir y cerrar el circuito, son numerados individualmente. Todos los paquetes excepto el primer paquete TCP enviado durante la conversación contiene un número de aceptación (acknowledgment) que da el número de secuencia del último byte recibido exitosamente.

Cada mensaje TCP es marcado como proveniente de un host en particular y de un número de puerto, y hacia un puerto y host destinos.

Uno de los más fascinantes agujeros de seguridad fue descrito primeramente por Morris[iv]. De manera breve podemos decir que utilizó la predicción de secuencia de números TCP para construir una secuencia de paquete TCP sin recibir ninguna respuesta del servidor. Esto le permite engañar a un host "confiable" en una red local y hacerle creer que está comunicándose con una máquina confiable.

[iii] Cheswick, William R. and Bellovin, Steven M. *FireWalls and Internet Security*. Addison Wesley Publishing, USA, 1994 págs. 23 y 24

[iv] Morris, R.T. 1985 *A Weakness in the 4.2BSD UNIX TCP/IP Software*. Computing Science Technical Report No. 117, AT&T Bell Laboratories, Murray Hill, New Jersey

La secuencia para el establecimiento de una conexión TCP normal implica tres caminos, esto es el mensaje tendrá que pasar por tres manos diferentes. El cliente selecciona y transmite un número de secuencia inicial ISN_C, el servidor lo reconoce y envía su propio número de secuencia ISN_S y el cliente reconoce a éste y envía los datos. Siguiendo estos tres mensajes, la transmisión de datos toma lugar. El intercambio puede mostrarse esquemáticamente como sigue:

C->S: SYN (ISN_C)

S->C: SYN (ISN_S), ACK(ISN_C)

C->S: ACK(ISN_S)

C->S: datos

y/o

S->C: datos

Esto es, para que una conversación tenga lugar, C debe primero escuchar ISN_S, un número más o menos aleatorio.

Supóngase, sin embargo, que hubiera un camino para un intruso X para predecir el ISN_S. En ese caso, éste podría enviar la siguiente secuencia para hacerse pasar por un host confiable T:

X->S: SYN(ISN_X),SRC=T

S->T: SYN(ISN_S),ACK(ISN_X)

X->S: ACK(ISN_S),SRC=T

X->S: ACK(ISN_S),SRC=T, datos-intrusos

Aunque el mensaje S->T no va hacia X, X fue capaz de conocer su contenido, y por lo tanto enviar datos. Si X llevara a cabo este ataque en una conexión que permite ejecución de comandos (por ejemplo los comandos "r"), podrían ser ejecutados comandos maliciosos.

Cabe entonces la pregunta: ¿cómo predecir el aleatorio ISN?, en algunos sistemas, la variable del número de secuencia inicial es incrementado por una cantidad constante cada segundo, y a la mitad de esa cantidad de tiempo se inicia una conexión. De esta manera, si uno inicia una conexión legítima y observa el ISN_S usado, uno puede calcular, con un alto grado de confiabilidad, el ISN'_S que se empleará en el siguiente intento de conexión.

A) Defensas para predicción de secuencia de números TCP

Si parte del problema es el tiempo que pasa para incrementar el número de secuencia, entonces parece una solución disminuir este lapso. Si asumimos que la

estabilidad es suficientemente buena probablemente podemos reducirlo a unos 10 milisegundos aproximadamente. Sin embargo, es claro que la Internet no cuenta, en general, con la suficiente estabilidad.

Otra solución es: hacer aleatorio el incremento, teniendo el cuidado de emplear suficiente número de bits, un generador de 32 bits es sugerible. Debe asegurarse que el generador sea confiable de manera que sea difícil al atacante encontrar la semilla. Una buena opción es emplear un algoritmo o dispositivo criptográfico para la generación de los ISN_S . El DES (Data Encription Standard) es una opción atractiva. De cualquier forma debe de tenerse cuidado al elegir la clave empleada, por ejemplo la hora en que se hace la inicialización (boot), asegurándose de que esta hora esté encriptada con una clave secreta.

Sin embargo hay dos puntos sobre este caso en particular, que vale la pena discutir. Primero, el *Ataque de Morris*, depende en su totalidad de crear una conexión legítima hacia la máquina objetivo. Si ésta es bloqueada, por un firewall por ejemplo, el ataque no tendría éxito. En forma paralela, si los servidores confían toda la infraestructura de seguridad hacia su mismo software, ésta depende totalmente de su configuración (un riesgo más, donde se involucra el factor humano como pilar de la infraestructura). Por otro lado, el concepto de ataque al número de secuencia puede ser generalizado, muchos otros protocolos aparte de TCP son vulnerables a este tipo de ataques[v]. De hecho, la implementación en TCP del algoritmo *three-way handshake* (ver Apéndice A) al momento de establecer la conexión, provee una ayuda hacia este tipo de ataques.

4.1.4.2 UDP (USER DATAGRAM PROTOCOL)

Este protocolo[vi] extiende el mismo nivel de servicio usado por IP a programas de aplicación. No existe corrección de errores, retransmisión o detección de pérdidas, duplicados o reordenamiento de paquetes. Aunque la detección de errores es opcional.

Como compensación de estas desventajas, existe mucho menos sobrecarga, en particular no hay establecimiento de conexión, esto hace a UDP apropiado para aplicaciones pregunta/respuesta, donde el número de mensajes intercambiados es corto en comparación con los que incurre TCP.

UDP emplea el mismo número de puerto y convenciones para el servidor como lo hace TCP, pero los segmentos de memoria detinados a estos puertos son diferentes. De manera similar los servidores usualmente, no siempre, habitan los puertos de menores números. No existe a la implementación de circuitos virtuales, debido a que no provee un servicio orientado a conexión: los paquetes llegan al nodo destino sin importar el nodo fuente o la ruta por la cual llegó.

---

[v] Bellovin, Steven.M. *Security Problems in the TCP/IP Protocol Suite*, Computer Communication Review, Vol 19 No. 2, Abril 1989 (Tomado con ftp de ftp.research.att.com en /dist/internet_security/ipext.ps.Z)

[vi] Postel, Jon *User datagram Protocol. RFC 768*, 28 agosto 1980

El peligro radica en que es más fácil burlar a los paquetes UDP que a los TCP, ya que no hay "saludos" o números de secuencia.

4.1.5 CAPA DE APLICACION

4.1.5.1 SMTP -SIMPLE MAIL TRANSFER PROTOCOL

El protocolo utilizado para correo electrónico en Internet es SMTP[vii].[viii] SMTP Transporta caracteres tipo texto de 7 bits, utilizando una estructura de protocolo muy simple. A continuación se muestra un ejemplo de como funciona SMTP:

<--- 220 red.dgsca.unam.mx SMTP

----> HELLO A.SOME.EDU

<--- 250 red.dgsca.unam.mx

----> MAIL FROM:<lli.segip.A.SOME.EDU>

<--- 250 OK

---> DATA

<-- 354 Start Mail input; end with <CRLF>.<CRLF>

---> From lli.segip.A.SOME.EDU Thu Jul 27 23:35:05 EST 1995

---> From: lli.segip.A.SOME.EDU

---> To: Faby security.unam.mx

---> Date: Thu Jul 27 23:35:05 EST 1995

--->

---> Cita proxima revisión después de la sesión.

--->

---> lliana

---> .

<--- 250 OK

.... A.SOME.EDU!lli.segip sent 273 bytes to security.unam.mx!mark.farkle

---> QUIT

[vii] Postel, Jon. *Simple mail transfer protocol. RFC 821.* Agosto 1982

[viii] Braden, Robert, editor. *Requirements for Internet hosts -application and support. RFC 1123.* Octubre 1989

<--- 221 red.dgsca.unam.mx Terminating

En el ejemplo anterior el hosts remoto A.SOME.EDU, esta transfiriendo correo electrónico hacia la máquina local red.dgsca.unam.mx. Las máquinas administradoras de correo aprenden de esta información para realizar sus funciones. Los hackers también pueden aprender de estos comandos para realizar funciones ilícitas en la red.

En el ejemplo anterior se muestra que el host que realiza la llamada, especifica una dirección de regreso en el comando MAIL FROM. En este nivel (de la capa de protocolos), el protocolo no especifica que la máquina verifique esta dirección de regreso, en ese caso, no se puede estar seguro acerca de quien mandó el correo. Pero es posible implementar la opción . Sin embargo, para poder implementar mayor seguridad es necesario el apoyarse en un protocolo de más alto nivel.

Una organización que utiliza correo electrónico podría, tomando en cuenta sus propias restricciones, centralizar las funciones administrativas en un servidor de correo electrónico, administrado por una persona experta en correo electrónico, de esta forma se eliminan los riesgos que se correrían al permitir manipular a los usuarios finales la administración de correo. La lista que contiene los *alias* de los usuarios para correo electrónico deben estar cuidadosamente resguardados para evitar espionaje industrial. Desde el punto de vista de seguridad, los comandos:

VRFY <postmaster>

VRFY <root>

que traducen alias de correo a la actual cuenta (*login*), son un punto vulnerable. Con estos comandos se puede obtener información acerca de quien es el administrador y cuáles son las cuentas a las que se tendría acceso por medio de correo. Es tarea del administrador del correo electrónico, alojar la lista de alias de correo electrónico en un lugar del sistema de archivos seguro.

La implementación de SMTP esta contenida dentro de *sendmail*.[ix] Este programa viene incluido en la mayoría de los sistemas UNIX. *Sendmail* ha sido parte de la historia de violaciones al principio de "seguridad mínima", y de agujeros de seguridad. Contenida en el agujero de seguridad usado por el "Internet Worm" [x], mencionado en el artículo de - New York Times [xi].

Para la mayoría de los mecanismos barreras, el mayor problema con *Sendmail*, es la configuración del mismo. Las reglas de configuración son difícilmente comprendidas por los usuarios normales del sistema. Aún cuando es relativamente fácil sobreescribir

---

[ix] Costales, Bryan et al. *Sendmail* O'Really and Associates, Sebastopol, CA. 1993.

[x] Spafford, Eugene, *An analysis of the Internet worm. Sep 1989.* Disponible con ftp anónimo a ftp.cs.purdue.edu en /pub/spaf/security/IWorm.PS.Z .

[xi] Markoff, John. Computer invasion: 'back door' ajar. In New York Times, volume CXXXVIII, p. B10, 1989.

reglas de correo electrónico, en System V, puede ser difícil de comprender que hace y como lo hace. El RFC 822 y 1123 proporcionan información acerca de este tema.

Los problemas de seguridad que ocasiona el programa *sendmail*, pueden ser minimizados. De hecho si *sendmail* no está realizando funciones de entrega y distribución de correo (como lo haría una máquina servidora de correo electrónico), no necesita ser habilitado. Si no está realizando las funciones anteriormente descritas, y no se corre bajo la clave de superusuario, no se necesita permiso de escritura en el directorio de *spool* (generalmente localizado en el subdirectorio /var/spool/mqueue), el permiso de lectura del subdirectorio /dev/kmem, puede ser determinado por el actual programa que está llamando a los subprogramas, y de alguna manera obligar al puerto 25 a atender la llamada. Esto último es más fácil de implementar al correrlo bajo el programa *inetd*, de esta forma se evita que sendmail sea resultado de la llamada del programa *bind*.

Por otra parte, el contenido de la información del mensaje, es decir, el cuerpo del correo, puede ser un punto de vulnerabilidad para la seguridad del sistema, si en esta información estén contenidos reportes, estadísticas o información confidencial de la información.

Si una máquina está corriendo en forma automática el programa MIME (Mutltipurpose Internet Mail Extensions), se tiene un problema de seguridad en potencia. En la estructura de la información que maneja MIME, se pueden observar las acciones posibles a tomar para realizar la transferencia vía correo electrónico, de la información automáticamente.

A continuación se muestra un ejemplo donde se muestra una extracción de un anuncio de la publicación de un RFC.

Content-Type: Message/External-body;

Name="rfc1480.txt";

site="ds.internic.net"

access-type="anon-ftp";

directory="."

Content-type: text/plain

Un servidor de mail es capaz de traer el RFC automáticamente vía correo electrónico.

Pero por ejemplo, analicemos el siguiente mensaje falso que pudo ser enviado por un atacante:

Content-Type: Message/External-body;

Name=".rhosts";

site="ftp.visi.org"

access-type="anon-ftp";

directory="."

Content-type: text/plain

¿Estaríamos dispuestos a que en la ejecución automática de MIME se reescriba nuestro archivo *.rhosts*? Al administrar y configurar el correo electrónico se debe tener especial cuidado de éstos y otros posibles peligros que se corren al estar funcionando el programa MIME en forma automática.

4.1.5.2 CLIENTES LOCALES QUE HABLAN CON EL SERVIDOR CENTRAL

El servicio de correo es probablemente el más invaluable de los servicios en la Internet, siendo muy vulnerable al abuso. Como se implementa normalmente no provee mecanismos de autenticación. Deja la puerta abierta a mensajes falsos. El RFC 822 soporta una línea de encabezado encriptada, pero no es ampliamente usada.

A) El protocolo Post Office

El "Post Office Protocol" (POP) permite a usuarios remotos obtener correo almacenado en un servidor central. La autenticación se lleva a cabo por un simple comando conteniendo tanto el nombre del usuario como su contraseña. Combinando los dos en un solo comando obliga al uso de contraseñas convencionales. Y dichos contraseñas se hacen cada vez menos populares, ya que son vulnerables mediante cables intervenidos, divulgación intencional o accidental, etc.

B) PCMAIL

PCMAIL es más peligroso que el POP: soporta un comando de cambio de contraseña. Esta petición requiere que tanto la contraseña nueva como la anterior sean transmitidas sin encriptamiento.

4.1.5.3 ALGUNAS DEFENSAS EN EL CORREO

El RFC 822 soporta una línea de encabezado encriptada, pero no es ampliamente usada, sin embargo se han propuesto nuevos estándares de encripción para el correo electrónico, para ahondar más en el tema se sugiere ver el RFC 1040[xii].

4.1.5.4 TFTP -TRIVIAL FILE TRANSFER PROTOCOL

TFTP es un mecanismo de transferencia de archivos basado en UDP. No tiene autenticación implementada. Es comúnmente utilizado por estaciones de trabajo que no tienen disco y por terminales de X11.

[xii] Linn J. *Privacy Enhancement for Internet Electronic Mail: Part I. Message Encipherment and Authenticacion Procedures. RFC 1040.* 1988

Una configuración adecuada del demonio TFTP restringe la transferencia de archivos a uno o dos, generalmente /usr/local/boot y la librería de X11. Anteriormente los fabricantes diseñaban su software con accesos TFTP no restringidos, lo que dejaba la puerta abierta del sistema a los hackers.

A continuación se muestra un ejemplo de lo fácil que es bajo este esquema entrar a un sistema, y realizar operaciones ilícitas dentro del mismo.

$ tftp csg.sc.buca.info

tftp> get /etc/passwd /etc/tmp

Received 1205 bytesin 0.4 seconds

tftp> quit

$ crack < /tmp/passwd

Como se puede observar es demasiado simple realizar esta operación. Pudiendo obtener las contraseñas, la máquina y sus usuarios son vulnerables al intruso.

Algunos enrutadores utilizan TFTP para llamar sus imágenes ejecutables o archivos de configuración. Esto también implica un riesgo por que un hacker podría generar un archivo falso (aunque realmente esto es muy difícil), pero el riesgo se corre en sí, por que estos archivos de configuración contienen las contraseñas de los enrutadores.

Excepto por el caso mencionado anteriormente, y otros servicios muy similares, es decir, acceso a los hosts por enrutadores o máquinas que no posean disco, o que requieren de redundancia de archivos de configuración, no se justifica que se deje acceso libre TFTP. Y si se decide dejar el servicio TFTP en una máquina, debe asegurarse de que esté correctamente configurado, de forma que sólo accedan a los archivos especificados los clientes autorizados. Es posible implementar que no acepte rutas en las solicitudes sino únicamente peticiones de archivos localizados en el directorio actual.

4.1.5.5 FTP -FILE TRANSFER PROTOCOL

FTP [xiii]soporta transferencia de archivos y traducción de archivos tipo texto a archivos binarios. En una sesión normal de FTP, los comandos del usuario abren un canal de control hacia la máquina destino. Después de que el comando USER ha sido enviado, el código que despliega la máquina destino indica que el demonio que se está corriendo bajo FTP, ha iniciado un modo de trabajo especial. Durante toda la sesión FTP, se restringe el uso de comandos, a sólo aquellos necesarios para manipular archivos, de los cuales, no se incluyen todos los comúnmente disponibles en el sistema en una sesión normal.

[xiii]Postel, Jon and and Reynolds, Joyce. *File Transfer Protocol. RFC 959.* Octubre 1985

El dato actual, sea una transferencia de archivos o sea escuchado desde un directorio de comandos, son enviados por un canal de datos separado. Los servidores utilizan el puerto 20 para servicios FTP. Por omisión, el cliente utiliza el mismo número de puerto que el usado por el canal de control. El protocolo FTP sugiere que un sólo canal de datos sea abierto, y que se mantenga abierto durante toda la sesión de transferencia de archivos.

La mayoría de las implementaciones abren un canal para cada una de las transferencias de archivos, en una misma sesión al cliente le es asignado por su sistema un número de puerto aleatorio, el cual se le envía al servidor por medio del comando PORT. De regreso el servidor hace un llamada a este número de puerto.

El protocolo provee la facilidad de que el servidor elija un número de puerto y reciba la llamada sin tener que inicializarla. Esta característica fue pensada para transferencias por parte de terceros -un cliente FTP inteligente puede hablar con dos servidores simultáneamente: tener uno haciendo un requerimiento en forma pasiva, y otro hablando con esa máquina y puerto, en algunos servidores también se utiliza esta característica (comando PASV).

Aunque FTP Anónimo no es requerido por las especificaciones de FTP es parte de las costumbres en la Internet. Después del correo electrónico, este servicio es el más importante dentro de los servicios que provee la Internet.

Desde el punto de vista de seguridad, se deben tener consideraciones especiales en los servidores que proveen FTP anónimo, ya que el acceso al servidor está abierto a todo el mundo. La administración del servidor debe ser planeada de forma tal que el monitoreo, auditorías y mantenimiento al sistema, sean reforzadas.

Los servidores que proveen este servicio deben configurar el sistema de forma que el mundo exterior (miembros de Internet) tengan acceso a una parte de su sistema de archivos para extraer información. Generalmente los usuarios que utilizan este servicio se registran en la cuenta con la contraseña *anonymous*. En algunos otros servidores que proveen este servicio (sobre todo en los últimos años), el usuario debe introducir su dirección de correo electrónico como contraseña.

Una parte del problema es que algunas implementaciones requieren la creación de una réplica parcial del árbol de directorios; se debe tener mucho cuidado en que estos archivos no estén sujetos a compromiso, que no contengan información sensible, como contraseñas encriptadas.

El segundo problema es que el FTP anónimo es completamente anónimo, no hay un registro acerca de quién pidió qué información. Los servidores basados en correo (Mail-based) pueden proveer esos datos; ellos también proveen técnicas útiles para limitar las conexiones, las transferencias en segundo plano (background), etc.

La primera regla es que el directorio al cual tengan acceso los usuarios no tenga el permiso de escritura, y que el dueño del subdirectorio no sea la cuenta FTP, ya que FTP anónimo corre bajo ese *userid*.

La siguiente regla es no dejar un archivo de */etc/passwd* en el área de FTP anónimo. Si es necesario, se debe crear uno con basura (información no verídica de las contraseñas de los usuarios del sistema).

4.1.5.6 FSP -SEAKY FILE TRANSPORT PROTOCOL

FSP es otro protocolo de transporte de archivos. Utiliza un puerto UDP (generalmente el puerto 21) para implementar un servicio similar al que provee FTP. FSP no es un protocolo oficial y es muy poco utilizado, excepto por los hackers, quienes lo utilizan por su simplicidad.

4.1.5.7 TELNET

Telnet provee la conexión remota a hosts. El protocolo incluye funcionalidad para emular varios tipos de terminales. Como regla, el demonio Telnet llama al programa *login* para la autenticación del usuario para inicialzar la sesión remota.

Una sesión Telnet puede ocurrir entre dos hosts seguros. En este caso, un *Telnet seguro* puede ser usado para encriptar toda la sesión Telnet, protegiendo así, desde la contraseña hasta el contenido de la sesión.

La mayoría de las sesiones Telnet, son realizadas desde máquinas no confiables. Ni la llamada del programa, ni la llamada del sistema operativo, o la red por la cual se están transmitiendo los datos, podemos presumir son seguros: las contraseñas y la sesión de la terminal están a la vista de posibles atacantes.

Las contraseñas tradicionales no son confiables cuando algún enlace de la comunicación ha sido intervenido. Los hackers han realizado esto muy a menudo, y su punto de atención, principalmente son los *backbones*. Se recomienda utilizar esquemas de *contraseñas* más robustas que las comúnmente usadas por el sistema. Este es un tema que se trata por separado en este trabajo. Pero en cuanto a autenticación se refiere, una vez implementado el sistema robusto de contraseñas, nada nos asegura la protección del resto de la sesión Telnet. Por ejemplo los hackers que intervienen los enlaces, pueden obtener la información que está siendo transmitida dentro de la sesión Telnet. Si el comando Telnet ha sido accedido, se podrían insertar comandos dentro de la sesión, para poder reciclar la conexión después de que las máquinas cerraron la sesión. Lo mismo podría suceder si se ha intervenido físicamente el enlace, esto es mucho más difícil de realizar, pero no por eso se puede descartar la posibilidad.

La sesión Telnet completa, puede ser encriptada, pero esto no sería útil, si algunos de los puntos finales no es confiable, es más, puede ser peor que inútil: si alguno de los puntos finales es un impostor, obtiene las llaves de seguridad, y con esto el sistema está completamente comprometido.

4.1.5.8 NTP -NETWORK TIME PROTOCOL

El NTP[xiv],es un aditamento valioso de las máquinas *gateways*. Como su nombre lo indica, la función de NTP es sincronizar los relojes de las máquinas con el mundo exterior (en la red, o en la inter-red). Cada máquina habla con uno o más vecinos, y se sincroniza de acuerdo a una fuente autorizada de sincronización de reloj. Las comparaciones entre múltiples fuentes de información de tiempo, permiten a las máquinas descartar entradas erróneas, lo que provee al sistema de un alto grado de protección, en cuanto a deliberadas alteraciones.

El cronómetro utilizado por NTP es muy eficaz (con una exactitud de 10 ms o menor), y esto permite emparejar archivos tipo *log*, desde diferentes máquinas. Esta información es muy útil para entender la tecnología de los atacantes. Un uso de esta información generada por el NTP, es en protocolos de encriptamiento: ciertas vulnerabilidades de estos protocolos pueden ser reducidas si se puede confiar en la sincronización de los relojes.

Los archivos creados por NTP proveen pistas sobre las penetraciones no autorizadas al sistema. Los hackers alteran comandos del sistema, y las etiquetas de tiempo en sí, para borrar las evidencias de sus actividades. Sobre sistemas UNIX, algunas etiquetas de tiempo no pueden ser alteradas explícitamente - como el campo "*i-node changed*"- lo cual refleja al reloj del sistema como cuando se ha hecho el último cambio al archivo. Si se reinicializa este campo, los hackers pueden hacer cambios al reloj del sistema.

NTP por sí mismo puede ser objetivo de varios tipos de ataques[xv]. Un caso muy específico, es por ejemplo, los protocolos o dispositivos de autenticación basados en la sincronización de relojes. Si el reloj de una máquina es reinicializando, se puede repetir la cadena de autenticación.

Una opción para evitar agujeros de vulnerabilidad en protocolos de autenticación basados en sincronización de relojes se halla en las nuevas versiones de NTP en las que se ha implementado la encriptación de los mensajes de autenticación. Como la información de NTP se obtiene de hosts-a-hosts, es probable que se pueda interferir el paso de esta información como tal, y una forma de evitar esto, es configurar NTP de forma que no responda a requerimientos de fuentes no autorizadas, y que las fuentes de las cuales obtiene su información también sean fuentes confiables.

Existen algunos protocolos que no son inherentemente defectuosos, no obstante son susceptibles de abuso. Al implementar los servicios deben tomarse los siguientes problemas en consideración.

[xiv] Mills, David. Network Time Protocol (version 3) specification, implementation and analysis. RFC 1305. Marzo 1992.

[xv] Bishop, Matt. A Security analysis of the NTP protocol. In Sixth Annual Computer Security Conference Proceedings. Pags. 20-29. Tuscon AZ, Dec. 1990.

4.1.5.9 EL SERVICIO FINGER

Este servicio despliega información útil acerca de los usuarios, como sus nombres completos, su teléfono, la última vez que entró, etcétera. Desafortunadamente, esos datos les son de mucho provecho a los rompedores de contraseñas, pues además de proveer información que los usuarios emplean comúnmente para crear sus contraseñas, también les indican cuáles cuentas pueden ser los blancos ideales debido a su desuso.

4.1.5.10 REMOTE BOOTING (Inicialización Remota)

Actualmente se emplean dos conjuntos de protocolos para inicializar estaciones de trabajo y enrutadores. Reverse ARP (RARP) con TFTP (Trivial File Transfer Protocol) y BOOTP con TFTP. Si alguien puede trastornar el proceso de inicialización, puede ser sustituido un nuevo *kernel* con mecanismos de protección alterados. La inicialización basada en RARP es más riesgosa porque confía el Ethernet como red, con todas las vulnerabilidades que esto implica. Uno puede llevar a cabo una modesta implementación de seguridad asegurándose de que la máquina inicializándose use un número aleatorio para su puerto de origen UDP; de otra forma un atacante puede impersonar al servidor y enviar paquetes de datos falsos.

BOOTP añade una capa de seguridad incluyendo un identificador de transacción aleatorio de 4 bytes. Esto prevee que un atacante genere falsas contestaciones a la máquina que está inicializándose. Es vital que estos números sean aleatorios. Se debe tener especial cuidado cuando se inicialice a través de enrutadores, mientras más redes sean atravesadas más oportunidades de impersonificación.

La mayor medida de protección es que normalmente el atacante tiene una sola oportunidad; un sistema siendo inicializado no permanece en ese estado. Sin embargo, si las comunicaciones entre el cliente y el servidor estándar pueden ser interrumpidas, pueden ser armados ataques a gran escala.

4.1.5.11 DNS (DOMAIN NAME SYSTEM)

El DNS provee un servicio de nombres distribuido y jerárquico (mapeo de nombres de hosts a direcciones IP y viceversa)[xvi]. En operación normal los hosts envían consultas UDP a los servidores DNS. Los servidores contestan ya sea con la respuesta apropiada o con información acerca de los servidores dañados. Las consultas pueden ser vía TCP, pero la operación con TCP está por lo general reservada para *zonas de transferencia.* Dichas zonas son usadas por los servidores de respaldo para obtener una copia completa de su porción del espacio de nombres. También lo utilizan los intrusos para obtener rápidamente una lista de posibles máquinas a atacar.

El espacio de nombres del DNS está estructurado en forma de árbol. Por facilidad en la operación, subárboles pueden ser delegados a otros servidores. Se emplean dos

[xvi] Cheswick, William R. and Bellovin, Steven M. *FireWalls and Internet Security* Addison Wesley Publishing, USA, 1994. págs. 27 y 28

distintos árboles lógicos. El primero mapea nombres de hosts como CANCUN.CECAFI.UNAM.MX a direcciones como 132.70.54.1 , el segundo árbol es para consultas invertidas (mapeo de direcciones IP a nombres). No existe una relación forzosamente entre los dos árboles.

Esta desconexión puede causar problemas. Un cracker que controla una porción del árbol de mapeo inverso puede hacerlo mentir. Esto es, el registro inverso puede falsamente contener el nombre de una máquina que nuestra máquina considera confiable. El atacante intenta entonces un *rlogin* a nuestra máquina, que confiando al registro aceptará la llamada.

Muchos de los sistemas actuales son inmunes a este ataque. Después de obtener el nombre del host vía DNS, usan ese nombre para obtener su conjunto de direcciones IP. Si la dirección actual usada para la conexión no está en la lista, la llamada es rechazada y se señala una violación a la seguridad.

Existe una variante de este ataque más dañina. En esta versión el atacante contamina el caché de respuestas DNS antes de iniciar la llamada. Cuando la máquina atacada hace la verificación no encuentra problemas y el intruso obtiene acceso.

Aunque las últimas implementaciones del software de DNS son inmunes a esto, es imprudente asumir que no existen más agujeros. Es recomendado fuertemente no confiar en la autenticación basada en nombres.

El DNS, aún cuando esté funcionando correctamente, puede ser empleado para distintos tipos de espionaje. El modo de operación normal es hacer consultas específicas y recibir respuestas específicas. Sin embargo una petición de zona de transferencia puede ser usada para obtener una sección entera de la base de datos; aplicándola recursivamente se puede reproducir un mapa completo. Si un atacante conoce la vulnerabilidad de un sistema operativo puede consultar la base de datos para obtener todos sus objetivos de ataque. Otros usos para espionaje serían: conocer el número y tipo de máquinas en una organización en particular, por ejemplo, pueden revelar datos valiosos acerca del tamaño de la organización, y por lo mismo los recursos con que cuenta un proyecto en particular.

Afortunadamente, el DNS incluye un código de error para "rechazar" transferencias de zona. Este código debe ser empleado para peticiones de transferencias de zona de cualquier host no conocido por un server secundario legitimador. Desgraciadamente no existe un mecanismo de autenticación provisto por la petición de transferencia de zona; la autenticación de dirección origen es lo mejor que puede hacerse.

4.1.5.12 DEFENSAS

Desafortunadamente no existe una implementación suficientemente buena como defensa en el DNS. Como una alternativa se ha desarrollado recientemente un sistema de autenticación por el MIT llamado Kerberos (por su importancia más adelante se tratará

más a fondo). El servidor de nombres usa boletos de Kerberos para autenticar solicitudes y respuestas dentro de un mundo atendido por él.

Como una alternativa a la autenticación basada en direcciones, algunas implementaciones usan al "Authentication Server" (Servidor de Autenticación)[xvii]. Un servidor que desea conocer la identidad de su cliente debe contactar con el Servidor de Autenticación de dicho host cliente y pedirle información acerca del usuario dueño de una conexión en particular. Este método es inherentemente más seguro que una simple autenticación basada en direcciones, ya que usa una segunda conexión TCP que no está bajo el control del atacante. Así se pueden defender de ataques de número de secuencia y ataques de origen de ruteo. Sin embargo, existen ciertos riesgos.

El primero es que no todos los hosts son suficientemente competentes (seguros) para correr servidores de autenticación. Si el host cliente no es seguro, no importa quien diga el usuario que es, la respuesta no puede ser confiable. Segundo, el mensaje de autenticación por sí mismo puede ser comprometido por ataques a las tablas de ruteo. Finalmente, si el host destino está abajo, podría emplearse una variante del ataque de número de secuencia TCP, el atacante puede completar la secuencia de abertura y enviar una falsa respuesta.

Un servidor que desea confiar en un usuario de otro host debería usar un medio de validación más seguro, como el algoritmo Needham-Shroeder[xviii]. TCP por sí mismo es inadecuado.

4.1.6 Ataques en el enrutamiento

Los protocolos de enrutamiento son mecanismos para encontrar dinámicamente las rutas correctas a través de la red. Son fundamentales para la operación de TCP/IP. La información de enrutamiento establece dos rutas: de la máquina cliente a la destino y de regreso. (Usualmente la segunda ruta es la inversa de la primera). Desde el punto de vista de seguridad, la ruta de regreso normalmente es más importante. Si el enemigo sabe como corromper los mecanismos de enrutamiento puede ser engañada la máquina haciéndole creer que el atacante es una máquina confiable[xix].

El abuso de los mecanismos de enrutamiento y de protocolos es probablemente el ataque disponible más simple basado en protocolos. Existen diferentes formas de llevarlo a cabo, dependiendo de exactamente cuáles protocolos se empleen. Algunos de estos ataques tienen éxito sólo si el host remoto realiza autenticación basada en direcciones.

[xvii] St. Johns, *M. Authentication Server. RFC 931*, 1985

[xviii] Needham, R. M. y Schroeder, M.D. *Using Encriptyon for Authentication in Large Networks of Computers*, Communications of the ACM, vol. 21, no. 12, pp. 993-999, December 1978

[xix] Cheswick, William R. and Bellovin, *Steven M. FireWalls and Internet Security*. Addison Wesley Publishing, USA, 1994. pág. 26

Un buen número de ataques descritos a continuación pueden ser usados para negar el servicio por confusión de tablas de enrutamiento en un host o en un enrutador.

4.1.6.1 LOOSE SOURCE ROUTING

Si está disponible, el mecanismo más fácil de abusar es el IP source routing. Asumimos que el host destino emplea la ruta de origen provista en una solicitud TCP para responder. Dicho desempeño es completamente razonable; si el originador de la solicitud desea especificar una ruta en particular por alguna razón -digamos que la ruta esté muerta- las solicitudes podrían no alcanzar al originador si se siguiera una ruta diferente.

De acuerdo con el RFC 1122 la máquina destino debe usar el inverso de la ruta como ruta de vuelta, tenga o no sentido, lo que significa que un atacante puede impersonar cualquier máquina en que confíe la máquina atacada.

Source routing es raramente usado, aunque existen algunas razones legítimas para hacerlo, por ejemplo, puede ser utilizado para la opción debug en la red para monitorear los problemas de la misma.

La forma más fácil de defenderse de los ataques a los problemas de source routing es rechazar los paquetes que tengan esa opción, muchos enrutadores ofrecen esa opción.Esto es menos práctico de lo que parece si notamos que algunos adaptadores de red Ethernet reciben sus propias transmisiones, y esta característica depende de algunos protocolos de alto nivel. Además, esta solución falla totalmente si una organización tiene dos redes confiables conectadas mediante un backbone multi-organización. Otros usuarios en el backbone pueden no ser confiables de la misma forma en que los usuarios locales presumen ser, o tal vez su vulnerabilidad a un ataque de fuera es mayor. Decididamente dichas topologías deberían ser evitadas en cualquier caso.

Un método simple podría ser rehusar conexiones preautorizadas si la información del enrutamiento de origen estaba presente. Esto presume que hay algunas razones legítimas para emplear esta opción IP, especialmente para operaciones relativamente normales. Una variación en defensa sería analizar la ruta de origen y aceptarla sólo si fueron listados enrutadores confiables; de esta forma aseguramos que el enrutador entregue el paquete al verdadero host destino.

4.1.6.2 ATAQUES AL PROTOCOLO DE INFORMACIÓN DE ENRUTAMIENTO

El RIP (Routing Information Protocol)[11] se emplea para propagar información de enrutamiento en redes locales. Típicamente la información recibida no es verificada y generalmente los hosts y enrutadores la creerán. Esto le permite a un intruso enviar información de enrutamiento falsa a un host destino, y a cada uno de los enrutadores en el

[11]Hedrick, C. *Routing Information Protocol RFC 1058*, 1988.

camino, personificando un host en particular. El ataque más probable de este tipo sería reclamar una ruta a un host particular no usado o cercano, esto podría causar que todos los paquetes destinados a ese host se le enviaran a la máquina intrusa. Ellos son reenviados usando la dirección origen IP al destino real. En este proceso el atacante podría capturar contraseñas y otro tipo de información sensible.

Algunos protocolos de enrutamiento tales como RIP versión 2 [xxi] y OSPF[xxii], proveen un campo de autenticación. Estos campos son de utilidad limitada por tres razones, primero, los únicos mecanismos de autenticación definidos actualmente son simples contraseñas, cualquiera que tenga la habilidad para jugar con los protocolos de enrutamiento, también es capaz de coleccionar las contraseñas que están viajando en el segmento local. Segundo, si el nodo fuente legítimo en el diálogo de enrutamiento ha sido plagiado, entonces sus mensajes (correcta y legítimamente firmados en apariencia) tampoco pueden ser confiables. Finalmente, en la mayoría de los protocolos de enrutamiento, cada máquina habla únicamente con sus vecinos y éstos a su vez, repiten lo que les dijeron (RIP, por ejemplo), de esta manera la información alterada se difunde.

No todos los protocolos de enrutamiento tienen estas debilidades. Aquellos enrutadores que involucran diálogos entre pares de hosts, son más difíciles de plagiar, aunque através del ataque a los números de secuencia, similarmente a los descritos anteriormente, todavía pueden ocurrir.

Una solución podríamos llamarla "enrutador paranoico", esto es que filtre los paquetes basándose en la dirección origen o destino, bloqueando cualquier forma que insinúe burlar un host.

Otra defensa es que el RIP sea más escéptico acerca de las rutas que acepta. En la mayoría de los ambientes no hay una buena razón para aceptar nuevas rutas para las redes locales. Un enrutador que hace esta verificación puede fácilmente detectar intentos de intrusos.

Sería recomendable también autentificar los paquetes RIP.

Una defensa más robusta, es la topológica. Los enrutadores pueden y deberían ser configurados de forma que conozcan las rutas válidas sobre un cable. En general esto puede ser una situación difícil, por lo que los *firewalls* en los enrutadores plantean un método de implementación relativamente simple.

[xxi] Malkin, Gary. *RIP version 2 -carrying additional information. RFC 1388.* January 1993.

[xxii] Moy, John. *OSPF version 2. RFC 1247.* Jul 1991.

4.1.7 PROTOCOLOS BASADOS EN RPC -REMOTE PROCEDURE CALL

4.1.7.1 RPC Y PORTMAPPER

El protocolo RPC[xxiii], es la base de muchos de los servicios de red que existen actualmente. Desafortunadamente muchos de esos servicios son problemas potenciales de seguridad, por lo que entender como funciona el protocolo RPC, es de vital importancia.

El concepto básico de RPC es simple: la persona que crea un servicio de red utiliza un lenguaje especial para especificar los nombres de las entradas externas y sus parámetros. Un precompilador convierte estas especificaciones en un parte de código que se adiciona en forma de subrutina a los módulos cliente-servidor. Con la ayuda de esta adición al código de servicios de red propietarios del sistema, el modulo del cliente puede hacer parecer la llamada a una de estas subrutinas de la aplicación (servicio de red personalizado) como una llamada del sistema propietario a un servidor remoto. La mayoría de las dificultades que podrían presentarse en la programación de los servicios de red son enmascaradas por RPC.

RPC puede existir sobre los protocolos UDP o TCP. Un subsistema que utiliza RPC sobre los servicios de UDP debe preocuparse acerca de la pérdida de paquetes, duplicación y perdida de orden de los mismos. La frontera de los registros son insertados en los servicios de RPC montados sobre TCP.

Los mensajes RPC inician con un encabezado. En este encabezado se incluye un *número de programa*, el *número de procedimiento* que indica el punto de entrada dentro de un procedimiento, y algunos otros números utilizados para el control de las subrutinas de la aplicación. Si se desean filtrar paquetes RPC se deben considerar estos campos. El encabezado incluye también un número de secuencia para relacionar las solicitudes con las respuestas a estas solicitudes.

Dentro de los mensajes RPC esta contenida un área de autenticación. Este campo tiene algunas variaciones, como por ejemplo, la "autenticación nula", la cual es utilizada para servicios anónimos. Para servicios más formales, se incluye un campo denominado "autenticación UNIX". Este campo incluye un número de *userid* (identificación de usuario) y un número de *groupid* (identificación del grupo al que pertenece el usuario), y el nombre de la máquina que realiza la llamada. Se debe tener especial cuidado con la información que contiene este campo. El nombre de la máquina nunca debe ser una máquina confiable para el sistema (los servicios importantes como NFS lo ignoran en favor de las direcciones IP), y tampoco el *userid* y el *groupid* tienen algún valor a menos que el mensaje venga desde un puerto privilegiado. En realidad, aún cuando estos campos tengan un pequeño valor para las aplicaciones que utilizan RPC basadas en el protocolo

[xxiii] Sun Microsystems. *Network Interfaces Programmer's Guide*. Mountain View, CA. Marzo 1990. SunOS 4.1

UDP, forzar una dirección fuente es relativamente fácil, por lo que no es de ayuda en cuanto a prevención de intromisiones no deseadas al sistema.

RPC soporta autenticación cifrada. Utiliza DES (Data Encryption Standar)[xxiv], denominado también *RPC seguro*. Todas las llamadas son autentificadas por medio de una *llave de sesión* compartida. La distribución de las llaves de sesión, es llevada a cabo por el algoritmo *Exponential Key Exchange* (también conocido como algoritmo Diffie and Hellman). La versión de Sun no es lo suficientemente robusta como para resistir un ataque muy sofisticado.

Desafortunadamente la mayoría de los sistemas no incorporan bien el protocolo RPC seguro. El único protocolo estándar que utiliza RPC seguro, es NFS, algunas versiones de Telnet y FTP, así como algunas implementaciones de X11, lo utilizan. Además, la distribución de las llaves de sesión es muy delicado, y no escalan su uso fuera de las redes locales.

OSF's Distributed Computing Environment (DCE)[xxv] usa el concepto de RPC Seguro, pero con *Kerberos* como mecanismo de distribución de llaves. DCE también provee en forma paralela control por medio de listas de acceso para la autenticación.

Se debe recordar que los mensajes RPC contienen parámetros resultado de una llamada a un procedimiento o para la llamada del mismo. Estos parámetros (incluyendo el encabezado) son codificados utilizando el protocolo XDR -eXternal Data Representation. A diferencia del estándar ASN.1 (11), XDR no utiliza etiquetas explícitas, de esta forma es imposible decodificar -y por lo tanto filtrar- un mensaje RPC sin un reconocimiento (*knowledge*) de la aplicación.

Con la notable excepción de NFS, los servidores basados en RPC no utilizan números de puerto fijos. Estos aceptan cualquier número de puerto que les es asignado por el sistema operativo de la máquina, y registran esta asignación por medio del *portmapper* (los servidores que necesitan puertos privilegiados eligen y registran un número bajo de puerto -menor que 1024 - no asignado todavía por el sistema). El portmapper -el cual utiliza el protocolo RPC para comunicarse -actúa como un intermediario entre RPC y los servidores. Al contactar con el servidor, el cliente pregunta el portmapper sobre el servidor del host para el número de puerto y protocolo (UDP o TCP) del servicio. Esta información la utiliza también la llamada actual de RPC.

El portmapper tiene otras habilidades menos benignas desde el punto de vista de seguridad del sistema. Si hay una llamada a un servicio no registrado, bien piensa para denegar el servicio a un atacante por que no está correctamente autentificado. El portmapper describe los procesos que están corriendo en un servidor, lo cual desde el punto de vista del atacante es muy útil. (Se han capturado archivos *log* de actividades

[xxiv]"Diffie, Whitfield and Hellman, Martin E. *Exhaustive cryptanalysis of the NBS data encryption standard.* Computer June 1977.

[xxv]"Rosenberry, Ward et al. *Understanding DCE.* O'Reilly Associates, Sebastopol, CA, 1992.

que realizan los hackers en un sistema, donde se muestra información, obtenida por medio del comando *rpcinfo*).

El problema de seguridad más serio que se presenta por la utilización del portmapper, es la habilidad de poder usar llamadas indirectas. Al evitar la sobrecarga (tráfico innecesario en la red) extra de ida y vuelta necesario para determinar el número de puerto real, un cliente pregunta que portmapper le envía la llamada RPC al servidor actual. Pero el mensaje reenviado, por necesidad, trae incluida la dirección del propio portmapper de regreso. De esta forma es prácticamente imposible para la aplicación distinguir entre un requerimiento local genuino, y de esta forma se calcula el nivel de seguridad que debe ser concedido a la llamada.

Algunas versiones de portmapper realizan su propio filtro de información. Aún, con esta medida, el hecho de que se logre bloquear el acceso al portmapper, no nos asegura que no existan intromisiones a los servicios por sí mismos.

Si se evitaran los problemas surgidos por el uso del portmapper, aún los servicios de RPC por sí mismos, tienen antecedentes de agujeros en la seguridad, la mayoría de los cuales se han dado localmente dentro de la conectividad de Ethernet. Windows utiliza los servidores de RPC para las operaciones de cortar y pegar texto, y para pasar referencias de archivos entre aplicaciones. Para algunos extraños al sistema, le es relativamente fácil abusar de este servicio para obtener copias de archivos del sistema.

4.1.7.2 NIS -NETWORK INFORMATION SERVICE

Uno de los servicios más peligrosos que utilizan el protocolo RPC es NIS, formalmente conocido como YP. (Inicialmente el servicio fue llamado *Yellow Pages*, pero el nombre infringió la marca registrada de un servicio de teléfonos en El Reino Unido). NIS es utilizado para distribuir bases de datos desde un servidor de NIS a sus clientes. Estas bases de datos incluyen los archivos de contraseñas, la tabla de direcciones de hosts, las llaves públicas y privadas de las bases de datos usadas por RPC Seguro. Los accesos pueden ser por búsqueda de llaves o todo el archivo puede ser transferido.

Al utilizar NIS, los riesgos que se corren son obvios. Si un intruso obtiene el archivo de contraseñas del sistema, tiene en realidad información valiosa. La llave de la base de datos puede ser casi tan bueno; las llaves privadas de las bases de datos para usuarios individuales son encriptadas generalmente con sus contraseñas.

Los hosts que tienen esquemas de seguridad utilizan un archivo de contraseñas denominado *shadow*. Si alguien obtiene los archivos vía NIS, no le es posible ver el contenido de /etc/password, si el sistema tiene implementado *shadow*. Tales sistemas utilizan mecanismos para aplicaciones que usan contraseñas cuando validan. Esto es realizado por medio de servicios basados en RPC, que pueden ser generados por los atacantes.

Los clientes de NIS necesitan conocer servidores de respaldo de NIS, en caso de que el servidor NIS fallara. En algunas versiones, a los clientes les es indicado remotamente usar un diferente, y posiblemente, fraudulento servidor de NIS. Este servidor puede insertar falsas entradas de archivos de contraseñas, direcciones de hosts incorrectas, etc.

En algunas versiones de NIS es posible deshabilitar la mayoría de estos servicios que implican un riesgo desde el punto de vista de seguridad. Y si es un host es expuesto a una red externa, se recomienda hacerlo.

4.1.8 World Wide Web (Servicios de Información)[xxvi]

Recordemos que en estas aplicaciones un host contacta un servidor, envía una pregunta o un apuntador a información y recibe una respuesta. La respuesta puede ser un archivo a desplegarse o un apuntador o conjunto de apuntadores hacia algún otro servidor. Las preguntas, los documentos y los apuntadores son todos fuentes potenciales de ataque.

El servidor está en peligro también, si acepta apuntadores. Estos apuntadores frecuentemente tienen nombres de archivos ligados a ellos. Mientras los servidores intentan verificar que los archivos solicitados son autorizados para transferir, el proceso de verificación puede ser (y en efecto ha sido) infestado. Las fallas aquí pueden permitirle a los atacantes obtener cualquier archivo del servidor.

En cuanto al servidor, el mayor de los peligros resulta cuando éste comparte el árbol de directorios con el FTP anónimo. En ese caso un atacante puede depositar primeramente archivos de control y solicitar al servidor de información que los interprete.

Los programas clientes interpretan los datos que se bajan de servidores arbitrarios en la Internet. Si no existen verificaciones del contenido de los datos importados, existe el peligro potencial de que estos datos alteren programas corriendo en los sistemas del cliente. Estos "caballos de Troya" pueden tomar diferentes formas, desde URLs maliciosos hasta códigos engañosos que corren a través de intérpretes (como Postscript) en el sistema cliente).

Otro problema de seguridad ocurre en algunos de los lugares que se han protegido mediante firewalls. Estos lugares han implementado políticas de seguridad en sus firewalls, que describen lo servicios de red que desean permitir dentro de su organización. Una característica de los URLs es que soportan diferentes tipos de recursos: FTP, HTTP, Gopher, WAIS, NNTP, Mailto, Prospero, TELNET, y RLOGIN, mientras que el firewall sólo permite un subconjunto de ellos. Por consiguiente, ya que los servidores de Web proveen estos servicios de manera independiente a los mecanismos normales, es posible traspasar las políticas de seguridad de los firewalls usando los servicios de Web.

[xxvi] Cheswick, William R. and Bellovin, Steven M. *FireWalls and Internet Security*. Addison Wesley Publishing, USA, 1994. págs. 44-48

Las amenazas a los servidores consisten principalmente de modificaciones no autorizadas a los datos del servidor, o bien comprometiendo al sistema del servidor explotando errores en el software del Web.

El protocolo HTTP provee diferentes métodos para escribir datos en un servidor. Es posible que alguno de estos métodos sean usados para modificaciones sin autorización a los datos.

4.1.8.1 DEFENSAS

Actualmente se encuentran en desarrollo algunas defensas para los ataques mencionados, algunas de las opciones con las que ya contamos son:

Los agentes pueden usarse en ambientes de firewalls para habilitar hosts en las redes protegidas para usar el Web. Un agente desarrollado por el CERN corre en un firewall y transmite las solicitudes entre ambos lados de la pared. Dado este diseño en el que el agente actúa como un intermediario, resulta más natural el diseño de la seguridad dentro del agente.

Actualmente distintas organizaciones están trabajando en el desarrollo de agentes. TIS está desarrollando un agente HTTP público que controlen muchos de los problemas que hacen que correr un Web resulte peligroso para muchas organizaciones.

EIT (Enterprise Integration Technologies, Inc) está diseñando el HTTP seguro, que consiste de un conjunto de cambios a los protocolos para manejar confidencialidad, integridad y autenticación. A TIS, RSA y NCSA se les ha solicitado revisar el protocolo.

Tanto EIT como RSA Data Security, Inc. han formado una unión llamada Terisa Systems que planean desarrollar los productos Mosaic seguro y HTTP seguro. El CERN está trabajando en "Shen", un conjunto de extensiones seguras para el protocolo HTTP que proveen autenticación y encripción. Shen es un protocolo alternativo para el HTTP seguro.

Algunos de los autores de Mosaic del National Center for Supercomputing Applications (NCSA) se han unido en una compañía llamada Mosaic Communications Corporation. Tienen planes de seguridad, pero todavía no existe más información disponible.

TIS está trabajando duro en está área, tratando de desarrollar estándares comunes en coordinación con otras organizaciones. Los desarrollos más recientes pueden encontrarse verificando la página de TIS en http://www.tis.com.

4.1.9 El sistema X11

X11[xxvii] usa la red para comunicación entre aplicaciones y los dispositivos de entrada/salida (la pantalla, el mouse, etc.), que permite a las aplicaciones residir en diferentes máquinas. Este es el origen del gran poder de X11, pero es también el de un gran peligro.

Como se mencionó anteriormente, el concepto fundamental de X11 es la noción de que la terminal del usuario es un servidor.

Las aplicaciones conectadas a un servidor X11 pueden hacer toda clase de cosas. Pueden detectar presiones de teclas, leer el contenido de la pantalla, generar conjuntos de teclas para aplicaciones que lo permitan, etc.

