UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

SISTEMA DE MONITOREO CENTRALIZADO PARA SISTEMAS REMOTOS UNIX A TRAVES DE SERVICIO HTTP, QUE CUENTE CON SEÑALES PARA LA IMPLEMENTACIÓN EN SISTEMAS DE ALTA DISPONIBILIDAD EN EMPRESAS DE COBERTURA LAN Y WAN.

DISEÑO DE UN SISTEMA DE MONITOREO PARA EQUIPO UNIX EN EMPRESAS CORPORATIVAS PARA OBTENER EL TITULO DE:

INGENIERO EN COMPUTACIÓN.

PRESENTA:

LUIS GUILLERMO VIDAL GAONA.

ASESOR : ING. FERNANDO SOLORZANO PALOMARES

MÉXICO, D.F. 2003

AGRADECIMIENTOS.

Mama, gracias por darme todo para poder seguir adelante, este logro no solo es mío sino tuyo tambien ya que siempre estuviste conmigo apoyandome, eres y siempre serás mi mayor orgullo. Nunca terminaré de agradecertelo. Todo lo que soy es por ti.

A mis hermanos que siempre estuvieron ayudandome en el dia a dia.

Yadira te agradezco por apoyarme para finalizar esta etapa profesional, gracias bonita.

Tio David, le agradezco su apoyo que siempre mostró por que terminará esta etapa profesional. Espero no haberle fallado.

INDICE

CAPÍTULO III

SEGURIDAD DE SISTEMAS DE MONITOREO.

CAPÍTULO IV

DISEÑO DE UN SISTEMA DE MONITOREO CENTRALIZADO.

CAPÍTULO V

IMPLEMENTACIÓN DEL SISTEMA DE MONITOREO CENTRALIZADO.

CAPÍTULO VI

RESULTADOS Y CONCLUSIONES.

Introducción.

En el comienzo, con las primeras maquinas, era algo muy complicado ser programador y no solo porque los lenguajes de programación no habían evolucionado, sino porque se debía manejar la computadora desde la consola y la consola en aquellos tiempos significaba un puñado de interruptores. Afortunadamente, esto ha ido cambiando y se lo debemos, en parte, a que han nacido y evolucionado los sistemas operativos. Como también lo han hecho las maquinas, los lenguajes de programación e incluso las ideas desde la mas básica hasta ideas fascinantes como teorías sobre viajar a la velocidad de la luz o mas allá sobre las partículas llamadas taquiones mas rápidas que la velocidad de la luz[1].

¿ Cual es la necesidad de un Sistema Operativo ?

En el principio solo existía el hardware de las comienzos de lo que conocemos hoy como computadora. Las primeras computadoras eran físicamente grandes maquinas que se operaban desde una consola. El programador escribía un programa y luego lo controlaba directamente. En primer lugar el programa era cargado manualmente en la memoria, desde los interruptores del tablero frontal, desde una cinta de papel o desde tarjetas perforadas. Luego eran activados botones adecuados para establecer la dirección de inicio y comenzar la ejecución en verdad era una tarea muy laboriosa que en la actualidad el gran porcentaje que tiene una computadora en casa ni siquiera saben de este comienzo. Conforme transcurría el tiempo, se desarrollaron software y hardware adicionales para minimizar espacios, costos y tiempos. Sin embargo desde sus comienzo surgieron problemas que hasta la fecha no ha sido solucionados del todo, aunque un CPU de la mayoría de las computadoras actuales son mayores de 2GHz de velocidad se mantiene problemas de intervención sobre los demás componentes (discos, tarjetas de red, memoria, etc) que aparentemente los usuarios solo se percatan cuando algún dispositivo envía mensajes en código hexadecimal sin saber que pasa, la mayoría de estos usuarios simplemente utilizan la operación del termino universalmente mal empleado " resetea la maquina" apagar la computadora.

[1] La palabra taquio viene del griego tachys que significa <rápido, veloz>. Referencia La relatividad, espacio, tiempo y movimiento. Autores Delo E. Mook y Thomas Vargish, pag. 263.

1.0 Evolución de los sistemas operativos.

GENERACIÓNES DE SISTEMAS OPERATIVOS.

Los sistemas operativos, al igual que el hardware, han sufrido una serie de cambios revolucionarios llamados generaciones. En el caso particular del hardware han sido enmarcadas por grandes avances en los componentes utilizados pasando de válvulas (primera generación), a transistores(segunda generación), a circuitos integrados (tercera generación), a circuitos integrados de gran y muy gran escala (cuarta generación). En cada fase sucesiva de hardware han sido acompañadas de reducción substanciales de costos, tamaño, emisión de calor y consumo de energía, y por incrementos notables en velocidad y capacidad, adicional a esto se han incrementado las nuevas líneas de conocimiento sobre diferentes áreas en la actualidad una muy nombrada es la bioingeniería.

Generación Cero (década de 1940)

Los sistemas operativos han ido evolucionando durante los últimos 40 años a través de un numero de distintas fases o generaciones que corresponden a décadas. En 1940, las computadoras eléctricas digitales mas nuevas no tenían sistemas operativos. Las maquinas de ese tiempo eran tan primitivas que los programadores por lo regular manejaban un bit a la vez en comunas de switch's mecánicos. Eventualmente los programas de lenguaje maquina manejaban tarjetas perforadas y leguajes ensambladores fueron desarrollados para agilizar el proceso de programación. Los usuarios tenían completo acceso al lenguaje de la maquina. Todas la instrucciones eran codificadas a mano.

Primera generación (década de 1950)

Los laboratorios de investigación de General Motors implementaron el primer sistema operativo de los 50's generalmente corría una tarea a la vez y suavizo la transición entre tareas para obtener máxima utilización del sistema de la computadora. Los sistemas operativos de los años cincuenta fueron diseñados para hacer más fluida la transmisión entre trabajos. Antes de que los sistemas fueran diseñados, se perdía un tiempo

considerable entre la terminación de un trabajo y el inicio del siguiente. Este fue el comienzo de los sistemas de procesamiento por lotes, donde los trabajos se reunían por grupos o lotes. Cuando el trabajo estaba en ejecución, dicho proceso mantenía control total del equipo. Al finalizar el trabajo ya sea en ejecución exitosa o no, el control era transmitido al siguiente programa en fila.