Es posible para una aplicación acaparar el control del teclado, cuando quiere hacer cosas como leer un contraseña. Pocos usuarios utilizan esta capacidad. Aún si lo hicieran existen otros mecanismos que permiten evitar que se apropien del teclado.

El problema es claro, un atacante en cualquier parte de la Internet puede sondear por servidores X11. Si están desprotegidos, como es el caso regular, la conexión tendrá éxito, generalmente sin notificación al usuario ya que el puerto no es difícil de adivinar (6000), que no es privilegiado en muchos sistemas.

4.1.10 Algunos Mecanismos de Ataques

4.1.10.1 VULNERABILIDAD DE LA RED LOCAL

Algunas redes locales, especialmente las redes Ethernet, son extremamente vulnerables al "eavesdropping"* y a burlar hosts. Si se emplean dichas redes, el acceso físico debe ser controlado.

Si la red local usa el Protocolo de Resolución de Direcciones (ARP) son posibles más formas sutiles del engaño de hosts. En particular se vuelve trivial interceptar, modificar y seguir los paquetes, únicamente espiando el tráfico o haciéndose pasar por un host.

Es posible realizar ataques de niegue de servicio disparando tormentas de *broadcasts*. Existe una variedad para realizar esto; es muy fácil si la mayoría o todos los hosts están actuando en la red como enrutadores. El atacante puede "broadcast" un paquete destinado a una dirección IP no existente. Cada host después de recibirlo intentará reenviarlo a su destino apropiado. Esto por sí mismo significa una cantidad significativa de tráfico, como si cada host generara un broadcast ARP para el destino.

[xxvii] Sheifler Robert and Gettys James. *X Window System*. Digital press, Burlington, Massachusets, 3a. ed.,1992

* Escuchar ilícitamente de manera oculta el tráfico de la red.

4.1.10.2 PUERTOS RESERVADOS

Los TCPs y UDPs derivados de Berkeley tienen la noción de "puertos privilegiados". Esto es, números de puertos menores que 1024 únicamente pueden ser asignados a procesos privilegiados. Esta restricción es usada como parte del mecanismo de autenticación. Sin embargo, ni las especificaciones de TCP ni las de UDP contienen dicho concepto. Los administradores no deben confiar nunca en los esquemas de autenticación de UNIX cuando se comuniquen con dichas máquinas.

4.1.11 Resumen

Finalmente como recopilación de la información anteriormente descrita, podemos hacer un resumen de los más comúnes ataques al protocolo TCP/IP.

1. Las fallas más comunes de seguridad que se presentan en un sistema son los generados por contraseñas débiles, la ausencia de las mismos, o de las políticas de asignación de éstas.
2. Ataques a la asignación de números de secuencia para burlar la autenticación basada direcciones.
3. Es muy fácil burlar a los paquetes UDP.
4. Los paquetes ICMP pueden romper todas las conexiones entre dos pares de hosts. Por medio de mensajes Redirect -pueden alterar las tablas de enrutamiento para usos no legítimos.
5. Por medio de la opción IP Source Routing se puede alterar la autenticación por medio de direcciones IP.
6. Es relativamente fácil insertar mensajes falsos de RIP.
7. Los árboles inversos de DNS pueden ser utilizados para alterar la información de la base de datos de nombres.
8. El caché del servidor DNS puede ser contaminado para frustrar la verificación.
9. Sendmail es un comando de alto riesgo de seguridad (configuración complicada).
10. Es relativamente fácil bloquear vía enlace una sesión Telnet.
11. Alterar el cronómetro de NTP para atacar los protocolos de autenticación.
12. El comando *finger* despliega información valiosa de los usuarios y sus sesiones.
13. Los campos de nombres de máquinas de RPC no son confiables.
14. Se pueden obtener archivos de contraseñas desde NIS.
15. Es posible direccionar máquinas a servidores de líneas telefónicas vía NIS.

16.Es dificil revocar accesos no legítimos vía NFS.

17.Si no es correctamente configurado TFTP, se podrían manipular los archivos /etc/password .

18.Si el subdirectorio de trabajo FTP tiene permisos de escritura, se puede hacer mal uso de la información.

19.Si los archivos de contraseñas reales de los usuarios de la máquina están en un lugar alcanzable para los accesos FTP anónimo se pueden obtener vía este tipo de sesión.

20.El formato de la información WWW puede comprometer la seguridad del sistema

21.Los apuntadores de archivos de los servidores WWW pueden ser punto vulnerable de la seguridad de los mismos.

22.Los scripts de consulta mal escritos de WWW son un riesgo de seguridad.

23.Un posible atacante dentro de Internet puede sondear el acceso a la máquina por medio de los servidores X11.

24.Los números de puerto asignados fuera del sistema de la máquina no son confiables.

25.Es casi imposible poder aplicar filtros de seguridad correctamente a todos los paquetes UDP.

26.Los firewalls no pueden bloquear protocolos de alto nivel.

27.Las herramientas de monitoreo pueden ser peligrosas si se están corriendo en máquinas expuestas al resto del mundo (Internet).

28.Existen muchas formas de obtener el archivo de contraseñas.

29.El administrador de la máquina puede ser responsable de las actividades del hacker.

4.2. Problemas de Seguridad UNIX y defensas

A continuación mencionamos los problemas de seguridad en UNIX por ser éste el principal y más difundido sistema operativo en las redes que utilizan TCP/IP, y por consecuencia de gran parte de la RedUNAM.

Los problemas que se pueden presentar en esta área son muy variados y de diversa índole. Tratar de explicarlos todos está más allá del alcance de esta tesis. Por lo anterior se describirán los ataques a la seguridad más importantes que no se traslapen con otros ya mencionados, aunque dependan también del sistema operativo, por medio de una visión general.

Los problemas de seguridad de Unix surgen al mismo tiempo que el nacimiento del propio sistema, ya que no fue diseñado con medidas intrínsecas de seguridad pues su propósito era el de compartir información entre usuarios trabajando en proyectos comunes. Los problemas surgen después cuando el sistema funciona en otros ambientes menos abiertos, donde la privacidad es importante.

4.2.1 Seguridad en las cuentas

La manera más fácil y mayormente empleada por los intrusos de entrar a un sistema es entrar con la cuenta de alguien más.

4.2.1.1 CONTRASEÑAS

Los contraseñas son la parte vital de UNIX (y de casi cualquier sistema). Si un atacante descubre la contraseña de un usuario, obtendrá acceso al sistema y con ello todos los privilegios de dicho usuario, no es imposible imaginar lo que sucedería si este usuario fuera un super-usuario. Por estas razones resulta extremamente importante la elección de una buena contraseña.

El programa *passwd* de UNIX pone muy pocas restricciones a lo que debería usarse como contraseña, cuando más pide seis caracteres y menos (cuatro) si se emplean mayúsculas o caracteres no alfabéticos. Peor aún, si el usuario insiste en usar una contraseña más corta (tres ocasiones) el programa lo permitirá. No hace verificaciones de contraseñas obviamente inseguras. Entonces, es obligación del administrador asegurarse que lo sean.

En la referencia[xxviii] los autores describen experimentos que los conducen a conocer los hábitos de elección de contraseñas de los usuarios, 86% de ellas se podían considerar inseguras.

En otros experimentos[xxix] se demuestra que intentando con tres suposiciones en cada cuenta -la clave de entrada (login), la clave al revés, los dos concatenados- un atacante podría obtener acceso de entre un 8 a un 30 por ciento de las cuentas de un sistema típico. Un segundo experimento mostró que intentando con los 20 nombres femeninos más comunes, seguidos de un dígito (un total de 200 contraseñas), al menos uno era válido en una de cada doce máquinas. Siguientes experimentaciones por el autor encontraron que intentando con variaciones de la clave de entrada, el nombre, apellidos y una lista de cerca de 1800 nombres comunes, hasta un 50% de las contraseñas en un sistema dado podían ser descubiertos de uno a dos días.

[xxviii] Morris, Robert and Ken Thompson. *Password Security: A Case History*.Communications of the ACM, 594-597. Nov. 1979

[xxix] Grammp, F.T. and R. H. Morris. *UNIX Operating System Security*. AT&T Bell Laboratories Technical Journal. 1649-1672, Oct. 1984

Además del problema de selección de contraseñas existen otros más inherentes al manejo que los usuarios hacen de los mismos.

Es muy usual que los usuarios escriban sus contraseñas en teclados, hojas, en su escritorio, la envíen por mail o la guarden en un archivo, y en muchos de los casos se la den a otras personas, y si de por sí es difícil que el usuario cuide su contraseña los demás con menor razón lo harán.

4.2.1.2 FECHAS DE EXPIRACIÓN

Muchas organizaciones tienen cuentas de usuarios que ya no se encuentran en la organización o que muy raramente las utilizan. Estas cuentas son un gran agujero pues nadie notaría que se ha irrumpido en ellas.

4.2.1.3 CUENTAS "guest"

Las cuentas tipo *guest* (huésped) presentan otro agujero de seguridad. Por su naturaleza estas cuentas son raramente usadas, y lo son por personas que deberían de tener acceso únicamente por el corto período en el que sean huéspedes (por ejemplo el tiempo en que se les está dando curso).

4.2.1.4 CUENTAS SIN CONTRASEÑA

Algunos sitios tienen cuentas instaladas con nombres como "who", "date", "lpq", etcétera, que ejecutan comandos simples. Estas cuentas existen con el propósito de que los usuarios pueden ejecutar los comandos sin tener que entrar a la máquina. Típicamente estas cuentas no tienen una contraseña asociada y cualquiera puede usarlas. Muchas de estas cuentas tienen un *user id* (identificación de usuario) de cero, por lo que ellas se ejecutan con permisos de super-usuario.

El problema con estas cuentas es que abren hoyos de seguridad potenciales. Por no tener contraseñas asociadas y tener permisos de super-usuario son una invitación a que un intruso intente penetrar mediante ellas.

4.2.1.5 CUENTAS DE GRUPOS

Las cuentas para grupos se hicieron muy populares en muchos lugares, pero actualmente no ocurre tan frecuentemente. Una cuenta de grupo es una simple cuenta que es compartida por varias personas, por ejemplo los colaboradores de un proyecto. Pero como se mencionó en la sección de contraseñas una contraseña no debe ser compartida entre usuarios, el concepto de cuenta de grupo viola directamente esta política.

4.2.2 Defensas en los Contraseñas

4.2.2.1 SELECCIÓN DE LA CONTRASEÑA

Nuestro primer intento como defensa contra los ataques por contraseña es establecer cuál es una buena elección de contraseña, a continuación se muestran una serie de guías para la correcta elección.

- No usar la clave de entrada (login name) en ninguna forma (al revés, en mayúsculas, doblemente, etc.)
- No usar ni el nombre ni los apellidos en ninguna forma.
- No usar el nombre de la esposa, hijo(s) o amigos cercanos en ninguna forma.
- No usar información fácilmente obtenible acerca de sí mismo. Esto incluye el número de placa, telefónico, RFC, el nombre de la calle en que se vive, etc.
- No usar una contraseña donde sean sólo dígitos, o todos la misma letra. Esto decrece significativamente el tiempo de búsqueda de un cracker.
- No usar una palabra que se encuentre en el diccionario, o cualquier otra lista de palabras.
- No usar una contraseña menor a 6 caracteres.
- Usar una contraseña que mezcle mayúsculas con minúsculas.
- Usar una contraseña con caracteres no alfabéticos, como el punto y coma, dos puntos, dígitos, etcétera.
- Usar una contraseña fácil de recordar, de tal manera que no se tenga que escribir en alguna parte.
- Usar una contraseña que se pueda teclear rápidamente sin tener que estar viendo el teclado. Esto hace más difícil a aquél que desee robar la contraseña mirando por encima de los hombros.

Aunque esta lista puede verse como muy estricta, existen algunos métodos de elegir un contraseña seguro y fácil de recordar. Algunos de estos métodos incluyen lo siguiente:

- Elegir una línea o dos de una canción o poema, y usar la primera letra de cada palabra. Por ejemplo "¿Qué es esto? ¡Prodigio! Mis manos florecen." que se convierte en "QeePMmf.".
- Alternar entre una consonante y una o dos vocales, hasta ocho caracteres. Eso provee de ningún sentido a las palabras que usualmente son pronunciables, y esto es fácil de recordar. Por ejemplo "routboo,", "quadpop", y por el estilo.

- Elegir dos palabras cortas y concatenarlas con un signo de puntuación entre ellas. Por ejemplo "mar;ola", "sal:sol","gato?rata".

Las reglas mencionadas son una barrera eficiente contra las estrategias de los rompedores de contraseñas.

4.2.2.2 POLÍTICAS DE CONTRASEÑAS.

Aunque pedirle a los usuarios que elijan contraseñas seguras ayuda a mejorar la seguridad, por sí mismo no es suficiente. Es importante formar un conjunto de políticas de selección de contraseñas que todos los usuarios deban obedecer, con el objetivo de mantener las contraseñas seguras.

- Lo primero y más importante es enfatizar en los usuarios la necesidad de mantener sus contraseñas únicamente en sus mentes.
- La segunda política consiste en que los usuarios nunca deben dar sus contraseñas a otros.
- Finalmente, es importante establecer la política de que los usuarios deban cambiar sus contraseñas regularmente, digamos que por lo menos tres veces al año. Esto es difícil de reforzar en UNIX, pero ya existe software de terceros que puede implementar esta política.

El conjunto de políticas debe ser impreso y distribuido a todos los usuarios actuales del sistema. También deben ser dadas a todos los nuevos usuarios cuando reciban sus cuentas.

4.2.2.3 VERIFICAR LA SEGURIDAD DE LAS CONTRASEÑAS.

Las políticas y procedimientos mencionados, cuando se implementan correctamente reducen en gran medida las oportunidades de un atacante de irrumpir en el sistema vía robo de una cuenta. Sin embargo, aún con todas las medidas, el administrador del sistema debe verificar periódicamente que las políticas y los procedimientos se estén llevando a cabo.

La mejor forma de verificar la seguridad de las contraseñas en el sistema es emplear un "rompedor de contraseñas" (password-cracking) muy cercano al que un atacante emplearía. Si tiene éxito al adivinar algunas de las contraseñas, esas contraseñas deben ser cambiadas inmediatamente. Existen algunos de estos programas disponibles sin costo alguno en varios hosts de la Internet. Alternativamente, es posible escribir nuestro propio programa y acondicionarlo a nuestro propio sistema.

Uno de estos programas es el NPASSWD.

4.2.2.4 FECHAS DE EXPIRACIÓN

La manera más simple de prevenir cuentas no utilizadas es poner una fecha de expiración en cada cuenta. Un *shell script* sencillo puede usarse periódicamente para verificar que todas las cuentas tengan fechas de expiración, y que ninguna haya pasado. En la fecha de expiración se puede verificar si el usuario ha usado recientemente su cuenta y si no, puede ser deshabilitado (poniendo un asterisco en la parte correspondiente a la contraseña en el archivo de contraseñas) o incluso borrado del sistema.

4.2.2.5 CUENTAS GUEST

La manera más segura de manejar cuentas guest es crearlas sólo cuando es estrictamente necesario y borrarlas tan pronto como la gente para la que hayan sido dispuestas dejen de usarlas, o deban dejar de usarlas.

4.2.2.6 CUENTAS SIN CONTRASEÑAS

Simplemente, las cuentas sin contraseñas no deben ser permitidas en ningún sistema UNIX.

4.2.2.7 YELLOW PAGES (NIS)

Las páginas amarillas son un servicio muy útil como para sugerir suprimirlo, aunque esto significará más seguridad para el sistema. En vez de esto, se recomienda leer cuidadosamente la información de los manuales de Sun con el objeto de estar totalmente conscientes de la habilidades y limitaciones de este servicio.

4.2.3 Seguridad en la red

Muchos lugares se conectan a numerosas redes alrededor del mundo formando la Internet. Esto significa que los usuarios de nuestras máquinas pueden accesar hosts y comunicarse con otros usuarios en el mundo. Desafortunadamente esto también implica que otros hosts y otros usuarios pueden accesar nuestra máquina e intentar irrumpir en ella.

4.2.3.1 HOSTS CONFIABLES

Una de las implementaciones más convenientes del software de UNIX de red es el concepto de hosts confiables. El software permite la especificación de otros hosts (y posiblemene usuarios) que se consideran confiables, y les serán permitidos *"logins"* (accesos) remotos y ejecuciones de comandos remotos sin requerir que el usuario introduzca su contraseña. Esto es muy conveniente porque los usuarios no tienen que teclear su contraseña cada vez que usan la red. Desafortunadamente, por la misma razón, el concepto de host confiable es extremamente inseguro.

4.2.3.2 CORREO (MAIL)

Así como con el software de FTP, algunas versiones antiguas del *sendmail* tienen varios errores que permiten la violación de la seguridad.

Generalmente el *sendmail* es razonablemente seguro cuando se siguen los procedimientos de instalación de la mayoría de los vendedores. Existen, sin embargo, algunas precauciones a tomarse que asegurarán la operación confiable.

4.2.4 Seguridad en la Red (posibles soluciones)

En este punto se mencionan algunas herramientas y métodos para hacer nuestras redes UNIX lo más seguras posibles.

4.2.4.1 HOSTS CONFIABLES

En seguida describiremos cómo implementar la facilidad de host confiable preservando lo más posible la seguridad.

A) El archivo hosts.equiv

El archivo /etc/hosts.equiv indica los hosts confiables de manera que el usuario al accesar uno de ellos para entrar a los demás no se le requiera un contraseña.

Un cuidado adecuado debe verificar que sólo se permitan hosts locales en dicho archivo, tomando en cuenta en qué lugar se encuentran dichos hosts, es decir si tienen acceso "público" y podrían significar un agujero en la seguridad.

B) El archivo .rhosts

Este archivo permite acceso confiable sólo a ciertas combinaciones host-usuario, en vez de a hosts en general. Cada usuario puede crear un archivo .rhosts en su directorio *home*, y permitir acceso a su cuenta sin una contraseña. Este archivo presenta un problema de seguridad mayor: mientras que el hosts.equiv se encuentra bajo el control del administrador y puede ser manejado efectivamente, cualquier usuario puede crear un archivo .rhosts otorgando acceso a quien quiera que el desee, sin el conocimiento del administrador.

La única manera de manejar los archivos .rhosts es no permitirlos de ninguna manera dentro del sistema.

4.2.4.2 TERMINALES SEGURAS.

Un super usuario (root) nunca debe permitírsele conectarse desde una terminal no segura aún introduciendo la contraseña, preferentemente sólo debe permitírsele

conectarse desde la consola. El archivo /etd/ttytab es empleado para controlar que terminales se les considera seguras[xxx].

Lo configuración más segura es remover la designación "secure" (segura) de todas las terminales, incluyendo la consola. Esto requiere que aquellos usuarios con autoridad de super usuario primero se conecten como ellos mismos y después tornen a super usuario mediante el comando *su*. Esto también obliga a que la contraseña de *root* se introduzca cuando se reinicialice (reboot) en modo mono usuario, con el objeto de prevenir que los usuarios reinicialicen sus estaciones de trabajo y obtengan acceso de super usuario. Esta es la manera en que las máquinas clientes sin disco deben ser configuradas.

4.2.4.3 NFS (NETWORK FILE SYSTEM)

A continuación mencionaremos algunas opciones para hacer NFS más seguro y dificultar el acceso no autorizado mediante NFS.

A) El archivo exports.

El archivo */etc/exports* lista los archivos que son exportados (disponibles para montarse). Un archivo export por ejemplo sería:

/usr

/home

/export/root/client2 *-access=client2,root=client2*

El comando *root=* especifica la lista de hosts a los que les está permitido tener acceso de super usuario al sistema de archivos. El comando *access=* especifica la lista de hosts (separados por comas) a los que les está permitido accesar el sistema de archivos. Si no se especifica un comando *access=* para un sistema de archivos, cualquier host en cualquier parte de la red puede montar el sistema de archivos mediante NFS, por lo que es importante que siempre se incluya el comando *access=* en todo sistema de archivos.

Después de hacer cualquier cambio se debe correr el comando

 # exports -a

para que todos los cambios tengan efecto.

B) Restricción al acceso de super usuario

El archivo *exports* también permite otorgar el acceso de super usuario a ciertos sistemas de archivos por ciertos hosts usando el comando *root=* puede incluirse una lista de hasta diez hosts. Nunca se le debe dar acceso de "root"a un host no confiable.

[xxx] Bajo algunas versiones de de UNIX de Berkeley (no Sun) este archivo es llamado /etc/ttys

Un problema sobre NFS es la dificultad de sincronizar los números de usuarios en las dos máquinas, es decir, la que está exportando las tablas y la que las recibe, si los sistemas no son homogéneos respecto a sus usuarios puede crearse incompatibilidad y por lo tanto dejarse puertas abiertas. Por esto se debe ser precavido y verificar las tablas.

4.2.4.4 FTP

A) FTP Anónimo

Para configurar el FTP anónimo de manera segura se deben seguir las siguientes instrucciones:

1. Crear una cuenta "ftp". Deshabilitar la cuenta poniendo un asterisco (*) en el campo de la contraseña. Darle a dicha cuenta un directorio home especial, como */usr/ftp* o */usr/spool/ftp*.

2. Hacer dueño del directorio a "ftp" y no escribible por nadie:

```
# chown ftp ~ftp

# chmod 555 ~ftp
```

3. Hacer el directorio *~ftp/bin*, perteneciente al super usuario y no escribible por nadie. Poner una copia del programa *ls* en este directorio:

```
# mkdir ~ftp/bin

# chown root ~ftp/bin

# chmod 555 ~ftp/bin

# cp -p /bin/ls ~ftp/bin

# chmod 111 ~ftp/bin/ls
```

4. Hacer al directorio *~ftp/etc*, perteneciente al super usuario y no escribible por nadie. Poner copias de los archivos de contraseña y de grupo en este directorio, con todos los campos de contraseñas cambiados a asteriscos. Es posible borrar algunas de las cuentas de estos archivos, la única cuenta que es obligatorio que esté presente es "ftp".

```
# mkdir ~ftp/etc

# chown root ~ftp/etc

# chmod 555 ~ftp/etc

# cp -p /etc/passwd /etc/group ~ftp/etc

# chmod 444 ~ftp/etc/passwd ~ftp/etc/group
```

5. Hacer el directorio *~ftp/pub* perteneciente a "ftp" y escribible por todos. De esta manera los usuarios pueden poner archivos que sean accesibles mediante FTP anónimo en este directorio"

 # mkdir ~ftp/pub

 # chown ftp ~ftp/pub

 # chmod 777 ~ftp/pub

El FTP anónimo no debe estar disponible en todos los hosts de la red, preferiblemente se debe elegir una sola máquina. Si se le permite a la gente transferir archivos a nuestra máquina (en el directorio *pub*) se sugiere verificar frecuentemente el contenido de los directorios en los que les está permitido escribir. Cualquier archivo sospechoso debe ser borrado.

B) FTP Trivial

Este protocolo permite a los hosts sin disco el inicializarse desde la red. Se deben verificar los hosts ejecutando la siguiente secuencia de comandos mostrada en seguida:

% tftp

tftp> connect nuestrohost

tftp> get /etc/motd tmp

Error code 1: File not found

tftp> quit

%

Si no responde con *"File not found"*, y transfiere el archivo, se debe reemplazar la versión de tftpd por una nueva.

4.2.4.5 CORREO (MAIL)

El *sendmail* es razonablemente seguro cuando es instalado mediante los procedimientos indicados por los vendedores. Existen, sin embargo, algunas precauciones que deben tomarse para asegurar la operación:

1. Remover el alias "decode" del archivo de alias */etc/aliases/* o */usr/lib/aliases*.
2. Si nosotros creamos alias que permitan que se envíen mensajes a los programas, debemos asegurarnos de que no haya manera de obtener un shell o enviar comandos a un shell desde estos programas.

3. Asegurarse de que la contraseña "wizard" este deshabilitado del archivo de configuración *sendmail.cf*.

4. Asegurase de que nuestro *sendmail* no soporte el comando "debug". Esto puede hacerse mediante los comandos:

```
% telnet localhost 25
220 yourhost Sendmail 5.61 ready at 9 Mar 95 10:57:36 PST
debug
500 Command unrecognized
quit
%
```

Si nuestro sendmail responde con *"200 Debug set"*, entonces somos vulnerables a un ataque y debe reemplazarse el sendmail con una nueva versión.

4.2.4.6 FINGER

Si nuestra versión de *fingerd* es anterior al 5 de noviembre de 1988 debe ser reemplazada con una nueva ya que esta contiene varios errores que se pueden aprovechar como agujeros de seguridad.

4.2.5 Seguridad en el Sistema de Archivos

La última defensa contra los crackers en el sistema son los permisos ofrecidos por el sistema de archivos. Cada archivo o directorio tiene tres conjuntos de permisos asociados a él: un conjunto para el usuario que es dueño del archivo, otro para los usuarios en el grupo con el que el archivo está asociado, y uno último para todos los otros usuarios. Cada conjunto contiene tres bits de permisos idénticos, que controlan lo siguiente:

Lectura Si está habilitado el directorio o el archivo puede ser leído. En el caso del directorio significa que se puede ver el contenido del directorio (qué archivos y directorios contiene).

Escritura Si está encendido el directorio o archivo puede se modificado, en el caso de un directorio implica que se puede crear, borrar y renombrar archivos.

Ejecución El archivo puede ser ejecutado, en caso de un directorio implica la habilidad de accesar archivos contenidos en el directorio.

Además, un bit para un cuarto permiso está disponible en cada conjunto de permisos. Este bit tiene un diferente significado en cada conjunto de permisos:

setuid Si está prendido en los permisos de dueño, el bit controla el estado "set user id" (setuid) de un archivo. El estado setuid significa que cuando un programa

es ejecutado, se ejecuta con los permisos del usuario que es dueño del programa, en adición a los permisos del usuario ejecutando el programa. Por ejemplo, *sendmail* es setuid "root" permitiéndole escribir a archivos en el área de colas de correo, que los usuarios normales no pueden hacer. Este bit no tiene significado en los archivos no ejecutables.

setgid Si está encendido en los permisos de grupo, el bit controla el estado de un archivo "set group id" (setgid). Esto funciona exactamente de la misma manera en que el bit setuid, a excepción de que el identificador de grupo es el afectado. Este bit no tiene significado en los archivos no ejecutables.

sticky Si está habilitado en los permisos de todos los otros (world), el bit "sticky" le dice al sistema operativo que haga cosas especiales con la imagen del texto de un archivo ejecutable.

4.2.5.1 ARCHIVOS DE COMANDOS CON EL SETUID

Los archivos de comandos (shell scripts) que tienen el bit setuid o setgid encendidos no son seguros, independientemente de los muchos cuidados que se hayan tomado al escribirlos. Existen numerosos programas disponibles que dicen que tienen archivos de comandos seguros, pero hasta ahora todos los que se han liberado no resuelven todos los problemas.

4.2.5.2 EL VALOR UMASK

Cuando un archivo es creado por un programa, ya sea un editor de textos o un compilador, es creado típicamente con todos los permisos habilitados. Como esto es raramente deseable (nadie quiere que los otros usuarios puedan escribir a sus archivos) el valor umask puede usarse para establecer los permisos de un archivo al ser creado.

4.2.5.3 ENCRIPTAMIENTO DE ARCHIVOS

El comando estándar de UNIX *crypt*[xxxi] no es del todo seguro. El crypt implementa una máquina a través de las líneas del Enigma Alemán (German Enigma) roto en la Segunda Guerra Mundial. Los métodos de ataque en dicha máquina son bien conocidos, y un archivo lo suficientemente largo puede ser usualmente decriptado en unas pocas horas aún sin el conocimiento sobre qué contiene el archivo[xxxii]. De hecho, desde hace varios años, existen programas disponibles públicamente diseñados para "romper" archivos encriptados con crypt.

Existen implementaciones de otro algoritmo, el DES (Data Encryption Standard), disponible en algunos sistemas.

[xxxi] Sun Microsystems. *SunOS Reference Manual*, Part Number 800-1751-10, May 1988. p. 95

[xxxii] Reeds, J.A. and P.J Weinberger. *File Security and the UNIX System Crypt Command.* AT&T Bell Laboratories Technical Journal, 1673-1683 pp. Oct. 1984

4.2.5.4 DISPOSITIVOS

La seguridad de los dispositivos es un punto muy importante en UNIX. Los archivos de dispositivos (que usualmente residen en /dev) son usados por varios programas para accesar datos en los discos o en memoria. Si estos archivos de dispositivos no están apropiadamente protegidos, el sistema está ampliamente abierto a un cracker.

4.2.6 Seguridad en el Sistema de Archivos

La última defensa contra los atacantes a nuestro sistema son los permisos ofrecidos por el sistema de archivos. Anteriormente se explicaron estos permisos, la correcta y adecuada asignación de éstos evitará agujeros en la seguridad.

Además:

1. Los shell scripts con los bits setuid o setgid encendidos no deben ser permitidos nunca en ningún sistema UNIX.
2. La cuenta "root" debe tener la línea

   umask 022

en su archivo */.cshrc* para prevenir la creación accidental de archivos de escritura por todos y pertenecientes al super usuario.

4.2.6.1 ENCRIPTADO DE ARCHIVOS

Tal vez, lo mejor que hay que decir acerca de encriptar archivos en nuestro sistema es esto: si creemos que el contenido de un archivo es tan importante como para encriptarlo, entonces ese archivo no debería estar almacenado en esa computadora. Esto es especialmente cierto en sistemas con una seguridad limitada como los sistemas UNIX y las computadoras personales. Sin embargo, en la transferencia de archivos, el encriptarlos representa la única opción que podría garantizarnos (en mayor porcentaje) que no será accesada o alterada la información por algún atacante.

4.2.6.2 DISPOSITIVOS

La seguridad de los dispositivos es un punto muy importante en UNIX. Los archivos de dispositivos (usualmente residentes en */dev*) son empleados por varios programas para accesar los datos en los discos o en memoria. Si estos archivos de dispositivos no están protegidos adecuadamente, nuestro sistema está completamente abierto a un atacante. La lista completa de dispositivos es muy larga para incluirla aqui, y además varía de sistema a sistema. Sin embargo la siguiente guía es aplicable a todo sistema:

1. Los archivos */dev/kmem, /dev/mem* y */dev/drum* nunca deben ser leíbles por todos. Deben pertenecer a "root" y al grupo "kmem", y deben estar en modo 640.

2. Los dispositivos de disco, como */dev/sd0a*, */dev/rxy1b* deben pertenecer a "root" y al grupo "operador" y deben estar en modo 640.

3. Con muy pocas excepciones, todos los otros dispositivos deben pertenecer a "root" (una excepción son las terminales).

Un último comentario respecto a la seguridad en UNIX (y cualquier sistema operativo) es no olvidar que los proveedores frecuentemente ofrecen "parches" para el sistema, es muy importante probarlos e instalarlos. De la misma manera debemos estar atentos a los agujero de seguridad que encuentren administradores o usuarios, si el proveedor no ofrece alguna opción para cerrar dicho agujero deberemos considerar el hacerlo nosotros mismos.

4.3. Problemas Físicos de Seguridad

En el problema de mantener la seguridad de nuestra información es posible visualizar diferentes capas de protección[xxxiii] y atender cada una de ellas. Dichas capas generalmente son las de la figura:

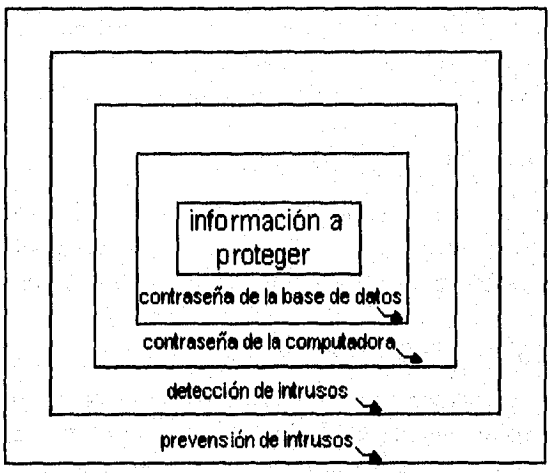


Fig. 4.1 Capas de protección de la información.

La figura nos confirma que la idea de seguridad física es frecuentemente la primer barrera con la que se enfrenta el adversario. De ahí la importancia de que analicemos dicha barrera.

Los aspectos a cuidar físicamente[xxxiv] incluyen hardware (computadoras, dispositivos de memoria, equipo periférico), redes (equipo de transmisión y recepción, y

[xxxiii] James Arlin Cooper .*Computer and Communications Security.* pág 51

[xxxiv] Ibid pág 41.

medios de comunicación), y personal (operadores, personas de mantenimiento, usuarios, administradores).

En lo que concierne al hardware específicamente nos referimos a computadoras y periféricos, y respecto a las redes incluye los componentes de red en sí mismos (debido a los problemas de intervención o sabotaje), así como los datos que son transmitidos a través de las redes.

El primer problema que trataremos será la prevención de intrusos que se relaciona con el cuidado del hardware. Posteriormente hablaremos de las debilidades de la parte física y de los problemas de seguridad en la red.

4.3.1 Prevención de Intrusos[xxxv]

Los peligros de penetración física tienen que ver con distintas vulnerabilidades. Incluidas en éstas se encuentran el robo de hardware, o software; comprometer datos (copiándolos o leyéndolos); daño o alteración de hardware, software o datos y la adjudicación no debida de recursos (robo de tiempo de cómputo o de comunicación, etc.)

A decir verdad, una barrera de protección física es más bien una medida de retrasamiento. Esto significa que un adversario, con suficiente tiempo y recursos, puede penetrar cualquier barrera física. La efectividad de las medidas es entonces dependiente del tiempo de retardo en que el adversario logre saltar la barrera.

4.3.1.1 DESENFATIZAR LOS INMUEBLES

Una de las implementaciones a la seguridad física más sutiles de los años recientes es que los inmuebles protegidos sean lo menos obvios posibles, ya que por ejemplo, las ventanas son un medio muy sencillo de accesar un centro de cómputo.

4.3.1.2 VULNERABILIDADES DE LAS DEFENSAS

Existe un gran número de defensas empleados como barreras. Existen puntos a saber sobre estas medidas. Por ejemplo las estructuras de alambre no son del todo efectivas contra determinados adversarios con habilidades atléticas, además de que resulta obvia la vulnerabilidad del alambre al corte. Las paredes son otro medio de acceso, ya que en muchos de los casos se emplean paredes removibles, fáciles de pasar o que dejan espacio unas entre otras.

En algunos lugares, para restringir la entrada, se implementan chapas electrónicas. Uno de sus problemas más comunes es que los códigos de entrada son frecuentemente fáciles de leer para un observador.

[xxxv] Ibid pág 52 a 58

En cuanto a los guardias de seguridad, existen varias debilidades al respecto. Muchas veces pueden no observar correctamente la fotografía, o cuando existe una entrada numerosa de gente resultan ineficientes.

4.3.1.3 FACTORES HUMANOS

Un punto principal a tomar en cuenta, es el factor humano. Independientemente de las implementaciones físicas de seguridad, si uno de los empleados que tienen acceso a los equipos y posiblemente a las cuentas no le valdrá barrera alguna si quiere dañar a la empresa, y esto no es raro, en muchas ocasiones, inclusive dentro de la Universidad Nacional Autónoma de México se ha dado el caso de Administradores o ex-administradores que dañan el equipo, destruyen información valiosa o cambian información en su beneficio, de tal manera que para que nuestras barreras físicas tengan un valor real y aseguren nuestra información es muy importante la adecuada selección del personal.

4.3.2 Debilidades en la topología física y problemas de red

Uno de los principales problemas de seguridad físicos es la intervención de las líneas para "escuchar a escondidas" conocido también como *eavesdropping*, inyectar información falsa, o alterar mensajes. Al seleccionar el equipo para la instalación de una red debe de tomarse en cuenta la susceptibilidad del mismo a ser intervenido o alterado. En la siguiente lista se mencionan los tipos de cableado de acuerdo a la de facilidad que presentan para ser intervenidos:

- Par trenzado
- Cable coaxial
- Fibras ópticas (muy difíciles de intervenir)

La expansión física de las redes crea un buen número de problemas. A mayores áreas se requieren más gastos en protección física, ya que les provee mayores oportunidades a los interceptores o saboteadores. Las redes de acceso mediante líneas telefónicas (dialup networks) aumentan el riesgo de que el acceso pueda ser intentado desde más partes del mundo.

Los principales problemas de nuestro interés y que abordan el hecho de que la red atraviesa grandes distancias y límites, pueden ser categorizados como intercepción, alteración, sabotaje y pérdida de la integridad (intencional o accidentalmente causada).

4.3.2.1 INTERCEPCIÓN DE LA INFORMACIÓN

La intercepción de la información mediante intervención de las líneas es una preocupación común sobre todo en las redes sin encriptamiento. La intervención es concebible dentro de edificios, en líneas superficiales o subterráneas. Esta intervención

puede ser realizada directamente en el cable o bien mediante interventores inductivos remotos, que son más difíciles de localizar.

Todas las técnicas de intercepción bien conocidas son aplicables (con algunas variantes) en seguridad de comunicaciones y computación. En la siguiente parte describiremos las técnicas más comunes empleadas hoy en día[xxxvi].

A) Intercepción mediante conexión directa

La intervención directa de líneas telefónicas es relativamente fácil. Equipo para este propósito y de un bajo precio se encuentra disponible en muchas tiendas de electrónica[xxxvii]. Un ejemplo de punto de acceso se encuentra en el pedestal donde las líneas locales encuentran el cable de alimentación principal.

B) Acoplamiento pasivo

El acoplamiento pasivo mediante respuesta inductiva o capacitiva es otro método de intercepción. Con este método el adversario tiene más amplias opciones de localización, y menor probabilidad de detección. Un dispositivo para intervenir (tap) inductivo puede comprarse en unos $10 dls y un equipo completo (tap, interruptor de activación de señal, y grabadora de capacidad extendida) tiene un costo menor a los 200 dls[xxxviii].

C) Intercepción de Energía Electromagnética

Las transferencia de información que se lleva a cabo mediante microondas o mediante enlaces satelitales son vulnerables a la intercepción desde una más amplia variedad de localizaciones. Esto le permite al adversario ser menos notorio. Mucho del tráfico satelital se lleva a cabo en frecuencias públicas.

Las señales electromagnéticas pueden ser interceptadas con antenas relativamente pequeñas, y éstas pueden ser fácilmente camuflajeadas dentro de forros electromagnéticamente transparentes (por ejemplo fibra de vidrio o madera).

D) Intervención de Fibra Óptica

Frecuentemente las comunicaciones mediante fibra óptica son citadas como "seguras", ya que en ellas no existen emanaciones o modos de acoplamiento pasivos, y además son difíciles de intervenir debido al reducido diámetro de la fibra (comparable o menor al de un cabello humano). Sin embargo, es importante notar que la intervención es posible mediante personal especializado. Hay evidencia de que existen dispositivos de intervención extremamente difíciles de detectar. Por estas razones la conexión con fibra

---

[xxxvi] Ibid pág.201 -203

[xxxvii] National Security Agency. *Information Systems Security Products and Services Catalogue*, July 1988

[xxxviii] Idem

óptica debe ser vista más que como una protección completamente segura como un impedimento.

E) TEMPEST

TEMPEST fue inicialmente un término empleado para describir un programa del gobierno de los Estados Unidos para controlar "emanaciones comprometedoras". Actualmente se emplea como sinónimo de dichas emanaciones.

Se ha descubierto que emanaciones electromagnéticas inadvertidas tienen un potencial de transportar información hasta distancias sorprendentes. Adversarios con conocimiento y el equipo apropiado pueden detectar dicha información. Por ejemplo[xxxix], el cable coaxial aunque esté blindado, puede permitir que se fugue energía. Algunas de las fugas se pueden deber a daños o deterioración en el blindado o en los conectores. El par trenzado es un medio que resulta con más serios problemas de emanaciones.

4.4. Defensas en la Parte Física.

4.4.1 Control de Acceso

4.4.1.1 DETECCIÓN DE INTRUSOS

Debido a que la prevención de intrusos (tanto física como lógicamente) es difícil, y debe considerarse como una acción de retardo, la detección de intrusos es una implementación importante.

Algunos de los medios para prevenir intrusos ya fueron mencionados anteriormente, así como la mucha o poca seguridad que implican chapas electrónicas, alambrado, guardias, etc. Falta mencionar a los sensores y un punto básico para evitar entradas no deseables, tanto físicamente, como mediante software o intervención de las líneas: *la autenticación.*

A) Sensores

Existen varias categorías de dispositivos de detección, unos de ellos son los sensores. Estos pueden ser interruptores electromecánicos (sensibles a los cambios de posición), transductores piezoeléctricos (que convierten tensión a energía eléctrica) (cuando un intruso entra hay un cambio de voltaje y se detecta) , geófonos (para detectar sonidos), y cables eléctricos (para detectar cambios de posición a través de impedancia).

4.4.1.2 AUTENTICACIÓN

Cuando hablamos de autenticación nos referimos a la necesidad de asegurarnos de que quién dice se encuentra en el otro lado de la línea sea realmente quien pretende ser.

[xxxix] James Arlin Cooper. *Computer and Communications Security* pág. 300, 301

La primera forma de autenticar al usuario puede ser la contraseña, pero como ya se ha mencionado, no es la más segura además de los múltiples riesgos que implica.

Existen[xi] tres diferentes métodos mediante los cuales se puede verificar a los usuarios en el momento de conectarse: haciendo uso de algo que ellos conocen, de algo que ellos tienen o algo que ellos son. A continuación describiremos dichos métodos.

El método "algo que ellos conocen" está tipificado por el identificador del usuario (por ejemplo el nombre de su cuenta) y su contraseña, o bien, como en algunos otros sistemas, se pregunta por otra información personal como el apellido de soltera de su mamá. La idea es que este conocimiento no esté escrito y difícilmente disponible a un intruso. Pero como se recordará, el uso de contraseñas acarrea muchos problemas y vicios difíciles de atacar.

Con el método "algo que ellos tienen" se agrega un segundo nivel de confidencialidad al proceso de autenticación, con este se requiere que el usuario posea un objeto que le autorice el acceso. Mientras que este objeto puede ser tan simple como una tarjeta de plástico magnética de barras, la solución más común actualmente toma la forma de un generador aleatorio de contraseñas o un dispositivo que envíe cierta respuesta como identificación.

Un tipo de *token*, que es como son llamados estos dispositivos, provee un número o palabra alfanumérico seudoaleatorio que cambia cada minuto o algo similar y se encuentra sincronizado con la base de datos. Esto resulta en un contraseña de una sola vez (*one time password*) que es bueno sólo en ese particular punto en el tiempo y para una sola conexión. El primer token que apareció en el mercado en 1987 -el securID de Security Dynamics en Cambridge, Massachussets- empleaba este sistema.

Otro tipo de token es un dispositivo como calculadora en el que el usuario teclea un número, el token genera una respuesta que el usuario introduce en su estación de trabajo. De nuevo, el resultado es un contraseña de una sola vez, no reutilizable. Sólo si el usuario tiene el token consigo es posible la conexión. Además, muchos de estos dispositivos pueden ser configurados para requerir al usuario que introduzca un número de identificación personal antes del proceso de autenticación.

Algunos otros tokens empleados actualmente son[xii]:

Identificación con Fotografía, la más común es una credencial con una fotografía que puede ser verificada por un guardia visualmente.

Un sistema de intercambio de identificaciones reduce la posibilidad de que una identificación se pierda, sea perdida o robada. Se mantienen identificaciones duplicadas

[xi] Russel Kay. *Distributed and Secure* BYTE June 1994. p. 165-178

[xii] Baker, Richard H. *Computer Security Handbook.* TAB Professional and Reference Book, 2nd edition, 1991. p139,140

en cada punto de entrada controlada. Cuando un empleado solicita el acceso, un guardia compara al individuo con la identificación correspondiente, si el individuo pasa la verificación se le otorga el acceso y se intercambian las identificaciones, cuando el empleado deja las instalaciones se repite el intercambio. Sin embargo este sistema no previene que alguien emplee maquillaje para coincidir con la imagen almacenada en la identificación.

Identificación óptica codificada. Este tipo de identificación contiene un arreglo geométrico de puntos impresos en una inserción que está laminada dentro de la identificación. Foto detectores en el lector de tarjeta verifican la transmisión óptica de los puntos. Para hacer esta identificación resistente a la intemperie, los puntos pueden ser ocultados de tal manera que sólo puedan ser detectados por luz infrarroja.

Identificación con circuito eléctrico codificado . Esta es una tarjeta de plástico que contiene un circuito impreso que puede selectivamente cerrar circuitos eléctricos cuando es insertada en una lectora.

Identificación magnética, ésta contiene una hoja de material magnético flexible en el cual un arreglo de puntos han sido permanentemente magnetizados. El lector de la tarjeta contiene sensores que son verificados eléctricamente o interruptores que se activan mecánicamente.

Los puntos pueden ser borrados accidentalmente si la tarjeta es expuesta suficiente tiempo a un campo magnético fuerte, pero en la práctica este no ha sido un problema muy serio.

Identificación magnética con código de barras, esta es una identificación ampliamente usada en sistemas de tarjetas de crédito, y muchos vendedores hacen equipo que es compatible con el American National Standard Institute (ANSI) para esta técnica.

Con este tipo de tarjeta una tira de material magnético es ubicada a lo largo de un lado de la tarjeta y es codificado con datos que la identifiquen. Una cabeza magnética lee los datos. La falsificación es relativamente fácil, porque los datos de la barra magnética puede ser decodificado y aplicado a varias tarjetas. Todo lo que se necesita es una grabadora común.

Identificación electrónica pasiva codificada. En esta el lector genera un campo de radio frecuencia y verifica las frecuencias a las cuales energía significativa es absorbida. Estas frecuencias corresponden con la información de identificación codificada en la tarjeta.

Una importante ventaja de esta técnica es que la tarjeta no necesita ser insertada en ningún mecanismo, todo lo que se necesita es acercarla a una antena.

Tarjeta codificada mediante capacitancia. Esta es una identificación en la cual un pequeño arreglo de platos conductores han sido laminados. Ciertos platos están

conectados y el código se lee mediante medidas de capacitancia de los platos y distingue cuáles están conectados y cuáles no.

Identificación de barra metálica codificada La barra metálica codificada usa barras de cobre que están laminadas dentro de la tarjeta. La presencia o ausencia de barras en ciertos renglones determina el patrón del código, que se lee mediante un sensor.

Identificación electrónica activa. Este sistema consiste de una unidad electrónicamente codificada, y estacionaria. La unidad provee energía a un inductor magnético en la tarjeta. Este recibe y decodifica el número transmitido por la tarjeta.

Colocando en lugares estratégicos la lectora, el usuario no tiene que hacer nada, porque la tarjeta se lee automáticamente mientras el usuario pasa por un campo de radio frecuencia generado por la unidad de interrogación.

Obviamente el uso de tokens crea problemas administrativos, tales como que el empleado olvide el token en su casa. Este problema puede solucionarse manejando un almacenamiento cuidadosamente controlado de los tokens.

Además de los tokens, están apareciendo en el mercado las llamadas *smartcards* que son también implementaciones a la seguridad, emplean un dispositivo del tamaño de una tarjeta de crédito que contiene un microprocesador y memoria de escritura. También NIST (National Institute of Standards and Technology) ha desarrollado una tarjeta para generar firmas digitales usando su estándar propuesto. Dicho estándar emplea el uso de un algoritmo de encripción de NIST para no permitir la imitación del "firmado" de documentos.

Existe un uso potencial a futuro en la autenticación de aplicaciones mediante la tecnología de marcado activo (active-badge technology) en la cual el usuario tiene una marca o distintivo que transmite una señal de radio que es captada por un receptor especial cuando la señal se encuentra dentro de un rango. La ventaja de este método es que la autenticación tiene lugar automáticamente sin necesidad de contacto físico entre el sensor y el distintivo.

Actualmente existen entre 15 y 20 proveedores de sistemas de autenticación basados en *smartcards* o *tokens*.

Cabe mencionar que aquellos sistemas que requieren de hardware adicional -como smartcards o tarjetas magnéticas- no han sido del todo exitosos por el incremento en los costos. El precio es un factor crítico para la selección de tecnologías de seguridad en muchas organizaciones.

Un producto reciente que trata de atacar a este problema es el *SmartDisk* de "SmartDisk Security" en Naples Florida. El costo del producto es de $150 (dólares) que consiste de una smartcard en un disco flexible que puede ser leído por cualquier unidad de lectura estándar de 3 1/2 pulgadas.

El último y más seguro de los niveles de autenticación es "algo que ellos son", este método envuelve un aspecto que es único e inimitable de un individuo que es su cuerpo. En el pasado la *autenticación biométrica* como se le ha llamado, ha estado basada en comparaciones de huellas digitales, de la palma de la mano, patrones de la retina del ojo, verificación de firmas o reconocimiento de voz. Más recientemente ha aparecido un sistema que reconoce patrones de tecleado. Otra tecnología puede leer patrones faciales infrarrojos empleando una cámara de video para capturar las imágenes.

El ejemplo que nos es más familiar es el de la huella digital[xiii] las elevaciones y líneas que aparecen en nuestros dedos son únicas en el mundo. Los actuales dispositivos biométricos reemplazan a la tinta con un lector electrónico rastrea el pulgar mientras este es presionado en un botón.

El sensor traduce su lectura en señales digitales, que pueden ser comprimidas y almacenadas electrónicamente. Cuando el sistema lee la impresión del dedo lo compara con la señal almacenada en sus archivos de usuarios autorizados.

Otros dispositivos biométricos trabajan en las mismas bases pero leen diferentes características:

Lectora de retina, el fondo del ojo contiene delgadas venas. Ellas se acomodan en ciertos patrones que son únicos. El rastreador lee el tamaño, localización y ordenamiento de estas venas.

Firma dinámica, se enfoca en la manera de escribir. Un atacante puede imitar la apariencia de la firma pero no replicar los cambios sutiles de presión y movimiento que usamos mientras firmamos. Esta técnica lee estas señales y las compara con una versión avanzada de nuestra firma personal.

Tecleo dinámico ofrece la misma técnica que la anterior pero en el teclado. El sensor almacena los movimientos y presión que son una diferencia única mientras tecleamos una frase estándar.

Geometría de la mano mide el largo de los dedos, lo angosto y la forma de la mano. También calibra que tan delgada es la piel.

Reconocimiento de Voz ha sido desarrollada hasta el punto en que no sólo es una medida del sonido. Rastrea la fisiología que produce el hablar.

Tecnología de Redes Neuronales es una versión de alta tecnología de la fotografía instantánea. Rastrea los patrones de los nervios en la cara.

La *huella del DNA* toma una imagen genética y la compara con una almacenada.

Los sistemas biométricos ofrecen el mayor grado de confianza de que el usuario es quien dice ser, pero generalmente son también los más caros de implementar, tal vez su

[xiii] Ibid, p. 133

mejor aplicación es en las puertas de acceso o en otros tipos de accesos físicos, donde un sistema biométrico puede operar sobre muchos empleados. Otra inconveniencia que acarrea es el tiempo en verificar un acceso, algunos sistemas pueden tomar de 10 a 30 segundos en responder a una petición de acceso. Quizá el más grave problema es que los usuarios se muestran renuentes a tener que introducir un dedo o una mano en un orificio, a firmar, o a sentarse mientras un sistema óptico captura la imagen de su retina.

A) Ocultamiento de la Información de Autenticación

Una de las principales ventajas de los tokens es su habilidad para transmitir sobre una red un código de acceso que no contiene información reutilizable y sin sentido útil. Existe software que puede hacer lo mismo, la NSA (National Security Agency) ha liberado lo que llama "generadores de contraseña no olfateables" que en vez de enviar un contraseña tal cual por el cable, envían una representación encriptada de éste, usando una llave de encriptado de una sola vez, cada vez que se inicia una conexión la contraseña es encriptada con una llave diferente.

B) *Kerberos*

Kerberos[xliii] es un servicio de autenticación desarrollado como parte del proyecto Athena en el MIT en 1983 y debe su nombre al perro de la mitología griega que resguarda la entrada al Hades, como el Kerberos griego tiene tres cabezas se suponía que el sistema Kerberos tendría tres componentes para resguardar a la red: autenticación, contabilización y auditabilidad. Las últimas dos cabezas nunca fueron implementadas.

El problema que intenta resolver es: asumiendo un ambiente abierto distribuido en que los usuarios en las estaciones de trabajo desean accesar servicios en servidores distribuidos a través de la red, desearíamos que los servidores fueran capaces de restringir el acceso a usuarios autorizados y de autenticar las peticiones de servicios.

Kerberos provee uno o más servidores de autenticación centralizado cuya función es probar la identidad de un usuario por cada servicio requerido y también probar la identidad de los servidores a los usuarios, confiando exclusivamente en encripción convencional.

Kerberos[xliv] es actualmente el estándar de facto para autenticación de las comunicaciones a través de la red.

Tal vez, el hecho más significativo es que la "Open Software Fundation" de la DCE (Distribuded Computing Environment) usa una variante de la versión 5 de Kerberos como su mecanismo para autenticación. Otra organización que soporta Kerberos es la OCSG (Open Computing Security Group) un especializado integrador de sistemas. OCSG provee paquetes de Kerberos que soportan un número de diferentes plataformas,

[xliii] Stallings, William. *Network and Internetwork Security*. Principles and practice. Prentice Hall, U.S.A., 1995 p.314

[xliv] Kay, Russel. *Distributed and Secure*. BYTE June 1994 p. 165 -178

incluyendo MS-DOS, Macintosh, SunOS, HP-UX de Hewlett-Packard, NextStep y AIX de IBM para las RS/6000.

Otros vendedores ofrecen implementaciones de Kerberos para Ultrix de DEC y plataformas VMS. IBM dice que ofrecerá Kerberos en sus mainframes MVS y sus sistemas OS/2. Y por supuesto también se puede obtener (con un permiso especial) el código fuente del MIT.

Sin embargo Kerberos no es la solución completa, aunque provee autenticación de usuarios y servidores, no maneja autorización para aplicaciones o para transacciones dentro de las aplicaciones. Cualquier determinación de autorizaciones de acceso y derechos debe ser manejado por otro sistema en la red.

Además, lo que se ha llamado *Kerberización* no es simple ni rápido. Afortunadamente parece que muchos de los vendedores de los productos relacionados (software, sistemas operativos, enrutadores, etc.) están trabajando duro en la kerberización de sus productos.

Según usuarios reconocidos Kerberos está todavía fuera de la realidad, algunos usuarios como John O'Leary (director de educación del CSI Computer Security Institute) mencionan que Kerberos agrega gran cantidad de tráfico y el manejo de boletos llega a ser una gran molestia, pero si se tiene un gran ancho de banda y enlaces de alta velocidad entonces si puede trabajar bien.

Por su importancia más adelante se hablará con más detalle del funcionamiento y filosofía de Kerberos.

4.4.2 Seguridad en la red

4.4.2.1 INTERCEPCIÓN DE LAS LÍNEAS

A) Cableado

Aunque nos parezca ya muy familiar el cable de cobre, no debemos tomarlo por seguro. Debería considerarse obligatorio el apropiado equipo de pruebas, especialmente cuando existe cableado oculto. Un examinador de cable es el único modo para verificar un cable que pasa a través de una pared. Probando cada cable durante la instalación se puede identificar cada extremo de las ligas antes de que se deterioren y causen fallas en la red.

La mínima tecnología empleada para pruebas de cable es probar la continuidad del mismo, ya sea mediante un voltímetro o con un baratísimo probador de baterías. Sin embargo, los requerimientos de los cables para datos son más complicados que el sólo conducir corriente. En las frecuencias de las señales empleadas por LANs, los cables de datos son dispositivos electrónicos extremadamente complejos que pueden hacerse inoperables por múltiples razones.

Por lo anterior se han desarrollado más sofisticados probadores de cables que los examinan bajo condiciones de red (el precio está entre los 1000 a 5000 dólares), aunque el costo del probador pueda parecer demasiado, el dispositivo pagará por sí mismo la primera ocasión que permita restaurar la función propia de la red en minutos en vez de horas o días.

Una de las técnicas disponibles actualmente para probar es el TDR (Time Domain Reflectometer) que puede examinar un cable por uno de los extremos y puede, por ejemplo, estimar la distancia desde el extremo a donde pudo haber ocurrido la falla.

Como una solución a la intervención de las líneas se sugieren detectores de nivel de señal y/o detectores de interferencia de onda que pueden ser empleados para proveer una señal de alarma al detectar una reducción en la energía recibida. Esto es útil principalmente en sistemas de banda ancha, incluyendo fibra óptica.

En la selección del cable a emplear al instalar una red deben tomarse en cuenta las necesidades de la empresa (costo-seguridad), ya que , a pesar de que se mencionó que el cable más seguro es el de fibra óptica, es más caro y de más difícil instalación, pero si la seguridad es importante la inversión se pagará por sí misma, sin embargo, cabe reafirmar que no podemos confiar en que la fibra óptica es segura, existen métodos de intervenirla por lo que se debe considerar más bien una barrera de seguridad, y una buena inversión debido a sus características de funcionamiento.

B) Intervención de las líneas

En general, para los diferentes tipos de intervención de las líneas la defensa principal y a veces la única es *la encripción* de la que hablaremos más adelante. También puede, en algunos casos, resultar útil el empleo de equipo especializado para detectar cambios de energía o similares para detectar una posible intercepción (por ejemplo los reflectómetros de tiempo[xliv]).

Para el caso específico de TEMPEST[xlvi], la técnica de defensa más obvia es asegurar una distancia considerable entre un adversario y el equipo, así como con conductores fortuitos como líneas telefónicas. La transmisión mediante fibra óptica es de gran ayuda, así como la encripción.

C) Módems

Estos dispositivos presentan un problema de seguridad potencial. Explicar como configurar cada módem llevaría varios volúmenes. Sin embargo los siguientes puntos ayudan a verificar nuestros módems:

[xliv] Arlin Cooper, James. *Computer and Communications Security*. p. 202

[xlvi] Ibid p.204,205

- Si un usuario conectado a un módem cuelga el teléfono, el sistema debe desconectarlo. Si esto no ocurre, se deben verificar las conexiones de hardware y la configuración de los puertos seriales.
- Si un usuario se desconecta, el sistema debe forzar al módem a colgar. Si no ocurre hacer lo mismo que en el punto anterior.
- Si la conexión de una terminal a el sistema es rota, el sistema debe desconectar al usuario.
- Si una terminal está conectada a los módems y el usuario cuelga la terminal debe informar al sistema que el usuario colgó.

Muchos manuales de los módems cubren en detalle como conectar apropiadamente estos dispositivos a nuestro sistema. En particular, se debe poner atención a las conexiones: *"Carrier Detect"*,*"Clear to Send"* y *"Request to Sent"*.

4.4.2.2 ENCRIPTADO

Si tomamos una carta, la guardamos en una caja de seguridad, y ésta última la escondemos en alguna parte, y entonces le pedimos a alguien leer la carta, eso no es seguridad. Eso es obscuridad. Por otro lado, si tomamos la carta, la guardamos en la caja de seguridad y le damos a alguien la caja así como las especificaciones de diseño de la misma, y un ciento de las mismas cajas de tal modo que los mejores rompedores de seguros puedan estudiar el mecanismo de la cerradura... y aún así no pueden abrirla y leer la carta, eso es seguridad.

Por muchos años esta clase de criptografía fue de dominio exclusivo de los militares.

Durante los últimos veinte años, ha habido una gran explosión de investigación académica pública en criptografía.

Ahora nos es posible emplear prácticas de seguridad para protegernos de los más poderosos adversarios -incluso de las agencias militares.

En este trabajo no se pretende proporcionar un texto matemático o explicar los diferentes algoritmos de encriptado, sino dar una introducción y las principales bases teóricas para adentrarse en el fabuloso mundo del criptoanálisis.

A) Terminología

Emisor y Receptor

En esta área suponemos que alguien, a quien llamaremos *emisor*, desea enviar un *mensaje* a alguien más, a quien llamaremos *receptor*. Por otra parte, el emisor desea asegurarse de que ningún intermediario pueda afectar el mensaje de ninguna manera;

específicamente, que no pueda interceptar y leer el mensaje, interceptarlo y modificarlo o interceptarlo y fabricar un mensaje substituto.

Mensajes y encriptado

A un mensaje se le llama de dos formas: *texto plano* y *texto claro*. El proceso de disfrazar al mensaje de manera que se esconda su contenido es llamado *encripción o encriptado*. A un mensaje encriptado se le llama *texto cifrado*. El proceso de regresar el texto cifrado a texto plano se llama *decripción o decriptado*.

Al arte y ciencia de mantener los mensajes seguros se le conoce como *criptografía*, y es practicada por los *criptógrafos*. Los *criptoanalistas* son quienes practican el *criptoanálisis*, el arte y ciencia de romper el texto cifrado. La rama de las matemáticas que abarca tanto a la criptografía como al criptoanálisis es llamada *criptología*, y quienes la practican son llamados *criptologistas*. Actualmente, casi todos los criptologistas son matemáticos teóricos (tienen que serlo).

Fig.4.2 Encriptado y Decriptado

El texto plano es denotado por *P*. Este puede ser una cadena de bits, un archivo de texto, voz o imagen de video digitalizada. En lo que concierne a computación, P son simplemente datos binarios. El texto plano puede existir tanto para transmisión como para almacenamiento. En cualquier caso P es el mensaje a encriptar. Entonces, la función de encriptado es:

$$E(P) = C$$

En el proceso de vuelta, la función de decriptado D opera en C para producir P:

$$D(C) = P$$

Desde el punto de vista de que encriptar y decriptar un mensaje es recuperar el texto plano original, la siguiente identidad debe ser cierta:

$$D(E(P))=P$$

Algoritmos y cifradores

Un algoritmo criptográfico es llamado también un *cifrador*. Un cifrador es la función matemática para encriptar y decriptar. Al encriptar un mensaje de texto plano se aplica el *algoritmo de encriptado*. Al decriptarlo se aplica al texto cifrado el *algoritmo de decriptado*.

Si la seguridad de un algoritmo se basa en mantener la naturaleza del encriptado secreta, se le llama *restringido*. La mayoría de los criptosistemas restringidos son triviales

de romper por criptoanalistas experimentados. A causa de esto, los algoritmos restringidos son enormemente populares en aplicaciones de baja seguridad.

Para una seguridad real, los algoritmos modernos de encriptado emplean una *llave*, denotada por *k*. Esta puede tomar uno de muchos valores (mientras mayor sea el número mejor). El rango de valores posibles para la llave se llama *espacio de llaves.*

El valor de la llave afecta a las funciones de encriptado y decriptado, de manera que estas se convierten en:

$E_k(P) = C$

$D_k(C) = P$

Y si la llave de encriptado y decriptado son las mismas entonces:

$D_k(E_k(P)) = P$

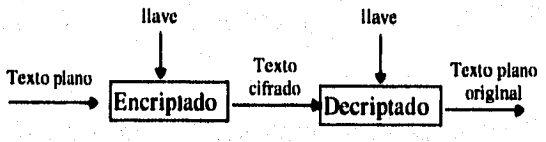


Fig. 4.3. Encriptado y Decriptado con llave

Existen algunos algoritmos en los que la llave de encriptado y decriptado vienen por pares. Esto es, la llave de encripción k_1, es diferente de la correspondiente llave de decriptado k_2. En este caso:

$E_{k1}(P) = C$

$D_{k2}(C) = P$

Y si la llave de encriptado y decriptado son la misma entonces:

$D_{k2}(E_{k1}(P)) = P$

Algoritmos simétricos y Algoritmos de llaves públicas

En general, hay dos formas de algoritmos basados en llaves: el simétrico y el de llaves públicas. Los *algoritmos simétricos* son aquellos donde la llave de encriptado puede calcularse de la llave de decriptado y viceversa. En muchos de estos sistemas, la llave de encriptado y decriptado son la misma. Estos algoritmos, también llamados *algoritmos de llave secreta* o *algoritmos de una sola llave* requieren que tanto el receptor como el emisor coincidan en la llave antes de intercambiar mensajes. Esta llave debe mantenerse en secreto. La seguridad de un algoritmo secreto reside en la llave, divulgarla significa que todo mundo pueda decriptar y encriptar mensajes en este criptosistema.

El encriptado y el decriptado con un algoritmo simétrico son denotados por:

$E_k(P) = C$

$D_k(C) = P$

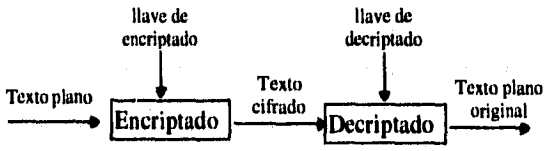


Fig. 4.4 Encriptado y Decriptado con dos llaves

Los algoritmos de llaves públicas son diferentes. Están diseñados de tal manera que la llave de encriptado es diferente de la de decriptado. Además, la llave de decriptado no puede ser (al menos en una cantidad de tiempo razonable) calculada de la llave de encriptado. Estos sistemas son llamados de llave pública porque la llave de encriptado puede hacerse pública: un extraño puede hacer uso de la llave pública para encriptar un mensaje, pero sólo con la correspondiente llave de decriptado se puede obtener el texto plano del mensaje. En estos sistemas la llave de encriptado frecuentemente es llamada *llave pública*, y la llave de decriptado *llave privada* o *llave secreta*.

Criptoanálisis

El principal propósito de la criptografía es mantener el texto plano, la llave o ambos, secretos a los interceptores (también llamados atacantes, adversarios, etc.). Criptoanálisis es la ciencia de recuperar el texto plano de un mensaje sin la llave. Un criptoanálisis exitoso puede recuperar el texto plano o la llave. También puede encontrar debilidades que tarde o temprano pudieran llevar a ese resultado.

Un intento de criptoanálisis es llamado *ataque*. Un ataque asume que el criptoanalista tiene los detalles del algoritmo criptográfico. Aunque este no es el caso en la vida real para el criptoanálisis, esta es una asunción convencional para el criptoanálisis académico. Y es una buena suposición ya que si nuestra seguridad depende de lo secreto del algoritmo, entonces sólo hay una mínima seguridad.

Hay seis tipos de ataques criptoanalíticos, listados en orden de poder. Cada uno de ellos asume que el criptoanalista tiene completo conocimiento del algoritmo de encriptado empleado:

Ataque sólo con texto cifrado. En este ataque, el criptoanalista tiene el texto cifrado de varios mensajes, todos los que han sido encriptados usando el mismo algoritmo de encripción. El trabajo del criptoanalista es recuperar el texto plano de tantos mensajes como sea posible, o mejor aún, deducir la llave (o llaves) empleadas para encriptar el mensaje con el propósito de decriptar o encriptar mensajes con las mismas llaves.

Ataque conociendo el texto plano. El criptoanalista no sólo tiene acceso al texto cifrado de varios mensajes sino también al texto plano de dichos mensajes. Su trabajo es

deducir la llave (o llaves) empleadas para encriptar los mensaje o un algoritmo para decriptar cualquier nuevo mensaje encriptado con la misma llave.

Ataque con texto plano seleccionado. Los criptoanalistas no sólo tienen acceso al texto cifrado y a su texto plano asociado para varios mensajes, sino que también pueden decidir cuál es el texto plano. Esto es más poderoso que un ataque conociendo el texto plano, porque el criptoanalista puede elegir bloques de texto plano específicos para encriptar, lo que puede dar más información acerca de la llave. Su trabajo es deducir la llave (o llaves) empleadas para encriptar los mensajes o un algoritmo para decriptar nuevos mensajes encriptados con la misma llave.

Ataque adaptativo con texto plano seleccionado. Este es un caso especial del ataque con texto plano seleccionado. El criptoanalista no sólo puede elegir el texto plano a encriptarse, sino que también puede modificar la elección basándose en los resultados de la encripción previa. En un ataque con texto plano elegido, el criptoanalista sólo podría ser capaz de elegir un bloque largo de texto plano a encriptarse; en un ataque adaptativo el puede elegir un bloque menor de texto plano y entonces elegir otro basado en los resultados del primero, y así consecutivamente.

Ataque seleccionando texto cifrado. Los criptoanalistas pueden elegir diferentes textos cifrados para decriptar y tener acceso al texto plano decriptado. En una instancia cuando los criptoanalistas tienen una prueba de intervención que automáticamente hace el decriptado, el trabajo es deducir la llave. Este ataque es aplicable primeramente a los criptosistemas de llave pública.

Ataque eligiendo la llave. No es propiamente un ataque cuando estamos dando la llave. Esto es extraño y obscuro.

Uno de los axiomas fundamentales de la criptografía es que el enemigo se encuentra en completa posesión de los detalles del algoritmo y solamente carece de la llave específica empleada para decriptar. Si otros no pueden romper un algoritmo aún con conocimiento de cómo trabaja, entonces ciertamente no podrán romperlo sin ese conocimiento.

Debemos tener cuidado de aquellas personas que presumen de las virtudes de sus algoritmos, pero que se niegan a hacerlos públicos.

Por otro lado, un buen algoritmo puede hacerse público sin preocupación. Podemos enviarlo a nuestros adversarios, publicarlo en una revista, o ponerlo a prueba. No hay problema si aún el diseñador del algoritmo no puede decriptar los mensajes sin la llave.

Seguridad de los Criptosistemas

Como hemos visto, todos los algoritmos sin excepción son teóricamente *rompibles*, con suficiente tiempo y suficientes recursos.

Algunos algoritmos son rompibles únicamente con el beneficio de más tiempo del que el universo ha existido y una computadora mayor que todo lo que hay en el universo. Estos algoritmos son teóricamente rompibles, pero no en la práctica. Un algoritmo que no es rompible en la práctica es *seguro*.

Un algoritmo es *incondicionalmente seguro* si, no importando cuanto texto cifrado tenga un criptoanalista, no existe suficiente información para recuperar el texto plano.

Un algoritmo es considerado *computacionalmente seguro*, o *fuerte*, si no puede ser roto con los recursos disponibles (actualmente o en un futuro lejano).

La cantidad de tiempo de cómputo y potencia requerida para recuperar la llave de encriptado es llamada el *factor de trabajo*, y es expresada como una magnitud. Si un algoritmo tiene un factor de trabajo de 2^{128}, entonces 2^{128} operaciones son requeridas para romper el algoritmo. Estas operaciones pueden ser muy complejas y consumir demasiado tiempo. El autor de "Applied Criptography"[xlvii] afirma:

"...Yo consideraría un algoritmo que toma un billón de veces la edad del universo para romperse, como computacionalmente seguro..."

Términos Históricos

Existen otros términos criptográficos. Un criptosistema también es llamado un *código* o un *cifrador*. Encriptar también es conocido como *codificar* o *cifrar*, y decriptar es también llamado *decodificar* o *descifrar*.

La palabra *cifrador* históricamente ha sido usada para referirse a criptosistemas en los que las letras son intercambiadas y substituidas por otras así como su orden alterado para ocultar el texto plano.

B) Criptografía clásica

Diferentes algoritmos criptográficos substituyen caracteres por otros o los trasponen con otros, actualmente trabajan con bits en vez de caracteres.

Cifradores de Substitución

Un cifrador de sustitución es aquél en el que cada caracter en el texto plano es sustituido por otro en el texto cifrado. Esta sustitución sirve para obscurecer el texto plano para todos excepto para el receptor, quien invierte la sustitución en el texto cifrado para recuperar el texto plano.

Un *cifrador por sustitución simple* es aquél en el que un caracter en el texto plano es reemplazado con un correspondiente caracter en el texto cifrado.

[xlvii] Bruce Schneier. *Applied Criptography: Protocols, Algorithms and Source Code in C*. Ed. Wiley, USA, 1993. p. 7

Un *cifrador por sustitución homofónica* es parecido al anterior excepto por qué un caracter de texto plano puede corresponder a varios caracteres de texto cifrado.

Un *cifrador de sustitución polialfabética* está hecho de múltiples cifradores por sustitución simple. Por ejemplo, podría haber cinco diferentes sustituciones empleadas.

El *cifrador de sustitución poligramática* es en cual bloques de caracteres son encriptados en grupos. Por ejemplo, "ABA" podría corresponder a "RTQ".

Estos cifradores pueden ser rotos fácilmente por que el cifrador no oculta las frecuencias de las diferentes letras del texto plano.

Los cifradores de sustitución homofónica[xlviii] son mucho más complicados de romper que los de sustitución simple, sin embargo tampoco ocultan las propiedades estadísticas del lenguaje en texto plano.

Cifradores de Transposición

Un cifrador de transposición es aquél en el cual los caracteres del texto plano permanecen igual pero su orden es alterado. En un cifrador de columnas simple, el texto plano es escrito horizontalmente y puede leerse verticalmente. La decripción sería la manera de escribir el texto cifrado verticalmente y leerlo horizontalmente.

C) Algoritmos computacionales

Existen muchos algoritmos criptográficos. Los tres más comunes son:

- DES (Data Encription Standard), actualmente es el más popular de los algoritmos de encriptado computacionales. DES es un estándar de encripción del gobierno de los Estados Unidos desarrollado en 1970. Es un algoritmo simétrico (la misma llave es empleada tanto para encripción como para decripción). En 1988[xlix] la Agencia de Seguridad de los Estados Unidos detuvo el uso de DES para necesidades de seguridad clasificada, por razones que aún no están claras.

- Lucifer, es un algoritmo de encriptado de una sola llave, como DES, perteneciente a IBM.

- RSA (llamado así por sus creadores: Rivest, Shamir y Adleman) es el algoritmo de llaves públicas más popular, y uno de los mejores. Puede ser empleado para encripción y para firmas digitales. RSA[l] es una herramienta de autenticación que verifica al originador y la integridad de los datos transmitidos.

[xlviii] Ibid 462

[xlix] Baker, Richard H. *Computer Security Handbook.* McGrawHill, Blue Ridge Summit, P.A 2nd ed. p.183.

[l] Ibid p.184.

- DSA (Digital Signature Algorithm, usado como parte del Digital Signature Standard) es otro algoritmo de llave pública. No puede ser usado para encripción pero sí para firmas digitales.

Pero si la encripción es tan buena,[11] ¿por qué no está todo mundo usándola?. El problema con la encripción radica en el intercambio seguro tanto como con el manejo de las llaves de encriptado. Esto es un dolor de cabeza por lo que fue inventado el enfoque de las llaves públicas para sobre llevar el problema del manejo de llaves empleando una llave para el encriptado y otra para el decriptado, esto permite al emisor encriptar un mensaje empleando la llave pública del receptor, que no necesita mantenerse en secreto, y de hecho es usualmente fácil de obtener. El mensaje sólo puede ser decodificado con la llave privada del receptor.

La criptografía de llaves públicas, también permite implementaciones simples de firmas digitales, no repudio y autenticación de mensajes. Estos pueden ser llevados a cabo junto con otros métodos criptográficos.

Sin embargo cualquier sistema de encriptado tiene dos principales debilidades: la primera es que sólo el más complejo de los códigos es irrompible (y tal vez ni siquiera ese) y la segunda es que la encripción puede hacer creer un falso sentido de seguridad. No debemos creer que el encriptado es la solución a todos nuestros problemas de seguridad.

Además, uno de los mayores obstáculos para la aceptación de la criptografía de llaves públicas es el hecho de que el RSA, uno de los más fuertes algoritmos conocidos, haya sido patentado en los Estados Unidos y por esto no puede se empleado en ese país sin permiso de la RSA Data Security of Redwood City, California. Por ejemplo, actualmente se ha popularizado en la Internet el uso de un sistema criptográfico desarrollado por Philip Zimmerman, el PGP (Pretty Good Privacy) y a pesar de su alta funcionalidad no puede ser empleado en los Estados Unidos debido a que es ilegal ya que usa el RSA sin licencia.

4.5. Soluciones Generales

Existen una serie de defensas que pueden emplearse como barrera para más de un problema, es decir, que pueden atacar a varios de los problemas citados. Se mencionan a continuación los más importantes, sin embargo existe una gran cantidad de software, además del mencionado, que pudiera resultar de utilidad dependiendo del problema que deseemos atacar, muchos de ellos se obtienen gratuitamente en la Internet. Si se desea más información acerca de otros programas de ayuda se sugiere consultar el archivo: FAQ Computer Security Frequently Asked Questions Article 1238 mantenido por Alec Muffet (aem@aber.ac.uk), que puede accesarse por ftp anónimo en comp.security.misc.

[11]Russell Kay *Distributed and Secure* Byte, June 1994 p. 173

4.5.1 Software de ayuda

4.5.1.1 COPS (Computer Oracle and Password System)

COPS[lii] en un verificador del estado de la seguridad de UNIX. Lo que esencialmente hace es verificar varios archivos y configuraciones del software para ver si han sido comprometidos (editados para plantar virus o puertas traseras) y verifica que estos archivos tengan los modos apropiados y los permisos establecidos para mantener la integridad del nivel de seguridad. COPS no detecta errores en el software que pudieran causar problemas de seguridad (como *tftpd* o el *sendmail*) y no corrige ningún error que encuentra.

Un programa como COPS[liii]también puede trabajar para el hacker, ya que puede puntualizar agujeros de seguridad en una manera automatizada. Muchos hackers tienen listas de agujeros de seguridad. Por lo que es mejor que nosotros corramos primero COPS que los intrusos y hagamos las correxiones necesarias.

A continuación presentamos algunos ejemplos de los principales problemas que COPS puede hacer notar (una lista más explícita se encuentra en el archivo *warnings* que se incluye con el paquete via ftp anónimo a la dirección: *cert.org: /pub/tools/cops*).

Principales avisos de problemas:

1) archivo_x is World_Writable!

archivo_x is group readable!

Esto significa que el *archivo_x* puede ser escrito por todos. Cualquiera puede modificar o borrar este archivo. Esto puede ser especialmente dañino si el archivo puede (aún indirectamente) dar acceso de root, como el archivo de contraseñas: /etc/passwd.

Para arreglar este problema ejecutar:

chmod a-w archivo_x

Esto elimina el acceso para el grupo "all-world".

2) archivo_x (in cron_file) is World writable!

File archivo_x (inside root executed file archivo_x2) is World writable!

File archivo_x(in /etc/rc) is World_writable!

[lii] *FAQ Computer Security Frequently Asked Questions*. Article 1238 of comp.security.misc. Maintained by Alec Muffet (aem@aber.ac.uk)

[liii] Farmer, Dan and Spafford, Eugene H. *The COPS security checker system*. In USENIX Conference Proceedings. Pags. 165-170. Anaheim, CA, Verano 1990

Es similar a los mensajes del aviso uno , pero potencialmente más serios. Archivos en este grupo son usados por root, ya sea que sean empleados como salida, entrada o para ejecución. Los archivos para ejecutarse son los más peligrosos porque si son alterados, el nuevo archivo se ejecuta con permisos de root. Los archivos de entrada son los siguientes, por que cambiandolos pueden alterar lo que el programa realiza y causar efectos laterales indeseados.

Para solucionar el programa se pueden seguir dos caminos: uno, borrar la referencia al archivo_x dentro del cron/rc/archivo_x2/otro_archivo, o ejecutar:

chmod a-w archivo_x

para remover el permiso de escritura para el grupo "all/world".

4.5.1.2 NPASSWD[iv]

Este comando desarrollado por Clyde Hoover en la Universidad de Texas fue creado como un remplazo del comando estándar de UNIX *passwd* así como el *ypasswd* de Sun. *npasswd* hace las contraseñas más seguras no permitiendo a los usuarios seleccionar contraseñas inseguras. Las siguientes capacidades son provistas en *npasswd*.

- Configuración de un largo de contraseña mínimo.
- Se puede configurar para forzar a los usuarios a emplear mayúsculas con minúsculas o dígitos y signos de puntuación.
- Verificación de contraseñas "simples" como el repetir una sola letra.
- Verificación contra información del host (como el nombre).
- Verificación contra clave de entrada, nombres, apellidos, etcétera.
- Verificación contra palabras en varios diccionarios, incluyendo el diccionario del sistema.

La distribución del *npasswd* está disponible mediante FTP anónimo en *emx.utexas.edu* en el directorio */pub/npasswd*.

4.5.1.3 TCP wrapper

Este paquete es útil para monitorear y restringir peticiones de servicios de red tales como: SYSTAT, FINGER, FTP, TELNET, RLOGIN, RSH, EXEC, FTP, TALK y otros.

Como sabemos, cada aplicación de los protocolos TCP/IP está basada en un modelo cliente-servidor. Por ejemplo, cuando un usuario invoca el comando telnet para conectarse a uno de nuestros sistemas, un proceso servidor telnet se ejecuta en el host

[iv]Ibid p. 37,38

destino. El proceso servidor telnet conecta al usuario a un proceso login. Algunos ejemplos de programas cliente-servidor se muestran en la siguiente tabla:

Cliente	Servidor	Aplicación
telnet	telnetd	login remoto
ftp	ftpd	transferencia de archivos
finger	fingerd	muestra usuarios

El camino normal es correr un sólo proceso que espera por cualquier clase de conexiones que vengan de la red. Cuando se establece una conexión este proceso o *deamon* (usualmente llamado *inetd*) corre el programa servidor apropiado y vuelve a dormirse, esperando por otras conexiones.

Los programas protectores (wrappers) se basan en un mecanismo simple pero poderoso. En vez de correr el servidor deseado, el inetd es alterado de tal manera para que corra un programa protector. El protector obtiene el nombre del host remoto o su dirección y desempeña algunas verificaciones adicionales. Cuando todo está bien, el protector ejecuta el servidor solicitado y termina.

Los programas protectores no tienen interacción con el usuario remoto (o proceso cliente). Esto tiene dos ventajas mayores: 1)los protectores son aplicaciones independientes, de manera que el mismo programa puede protejer muchos tipos de servicios de red; 2)el que no haya interacción significa que los protectores son invisibles para el exterior (al menos para los usuarios no autorizados).

Otra propiedad importante es que los programas protectores están activos sólo mientras se establece el contacto inicial entre el cliente y el servidor. Una vez que el protector ha hecho su trabajo no existe sobre carga en la comunicación cliente-servidor.

Existen dos formas de instalar el TCP-wrapper:

1) La manera fácil: mover los procesos de red a algún otro directorio y llenar los agujeros resultantes con copias de los programas protectores. Este camino no implica cambios a la configuración de los archivos del sistema, por lo tanto existe muy poco riesgo de corromper o disturbar algo.

2) La forma avanzada: dejar los procesos donde están y modificar la configuración del archivo inetd. Por ejemplo, una entrada como la que sigue:

tftp dgram udp wait root /usr/etc/tcpd in.tftpd -s /tftpboot

Cuando llega una petición, inetd correrá el programa protector (tcpd) con un proceso llamado 'in.tftpd'. Este es el nombre que el protector empleará cuando se ejecute la petición y cuando se recorran las tablas de control de acceso.

En el siguiente capítulo se describirá la manera fácil de instalar, especificando los pasos que se realizaron en las estaciones de trabajo hpux 9.0 de la Unidad de Servicios de Cómputo Académico de la Facultad de Ingeniería de la UNAM.

4.5.1.4 Kerberos[iv] [vi]

Kerberos es un sistema que emplea boletos electrónicos para auntentificar un usuario a un servidor. Un boleto, que es bueno sólo para un servidor único y para un usuario único durante un cierto periodo de tiempo, es un mensaje encriptado que contiene el nombre del usuario y del servidor, la dirección de red del usuario, una marca de tiempo, y una clave de sesión. En cuando el usuario toma el boleto, puede usarlo para accesar al servidor cuántas veces lo desee hasta que el boleto expire. El usuario no puede decriptar el boleto sólo lo puede presentar al servidor. Nadie que se encuentre escuchando en la red puede leer o modificar al boleto mientras pasa a través de la red sin ser detectado o invalidado.

El protocolo Kerberos abarca dos servidores, el Servidor de Autentificación Kerberos y uno o más TGSs (Ticket Granting Servers). Los pasos relacionados en el proceso Kerberos son los siguientes:

1.Obtener un boleto a un servidor destino en particular, el usuario primeramente solicita al Servidor de Autenticación Kerberos un boleto para el TGS Kerberos. Esta respuesta toma la forma de un mensaje que contiene el nombre del usuario y el nombre de su TGS (puede haber varios).

2.El servidor de Autenticación busca al usuario en su base de datos y genera una clave de sesión para ser usada entre el usuario y el TGS. Kerberos encripta la clave de sesión usando la llave privada del usuario (basada de alguna manera en la contraseña del usuario). Crea entonces un TGT (ticket-granting ticket) para que el usuario lo presente al TGS y encripta el TGT usando la llave privada del TGS (que sólo es conocida por el Servidor de Autenticación y el TGS). El servidor de autenticación envía ambos mensajes encriptados hacia el usuario.

3.El usuario decripta el primer mensaje y recupera la clave de sesión. En seguida, el usuario crea un autenticador consistente de su nombre, dirección y marca de tiempo, todo encriptado con la llave de sesión recién generada por el servidor de autenticación de Kerberos.

El usuario envía una solicitud al TGS de un boleto para un servidor en particular. Esta solicitud contiene el nombre del servidor, el TGT recibido de Kerberos (que ya está encriptada con la llave primaria del TGS) y el autenticador encriptado.

[iv] Kay, Russel. *Distributed and Secure.* Byte, June 1994. p. 172,173

[vi] Bruce Schneier. *Applied Criptography: Protocols, Algorithms an Source Code in C.* Ed. Wiley, USA. 1993

4. El TGS decripta el TGT con su llave primaria y usa la llave de sesión incluida en el TGT para decriptar el autenticador. Compara la información en el autenticador con la información en el boleto, la dirección de red del usuario con la dirección de la cual viene la solicitud, y la marca de tiempo con el tiempo actual. Si todo coincide permite que se procese la solicitud.

El TGS crea una nueva clave de sesión para el usuario y el servidor destino e incorpora esta llave a un boleto válido para que el usuario lo presente al servidor. Este boleto también contiene el nombre del usuario, la dirección de red, una clave de tiempo y un tiempo de expiración del boleto -todo encriptado con la llave privada del servidor destino- y el nombre del servidor. El TGS también encripta la nueva llave de sesión del usuario usando la llave de sesión compartida entre el usuario y el TGS. Envía ambos mensajes al usuario.

5.El usuario decripta el mensaje y extrae la clave de sesión para usarse con el servidor destino. Finalmente, el usuario se encuentra listo para autenticarse a sí mismo con el servidor. El usuario crea un nuevo autenticador encriptado con la llave de sesión destino que el TGS generó. Para solicitar acceso al servidor destino, el usuario envía el ticket recibido de Kerberos (que se encuentra encriptado con la llave primaria del servidor destino) y el autenticador encriptado.

Ya que el autenticador contiene texto plano encriptado con la llave de sesión prueba que el originador conoce la llave. Cabe señalar la importancia de encriptar el tiempo y la fecha, ya que esto previene que un interceptor que almacene tanto al boleto como al autenticador los reenvíe después.

6. El servidor destino decripta y verifica el boleto y el autenticador, también verifica la dirección del usuario y la marca de tiempo. Si todo cuadra, el servidor reconoce que el usuario es quien dice ser, y ambos comparten la llave de encriptado que pueden emplear para una comunicación segura. (Ya que sólo el usuario y el servidor comparten esta llave, pueden asumir que un mensaje encriptado con esa llave fue originado en la contraparte).

7.Para aquellas aplicaciones que requieren mutua autenticación, el servidor envía un mensaje al usuario consistente de la clave de tiempo más uno, encriptado con la clave de sesión. Esto le sirve de prueba al usuario de que el servidor realmente conoce su llave privada y fué capaz de decriptar el boleto y el autenticador.

De manera esquemática podemos resumir el funcionamiento de Kerberos:

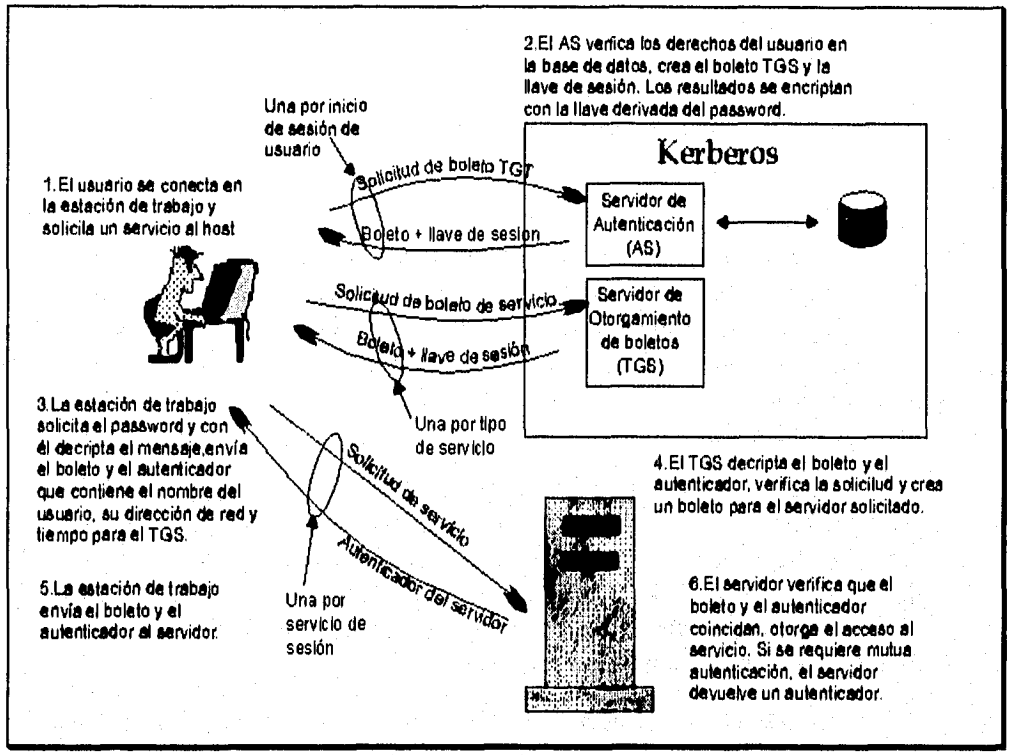


Fig. 4.5 Kerberos.

4.5.1.5 Passwd+

Ya se ha mencionado que la selección de una contraseña segura es una tarea muy importante para la seguridad, siendo esta la principal entrada a nuestro sistema y el principal agujero a cuidar. Hacer que por convicción un usuario seleccione una contraseña no pasa más que de ser una bonita idea, pero la realidad es que si queremos que los usuarios elijan adecuadamente su contraseña, tenemos que obligarlos a ello. Para este objetivo podemos realizar programas que lo controlen estableciendo nuestras normas, o bien seleccionar una de las muchas herramientas que existen de manera gratuita o mediante algún costo, que vienen configuradas con las mínimas normas que debe cumplir un contraseña más o menos segura. No nos garantizan que el usuario no va a escribir su contraseña junto al teclado, pero por lo menos le hacen la vida más difícil a aquél que intente adivinar la contraseña de uno de nuestros usuarios. La tarea del administrador será, además de configurarlo conforme a sus necesidades, establecer el tiempo mínimo en el cuál un usuario debe cambiar su clave de acceso.

Una de las herramientas más importantes que nos ayudan a este propósito es *Passwd+*. Este es un conjunto de programas en C que sustituyen al *passwd* normal de UNIX, pero con la diferencia de que sólo permitirá hacer el cambio de contraseña al

usuario si la nueva contraseña reune ciertas características de seguridad que el programa prueba y que además pueden ser configuradas por el administrador.

Debe tenerse cuidado al instalar este producto ya que debe hacerse por el super usuario y debe correr con permisos de *root*.

Esta utilería fue desarrollada por Matt Bishop y está disponible mediante ftp anónimo a *dartmouth.edu*, y el archivo se encuentra en *pub/passwd+.tar.Z*. Se encuentra comprimida por lo que debe transferirse de modo binario. Es un programa beta por lo que debe de acondicionarse al sistema en que se instale.

Las principales pruebas que hace passwd+ a la contraseña propuesta y, que en caso de coincidir, causan que sea rechazada son:

- Iniciales del usuario
- Primer nombre del usuario
- Apellido del usuario
- Teléfono del usuario
- Clave de acceso al sistema
- Alguna de las anteriores invertida
- Sólo minúsculas (sin dígitos o caracteres especiales)
- Sólo mayúsculas (sin dígitos o caracteres especiales)
- Sólo números
- Menor a la longitud mínima establecida

Para mayor información se sugiere accesar dicho conjunto de programas o mediante correo electrónico en la Internet al autor en:

Matt.Bishop@dartmouth.edu

4.5.2 Firewalls (paredes de fuego)

4.5.2.1 ¿Qué es un firewall?

La primer pregunta que necesitamos contestar es ¿qué es un firewall?, una respuesta sencilla la enuncia John Bryan[lvii]:

"Una pared de fuego es una barrera colocada entre nuestra red y el mundo externo para prevenir intromisiones no deseadas y potencialmente dañinas a nuestra red.

[lvii]Bryan, John. *Build a Firewall.* Byte April 1995, p. 91

Así como una pared contra el fuego no es una protección perfecta, un firewall no puede hacer 100 por ciento segura a nuestra red contra intrusos. Pero puede estar remarcablemente cerca."

Y una definición más formal la describe William R. Cheswick[lviii]:

"*Un firewall, en general, consiste de varios componentes:*

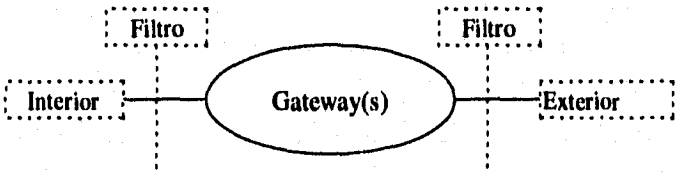


Fig. 4.6 Componentes de un Firewall.

"Los filtros (a veces llamados pantallas -screens-) bloquean la transmisión de ciertas clases de tráfico. Un gateway es una máquina o un conjunto de máquinas que proveen servicios confiables par compensar los efectos del filtro. La red habitada por el gateway frecuentemente es denominada zona desmilitarizada (DMZ). Un gateway en la DMZ es asistido en ocasiones por un gateway interno. Típicamente, dos gateways tendrán una comunicación más abierta a través del filtro interno que las que tiene el gateway externo con los otros hosts internos. Ya sea el filtro, o el gateway en sí mismo, puede ser omitido, los detalles varían de firewall en firewall. En general el filtro externo puede ser usado para proteger al gateway de ataques, mientras que el filtro interno es empleado para salvaguardar contra consecuencias de un gateway comprometido. Uno o ambos filtros pueden proteger la red interna de asaltos. Un gateway expuesto es denominado host bastión".

El concepto de firewall puede ser tan complejo como la organización que lo implante lo sea.

Esa barrera colocada entre nosotros y el mundo externo debe poseer las siguientes propiedades:

- Todo tráfico de afuera hacia adentro y viceversa debe pasar por el firewall.
- Sólo tráfico autorizado, establecido previamente en las políticas de la organización (en general, incluye las políticas de seguridad en sí mismas), debe pasar por el firewall.
- El firewall mismo debe ser invulnerable.

Es importante reflexionar sobre la última propiedad: *el firewall por sí mismo debe ser invulnerable*. De acuerdo a esta propiedad, es importante delimitar contra qué nos

[lviii] Cheswick, William R. and Bellovin, Steven M. *Firewalls and Internet Security. Repelling de Willy Hacker.* Addison-Wesley Professional Computing Series. USA, 1994

puede proteger y contra qué no, para poder diseñarlo e implantarlo, de forma tal que cumpla de la manera más cercana, con esta propiedad.

4.5.2.2 CONTRA QUE NO NOS PROTEGE.

Es extremadamente difícil cuantificar los daños que podrían resultar de la destrucción de un firewall. Una forma de medir qué tan resistente es un firewall a las amenazas de ataques, es recopilar información de los ataques mismos. La peor cosa que puede pasar para un firewall que ha sido comprometido o destruido es no saber cómo sucedió esto (es prudente recordar que la necesidad de investigar, documentar e implementar esquemas de seguridad en la redUNAM, fue el resultado de un ataque a la red, y que es el caso, precisamente descrito: no se sabía como sucedió este ataque). Y lo peor que puede pasar para un firewall, es detectar e informar al administrador que está sucediendo un ataque, pero que el ataque tal vez no sea totalmente exitoso.

Una forma de ver los resultados de un firewall que ha sido comprometido, es ver estos resultados como "zonas de riesgo". En el caso de una red directamente conectada a la Internet sin un firewall, toda la red está expuesta a un ataque. Esto no implica que la red sea vulnerable a todo tipo de ataques, pero en una situación donde la red es alcanzable a redes no confiables, es necesario tener la certeza de la seguridad de cada uno de los servidores de esta red. La experiencia sobre este tipo de situaciones nos deja ver claramente que esto es muy difícil, ya que existen herramientas adecuadas vía configuración, que permiten a los usuarios explotarlas para fines no lícitos desde el punto de vista de la seguridad de los sistemas - los problemas de seguridad del protocolo TCP/IP, en las aplicaciones y en los sistemas operativos de la red. En el caso de los firewalls, la zona de riesgo es a menudo reducida al firewall mismo, o a una subred seleccionada, si es el caso, reduce significativamente la zona de posibles ataques, desde el punto de vista del administrador del sistema.

Si un firewall es comprometido, la zona de riesgo se expande a toda la red que se pretende proteger. Si un posible atacante, accesa el firewall, todavía existe la esperanza de que deje rastros de su intromisión, y pueda ser detectado a tiempo. En contra parte, si un firewall es completamente destruido, la red protegida queda a partir de este momento expuesta, ahora sí, a todo tipo de ataques desde el mundo exterior, y es prácticamente imposible reconstruir el curso del ataque.

Un firewall no puede protegernos contra tráfico que no pasa por él. Para citar un ejemplo de esto, es tan simple como que en una red, en la cual se ha diseñado e implantado un sofisticado firewall vía el equipo de comunicaciones, la información sea plagiada por medio de un disco o dispositivo magnético.

No puede protegernos contra virus. Existe toda una diversidad de formas de codificar archivos para ser transferidos por la red, y por si fuera poco existen muchas arquitecturas de virus como para poder contemplarlas dentro de tráfico no deseado. En otras palabras, los usuarios son responsables del tipo de información que envían, el

firewall de que llegue al destino indicado, sin que sea alterada, perdida o accedida por alguien no autorizado en el transcurso del viaje por la red.

4.5.2.3 CONTRA QUE NOS PROTEGE.

Partiendo de un análisis del estado actual de la red en la que se desea implantar un firewall, se establecen las políticas restrictivas del tráfico de la red, y de esta forma se bloquean los servicios que no deben ser permitidos. Es decir, el firewall es responsable de lo que pasa a través de él.

En general, los firewalls pueden ser vistos en términos de reducir las zonas de riesgos a un simple punto de falla. Un firewall mejora la seguridad en un host por reducir ataques a través de un estrecho hueco donde existe la posibilidad de detectarlos antes de que éstos sean exitosos. Otra importante característica es que ofrece una bitácora de los accesos mediante los cuales podemo detectar o rastrear ataques.

4.5.2.4 Elección de un firewall

Cuando se ha decidido que una de las opciones para establecer un esquema de seguridad, es la implantación de un firewall, se deben tomas varias decisiones.

La primera y más importante es establecer la política de cómo la compañía u organización operará el sistema, específicamente, se debe considerar que la decisión debe ser tomada bajo la prioridad de que la seguridad es más importante que el uso fácil del sistema, o viceversa. Existen dos aproximaciones que resumen este conflicto:

- Todo aquello que no es expresamente permitido, es prohibido.
- Todo aquello que no es expresamente prohibido, es permitido.

La importancia de esta distinción no debe ser pasada por alto. En el primer caso el firewall debe ser designado a bloquear todo, y los servicios deben ser habilitados sobre la base caso-por-caso solamente después de una valoración cuidadosa de las necesidades y los riesgos. Esta toma de decisión tiene repercusiones directas a los usuarios, quienes pueden ver al firewall como un obstáculo.

En el segundo caso, los administradores del sistema son puestos en modo reactivo, prediciendo que tipo de acciones podría tomar el usuario para debilitar la seguridad del firewall, y preparar defensas contra ello. Un usuario puede comprometer la seguridad de sus cuentas si no están consientes de las precauciones razonables de seguridad. Si un usuario tiene abierta una cuenta en el sistema del firewall mismo, puede resultar un agujero de seguridad. La presencia de las cuentas de usuarios en el sistema del firewall tiende a agrandar el problema de la integridad del sistema. Una segunda afirmación de las políticas está implícita en la postura de " Aquello que no está expresamente prohibido esta permitido". Esta postura podría implicar peligro desde que acepta que el administrador es ignorante de que puertos TCP son seguros, o que agujeros de seguridad existen en el software de aplicación. Además hay que tomar en cuenta que

los fabricantes de software son lentos en informar sobre los agujeros de seguridad en sus productos. Esto es una admisión del hecho de que no se conoce como pueden atacar un sistema.

4.5.2.5 Arquitecturas de firewall.

Podemos construir firewalls[lix] de diferentes maneras, usando una amplia variedad de mecanismos. Los siguientes son los más comunes:

- Enrutador Selectivo (Filtros basados en ruteadores)
- Una computadora (host) como Gateway, o "bastión"
- Una red aislada, separada (Dual-Homed Gateway)

A continuación describiremos cada uno de ellos, así como algunos otros que aunque están menos difundidos, no dejan de ser importantes.

A) Enrutador Selectivo

Tal vez, la forma más simple de crear un firewall se basa en el uso de un ruteador programable. Los ruteadores trabajan controlando tráfico a nivel IP, pasan o bloquean de manera selectiva los paquetes de datos basándose en la dirección origen/destino o la información de puerto en el encabezado del paquete.