Segunda generación (primera mitad de la década de 1960)

Una de las principales características de la segunda generación de los sistemas operativos fue el desarrollo de los sistemas compartidos con multiprogramación, y los principios del multiprocesamiento. En estos sistemas de multiprogramación, varios programas de usuarios se encuentran al mismo tiempo en el almacenamiento principal, y el procesador se cambia rápidamente de un trabajo a otro. Por estos tiempos surgió la ingeniería de software que tuvo muchos problemas debido a la poca capacidad de relación entre el software y hardware ya que para que un programador fuera exitoso tendría que saber no solo la logística del programación sino que tendría que adentrase al hardware en toda su definición.

Tercera generación (mitad de la década de 1960 a la mitad de la década de 1970)

El la tercera generación , el sistema operativo comenzó en forma efectiva, en 1964, con la introducción de la familia de computadoras sistemas/360 de IBM. Las variadas computadoras 360 fueron diseñadas para ser compatibles con el hardware, para usar el sistema operativo OS/360, y para ofrecer mayor poder computacional, en tanto se iba avanzando con la demanda la serie 360 fue creciendo, los computadores de la tercera generación aun mantenían el voluminoso espacio aunque eran mas funcionales para diferentes tareas y usuarios a la vez, algunos ya soportaban simultáneamente procesos por lotes, tiempos compartidos, procesamiento de tiempo real y multiprocesamiento.

Cuarta generación (de la mitad de la década de 1970 a nuestros días)

La cuarta generación la vivimos hoy en día, aun así muchos usuarios no tiene ni idea de las anteriores generación, prácticamente los sistemas operativos de hoy en día son multiusuarios, multiprocesamiento así como el hardware aunque hay inconvenientes son mínimos para las necesidad comunes de un usuarios.

CRONOLOGIA DE LA CREACIÓN DE LOS SISTEMAS OPERATIVOS.

S.O	Año	Autor	Gestion de procesos	Arquitectura	Multiusuar
Atlas	50-60	University of Manchester	Lotes	Monolitico	No
The	66	Universidad de Eindhoven	Lotes	Modular	No
RC4000	67	Brich Hansen de Regenecentralen	S.O. Completo	Modular	No
Solo	69	Brich Hansen de Regenecentralen	Multiprogramacion	Modular	No
CTSS		MIT	Multiprogramacion	Monolitico	Si
Multics		MIT	Multiprogramacion	Monolitico	Si
Unix	1969	Ritchie/Thompson	Multiprogramacion	Monolitico	Si
MINIX EDUCATIVO		Andrew Tanenbaum	Multiprocesos	Modular	Si
QDOS MS-DOS	1981	Microsoft	Lotes	Monolitico	No
MINIX LINUX	1991	Linus Torvalds	Multiprogramacion	Monolitico	Si
Sprite	1984		Multiprogramacion	Modular	Si
Merlin	1984		Lotes	Monolitico	No
Windows NT	1985	Microsoft	Multiprogramacion	Modular	Si
OS/2	1987	IBM/Microsoft	Multiprgramacion	Monolitico	No
Mach	1986	Darpa	Multiprogramacion	Monolitico	Si
Amoeba	1994		Distribuido	Microkernel	Si
Windows 95/98	1995 1998	Microsoft	Multiprogramacion	Monolitico	No
Coyote	1996	Trinity Collage Dublín	Distribuido	Modular	Si

1.1 Formas de monitoreo y auditoria de Sistemas Operativos.

Antes de profundizar sobre los temas de monitoreo y audítoria es bueno señalar que en la actualidad se ha incrementado el interés por estos puntos debido a su intrínseca importancia toda compañía que se vea a futuro en función de optimizar sus recursos y su propia sobre vivencia en el mercado se tomara el tiempo para realizar la plantación sobre estos dos rubros. El punto de monitoreo se orientara a las funcionalidades del hardware que se tenga en una empresa en este capítulo se darán las bases sobre los puntos administrativos que se deben cumplir para realizar un monitoreo y audítoria aplicando normas COBIT (Control Objectives for Information and related Technology), las cuales son el resultado de uno de los mayores proyectos de investigación completado y publicado por la Organización Mundial de Auditores de Sistemas de Información (ISACF).

ELABORANDO UNA POLITICA DE AUDÍTORIA Y SEGURIDAD INFORMÁTICA.

La política de audítoria y seguridad informática debe formar parte de los lineamientos generales a desarrollarse con base en las directivas emanadas de la gerencia general y formará párrate del compromiso de esta en su aplicación. En ellas se deberá establecer con claridad y precisión las metas a alcanzar y las responsabilidades asignadas. En ella se deberá establecer con claridad y precisión las metas a alcanzar y las responsabilidades asignadas. Podríamos sintetizar brevemente cuatro ejes por los cuales debería establecer esta política:

- Programa General de audítoria y seguridad de la organización.
- Programa de audítorias informáticas a implementar.
- Programa sobre temas específicos como contingencias, seguridad física, etc.
- Programas específicos sobre sistemas informáticos de información determinada.

Estos ejes de desarrollo deberán propender a la utilización de la información con una optima relación costo-beneficio permitiendo compartir la información y ayudar a aprovechar mejor los recursos destinados a la Tecnología Informática (TI) en todo su potencial. Algunos de los elementos a tener en cuenta al momento de implementar una política de auditoria informática serán , por ejemplo:

Establecimiento de una función que lleve a cabo la administración del programa de auditoria y seguridad informática que sea reconocida por toda la organización.

Estándares, manuales y guías que respalden las medidas tomadas.

Por ultimo se deberá analizar con detalle las pautas a considerar para la evaluación del nivel de cobertura existente ante situaciones de desastres, la identificación en forma anticipada de los factores de riesgo y planificación de las acciones a seguir.

CONSIDERACIONES PARA LA ELABORACION DE PROGRAMAS DE SEGURIDAD.