Cuando menos, podemos usar un ruteador como filtrador de paquetes. Este es tal vez el mecanismo de seguridad más empleado actualmente. Mientras que firewalls razonablemente buenos pueden ser creados únicamente con ruteadores, resulta algo difícil programar al ruteador para dejar afuera absolutamente todo lo que se desea. Desafortunadamente la mayoría de los ruteadores vienen configurados con un mínimo de protección, y muchas organizaciones simplemente los instalan de esta manera.

El problema con los sistemas basados en el enfoque de ruteadores se origina en la variedad de protocolos empleados en la Internet. Por lo menos tres de los más importantes servicios de la red no son manejados eficientemente por los filtros de paquetes: FTP, DNS y X11.

B) Host Bastión

Una alternativa para construir un firewall es usar una computadora en vez de un ruteador. Esto ofrece mayores capacidades, incluyendo la habilidad de registrar todas las actividades sobre el gateway. Realmente, cuanto se piensa en un firewall de red, primeramente se piensa en un sistema de cómputo separado altamente seguro, que permanezca en guardia por nuestras redes.

[lix] Bryan, John . *Build a Firewall*. Byte April 1995, p. 91-95

Este sistema centinela, algunas veces llamado host bastión (fortín, baluarte), es un punto de defensa crítico que debe ser diseñado cuidadosamente, altamente controlado y auditado regularmente.

Mientras que un firewall basado en ruteadores monitorea los paquetes de datos al nivel IP, los hosts extienden su control a nivel aplicación, donde el tráfico puede ser examinado más profundamente. Sin embargo, no podemos usar cualesquiera aplicaciones; necesitamos conocer que el software de la aplicación que corremos (e inclusive el sistema operativo) puede tener sus propios agujeros de seguridad.

Para atacar estos problemas y tratar con los errores potenciales de los protocolos, los firewalls basados en hosts deben usar software especializado. En esencia son versiones alteradas de los programas originales, que no incluyen toda la funcionalidad de la versión original, y verifican que los mensajes coincidan con las restricciones programadas.

Es la configuración más sencilla de implementar, la cual requiere un mínimo de hardware. Por medio de esta configuración se bloquea el acceso desde la Internet hacia la red privada. Ya que no existe tráfico fluyendo en forma directa entre la red privada y el mundo exterior, no es necesario dar a conocer las rutas dentro de la red protegida, por lo que ésta se vuelve "invisible" para los demás sistemas en la intered, excepto para el *Host bastión*. La mayoría de los servicios, como ftp, telnet, smtp. etc., son proporcionados vía administradores ejecutándose en el *Host bastión*, por lo que esta configuración sigue el paradigma de "lo que no está expresamente permitido, está prohibido".

Fig. 4.7 Host Bastión.

C) Una red aislada, separada

Otra manera de establecer un firewall es similar al de los sistemas basados en hosts, pero en vez de interponer una computadora, creamos otra red, una subred aislada que se encuentra entre la red interna y la externa. Típicamente, esta red es configurada de manera que tanto la Internet como la red interna pueden accesarla, pero el tráfico a través de la red aislada es bloqueado. Esto es, el único destino que puede alcanzar el host externo es el host interno (generalmente la red aislada está compuesta por un host). La máquina interna no confía en la externa. Ofrece unos cuantos servicios al externo, típicamente una conexión autenticada y un correo de gateway. La máquina externa ofrece otros servicios más comunes al mundo exterior como DNS, FTP, SMTP mail y acceso telnet mediante la conexión autenticada.

D) Host Gateway Filtro

Posiblemente la configuración de firewalls más comúnmente implementada: utiliza un Enrutador selectivo y un Host bastión. Usualmente el Host bastión sobre la red protegida, y el Enrutador selectivo está configurado de forma tal que el Host bastión es el único sistema alcanzable desde el exterior. A menudo el enrutador selectivo es configurado para bloquear tráfico hacia el Host bastión sobre un puerto específico, permitiendo a un número pequeño de servicios comunicarse con él.

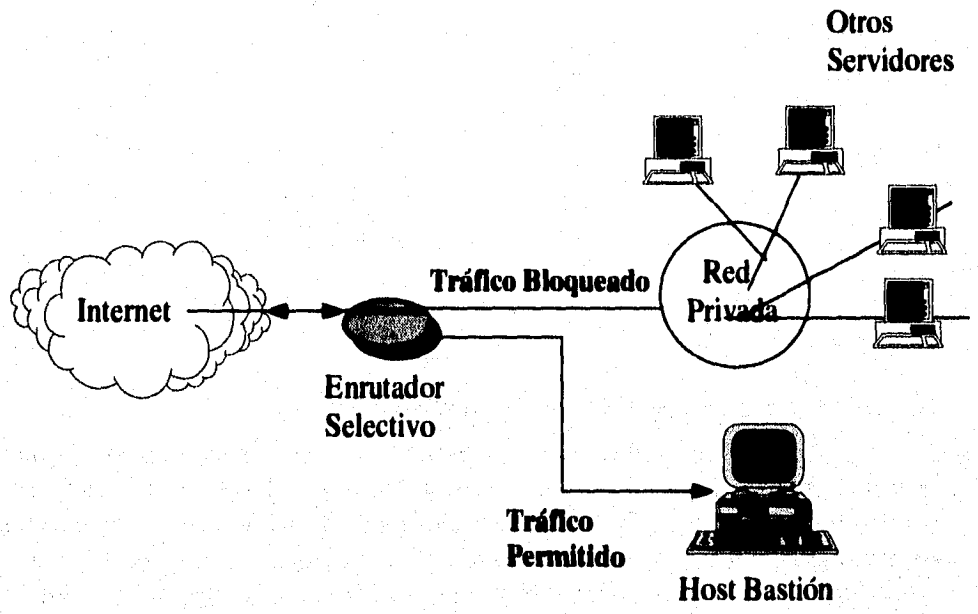


Fig. 4.8 Host Gateway Filtro

E) Subred Gateway Filtro

En este tipo de configuración, una subred aislada se crea, entre la red protegida y el mundo exterior. Típicamente esta subred es aislada utilizando un *Enrutador selectivo*, en el cual pueden ser implementados varios niveles de filtrado de paquetes. El acceso a la subred esta controlado mediante reglas de selección en enrutadores que restringen el tráfico, de tal forma que los servidores en la subred son los únicos puntos alcanzables desde ambos lados. Se comporta de manera similar al concepto de red aislada o separada sólo que aplicado a toda un subred. La diferencia radica en que esta configuración permite la existencia de servidores fuera de la zona protegida. Una ventaja de este tipo de configuración, es la posibilidad de configurar el enrutamiento, de tal forma que no se den a conocer las rutas hacia la red privada desde Internet y viceversa. De esta manera se incrementa el grado de seguridad de una red privada, pues difícilmente un usuario externo podrá direccionar tráfico hacia dicha red. Una vez que el enrutamiento es bloqueado, todo

tráfico debe pasar a través de una aplicación en el *Host Bastión*, de manera similar que en la red aislada.

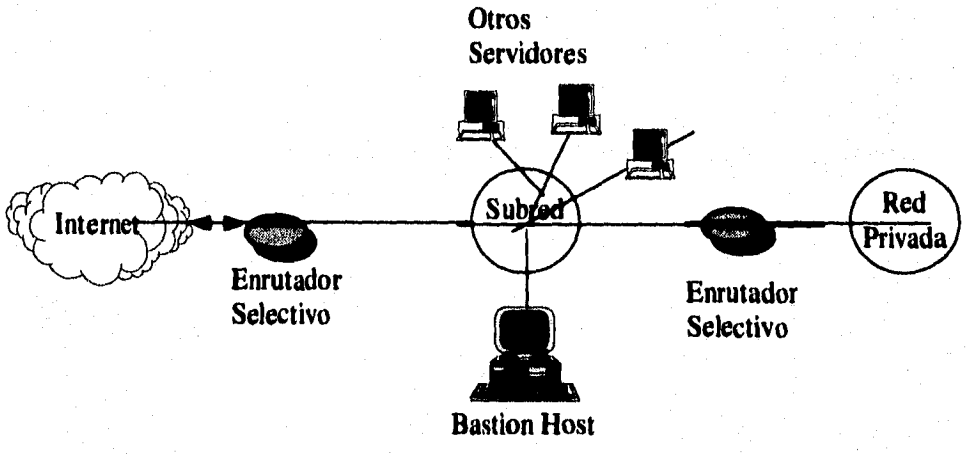


Fig. 4.9 Subred gateway filtro.

F) Gateway agente o de nivel aplicación -Proxy Gateway.

La mayoría del software trabaja en modo *almacena-y-reenvía*; los servidores de correo electrónico coleccionan las entradas, las analizan y las reenvían. Los gateways de nivel aplicación son reenviadores de servicios específicos, los cuales usualmente operan en modo usuario más que a nivel protocolo. El famoso agujero de seguridad causado por *sendmail*, explotado por el "Internet Worm" de Morris, es un ejemplo de los tipos de problemas de seguridad que se pueden presentar en algunos gateways de nivel aplicación. Otros gateways de nivel aplicación son interactivos como los gateways para FTP y TELNET que corren en los firewalls de DEC -Digital Equipment Corporation. En general el termino gateway de nivel aplicación será utilizado para describir algún tipo de servicio de reenvío que corre sobre un firewall y es un riesgo potencial en cuestiones de seguridad. Generalmente, las aplicaciones cruciales que se están corriendo en un gateway de nivel aplicación, están corriendo en algún tipo de Host bastión.

G) Gateways Híbridos

Los gateways híbridos son considerados como la categoría "algo más". Como ejemplo de este tipo de arquitecturas podemos mencionar algunos servidores conectados a Internet, pero accesibles únicamente a través de líneas seriales conectadas a un servidor de terminales Ethernet en una red privada. Estos gateways pueden tomar ventajas de múltiples protocolos o bien implementar un *túnel* de un protocolo sobre otro. El enrutador puede mantener y monitorear todo el tráfico TCP/IP o de alguna manera examinar tráfico para tratar de detectar y prevenir ataques. El firewall corporativo de AT&T es un Gateway Híbrido combinado con un Host bastión.

Otra opción es crear nuestro propio firewall si tenemos la experiencia y el tiempo requeridos. Existe una herramienta para crear firewalls llamada TIS Firewall Toolkit, creada por Trusted Information Systems y disponible gratuitamente en la Internet (puede buscarse por Web a el recurso http://www.tis.com).

Existe una gran variedad de firewalls en el mercado a diferentes costos y con distintas implementaciones de seguridad, por ejemplo el Sidewinder[lx] con un costo de $30,000 desarrollado por SCC (Security Computing Corp.), que respalda su producto con la experiencia de desarrollar sistemas clasificados para el gobierno de los Estados Unidos. Digital También ofrece su firewall llamado Digital's Firewall Service, que incluye consultoría, instalación del software y hardware, entrenamiento y soporte. También ofrece otras capacidades como la de encriptado.

4.5.2.6 Problemas de seguridad en los firewalls

Tomando en cuenta las arquitecturas descritas anteriormente, podemos describir exactamente la mayoría de las formas que pueden tomar los firewalls, y al mismo tiempo, podemos hacer algunas afirmaciones generales acerca de los problemas de seguridad que cada uno de las arquitecturas presenta. Asumiendo que un firewall cumple su propósito básico de ayudar a proteger la red, es importante examinar cada tipo de firewall con respecto a:

Control de Daños. Si el firewall está comprometido, ¿A qué tipo de amenazas deja abierta la red? Si se destruye, ¿A qué tipo de amenazas deja abierta la red?

Zonas de Riesgo. ¿Qué tan grande es la zona de riesgo en operación normal? Una forma de medir esta zona de riesgo es el número de hosts o enrutadores que pueden ser alcanzados desde fuera de la red protegida.

Modo de Falla. Si el firewall es transgredido o destruido, ¿qué tan sencillo es de detectar? Una vez destruido el firewall, ¿qué tanta información mantiene para poder diagnosticar el ataque ?

Facilidad de Uso. ¿Qué tanto altera la red a nivel administración y configuración el implantar un firewall?

Postura. ¿Cuál es la filosofía básica del firewall: "aquello que no está expresamente permitido, está prohibido", o bien, "aquello que no está expresamente prohibido, está permitido"?

A) Firewalls que utilizan Enrutador selectivos.

Algunas redes se protegen solamente por medio de un Enrutador selectivo entre la red privada y el mundo exterior. Este tipo de firewall se diferencia de un *Host Gateway*

[lx] Bryan, John . *Firewalls for Sale. A look at five diferent firewalls products and services you can install today.* Byte, April 1995 p. 99-100

Filtro en que usualmente existe comunicación directamente entre múltiples hosts sobre la red privada, y múltiples hosts sobre Internet. La zona de riesgo es igual al número de hosts de la red privada, y el número de servicios a los cuales el Enrutador selectivo permite el paso. Para cada servicio punto a punto que es permitido, la zona de riesgo aumenta bruscamente, por lo cual es prácticamente imposible de cuantificar. El control de daños es también difícil, por que el administrador tendría que examinar regularmente cada host para monitorear las posibles interrupciones en cada uno de ellos.

En el caso de una destrucción total del firewall, puede ser muy difícil de rastrear o de descubrir la causa. Si un enrutador comercial (el cual no mantiene registros de los accesos a él) es usado, y la contraseña del administrador es comprometida, toda la red privada puede ser dejada al descubierto. Son conocidos casos en que enrutadores comerciales son configurados con reglas no apropiadas para su tipo de servicio.

Los firewalls implementados por medio de la arquitectura de *Enrutador selectivo* no son la solución más segura, pero son muy populares debido a que permiten el acceso libre a la Internet desde algún punto de la red privada.

B) Red aislada -Dual-Homed Gateways

La más usada arquitectura de firewalls y la más fácil de implementar es ésta. Como no reenvía tráfico TCP/IP, prácticamente es una barrera entre la red privada y el mundo exterior. Su fácil uso es determinado por cómo los manejadores del sistema eligen el acceso, o por proveer aplicación gateway tales como reenvío de sesiones TELNET o por permitir acceso por medio del comando *login* a los usuarios en el mismo gateway. De acuerdo a esta descripción, la filosofía bajo la cual trabaja es "Todo lo que no está expresamente permitido es prohibido", de esta forma los usuarios sólo pueden acceder a servicios Internet para los cuales se ha definido una *aplicación gateway*. Si dentro de estas aplicaciones gateway está permitido el acceso a los usuarios vía el comando login, la seguridad del firewall esta seriamente debilitada. Durante la operación normal del firewall la zona de riesgo se reduce al mismo gateway, debido a que es el único alcanzable desde el mundo exterior. Si existe una cuenta para un usuario en el gateway, y su contraseña está considerada como débil, o su cuenta ya ha sido comprometida, la zona de riesgo crece a toda la red privada. Desde el punto de vista del control de daños, la información del acceso del intruso, puede servir al administrador para rastrear de qué forma entró al sistema (por medio de una cuenta comprometida, o una contraseña débil), pero esto es difícil de realizar. Si un Dual-homed Gateway es configurado sin acceso directo a los usuarios, el control de daños es relativamente sencillo de cuantificar, esto es, cualquier acceso se puede considerar generalmente un evento seguro. Este tipo de configuración tiene una ventaja sobre el Enrutador selectivo desde el punto de vista de que su software es fácil de adaptar a los sistemas de *log* (archivos de bitácora). Esto puede hacer más fácil la detección de la causa en caso de la destrucción del firewall, pero no quiere decir que la información pueda servir para detectar otro tipo de intrusiones en los hosts de la red privada.

Si un Dual-homed Gateway es objetivo de un *hacker*, éste tiene un amplio rango de opciones para consumar la intromisión. Si el intruso tiene una cuenta en la red local, los ataques pueden traslaparse a través de la red local desde la cual tiene acceso hacia la red privada o traspasar la frontera hacia una red privada vía un Dual-homed Gateway. Sistemas de archivos montados sobre NFS, con debilidades de seguridad en .rhosts, software de automatización, programas de respaldo de la red, todos proveen una forma de control sobre la red interna. Este control es seguro, por lo que provee una base sobre la cual retiene al intruso fuera del gateway mismo. El aspecto más débil de un Dual-homed Gateway es su modo de falla. Si un firewall es destruido es posible que el atacante logre rehabilitar las funciones de enrutamiento, y programarlas de tal forma que deje acceso libre a toda la red privada. En un Dual-homed Gateway basado en un servidor UNIX, el enrutamiento de paquetes TCP/IP es comúnmente deshabilitado modificando una variable del *kernel* denominada *ipforwarding*, si un intruso modifica los permisos del sistema de archivos, puede modificar esta variable. Esto parece muy difícil de realizar, pero un posible atacante con la información del sistema operativo y experiencia como administrador y programador puede realizarlo sin mayor problema.

C) Host Gateway Filtro

Generalmente un Host Gateway Filtro es muy seguro. Típicamente un Host bastión es configurado sobre la red privada, con un Enrutador selectivo entre la red privada y el mundo exterior, el cual sólo permite accesos desde el mundo exterior hacia el Host bastión. Como el Host bastión pertenece a la red privada la comunicación de la red local hacia afuera es transparente para los usuarios de esta red, eliminando los problemas por configuración de enrutamiento complicados. Si la red privada es (como en muchos casos) una red local virtual extendida (no subredes o enrutamiento) el Host Gateway Filtro trabajará sin requerir cambios a la red local, tan grande como la red local es usando un conjunto de direcciones legítimamente asignadas. La zona de riesgo de un Host Gateway Filtro se restringe al Host bastión y al Enrutador selectivo, y la postura de la seguridad es determinada por el software que está corriendo en el sistema. Tenemos el mismo caso respecto a los accesos vía el comando login descrito anteriormente en la red aislada: si el atacante logra entrar vía el comando login al Host bastión, la zona de riesgo se extiende a toda la red privada. El Host Gateway Filtro y la red aislada son muy similares tanto en características como en consideraciones de diseño con respecto al software que está corriendo en el Host Bastión.

D) Subredes Filtro

Una Subred Filtro es usualmente configurada con un Host bastión como el único punto de acceso a la subred. La zona de riesgo se restringe al Host Bastion y los Enrutadores selectivos que hacen la conexión entre la red privada, la subred y el mundo exterior. La fácil implementación de un Subred Filtro varía de acuerdo al número de servidores de la subred y el número de Enrutadores selectivos involucrados, pero en general esta configuración es la más usada de los filtros. Sobre una Subred Filtro se forza (los diseñadores y administradores del sistema deben asegurarse de que esto suceda) a

que todos los servicios que pasan a través del firewall sean provistos por gateways de aplicación y diseñados bajo la postura "Todo lo que no está expresamente permitido es prohibido".

Si una Subred Filtro con el enrutamiento entre las redes (el mundo exterior, la red privada) bloqueado es atacado, con el intento de destruirlo, el atacante tendría que reconfigurar el enrutamiento entre las tres redes(el mundo exterior, la subred y la red privada), sin que sea descubierto por los administradores del sistema, y sin que los cambios en las tablas de enrutamiento sean descubiertos. No existe duda de que esto puede suceder, pero es difícil deshabilitar los accesos de la red a los Enrutador selectivos, ya que para poderlo hacer, el atacante necesita entrar al Host Bastión, posteriormente de ahí pasar a un host de la subred, y a través de los servicios de un *gateway agente* de la subred, pasar a un host de la red privada.

Una ventaja de las Subredes Filtro, con el tráfico entre la red privada y el mundo exterior bloqueado, por ejemplo a la Internet, es la facilidad de conectarse sin necesidad de cambiar o reestructurar su direccionamiento interno, y ocultar la topología de su red al mundo exterior. Esto es muy útil, desde el punto de vista de administradores y diseñadores de la red. Si el direccionamiento de la red privada, no es apropiado para adaptarse al direccionamiento que se le asigne al conectarse a la Internet, puede conectarse por medio de esta arquitectura, y si es necesario el cambio de direccionamiento o de implementar subredes, lo puede hacer en forma paulatina una vez ya conectados.

La subred Filtro depende totalmente del conjunto de software que trabaja en ella, es muy similar a la red aislada y al Host Gateway Filtro en su funcionamiento, pero difieren en la complejidad de la programación de los Enrutador selectivos.

E) Gateways Híbridos.

Los Gateways Híbridos son mencionados como "algo más", por que pueden involucrar algunas de las combinaciones anteriormente mencionadas, modificadas o combinadas. Por ejemplo, un firewall que consiste de una "caja negra" entre la red privada y el mundo exterior, el cual enruta tráfico, pero también mantiene una bitácora del estado de las conexiones TCP, como el tipo de conexión, su origen y destino. Estas conexiones pueden ser filtradas con reglas como: "Permite la conexión del Host A de la red privada hacia una red B en la Internet vía un servicio TELNET desde las 9:00 hasta las 15:00 hrs. únicamente".

4.5.3 Monitoreo

Este es un punto muy importante a reafirmar. Ya que pueden ser muy buenas nuestras políticas de seguridad y los sistemas implementados pero es bien sabido que todo sistema es perfectible y susceptible de errores, por lo que es importante que seamos nosotros los primeros que los encontremos.

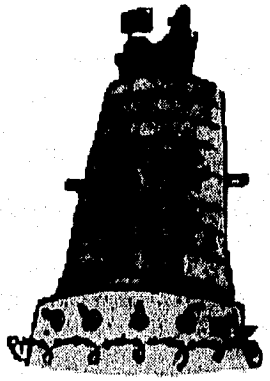
Debemos vigilar que las políticas se lleven a cabo como lo establecimos. Que no accese quién no debía y que no realice ciertas actividades quién no tenía permisos. Cualquier violación debe corregirse.

Conocer nuestro sistema es lo más importante, así como las herramientas que estamos empleando, y estar al tanto de las nuevas opciones que podrían mejorar nuestro sistema, en caso de que sean aplicables y necesarias.

Monitorear que nuestras políticas funcionen es tan importante como vigilar que tampoco sean una barrera para los desarrolladores y que el costo de tenerlas llegue a ser más alto que el de no tenerlas.

Cada implementación de seguridad debe ser monitoreada de manera distinta y es recomendable que para cada caso exista el personal especializado. Ningún área debe descuidarse.

CAPÍTULO 5



Modelo de Seguridad. Caso Práctico: Unidad de Servicios de Cómputo Académico de la Facultad de Ingeniería

5.1. Introducción

En el presente capítulo se propone un modelo de análisis de seguridad genérico, y con el propósito de dar una ilustración práctica de los problemas y soluciones señalados, se analizó la situación referente a la seguridad de la Unidad de Servicios de Cómputo de la Facultad de Ingeniería, antes conocida como CECAFI (Centro de Cálculo de la Facultad de Ingeniería), en dicho punto se detalla la implantación del modelo en la Unidad de Servicios de Cómputo de la Facultad de Ingeniería.

El modelo de seguridad se diseñó pensando en un amplio rango de casos posibles. La implantación de este modelo es tan variada como cada una de las organizaciones en las que se aplique, en este sentido es importante mencionar que existen algunos puntos que determinan particularmente el tiempo y los recursos necesarios para la implantación del modelo de seguridad propuesto: Tipo de organización, Tipo de información y Recursos (humanos, económicos) con que actualmente cuenta la organización.

La implantación del modelo en el caso práctico, tiene como objetivo ejemplificar la investigación que se realizó en el presente trabajo.

5.2. Modelo de Seguridad

En la siguiente página se muestra el modelo de seguridad que proponemos. Primero se enuncian los diferentes pasos, se muestra un diagrama general y finalmente se explican de manera más detallada cada uno de los puntos.

1. *Análisis del caso particular.* Definición de la organización técnica, lógica y organizacionalmente. Establecer:
   a) Tipo de Organización.
   b) Tipo de Información que maneja la organización y valor de la misma.
   c) Análisis de la Situación Actual, condiciones físicas y lógicas de:
      (1) Planta Física
      (2) Topología Física
      (3) Topología Lógica
      (4) Recursos Humanos
2. *Análisis de riesgo.* Definir los riesgos y vulnerabilidades en la seguridad para los distintos elementos, probabilidad de que ocurran y costo para la organización, asignando niveles de seguridad para identificación.
3. *Planteamiento de los esquemas de seguridad.*
   a) Selección de las Políticas de Seguridad adecuadas para cada problema, estableciendo métodos de solución.
   b) Verificación de la validez de los esquemas, que correspondan con los problemas, necesidades y recursos.
   c) Plan de Implantación.
   d) Limitación del alcance de los Esquemas de Seguridad.
      (1) Líneas Inmediatas
      (2) Líneas Futuras
4. *Aplicación de las líneas inmediatas.*

5. *Prueba del modelo y retroalimentación*, calculando los niveles de riesgo residuales.

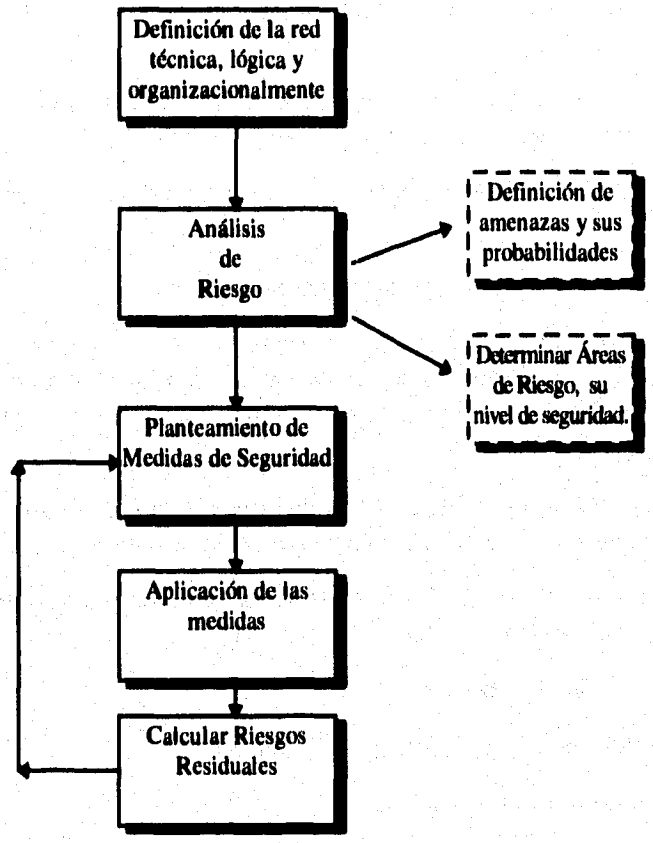
6. *Mantenimiento*, verificación y, en caso necesario, replanteamiento de las políticas que así lo requieran.

7. *Implantación de las líneas futuras* conforme sea posible (o necesario).

Los pasos v, vi y vii deben ser una actividad constante.

5.2.1 De manera gráfica:

MODELO DE SEGURIDAD



5.2.2 Descripción de las distintas etapas.

5.2.2.1 ETAPA *i* Análisis del caso en particular. Definición de la organización técnica, lógica y organizacionalmente

Para definir cualquier esquema de seguridad es importante definir cual es el conjunto de necesidades y características de la organización (*a) Tipo de Organización*),

el primer paso es definir cómo está estructurada nuestra organización, cuáles son sus objetivos y funciones de manera jerárquica y cuáles son sus necesidades de seguridad.

En esta parte debe de descomponerse el sistema perfectamente en sus distintos elementos, es decir hacer una separación lógica de planta física, topología física, información, topología lógica y recursos humanos.

Antes de definir las medidas de seguridad para cualquier sistema es primordial entender y hacer entender la importancia de dichas medidas y los beneficios que espera la organización de ellas. Se deben establecer sus objetivos y no perderlos de vista. Estos objetivos deben incluir:

- Las medidas de seguridad deben estar definidas por necesidades específicas.
- Las medidas no son independientes del resto de las funciones de nuestra red por lo que deben desarrollarse tomando en cuenta todas las disciplinas que intervengan en ella.
- La implantación de las medidas no deben tener impacto (o por lo menos no significativo) en el desempeño y funciones de la organización.
- El costo de las soluciones debe estar de acuerdo con el valor de los recursos a proteger.
- Nunca se debe creer que el sistema es completamente seguro.
- Debido a que la tecnología avanza a pasos agigantados, siempre debemos investigar las nuevas opciones de seguridad, pues no porque una medida o procedimiento haya funcionado en el pasado o en otra organización significa que vaya a funcionar aquí y ahora.
- La seguridad tiene muy alta prioridad en cualquier organización. Si la seguridad se planea, implementa y maneja adecuadamente proveerá beneficios notables.

En el punto correspondiente a *b) Tipo de Información que maneja la organización y valor de la misma* debemos definir:

- Misión y objetivos de la organización.
- Tipos de información sensible que maneja la organización
- Valor de la información para la organización.

En esta parte debemos indicar la información de los usuarios así como la del sistema, tales como bases de datos, listas de control, señalando dónde se encuentra almacenada dicha información.

Para el punto *c) Análisis de la situación actual, condiciones físicas y lógicas*, debemos describir todos los elementos físicos (topología física), entidades que componen nuestra planta física como:

Edificios, muebles, aire acondicionado, electricidad, equipo contra incendio, sistemas de respaldo, hardware: terminales, servidores, pc's, enrutadores, puentes, concentradores, cableado, gateways, impresoras, *scanners*, módems, dispositivos de cinta.

Los elementos que conforman la parte lógica, software: sistemas operativos, sistemas de manejo de archivos, manejadores de bases de datos, procesadores de palabras, aplicaciones de red como correo o detransferencia de archivos, protocolos de comunicación, componentes de red especializados, como programas para enrutamiento.

Debe quedar perfectamente delimitada y documentada nuestra red. Resultaría muy útil en este proceso discutir los planes a futuro para la red a fin de tomarse en cuenta en la delimitación.

El último punto, pero no por eso menos importante, consiste en delimitar el grupo de trabajo, debe tomarse en consideración el alto valor que tiene la gente (*c) Recursos Humanos*) para la organización. Por lo que toda organización debe asegurar la protección de su personal. Su importancia radica no sólo en el costo de entrenamiento, sino de reemplazo preparando nuevo personal.

5.2.2.2 ETAPA *ii* Análisis de Riesgo.

Esta es la actividad más importante y también la más compleja. Vale la pena recordar que no existe ninguna red libre de riesgo aún después de establecidas las medidas, sin embargo los riesgos deben reducirse a un punto controlable no perdiendo de vista los costos.

En esta etapa se debe determinar:

- El valor de los datos actuales.
- Posibles amenazas contra la red, cuantificando las pérdidas en caso de que ocurrieran dichas amenazas.
- Las vulnerabilidades de la red contra dichas amenazas.

El análisis de riesgo define el sistema a ser evaluado y establece las acciones a tomar a futuro basándose en las características de la red establecidas en el punto anterior.

Para el análisis de riesgo debe formarse un equipo de trabajo, dependiendo el tamaño de la organización será el número de elementos, es sugerible que exista un líder preferentemente externo de tal forma que no esté viciado, debe proveer también la experiencia y los conocimientos necesarios. Nunca se deben dejar fuera de este equipo los dueños y/o administradores y/o usuarios de la red que proveerán su conocimiento y experiencia respecto a los problemas y desempeño actual, qué datos se usan y donde, los recursos, aplicaciones, cómo y quien usa qué servicios, qué usuarios requieren qué datos para desempeñar sus labores, etc. Se sugiere tener (sino en el grupo, por lo menos disponible de algún modo) la experiencia técnica para manejar los posibles problemas que se presenten de manera que en el análisis no se asuman cosas que no se deban.

Sea cual fuere la organización, se deben establecer niveles de responsabilidad para todo el proceso.

Para los distintos elementos en los que se haya dividido nuestra red, se establecerá su valor definiendo si resulta un elemento crítico o no. Podemos decir que desde la perspectiva de red, se debe establecer el valor de:

- La información actual.
- Los componentes de la red tanto de hardware como de software.
- Los servicios que provee la red.

Para establecer estos valores se deben de tomar los criterios:

- Confidencialidad
- Integridad
- Disponibilidad.

Debe determinarse si la precisión de la información puede ser mucho más importante y de mayor valor que la confidencialidad de la información.

Cuando se establezca el valor de los elementos en cuanto a disponibilidad debe considerarse el impacto de que deje de existir o sea destruido completamente o su valor respecto al tiempo que deje de estar disponible.

El valor tiene dos formas distintas, el valor intrínseco al objeto (valor físico), por ejemplo el valor de compra, y el valor que ha adquirido en el tiempo, éste generalmente depende de la información que contiene.

El proceso de valuación debe documentarse.

Durante este proceso es importante no perder de vista que pueden existir miles de amenazas potenciales, sin embargo, la mayoría de éstas pueden no ser relevantes para la red en cuestión, estas dependen de la red, el tipo de información y sus objetivos.

En general, en los capítulos anteriores ya se trataron las distintas amenazas, la mayoría pueden ocurrir en cualquier red, pero como se ha mencionado, algunas son menos o más importantes. El siguiente paso es establecer la lista de prioridades respecto a seguridad y costo de las vulnerabilidades de nuestra red a tomarse en consideración.

Primeramente se establecerá la probabilidad de que ocurran las amenazas, éstos niveles de gravedad pueden ser 3 alto, 2 moderado y 1 bajo, estos niveles pueden ser distintos según defina el grupo de trabajo.

Posteriormente se debe determinar el costo de que una amenaza se lleve a cabo.

5.2.2.3 ETAPA *iii* Planteamiento de los esquemas de seguridad.

Después de que han quedado completamente claras las amenazas, vulnerabilidades y riesgos de la red, se procede a plantear un esquema de seguridad. Dependiendo de nuestros recursos y necesidades se elegirán la amenazas cuyo nivel de probabilidad sea más alto, así como el costo que implicaría a nuestra organización que se llevaran a cabo. Una opción factible es elegir diez de las amenazas (o un número similar) y atacarlas (*líneas inmediatas*). Se seguirán los puntos que se mencionan después. Y después de cumplida la misión se elegirán otras nuevas vulnerabilidades a solucionar (*líneas futuras*). Recordemos que la actividad de seguridad es constante y que nunca será completamente segura nuestra red, por lo que no es recomendable ser demasiado ambicioso al inicio ya que el elemento tiempo puede ser básico en la defensa de nuestra red, así que primero cerraremos las puertas más obvias y más peligrosas y poco a poco iremos reforzando los agujeros más pequeños y que convengan de acuerdo a la misión de la organización.

Ya que se seleccionaron los problemas a atacar los siguientes pasos son:

Seleccionar las Políticas de Seguridad adecuadas para cada problema, estableciendo métodos de solución (muchos de estos métodos han sido citados en capítulos anteriores), verificando que dichos esquemas de seguridad correspondan con la realidad de nuestra red en cuanto a necesidades y recursos, recordemos que el costo de perder algún recurso no puede ser menor que el de la barrera que se implanta.

Después de establecidos los esquemas de seguridad se procede a delinear un plan de implantación en cuanto a cronología de actividades y prioridades.

Como se dijo ya, los esquemas de seguridad pueden no ser alcanzados en un corto plazo, dependiendo de las necesidades habrán actividades a futuro o de aplicación inmediata.

Un punto importante es calcular el valor de aplicar una medida de seguridad. Esto justificará la aplicación (o la no aplicación) de una medida.

Todos estos esquemas de seguridad constituyen la política de seguridad de la red. Todo este programa se debe documentar, explicando cuál información se va a proteger, y las reglas que reforzan estas determinaciones. La política también documenta cuales datos y recursos deben de protegerse, por qué y cómo se protegerán. También se debe enunciar al personal que tendrán la responsabilidad y sobre qué actividades. En cuanto a las medidas de seguridad, se explicarán claramente y cómo son usadas.

El control de acceso, la seguridad física, lógica, contabilización y todas las medidas deben estar especificadas en la política de seguridad de la red.

5.2.2.4 ETAPA *iv* Aplicación de las Líneas Inmediatas.

Los diez primeros puntos (o el número que defina el grupo de trabajo), de aplicación inmediata por su importancia, se llevan a cabo en esta etapa. Para el desarrollo

de las actividades deberá elegirse el personal adecuado, recordemos que la elección correcta puede significar el éxito o fracaso de nuestro proyecto.

5.2.2.5 ETAPA *v* Prueba del modelo y retroalimentación calculando niveles de riesgo residuales.

Esta etapa y la siguiente se llevan a cabo de manera conjunta.

El riesgo residual se define como el riesgo remanente después de aplicar las medidas de seguridad. Este se lleva a cabo recalculando las probabilidades de que ocurran las amenazas encontradas. Puede ser necesario iterar los pasos anteriores hasta que el riesgo residual se encuentre en un nivel aceptable.

5.2.2.6 ETAPA *vi* Mantenimiento y en caso necesario, replanteamiento de las políticas que así lo requieran.

Esta etapa generalmente ocurre como consecuencia del punto anterior, es decir, después de calcular los riesgos residuales, ya sea en la implantación de las políticas o en el mantenimiento propiamente de un esquema ya establecido, se procederá a replantear políticas, elegir nuevos métodos y verificar si son o no aplicables.

En cualquier momento, la utilidad de una medida adicional puede no incrementar la seguridad de la red pero si actuar en detrimento del desempeño de sus funciones y/o incrementar el costo de mantenimiento. Por lo que la reducción del riesgo debe ser un balance entre el nivel de protección y el impacto en el desempeño y costos.

La última decisión de aplicar o no una medida adicional debe recaer en el jefe responsable de la red, en conjunción con las realidades de presupuesto y recursos.

En muchos de los casos una segunda opinión puede ser muy útil en el proceso de análisis de riesgo.

Cuando se realicen cambios substanciales o se modifique una política, se debe actualizar el análisis de riesgo documentando las modificaciones y los argumentos para éstas.

5.2.2.7 ETAPA *vii* Implantación de las líneas futuras.

Conforme sea posible o necesario se irán implantando las líneas futuras, este punto podemos decir que es volver al inciso *iv* sólo que las líneas futuras se convierten en inmediatas, repitiéndose el ciclo.

A partir de la etapa *v* podemos decir que se entra a un ciclo constante. El ciclo de la seguridad nunca termina. Las políticas de seguridad deben regir las actividades diarias, siendo desarrolladas por personal confiable, revisadas, documentadas y formalmente publicadas y entregadas al personal correspondiente.

Existen algunos puntos respecto a la política de seguridad en general que debemos seguir:

- Todos los usuarios antes de tener acceso a la red deben de estar entrenados en sus respectivas responsabilidades (especialmente respecto a seguridad), sobre los mecanismos de seguridad y las penalizaciones a las que se hacen acreedores en caso de infringir o intentar sobrepasar algún control de seguridad.
- Las políticas y procedimientos deben ser documentados para hacerlos del conocimiento de los usuarios.
- Todos los usuarios que requieran acceso a la red deberán atender a un entrenamiento de seguridad antes de que se les otorguen privilegios a sus cuentas.
- Debe llevarse a cabo un proceso de contabilización que provea información sobre las actividades de los usuarios, acceso a la información y eventos relevantes de seguridad.
- Vale la pena insistir en que el entrenamiento del personal los capacita para entender la importancia de la seguridad. Además les indica sus responsabilidades y las posibles penalizaciones.
- El proceso de respaldo es uno de los más importantes en cualquier red, por lo que se le deberá prestar especial atención.
- Las políticas de seguridad generalmente serán establecidas después del reporte de algún incidente, por lo que se deberá definir lo que se considera como un incidente de seguridad y quién deberá ser alertado. Los usuarios deberán estar atentos a sus responsabilidades respecto a la información de los posibles incidentes. Todos los incidentes deberán ser documentados.

5.3. Caso Práctico

A continuación se describe la implantación del modelo de seguridad anteriormente descrito para la Unidad de Servicios de Cómputo Académico (o CECAFI).

5.3.1 - Análisis del Caso Particular

5.3.1.1 -Tipo de Organización

Desde hace más de 15 años ha sido la principal instancia de servicio y desarrollo de sistemas de cómputo para la Facultad de Ingeniería, además de proveer este tipo da servicios a otras instituciones. Recientemente fue dividida en dos Unidades: la Académica y la Administrativa. Este trabajo se enfoca a la Académica debido a que es la que mantiene el equipo que se enlaza con la Internet utilizando TCP/IP.

A inicios de 1990[1] el Centro de Cálculo instaló una pequeña red local para unir sus sistemas de cómputo principales que hasta antes se utilizaban únicamente para

[1]Arrieta Marcos, Norberto. *Introducción a la RedCECAFI*. UNAM Facultad de Ingeniería, 1992

transferencia de archivos entre los sistemas internos del Centro. En 1991 se registran dos importantes sucesos que incrementan notablemente las aplicaciones de la red:

- Se integra el Centro de Cálculo a la RedUNAM
- Adquisición de estaciones de trabajo y conexión de las mismas a dicha red.

Debido a esta conexión que nos enlaza con la Internet tenemos en nuestras manos un universo de conocimientos y por lo tanto de crecimiento. Sobra mencionar que la Internet significa accesar a cualquier parte del mundo, comunicación con la gente y los grupos que hacen la investigación y el desarrollo de vanguardia en sólo cuestión de segundos. Significa poder encontrar rápida y verazmente el conocimiento de actualidad. Es también un detonador de la curiosidad para la investigación y el desarrollo no sólo del área de cómputo sino de cualquier área del conocimiento ya que la Internet tiene puertas abiertas para los más inimaginables temas.

De modo que este recurso implica que cualquier Ingeniería puede crecer, encontrar información y respuestas por medio de la red.

Pero que nosotros podamos tener acceso a casi todo el resto del mundo significa que el mundo también puede tener acceso a nuestros recursos y por lo tanto dañarlos.

Ahora bien, ya sabemos que tenemos un gran recurso en nuestras manos y que de él podemos obtener gran provecho, tenemos conciencia de que podemos sufrir daños. Lo importante es encontrar cuáles son los ataques que pueden perjudicar a nuestros sistemas, cómo prevenirlos o combatirlos, si nos conviene hacerlo y cómo.

5.3.1.2 - Tipo de Información que maneja la Organización

Esta institución se encarga de dar atención a usuarios, cursos, asesorías, desarrollar sistemas para otras áreas de control de alumnos, profesores; también difunde novedades en el conocimiento de computación hacia todas las áreas de la Ingeniería y se encuentra conectada directamente con la Unidad que lleva el control de alumnos, realiza inscripciones y lleva a cabo estadísticas de profesores.

Por lo que podemos decir que en su mayoría la información es académica (tareas, proyectos de investigación, programas), y en menor cantidad sistemas en desarrollo para el servicio de la comunidad.

La información de los usuarios es importante en cuanto a que de ella dependen los alumnos y profesores para la evaluación principalmente. Además se encuentran las herramientas de desarrollo de la cual dependen alumnos y profesores que no necesariamente estén desarrollando una tarea para una materia en particular sino como investigación para otros proyectos particulares, tales como tesis. Por lo que la información no sólo debe estar inalterada sino disponible en el momento preciso.

Podemos dividir de manera jerárquica la información en:

TIPO DE INFORMACION			VALOR
A.	Sistemas de Servicio para la Facultad de Ingeniería		3
B.	Software de desarrollo y del propio sistema:		
	1.	Protocolos	3
	2.	Sistemas Operativos	3
	3.	Bases de Datos	2
	4.	Compiladores	2
C.	Información de Alumnos y Profesores:		
	1.	Tareas	1,2
	2.	Proyectos de Investigación	1,2
	3.	Tesis	1,2
	4.	Otros	0,1

La información a) no se encuentra almacenada en equipos conectados directamente a la red. Esta es una muy buena medida de seguridad. Si un sistema no tiene por que estar conectado a la red externa y por lo tanto ser susceptible a intromisiones, no lo hagamos. Mantener aislada la información más sensible sin afectar a quienes hacen uso de ella es muchas veces la mejor opción.

Por lo tanto la información sensible a resguardar será la b) y c), atendiendo primeramente a la de valor 3, posteriormente la 2 y 1, dejando para después, tal vez para otro ciclo, la 0.

Dicha información se encuentra almacenada en las distintas estaciones de trabajo con disco (ALIKA, CANCUN, COZUMEL, TORK y BALAM). El software de desarrollo se encuentra instalado en su mayoría en todas las estaciones, excepto ORACLE que está en Alika e IDEAS que está en Cancún.

Hemos clasificado a la información en un orden de 0 a 3, donde la 3 es importante para las funciones de los sistemas durante todo el período escolar, la 2 y la 1 pueden llegar a ser incluso 0 dependiendo delpunto en el tiempo en que nos encontremos, es decir, al principio del semestre existe muy poca demanda del equipo y sus servicios por lo que un ataque en este momento tendría un costo muy bajo. Sin embargo, al final del semestre el perder la información o la disponibilidad, incluso por horas puede ocasionar un caos. Por esta razón cieta información fue clasificada como 0 y en ocasiones1, o 1 y posiblemente 2, ya que en cierto período puede tornarse más o menos valiosa (o el costo de perderla menos o más alto).

5.3.1.3 Análisis de la Situación Actual

La redCECAFI se basa en la tecnología Ethernet, que es una derivación de la de bus, llamada bus ramificado. En ésta puede haber varios buses conectados mediante dispositivos llamados repetidores que se encargan de duplicar cada uno de ellos.

La topología exacta de la redCECAFI se muestra en la figura de la siguiente página. En ella se observan cinco buses unidos por un repetidor multipuerto, en los que se distribuyen: dos microcomputadoras VAX, 14 estaciones de trabajo, cuatro servidores con terminales (con una capacidad de 16 terminales cada uno) y un numero variable (aproximadamente 14) de computadoras personales.

Este equipo cuenta con accesos desde varios lugares de la Facultad: de las dos Unidades que conforman el Centro de Cálculo, tanto en el edificio principal como en el Anexo; la División de Ciencias Básicas, la División de Ingeniería Mecánica e Industrial, la División de Ingeniería Eléctrica, Electrónica y en Computación; la División de Ingeniería Civil; el Departamento de Fluidos y Térmica; la Secretaría de Servicios Escolares y la Oficina de Servicios Escolares.

La red del Centro se encuentra conectada a la redUNAM mediante un repetidor de fibra óptica. La redUNAM tiene sus nodos principales en la Dirección General de Servicios de Cómputo Académico, el Instituto de Matemáticas Aplicadas y Sistemas y el Instituto de Astronomía, incluyendo muchos institutos y facultades (en la siguiente página podemos observar el diagrama de la red del Centro).

Las estaciones de trabajo con las que cuenta el Centro emplean como sistema operativo UNIX, y para las comunicaciones TCP/IP. Las microcomputadoras VAX emplean VMS y DECNET, de manera correspondiente. Por lo que nuestro punto de interés se enfocará a las estaciones Apollo-HP.

Los sistemas HP 7000 y Apollo, le dan entrada no sólo al Centro sino a toda la comunidad de la Facultad al inmenso mundo de la Internet, utilizando como columna vertebral al conjunto de protocolos TCP/IP.

El equipo cuenta con sistema operativo HPUX 9.0, manejador de base de datos ORACLE, compiladores de C y C++, IDEAS (paquete de diseño), distintas utilerías (por ejemplo de compactación), X11, acceso al WWW.

En cuanto a recursos humanos el centro se compone esencialmente de becarios, alumnos de la Facultad de Ingeniería, los que pasan generalmente de 2 a 3 años como estudiantes y becarios en el centro, algunos pocos son contratados y permanecen por más tiempo, pero la gran mayoría parte a integrarse a otras empresas. Esto se debe a que el centro no es sólo de servicio sino también formativo. El personal encargado del equipo se encuentra alrededor de 12 personas entre el jefe del centro, jefe de departamento y encargado del centro de cómputo, administradores de las estaciones y administradores de la base de datos.

Existe además personal contratado por la Universidad encargado del control de entrada y salida de usuarios así como de la asignación del equipo.

5.3.2 Análisis de Riesgo

5.3.2.1 Vulnerabilidades Físicas

En el punto anterior se mencionó la información que maneja el centro su valor así como el equipo con que cuenta, como es de imaginarse el equipo por ser el receptáculo de la información y por su costo es en sí una de las entidades de mayor valor.

Por lo que el sitio en el que se encuentra ubicada la sala de cómputo es ahora nuestro primer objetivo, debido a que de alguna manera su construcción puede llevar a hacer más o menos fácil la intervención o destrucción del equipo.

El problema respecto al edificio es que no se cumplen los requerimientos de construcción de un centro de cómputo, los más obvios son que no existe aire acondicionado; las ventanas, que en ocasiones tienen los vidrios incompletos, permiten la entrada de animales, polvo y lluvia, además de que son demasiado grandes y por lo mismo fáciles de romper e irrumpir en el Centro.

El control de usuarios no es totalmente estricto ya que si existe un gran número de usuarios solicitando entrada, la persona que verifica las credenciales no tiene la visibilidad suficiente para notar si alguien sin autorización entra a usar el equipo.

Las máquinas son fácilmente reinicializables por el usuario y no se ha implementado una medida de seguridad que no permita que la máquina entre como "single user" y solicite la contraseña de *root* (esto es parte de la configuración de UNIX).

Los cables se encuentran expuestos por lo que resulta fácil que algún usuario (con o sin intención) desconecte algún equipo.

En resumen, el equipo es un elemento crítico para la funcionalidad de la red, existe una alta vulnerabilidad del equipo a ser atacado, en caso de que una amenaza se llevara a cabo las pérdidas podrían llegar a ser de miles de dólares si la pérdida fuera total, en caso de que sólo llegará a dejar de estar disponible el valor respecto a tiempo es importante, sin embargo no es de minutos, esto es, si el equipo deja de estar disponible inclusive un día (en gran parte del semestre) las funciones pueden reestablecerse con pocas pérdidas debido a que la información debe estar disponible y estar íntegra, pero su disponibilidad, debido a que los sistemas no son de tiempo real, no es inmediata. Por ejemplo, si un alumno no tiene acceso a su tarea hoy es fácil que el profesor entienda que es por fallas en el equipo y espere un día más. El valor de disponibilidad crece dependiendo de la fecha, esto es si nos encontramos al final del semestre posiblemente el ataque tenga mayores consecuencias ya que son más usuarios, y los profesores pueden esperar un día pero no semanas ni meses para tener acceso a la información.

El último punto a señalar es el acceso que se podría tener por medio de esta red al sistema en que se llevan a cabo las inscripciones, este sería uno de los más valiosos puntos de ataque para un enemigo. Pero debido a la división de las Unidades queda fuera de la jurisdicción de la Unidad de Servicios de Cómputo Académico la red que maneja estos procesos.

5.3.2.2 Vulnerabilidades Lógicas

Los problemas de seguridad que provienen de fallas en el protocolo TCP/IP se encuentran a un nivel muy bajo, siendo casi seguro que ningún atacante intentaría entrar por este medio, ya que la información que fluye de la Unidad hacia afuera (archivos de tareas de usuarios o de investigación por cuenta propia del mismo, correo electrónico - siempre y cuando no sean contraseñas-, etc.) no es tan costosa como lo que implicaría un sistema para desviar, enviar o alterar paquetes.