Para el desarrollo de los diversos programas de seguridad es necesario basarse en una adecuada administración de riesgos. La administración de los riesgos deberá entenderse como su identificación, evaluación y la posterior adopción. Los DRP (Disaster Recovery Planning) son tratados en la actualidad con mas seriedad, aunque debido a sus costos todavía aun elevados las empresas optan por empezar sus lineamientos así esos rumbos, claro que esto dependerá del nivel del corporativo, estructura económica, futuros crecimientos, demanda de servicios y por su puesto el nivel de prioridad de información que se maneje, por ejemplo, los riesgos en aplicaciones criticas como las bancarias evidentemente tienen una evaluación distinta que una aplicación corriente, la administración de riesgos y su implementación comprende el desarrollo de una serie de actividades que serán objeto de tareas especificas de tecnología de seguridad informática. Asimismo, es de gran importancia en la elaboración de de programas de monitoreo y seguridad, tener especial consideración en el imprescindible equilibrio entre los costos y los beneficios a obtener con la puesta en marcha de los mismos.

Aplicación de normas COBIT, este conjunto de normas constituye un estandar internacional para la aplicación de un correcto control de los sistemas de información. Es aplicable a un amplio rango de sistemas de información que van desde el nivel de computadoras personales de rango medio, grandes computadoras (Mainframe) e instalaciones Cliente-Servidor. Este trabajo, esta basado en una rigurosa crítica a las tareas y actividades de todo tipo de ámbito informático. Para su confección se han tenido en cuenta las principales normas y estándares de las organizaciones que supervisan a auditores americanos y europeos, requerimientos y especificaciones de la industria bancaria y del gobierno. En general, se puede sintetizar los objetivos de las normas COBIT en los siguientes puntos:

Es una guía muy importante para la gerencia en la toma de decisiones sobre riesgos y controles.
Ayuda al usuario de tecnología a obtener seguridad y control sobre los productos y servicios que adquiere.

Provee a la audítoria de sistemas informáticos, una herramienta fundamental para evaluar controles internos, controles gerenciales, y los mínimos requerimientos de control compatibles con el necesario balance costo-beneficio de la organización.

La aplicación de estos estándares a cualquier instalación o emprendimiento informático garantiza una correcta y segura utilización de la información. En la medida que le uso de los sistemas de información se expande y más personas dependen de su continuidad operativa tanto mas importante es contar con un adecuado plan de contingencia y recuperación que facilite superar situación no deseadas. Esto que parece algo elemental o trivial, no lo es tanto a la hora de su implementación practica. Estamos muy habituados a encontrarnos en situaciones donde grandes organizaciones encargadas de procesar volúmenes importantes de datos, recurren a diferentes formas de respaldos, no siempre hechos con criterios correctos para poder actualizar o recuperar datos nuevos o perdidos,

enseguida se dan algunos puntos que se deben evaluar cuando se implementa esta fase según normas:

- Identificación en forma preliminar los factores de riegos ante situaciones de desastres.
- Evaluar con estándares las políticas de respaldo de información.
- Planificación de acciones a seguir
- Designación de responsables de implementación del plan
- Asegurar el correcto funcionamiento del plan.

SEGURIDAD EN INTERNET

Sin duda este punto es medular en seguridad, la expansión que ha tenido en estos últimos años la "Red de Redes" o lo que comúnmente conocemos como "Autopista de la Información", permitiendo el acceso de la información de todo tipo a todo el mundo a un costo considerablemente bajo. Además esta tecnología realmente ha convertido el modo de comunicarse, relacionarse comercialmente, académicamente y profesionalmente, de una manera como nunca antes existió, este crecimiento exponencial, ha hecho que , tecnológicamente , aun no se hayan desarrollado los elementos de seguridad suficientes para garantizar una absoluta privacidad e integridad de los datos que viajan por la red.

No obstante, los puntos principales se nombran a continuación, sin embargo en el capítulo III de **"SEGURIDAD EN SISTEMAS DE MONITOREO"** se profundizará en este tema.

Puntos a considerar en audítoria, basado en las normas para protección de datos:

Autentificación e Identificación.- Técnica que nos permiten individualizar al autor de determinada acción.

Autorización.- Técnica que permite determinar a que información tienen acceso determinadas personas.

Integridad de datos.- Técnica que garantiza que los datos que viajan por la red lleguen intactos a su destino.

Privacidad de datos.- Técnica que determina quien puede leer la información una vez que salio del sistema de almacenamiento.

ADMINISTRACION DEL PERSONAL Y DE LOS USUARIOS.

Los elementos a tener en cuenta en seguridad referentes a la administración del personal encargado de operar los sistemas de información y de los usuarios en general están relacionados con las interacciones de las personas, los equipos informáticos y las autoridades que cada uno necesita para llevar a cabo su trabajo. Para ello podemos mencionar cuatro puntos que se consideran los fundamentales para esta tarea:

- Organización de personal.
- Administración de usuarios.
- Permisos de accesos del personal contratado.
- Accesos públicos.

AUDÍTORIA EN REDES DE PROCESAMIENTO.

En el caso de las redes de procesamiento, se tiene que analizar sin duda a la par sobre los requisitos de usuarios y aplicación que se este utilizando, dependiendo de los dos anteriores factores se realizará principalmente la planeación sobre un crecimiento futuro, aunado a esto se debe considerar los enlaces locales y remotos que se utilizan para saber el nivel de seguridad con que se cuenta. Por otra parte se tiene que llevar una estadística de funcionalidad en ancho de banda, protocolo que se utilicen, colisiones y sin duda el nivel de capacitación de los administradores. En el punto siguiente "Topologias de redes LAN, WAN y MAN" se tocará el tema de topologías sobre las cuales hay ciertas particularidades para el funcionamiento optimo así como los esquemas que se pueden conformar

dependiendo de la infraestructura empresarial que se maneja, dependiendo de los requisitos antes mencionados.

1.2 Topologías de redes LAN, MAN y WAN.

Hay muchas maneras de organizar los componentes de telecomunicaciones para formar una red, y por lo tanto, hay múltiples clasificaciones de redes. Una manera de describir las redes es por su forma o topología, es decir, el diseño físico de nodos en una red, en la clasificación se incluye el área que cubre la infraestructura a nivel de comunicaciones en una empresa, enseguida se darán las bases en las cuales se partirá para realizar el sistema de monitoreo que se plantea para la tesis.

En el comienzo de clasificación de cobertura de redes se postularon tres principales clasificación que se nombran a continuación sin embargo en la actualidad ya los significados llegan a mezclase, debido a las convenciones que se manejan y en cierto modo han dejado obsoleto la clasificación original, sin embargo se tienen los siguientes rubros:

Un criterio para clasificar redes de computadoras es el tomar en cuenta o con base en su extensión geográfica, es en este sentido cuando se puede hablar de las siguientes clasificaciones:

LAN (Redes de Area Local).- Son redes de propiedad privada , de hasta unos cuantos kilómetros de extensión. Por ejemplo una oficina o un centro educativo. Están restringidas en tamaño, lo cual significa que el tiempo de transmisión, en el peor de los casos, se conoce, lo que permite cierto tipo de diseños (deterministas) que de otro modo podrían resultar ineficientes. Además, simplifica la administración de la red. Suelen emplear tecnología de difusión mediante un cable sencillo, ha velocidades entre los rangos de 10 y 100 Mbps.

WAN (Redes de Area Metropolitana).- Son una versión mayor de la LAN y utilizan una tecnología muy similar. Actualmente esta clasificación ha caído en desuso, normalmente solo distinguiremos entre redes LAN y MAN.

MAN (Redes de Area Amplia).- Son redes que se extienden sobre un área geográfica extensa. Estas LAN acceden a diferentes subredes que la componen, adicional ha esto se tiene implementación de equipos de comunicación que trabajan en capas diferentes del modelo OSI[2].

Las anteriores estructuras se forman de topologías internas de redes según sea sus necesidades en función de; momento de creación (etapa tecnológica), decisión gerencial, nivel económico de la empresa, etc.

La palabra topología es un término de origen griego que se refiere al estudio de las formas y que se emplea en el diseño de redes de comunicación, para referirse precisamente a la forma en que están conectados los nodos de una red. Precisamente una red de comunicaciones esta formada básicamente por tres elementos, Nodo, Enlaces y Equipos terminales. Por lo tanto, definiremos como topología de una red a la forma como los equipo terminales se conectan entre sí y con los nodos, a través de los enlaces de comunicaciones.

Normalmente, las redes remotas como locales, se apoyan en la topología siguiente: estrella, malla, anillo, bus y árbol. A estas se les denomina topologías básicas pero existen por combinación de ellas las topologías mixtas o combinadas.

Las redes en general, consisten en "compartir recursos", y uno de sus objetivos es hacer que todos los programas, datos y equipos estén disponibles para cualquiera de la red que así lo solicite, sin importar la localización física del recurso y del usuario. Un segundo objetivo consiste en proporcionar una alta fiabilidad al contar con fuentes alternativas de suministro. Otro objetivo es el ahorro económico. Un punto muy relacionado es la capacidad para aumentar el rendimiento del sistema en forma gradual a medida que crece la carga, simplemente añadiendo mas procesadores. Además la red puede proporcionarnos un poderoso medio de comunicación entre personas que se encuentran muy alejadas entre sí.

[2] El modelo OSI se compone de 7 capas (Aplicación, Presentación, Sesión, Transporte, Red, Enlace y Fisica). .

Las redes tienden a crecer e innovarse, al principio se conectan unas cuantas personas y luego todo el mundo desea conectarse, hasta verse en la necesidad de conectarse a un correo electrónico. Con las redes actuales se pueden disponer de prestaciones hasta ahora inimaginables como compartir una impresora, un escáner, toda clase de datos e incluso un módem de forma tal que varios usuarios se conecten a Internet realizando una sola llamada de teléfono.

Hay una topología adecuada para cada fin, y elegir cual será la que usaremos no es tarea fácil. Al escoger la topología que implantaremos hay que tener en cuenta las ventajas y desventajas que tiene cada una de ellas.

Nuestro trabajo intentara presentarles estas ventajas y la seguridad que tiene cada una.

COMPONENTES DE UNA RED

Una red de computadoras está conectada tanto por hardware como por software. El hardware incluye tanto las tarjetas de interfaz de red como los cables que las unen, y el software incluye los controladores (programas que se utilizan para administrar los dispositivos y el sistema operativo, el software de red que administra la red).

Servidor : éste ejecuta el sistema operativo de red y ofrece los servicios de red a las estaciones de trabajo.

Estaciones de Trabajo: Cuando una computadora se conecta a una red, la primera se convierte en un nodo de la última y se puede tratar como una estación de trabajo o cliente. Las estaciones de trabajos pueden ser computadoras personales con el DOS, Macintosh, Unix, OS/2 o estaciones de trabajos sin discos.

Tarjetas o Placas de Interfaz de Red: Toda computadora que se conecta a una red necesita de una tarjeta de interfaz de red que soporte un esquema de red específico, como Ethernet, ArcNet o Token Ring. El cable de red se conectará a la parte trasera de la tarjeta.

Sistema de Cableado: El sistema de la red está constituido por el cable utilizado para conectar entre sí el servidor y las estaciones de trabajo.
Recursos y Periféricos Compartidos: Entre los recursos compartidos se incluyen los dispositivos de almacenamiento ligados al servidor, las unidades de discos ópticos, las impresoras, los trazadores y el resto de equipos que puedan ser utilizados por cualquiera en la red.

DISEÑO EN TOPOLOGÍAS DE REDES

Podemos considerar tres aspectos diferentes a la hora de considerar una topología:

1. La topología física, que es la disposición real de los host y de los cables (los medios) en la red.

2. La topología lógica de una red es la forma en que los hosts se comunican a través del medio. Los dos tipos más comunes de topologías lógicas son broadcast (Ethernet) y transmisión de tokens (Token Ring).