El verdadero problema radica en el acceso que tienen los usuarios que pueden ser atacantes potenciales, los fines de semana o en horarios no laborables, ya que en esos tiempos no existe ningún monitoreo por parte de los administradores, por lo que los atacantes pueden estar experimentando hasta aprovechar agujeros de seguridad y mediante ellos dañar los sistemas, teniendo todo el tiempo disponible para hacerlo sin ser detectados.

En cuanto a los principales problemas de UNIX, se resumen de manera simple:

- Problemas en la administración
- Agujeros de configuración
- Contraseñas

Debido a estos problemas la confidencialidad y la integridad de la información es muy dudosa. La probabilidad de que ocurran ataques que tomen provecho de estas vulnerabilidades es altísima, ya que ya han ocurrido ataques con anterioridad y pueden repetirse. El mayor problema es que debido a estas fallas se puede lograr que se pierda la disponibilidad de la información, es decir, un atacante con el suficiente conocimiento puede aprovechar los agujeros y destruir información valiosa.

El costo de que una amenaza se lleve a cabo es muy alta, pero si existen los respaldos adecuados puede llegar a ser muy baja.

5.3.2.3 Recursos Humanos

Puede parecer increíble pero como en la mayoría de los casos el más peligroso atacante está dentro de los propios miembros de la institución que tienen acceso con mayores privilegios que el usuario común.

Y para muestra un botón. A principios de 1994, las estaciones de trabajo eran administradas por una sola persona que tenía control total sobre ellas, la única seguridad que funcionaba era la "oscuridad", ya que se practicaba el no conocimiento para otros administradores. Pero cómo ya se ha aclarado a través de esta tesis, el negar manuales o información a los usuarios no les impide que por su propia cuenta aprendan. Con el cambio de dirección en la Unidad surgieron nuevas ideas y por distintas razones la ex-administración decidió dañar el equipo, así que destruyó información, tomó con el dispositivos de hardware y aprovecho otros agujeros en beneficio propio. La gente del centro tuvo que trabajar a marchas forzadas para recuperar lo recuperable y reorganizar la administración.

Además de las "malas intenciones" del personal, se encuentran aquellas deficiencias en el manejo de la administración, o bien un mal manejo de los altos privilegios que pueden tener y por deficiencias en las políticas y en la comunicación puedan llevar a que distintas personas realicen actividades que se contrapongan con las de otros.

Como se puede ver la probabilidad de que ocurra una de estas amenazas depende del tipo de gente con que se trabaja y del ambiente de trabajo, el costo de que se lleve a cabo puede ser muy alto.

Otro problema respecto a los recursos humanos es el hecho de que en cualquier momento pueden dejar de laborar para la Unidad llevando consigo el conocimiento y pudiendo dejar desprotegida alguna área por un cierto período, que dependiendo las políticas de entrenamiento puede ser mayor o menor. El costo de reentrenar personal puede llegar a ser muy alto. Este problema tiene mucho que ver con una deficiente documentación, ya que sin ella se pierden todos los logros del personal, los agujeros que hayan encontrado, su experiencia respecto a ataques y las medidas implantadas, etc.

5.3.3 Planteamiento de los Esquemas de Seguridad

5.3.3.1 Políticas de Seguridad

De acuerdo al tipo de organización y al tipo de información que se maneja en este caso particular, la cual es totalmente académica, las políticas de seguridad se deben tomar pensando en los servicios y funciones del CECAFI dentro de la Facultad de Ingeniería y aún más dentro de la UNAM.

De acuerdo a la experiencia como usuario y como personal que labora dentro del CECAFI se elaboró el siguiente esquema de políticas de seguridad. Las primeras se refieren a la parte física, posteriormente a la lógica y finalmente a los recursos humanos. Esta última parte se divide en dos: Políticas de Seguridad para los usuarios finales (alumnos) y Políticas de Seguridad para el personal que labora en el CECAFI. Siendo el manejo de los recursos humanos el más importante y que repercute directamente en los dos primeros ya que de ellos depende el buen o mal funcionamiento y los resultados, se mencionan en primera instancia.

5.3.3.2 Políticas para la parte física

De acuerdo a la teoría presentada en el capitulo 3 del presente trabajo, existe equipo de comunicaciones en el cual se pueden implantar medidas de seguridad. Debido a que el equipo de comunicaciones que utliza la red del CECAFI, no está ubicado físicamente dentro de las instalaciones del mismo, no es posible realizar un análisis preciso de cuestiones de seguridad al mismo.

El equipo de comunicaciones por medio del cual se accesa al mundo exterior, se encuentra ubicado en el edificio de la DGSCA. En cuanto al acceso a este equipo, por razones obvias y para ser congruentes con el perfil y objetivo de este trabajo, no se nos permite.

El personal que labora en el área de redes y Telecomunicaciones en la DGSCA, se ha encargado de implantar ciertas medidas de seguridad en la red. Como ejemplo de una de ellas es el diseño e implantación de listas de acceso en los enrutadores de la RedUNAM, así como el diseño de enlaces alternos y redundancia en las rutas.

Pasando a lo que respecta a los problemas de acceso físico, podemos decir que en la Unidad de Servicios de Cómputo el control de acceso de los usuarios ha tenido un avance con la introducción de fotografía en la credencial, pero aún falta mayor cuidado para evitar totalmente que alguien salte la barrera de la autorización de entrada para hacer uso del equipo. Por lo que se hacen las siguientes sugerencias:

⇒ Establecer controles de acceso y autenticación de usuarios, por ejemplo, credencial con fotografía y código de barras, cuando menos. Y si es posible la selección de algún otro dispositivo de autenticación como los mencionados en el capítulo anterior, dependiendo de los recursos. Una posible solución para la implementación a bajo costo de alguno de estos sistemas de autenticación sería el permitir que algún (os) alumnos desarrollarán un proyecto que cumpliera con las características deseadas.

⇒ Establecer horarios, y respetarlos, según privilegios de usuarios y administradores.

⇒ Mejorar las instalaciones, colocando ventanas que mantengan sellado el centro para el resguardo del equipo.

⇒ Instalación de aire acondicionado.

⇒ Instalación de alarmas de seguridad y contra incendio.

⇒ Ocultar el cableado.

⇒ Segmentar y centralizar las redes de acuerdo a niveles de seguridad.

⇒ Establecer políticas para casos de siniestro.

⇒ Establecer políticas de respaldo periódicas.

⇒ Mantener los respaldos en lugar seguro y al alcance de los administradores (no en el mismo centro).

⇒ Contratar mantenimiento preventivo para el equipo.

5.3.3.3 Políticas para la parte lógica

Hablar de la seguridad que podría implantarse en UNIX en este aspecto podría llevar uno o más tomos, por lo que sólo hablaremos de los principales problemas, o bien, los primeros y más importantes a resolver en nuestro sistema en particular. Es obligación de los administradores la implementación de otras alternativas de seguridad, existe suficiente bibliografía del tema, e incluso los manuales del sistema sugieren esquemas de seguridad que no traen configurados por omisión. Además de los manuales y la bibliografía existen bastantes grupos de discusión de administración y de seguridad en

UNIX, no sólo en la Universidad sino en el mundo entero, con los que se pueden resolver dudas y reconocer agujeros de la seguridad que no se habían imaginado.

El primer problema a resolver es el de las contraseñas, éstas son un dolor de cabeza ya que son la principal entrada a nuestro sistema, y por lo tanto la más atacada, debe establecerse una política de selección de contraseñas seguras y un período para que esta sea cambiada. Deben implementarse los mecanismos para hacer cumplir esta disposición, para ello se puede adquirir software del mercado, si el sistema operativo lo permite hacer las necesarias adecuaciones en donde los manuales indiquen, o bien tomar software disponible gratuitamente en la Internet que ya controlan las contraseñas y pueden adecuarse a nuestras necesidades (algunos ya fueron mencionados en el capítulo anterior).

La mayoría de los problemas de UNIX mostrados en los reportes, anexados al final de este trabajo, se arreglan con un vistazo rápido al manual o con políticas de administración más estrictas.

De cualquier manera COPS trae información adicional sobre el significado de las advertencias que envía y posibles soluciones, la administración debe revisarlas, seleccionando y aplicando las que crea convenientes. Por lo que se sugiere ampliamente instalar COPS.

Otras soluciones ya han sido planteadas en el capítulo anterior y en la bibliografía proporcionada se pueden encontrar otras opciones, sin embargo cabe recalcar que no en todo sistema conviene implantar toda medida de seguridad, los administradores serán los responsables de decidir si alguna medida puede afectar el buen funcionamiento y utilización de los recursos a tal grado que más que ayudar sea un obstáculo para los usuarios.

Debe verificarse periódicamente que no surjan nuevos agujeros de seguridad, y si es posible monitorear las posibles intromisiones o daños que pueda sufrir nuestro sistema para de esta manera estar alerta y corregir los problemas que surjan.

A) Políticas generales.

Repartir a los usuarios todas las reglas por escrito en cuanto se les otorgue acceso al centro así como las sanciones a las que se hacen meritorios en caso de no cumplirlas.

Repartir las políticas establecidas al personal que labora en el centro así como las sanciones que podrían sufrir en caso de no acatarlas.

Establecer una política de retroalimentación entre personal y usuarios para verificar que las políticas se estén llevando a cabo correctamente, definir algunas nuevas y retirar las obsoletas o que se decida si son más bien una barrera para el buen desempeño de las funciones.

B) Políticas para los usuarios finales

Los recursos del CECAFI deben ser utilizados única y exclusivamente para fines académicos.

Todo usuario debe presentar su credencial para tener acceso al Centro de Cálculo.

El uso de los recursos deben ser respetados de acuerdo a la asignación a cada uno de los usuarios, los cuales varían en privilegios de acceso y horarios dentro del Centro de Cálculo.

Por ningún motivo se debe de permanecer dentro del Centro de Cálculo cuando no sea horario laborable.

C) Para el personal que labora en el Centro de Cálculo.

La jefatura del centro estableció políticas para los administradores, que son los únicos con acceso privilegiado al equipo. (Ver Apéndice 3).

Además se sugieren las siguientes.

Cualquier modificación a la configuración de los equipos debe ser hecha por orden del jefe inmediato.

El personal debe ser capacitado regularmente de acuerdo a las actividades que realice.

5.3.3.4 Plan de Implantación

De acuerdo al análisis realizado, se diseñó el siguiente plan de implantación del modelo de seguridad, en el cual se toman algunas consideraciones preliminares para la implantación del mismo. Debido a que no se tiene acceso a todos los recursos necesarios para la implantación del modelo, en esta primera etapa, no es posible abarcar todos los rubros del modelo de seguridad propuesto. Se propone la instalación de programas que nos ayudarán a monitorear y evitar los problemas de seguridad en el Centro de Cálculo, los cuales forman parte de las líneas inmediatas dentro de la implantación del modelo. Todas las acciones que no van a ser tomadas en esta primera etapa, se contemplarán dentro de las líneas futuras. El plan de implementación es el siguiente: (líneas inmediatas)

1.Reasignación de funciones del personal del Centro de Cálculo (principalmente de administradores)

2.Capacitación del personal que labora en el Centro de Cálculo

3.Lineas inmediatas: Seguridad en TCP/IP y UNIX

3.1 Instalación y configuración de TCP Wrapper

3.1 Instalación y configuración de COPS

3.3 Instalación y configuración de Password+

3.4 Establecer política de Respaldos

4. Diseño de las líneas futuras: Planta Física, Recursos Humanos, Topología Lógica.

5.3.4 - Aplicación de las líneas inmediatas

A continuación mencionaremos las principales soluciones que fueron implementadas, las que no fueron pero que convendría hacerlo y algunas opciones más.

5.3.4.1 Seguridad de TCP

Retomando la información recavada durante el análisis, se vió que el verdadero problema radica en el acceso que tienen los usuarios que pueden ser atacantes potenciales, los fines de semana o en horarios no laborables.

Por esta razón se implementó el TCP-Wrapper de la siguiente manera.

Configuración e instalación sencilla del TCP wrapper

1. El primer paso es conseguir los programas fuentes del TCP wrapper que son de acceso público. Existen varias direcciones en la Internet que poseen acceso mediante ftp anónimo y que pueden proveernos del TCP-wrapper, una de éstas se encuentra en: *ds5000.dgsca.unam.mx*, el archivo se encuentra compactado bajo el nombre de tcp_wrapper.tar.gz.

2. El archivo se encuentra compactado con gzip (utilería también de acceso público), con el comando: *gunzip tcp_wrapper.tar.gz* es posible descompactarlo.

3. En alguna cuenta de usuario especialmente creada (como sugerencia) para el manejo de estos programas se crea un subdirectorio llamado *tcp_wrapper*, bajo este se copia el archivo tcp_wrapper.tar y se ejecuta el comando:

tar -xvf tcp_wrapper.tar

En este subdirectorio quedarán todos los archivos que constituyen el TCP-wrapper

4. Escribir 'make' y seguir las instrucciones que se despliegan. El Makefile viene con plantillas listas para utilizarse en las implementaciones más comunes de UNIX (sun, ultrix, hp-ux, irix, etc.). Por lo que se tendrá que editar el Makefile y quitar los comentarios en donde se indique el tipo de implementación UNIX que se tiene, para nuestro caso donde dice *hpux* y se cambiará (si es necesario) la ruta donde se almacenarán los deamons originales, esto es, en la línea que dice REAL_DEAMON_DIR=/etc/... quitar los puntos suspensivos y escribir el nombre del subdirectorio a donde se moverán los deamons originales, por ejemplo:

REAL_DEAMON_DIR=/etc/real_deamon_dir

5.Ejecutar: # *make nombre_del_sistema_operativo* (dependiendo la plataforma), en nuestro caso:

make hpux9

6. Cuando el 'make' termine exitosamente se tendrán como resultado 4 programas ejecutables (en Ultrix 5). Estos son: 'try', 'safe-finger', 'try-from' y el principal: 'tcpd'.

El programa 'try' puede ser usado para jugar con las tablas de control de acceso para hosts que se describen posteriormente.

El comando 'safe-finger' debería ser usado cuando se implementan trampas ya que da mayor protección contra basura que los hosts pueden enviar en respuesta a preguntas del finger.

El programa 'try-from' prueba el código de host y username. Se puede correr este desde un comando de shell remoto ('rsh host /some/where/try-from') y éste podrá darse cuenta desde que sistema está siendo llamado.

El programa tcpd puede usarse para monitorear los servicios telnet, finger, ftp, exec, rsh, rlogin, tftp, talk, comsat y otros servicios tcp o udp que tengan su correspondiente en los archivos ejecutables.

7. Crear el subdirectorio /etc/real_deamon_dir.

8. Decidir que servicios se quiere monitorear. Mover los demonios del proveedor correspondientes a una localización especificada por la constante REAL_DEAMON_DIR en el Makefile (subdirectorio anterior). Por ejemplo:

mv fingerd real_deamon_dir

9. Llenar los espacios con copias del tcp wrapper. Esto es, una copia de (o liga a) el programa tcpd para cada servicio que quieras monitorear, por ejemplo para monitorear el uso del servicio finger:

cp tcpd /etc/fingerd

Con otras implementaciones de UNIX los deamons se encuentran en /usr/libexec, /usr/sbin o en /etc, o tienen el prefijo "in." en sus nombres

10. Cambiar los permisos al programa protector a cómo los tenía el proceso original (544), así como el nombre del usuario dueño y del grupo.

11. Probar con el programa try el funcionamiento de los deamons.

12.Definir las polítticas de control de acceso y a partir de ellas generar las listas de control de acceso, éstas se componen de dos archivos: (/etc/hosts/hosts.allow y /etc/hosts/hosts.deny). El control de acceso está deshabilidado cuando no existen dichas listas o se encuentran vacías; en ausencia de tablas de control de acceso, los deamons protectores sólo mantendrán un registro de las conexiones de red hechas al sistema.. En la primer instalación se recomienda probar por unos días sin restricciones de acceso. Los registros del archivo logfile pueden dar una idea de los nombres de los procesos y de los nombres de los hosts que tendrán que incluirse en las listas de acceso.

Normalmente existen dos tipos de políticas de control de acceso: la primera, principalmente cerrada (sólo se permite el acceso a un número limitado de sistemas) y la

segunda, principalmente abierta (permitiendo el acceso a todo el mundo excepto a un número límitado de sistemas o usuarios problemáticos). De acuerdo a nuestra situación se elegirá el modelo que convenga más. También se pueden implementar políticas mezcladas.

Un ejemplo de lista de acceso puede ser:

para el hosts.allow:

ALL:LOCAL, .cecafi.unam.mx que implica que pueden ejecutar todos los servicios sólo los que se encuentren en la red local.

13. Después de las políticas se crean los archivos hosts.allow y hosts.deny y se prueba el sistema.

14. Para el caso en particular se crearon archivos hosts.allow y hosts.deny que se habilitan únicamente en la noche (9:00 pm) y se deshabilitan en la mañana (7:00) así como el fin de semana, evitan el acceso remoto y únicamente se permite el acceso local.El orden en que se controla el accesos se toma: primero el niegue de servicio y después se atienden los permisos.El archivo hosts.deny queda:

ALL:ALL

es decir a todos se le quitan todos los permisos de ejecutar servicios; y el hosts.allow:

ALL:LOCAL, .cecafi.unam.mx

que permite a los usuarios locales ejecutar todos los servicios.

15. Además se crearon archivos para ejecutarse con el *crontab*, es decir que permite ejecutar de acuerdo con un horario un programa. Este programa realiza el cambio de las listas de acceso para permitir dentro del horario establecido el acceso a todos los servicios por todos los usuarios.

16. El último punto es realizar pruebas y verificar que las políticas funcionen cómo se esperaba, si entorpecen o ayudan al desempeño de las labores de los usuarios y administradores.

Después de la implementación del mismo se alcanzó un control de qué cuentas y en qué horarios accesan nuestros servicios. Por medio de las tablas de acceso se definen quiénes, por necesidad explícita pueden conectarse al sistema en días o en horarios en que no existe vigilancia, y con esto se podría hacer un seguimiento en caso de haber sufrido un ataque.

5.3.4.2 Seguridad en UNIX

Si nuestro sistema trabaja bajo ambiente UNIX, es implícito que tiene problemas de seguridad, por lo que se debe trabajar arduamente para instalar todas las barreras que sean posibles, corregir todos aquellos agujeros propios del sistema. Al respecto existe extensa bibliografía, grupos de discusión, información de acceso gratuito en la Internet.

Las soluciones dependerán de muchos factores, y puede ir desde el control de permisos de archivos, hasta manejo de encriptado, instalación de firewalls, dispositivos de otorgamiento de contraseñas de una sola vez, etc.

En el Centro de Cálculo se estableció una política de selección de contraseñas basada en los principales ataques mencionados en otros capítulos, se diseño, generó e imprimió una serie de recomendaciones a los usuarios para elegir y manejar adecuadamente su contraseña, pero, como es obvio que esto no es suficiente, se instaló el sistema Passwd+, mediante el cual se evita que el usuario cambie su contraseña por una fácilmente adivinable.

Para mejorar la seguridad en UNIX y facilitar la administración se instaló COPS.

Configuración e instalación sencilla del COPS

Hemos mencionado que es básico descubrir nuestros agujeros de seguridad antes de que otros lo hagan, por lo que se instaló y configuro el sistema COPS en los diferentes hosts. Por medio de éste se obtienen reportes semanales que informan los posibles agujeros de seguridad de nuestro sistema (además de los problemas de contraseñas ya tratados).

Estos reportes se entregan oportunamente a los administradores quiénes se encargan de subsanar aquellos problemas que así lo requieran, sobre este punto vale la pena recordar que no siempre un error reportado por COPS puede implicar un problema de seguridad en nuestro sistema.

Mediante una de las utilerías que se instalaron de COPS se intenta romper la contraseña de cada uno de los usuarios, y en caso de que ésta se encuentre se le envía un mensaje al usuario para que la cambie con un tiempo de gracia, en caso de que el usuario no cambie su contraseña dentro del tiempo establecido la cuenta es deshabilitada hasta que el usuario acuda con el administrador para habilitarla nuevamente y asignar una contraseña seguro.

Además de la verificación mediante los dos paquetes mencionados, se estableció una fecha de expiración de contraseña (mediante el uso del archivo *passwd*).

Las cuentas de visita (*guest*) simplemente no existen.

En el caso de las cuentas que se asignan para cursos, se estableció que fueran borradas en cuanto se terminara el período de curso.

En cuanto al acceso del super usuario se limitó únicamente a la consola.

Anteriormente al dar de alta una cuenta se creaba con los permisos habilitados para todo el mundo (lectura, escritura y ejecución), por lo que se cambió esta convención y se crea sólo con los permisos para el dueño. Esta seguridad se instaló por que a pesar de que cada usuario puede cambiar los permisos una gran parte no lo hace. Sería fácil pensar que es responsabilidad del usuario cuidar su información, pero no siempre el usuario está consciente de eso y no está de más que por nuestro propio bien, el sistema desde un principio ofrezca algunas barreras.

Y ahora que tocamos el punto de los usuarios y su manejo de la seguridad, cabe señalar que tal vez una de los más importantes acciones ha sido el dar cursos de manera extensiva y casi gratuita a todo miembro de la comunidad de la Facultad de Ingeniería que así lo desee, de esta manera se difunde la información y se puede aprovechar con mayor énfasis los recursos, además de que se le da al alumnado una herramienta poderosísima. Se difunden las barreras que el propio usuario puede crear, a través de un conocimiento más profundo del ambiente y del sistema en el que trabaja. Otro logro a través de los cursos es la retroalimentación como un medio para corregir fallas.

5.3.4.3 Recursos Humanos

El factor humano es de los más importantes, debe tenerse cuidado no sólo con los usuarios, sino con el personal que labora en el Centro, y con el que dejo de laborar.

Algunos exbecarios o exadministradores rencorosos pueden representar un serio problema. Muchos de ellos pueden conocer vulnerabilidades de los que no estemos al tanto aún y aprovecharse de ellas para hacer daño. La selección, manejo y monitoreo del personal deben realizarse minuciosamente con políticas estrictas.

Negar el servicio a aquellas personas que se sabe problemáticas en ese aspecto no debe dudarse. Para prevenir este tipo de situaciones es mejor seleccionar al personal con sumo cuidado.

En cualquier caso un punto muy importante es no olvidar tener un buen manejo de respaldos para que a pesar de cualquier daño, se pueda recuperar la información lo más cercana posible a su último estado. Recientemente se implementó la política de respaldos semanales, mensuales y semestrales. Además de los respaldos no se debe olvidar que el almacenaje de éstos es básico, esto es, que no se encuentren en el mismo lugar en el que se encuentra el equipo y no al alcance de cualquiera, pero sí que los administradores tengan un fácil acceso a ellos en caso de necesidad.

Un agradable ambiente de trabajo en el cual se cumplan las expectativas de los que laboran, alcanzando de manera conjunta las metas es básico. Nos atrevemos a afirmar que el factor humano es el primero en cuidarse. En el Centro se dió la reestructuración de la administración y manejo del equipo, ahora sí, con un concepto de trabajo en equipo y de SEGURIDAD, esto es, el propósito era y es evitar que la responsabilidad recaiga en una sola persona, que no importa el daño que se intente hacer los datos de importancia siempre sean recuperables así como ofrecer a los alumnos y maestros un servicio de calidad que les garantice, en la medida de las posibilidades y conveniencias del centro, que sus datos se encuentran a salvo.

En esta parte se ha conformado un grupo de trabajo con objetivos bien planteados, con programas de trabajo y estudio respecto a administración y seguridad, ejemplo de ello se encuentra en el Apéndice C en el que se muestran los lineamientos para los administradores. Además, un punto que cabe resaltar es el hecho de que se siguió un programa de capacitación, desde UNIX, programación para UNIX, administración del equipo, uso de las herramientas y profundización en el aspecto de seguridad. Este proceso

es continuo y los nuevos becarios son entrenados por los anteriores de manera que no se pierda la liga de conocimientos y experiencia adquirida.

5.3.4.4 Seguridad con Dragones

En el Centro de Cálculo no se implementó solución para todos estos problemas, sin embargo con el TCP-Wrapper se reinstalaron demonios que monitorean los accesos por medio de Telnet, ftp, y el uso del ping (siendo barreras, como ya se mencionó, en ciertos horarios).

5.3.5 Etapa de Mantenimiento

Una vez implantado el Modelo de Seguridad, es responsabilidad de los encargados de los respectivos equipos involucrados en la red, realizar un plan de mantenimiento al modelo.

En la etapa de mantenimiento se debe considerar la implantación de líneas futuras, y el monitoreo de los programas instalados y de los reportes generados, para por medio de esta información ir depurando los problemas de seguridad que se vayan presentando.

Debido a que nosotros no somos ya parte del equipo de trabajo de la Unidad es conveniente que los actuales administradores tomen a su cargo la retroalimentación en cuanto a los efectos de las medidas y determinen cuáles de las líneas futuras son aplicables y en su caso lo hagan. De la misma manera, si resulta que alguna medida es obsoleta o existen opciones que den mejores resultados sustituyan dichas soluciones por las nuevas.

5.3.6 Líneas Futuras

En este punto queremos mencionar las líneas futuras para la Unidad, pero también hacer algunos comentarios acerca de soluciones que pueden llegar a ser útiles en un futuro cercano para esta institución o para otras donde la información es aún más valiosa. Seguramente los métodos cambian, o son más exhaustivos.

Ahora bien, las soluciones que podrían proponerse serían muy variadas y a diferentes niveles pero al analizarlas es importante no olvidar las actividades que el Centro realiza para no caer en la paranoia y terminar poniendo barreras a los usuarios del equipo o tratando de implementar soluciones sofisticadas y fuera del alcance del Centro.

5.3.6.1 Planta Física

Definitivamente el buen diseño, construcción y localización del centro de cómputo es primordial. El acceso debe ser restringido y, dependiendo la empresa, se implementarán diferentes métodos de autenticación de usuarios como control de acceso al lugar, esto es desde unas credenciales magnéticas pasando por los diferentes tókens mencionados hasta la autenticación biométrica.

Una posible solución es la implantación de puertas de seguridad mediante las cuales sólo tengan acceso los usuarios autorizados, credenciales con fotografía magnéticas podrían ser una buena opción no muy lejos de la realidad del Centro.

También deben instalarse alarmas de seguridad y equipo contra incendio. Debe llevarse un control preciso del equipo, software y manuales, saber dónde está y bajo qué responsabilidad se encuentra, de esa manera se evitarán pérdidas o desapariciones misteriosas cuando algún elemento del centro deje de laborar ahí

5.3.6.2 Topología Física

Cuando hablamos del Centro no mencionamos la intervención de las líneas ya que seguramente un atacante no intentaría intervenir una línea de la Universidad que no contiene información de un alto costo. El precio de intervenir una línea debe ir en proporción al valor que la información obtenida provee.

Como en otros casos la información llega a ser extremadamente valiosa es mejor seleccionar equipos y configuraciones exhaustivamente.

En el caso de los cables la fibra óptica sería la mejor selección, además de instalación de equipo que detecte cambios de nivel para percepción de intervenciones en la línea.

La selección de equipo no deja de ser sólo una barrera, por lo que la respuesta al problema de intervención de líneas siempre será el encriptado, con las condiciones que ya se manejaron anteriormente.

Puede parecer redundante, pero nunca debe dejarse a un lado a que la otra parte del CECAFI (la USECAFI), es quien mantuvo en la división el ruteador del cual salen los cables por los que se conecta el centro de cómputo a la Internet, resulta fuera de su alcance vigilar su buen funcionamiento y en varias ocasiones se desconectan los cables haciendo que falle el equipo, por lo que se sugiere que la USCA tenga su propia conexión a la redUNAM.

5.3.6.3 Firewalls y Encriptado de la Información

Ahora bien, el que en el Centro de Cálculo no sea justificable una inversión para evitar los ataques TCP/IP no implica que en otros casos o, en un futuro en el que la información adquiera un mayor valor, no lo sea.

En otras instituciones en las que se maneja información confidencial como las gubernamentales o bancarias es importante que se implementen todas las barreras posibles, para este conjunto de problemas en particular las soluciones idóneas serían implementar un firewall y el manejo de encriptado.

Además deben implementarse algoritmos de autenticación, ya propiamente dentro del sistema, para asegurar que la información que recibimos venga de quién dice provenir y llegue a quien deba llegar. En este rubro incluimos un manejo de contraseñas seguro, el que puede basarse en smartcards o tókens, por poner un ejemplo. También como

mecanismo de autenticación se propone Kerberos, aunque no necesariamente tenemos que implementar este sistema, sino tomar su filosofía para realizar nuestro propio esquema de autenticación, exclusivo a nuestras necesidades y recursos.

Dependiendo del costo de la información y del tiempo en que el conocerla pueda implicar un daño, se deberá establecer un esquema de encriptado adecuado (un RSA es muy recomendable). Además de lo complicado del algoritmo debemos considerar el tipo de equipo para manejarlo, ya que el encriptado implica memoria y velocidad de ejecución que pueden entorpecer nuestra funcionalidad. El equipo es una parte básica, pero también lo es el personal capacitado para darle mantenimiento al algoritmo así como para implementarlo adecuadamente.

Un punto que no debemos olvidar cuando hablemos de encriptado es el cuidado que demos a las llaves, debe recalcarse la importancia de este manejo a nuestros usuarios, pues de eso depende la seguridad de su información. El sistema debe ser lo suficientemente bueno para que, aún habiendo violado la seguridad de uno de nuestros usuarios (es decir que tengan en sus manos su(s) llave(s)), no puedan generar un ataque en el tiempo suficiente como para hacer daño al sistema o a otro usuario, o bien que no sean detectados.

Una solución que no podemos dejar de pasar por alto es el uso de Firewalls, para quienes tienen los recursos, es una barrera muy importante. Pero no se debe confiar totalmente en ella, se sabe que es posible bajo ciertas circunstancias engañarlo y traspasar la pared, recordemos que un firewall no puede a ciencia cierta controlar el contenido de los mensajes, sólo filtrarlos y en su caso autenticarlos, por lo mismo no debe dejársele toda la responsabilidad.

Para aquellos que deseen instalar algunas barreras pero no tengan los recursos que implica un firewall, se les recomendaría programar los enrutadores como barrera hasta el límite de las posibilidades del tipo de ruteador con el que se trabaje.

5.3.6.4 Seguridad con Dragones

De acuerdo a como vayan evolucionando las necesidades y actividades del Centro, será importante restringir el uso del ftp anónimo, de manera en que no se tenga acceso, sino únicamente a lo que se debe de tener sin riesgo a daño alguno, una opción segura es que sólo un host ofrezca este servicio.

Si se llega a considerar que la información que proporcionan otros protocolos como el finger implican un peligro considerable, puede deshabilitarse o configurarse para dar sólo los datos que no sean relevantes. Como se mencionó esto es importante debido al ataque a contraseñas, pero si tenemos implementada una política de contraseñas confiable tal vez estos protocolos no impliquen un riesgo fuerte.

El correo es un gran problema, pero indudablemente es un muy importante servicio, sino el que más. La sugerencia aquí sería hacer notar a los usuarios la importancia de no transmitir por este medio información extremadamente peligrosa,

como las contraseñas (asombrosamente muchos administradores envían sus contraseñas por este medio !¡).

5.3.6.5 Seguridad en los Servicios de Información

En el caso del Centro el uso de éste tipo de utilerías es más bien benéfico debido al acceso que le brinda a los alumnos hacia la información. Los daños que pueden causar se pueden corregir si existe un adecuado manejo de respaldos en caso de que el daño llegará a extremos. De cualquier manera se sugiere que sólo un host maneje este tipo de utilerías, un host en el que no se pueda comprometer información.

Como el Centro no cuenta con un firewall puede sugerirse que, ya que es una institución que fomenta la investigación, se implemente alguno de los que se proporcionan gratuitamente en la Internet, se haga pruebas con él, de manera que se da un gran avance en la materia, fomentando el estudio de nuevas tecnologías y se puede llegar a mejorar el control de la seguridad en un muy alto porcentaje.

La última opción es esperar a que se liberen versiones más seguras de los protocolos que manejan los WEBS. Mientras tanto el servicio vale más que un posible daño, tomando las debidas precauciones.

En aquellos casos en que se tenga un firewall (sobre todo comerciales) existen posibilidades para controlar los manejos de los protocolos, y por lo tanto poner una barrera más para estos programas. Sin embargo, ni esto, ni esperar a que se liberen versiones más seguras, es una buena solución para aquellos en los que su información sea demasiado valiosa como para arriesgarla a ser destruida por programas que pueden saltar aún las barreras de un firewall. En estos casos es importante considerar en qué hosts se permite correr estos tipos de aplicaciones.

Existen otras soluciones, instalación de otro tipo de software o bien la creación de shell scripts que realicen el monitoreo de nuestras aplicaciones, de los usuarios, de posibles intromisiones, de puntos de inseguridad. Lo importante es que el equipo de trabajo del Centro de Cálculo sigue trabajando en ellas y tratando de llevarlas a cabo.

Una solución adecuada, para quienes tengan los recursos, es el manejo de contraseñas de una sola vez. En el mercado existe una gran cantidad de dispositivos a diferentes costos y con diferentes niveles de seguridad que se pueden adecuar a las necesidades. De esta manera aseguramos que sólo pueda ser leído el correo por quién corresponda y, como algunos de estos esquemas incluyen encriptado, el contenido correspondiente sólo puede ser conocido por aquél que posee el dispositivo.

CAPÍTULO 6

Conclusiones

Estamos seguras de que existen otras alternativas y que para cada caso que se mencionó se podría ser más específico, pero para cada problema existen volúmenes enteros de información, al final se da una bibliografía para quien desee ahondar más en cada tema.

Además de las opciones dadas existen otra serie de puntos a considerar cuando tratamos de hacer nuestro sistema seguro.

1. No estamos solos, contamos con la Internet y no sólo tenemos que defendernos de ella sino aprovecharla. Existen cantidades inmensas de información del tema; archivos con soluciones a problemas que otros han tenido; software, mucho de uso gratuito, para mejorar nuestras configuraciones, o bien para rellenar agujeros de seguridad; podemos suscribirnos a los grupos de discusión en los que usuarios de todo el mundo aportan ideas, hacen preguntas y sugieren soluciones. Además, ya existe en el mercado gran cantidad de bibliografía sobre varios temas de interés.

2. Nuestro sistema nunca es completamente seguro. Por muy buenas que sean nuestras políticas y las barreras que hayamos instalado, siempre será posible romperlas, pero nosotros debemos tratar de hacerlo antes que otros, estar alertas, monitorear y solucionar lo antes posible los defectos.

3. No dejemos a los usuarios la responsabilidad de la seguridad, está probado que es muy poco confiable hacerlo. Si nosotros no cuidamos la seguridad de nuestro equipo y de nuestra información, nadie lo hará.

4. Lo importante es mantener la visión de un sistema de seguridad, continuar investigando y trabajando, sobre todo manteniendo un ambiente de cordialidad, el factor humano puede ser el más importante.

5. La documentación es un factor vital para el mantenimiento y el ciclo de la seguridad, su buen realizamiento y continuación nos pueden ahorrar muchos dolores de cabeza posteriores.

6. No descuidemos ninguna puerta, ni passwords (y queremos recalcar este punto por que es la principal entrada y más problemática en cualquier sistema), ni acceso al equipo, ni permisos sobre archivos, agujeros en nuestros sistemas operativos, en los protocolos de comunicación o en los mismos sistemas que implementemos.

7. Nunca descuidemos los respaldos y verificar que estos se encuentren en buenas condiciones.

8. Implementemos medidas de seguridad, pero sin olvidar la funcionalidad de nuestro sistema, el costo de proteger la información no debe ser mayor que el de la misma información. Tengamos cuidado de que nuestras barreras entorpezcan las labores de nuestros usuarios.

9. El factor humano es en nuestra opinión, el más importante. Su buen manejo, preparación, cuidado, vigilancia son básicos para crear un ambiente de seguridad, de la misma manera es importante crear un ambiente de trabajo cordial en el que usuarios y administradores trabajen en armonía. La selección del adecuado personal de seguridad muy frecuentemente significará el éxito o el fracaso de un programa de seguridad.

10. No olvidemos que no hay enemigo pequeño.Y si recordamos que los principales hackers se encuentran entre el personal y los alumnos, debemos difundir los conocimientos y la conciencia de que el equipo y el software está para beneficio de la comunidad, para despertar la curiosidad, para encontrar nuevos caminos, y porqué no, agujeros; pero para resolverlos en beneficio de todos, no para aprovecharlos.

11. El entrenamiento y la adecuada información habilitará a usuarios y administradores a entender, respetar y defender la seguridad.

Podría pensarse que uno de los propósitos al implementar esquemas de seguridad es lograr clasificarlo dentro de algún criterio. Para el Centro de Cálculo no tiene sentido, por lo que quedaría en D (sistemas sin evaluar). Los costos de evaluar son muy altos, además de que algunas de las barreras a implementar son difíciles y de un valor mayor al de la información manejada.

Pero no porque la Unidad de Servicios de Cómputo no maneje información confidencial, no significa que no pueda hacerse daño, si permitimos un acceso fácil o que pueda darle grandes privilegios a un atacante implica que estamos dando una puerta fácil para quienes intentan accesar otras entidades de la Universidad, cómo las bases de datos de los historiales académicos de los alumnos, o de las inscripciones de los alumnos. Nuestro sistema puede ser una barrera o una invitación a atacar, según lo implementemos.

No necesitamos evaluarlo para difundir su clasificación, no estamos vendiendo ningún esquema. Necesitamos asegurar la información bajo nuestras necesidades y recursos.

Pero en otros casos, más que para difundir su clasificación, evaluar, aunque no con fines de clasificación, nos sirve para asegurarnos que las barreras hayan sido implementadas correctamente, conforme a nuestros objetivos.

Los puntos especificados para uno o más criterios nos pueden servir de guías, aunque no todos los apliquemos. Elegiremos los que nuestros sistema requiera y pueda emplear (recordemos que algunas barreras de seguridad pueden convertirse en obstáculos para nuestras labores).

Hubo algunos puntos sin tratar o que no se trataron a profundidad. Uno de los temas no contemplados fue el de los Caballos de Troya, pero este y otros temas podrían abarcar una tesis completa.

Respecto al trabajo en el la Unidad de Servicios de cómputo es importante recordar que dicha Unidad (antes CECAFI) en conjunción con la Facultad de Ingeniería ha sido fuente de profesionales competitivos y bien preparados para enfrentar con mejores herramientas y bases más fuertes cualquier medio de trabajo. Una muestra de ello es la gran cantidad de profesionales, no sólo de la carrera de Computación, que actualmente se encuentran desempeñando importantes cargos en distintas empresas, esta es la razón por la cual este trabajo puede representar para sus actuales miembros una muy completa base teórica que les actualice y los ponga un paso adelante respecto a un tema que más que estar de moda es indispensable para cualquier institución seria que se precie de serlo. Pero este trabajo no sólo es útil para el Centro y sus miembros, sino para todos los elementos de nuestra Facultad (¿acaso no se encuentra una red de computadoras en casi cualquier área de ella, y no es su información suficientemente valiosa y confidencial?).

En un sólo volumen pueden encontrar información actualizada y vital para tener una imagen global no superficial pero tampoco demasiado técnica sobre el problema de seguridad. Además de la adecuada conceptualización del problema no sólo el Centro de Cálculo sino cualquier centro de cómputo encontrará un esquema de seguridad básico.

El sistema en el Centro de Cálculo aún no es lo completamente seguro (y ningún sistema lo es), pero ya se ha hecho una buena parte, tal vez no toda la información respecto al Centro esté completa ya que día a día el grupo de personas que ahí laboran trabajan al respecto y es difícil mantener actualizada esta información, al menos para nosotras que ya no tenemos un contacto directo con ellos y que por razones de seguridad es mejor que no lo tengamos. Las autoras de esta tesis intentan aportar algunas sugerencias, pero no son todas y tal vez no sean completamente aplicables, eso se decidirá conforme a las circunstancias y objetivos actuales.

El punto importante a recalcar en sus avances, es que actualmente ya existe un grupo de trabajo bien conformado, en el cual se intercambian ideas, problemas y soluciones halladas a través del trabajo, la investigación y de encarar día a día los problemas.

Durante el desarrollo de esta tesis se marcó el inicio de una nueva visión de seguridad. Podemos resumir los logros en la siguiente lista:

- Se instaló software de seguridad para evitar agujeros en las configuraciones (COPS, TCP Wrapper, Passwd+).
- Se establecieron políticas tanto de acceso físico como lógico (mejores credenciales y políticas de passwords).
- Definición y empleo y de políticas de administración más seguras.
- Establecimiento de líneas de comunicación para retroalimentación mediante información escrita y electrónica a usuarios acerca de mejoras en el sistema, políticas, derechos, obligaciones y penalizaciones.
- Mejoras en la documentación mediante bitácoras en las que se almacena información sobre agujeros, ataques y soluciones implementadas.
- Alta capacitación y difusión de los conocimientos mediante cursos impartidos por el personal de la Unidad a toda la comunidad.
- Mayor capacitación del personal por medio de cursos, uso de manuales, retroalimentación con otros administradores, recursos ofrecidos en la Internet, usuarios y el mismo personal.

Pero no sólo son importantes las herramientas configuradas, los agujeros subsanados, y los demás puntos citados, lo más importante es haber sembrado las bases de una conciencia de seguridad, mayormente en el personal (jefes, administradores y becarios) y en menor grado en usuarios, éste último será un trabajo bastante más largo y difícil para la Unidad y sus miembros.

De esta manera la Unidad sigue cumpliendo sus objetivos, ofreciendo un servicio confiable, eficiente, actualizado y con tendencias positivas a una institución que lo merece: la Facultad de Ingeniería.

En el centro de cálculo, la Facultad de Ingeniería o cualquier institución que emplee recursos computacionales debemos trabajar pensando en la seguridad, que más que ser otro ladrillo es un modo de vida.

No esperemos ni basemos nuestros pasos, en el hecho de que se genere una conciencia y sobre todo, un compromiso, cuidemos nuestra seguridad sino nadie más lo hará.

Anexos

Los archivos anexos de este trabajo se proporcionan en diskette, éstos son:

- Programas de Instalación del TCP Wrapper.
- *Warnings* de COPS.
- Documento de instalación de COPS.
- Documento sobre seguridad en TCP/IP.
- Documento sobre seguridad en WWW.
- Documento de seguridad en UNIX.
- Reportes de COPS de las máquinas donde fue instalado mostrando los avances en la configuración.

Apéndice A

TCP/IP -Transport Control Protocol/Internet Protocol[1]

1. Conceptos.

1.1 Internet Protocol Suite (TCP/IP)

TCP/IP está organizado en cuatro capas, que están contempladas sobre una capa cinco- el hardware de la red. Las especificaciones de hardware no están construidas dentro del modelo TCP/IP.

El siguiente diagrama muestra las capas del modelo y varios de los protocolos que componen las diferentes capas.

Aplicación. La capa de aplicación es la capa de más alto nivel. Esta capa provee servicios a usuarios finales, tales como transferencia de archivos, correo electrónico y acceso a terminales remotas. Los programas de aplicación (algunos tienen definido su propio protocolo) interactúan con la capa de transporte. Estos tienen una alternativa entre varios protocolos de transporte, dependiendo del tipo de transporte que se requiera.

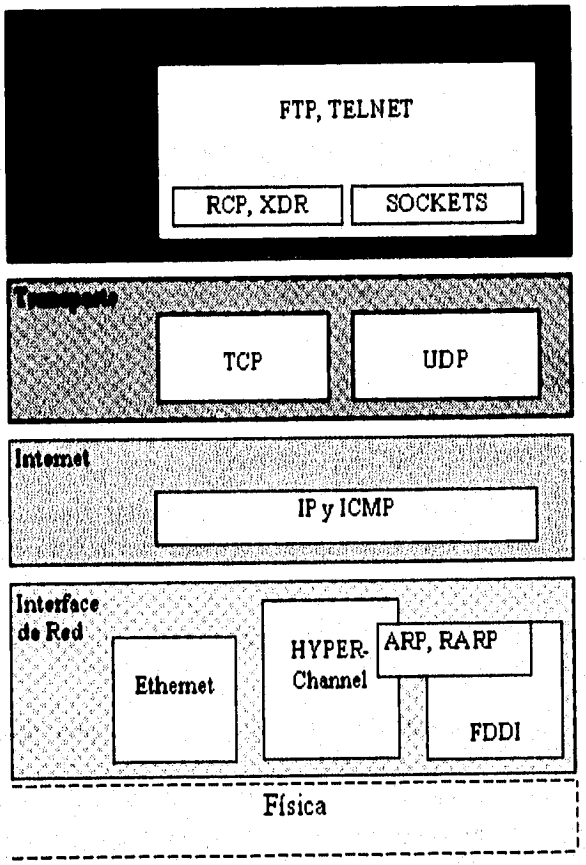
Transporte. El principal objetivo de la capa de transporte es proveer una comunicación punto-a-punto entre aplicaciones. Los protocolos de transporte usan el servicio de liberación de paquetes que provee la capa de Internet.

Internet. La capa de internet provee servicio de liberación de paquetes desde una máquina a otra. La rehabilitación de los paquetes no se lleva a cabo en esta capa, de esto se encargan los protocolos de más alto nivel (transporte o aplicación).

---

[1]Cray Research Inc. *Introduction to TCP/IP self study.* April 1992 Revision B.

Interfase de Red. La capa de "Network Interfase" (algunas veces referida como "enlace de datos") acepta datagramas desde la capa Internet y los envía físicamente. Un módulo de interfase de red es a menudo un manejador de un dispositivo de hardware específico. La capa de "Network Interfase" consta de varios módulos.



1.2 Modelo OSI.

1.2.1 Open Systems Interconections/Internetional Standar Organization -OSI/ISO

La Organización Internacional de Estándares (ISO), es una organización mundial que tiene como objetivo definir una nueva arquitectura de redes estratificadas conocida como el modelo OSI (Open Systems Interconections). Esta nueva arquitectura intenta servir como base para los nuevos protocolos que eliminaran algunas de las limitaciones inherentes a TCP/IP, tales como el máximo tamaño de un mensaje. Se espera que en los próximos 10 años, los protocolos basados en OSI, vayan reemplazando gradualmente a TCP/IP. En la actualidad el modelo OSI, todavía no se ha logrado implementar, y está siendo sometido a pruebas.

El modelo OSI está siendo definido por varios comités internacionales que están contemplados dentro de ISO, los cuales están compuestos por representantes de gobernación, de Universidades y Corporaciones. Su objetivo es diseñar una arquitectura de red que eliminará las limitaciones inherentes a los protocolos que existen hoy en día, haciendo posible ser eficaz en un futuro. Está siendo diseñado para ser fácilmente expansible y modular, con varias funciones de red divididas dentro las capas que componen el modelo.

Modelo OSI.

7. Aplicación.	Realiza servicios de alto nivel para usuarios finales, tales como transferencia e archivos, transferencia de correo electrónico, y acceso a terminales remotas.
6. Presentación.	Realiza las funciones comunes de manipulación de datos, tales como compresión, encriptación, translación de código ASCII/EBCDIC, etc. Se asegura que todos los usuarios envíen información por la red en forma común. Efectúa cambios de formato, entre formato de red a formato host, siempre que sea necesario.
5. Sesión.	Controla la conexión de los procesos de comunicación, secuenciamiento de datos y el enlace entre un protocolo y una aplicación específica.
4. Transporte.	Controla la conexión punto-a-punto entre hosts, por ejemplo: Control de flujo, ajuste de velocidad de transmisión, corrección de errores y retransmisión.
3. Red.	Controla el enrutamiento de información a través de la red, controla el congestionamiento a través de la red, realiza la fragmentación y el reenzamble de frames.
2. Enlace.	Libera frames de una máquina a otra, durante lo cual puede incluir el chequeo de hardware, provee multiplexaje para varias líneas lógicas sobre una sola línea física. Maneja el hardware de la red.
1. Físico.	El hardware de los componentes físicos de la red, la conectividad entre host.

1.3 TCP/IP y el modelo OSI.

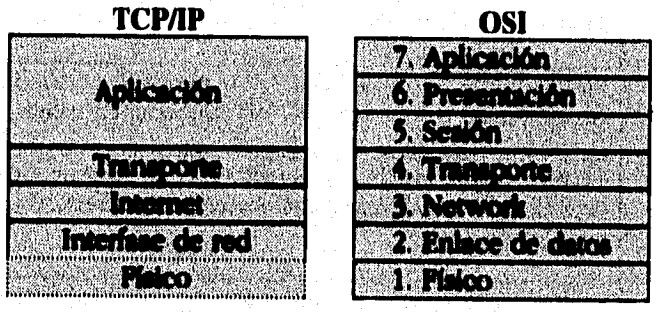
Aún cuando los esquemas de las capas son diferentes, una comparación entre la familia de protocolos TCP/IP y el modelo OSI es válida. Esta comparación puede ayudarnos a entender la función que realizan varios protocolos de TCP/IP.

El modelo OSI está diseñado para detectar y manipular errores en todos los niveles del modelo. TCP/IP realiza una detección de errores específicos en varias capas, y en algunas capas realiza un mínimo procedimiento de detección de errores.

El protocolo TCP/IP pone más énfasis en tener la aplicación que decida la mejor forma de realizar la detección de errores y manipularlos.

En el modelo OSI la mayor parte de la inteligencia de la red depende de la red misma. Una red OSI es un sistema complejo e independiente. Una red TCP/IP requiere de host inteligentes para participar en cada fase de la red. Una red Internet TCP/IP es una red con host inteligentes.

También existen diferencias desde la previa presentación de cada uno de los esquemas: OSI fue y ha sido diseñado, la historia de TCP/IP es la de una evolución - los protocolos fueron construidos y modificados conforme se fueron resolviendo problemas y se fue aprendiendo de los mismos dinámicamente. El proceso de diseño contra el proceso de evolución tiene como resultado una diferencia de filosofía entre OSI y TCP/IP. La filosofía de TCP/IP es "hacerlo simple".



2. Direccionamiento

2.1 Direccionamiento Físico.

Toda la comunicación en la red ocurre en el medio físico o hardware. Cada medio físico es diferente y tiene su propia forma de direccionar los nodos que están interconectados en la red.

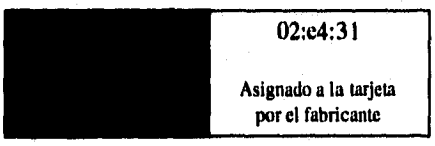
El direccionamiento físico está dividido en dos tipos:

1) Largo, fijo, única dirección física en todo el mundo

2) Corto, cambiable, dirección física asignada localmente

1) Un ejemplo de este tipo es Ethernet, el cual utiliza una dirección de 48 bits (6 bytes). Los primeros 3 bytes son asignados por el fabricante a través de un administrador global (IEEE), los tres últimos bytes son asignados por el fabricante. Esta dirección es

puesta en la interfase y es única dentro del mundo Ethernet. Esta dirección es representada por seis pares de dígitos hexadecimales separados por comas (08:00:20:01e6:31). Si el primer dígito del par es 0 pude ser omitido.



Ethernet

2) El "Network System Corporation HYPERchannel, es un ejemplo de este tipo de direccionamiento físico. HYPERchannel utiliza una dirección de 16 bits.

La comunicación física en la red utiliza estos tipos de direccionamiento para liberar los mensajes de un nodo a otro.

2.1.1 Comunicación en el medio físico.

2.1.1.1 Encapsulamiento de datos.

Así como los medios físicos tienen diferentes formas de direccionar los nodos en la red, también tiene diferentes formas para encapsular los datos del usuario que se liberan.

Generalmente los datos son encapsulados en un FRAME o paquete, con información acerca del direccionamiento en el encabezado (headers) y la información restante (trailers). El encabezado y el resto de la información realiza las funciones de un sobre de correo. Estos encierran los datos (carta) y contienen información de direccionamiento y liberan instrucciones que son utilizadas por la interfase de la red para entregar los datos.

El frame Ethernet.

64 bits	48 bits	48 bits	16 bits	368-1200 bits	32 bits
Preambulo	Destination Address	Source Address	Frame Type	Frame Data	CRC

Campo	**Descripción**
Preamble.	6 4 bits de 1 y 0 alternados para ayudar a recibir la sintonía de nodos.
Destinations Address.	Los 48 bits de la dirección de Ethernet del nodo que va a recibir este frame
Suorce Address	Los 48 bits de la dirección Ethernet del nodo que envió el frame.
Frame Type.	Un entero de 16 bits que identifica el tipo de datos contenidos en el frame. El protocolo TCP/IP utiliza este campo para distinguir entre varios protocolos.
Frame Data.	Los datos del usuarios que están siendo transportados actualmente. EL MTU (Unida Máxima de Transmisión) para el medio es de 1500 bytes; este es el monto máximo de datos del usuario que pueden ser transportados dentro de un frame.
CRC.	Los 32 bits de CRC (Cheque de Redundancia Cíclica) es un número calculado

como una función de los datos en el frame. Este calculo es realizado por el emisor y el receptor del frame y usado para detectar errores de transmisión

2.2 Dirección Internet.

Los diferentes medios físicos requieren de diferentes formas de direccionamientos y utilizan diferentes tipos de frames para empaquetar los datos. Esto dificulta escribir protocolos de comunicación que puedan trabajar con cada tipo de hardware. Los protocolos de arquitecturas por capas, manipulan este problema para aislar los medios específicos y detallarlos en las capas más bajas. Esto permite a las capas medias y superiores de la red ver a las capas más bajas como una larga red lógica o Internet.

Para comunicar los nodos de la red en el nivel lógico en vez del nivel físico, en TCP/IP se ha definido una Dirección Internet, que es independiente del medio físico.

La dirección Internet es una construcción lógica que está implementada completamente en software.

La dirección Internet es un entero de 32 bits, representado por una secuencia de cuatro números decimales separados por puntos. Una dirección Internet asignada a una Interfase de red identifica a un solo host. Si un host tiene mas de una interfase de red (tarjetas de red), debe tener asignada una dirección IP por cada interfase de red. La dirección IP consta de dos campos: *netid y hostid*

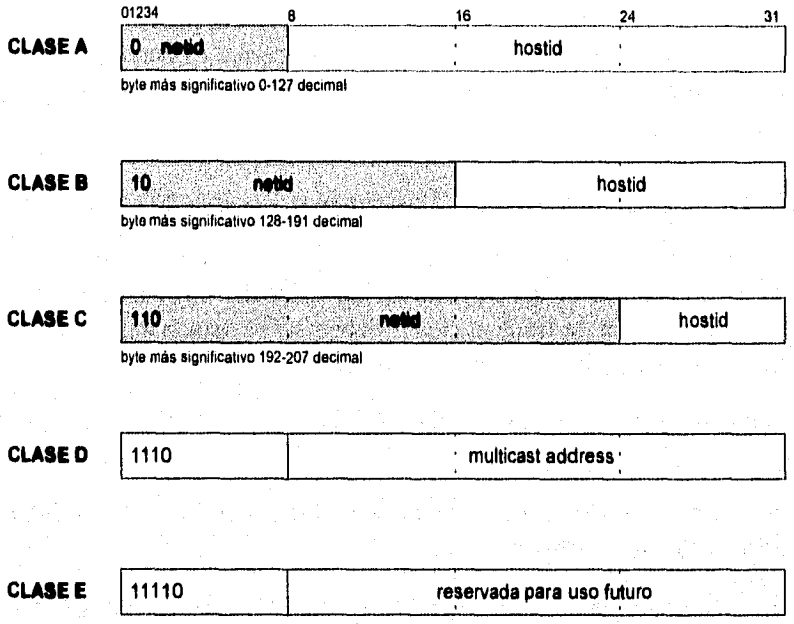
2.2.1 Clasificación de la Dirección IP.

Clase A. El primer byte identifica el numero de red (netid), y los siguientes tres bytes identifican la interfase de host (hostid). La clase A se distingue por un 0 en el primer bit del primer byte. La clase A puede tener 126 subredes, cada una conteniendo un máximo de 16,777,215 hosts.

Clase B. Los dos primeros bytes identifican el hostid y los dos siguientes el netid. La clase B se distingue por la combinación 1-0 en los dos primeros bits de los dos primeros bytes. Esto resulta que las subredes estén numeradas del 128.1 al 191.254 cada una con máximo cada una de 65, 535 hosts.

Clase C. Los tres primeros bytes identifican el netid y el último byte el hostid. La clase C se distingue por la combinación 1-1-0 en los primeros tres bits del primer byte. Esta red tiene la capacidad de soportar 16,000,000 de subredes, numeradas de la 192.0.1 a la 223.255.254, cada una conteniendo un máximo de 254 hosts.

Clase E. La clase E está reservada para un uso futuro.



Direcciones reservadas.

127.hostid Esta dirección se conoce como *localhost o loopback*. Esta dirección es definida en todos los host y cuando es usada, indica un loopback. Los datos no son enviados en la red, pero van a través de la pila del protocolo y regresan al mismo host.

netid.255 Una dirección con la parte de hostid puesta en 1, es una dirección de broadcast, donde el mensaje es enviado a todos los hosts de la red.

Clases de Direcciones	Ejemplos de dirección broadcast
A	89.255.255.255
B	128.62.255.255
C	192.110.26.255

netid.0 Una dirección con la porción de hostid puesta en ceros, hace referencia al mismo host (el host en el cual están los datos actualmente).

0.hostid Una dirección Internet con la porción de hostid puesta en ceros, hace referencia a la misma red. En los casos en donde un host tiene mas de una interfase de red, puede resultar ambiguo.

3. Mapeo de Direcciones

El objetivo del direccionamiento y del mapeo de estas direcciones es para facilitar la convivencia entre protocolos, aplicaciones y usuarios finales. El objetivo de mapear direcciones es: mapear una dirección física a una dirección Internet, y de una dirección

Internet a un nombre de host. El mapeo de direcciones físicas es útil para el medio físico, pero no para los protocolos de mas alto nivel. La dirección Internet es apropiada para los protocolos de alto nivel, pero no para los usuarios finales, para estos es mas fácil utilizar el nombre del host.

3.1 Mapeo de Dirección Internet a Dirección Física.

Si la comunicación realmente existe en el medio físico, ¿Cómo pueden comunicarse los hosts sin conocer cada uno la dirección física del otro? La respuesta es simple, no pueden. La resolución de las direcciones deben ser realizada antes de pasar a los protocolos de más alto nivel, y estos puedan comunicarse.

Se han implementado varios métodos para resolver este problema, se examinaran las tres siguientes:

•Mapeo Directo por la codificación de la dirección física en la dirección Internet (Resolución de direcciones por "Mapeo Directo").

•Uso de las tablas estáticas para obtener el mapeo de dirección Internet a Dirección Física (Resolución de direcciones "Tabla Estática").

•Implementación de un protocolo de asociación dinámico para obtener el mapeo de direcciones como se vaya necesitando (Resolución de direcciones por "Asociación Dinámica").

3.1.1 Resolución de direcciones por "Mapeo Directo"

Una forma de mapear la dirección física a una dirección Internet es codificando la dirección física dentro de la dirección Internet, este método es conocido como "Mapeo Directo", y solo es posible en un medio físico con cortas, y localmente asignadas direcciones. La resolución de direcciones por mapeo directo es fácil de entender, pero no es flexible por que la dirección lógica asignada depende de la dirección física.

3.1.2 Resolución de dirección por medio del uso de una "Tabla estática".

Con este método se crea una tabla que contiene un par de direcciones: la dirección física y las dirección Internet. Usualmente este es un proceso manual. La resolución de la dirección se realiza al consultar la dirección de Internet en la tabla y regresar la dirección física asociada a esta dirección Internet. Este método es mas flexible que el anterior, y fácil de implementar. Pero por tratarse de un proceso manual, existen riesgos de errores del administrador de la red.

3.1.3 Resolución de direcciones por "Asociación Dinámica"

El objetivo de éste método es que la computadora realice el trabajo. Un protocolo de bajo nivel denominado ARP (Address Resolution Protocol), fue diseñado para realizar la asociación dinámica de la dirección física a la dirección Internet.

3.1.3.1 ARP (Address Resolution Protocol)

La idea de asociación del protocolo ARP es simple. Debido a que ARP es una *broadcast*, todos los hosts de la red reciben la petición de la resolución de la dirección física asociada a la dirección Internet , pero solo el host al cual le pertenece esta dirección Internet responde. Este hosts, llena la información referente a su dirección física y devuelve el mensaje al emisor. Ahora que el emisor ya tiene la dirección física del host al cual se quería comunicar, entonces envía el paquete que había empezado a enviar.

Este esquema presenta algunas ineficiencias:

por ser un broadcast, la red se saturaría con solo las peticiones de resolución de direcciones. Estas ineficiencias se eliminan por medio del "ARP Caching".

3.1.3.1.1 ARP Caching.

Una característica de ARP es que cada host "atrapa" (caching) las direcciones de los hosts con los que se ha comunicado, de esta forma no realiza la petición a través de la red, sino que lo consulta en el *cache* de ARP: cuando una respuesta a una petición de resolución de dirección es recibida, el mapeo es almacenado en el *cache* de ARP. De esta forma la próxima vez que envíe un dato por la red, el emisor antes de realizar la petición de resolución de dirección, lo consulta en el cache de ARP.

Paquete ARP

Hardware Type		Protocol Type
HLen	PLen	Operation
Sender hardware address (HA)		
Sender protocol address (PA)		
Target hardware address		
Target protocol address		

Campo	Descripción
Harware Type	Selecciona el tipo de medio usado Ethernet=1
Protocol Type	Tipo de protocolo de dirección usado Dirección Internet (IP)=0800(16)
HLen, PLen	Longitud en bytes de la dirección física(HLen) y del protocolo de dirección(PLen) Ethernet=6, Dirección IP=4
Operation	Petición ARP=1, respuesta ARP=2 Petición RARP=3, respuesta RARP=4
Sender HA	Dirección física (hardware) del host que envía el paquete (Dirección Ethernet)
Sender PA	Dirección del protocolo que envía el paquete (Dirección IP)

Target HA	Dirección Hardware (Física) del host con el que se desea comunicar
Target PA	Dirección del Protocolo (IP) del host con el cual se desea comunicar

3.1.3.2 Reverse ARP (RARP)

Otra dificultad para la resolución de direcciones ocurre para las maquinas que no tienen unidades de almacenamiento secundario. Para que una máquina se pueda conectar a la red necesita conocer su dirección lógica, pero para conocer su dirección lógica tiene que acceder el disco de almacenamiento remoto... Esta dificultad se soluciona por medio de otro protocolo denominado RARP. RARP es usado por maquinas sin unidades de almacenamiento secundario para obtener su dirección IP al tiempo de inicializar la maquina (boot).

RARP utiliza el mismo formato del paquete de ARP, solo cambia el campo de operación, 3 para una petición de RARP y 4 para una respuesta de RARP. Al tiempo de inicializar (boot), la maquina sin unidades de almacenamiento secundario construye un paquete de RARP con su dirección física y se envía un broadcast en la red. Un servidor de RARP contiene una tabla con el mapeo de direcciones Física-Internet. En muchas maquinas UNIX el archivo /etc/ethers es usado para almacenar este mapeo. Cuando un servidor de RARP (rarpd en algunas máquinas UNIX) recibe una petición , consulta la tabla de mapeo y regresa el paquete con la dirección Internet asociada a la dirección de hardware presentada en el paquete inicialmente.

3.2 Mapeo de Nombre de Host a Dirección Internet.

Es mas fácil para los usuarios recordar el nombre de una maquina que el dirección Internet de 32 bits. De igual forma que existen varios métodos para realizar un mapeo de una dirección Física a una dirección Internet, también existen varias formas de mapear un nombre de host a una dirección Internet:

Sistema de información de red (Network Information System -NIS)

Sistema de Dominio de Nombres (Domain Name Server -DNS)

Archivo /etc/host

La forma mas común de mapear el nombre de host a una dirección Internet es manualmente por medio de una tabla estática: el archivo /etc/host. Este archivo permite asociar un nombre de host con una dirección Internet y con uno o más *alias* (nombre pequeño).

3.2.1 Sistema de información de red (Network Information System -NIS)

Formalmente conocido como "Yellow Pages" (YP), es una aplicación desarrollada originalmente por SUN. NIS es una base de datos de propósito general de búsqueda, que almacena información en mapas. Uno de los mapas es el "host map", el cual contiene información similar a la contenida en el archivo /etc/host de cada host.

Las máquinas conectadas a la red están divididas en *Dominios*. Cada dominio contiene *un Servidor Maestro*, que contiene todos los datos, un *Servidor Esclavo*, (opcional) el cual contiene una copia de los datos para dar mayor rendimiento, confiabilidad y redundancia, *Clientes* que no poseen información propia pero que requieren servicios de mapeo de nombre de host a Dirección Internet de los servidores maestro y esclavo.

En NIS, el administrador actualiza únicamente la información en el Servidor Maestro y todos los cambios se en reflejados en todas las demás máquinas del dominio.

NIS requiere de archivos de configuración especial y de *Deamons*.

3.2.2 Servidor de Dominio de Nombres (Domain Name Server -DNS)

Es un servicio de búsqueda fácil de Nombre de Host. Provee un esquema de asignación de nombres en forma jerárquica a todos los sistemas conectados a la red Internet. Los Nombres de Host. como todos los encontrados dentro de Internet deben ser únicos. DNS agrupa y organiza todo el mundo contenido en Internet, de esta forma las organizaciones gubernamentales se agrupan en el dominio GOB, las organizaciones educacionales pertenecen al dominio EDU, etc. Como sitios que requieren servicios Internet , estas organizaciones también se pueden registrarse con un nombre de dominio en el dominio apropiado.

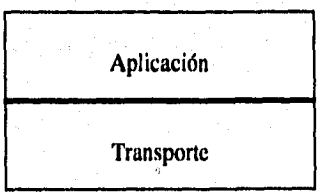
Los dominios utilizados por DNS son completamente separados y no relacionados con los dominios utilizados por NIS. DNS provee un sistema de nombramiento global, jerárquico que garantiza un Nombre de Host único dentro de toda la red Internet. NIS provee una local-lógica división de máquinas para servicios de NIS. Un lugar puede organizar un servicio local como lo necesiten. DNS se organiza por personas que trabajan y manipulan los servicios de Internet a nivel internacional.

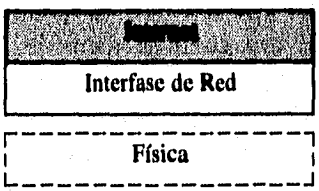
DNS es similar a NIS, en que ambos manejan varios niveles de servidores que contienen mapeos de Nombre de Host-a-Dirección Internet para un dominio.

DNS requiere de archivos de configuración y *Deamons*.

4. Protocolos Internet

Los protocolos Internet están posicionados entre la capa de Transporte y la Capa de Interfase de red (módulos dependientes de hardware) dentro del Internet Protocol Suit.





El protocolo de Internet (IP) es uno de los protocolos más usados en Internetworking. IP es llamado por los protocolos de alto nivel para utilizar sus servicios de envío de paquetes, y por las rutinas manejadoras de la red cuando se reciben paquetes.

EL Internet Control Message Protocol (ICMP) es una parte integral de IP: ICMP es usado para comunicación (reportes de errores en particular) entre los módulos de software IP en diferentes máquinas.

4.1 Protocolo Internet

El protocolo Internet realiza el Servicio de Entrega de Paquetes, con las siguientes características:

- No Orientado a Conexión
- No confiable
- Mejor esfuerzo

El protocolo Internet tiene como responsabilidades:

- Definir el Datagrama IP
- Enrutamiento
- Fragmentación y Reensamble de datos

4.1.1 Servicio de Entrega de Paquetes.

4.1.1.1 Servicio de Entrega de Paquetes No Orientaod a Conexión.

Realiza la coordinación punto a punto. Cada paquete es tratado en forma independiente, los paquetes pueden llegar en forma desordenada, y otros pueden no llegar completos. Es un proceso similar al postal: tse deposita la carta en el buzón y el sistema de correo se encarga de hacerlo llegar a su destino.

4.1.1.2 Servicio de Entrega de Paquetes No Confiable.

El protocolo Internet no garantiza la llegada de los paquetes

4.1.1.3 Servicio de Entrega de Paquetes realizando el Mejor Ezfuerzo.

Esto es, los paquetes no son descartados fácilmente, los paquetes pueden no ser entregados por alguna falla de hardware.

4.1.2 Responsabilidades.

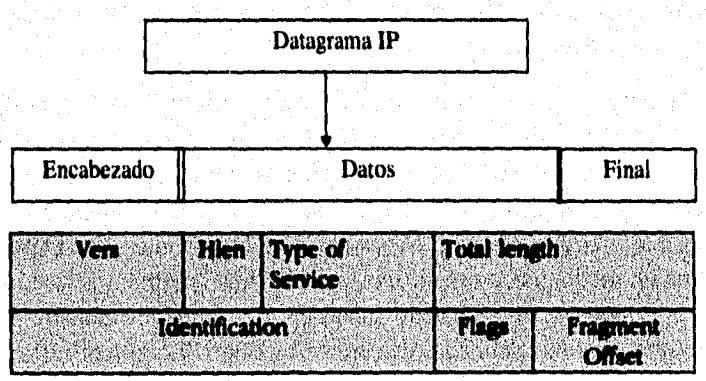
4.1.2.1 Definir el Datagrama IP

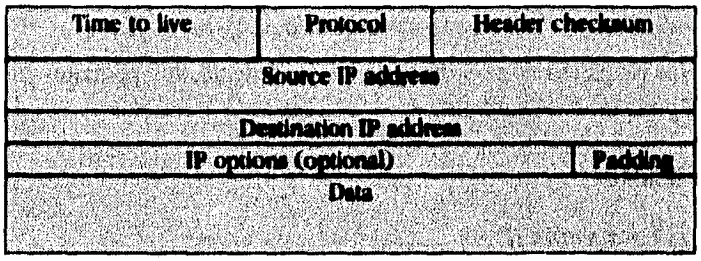
Una de las funciones de esta capa es definir la unidad básica de datos utilizado por las redes TCP/IP.

4.1.2.1.1 Datagrama IP

TCP/IP a través del Protocolo Internet define la unidad básica de transferencia de datos: el Datagrama IP. Este datagrama está encapsulado dentro del frame físico de la red.

Campo	Descripción
Vers	Versión del Protocolo IP
Hlen	Longitud del encabezado en palabras de 32 bits
Type of sevice (TOS)	Utilizado para enrutamiento
Total Length	Número de bytes en el Datagrama completo (encabezado y datos)
Identificación	Usado para la fragmentación y reensamble de datos
Flags	
Fragment Offset	
Time to live (TTL)	Cantidad de tiempo (segundos) de un Datagrama en la red, utilizado en
enrutamiento	
Protocol	Protocolo de más alto nivel al cual va a ser enviado el paquete
Header Checksum	Utilizado para asegurar la llegada de los datos
Source IP Address	Dirección IP del emisor
Destination Address	Dirección IP del recibidor
IP options	Utilizado para pruebas y depurar
Padding	
Data	Un dato está siendo entregado, usualmente un paquete desde un
protocolo de	
	más alto nivel (TCP, UDP, etc)





4.1.2.2 Enrutamiento de Paquetes.

El protocolo Internet también es responsable de seleccionar las rutas por las cuales van a viajar los datos.

4.1.2.3 Fragmentación y Reensamble de datos.

IP incluye también una serie de reglas para definir como se van a procesar los datos. Incluyendo cuando se generan mensajes de error y cuando se descartan paquetes, Parte de este proceso incluye reensamble y Fragmentación de Datos, este proceso se realiza sólo cuando el hardware lo requiere.

La función más importante de IP es la Fragmentación y el Reensamble de Datos.

IP lleva a cabo la fragmentación de datos únicamente cuando el datagrama es demasiado largo para ponerlo dentro de un frame físico para el medio por el cual los datos van a ser enviados. Usualmente esto ocurre en un enrutador IP, cuando los datos llegan y se tienen que enviar a través de la red por un medio físico el cual tiene una "Unidad de Transmisión Máxima" (MTU) más pequeña que el medio por el cual se envió, donde la MTU indica el mayor número de datos que pueden viajar por un frame físico.

Cuando se fragmenta un Datagrama, cada Datagrama fragmentado tiene el mismo formato que el Datagrama original.

La identificación del campo consiste en un único entero que identifica el datagrama. Cuando IP fragmenta el Datagrama este campo es copiado en cada encabezado de los Datagramas que incluyen cada fragmento del Datagrama original. Esto permite al módulo IP destino saber cuales fragmentos pertenecen a cada datagrama cuando éstos se reemzamblan.

El bit 3 del campo FLAGS controla la fragmentación. El bit menos significativo es denominado "más fragmentos", si este bit está encendido, significa que este Datagrama está fragmentado y que los siguientes Datagramas también son fragmentos. Un Datagrama con el bit "más fragmentos " apagado significa que el Datagrama no debe ser fragmentado bajo ninguna circunstacia. Si se necesita fragmentar este Datagrama para poderlo transmitir a un media físico específico, y el bit "más fragmentos" está apagado, el Datagrama es descartado y se envía un mensaje de error al emisor. El bit más significativo es reservado y normalmente es cero. El campo "Fragment Offset" indica la

localización del fragmento en el Datagrama original al que este fragmento pertenece. Es medido por unidades de 64 bits y el primer fragmento tiene un offset de cero.

El módulo IP en el nodo destino utiliza los campos "Fragment Offset" , "Identification" y "Total length" para reeensamblar el Datagrama.

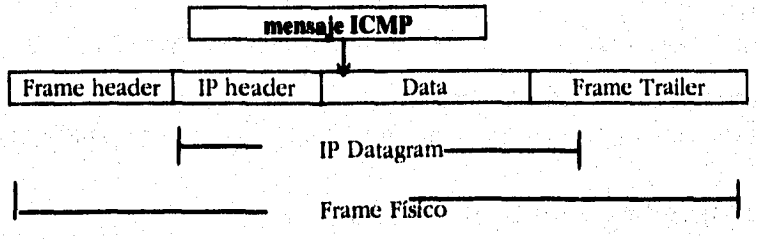
El bit "más fragmentos" del campo FLAGS indica si el Datagrama está fragmentado o no. El campo

Al sustraer al longitud total del Datagrama original del campo "Total length" el módulo IP destino sabe la cantidad de datos que tiene que reensamblar para obtener el Datagrama original. Esta información es necesaria por que IP es un servicio Conectionless y los Datagramas no están garantizados para llegar en el orden en el cual fueron enviados.

4.2 Internet Control Message Protocol (ICMP).

Su función es reportar errores y otros mensajes de control entre los módulos IP sobre varios hosts y gateways.

Los mensajes ICMP viajan dentro de un Datagrama IP, que a su vez viaja dentro de un frame físico. Existen varios tipos de mensajes ICMP y cada uno tiene su propio formato.



4.2.1 ICMP "Echo Request and Echo Replay"

EL mensaje ICMP más conocido es el "Echo Request". Un programa que envía un "Echo Request" en *ping*. Al utilizar el comando *ping*, se está enviando un Datagrama IP que contiene un ICMP "Echo Request" a un host específico. Si el host recibe el Datagrama, éste responde y regresa un Datagrama con un ICMP "Echo Reply". Esto ocurre en el nivel IP y los protocolos de más bajo nivel , por esto el comando ping es una herramienta valiosa para realizar pruebas de hardware y software a bajo nivel.

Otros mensajes ICMP:

Tipo de Mensaje	Descripción
0	Echo reply
3	Destination unreachable

4	Source qench
5	Redirect (change rute)
8	Echo request
11	Time exceeded fro datagram
12	Parameter problem on datagram
13	Timestamp request
14	Tiemstamp reply
15	(Obsolete)

5. Enrutamiento de Paquetes.

5.1 Anatomía de una Internet

Como se vio anteriormente en direccionamiento físico, la dirección lógica Internet es necesaria para permitir a protocolos de alto nivel y aplicaciones tratar a varias redes físicas (de bajo nivel), como una red lógica. Estas diferentes redes físicas deben ser interconectadas de dos formas: lógica y físicamente.

Físicamente las redes están interconectadas entr sí.Esta conectividad física no asegura comunicación entre los hosts de diferentes redes lógicas. Para asegurar ésta conexión, la máquina que conecta las redes físicas debe transmitir paquetes entre los diferentes tipos de red. Esta tarea es conocida como "enrutamiento", y provee conexión lógica entre las redes. Una máquina que realiza esta tarea es conocida como enrutador.

El enrutamiento en una red TCP/IP es realizada por la capa IP.

Un enrutador es un máquina que envía paquetes por varias rutas entre varias redes a las que está conectada. Un gateway transforma paquetes de un protocolo a otro.

5.2 Enrutamiento Directo e Indirecto.

Las máquinas que están conectadas a la misma red física realizan enrutamiento directo: un Datagrama puede ser encapsulado dentro de un frame físico y ser enviado directamente a la máquina conectada. *Ruteo Directo*: "Transmisión física de datos sobre un medio de comunicación", es la base de la comunicación en una red TCP/IP.

Para las máquinas que no están conectadas a una misma red física, se deben enrutar paquetes a cada una de las redes que estén interconectadas a través de un gateway. Este proceso es conocido como Enrutamiento *Indirecto*.

5.2.1 Tablas de Enrutamiento (Table-Driven).

El enrutamiento básico de IP es "Table-Driven". Las decisiones de enrutamiento están basadas en una estructura conocida como la tabla de rutas (ruoting table".

Un método de mantenimiento de la tabla de rutas es por medio del comando "route, este método es conocido como "Enrutamiento Estático" (Static Ruoting). Las rutas creadas por el comando *route* no cambian después de su creación, a menos que el administrador de la red explícitamente las altere.

En casos más complejos , el mantenimiento y la actualización de la tabla de enrutamiento se da automáticamente por medio de software, este proceso es denominado "Enrutamiento Dinámico".

5.2.2 Tabla de Enrutamiento Estándar.

La tabla de enrutamiento IP estándar es un par de datos *(netaddr, netx-gateway)*

Netaddr, es una dirección Internet de un host específico o de una red específica.

Netx-gateway, es un nombre o una dirección de un host directamente conectado que puede enrutar al host destino o a la red destino (netaddr). Existen tres tipos de rutas diferentes: host específico, red específica y default.

Tabla de Enrutamiento IP

Netaddr	Netx.gateway
Ruta de hot específico 192.162.90.2	192.162.70.3
Ruta específica de red 192.162.80 128.185	 192.162.70.2 192.162.70.4
 192.162.70.1	

El *netx-gateway* debe ser un host conectado directamente conectado a la red física.

5.3 Algoritmo de enrutamiento IP estándar.

{

Get **Destination IP Address** *from IP header;*
Extract ***Destination Network Address*** *from* ***Destination IP Address;***

Else if ***Destination IP Address*** *matches a* ***Host-specifc Route***

Then (Route datagram per Routing Table)

Else If ***Destination Network Address*** *matches a* ***Network-specific Route***

Then (Route datagram per Routing Table)

Else If ***Default Route***

Then (Route datagram per Routing Table)

If ***Destination Network Address*** *matches a directly connected network*

Then
(Bind Destination IP Address to a Physical Address;
Encapsulate the Datagram in a Physical frame;
Send the frame)

Enrutamiento Directo

Enrutamiento Indirecto

Else { Report a routing error}
}

A continuación se discute el algoritmo IP estándar.

Get **Destination IP Address** *from IP header;*
Extart ***Destination Network Address*** *from Destination IP Address;*

La dirección IP del host al cual se le va a enviar el paquete está dentro del encabezado del datagrama. IP extrae la dirección de la red de la dirección IP, y analizando los primeros bytes de ésta dirección se determina la clase de dirección Internet. La clase de dirección Internet nos dice la cantidad de bytes que se van a ocupar para la porción de la dirección de red.

If ***Destination Network Address*** *matches a directly connected network*
Then
(Bind Destination IP Address to a Physical Address;
Encapsulate the Datagram in a Physical frame;
Send the frame)

IP puede decir que la red está directamente conectada por que a cada interfase física de la red se le debe asignar una dirección IP. IP extrae la porción de red de la dirección IP de su propia interfase y la compara con la dirección de red destino. Si éstas son iguales, el datagrama es enviado. Este proceso se conoce como enrutamiento directo.

Else if ***Destination IP Address*** *matches a* ***Host-specific Route***
Then (Route datagram per Routing Table)

IP busca en la tabla de rutas si existe una ruta especialmente definida para el host destino, comparando la dirección destino con las rutas que se encuentran definidas en la tabla. Si se encuentra una ruta especifica para el host destino, IP rutea el datagrama al gateway especificado en la tabla de rutas.

*Else If **Destination Network Address** matches a **Network-specific Route***
Then { Route datagram per Routing Table }

En vez de definir rutas para cada host, se puede definir rutas para redes. La próxima revisión es hecha para comparar la dirección destino de red con las direcciones de red definidas en la tabla de enrutamiento. Si existe una ruta para la dirección de red especifica , IP rutea el datagrama al Gateway definido para esta ruta.

Else if Default Route
Then {Route Datagram Per Routing Table}

Finalmente IP busca una ruta por defautl, una ruta por defautl es útli por que se puede configurar a un host una ruta para cada destino en caso de que no exista un gateway "más inteligente" y le permite al gateway decidir como enrutar el datagrama. Este uso de la ruta por defautl ayuda a obtener la tabla de enrutamiento en la mayoría de los host relativamente pequeños.

Else { Report a routing error }

Si todo lo anterior falla, IP regresa un error de enrutamiento. El usuario lo recibirá en un mensaje de error "destination unreachable".

5.4 División en Subredes (Subnetting).

" Subnet addresing", "Subnet routing" o "Subnetting" es un método que sigue una lógica, dirección Internet para extenderse sobre varias redes físicas.

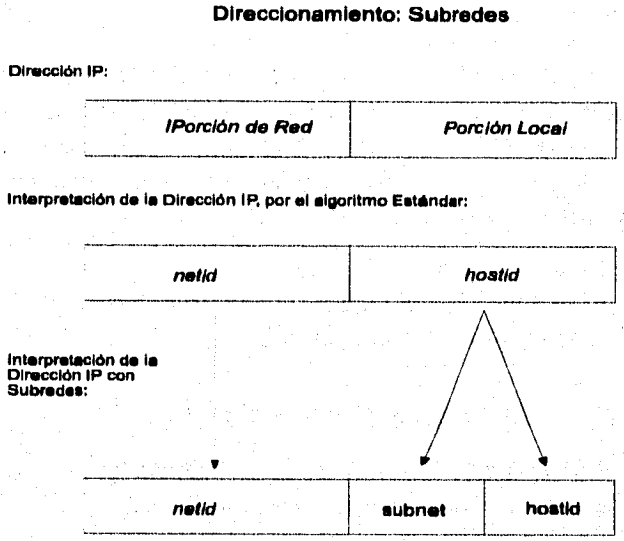
Antes de la implementación de Sunnetting, existía una corespondencia uno a uno entre las redes físicas y la dirección internet. Esto ha causado problemas debido a que las redes físicas tienen limitaciones físicas (núm. de nodos, longitud, etc), y se tenía que sumar redes físicas para soportar el crecimiento, y esto forzaba a tener una dirección de red Internet para cada una de estas redes físicas.

El mantenimiento de múltiples direcciones redes Internet puede hacer difícil la tarea del administrador de la red, pero es aún más complicado para los gateways del *backbone* de la red, manipular esta información, estos gateways necesitan saber una ruta para cada red, y esto se refleja en la tabla de rutas: un problema potencial de rendimiento.

También debido a que existe un número fijo de direcciones Internet, los diseñadores de Internet están preocupados por que se agoten las direcciones, sería bueno que para cada lugar físico aunque existieran varias redes físicas se le asigne una dirección Internet.

Subnetting fue diseñada para eliminar o minimizar estos problemas. La idea básica de subnetting es permitir (al lugar físico) interpretar la porción de hostid. El éxito de la flexibilidad de subnetting reside en su interpretación local.

Los lugares físicos pueden asignar una diferente dirección de subnet a cada red física. En el mundo exterior (fuera del lugar físico), se sigue interpretando la dirección Internet en forma estándar en los primeros bits del primer byte de la dirección. La interpretación de "subnet" se realiza localmente dentro de la subred de la red. Esto permite al lugar físico local interpretar o cambiar su esquema de subnetting sin afectar alguien más en Internet. Esto también libera espacio en los gateway del backbone: solo necesita conocer una ruta para una dirección Internet para todo el lugar físico.



5.4.1 Mascara de Subnet.

Subnetting se implementa al poner una máscara de bits denominada "Subnet Mask", para indicar la cantidad de bytes de la dirección que va a ser interpretada como netid y hostid respectivamente. A parte de determinar estas porciones, el resto de enrutamiento IP lo toma igual.

La máscara de subred es una máscara de 32 bits, usualmente escrita en formato hexadecimal Por ejemplo para las redes de clase "b" se selecciona el tercer byte de la dirección Internet como subred y el último byte como hostid. La máscara de subred utiliza un esquema como el siguiente:

0xffffff00

Este es un esquema de subred muy común, pero los bits seleccionados para subred son definidos por el lugar físico, y puede ser un esquema más complejo. Todas las clases de direcciones Internet pueden ser subdivididas , y la dirección de subred es asignada para

cada inteface de red (en los gateways se pueden definir diferentes máscara para cada red), lo cual hace subnetting más flexible.

Para obtener la porción de netid en una dirección donde se está implementando subnetting, se realiza una operación lógica AND entre la máscara de subred y la dirección Internet.

5.5 Algoritmo de enrutamiento IP con subredes (subnetting)

La implementación de subnetting en el algoritmo de enrutamiento IP y en la tabla de enrutamiento requiere de algunas modificaciones. Se necesitan almacenar las mascaras de las subredes en la tabla de enrutamiento. El par (netaddr, netx-gateway) se convierte ahora en un trío:

(netmask , netaddr , netx-gateway)

Para modificar el trío de información (netmask, netaddr, netx-gateway) para "casos especiales " (tales como la "host-spacific route" y la ruta por default) y ordenar la taba de rutas correctamente, el algoritmo original se colapsa en un simple uy eficiente ciclo (loop).

{
Get **Destination IP Address** from IP header;

For each route table entry **e** do

netid = Destination IP Address AND netmask (e);

If netid = netaddr (e)

Then if **netaddr(e)** is a directly connected network

Then [Bind Destination IP Address to a Phisical Address;
Encapsulate the datagram in a Physical frame;
Send the frame}

Forend;

If no match was found

Else { Route datagram to **netx-gateway(e)** }

Then {
Report a routing error}

}

Direct Routing

Indirect Routing

A continuación se explica el algoritmo de enrutamiento IP con subredes.

*Get **Destination IP Address** from IP header;*
*For each route table entry **e** do*

El algoritmo ahora es un ciclo, el cual revisa en la tabla de rutas una por una de las de rutas hasta que es encontrada la entrada *e*. Esto implica que se deben de dar de alta rutas para las redes directamente conectadas, por que la prueba para conexiones directas está contemplada dentro del ciclo. El software realiza esto automáticamente en este tipo de implementaciones; el administrador no tiene que realizar modificaciones para las actualizaciones especiales.

netid = Destination IP Address AND netmask (e);

Se realiza una AND lógica entre la dirección IP destino y la netmask asociada en la tabla de rutas con la entrada e, y almacena el resultado en una variable temporal netid.

If netid = netaddr(e)

Compara el resultado de la operación lógica AND entre la dirección de red asociada con la entrada *e* en la tabla de rutas. El termino dirección no es tomado literalmente en este caso. Para una ruta a un host-específico la dirección de red será una dirección de host entera; esto es el "match" debe incluir la porción de host para la ruta a un host-específico que va a ser usada.

Then if netaddr (e) is a directly connected network
Then [Bind Destination IP Address;
Encapsulate the dtagram in a physical frame;
Send the frame]

Si la comparación entre la entrada e y la dirección asociada a esta entrada e en la tabla es válida, verifica si la dirección de red es una dirección directamente conectada. IP podría hacer esto aplicando una operación lógica AND entre la(s) interfase(s) de red de sus propias interfases de red con la netmask(e) y comparándola con la netaddr(e), pero no sería eficiente. La mayoría de las implementaciones almacenan una bandera en la ruta de entrada que indica si está o no directamente conectada.

*Else [Route datagram to **netx-gateway(e)**]*
forend;
Then [rreport a routing error]

Si el resultado de la operación anterior no es una red directamente conectada, realiza un enrutamiento indirecto para enviar el datagrama al gateway indicado en la tabla de rutas.

Este es el fin del ciclo. Continua hasta que la entrada a sido encontrada o hasta que la tabla se haya recorrido toda, en este caso un mensaje de error es regresado.

5.6 Enrutamiento entre Gateways.

El algoritmo de enrutamiento IP es muy simple, y la clave está en la tabla de rutas. La eficiencia y el rendimiento de un gateway depende de la actualización de la tabla de rutas.

El mantenimiento de la tabla de rutas (en los gateways) idealmente debe ser realizado en forma automática vía software. Este mantenimiento en forma dinámica a la tabla de rutas, es exactamente para lo que los protocolos de enrutamiento dinámico o protocolos de gateway están diseñados a realizar. Los protocolos de enrutamiento dinámico soportan el intercambio de información sobre enrutamiento entre varios tipos de gateway.

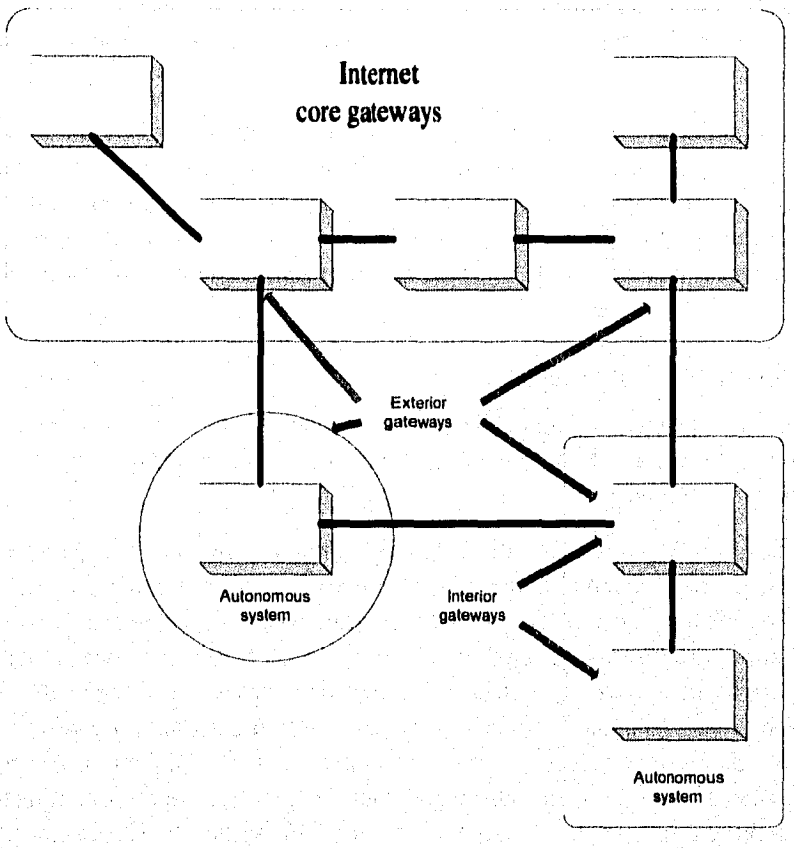
5.6.1 Diferentes tipos de gateway.

La Internet puede ser interconectada a través de una colección de sistemas autónomos por un conjunto de anillos de gateways (core gateways) que por sí mismos forman un sistema autónomo.

Un "Sistema Autónomo" es un grupo de computadoras, redes y gateways que están controlados por el mismo cuerpo administrativo (por ejemplo, una corporación de fuentes de cómputo pueden forman un sistema autónomo). Los gateways dentro de un sistema autónomo se dice que son interiores. Por que todos ellos están controlados por el mismo cuerpo administrativo, un protocolo Internet de Gateway puede ser seleccionado para correr sobre esos sistemas que son específicos al sistema autónomo y conoce mejor las necesidades de intercambio de información de enrutamiento dinámico. El anillo de Gateways de Internet son también todos controlados por un solo cuerpo administrativo, por lo que también forman un sistema autónomo. El anillo de Gateways tiene la información completa sobre enrutamiento de toda la red Internet. Es por esto lo que se refieren a formar un anillo (core).

Los gateways que están físicamente conectados, pero residen en diferentes sistemas autónomos (que están bajo el control de diferentes organizaciones) se dice que son exteriores a algún otro sistema autónomo. Estos gateways también necesitan estar de acuerdo sobre la forma en que intercambiaran la información de enrutamiento. Los gateways sobre sistemas autónomos que están directamente conectados a la Internet deben advertir (intercambiar información) a los gateways del anillo, y así pueden mantener una visión completa de los nodos de Internet. El protocolo de Gateway Exterior es usado para el intercambio de información entre Gateways exteriores.

Tipos de Gateways



5.6.2 Protocolos entre Gateways.

GGP. Gateway to Gateway Protocol, fue usado por el anillo de Gateways de ARPANET para intercambiar información de enrutamiento. GGP es un protocolo de enrutamiento vector-distancia. Los mensajes GGP viajan dentro de un datagrama IP, en forma similar a los mensajes UDP y TCP. Cuando usan GGP, un anillo de gateways , intercambian periódicamente información sobre enrutamiento con sus vecinos, ésta información de enrutamiento consiste en un par (netid, distancia), que muestra todas las redes que el protocolo puede alcanzar, a través de un "hop count". Los vecinos actualizan las tablas de rutas con esta información , y reenvían la información a sus propios vecinos. De esta forma la información sobre enrutamiento se propaga en todos los anillos de gateways de Internet. GGP fue reemplazado por un protocolo de enrutamiento denominado "Shortest Path First" (SPREAD) que aún no ha sido documentado como estándar en Internet.

5.6.3 Protocolos de Gateways Exteriores.

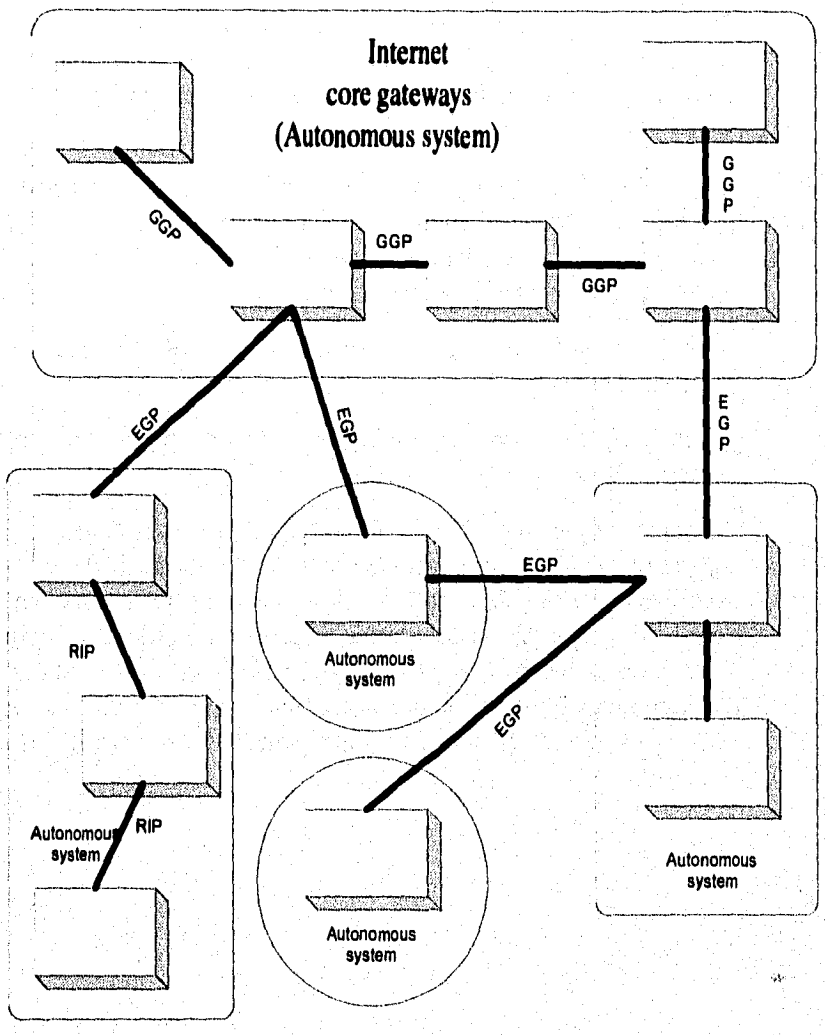
EGP. Exterior Gateway Protocol es usado por un gateway para asegurar la funcionalidad en la transferencia de información entre sistemas autónomos , los cuales a su vez pertenecen a otros sistemas autónomos incluyendo el anillo de gateways de Internet. Las principales características de EGP son las siguientes:

-Definición de un proceso similar de adquisición, un gateway utiliza este proceso para establecer comunicación EGP con otro gateway. La decisión de cual gateway intercambiará la información de enrutamiento, es estrictamente administrativa, y no es parte del software EGP.

-Pruebas constantes entre gateway con EGP, para verificar su funcionamiento.

-Periódicamente se realiza intercambio de información (actualización) de enrutamiento entre gateways que utilizan EGP.

BGP. Border Gateway Protocol, es un protocolo exterior nuevo que ha sido propuesto como estándar de Internet. BGP fue diseñado para ajustar las necesidades de Internet como existe actualmente y para ser fácilmente extensible, y que se pueda ajustar a las necesidades futuras. BGP elimina algunas de los problemas asociados con EGP por proveer un ciclo libre de intercambio de información de enrutamiento, y una fase de nuevas versiones. Varias versiones de BGP están siendo sometidas a pruebas dentro de Internet.



5.6.4 Protocolos de Gateways Interiores (IGP).

IGP, Interior Gateway Protocol (IGP) es usado por los gateways dentro de un sistema autónomo para intercambiar información de enrutamiento. IGP es un término general no un protocolo específico. Algunos IGP´s comunes son: RIP, HELLO, OSPF.

RIP. Routing Information Protocol es un protocolo de enrutamiento vector-distancia que es distribuido como parte del sistema operativo UNIX 4BSD. Un gateway en el cual se está corriendo RIP en forma activa, envía broadcast de actualización de enrutamiento cada 30 seg. Los host pueden correr RIP en forma pasiva, recibiendo los broadcast de actualización de enrutamiento, pero no los reenvía. De ésta forma la actualización de la información de

enrutamiento llega a todos los hosts y gateways en el sistema autónomo. Los "ruouted deamons" son programas que implementan RIP.

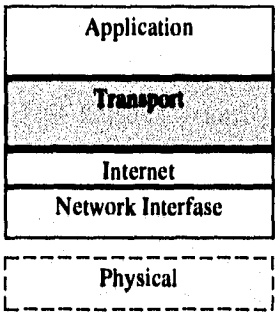
HELLO. Hello es otro protocolo vector-distancia IGP. Utiliza una métrica basada en el tiempo que tarda en llegar un paquete por una ruta específica, en vez de "hop count". La operación general de Hello es similar a RIP. Un gateway transmite actualizaciones de enrutamiento periódicamente y los demás gateways reciben la actualización recalculando la mejor ruta al destino especificado, determinada por el tiempo que tardo en llegar. Debido a que las actualizaciones Hello utilizan un "timestamp" (usado para calcular el tiempo que tarda en llegar a un destino), también puede ser utilizado entre host para sincronizar el reloj. Históricamente Hello es importante por que fue el IGP originalmente usado en el backbone NSF NET.

OSPF. Open SPF, o Open Shortest Path First es un nuevo IGP que ha sido diseñado para superar varias de las limitaciones técnicas de los primeros IGP's (tales como RIP y HELLO). Una de las ventajas de OSPF es que utiliza un algoritmo SPF el cual compara (escala) las redes grandes de forma más eficiente que los algoritmos usados en RIP y HELLO. OSPF incluye también características avanzadas tales como el tipo de servicio de enrutamiento (selecciona la mejor ruta basado en factores como el costo o el tiempo que tarda en vez de una simple métrica con lo es el "hop count"). y balancear (distribuir el tráfico sobre sobre múltiples rutas a un destino). OSPF maneja también mensajes de autenticación sobre actualizaciones de enrutamiento, para incrementar la seguridad.

gated. EL "UNIXGATEway Deamon" fue diseñado originalmente por la Universidad de Cornell. *gated* puede aceptar actualizaciones de enrutamiento desde EGP, RIP, HELLO, y produce una tabla de actualización de enrutamiento. *gated* puede también advertir (informar) sobre actualizaciones de enrutamiento dentro de un sistema autónomo usando RIP, y/o HELLO, y a otros sistemas autónomos usando EGP. En algunas aplicaciones *gated* ha sido modificado para manipular actualizaciones de rutas BGP.

6. Protocolos Host-to-Host: TCP y UDP.

Los protocolos Host-to-Host, son también referidos como protocolos de transportes, posicionados debajo de la capa de aplicaciones y sobre la capa de Interfase de Red, en el modelo de TCP/IP. Las aplicaciones utilizan los servicios de uno de los protocolos de transporte para comunicarse (al mismo nivel) con otros hosts sobre la red. En forma subsecuente los protocolos de transporte utilizan los servicios de la IP para entregar los paquetes.



Transmission Control Protocol (TCP) es el protocolo mayor y que da nombre al Internet Protocol Suite. TCP provee una conexión-orientada (connection-oriented), y un servicio de entrega de paquetes confiable.

User Datagram Protocol (UDP) Provee a la capa de aplicación con el mismo servicio no confiable sin conexión que provee el protocolo IP.