La topología de broadcast simplemente significa que cada host envía sus datos hacia todos los demás hosts del medio de red. Las estaciones no siguen ningún orden para utilizar la red, el orden es el primero que entra, el primero que se sirve. Esta es la forma en que funciona Ethernet.

La transmisión de tokens controla el acceso a la red al transmitir un token eléctrico de forma secuencial a cada host. Cuando un host recibe el token, eso significa que el host puede enviar datos a través de la red. Si el host no tiene ningún dato para enviar, transmite el token hacia el siguiente host y el proceso se vuelve a repetir.

3. La topología matemática, donde los mapas de nodos y los enlaces a menudo forman patrones.

Vamos a ver a continuación los principales modelos de topología. El término topología en redes se refiere a la ubicación física de las computadoras, cables y otro componentes de la red. Topología es un término que muchos profesionales utilizan cuando se refieren del diseño básico de una red. Otros términos que se utilizan para definir un diseño de red son:

Ubicación física.
Diseño
Diagrama
Mapa

La elección de una topología sobre otra va a tener un fuerte impacto sobre:
El tipo de equipo que la red necesita
Las capacidades de este equipo
Desarrollo de la red
La forma en que la red es manejada

Sabiendo sobre las distintas topologías, se llega a entender más las capacidades de los distintos tipos de redes.
Para que las computadoras puedan compartir archivos y poder transmitirlos entre ellos tienen que estar conectados. La mayoría de las redes usan un cable para conectar una computadora a otra, para hacer esto posible.
Sin embargo, esto no es tan simple como conectar un cable de una computadora a otra. Diferentes tipos de cable requieren diferentes tipos de arreglos.
Para que una topología en red funcione bien, necesita un diseño previo. Por ejemplo, una topología en particular puede determinar el tipo de cable que se necesita y como ese cableado recorre el piso, las paredes y el techo.

Las diferentes topologías en redes, las cinco topologías básicas son:

BUS
ESTRELLA
ANILLO
MALLA
ARBOL

Estas topologías pueden ser combinadas en una variedad de topologías híbridas más complejas.

Topología BUS

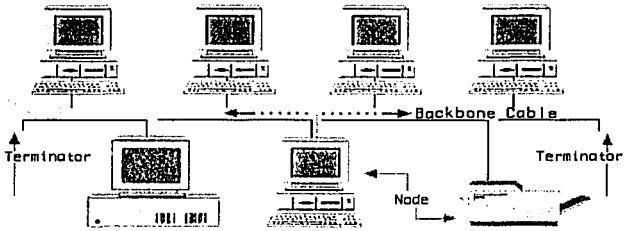
En esta topología las computadoras están conectadas por un canal de comunicación en línea recta. Esta red es la más común y la más simple. El canal de comunicación único se le suele llamar backbone.

Para entender como las computadoras se comunican en esta topología en red, hay que familiarizarse con tres conceptos:

Mandar la señal
Que la señal rebote
Que termine de rebotar la señal

Los datos de la red se mandan en forma de señales electrónicas a todas las computadoras de la red. Solo una computadora a la vez puede mandar mensajes en esta topología, por esto, el número de computadoras al bus va a afectar el rendimiento de la red. Cuantas más computadoras están conectadas, más computadoras van a estar esperando para mandar datos por el bus y como consecuencia más lenta va a ser la conexión por red. Todos los factores van a alterar el rendimiento de la red:

Tipos de cables utilizados en la red
Distancia entre computadoras en la red
Tipo de aplicaciones siendo ejecutadas en red



Las computadoras conectadas a un bus, o transmiten datos a otras computadoras en la red o esperan recibir datos de otras computadoras de la red. Ellas no son responsables de transmitir datos de una computadora a otra. Por consecuencia, si una computadora falla, no afecta al resto de la red.

Como los datos son mandados a toda la red, estos viajan de una punta del cable a la otra. Si se permite que la señal continúe ininterrumpidamente, esta va a seguir rebotando ida y vuelta por el cable y va a prevenir que las computadoras sigan transmitiendo datos. Para que esto no pase, la señal tiene que ser parada si tuvo la oportunidad de llegar al destino correcto.

Para impedir que la señal siga rebotando por todo el cable, se coloca un terminador en cada punta del cable para que absorba todas las señales. Esto permite que el cable se libere de estas señales para que otras computadoras puedan mandar datos.
Cada punta del cable tiene que ser enchufada a una terminal para impedir que la señal siga rebotando. Si esto no ocurre la actividad en red se va a interrumpir. Las computadoras en la

red van a poder seguir funcionando solas pero mientras que haya un cable desconectado no van a poder compartir datos.

El cable de una topología bus se puede expandir de dos formas distintas:

Una forma, es utilizando un componente llamado barrel conector que lo que hace es conectar dos partes del cable, creando un cable mas largo. La desventaja de este conector es que hace que la señal sea más débil. Si se utilizan demasiados conectores, se puede llegar a dar el caso que la señal no se reciba correctamente.

La otra forma es utilizar un dispositivo llamado repetidor que se utiliza para conectar dos cables. Lo que hace este dispositivo es aumentar la señal para que llegue a su destino. Recibe una señal débil y la transforma en una señal de normal transferencia. Se sobreentiende que este dispositivo es mejor que el barrel conector porque permite que una señal viaje a grandes distancias sin que se debilite

VENTAJAS

Facilidad de añadir estaciones de trabajo
Manejo de grandes anchos de banda
Muy económica
Soporta de decenas a centenas de equipos
Software de fácil manejo
Sistema de simple manejo

DESVENTAJAS

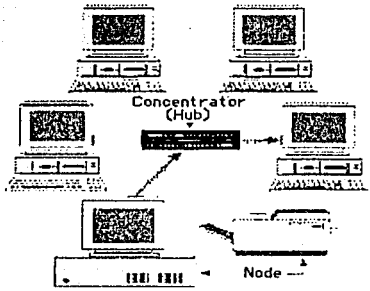
El tiempo de acceso disminuye según el número de estaciones.
Cuando el número de equipos es muy grande el tiempo de respuesta es más lento.
Dependiendo del vínculo puede presentar poca Inmunidad al ruido.
Las distorsiones afectan a toda la red.

La rotura de cable afecta a muchos usuarios.
Como hay un solo canal, si este falla, falla toda la red.
Posible solucionar redundancia.

El cable central puede convertirse en un cuello de botella en entornos con un tráfico elevado, ya que todas alas estaciones de trabajo comparten el mismo cable. Es difícil aislar los problemas de cableado en la red y determinar que estación o segmento de cable los origina, ya que todas las estaciones están en el mismo cable. Una rotura de cable hará caer el sistema.

Topología en ESTRELLA

En esta topología todos los cables de todas las computadoras son conectados a un dispositivo central llamado hub. Los datos de una computadora son transmitidos por el hub al resto de las computadoras en red. Esta topología apareció con la utilización de la computadora mainframe. La ventaja de esta topología es que todos lo procesos son centralizados y esto permite un fácil control de trafico. Sin embargo, como cada



computadora tiene que ser conectada al hub, esta topología requiere un gran cablerio para que funcione. Si el hub deja de funcionar, toda la red se para. Si una computadora se rompe el resto de la red sigue funcionando normalmente.

Este tipo de red es adecuado cuando se tiene una computadora central muy poderosa rodeada de maquinas menos potentes que sirven únicamente como terminales de entrada y salida de datos, ya que todos los extremos de la red tienen acceso a los recursos de la maquina principal de manera directa, sin interferencia de elementos intermedios.
También pude ser usada con redes Punto a Punto, de tal forma que todas las computadoras, con iguales características, están conectadas al HUB o concentrador y cualquiera de ellas puede tener acceso a las demás. Es una configuración ampliamente utilizada a nivel empresarial.
De esta manera se consiguen enormes velocidades de transferencia de datos, lo que resulta ideal para sistemas que manejen flujos muy grandes de información entre la computadora central y sus terminales. Su principal inconveniente es la necesidad de colocar un cable exclusivo para cada terminal.

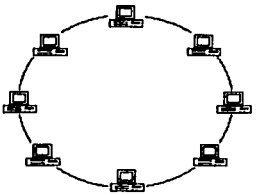
VENTAJAS

Estructura simple
Cada PC es independiente de los demás
Facilidad para detectar pc's que estén causando problema en la red
Fácil conexión a la red
Son las mejores para aplicaciones que estén ligadas a gran capacidad de procesamiento
Permite añadir nuevas computadoras a la red.
Control de tráfico centralizado.
LA falta de una computadora no afecta a la red.

DESVENTAJAS

Limitación en rendimiento y confiabilidad
Su funcionamiento depende del servidor central
Su crecimiento depende de la capacidad del servidor central
La distancia entre las estaciones de trabajo y el servidor

Topología ANILLO



Topología Anillo

Esta topología conecta a las computadoras con un solo cable en forma de circulo. Con diferencia de la topología bus, las puntas no están conectadas con un terminados. Todas las señales pasan en una dirección y pasan por todas las computadoras de la red. Las computadoras en esta topología funcionan como repeaters, porque lo que hacen es mejorar la señal. Retransmitiéndola a la próxima computadora evitando que llegue débil dicha señal. La falla de una computadora puede tener un impacto profundo sobre el funcionamiento de la red.

La principal ventaja de la red de anillo es que se trata de una arquitectura muy sólida, que pocas veces entra en conflictos con usuarios.
Muchas empresas e instituciones grandes prefieren tener sus computadoras conectadas en una arquitectura de anillo gracias a que esta forma de conexión es especialmente favorecida por los grandes proveedores de acceso a Internet. Debido precisamente a su poderío.

Doble anillo (Token ring): Un método de transmisión de datos alrededor del anillo se denomina token passing. Esta técnica consiste en que la computadora emisora transmita un dato que la computadora receptora la reciba y que esta mande una señal de respuesta informando que recibió el dato correctamente. Todo esto se hace a la velocidad de la luz. Las redes Token Ring no tienen colisiones. Si el anillo acepta el envío anticipado del token, se puede emitir un nuevo token cuando se haya completado la transmisión de la trama.

Las redes Token Ring usan un sistema de prioridad sofisticado que permite que determinadas estaciones de alta prioridad designadas por el usuario usen la red con mayor frecuencia. Las tramas Token Ring tienen dos campos que controlan la prioridad: el campo de prioridad y el campo de reserva.

Sólo las estaciones cuya prioridad es igual o superior al valor de prioridad que posee el token pueden tomar ese token. Una vez que se ha tomado el token y éste se ha convertido en una trama de información, sólo las estaciones cuyo valor de prioridad es superior al de la estación transmisora pueden reservar el token para el siguiente paso en la red. El siguiente token generado incluye la mayor prioridad de la estación que realiza la reserva. Las estaciones que elevan el nivel de prioridad de un token deben restablecer la prioridad anterior una vez que se ha completado la transmisión.

Las redes Token Ring usan varios mecanismos para detectar y compensar las fallas de la red. Uno de los mecanismos consiste en seleccionar una estación de la red Token Ring como el monitor activo. Esta estación actúa como una fuente centralizada de información de temporización para otras estaciones del anillo y ejecuta varias funciones de mantenimiento del anillo. Potencialmente cualquier estación de la red puede ser la estación de monitor activo.

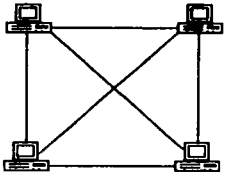
Una de las funciones de esta estación es la de eliminar del anillo las tramas que circulan continuamente. Cuando un dispositivo transmisor falla, su trama puede seguir circulando en el anillo e impedir que otras estaciones transmitan sus propias tramas; esto puede bloquear la red. El monitor activo puede detectar estas tramas, eliminarlas del anillo y generar un nuevo token.

VENTAJAS

El sistema provee un acceso equitativo para todas las computadoras
El rendimiento no decae cuando muchos usuarios utilizan la red.

DESVENTAJAS

La falla de una computadora altera el funcionamiento de toda lea red.
Las distorsiones afectan a toda la red.



Topología Malla

Topología en MALLA.