6.1 Demultiplexaje basado en Número de puerto.

Una característica importante que ambos protocolos TCP y UDP comparten, es la habilidad para distinguir entre múltiples destinos (procesos) en un sólo sistema, a esta habilidad se le conoce como "Demultiplexaje basado en Número de puerto". Un puerto puede ser pensado como un punto destino, y es representado por un entero positivo. A cada proceso en el nivel de aplicación que utiliza uno de los protocolos del nivel de transporte le es asignado un número de puerto. Los protocolos de transporte utilizan éste número de puerto para determinar que proceso se está comunicando.

Hay dos puntos importantes que deben considerarse para los números de puertos. Primero el puerto es una abstracción lógica, no una entidad física (como del puerto serial en la PC). Segundo TCP y UDP usan los mismos números de puertos. Por ejemplo: el puerto 21 de TCP es lógicamente diferente del puerto 21 UDP. Esto se debe pensar como si fuera un dominio TCP y un dominio UDP, teniendo cada dominio su propio conjunto de puertos.

El concepto de puerto es importante por que es un mecanismo que permite a múltiples procesos sobre un solo host, utilizar (a todos) simultáneamente los medios de comunicación.

6.2 User Datagram Protocol (UDP)

Servicio de entrega de paquetesen UDP es;

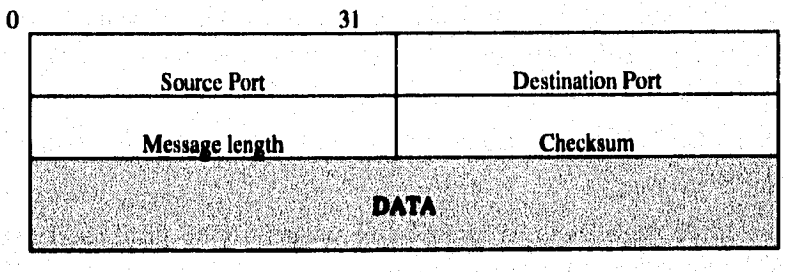
- No Confiable
- No Orientado a Conexión

•Mejor Ezfuerzo

El servicio es llamado no orientado a conexión, por que la coordinación End-to-end no es realizada. Cada paquete es tratado en forma independiente. El servicio es no confiable (unreliable), por que los paquetes pueden llegar fuera de orden, más de una vez (duplicados), o no llegar completos. Es el mismo servicio mejor ezfuerzo que se provee en la capa del protocolo IP.

6.2.1 Formato del Datagrama UDP.

EL datagrama UDP es encapsulado dentro de un datagrama IP, el cual a su vez en encapsulado en un frame físico que viaja a través de la red. En otras palabras el datagrama UDP viaja en la porción de datos del datagrama IP.



Campo	Descripción
Source Port	Opcional, El número de puerto UDP al cual este datagrama va a ser enviado. Este campo debe ser puesto en cero si no es usado.
Destination Port	El número de puerto UDP de el proceso en el host destino que va a recibir este datagrama.
Message length	La longitud, en bytes (octetos) , de el datagrama completo, incluyendo los 8 bytes del encabezado UDP.
Checksum	Opcionalmente un campo de checksum debe ser calculado sobre el datagrama completo. Si no es usado este campo debe ser puesto en cero. falta

6.2.2 Usos de UDP.

•Número pequeño de paquetes

•Modo de transacción síncrono

•Fácil recuperación de "crash"

UDP provee un mecanismo de comunicación sencilla (low-overhead) para procesos que están dispuestos a asumir la responsabilidad de la confiabilidad de sus comunicaciones.

Dos procesos están comunicándose:

El proceso A envía un mensaje al proceso B: NOW, el proceso B responde regresando un mensaje que contiene un día. La comunicación entre ellos está ahora completa. Si el proceso B recibe una petición dañada, el paquete es ignorado. SI A no recibe una respuesta dentro de un tiempo razonable, o el campo checksum no es correcto, la petición se retransmite. Siempre que la petición (paquete) se pierde, o la respuesta se pierda, o se dañen, una retransmisión es todo lo que se necesita.

Cuando se usa UDP para una aplicación como ésta, solo dos datagramas son necesarios: uno de A a B y otro de B a A. Sin embargo con TCP, tres paquetes deberían ser enviados para establecer una conexión: dos de datos y al menos uno más de reconocimiento, y varios más para cerrar la conexión. Cada uno de estos paquetes tendría el doble de tamaño de un paquete UDP.

Una aplicación que utiliza UDP es el Network File System (NFS). NFS utiliza un protocolo "statless", lo cual significa que cada transacción es independiente. Nada acerca de las transacciones previas debe ser recordada. Este protocolo habilita una muy simple recuperación de rompimientos

En resumen UDP puede ser una buena elección si se está dentro de una de las siguientes categorías:

1.El número de paquetes intercambiados es pequeño y no justifica el overhead adicional de TCP. (como el ejemplo mostrado)

2.Opera en una transacción de modo síncrono, esto es, se espera por una respuesta después de que el paquete es enviado.

3.Requiere de un fácil "crash recovery" similar a NFS.: desde que no existe conexión no tiene por que preocuparse de reestablecer una conexión.

4.La aplicación es responsable de verificar la integridad de los datos con UDP, habilitando el checksum UDP, o incluir un procedimiento de chequeo de errores dentro de la aplicación.

6.3 Transmission Control Protocol (TCP).

El protocolo TCP realiza un servicio de entrega de paquetes con las siguientes características:

- Orientado a Conexión
- Confiable

Dentro de las responsabilidades de TCP se encuentran las siguientes:

•Secuenciación de datos recibidos

•Reconocimiento de datos recibidos

•Retransmisión por perdida o daño de datos.

6.3.1 Servicio de Entrega de Paquetes.

6.3.1.1 Conexión-orientada .

Indica que los procesos que se están comunicando a través de TCP quieren establecer una relación de Término-largo, involucrando el movimiento de varios paquetes en una sesión relativamente larga.

6.3.2 Conexión punto-a-punto confiable.

Indica que TCP, acepta la responsabilidad de la secuencia de datos, validación, y si es necesario la retransmisión. La aplicación que usa estos servicios TCP no necesita preocuparse a cerca de esto, asume que los datos enviados serán recibidos en su totalidad, en el orden exacto en el que fueron enviados.

6.3.3 Control de Flujo.

El cual es un mecanismo que previene al emisor acerca de una transmisión de datos más rápida de la que el receptor puede manipular.

7. Algoritmos TCP.

7.1 La conexión TCP.

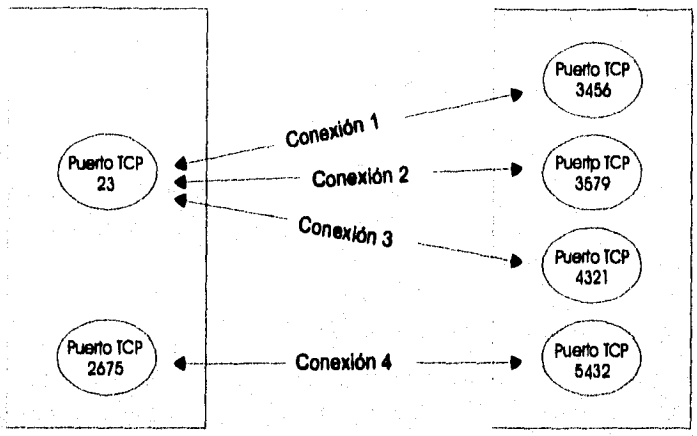
La conexión TCP consiste de dos puntos finales .Cada punto final consiste de un puerto particular en un host particular. Cada punto final consiste de un par:

(dirección IP, número de puerto TCP)

Una conexión es un par de puntos finales:

((dirección IP, número de puerto TCP), (dirección IP, número de puerto TCP)).

La Conexión TCP



Cada conexión es unica y consiste en dos puntos finales

Conexión1: ((Host A, TCP puerto 23), (Host B, TCP puerto 3456))

Conexión2: ((Host A, TCP puerto 23), (Host B, TCP puerto 3579))

Conexión1: ((Host A, TCP puerto 23), (Host B, TCP puerto 4321))

Conexión1: ((Host A, TCP puerto 2675), (Host B, TCP puerto 5432))

El concepto de conexión es importante por que nos permite por una puerto local servir a varios puertos remotos en forma concurrente. Esta es la base de la aplicación del modelo Cliente-Servidor, que es frecuentemente usado en redes.

7.1.1 El modelo "Byte Stream".

TCP trata la conexión como una corriente bidireccional de bytes. Esto no impone una estructura sobre la corriente de bytes, tampoco conoce acerca de registros o conjunto de datos. A cada byte de cada una de las direcciones le es asignado una única secuencia de números.

Cada una de las conexión establece una secuencia con un número diferente de cero como parte inicial de la "three-way handshake" que establece la conexión entre los procesos que se están comunicando. En adición cada lado establece una inicialización de ventana durante la "three-way handshake"; el tamaño de la ventana es igual al número de bytes que está preparado a recibir sobre la conexión.

Cuando TCP recibe exitosamente un paquete, es enviado un reconocimiento al emisor. Este reconocimiento le indica al emisor que el próximo byte a recibir es el byte "XX", en otras palabras, le indica que ya recibió todos los bytes en orden excepto el que espera recibir. El reconocimiento no ocurre inmediatamente después de que el paquete es

recibido. El retardo del reconocimiento le permite a TCP enviar un reconocimiento de varios paquetes recibidos, reduciendo el overhead del protocolo en la red.

El lado del receptor de TCP también le comunica su actual ventana al emisor, reflejando los cambios en la disponibilidad de área de datos, para los datos que están llagando. El tamaño de la ventana es igual al número de bytes que el receptor autoriza al emisor a transmitir sobre la conexión, y representa el espacio disponible en el buffer de la máquina receptora. Cuando los datos llegan desde el emisor el posible espacio de buffer decrece. Cuando la aplicación local acepta los datos, el espacio disponible en el buffer aumenta.

7.1.1.1 El paquete TCP

Al igual que UDP, un paquete TCP viaja en la porción de datos del datagrama IP. El paquete TCP está dividido en un encabezado y en una porción de datos. El encabezado consiste de información que TCP necesita para realizar sus tareas. TCP tiene un monto grande de información de control, especialmente en comparación de UDP,

Campo	Descripción
Source Port	A través de la direcciones IP de los hosts. estos puertos identifican
Destination Port	la conexión TCP a cual este paquete pertenece.
Secuence Number	Identifica la posición de los datos en este paquete en la corriente de bytes. Este número de secuencia es usado para ordenar los datos.
Acknowledgment number	El número de secuencia del próximo byte de datos que TCP espera recibir del emisor. En otras palabras, TCP conoce todos los datos que han llegado, excepto este byte (el próximo a recibir).
Data offset	El apuntador a donde los datos empiezan en este paquete. Esto es necesario por que el campo de Options puede varias en longitud.
URG	(Urgent) SI esta bandera es activada, indica que el apuntador urgent está siendo usado en este paquete; esto es, el paquete contiene un dato urgente.
ACK	(Acknowledgment) Si esta bandera es activada indica que el campo de "Acknowledgment number" es válido para este paquete.
PSH	(Push) Si esta bandera está activada se reescribirá el buffer de TCP y causa que el paquete sea enviado inmediatamente.
RST	(Reset) Cuando esta bandera está activada en un paquete que TCP está recibiendo, causa que la conexión sea inicializada , esto significa que ha ocurrido un error.
SYN	(Synchronize) Cuando esta bandera está activada significa que dos host están abriendo una conexión y sincronizando la secuencia de números.
FIN	(Finish) Cuando esta bandera está activada significa que dos host están cerrando una conexión TCP.
Window	Este campo indica cuantos bytes puede aceptar un host, esto es cuanto espacio en el buffer está disponible.
Checksum	Este campo verifica que el encabezado y los datos lleguen intactos.
Urgent pointer	Puntos a la posición del byte en el flujo de datos justo después del final del dato urgente.

Options	La única opción que es implementada es la opción de "Maximun Segment Size (MSS)", la cual indica el paquete más largo que va a ser aceptado.
Padding	The option are padded to an even 32 bit boundary

7.1.2 Estableciendo una conexión TCP

7.1.2.1 "Three -way Handshake".

Los procesos que se comunican utilizando TCP deben establecer una conexión término-largo entre procesos, en la cual TCP automáticamente secuencía y valida los datos que se están moviendo de un proceso a otro. Para que dos procesos inicien una conexión TCP se necesita que intercambien información para iniciar esta conexión. El proceso para establecer esta conexión con el protocolo TCP se denomina "three-way handshake".

Cuando un proceso quiere abrir una conexión TCP, TCP envía un paquete al destino designado (host, port) para está conexión, con el bit SYN encendido, cuando el nodo destino recibe este paquete envía al receptor un paquete con el bit ACK encendido (que reconoció el paquete con el bit SYN encendido) y el bit SYN encendido también. Cuando el emisor recibe este paquete envía nuevamente un paquete al destino con el bit ACK encendido, reconociendo que recibió el anterior paquete con el bit SYN encendido. Ahora ambos lados de la comunicación (emisor- destino), saben que ambos están listos para el intercambio de datos.

7.1.3 Iniciando una conexión TCP

7.1.3.1.1 Proceso para la Secuencia Inicial de Números.

Para asegurarse que ambos lados de la conexión están listos para comunicarse, el "three-way handshake" establece la secuencia inicial de números para la comunicación. Recordando que la conexión TCP es una corriente bidireccional de bytes y que cada dirección (emisor - receptor, receptor - emisor) tiene su propia secuencia de números.

El bit SYN es tratado como el primer byte de datos en cada dirección. El emisor asigna su secuencia de números, la cual debe ser reconocida por el receptor.

Por ejemplo:

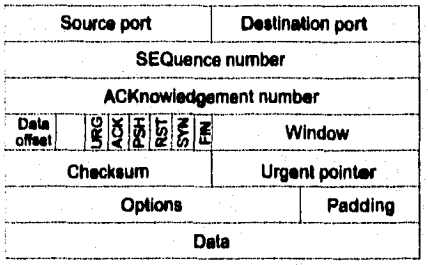
A (emisor) y B (receptor) van a establecer una conexión TCP. A envía un paquete con el bit SYN encendido y el número de secuencia 1000.

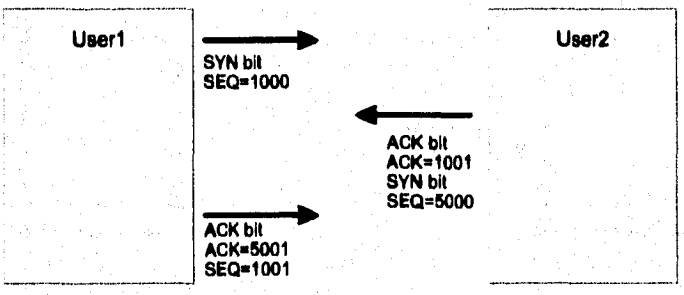
B recibe el paquete y regresa un paquete con el bit ACK encendido y el número_ ACK de reconocimiento 1001. En adición B envía este paquete con el bit SYN encendido y el número de secuencia 5000.

A recibe el paquete y regresa un paquete con el bit ACK encendido y el número_ACK de reconocimiento 5001.

Hasta este momento ambos lados de la conexión (emisor-receptor) han reconocido el bit SYN, y la conexión está lista para empezar el intercambio de datos.

Estableciendo una conexión TCP: Three-way Handshake

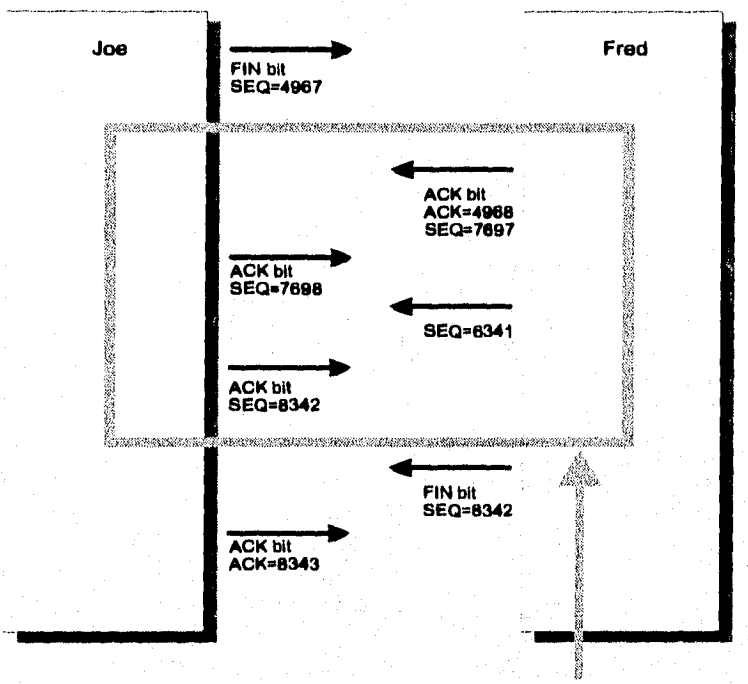




7.2 Cerrando una conexión TCP

De igual manea que al iniciar una conexión TCP, para cerrar un a conexión TCP, ambos lados de la conexión deben estar listos para cerrar la conexión. Recordando que la conexión TCP es un flujo bidireccional de bytes. Como este flujo de información es independiente para cada lado de la conexión, se puede dar el caso de que alguno de los lados no este listo para terminar el intercambio de información y cerrar la conexión. Para cerrar la conexión TCP, el algoritmo *tree-way-handshake* ha sido modificado. De igual forma que el bit SYN es tratado como el primer bit de datos de una conexión, el bit FIN es tratado como el último bit de la conexión.

Cerrando una conexión TCP:
Three-way Handshake



7.3 Restablecer una conexión TCP (Reset)

EL método utilizado por TCP cuando ocurre alguna anomalía en una conexión TCP, la conexión es restablecida. Se envía un paquete con el bit RST encendido. Cuando alguno de los lados recibe este paquete, TCP deja de transmitir datos y restablece la conexión. TCP no realiza una ordenación o recobro de los datos perdidos. La única alternativa es reiniciar la conexión. El estado de la anormalidad ocurrida es enviada al proceso o programa aplicación que detecto el reset. El restablecimiento de la conexión TCP es manipulado por el programa de aplicación.

7.4 Reconocimiento de Datos y Retransmisión

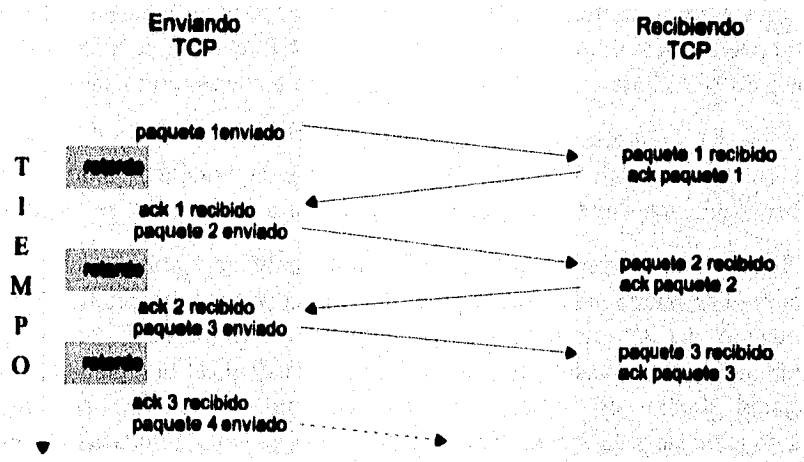
7.4.1 "Sliding Windows".

Un concepto utilizado por TCP es *Sliding Windows*, el cual implementa un esquema de reconocimiento y retransmisión de datos. Bajo TCP cada paquete enviado (de hecho, cada byte enviado) debe ser reconocido y tiene que haber sido recibido. Si TCP tiene que enviar un reconocimiento por cada uno de los paquetes enviados, el retardo que causaría en la red sería considerable. Mediante *Sliding Windows*, se implementa un esquema en el cual el emisor puede enviar varios paquetes al mismo tiempo y solo

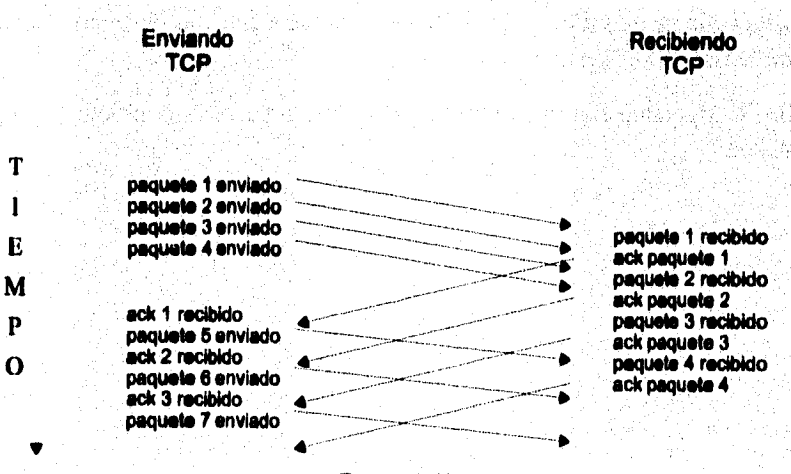
esperar un sólo reconocimiento por todos los paquetes, en vez de esperar un reconocimiento individual por cada uno de ellos. El tamaño de la ventana es de hecho, la cantidad de buffer disponible en cada lado de la conexión.

El siguiente ejemplo ilustra las dos diferentes conexiones; una implementando el esquema Sliding Windows y la otra sin ésta implementación.

Reconocimiento de Datos:
Sliding Windows



Tamaño de Ventana=1 paquete



Tamaño de Ventana=4 paquetes

A cada byte que fluye dentro de una comunicación TCP le es asignado un número de secuencia. El número inicial de una secuencia debe ser diferente a cero, y este número se incrementa con cada conexión TCP que inicia. Esto ayuda a TCP a desechar los paquetes de conexiones anteriores.

TCP recibe los paquetes de su análogo en la capa de red del host con el cual está manteniendo la conexión, el cual le envía un reconocimiento por los paquetes recibidos. La forma de trabajar el reconocimiento de paquetes es el siguiente: el host (emisor/receptor, según sea el caso) envía dos números de secuencia - el número de secuencia reconocimiento del último paquete recibido y reconocido (ACK) y el número de secuencia del paquete que espera recibir (número de secuencia ACK + 1). Esto es conocido como C*umulative Acknowledgment*, es denominado así por que este número secuencia de reconocimiento ACK representa el cúmulo de los paquetes recibidos y reconocidos desde que se inició la secuencia

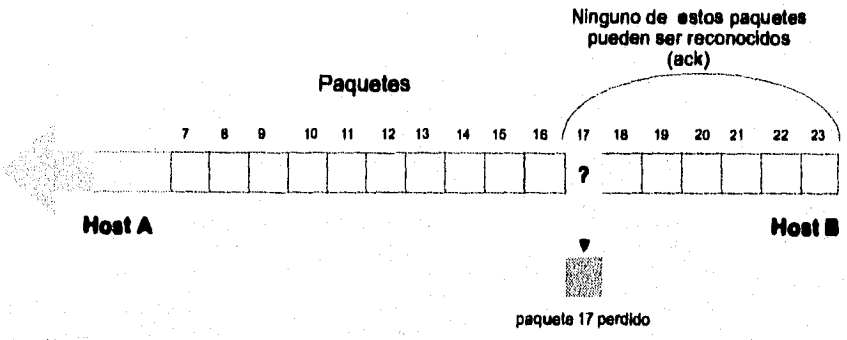
Si un paquete enviado nunca llego al host receptor, el número de secuencia ACK para este paquete llega fuera de orden o no llega. TCP retransmite a partir del número de secuencia ACK recibido +1. TCP implementa varias técnicas retransmisión.

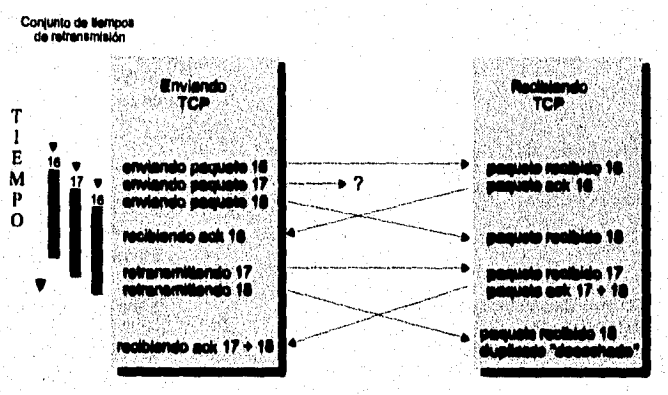
Una de las técnicas es la implementación de un *timer*. Este timer es inicializado cuando el dato es enviado y su valor es predeterminado e igual para todos los paquetes. Si este timer expira antes de que llegue el reconocimiento del paquete, se retransmite.

El proceso de reconocimiento en la red es ambiguo, debido a que no tiene forma de saber si el reconocimiento que llegó es del primer paquete transmitido (por un retardo en la red) o por la retransmisión del paquete (el primero se perdió). El método anteriormente descrito de sintonización del timer no calcula el tiempo de retransmisión de un paquete dentro del timer. Este contratiempo se elimina por medio de la implementación de otra técnica denominada *backoff*. Esta técnica ajusta el tiempo de retransmisión cada vez que una retransmisión ocurre, incrementando el timer destinado para el paquete (usualmente al doble). Eventualmemte TCP empieza a recibir ACK por los paquetes que no han sido retransmitidos y vuelve a calcular el timer, alterando el valor del timer de retransmisión acordado.

En la siguiente figura se muestra un ejemplo de reconocimieto y transmosión de datos en TCP.

Como afectan los paquetes perdidos el bit Ack, y Retransmisión



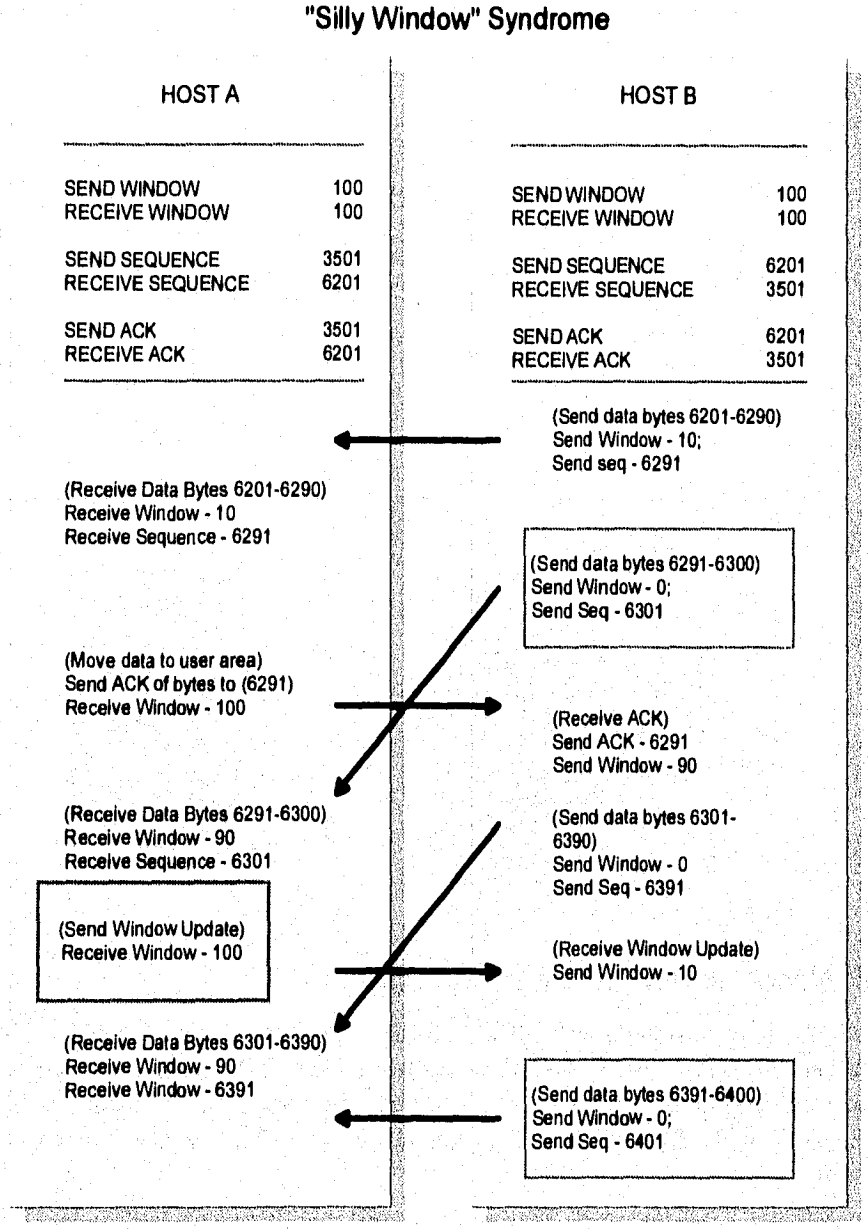


7.5 *Mecanismos de control de Flujo.*

Los mecanismo de control de flujo de TCP, tienen como función evitar que un host transmita datos más rápido que el receptor pueda efectivamente procesarlos. Esto se realiza al determinar el tamaño del buffer disponible en el campo Window del encabezado TCP. Este dato es el mismo al que se refiere en el concepto de Sliding Window.

7.5.1 Síndrome "Silly Window"

El síndrome *Silly Window* se presenta en una conexión el cual se transmiten alternativamente paquetes muy largos o paquetes muy cortos. Esto, decrementa el rendimiento de la red, debido a que cada paquete debe llevar encabezado y además la información del usuario.



Existen dos métodos para evitar la existencia del síndrome Silly Window en una conexión TCP.

Receptor. Este primer método se aplica al receptor, el cual se abstiene de mandar una actualización sobre el tamaño de la ventana, hasta que se tenga al menos el 35% de espacio libre del espacio total del buffer disponible.

Emisor. Esta técnica se aplica al emisor, el cual el abstiene de enviar un paquete que no es al menos 35% de la ventana el receptor (varias condiciones especiales pueden sobreescribir esto).

7.5.2 Algoritmo Start-up

Una de las limitaciones de los algoritmos descritos anteriormente es que basan las soluciones de los problemas de TCP en el tamaño de ventana y no en otros alternativas. Otro de los problemas que se presentan es la diferente medida de los medios por los cuales se realiza la transmisión (cuellos de botella). Para solucionar este problema se ha implementado un algoritmo desarrollado por Berkley 4.3 UNIX (TAHOE). El algoritmo se denomina *Slow-start-up,* El objetivo de este algoritmo es determinar el espaciamiento entre la transmisión de datos o de paquetes, para evitar el congestionamiento ocasionado por los cuellos de botella. Esta medida es puesta cuando se inicia la conexión en paquete SYN, en el *Maxime Segment Size - MSS* .

8. Programación de la Red.

En la presente sección se describen dos formas de programar interfases que son usadas para codificar las aplicaciones de red. Estas interfases no son parte de TCP/IP. Son programas (métodos) de acceder a los protocolos y son sistemas independientes. Las interfases de programación unen la Capa de Transporte y la Capa de Aplicaciones. Los diferentes sistemas operativos utilizan distintos tipos de interfases. Pero el modelo de capas de red se sigue comportando (interactuando de la misma forma).

La descripción presentada será en cuanto a las bases de esos programas interfases, para referirse a un sistema operativo específico se tendrá que revisar la documentación específica para el caso particular. En esta descripción cuando se tenga una referencia particular será UNIX.

8.1 El modelo Cliente-Servidor.

La mayoría de la aplicaciones TCP/IP están construidas bajo el concepto Cliente-Servidor. En este modelo existe un programa servidor el cual está listo para realizar los servicios que le soliciten. También existe un programa cliente que es el que realiza la petición de los servicios del programa servidor. Cuando el servidor recibe una petición, realiza el servicio y regresa e resultado al programa cliente.

El programa servidor es referido como "pasivo", debido a que generalmente se espera que le requieran un servicio. La mayoría de los programas servidores son implementados como *deamons* (programas que son inicializados cuando el sistema es inicializado -booted y se ejecuta en modo background esperando por una petición.

El programa cliente es "activo". El cliente inicia su actividad al realizar una petición al servidor. La interfase de usuario para obtener algún servicio es usualmente un programa cliente (por ejemplo el comando *ftp* o *telnet).*

8.2 Sockets.

El concepto socket es la base del sistema de Entrada/Salida implementado en el sistema operativo 4BSD UNIX. Los sockets son tratados de igual forma que un archivo en UNIX. Como un archivo de E/S, un socket es basado en cuatro operaciones simples: abrir, leer, escribir y cerrar. EL concepto de socket es muy general.

Un socket puede verse como un punto final de comunicación. Esto es especialmente aplicable a programación de redes con sockets. En TCP/IP se puede pensar que un socket es la mitad de una conexión TCP. Recordando que la conexión TCP consiste en dos puntos finales:

((Dirección IP, Puerto TCP) , (Dirección IP, Puerto TCP))

Cada uno de esos puntos finales se convierte en una socket.

Operación de los Servidores.

Un programa servidor realiza (abre) un socket por medio de las siguientes llamadas al sistema.

Socket Posiciona un estructura de datos del socket

bind Asocia el socket con una dirección local (puerto)

listen Anuncia que el socket (puerto) está listo a aceptar una petición.

accept Espera hasta que el cliente se conecta al puerto, entonces acepta la conexión.

Operación de un Cliente.

el cliente realiza una actividad open sobre un socket haciendo las siguientes llamadas al sistema:

socket Posiciona la estructura de datos del socket

connect Trata de conectarse al puerto del servidor

Por ejemplo en un socket TCP, la llamada *connect* causa que TCP inicie el proceso three-way handshake para abrir una conexión, esto es, el cliente envía un paquete con el bit SYN encendido y un número de secuencia inicial. La llamada *accept* sobre el servidor causa que el segundo paquete de three-way handshake, sea devuelto al cliente (SYN, ACK). Después que el cliente ha reconocido el paquete, la conexión es abierta y el cliente y el servidor puedan realizar operaciones *read* y *write* sobre el socket.

Puertos Conocidos.

La llamada *connect* revela que el cliente debe conocer dos parámetros antes de poderse conectar al servidor: la dirección IP del servidor y el número de puerto del servidor.¿Cómo puede el cliente conocer el puerto del servidor por el cual se va a comunicar?

El concepto de "puertos bien conocidos" resuelve este problema. Estos bien conocidos puertos son números que están reservados para ciertos servicios. Los números de puertos del 0 al 1023 están reservados para servidores "estándar". Estos números de puertos son asignados por las autoridades de Internet y están listados en el RFC 1060. En los host en el archivo /etc/services se encuentra una lista similar al RFC 1060. Este archivo está disponible para asociar (mapear) un servicio con un número de puerto.

getservbyname por medio de esta llamada al sistema se obtiene el número de puerto de un servicio, esta llamada al sistema revisa el archivo /etc/services

bind la llamada al sistema bind sirve para asociar un programa servidor su número de puerto. El cliente no necesita asociar a algún puerto específico, en vez de esto, deja abierto el número de puerto y deja que el sistema escoja uno disponible, no reservado número de puerto (alguno más grande que 1024) uando se realiza la llamada al sistema *connect*

8.3 Remote Procedure Call -RPC y eXternal Data Representation -XDR

RPC es otra forma de programar aplicaciones de red. RPC es un conjunto de llamadas. RPC simplifica la programación de red, debido a que el programador no tiene que estar enterado de la estructura de la red. La rutina -RPC hace este trabajo. RPC implementa un sistema de comunicación lógico entre cliente-servidor.

El modelo RPC es similar a las demás llamadas, en un procedimiento local el programa transfiere el control a un procedimiento, el procedimiento se ejecuta y regresa un resultado y el control del programa al procedimiento original. RCP trabaja bajo la misma idea solo que el procedimiento en el que se hace la llamada y el procedimiento al cual hace referencia pueden residir en diferentes hosts en la red.

XDR es un formato de representación de datos independiente de la arquitectura de la máquina en la cual resida. Por medio de este formato de datos, éstos pueden ser transportados por la red sin importar la arquitectura de los host. RCP está construido sobre XDR, es decir las aplicaciones programadas en RPC utilizan el formato XDR para manipular los datos. Estos estándares fueron desarrollados por Sun Microsystems.

8.3.1 Portmap

Los clientes RPC tienen el mismo problema que se presentaba con los sockets: se debe conocer la dirección IP y el número de puerto del host con el cual se quiere comunicar. Un servicio de red denominado *Portmap* ha sido desarrollado para sumar la resolusión de este número de puerto para aplicaciones RPC.

EL servidor portmap asocia un bien conocido puerto 111 sobre todos lo hosts. De esta forma los clientes pueden hacer una llamada RPC al servicio portmap a algún host y requerir el número de puerto para un servicio específico.

9. Aplicaciones.

La capa de aplicación provee servicios de alto nivel, los cuales utilizan los protocolos de transporte para comunicarse con otros hosts sobre la red. Los servicios de la capa de aplicación con los siguientes:

- Acceso Remoto a Terminales
  - TELNET
  - rlogin
- Transferencia de Archivos
  - FTP
  - rcp
- Correo Electrónico
  - SMTP
- Acceso Transparente a Archivos
  - NFS

9.1 Telnet.

TELNET es un protocolo de terminal remota que es un estándar de los protocolos TCP/IP. TELNET le permite a un usuario conectarse a una máquina remota por medio de una conexión TCP. El hecho de que un usuario este introduciendo comandos desde la red y no desde una terminal conectada directamente es transparente al sistema operativo del host remoto.

TELNET es implementado como una aplicación cliente-servidor. El cliente es el programa que el usuario ejecuta (por medio del comando Telnet). EL servidor es un programa *deamond* (usualmente denominado *telnetd*) en el host remoto.

El protocolo TELNET realiza tres servicios básicos:

1.Define la interfase Network Virtual Terminal -NVT para sobreponerse a las dependencias del sistema. NVT permite al cliente y al servidor implementados sobre diferentes sistemas operativos comunicarse definiendo una interfase común.

2.Provee un proceso para que el cliente y el servidor puedan negociar sobre varias opciones (el conjunto de caracteres a usar, por ejemplo).

3.Trata ambos puntos de la conexión simétricamente. Esto significa que el cliente no debe estar necesariamente conectado a una terminal físicamente. Esto permite hacer Telnet a otra máquina, y desde ahí otro Telnet, etc.

9.2 Remote Login -rlogin

rlogin es otro servicio de terminal remota que opera sobre una conexión TCP. rlogin no es parte de los protocolos estándares de TCP/IP. Existen dos diferencias principalmente entre Telnet y rlogin:

La primera, es que Telnet es independiente del sistema operativo . pone los datos dentro del formato NVT permitiendo al servidor de Telnet manipular los datos a un formato que el sistema operativo local pueda entender. En contraste a esto, rlogin solo corre en hosts UNIX. Como rlogin solo corre en hosts UNIX, una sesión con rlogin aprovecha las ventajas del sistema, tales como el direccionamiento de la entrada y salida estándares.

La segunda diferencia es que rlogin soporta el concepto de *trusted hosts* a través del uso de los archivos *hosts.equiv* y *.rhosts*. A los trusted hosts le es permitido acceder a un hosts remoto sin necesidad de validar un password.

Basado en el principio de trusted hosts, se han desarrollado utilierías como *Remote Shell -rsh*. rsh ejecuta un comando en un host remoto.

9.3 File Transfer Protocol -FTP

FTP es un estándar de los protocolos de TCP/IP que realiza la transferencia de archivos entre host a través de la red. FTP es implementado como una aplicación cliente-servidor. El usuario ejecuta el programa cliente (por medio del comando FTP), y el programa servidor es un *deamond -ftpd*. FTP provee algunos servicios importantes, incluyendo los siguientes:

•Una interfase de usuario interactiva

•Especificación del formato. El usuario puede seleccionar la transferencia de datos en forma binaria o texto, y en forma de texto ASCII o EBCDIC.

•Autenticación. El usuario debe conectarse por medio de una cuenta disponible en el host remoto y validar la cuenta por medio de una palabra clave (passwd).

9.4 Remote Copy -rcp

rcp es otra servicio popular de transferencia de archivos disponible. rcp, al igual que rlogin no es un estándar de los protocolos TCP/IP. Las diferencias entre rcp y ftp son muy similares a las diferencias entre telnet y rlogin.

FTP es independiente del sistema operativo, mientras rcp es un servicio basado en UNIX. Debido a que rcp manipula unicamente archivos UNIX, éstos conservan sus atributos y permisos. Al igual que rlogin rcp soporta el concepto de trusted hosts. De hecho, rcp requiere el uso de trusted hosts debido a que no proporciona el prompt para password.

9.5 Simple Mail Transfer Protocol -SMTP

El correo electrónico es uno de los servicios de red más ampliamente usados. Existen dos estándares de para proveer servicio de correo electrónico sobre TCP/IP en Internet.

1.Formato Mail message, especificado por el RFC 822

2.Mail Exchange, especificado por el Simple Mail Transfer Protocol, en el RFC 821

El estándar *mail message* definido en el RFC 822 es muy simple, consiste de un encabezado y de un cuerpo. El encabezado consiste de palabras clave y valores separados por punto y coma. Algunos ejemplos de palabras claves son TO, FROM, SUBJECT, etc. El formato del cuerpo es determinado por el emisor, dando flexibilidad al correo electrónico.

SMTP es una aplicación implementada como cliente-servidor. Típicamente el cliente establece una conexión TCP con el servidor y espera. El servidor envía el mensaje READY FOR MAIL. Cuando el cliente recibe este mensaje, envía el mensaje HELLO y se identifica a si mismo. Después de esto el cliente envía el mensaje. Cuando la transferencia del mensaje ha sido terminada, se puede intercambiar el papel con el servidor. Este intercambio de papeles, permite al servidor convertirse en cliente, tomar el control de la conexión y transferir mensajes. En todos los casos el receptor debe reconocer los mensajes. El emisor no debe borrar su copia del mensaje hasta que no haya recibido el reconocimiento del mensaje.

Varias de las partes importantes del intercambio de correo electrónico no han sido definidas ni por SMTP ni por RFC 822, y algunas de ellas son:

- La interfase del usuario al sistema de correo
- Cómo o donde es almacenado el correo (spooled)
- Que tan seguido el sistema de correo intenta enviar mensajes
- Por esta razón han sido creadas aplicaciones para leer y escribir correo.

9.6 Network File System -NFS

NFS fue desarrollado originalmente por Sun Microsystems, Inc. NFS provee acceso transparente a los usuarios a sistemas de archivos sobre hosts remotos. NFS utiliza el Remote Procedure Call RPC a través del eXternal Data Representation XDR. EL uso de RPC y XDR hacen posible la implementación de NFS en una amplia variedad de diferentes sistemas operativos.

NFS es implementado como una aplicación cliente-servidor. Un sistema cliente monta un sistema de archivos desde un servidor. Un sistema servidor concede acceso al sistema de archivos exportándolo al cliente con modos de acceso específico. Es posible para un host desempañar ambos papeles, como cliente monta un sistema de archivos desde un servidor remoto, y como servidor, exporta un sistema de archivos a clientes remotos, al mismo tiempo.

Después de que NFS es configurado, el sistema operativo pasa la petición de acceso al cliente NFS en vez de pasárselo a las rutinas de acceso del sistema de archivos local. El cliente hace una llamada RPC a través de la red al servidor , quien realiza la autenticación de la petición y realiza el servicio. NFS es un protocolo *stateless*, cada transacción se realiza individualmente. Esto hace posible a NFS recobrarse fácilmente de interrupciones en la transacción de clientes o servidores NFS. La disponibilidad debe ser sacrificada en nombre de la recuperabilidad en el proceso. NFS utiliza el protocolo de transporte UDP, la mayoría de la aplicaciones NFS no utilizan el checksum opcional de UDP.

Apéndice B

IPX/SPX -Internetwork Packet Exchange/Sequenced Packet Exchange.

Introducción

En el presente apartado, se describen los diferentes protocolos que interactúan en una red Netware. La pila de protocolos que interactuan con IPX/SPX, no está formalmente definida -dividida en capas. Por la similitud de las características y funciones de éstos protocolos con TCP/IP, sólo se explican los protocolos propietarios de Novell y sus funciones.

Se contempla dentro de éste trabajo el protocolo IPX/SPX, por que en la muchas de las redes locales que se conectan a la RedUNAM está instalado Netware Novell, por lo que consideramos importante documentar éste protocolo.

XNS Y Novell.

Los protocolos de red de Novell son derivados del *Xerox Network System* -XNS, de "Xerox Palo Alto Research Center" - PARC. XNS forma la base de varias redes (Banyan´s VINES, 3Com´s 3+, Metaphor´s Products). XNS define una serie de protocolos que cubren las arquitecturas de redes. Como muchas arquitecturas de redes, en ésta, los enlaces de datos son incorporados por referencia - los fabricantes deben soportar una amplia gama de enlaces de datos (Ethernet, ARCnet. X.25 y conexiones asíncronas como Módem y líneas telefónicas).

Novell ha tomado de la arquitectura de red de XNS la parte superior de su modelo e incorporado su propia arquitectura. En la capa de red de XNS el protocolo *Internetworking Datagram Protocol* -IDP, ha sido adoptado por Novell y renombrado como IPX (el cual es virtualmente idéntico a IDP).

Novell también utiliza variantes de dos protocolos más de la capa de red de XNS, el *Sequenced Packet Protocol* -SPP y *Packet Exchange Protocol* -PEP. PEP ofrece un servicio de más bajo nivel que SPP (o el que ofrece su análogo en la versión de Novell *Sequenced Packet Exchange* -SPX).

Algunos de los protocolos de las capas superiores de XNS no han sido adoptados por Novell. Por ejemplo: en vez de adoptar el protocolo *Clearinghouse* el cual es utilizado para encontrar nombres de servidores de red , Novell ha desarrollado su propio protocolo *Service Advertisement Protocol*, tampoco ha sido adoptado *el XNS Mail Transport Protocol*, Novell utiliza para estos servicios el *Message Handling Service* de "Action Tecnologies". Para llamadas remotas Novell ha implementado *Netwise's Remote Procedure Call*, en vez de *Xerox's Courier Protocols*.

En otros casos los usuarios han adoptado protocolos equivalentes como *Postcript* en vez del estándar *Interpress* usado por XNS:

No se puede decir que no ha sido una buena idea no implementar la versión completa de XNS. XNS forma la base de las capas de la arquitectura de redes para construir otros servicios de red tales como acceso a impresoras, datos o fuentes de comunicación.

Capa de Red

En la capa de red es donde se interconectan varias LANs. La capa de red es un módulo de software que hace uso de la capa de enlaces de datos para enviar datos a una particular LAN.

A la Capa de Red le conciernen dos puntos importantes:

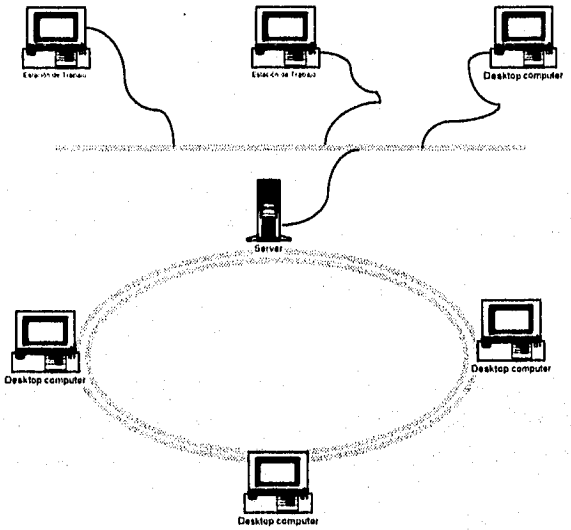
Primero, Considera diferentes tipos de direcciones para diferenciar nodos que residen sobre diferentes enlaces de datos. Conocer sobre que red reside le permite a la capa de red entre dos instancias de la misma dirección local (dos nodos ARCnet con la dirección 254). La capa de red preserva la dirección local de los nodos, de esta forma cuando los nodos envían información, no tiene que obtener su dirección local. A esta dirección la Capa de Red le suma otra dirección -*la dirección de red*. La dirección completa de un nodo en este nivel es ahora la dirección de red + la dirección local del nodo en esa red.

Segundo, la capa de red obtiene por si misma las diferentes rutas disponibles para alcanzar una particular red, sobre la Internetwork. Los nodos que conocen sobre la topología de la red son conocidos como *enrutadores*.

Internetwork Packet Exchange (IPX)

Un nodo puede ser un enrutador o un nodo no enrutador (conocidos como nodos finales). Cuando un nodo enruta paquetes a través de diferentes redes Novell lo denomina

"Bridge" (cabe hacer notar que este *brigde* descrito en la literatura de Novell desempeña funciones diferentes al bridge definido para la capa MAC). Si un servidor está conectado a dos redes diferentes , entonces es un nodo enrutador. Hablando de conexión de dos o más redes diferentes -concepto conocido como *Internetwork*, el enrutador tiene dos o más direcciones, una para cada red en la cual reside. Para cada computadora su dirección *Internetwork* está compuesta por la dirección local más un número de red.

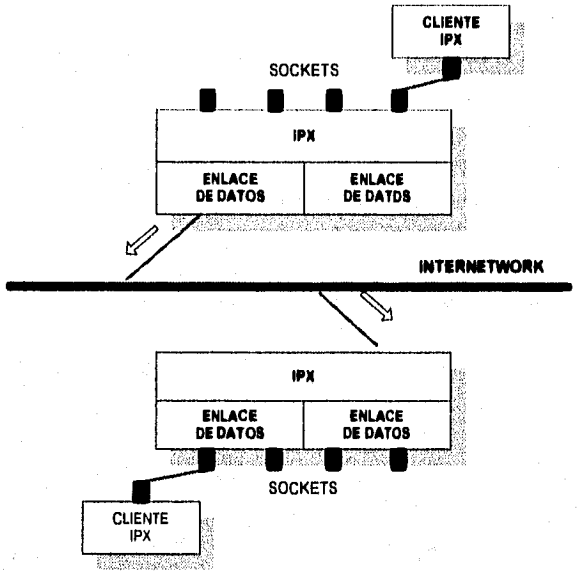


La dirección local es cualquier dirección que el nodo tiene antes de trabajar dentro de *Internetwork*. La dirección de red es fácilmente configurada por software.

Una estación de trabajo, un nodo no enrutador, tiene instalada una simple versión de IPX. Si la dirección destino a la cual se quiere comunicar una estación de trabajo, no está en la red local, envía el paquete al enrutador designado. La versión *server* de IPX, examina el encabezado de IPX, si está dirección es diferente al encabezado de la capa de Enlace de Datos, significa que el paquete debe ser redireccionado. El servidor revisa su directorio de rutas, para decidir por cual de éstas envía el paquete.

Sockets

IPX provee servicios a los programas de las capas de más alto nivel a través del uso de *sockets*. Un *socket* es una dirección de un programa de alto nivel que está usando los servicios de IPX. Esto es similar al servicio local de direcciones que la capa que los Enlaces de Datos utiliza para distinguir entre sus usuarios. El orden para que un cliente IPX se pueda comunicar con su igual en la red es el siguiente: primero, conocer el número de red del host remoto, segundo la dirección local sobre la red remota, tercero necesita conocer el número de socket del programa remoto con el cual se está comunicando.



Un nodo tiene dos direcciones diferentes: una dirección de la capa de red, y una dirección de la capa de Enlace de Datos. En la mayoría de los casos la dirección local del host sobre el Enlace de Datos corresponde a la porción de host local de la dirección Internetwork.

Normalmente el paquete traerá un encabezado de la capa LLC después del encabezado de la capa MAC. En vez de este formato, el paquete IPX viene inmediatamente después de la capa MAC. Esto es una artificio que Novell ha elegido para usar Ethernet. Esta forma no estándar de usar Ethernet puede ser eliminada utilizando la utilería ECONFIG de Novell que forza el uso estándar de protocolos (esta operación toma real importancia cuando la red se está compartiendo con usuarios Ethernet, LAN Manager, DECNet , TCP/IP, etc).

EL encabezado IPX contiene los siguientes campos: dirección fuente, dirección destino, checksum, control de transporte y tipo de paquete.

EL campo *Checksun* solamente analiza cierto tipo de errores generales, como por ejemplo, el que un paquete no haya sido debidamente construido. Si el campo *Checksum* tiene "FFFF" significa que el campo está desactivado -si la red es Ethernet, la cual teóricamente está libre de errores, no es necesario activar este campo.

La longitud de un paquete Ethernet puede ser hasta de 576 bytes, pero en realidad el tamaño del paquete está limitado por la longitud del *frame* definido en la capa de Enlace de Datos.

El campo de Control de Transporte es utilizado para determinar cuantos *Hops* ha realizado el paquete. Cada vez que el paquete es reenviado por un enrutador, es un *hop*.

El último campo es el del tipo de paquete, el cual especifica que protocolo está enviándolo.

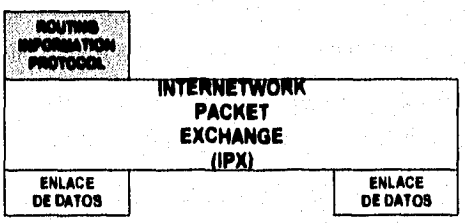
Algunos ejemplos de *sockets* utilizados por los protocolos de Novell son:

1107 -*socket* RIP.

1105 -*socket* NCP

Routing Information Protocol (RIP) y la Internet

El *Routing Information Protocol* -RIP utiliza los servicios de IPX para permitir a los nodos enrutadores intercambiar información y actualizar sus directorios de rutas. Es un protocolo de mantenimiento, el cual es un servicio oculto a los programas usuarios de la red (bases de datos, mensajes del sistema, etc.). RIP es usado para dar mantenimiento a las tablas de rutas.



Otros servicios tales como SPX, SAP y NCP son también clientes de IPX. Estos protocolos de transporte son visibles a los programas que usan la red, y no directamente a los usuarios finales.

Las tablas de rutas tienen cuatro entradas:

- Red Destino
- Puerto enrutador
- Nodo Intermedio
- Hop Count

Red Destino. El número de red destino. Si la topología de la red contempla una Internetwork, existen varias subredes, las cuales son identificadas por un número de red.

Puerto Enrutador. Indica cual Enlace de Datos sobre este enrutador, va a ser usado para llegar a la red destino.

Nodo Intermedio. Es la dirección local del nodo sobre el Enlace de Datos indicado por el Puerto Enrutador. Esta dirección es la dirección del enrutador que redireccionará el paquete a través de la Internetwork.

Hop Count. Indica por cuantas rutas existen en la red antes de alcanzar la red destino. Un Hop Count=16 indica que el destino es inalcanzable (*unreachable*). El límite máximo de Hop Count es de 15, aunque prácticamente esto no es una limitación. Un Hop Count=15 significa se deben recorrer 15 LANs o enlaces a redes para alcanzar el nodo destino lo cual es suficiente para algunas topologías. Basado en la información registrada por el protocolo RIP, el Hop Count indica la mejor ruta actualmente disponible para alcanzar la red destino.

Cada enrutador debe conocer la información para estas entradas para cada nodo de la red. Cuando un enrutador recibe un paquete RIP, examina el campo *Objetct network*. Si ve un *object network* listo en su tabla de enrutamiento, compara la información. Si el host que está enviando el paquete es el mismo que el Nodo Intermedio indicado en la tabla de rutas, esta información contiene una actualización de rutas. Para realizar una actualización el enrutador revisa el Hop Count para ver si ha cambiado . Si el Hop Count=16, indica que el host que envió el *broadcast* puede no alcanzar la red destino.

Si el *broadcast* RIP contiene una nueva red, esta información será sumada a la tabla de rutas, permitiendo a éste enrutador enviar paquetes a este destino (red).

La última posibilidad es que existe una red lista en la tabla de rutas, y el host que envía el *broadcast* RIP es una nodo diferente a los Nodos Intermedios definidos en la tabla de rutas, significa que existe una ruta alternativa para esta red destino. El enrutador verá el Hop Count para comparar si esta nueva ruta es más rápida que la definida en la tabla de rutas para este caso particular. Si es así actualiza la tabla de rutas, para indicar el nuevo enrutador para esta red destino.

Un nodo en una red Novell envía un *broadcast* RIP cada 60 seg. En una red XNS cada 30 seg. En una red Novell cada enrutador que es un "Hop Away" , desde el envío del broadcast RIP conocerá el estado de la tabla de rutas del transmisor. Esto permite al receptor a actualizar su tabla de rutas, y retornará un broadcast periódicamente, de esta forma los cambios en la topología de la red son gradualmente propagados a través de la misma.

Cada entrada de la tabla de rutas tiene asociado un "Timer", el cual indica cuando fue realizada su última actualización. Si el nodo detecta que una entrada no ha sido actualizada (timer=90seg.) esto indica que el enrutador que provee este servicio no está funcionando. Cuando un timer expira, el Hop Count para esta red está puesto en 16. Sólo en caso de falla de servicio temporal, la entrada en la tabla de rutas estará disponible por otros 90 seg, esto asegura que la información acerca de que esta dirección es inalcanzable (*unreachable*) será enviada vía broadcast a los enrutadores vecinos después que los segundos 90 seg. expiren, la red destino será borrada de la tabla de rutas.

RIP utiliza lo que es conocido como "Flat Network Topology". Cada enrutador debe conocer como llegar a todos las otras redes en la Internetwork.

Las redes como DECnet, utilizan un sistema de "Enrutamiento Jerárquico", en el cual colecciones de redes son agrupadas en dominios. Un enrutador local conoce como llegar a todos las redes definidas dentro de un dominio. Si es un paquete destinado a otro dominio, un segundo nivel de enrutamiento es dado a este paquete.

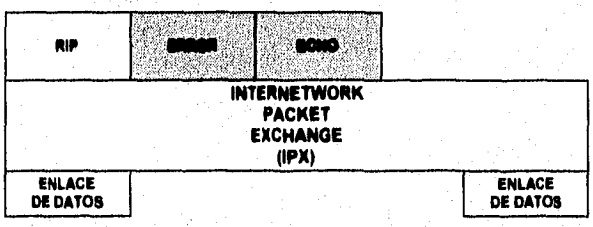
Capa de Transporte

La Capa de Transporte es responsable de mantener una conexión lógica sobre la red -mantener la comunicación entre dos entidades entre dos diferentes computadoras. La Capa de Transporte realiza principalmente dos actividades:

Primero, Poder identificar un software particular que reside en algún nodo de la red. Segundo, Garantizar la entrega de paquetes en la dirección destino

Protocolos ERROR y ECHO.

ERROR y ECHO son dos protocolos adicionales que hacen uso de IPX, los cuales son usados por otros programas para mantenimiento interno. ERROR es usado para informar a un *socket* destino que ha ocurrido un error. ECHO es usado para probar que determinada ruta está trabajando normalmente.



El *header* IPX incluye dos tipos de información destino: una es el tipo de paquete y la segunda es un *socket* destino (esto puede parecer redundante, ya que el *socket* siempre recibe el paquete).

Cuando IPX recibe un paquete, lo envía al *socket* destino. Los diferentes tipos de paquete son: Tipo 1, paquete RIP, Tipo 17 es un paquete NCP, Tipo 2 paquete ECHO, Tipo 3 paquete ERROR.

ERROR y ECHO son protocolos muy generales, en su mayoría estos paquetes son de interpretación, esto es la interpretación la hará protocolos de más alto nivel o los programas usuarios.

Para generar un paquete ECHO, el programa usuario suministrará algún paquete normal a IPX, pero pone el tipo de paquete de ECHO. EL nodo remoto recibe este paquete y lo pasa simplemente en vez de procesarlo.

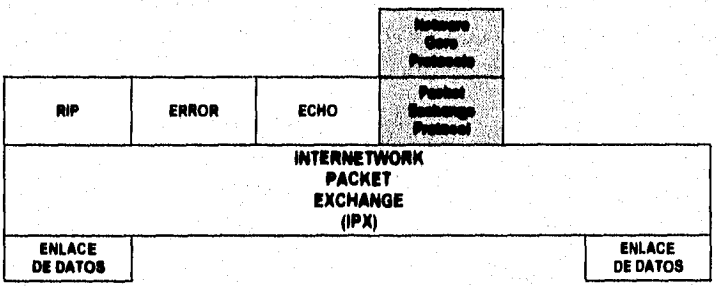
Un ERROR es igualmente simple, permitiendo ser usado en una gran variedad de situaciones. Consiste de un encabezado IPX más un número de error, un parámetro de error y una porción del "offending packet".

EL protocolo RIP interpreta de forma diferente un paquete ERROR, que la forma en que lo interpreta NCP.

Se acostumbra al menos incluir 42 bytes de "offendign data". Esto permite que los 30 bytes del encabezado de IPX más 12 bytes del encabezado de la próxima capa sean incluidos en el mensaje de error, permitiendo al programa receptor cual de sus paquetes causó el problema.

Netware Core Protocols (NCP)

El primer usuario de IPX en una red Novell es el NCP. NCP permite al usuario acceder a datos remotos, archivos de impresión y otras operaciones básicas. NCP utiliza una forma primitiva de protocolos de transporte basado en PEP de XNS. PEP existe, pero no es considerado como un protocolo independiente, si no es considerado dentro de NCP.



Esto en contraste con SPX, otro protocolo de la capa de transporte , el cual provee una interface de propósito general.

NCP permite a los usuarios crear una conexión al servidor. En este intercambio de tráfico de estaciones de trabajo, obtiene la dirección por medio de RIP, una vez que tiene su dirección local, envía un mensaje para crear una conexión al servidor. Esta conexión (*request*) incluye información sobre la negociación del periodo del buffer. Dos nodos que tienen acuerdo sobre un buffer de 1024 bytes, está conectados en la misma red.

El programa NCP, tanto en el servidor como en la estación de trabajo, puede tener varias conexiones activas. NCP asigna una identificación para cada una de estas conexiones. Este ID puede ser diferente para cada lado de la conexión.

Cada paquete dentro del *socket* NCP contiene cuatro campos de información. El primero es un número de conexión, este número indica cual de las conexiones que NCP mantiene activas deberá trabajar con este paquete. El tipo de *request* indica si este paquete es una petición (*request*) o una respuesta (response). Un número de secuencia asignado a esta conexión garantiza el seguimiento lógico de la secuencia de paquetes.

En un ambiente Novell típico la mayoría de los paquetes son una serie de peticiones y respuesta de NCP. NCP divide la conexión en una serie de tareas. Una tarea puede ser una secuencia simple de una petición/respuesta o una secuencia de varias peticiones/respuestas.

NCP utiliza esta división por tareas para asegurarse que la transacción es terminada exitosamente.

Recordando que IPX no garantiza la entrega de los datos. NCP mantiene un "Timer". Cada vez que un paquete de datos es enviado, es puesto el timer. Se espera obtener una respuesta por parte del nodo destino antes que el timer del nodo transmisor expire. Si el *timer* del nodo transmisor expira, se tratará de enviar el paquete otra vez. y en cada envió el *timer* es puesto. Después de una cierta cantidad de veces que el paquete es reenviado, se asume que el nodo destino es inalcanzable y se avisa al usuario.

Cuando no existe intercambio de tráfico entre dos usuarios pero la conexión está activa (esto es, que un usuario está examinando los datos, por ejemplo), el *timer* opera para asegurarse que la conexión está trabajando. Esto es por medio de un intercambio de mensajes "*alive*". La operación "*alive*" tiene el mismo propósito de los paquetes periódicos de RIP -para asegurarse que las fuentes de la red están disponibles. Los paquetes RIP son usados por el nodo destino. Las peticiones "alive" de NCP son para asegurarse que los procesos NCP en el host remoto de hecho existen.

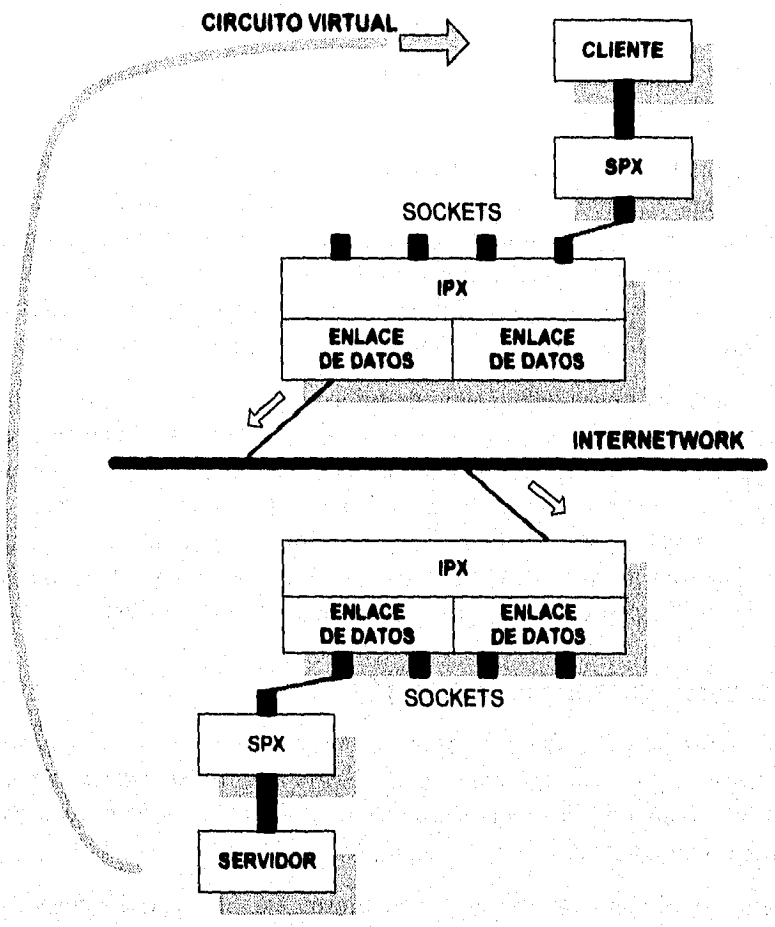
Sequenced Packet Exchange (SPX)

Un tipo de servicios más general es *Sequenced Packet Exchange* -SPX. SPX es una interface de propósito general disponible para desarrolladores de software (terceros). SPX provee una conexión virtual entre dos clientes SPX. SPX construye sobre IPX para garantizar la entrega de datos entre dos programas.

En vez de usar un protocolo petición/respuesta, SPX permite que varias peticiones sean enviadas a un mismo tiempo. a través de una "Ventana" de peticiones no reconocidas. Esta forma de intercambio de tráfico es eficiente por ejemplo en la trasferencia de archivos. En vez de reconocer cada uno de los paquetes, espera a que la "Ventana" (se denomina así al número de paquetes enviados permitidos) se sature. Entonces el nodo receptor envía un reconocimiento por todos estos paquetes.

Una de las funciones de SPX es el recobro de datos duplicados y perdida de datos ocasionadas por errores. Cada lado de la conexión SPX mantiene una secuencia de números para los paquetes que se están enviando. Por cada secuencia de números de un paquete recibido, SPX envía dos números de secuencia, uno para indicar que el anterior paquete ha sido recibido y otro para continuar con la secuencia de la conexión. En adición a estos dos números de secuencia enviados por cada paquete, se envía un tercer número de secuencia "*Allocation*". Este último número indica el número máximo de secuencia que el nodo remoto puede enviar. Este número lo envía el nodo remoto, cuando al recibir paquetes, su buffer se satura, este nodo remoto, cambia el número de "*Allocation*". El

nodo transmisor esperará hasta que el buffer se descongestione para poder enviar paquetes.



Es posible que el nodo remoto envíe un reconocimiento negativo con el número de secuencia. EL reconocimiento negativo -NAK indica que un paquete (un número de secuencia) esperado no ha sido recibido. El nodo que recibe este NAK revisará por el último número de secuencia que ha sido reconocido, y reenviará el número de secuencia en el paquete NAK. En una ventana se permite enviar varios paquetes al mismo tiempo en vez de tenerlos encolados. Sin embargo, cuando ocurre un error, una ventana grande puede causar problemas, esto es, SPX debe conservar una copia de todos los paquetes que no han sido reconocidos, consumiendo gran parte del buffer disponible.

SPX provee un servicio de más alto nivel que el proporcionado por PEP en NCP. SPX libera al usuario del modelo de petición/respuesta, permitiendo a ambos lados de la conexión enviar mensajes al mismo tiempo.

Los clientes se comunican con SPX para abrir una conexión. Entonces, un cliente pone una serie de comandos SPX, diciéndole a su modulo SPX que ha reservado

memoria para esperar la llegada de paquetes. Cuando SPX recibe los paquetes de la conexión abierta, se mueve dentro de su buffer para indicarle al cliente sobre el estado de este espacio.

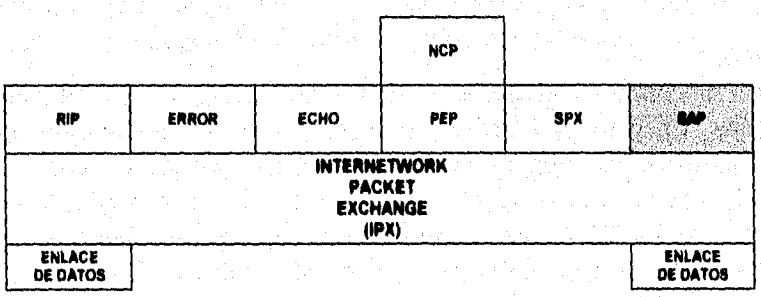
Un cliente puede enviar paquetes de datos suministrándolos a SPX. SPX garantiza la entrega de estos paquetes en el orden en que son liberados. Es posible que este cliente use también los servicios de IPX al mismo tiempo. SPX puede usarse entre servidores y estaciones de trabajo para comunicarse uno con el otro. Los servidores se comunican entre sí por medio de los broadcast de IPX. Desde que un broadcast de IPX es repetido periódicamente, no es necesario garantizar la entrega de los datos.

Service Advertisement Protocol (SAP)

Estrictamente hablando , el concepto de servicios de "advertencia" es un punto concerniente a la capa de aplicación de la red. Sin embargo, dos consideraciones acerca de este punto hacen que valga la pena discutir sobre este punto.

En primer lugar, SAP hace uso directamente de IPX., los servicios que provee este protocolo son normalmente construidos en la cima de los servicios de la capa de transporte, desde que SAP es el último protocolo que hace uso directamente de IPX es discutible que des un servicio de la capa de aplicación.

Segundo, SAP es utilizado por una amplia variedad de otros servicios de red. Pro ejemplo NCP utiliza SAP para encontrar servidores de archivos y de impresión en la red. Los programas de aplicación escritos por terceros desarrolladores utilizan SAP para anunciar los servicios "*custom*" en la red.



SAP anuncia la presencia de servicios sobre la red, existen servicios definidos como por ejemplo servidores de archivos, de impresión de fax, etc. Cada hacer notar las diferentes acepciones del concepto cliente-servidor, que toman los diferentes niveles en el modelo de capas de una red. Por ejemplo, SPX es una cliente de IPX, SPX a su vez provee servicios a sus propios clientes. En este contexto SAP está posesionado en la parte superior de los servicios que provee la capa de red. Por ejemplo los servidores de archivos (clientes de IPX) anuncian su presencia en la red a sus clientes, usuarios sobre una estación de trabajo accesando datos.

Los servicios de un protocolo de anuncios permite a un programa registrar su nombre en la red. De esta forma una petición de SAP preguntará por la translación de ese nombre a un *socket* sobre un nodo particular en la red. Un ejemplo de esto es que un usuario envíe una petición de SAP para todos los servidores de archivos sobre la red , esta petición es un broadcast a todos los nodos de la red y posiblemente todas las redes que están interconectadas. La petición SAP será recibida por los servidores, los cuales mantienen un *socket* para tener presente todas esas peticiones. El servidor busca en su base de datos y revisa si el servicio pedido está presente. Si el servicio está disponible en su lista, contesta con su dirección Internetwork.

Cuando una estación de trabajo inicializa su función dentro de la red, envía un requerimiento al servidor más cercano. Esta petición es contestada por el servidor de archivos con su dirección Internetwork que contiene también un número de red. Cuando la estación de trabajo inicia su actividad en la red, todavía no conoce su número de red. Cuando se envía el requerimiento (un paquete RIP) al servidor, incidentalmente aprende su propio número de red, como fue explicado anteriormente. Posteriormente la estación de trabajo conociendo ya a partir de este momento su dirección de red, por medio de NCP, envía un requerimiento NCP para activar la conexión.

Como se describió en los ejemplos anteriores el requerimiento SAP es un "*query*" (preguntar por el servidor de archivos más cercano). También existen *request query* generales del cual pueden surgir varias respuestas. Otro tipo de request SAP es un *broadcats*. Un servidor envía periódicamente vía broadcast sus servicios a la red, permitiendo a otros servidores actualizar su base de datos de servicios disponibles en la red. Un tipo especial de *broadcats* es el de "*shtdown*", el cual indica a los demás servidores de un servicio específico que no estará disponible. Los *broadcats* de este tipo aseguran que la disponibilidad de servicios están siendo propagados por la red.

Un servicio como el que provee SAP es útil y funcional en redes no muy grandes o complejas. Un grave problema se presenta cuando se tiene que propagar la replica completa de todos los nombres de todos los servicios en una red compleja, la base de datos SAP debe conocer acerca de cada nombre de servicio disponible en la red, esto significa que no existe una real coordinación de todos estos nombres de todos los nodos que componen la red -cada nodo ofrece sus propios servicios.

En nuevo servicio de Novell es el "Netware Name Services", el cual es un servicio global de nombres. Cada porción de la red mantiene un servicio local de nombres, el cual registra los servicios que proveen los nombres registrados. Si un servicio cambia su colocación en la red, el servidor de nombres actualiza los cambios. En un área determinada pueden existir varios servidores de nombres. El mecanismo implementado en este servicio, asegura que la información duplicada en los servidores de nombres sea consistente.

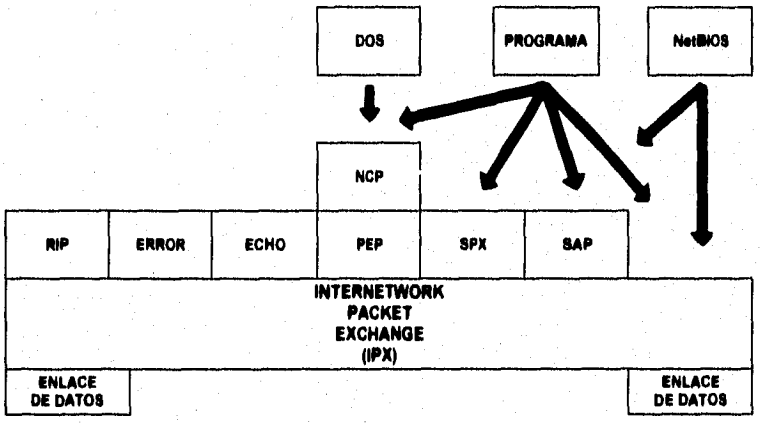
Aplicaciones.

Clientes de IPX y SPX.

Hemos examinado varios protocolos que forman la base del modelo de red de Novell. Un programa puede hacer uso de esos servicios para construir servicios más complejos, en cada una de las capas. Por ejemplo NCP se construye sobre los servicios que proporciona PEP. PEP provee servicios tipo requerimiento/response. NCP en adición a estos servicios, provee la capacidad de que dispositivos remotos sean vistos como locales por los usuarios de la red, y hacer uso directo de ellos.

Un programa puede hacer que diferentes llamadas mezcladas a SPX, NCP, IPX o SAP hagan uso propiamente de cada una de estas llamadas.

Un tipo especial de usuario de este tipo de llamadas es la implementación de NetBIOS en Novell. NetBIOS es actualmente otra arquitectura de red que ha sido implementada en las redes Novell. Novell provee un programa que ve la interface de NetBIOS, pero que de hecho utiliza los servicios de IPX y SAP.



Resumen.

•IPX es una versión de Novell del protocolo original XNS de Xerox. IPX es un protocolo de la capa de red, el cual determina cual enlace de datos va a utilizar para enviar un paquete a un *hop* cerca de su destino.

•RIP es usado para informar a los enrutadores de la red (o *Bridges*, en la literatura de Novell) acerca de cambios en la arquitectura de la red.

•SPX es un protocolo de la capa de transporte que provee un servicio que garantiza la entrega de paquetes en la red. SPX es usado por desarrolladores de terceros.

•NCP usa un simple protocolo de la capa de transporte, conocido como PEP

•SPX permite enviar varios paquetes sin esperar un reconocimiento por cada uno de ellos en forma individual, sino espera un solo reconocimiento por todos ellos. PEP en cambio, espera un reconocimiento individual por cada uno de los paquetes enviados.

•Los protocolos ERROR y ECHO son usados para reportar errores y mensajes a través de la red.

Apéndice C

Reglas Básicas para el Grupo de Administradores de Unix en la Unidad de Servicios de Cómputo de la Facultad de Ingeniería:

I. Sólo podrán existir dos administradores por máquina, uno que será el que tenga mayor experiencia en el manejo y mantenimiento del equipo y otro que aprenderá del primero.

   A. Al administrador de mayor experiencia se le denominará administrador.

   B. Al administrador de menor experiencia se le denominará coadministrador.

II. El administrador se compromete a avisar con al menos un meses de anticipacion cuando desee dejar de lado la administracion de su computadora, con el fin de que la jefatura tome las medidas pertinentes. A si mismo se compromete a dejar todo el equipo en funcionamiento y pleno conocimiento de los detalles necesarios para que el coadministrador pueda retomar las funciones del administrador. Una vez que el coadministrador funja como administrador, se le asignará un coadministrador.

III. El coadministador se compromete a avisar al menos con un mes de anticipación cuando desee dejar de lado su cargo de coadministrador. Del mismo modo se compromete a ayudar al nuevo coadministrador a conocer el equipo por el tiempo que el administrador considere necesario.

IV. En caso de que la renuncia del administrador y el coadministrador se presente al unísono, ambos estarán comprometidos a formar a una tercer persona que pueda tomar la administacion del equipo por el tiempo que la jefatura de la unidad considere necesario.

V. Existirá una única clave de supervisor, misma que se encontrará bajo el control exclusivo del jefe de la unidad. No podrá darse privilegios de supervisor a ninguna clave si no es autorizado por el jefe de la unidad.

VI. No debe existir una contraseña (password) única de administracion para las computadoras. Será responsabilidad del administrador y del coadministrador la elección de la contraseña.

VII. Queda estrictamente prohibido el realizar la administracion de equipo desde terminales remotas. La administracion deberá realizarse siempre desde la consola del equipo. (Queda expresamente prohibido el aceso de root desde una terminal remota).

VIII. El administador y el coadministrador se comprometen a mantener una bitácora de trabajo de su equipo, la bitácora se puede llevar en la misma computadora, siempre y cuando se imprima en papel cada actualizacion de la bitácora, con el fin de contar con un respaldo de la bitácora de administración.

IX. Si se cuentan con claves que tengan privilegios especiales, será necesario llevar una bitácora de las acciones realizadas dentro de esta cuenta.

X. Sobre el grupo de adminstración de Unix de Cómputo Académico de la Facultad de Ingeniería:

Es necesario que en las reuniones periódicas de administracion se encuentren presentes los administradores de cada máquina.

En caso de no poder asistir el administador será necesaria la presencia del coadminstrador en las juntas.

En cada reunion se llevará una minuta.

El comité de administradores estará formado por tantas personas como servidores existan. Esto es: actualmente existen 5 equipos pricipales, por lo cual el comité de administradores estará formado por cinco personas.

Administrador de Cancun

Administrador de Balam

Administrador de Cozumel

Administrador de Tork

Administrador de Alika.

Existirá un presidente de Administradores, y un secretario (mismo que se encargará de llevar la minuta).

Las decisiones se realizarán por votación. Sin embargo el presidente del grupo de administradores cuenta con voto de calidad.

Cuando el punto de vista de los tres o más administradores se contraponga a las desiciones impuestas por presidente del grupo de administradores, se podrá recurrir al arbitraje del jefe de la unidad, para llegar a un arreglo satisfactorio entre ambas partes.

II.- Pasos primarios necesarios para la administración del equipo.

Se debe enteneder el sistema de administracion vigente, hasta poder manejarlo aceptablemente.

Es necesario que los administradores de los equipos principales propongan cada uno un arreglo de archivos, para evaluar entre todos cual resultaría mas eficiente.

Una vez llegados a un acuerdo y un modelo que se considere efectivo por todos, se procederá a cambiar la estructura de los directorios y subdirectorios.

Es necesario definir los tipos de clave que se manejan, así como los atributos con que pueden contar.

Alumnos

Maestros

Investigadores

Personal de la unidad

Claves de cursos de la unidad

Claves de cursos de la facultad

Claves temporales

Otras

Se debe encontrar un mecanismo que force a los usuarios a cambiar su contraseña de acceso tan pronto cuenten con una clave actualizada, indicándoles que deben hacerlo en las cinco máquinas principales de trabajo.

Se debe contar con un diagrama que muestre a los usuarios la estructura de la red, las máquinas principales, e indicaciones para que no se presente una marcada preferencia por una sola máquina (instrucciones para copiar archivos entre directorios, hacer sesiones remotas, transferir archivos entre máquinas y cambiar los atributos de los archivos de usuario).

Sólo se podrá permitir el acceso remoto desde un conjunto bien definido de máquinas.

Si un usuario solicita acceso remoto para alguna máquina quedará a criterio del presidente de Administradores el concederlo o no.

Se deben definir bien las políticas de desactivacion de cuentas. (semestralmente, si no se usa la cuenta por un periodo de 90 días por ejemplo, si se permitirá la existencia permanente de cuentas, etc).

Se deben definir bien los mecanismos de desactivacion de las cuentas (temporal, permanente, etc)

No se podrá contar con cuentas públicas del tipo guest, games o anonymous.

En caso de contar con una cuenta de Demo, no podrá accederse desde un sistema remoto.

Se debe implantar un programa de respaldo de informacion periódico.

Debe hallarse un sistema que impida el abuso de las ventanas en sesiones gráficas. (Por ejemplo, que un usuario tenga abierto al mismo tiempo cuatro directorios, dos imágenes gif, 5 ventanas de trabajo, un reloj, una calculadora y un calendario)

Definir si en la política de respaldo se hará al usuario o al centro responsable de realizar el respaldo de archivos para usuarios.

Es necesario imponer un calendario de administración, y un horario. En caso de que sea molesto para los administradores el contestar preguntas de usuarios mientras se encuentren realizando la administración, se debe proponer un horario de administración el cual se debe cumplir por todos los administradores.

Es importante notificar a los usuarios de este horario de administración una vez que se encuentre definido, de manera que se no se entorpezca la labor del administrador.

Se debe definir el tiempo máximo que una persona puede durar como administrador de máquina, de manera que el coadminstrador tenga oportunidad de ocupar su puesto en la toma de desiciones, y al mismo tiempo se tenga la libertad de preparar mas gente dentro de la administración del equipo. Este relevo de administradores se debe realizar de manera escalonada, con el fin de que en un determinado periodo no se cuente con una mayoría de administradores nuevos.

La diferencia de tiempo entre el egreso-ingreso de administradores de diferentes máquinas será de 4 meses.

El presidente de administradores será elegido por el grupo de administradores. El secretario será elegido por el presidente.

Cada fin de semestre se debe realizar una evaluación de los logros y problemas que se presentaron en la administracion del equipo. Si se consideran necesarios cambios se deberán realizar antes del principio del siguiente semestre.

Se sugiere que los cambios se hagan gradualmente y que se mantenga informado al usuario de los cambios realizados.

Ningun administrador tiene derecho de imponer en su máquina un sistema de administración diferente al que se decida en el grupo de administración. La sanción si el caso se presenta será la suspension de sus funciones como administrador, pero tendrá la obligación de prestar asistencia al administrador que le sustituya por el lapso que el jefe de la unidad y el nuevo administrador consideren pertinente.

Las disposiciones de este documento pueden tratarse con el jefe de la unidad. Con el fin de mejorar o ampliar ideas no consideradas en el documento original.

Apéndice D

Ejemplo documento informativo para usuarios de la Unidad de Servicios de Cómputo Académico de la Facultad de Ingeniería

Cómputo Académico de la Facultad de Ingeniería

Usuario del equipo HP:

Estamos reforzando las medidas de seguridad en los equipos para beneficio de todos, para lo cual necesitamos de tu colaboración.

Uno de los primeros pasos es implementar una política de selección de passwords seguros.

Cuando compras tu clave se te asigna una clave de usuario (username), por ejemplo rla000, y una contraseña (password), por ejemplo kiabe, este último es el mismo para todos los usuarios (!!), como observarás todos pueden entrar a tu cuenta y conocer tu información y por lo mismo destruirla o alterala. Por estas y otras razones debes de cambiar periódicamente tu password y elegirlo adecuadamente.

Dentro de poco se implementará un sistema de control de passwords en el que se te pedirá que lo cambies periódicamente lo selecciones con reglas definidas para evitar que "cualquiera" adivine tu password.

Por ahora se ha establecido que debes cambiar tu password antes del 5 de julio del presente si no lo haces se deshabilitará tu cuenta por tu propia seguridad, las instrucciones para cambiar el password son:

1. En el prompt (por ejemplo $, alika.unam.mx >, cozumel>) escribe la palabra **passwd** y presiona enter.
2. El sistema escribirá **old password:** en seguida debes escribir el password que tienes actualmente (no se despliega) y presionar enter.

3. Si tecleaste correctamente el password y bajo las reglas de seguridad, te aparecerá **New Password:** a lo que debes responder con el nuevo password que hayas seleccionado (en seguida se te indicarán reglas que debes seguir para elegirlo) y presionar enter.

4. Para verificar que tecleaste correctamente aparecerá en la pantalla **Re-enter New Password:** a continuación debes volver a teclear tu nuevo password.

5. Si no envía ningún mensaje de error tu password ha sido cambiado y con el accesarás al sistema a partir de tú próxima sesión, si ocurre algún error vuelve al paso 1 .

Si tienes alguna duda pregunta al asesor en turno o al administrador del sistema.

Las reglas para elegir un password seguro son:

1. No uses tu clave de usuario (username, p, ej. sre000) en ninguna forma (en mayúsculas, en reversa, dos veces, etc.).

2. No uses tu nombre o tus apellidos de ninguna manera.

3. No uses el nombre de alguien conocido de tu familia o de tu novio(a).

4. No uses cualquier otro tipo de información sobre ti que sea fácil de obtener, por ejemplo tu número de cuenta, tu número telefónico, tu rfc, el nombre de tu calle, etc.

5. No uses un password de dígitos o letras únicamente, ni de la misma letra.

6. No uses una palabra que se encuentre en el diccionario (ni de español, ni inglés, sobre todo).

7. No uses un password menor a seis caracteres

8. Emplea sin password que te sea fácil memorizar sin que tengas que escribirlo en alguna parte.

9. Emplea un password que puedas teclear rápidamente, para que alguien que esté mirando no pueda memorizarlo.

Existen algunos métodos que te pueden ayudar a elegir un password adecuado:

Elegir una línea o dos de una canción o poesía, y usar la primera letra de cada palabra, por ejemplo "Why, she had to go I don't know" que sería "Wshtgldk".

Alternar entre consonantes con números y signos especiales de puntuación.

Escoger dos palabras y concatenarlas con signos de puntuación. Por ejemplo: "perro;liuvia", "kid?goat".

Recuerda que el password debes cambiarlo en las máquinas: balam, cancun, cozumel, alika y tork.

Contamos con tu colaboración.

Cualquier duda u opinión respecto al contenido de este documento o al servicio en general acude con cualquier asesor o con el Administrador del Sistema (cuibículos 21, 23 y 29 del edificio principal).

El jefe de la Unidad de Servicios de Cómputo Académico de la Facultad de Ingeniería

Apéndice E

Muestra de un reporte generado por COPS en el host CANCUN

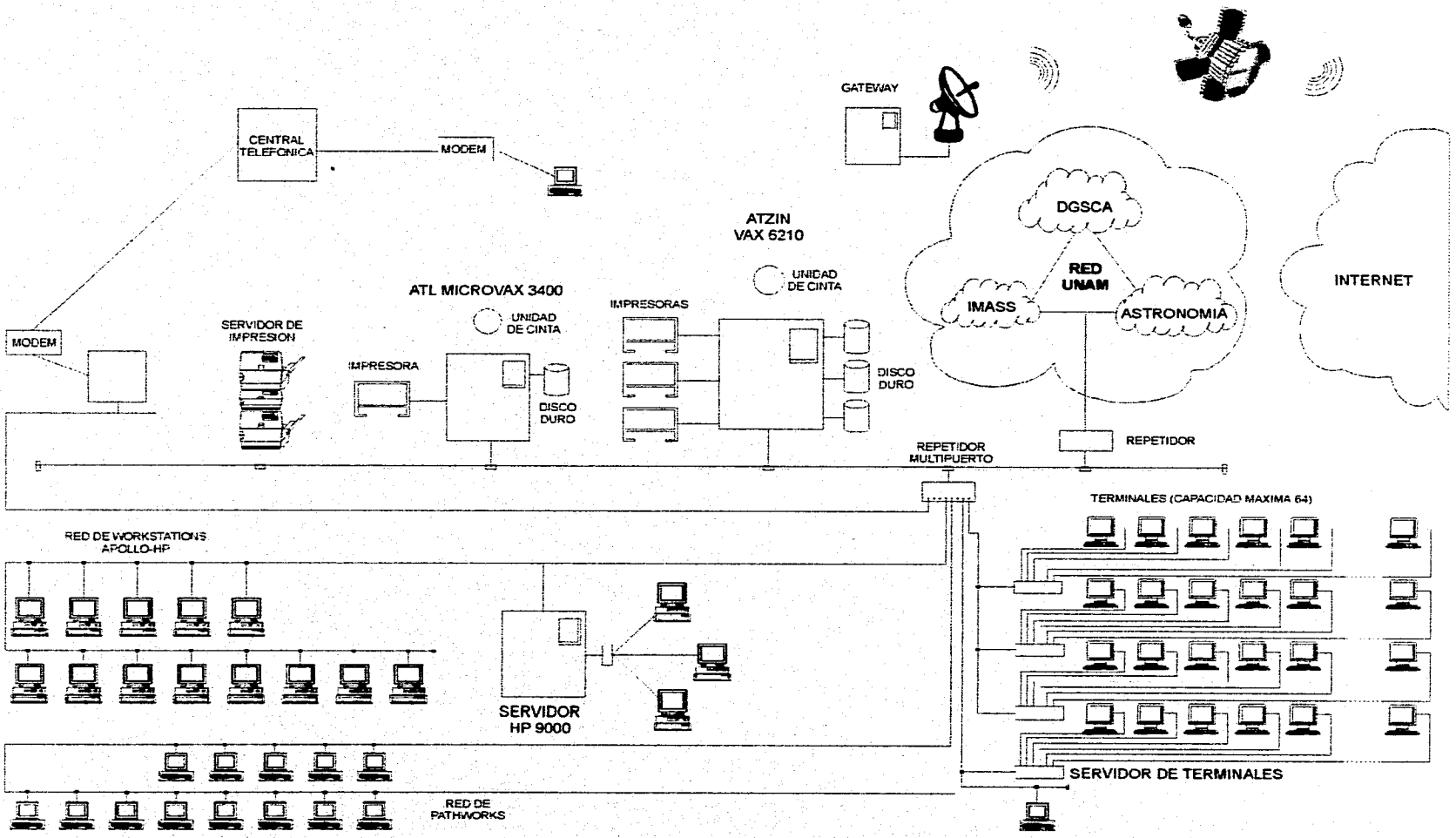
ATTENTION:

Security Report for Wed Sep 28 20:23:41 CDT 1994
from host cancun

Warning! /etc/#clusterconf is _World_ writable!
Warning! /etc/btmp is _World_ writable!
Warning! /etc/gated.conf is _World_ writable!
Warning! /etc/gated.pid is _World_ writable!
Warning! /etc/lanscan_data is _World_ writable!
Warning! /etc/named.pid is _World_ writable!
Warning! /etc/netfmt,old is _World_ writable!
Warning! /etc/nettl,old is _World_ writable!
Warning! /etc/rcflag is _World_ writable!
Warning! /etc/snmpd.pid is _World_ writable!
Warning! /etc/syslog.pid is _World_ writable!
Warning! /usr/adm/OLDrld.log is _World_ writable!
Warning! /usr/adm/OLDsyslog is _World_ writable!
Warning! /usr/adm/nettl.LOG00 is _World_ writable!
Warning! /usr/adm/rbootd.log is _World_ writable!
Warning! /usr/adm/rld.log is _World_ writable!
Warning! /usr/adm/syslog is _World_ writable!
Warning! /usr/lib/libnsipc.a is _World_ writable!
Warning! /etc/btmp is _World_ writable!
Warning! /etc/btmp is _World_ readable!
Warning! File /usr/adm/OLDrld.log (in /etc/netlinkrc) is _World_ writable!
Warning! File /usr/adm/rld.log (in /etc/netlinkrc) is _World_ writable!
Warning! File /etc/rcflag (in /etc/rc) is _World_ writable!
Warning! File /etc/syslog.pid (in /etc/rc) is _World_ writable!
Warning! File /usr/adm/OLDsyslog (in /etc/rc) is _World_ writable!

Warning! File /usr/adm/syslog (in /etc/rc) is _World_ writable!
Warning! User uucp's home directory /usr/spool/uucppublic is mode 0766!
Warning! /usr/bin/uudecode creates setuid files!
Warning! /etc/ftpusers should exist!

RED CECAFI


GATEWAY
CENTRAL TELEFONICA
MODEM
ATZIN
VAX 6210
UNIDAD DE CINTA
ATL MICROVAX 3400
UNIDAD DE CINTA
IMPRESORAS
SERVIDOR DE IMPRESION
MODEM
IMPRESORA
DISCO DURO
DISCO DURO
DGSCA
RED UNAM
IMASS
ASTRONOMIA
INTERNET
REPETIDOR MULTIPUERTO
REPETIDOR
TERMINALES (CAPACIDAD MAXIMA 64)
RED DE WORKSTATIONS APOLLO-HP
SERVIDOR HP 9000
SERVIDOR DE TERMINALES
RED DE PATHWORKS

Bibliografía

Amoroso, E.
Fundamentals of Computer Security Technology.
Prentice Hall, Englewood Cliffs, NJ, 1994.

Anklesaria, Farhad et al.
The Internet gopher protocol (A distributed document search and retrieval protocol) RFC 1436.
Marzo 1993

Arrieta Marcos, Norberto.
Introducción a la RedCECAFI.
UNAM Facultad de Ingeniería, 1992

Baker, Richard H.
Computer Security Handbook.
TAB Professional and Reference Book, McGrawHill, Blue Ridge Summit, P.A 2nd edition, 1991.

Bellovin, Steven.M.
Security Problems in the TCP/IP Protocol Suite
Computer Communication Review, Vol 19 No. 2, Abril 1989 (Tomado con ftp de ftp.research.att.com en /dist/internet_security/ipext.ps.Z)

Bishop, Matt.
A Security analysis of the NTP protocol.
In Sixth Annual Computer Security Conference Proceedings. Pags. 20-29. Tuscon AZ, Dec. 1990.

Borman, David, editor.
Telnet authentication option. RFC 1416.
Feb. 1993

Braden, Robert, editor.
Requirements for Internet hosts -application and support. RFC 1123.
Octubre 1989

Bruce Schneier.
Applied Criptography: Protocols, Algorithms an Source Code in C.

Ed. Wiley, USA, 1993.

Bryan, John.
Build a Firewall.
Byte April 1995.

Bryan, John.
Firewalls for Sale.
A look at five diferent firewalls products and services you can install today. Byte, April 1995.

Comer, Douglas E. and Stevens, David L
Internetworking with TCP/IP Volume III
Prentice Hall, New Jersey, 1993.

Cooper, James Arlin
Computer & communications Security
Strategies for the 1990's
McGrawHill Communications Series, New York, 1989

Costales, Bryan et al.
sendmail
O'Reilly and Associates, Sebastopol, CA, 1993.

Cheswick, William R. and Bellovin, Steven M.
Firewalls and Internet Security.
Repelling de Willy Hacker.
Addison-Wesley Professional Computing Series. USA, 1994

Dalva, David Y.
Security and the World Wide Web.
Home page of TIS: http://www.tis.com/Home/Network Security/WWW/Article. June 1994

Departament of Defense (USA)
Trusted Computer System Evaluation Criteria
DOD 5200.28-STD 1985

Diffie, Whitfield and Hellman, Martin E.
Exhaustive cryptanalysis of the NBS data encryption standard.
Computer June 1977.

FAQ Computer Security Frequently Asked Questions
Article 1238 of comp.security.misc.
Maintained by Alec Muffet (aem@aber.ac.uk)

Farmer, Dan and Spafford, Eugene H.
The COPS security checker system. In USENIX Conference Proceedings.

Pags. 165-170. Anaheim, CA, Verano 1990

Garfinkel, Simson and Spafford, Gene.
Practical UNIX Security
Ed. O'Reilly & Associates, Inc. USA, 1994.

Grammp, F.T. and R. H. Morris.
UNIX Operating System Security.
AT&T Bell Laboratories Technical Journal. 1649-1672, Oct. 1984

Guy, L. Steele.
The Hacker Dictionary.

Harrenstien Ken and White,Vic.
NICNAME/WHOIS. RFC 812
Marzo 1, 1982

Harrenstien, Ken.
NAME/FINGER protocol. RFC 742.
Dec 30, 1977

Hedrick, C.
Routing Information Protocol RFC 1058, 1988.

Housley, Russell.
Security Label framework for the Internet. RFC 1457.
Mayo 1993

Howard, John H.
An overview of the Andrew File System.
In USENIX Conference Preceedings
Dallas TX. 1988.

Kantor, Brian and Lapsley, Phil.
Network News transfer Protocol. RFC 977.
Feb. 1986

Kay, Russell.
Distributed and Secure
BYTE June 1994.

Kazar, Michael Leon.
Syncronization and caching issues in the andrew file system.
In USENIX conference Proceedings
Dallas TX, 1988

Linn J. *Privacy*
Enhancement for Internet Electronic Mail: Part 1. Message Encipherment

and Authenticacion Procedures. RFC 1040. 1988

Lynch, Daniel C. and Rose, Marshall T
Internet System Handbook
Addison Wesley Publishing Company, INC. , 1993

Madron, Thomas W.
Network Security in the 90's.
Issues and Solutions for Managers.
Ed. Wiley Professional Computing. U.S.A. 1992. p.3-24.

Malkin, Gary.
RIP version 2 -carrying additional information. RFC 1388.
January 1993.

Markoff, John.
Computer invasion: 'back door' ajar.
New York Times, volume CXXXVIII, p. B10, 1989

Mills, David.
Network Time Protocol (version 3) specification, implementation and analysis. RFC 1305.
Marzo 1992.

Morris, Robert and Ken Thompson.
Password Security: A Case History.
Communications of the ACM, 594-597. Nov. 1979

Morris, Robert.
A Weakness in the 4.2BSD UNIX TCP/IP Software.
Computing Science Technical Report No. 117, AT&T Bell Laboratories, Murray Hill, New Jersey, 1985

Moy, John.
OSPF version 2. RFC 1247.
Jul 1991.

N. Derek Arnold.
UNIX Security a practical Tutorial
McGrawHill 1993

National Security Agency.
Information Systems Security Products and Services Catalogue,
July 1988

Needham, R. M. y Schroeder, M.D.
Using Encriptyon for Authentication in Large Networks of Computers,
Communications of the ACM, vol. 21, no. 12, pp. 993-999, December

1978

PC Journal
TCP/IP está derribando fronteras entre usuarios
México, 1992

Postel, Jon
User datagram Protocol. RFC 768, 28 agosto 1980

Postel, Jon
Internet Control Message Protocol. RFC 792, 1981

Postel, Jon and Reynolds,Joyce.
File Transfer Protocol. RFC 959.
Oct. 1985

Postel, Jon.
Simple mail transfer protocol. RFC 821.
Agosto 1982

Postel, Jon.
Internet Protocol. RFC 791,
Septiembre 1981.

Postel, Jon.
Transmision Protocol. RFC 793.
Sept. 1981

Reeds, J.A. and P.J Weinberger.
File Security and the UNIX System Crypt Command.
AT&T Bell Laboratories Technical Journal, 1673-1683 pp. Oct. 1984

Richard Baker
Computer Security Handbook.
Mc Graw Hill, Second Edition 1991

Rosenberry, Ward et al.
Understanding DCE.
O'Reilly Associates, Sebastopol, CA, 1992.

Safford, David R. et al.
Secure RPC authentication (SRA) for TELNET and FTP.
In Proceedings of the Fourth Usenix UNIX Security Simposium.
Santa Clara CA. Oct. 1993

Sheifler Robert and Gettys James.
X Window System.

Digital press, Burlington, Massachusets, 3a. ed.,1992

St. Johns
M. Authentication Server. RFC 931, 1985

Stallings, William.
Network and Internetwork Security.
Principles and practice.
Prentice Hall, U.S.A.

Stephen, Kent.
Security Options for the Internet Protocol. RFC 1108.
Nov. 1991

Sun Microsystems
NFS: Network file system protocol specification. RFC 1094.
Marzo 1989.

Sun Microsystems.
Network Interfaces Programmer's Guide.
Mountain View, CA, Marzo 1990. SunOS 4.1

Sun Microsystems.
SunOS Reference Manual.
Part Number 800-1751-10, May 1988. p. 95

Tanenbaum, Andrew S.
Redes de Ordenadores.
Prentice Hall 2a de. 1991, México.

The Free Dictionary of Computing
en el URL: http://wombat.doc.ic.ac.uk/