La topología en malla principalmente nos ofrece redundancia. En esta topología todas las computadoras están interconectadas entre sí por medio de un tramado de cables. Esta configuración provee redundancia porque si un cable falla hay otros que permiten mantener la comunicación. Esta topología requiere mucho cableado por lo que se la considera muy costosa. Muchas veces la topología MALLA se va a unir a otra topología para formar una topología híbrida.

Las redes en malla son aquellas en las cuales todos los nodos están conectados de forma que no existe una preeminencia de un nodo sobre otros, en cuanto a la concentración del tráfico de comunicaciones.

En muchos casos la malla es complementada por enlaces entre nodos no adyacentes, que se instalan para mejorar las características del trafico.

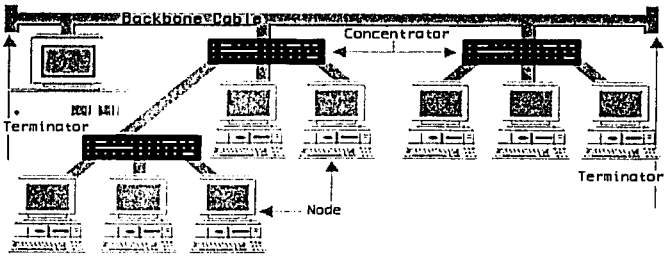
Este tipo de redes puede organizarse con equipos terminales solamente (en lugar de nodos), para aquellos casos en que se trate de redes de transmisión de datos.

Estas redes permiten en caso de una iteración entre dos nodos o equipos terminales de red, mantener el enlace usando otro camino con lo cual aumenta significativamente la disponibilidad de los enlaces.

Baja eficiencia de las conexiones o enlaces, debido a la existencia de enlaces redundantes.
Por tener redundancia de enlaces presenta la ventaja de posibilitar caminos alternativos para la transmisión de datos y en consecuencia aumenta la confiabilidad de la red.
Como cada estación esta unida a todas las demás existe independencia respecto de la anterior.
Poco económica debido a la abundancia de cableado.
Control y realización demasiado complejo pero maneja un grado de confiabilidad demasiado aceptable.

Topología en ÁRBOL

La topología de árbol combina características de la topología de estrella con la BUS. Consiste en un conjunto de subredes estrella conectadas a un BUS. Esta topología facilita el crecimiento de la red.



VENTAJAS

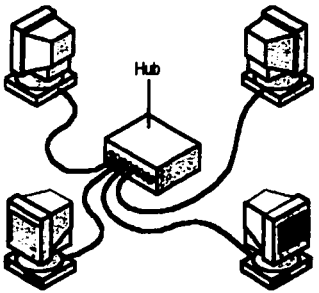
Cableado punto a punto para segmentos individuales.
Soportado por multitud de vendedores de software y de hardware.

DESVENTAJAS

La medida de cada segmento viene determinada por el tipo de cable utilizado.
Si se viene abajo el segmento principal todo el segmento se viene abajo con él.
Es más difícil su configuración.
Las redes de computadoras se montan con una serie de componentes de uso común y que es mayor o menor medida aparece siempre en cualquier instalación.

Cuadro comparativo de Topologías

	TRAFICO	VINCULO REQUERIDO	COSTO	FACILIDAD DE AÑADIR EQUIPOS	DESVENTAJA MÁS IMPORTANTE
BUS	Fácil controlar el tráfico en Distintos equipos terminales	fibra óptica porque el tráfico es muy importante	no es alto el costo en vínculos	muy fácil la nueva terminal debe "colgarse" del cable simplemente	depende de un solo vínculo toda la red
ESTRELLA	fácil de controlar su tráfico, el cual es muy sencillo	el par trenzado es aceptable ya que no hay problemas de tráfico	se usa más cantidad de cables y hubs	depende de la posibilidad del hub (cantidad de puertos)	se debe usar un cable para cada terminal
ANILLO	son raras las congestiones Causadas por el cableado	preferentemente fibra óptica	moderado	para conectar otro nodo se debe paralizar la red	la falla de una PC altera la red, asi como las distorsiones
MALLA	en caso de averías se orienta el Tráfico por caminos alternativos	aunque lo ideal es la fibra óptica, el par trenzado es aceptable	muy alto debido a la redundancia	quizá el más complicado por la estructura del cableado tan abundante	poco económica aunque el costo trae beneficios mucho mayores



HUB

Existen tres tipos de hubs:

- Hubs activos
- Hubs pasivos
- Hubs híbridos

Hubs Activos

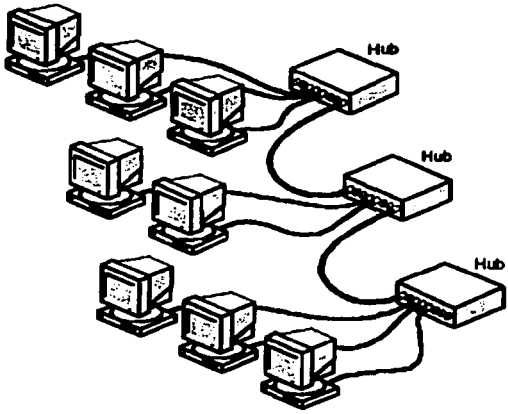
La mayoría de los hubs son activos, lo que quiere decir que regeneran y retransmiten señales de la misma manera que los repeaters. Estos hubs generalmente tienen de ocho a doce puertos para poder conectarse a distintas computadoras. Los hubs activos necesitan de corriente eléctrica para poder funcionar.

Hubs pasivo

Algunos hubs son pasivos, su tarea es permitir que las señales pasen sin regenerarlas. Los hubs pasivos no requieren corriente eléctrica para su funcionamiento.

Hubs híbridos

Estos hubs son los más avanzados, se conectan a diferentes tipos de cables para mantener una conexión en red. Generalmente se conectas a otros hubs denominados subhubs.

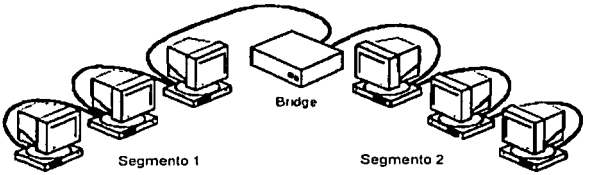


Consideraciones sobre los hubs

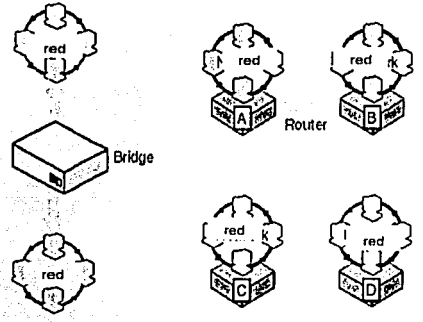
Los sistemas que utilizan hubs son versátiles y ofrecen ventajas con respecto a otros sistemas que no utilizan hub.

En una topología bus si un cable se rompe la red deja de funcionar. Con hubs, sin embargo, si un cable se rompe esto solamente va a afectar un segmento limitado de la red.
Las topologías que utilicen hub tienen los siguientes beneficios:

Los cables pueden ser cambiados
Diferentes puertos pueden ser utilizados para una gran variedad de cables
Permite un control del trafico centralizado
Brigde



El bridge es una unidad funcional que interconecta dos redes de area local (LAN)que usan el mismo protocolo de control de enlace lógico pero que pueden usar distintos protocolos de control de acceso al medio.



Características:

Lectura de todas las tramas de ambas LAN, transmisión y retransmisión de A hacia B y de B hacia A, haciendo uso del protocolo de control de acceso al medio de red.

No modifica el contenido de la trama. Las tramas son copiadas de una LAN y repetidas con el mismo patrón de bit de la otra LAN. Debe tener memoria temporaria para los picos de demandas. Puede conectar más de dos LAN.

Tiene capacidad de Encaminamiento (guía al paquete de transmisión a la LAN de destino) y de Direccionamiento (según el N° de IP de destino que tiene el paquete, el bridge se encarga de direccionar su distribución a la estación de trabajo correspondiente). La LAN se desentiende de dirigir la información de a una determinada LAN

Un Hub no tiene capacidad de encaminamiento y direccionamiento.

Combinaciones híbridas de topologías

Existen tres:

- Estrella bus
- Estrella anillo
- Peer to peer

Estella bus: la estrella bus es una combinación de la topología bus y estrella. En esta topología varias redes de topología estrellas son unidas a través de una comunicación en forma de bus.

Si una computadora deja de funcionar, esto no afecta al resto de la red. Si un hub deja de funcionar, todas las computadoras conectadas a ese hub dejaran de comunicarse entre sí, si este hub esta conectado a otros hubs la conexión entre ellas deja de funcionar.

Estrella anillo: Esta topología es muy parecida a la anterior; esta consiste en un hub que mantiene la conexión en forma de anillo

Peer to peer: esta topología se utiliza en las pequeñas oficinas para transmitir datos entre computadoras. Esta red puede tener la topología estrella o bus.

Seguridad en redes

La seguridad de los datos puede conseguirse por medio de los servidores que posean métodos de control, tanto software como hardware. El objetivo es describir cuales son los métodos más comunes que se utilizan hoy para perpetrar ataques a la seguridad informática (confidencialidad, integridad y disponibilidad de la información) de una organización o empresa, y que armas podemos implementar para la defensa, ya que saber cómo nos pueden atacar (y desde donde), es tan importante como saber con que soluciones contamos para prevenir, detectar y reparar un siniestro de este tipo. Sin olvidar que éstas últimas siempre son una combinación de herramientas que tienen que ver con tecnología y recursos humanos (políticas, capacitación).

Los ataques pueden servir a varios objetivos incluyendo fraude, extorsión, robo de información, venganza o simplemente el desafío de penetrar un sistema. Esto puede ser realizado por empleados internos que abusan de sus permisos de acceso, o por atacantes externos que acceden remotamente o interceptan el tráfico de red.

A esta altura del desarrollo de la "sociedad de la información" y de las tecnologías computacionales, los piratas informáticos ya no son novedad. Los hay prácticamente desde que surgieron las redes digitales, hace ya unos buenos años. Sin duda a medida que el acceso a las redes de comunicación electrónica se fue generalizando, también se fue multiplicando el número de quienes ingresan "ilegalmente" a ellas, con distintos fines. Los piratas de la era cibernética que se consideran como una suerte de Robin Hood modernos y reclaman un acceso libre e irrestricto a los medios de comunicación electrónicos.

Genios informáticos, por lo general veinteañeros, se lanzan desafíos para quebrar tal o cual programa de seguridad, captar las claves de acceso a computadoras remotas y utilizar sus

cuentas para viajar por el Ciberespacio, ingresar a redes de datos, sistemas de reservas aéreas, bancos, o cualquier otra "cueva" más o menos peligrosa. Como los administradores de todos los sistemas, disponen de herramientas para controlar que "todo vaya bien", si los procesos son los normales o si hay movimientos sospechosos, por ejemplo que un usuario esté recurriendo a vías de acceso para las cuales no está autorizado o que alguien intente ingresar repetidas veces con claves erróneas que esté probando. Todos los movimientos del sistema son registrados en archivos, que los operadores revisan diariamente.

Evidentemente las redes, como otros sistemas son susceptibles a múltiples ataques que pueden distorsionar el efecto de la información transmitida o capturarla simplemente. Al aumentar la complejidad de las redes se hace cada vez más patente la necesidad de articular mecanismos de seguridad y protección. El tema es muy amplio por lo que, esquemáticamente, puede decirse que los servicios de seguridad más significativas son la autenticación, el control de acceso, la confidencialidad de datos y la integridad de datos. La autenticación proporciona la verificación de la identidad de la fuente de los datos. El control de acceso proporciona protección contra el uso no autorizado de recursos accesibles a través de la red. La confidencialidad de los datos proporciona protección de datos, por ejemplo, mediante mecanismos de tipo criptográfico. Finalmente la integridad de datos proporciona una validación de la integridad de la información, detectando cualquier modificación, inserción o eliminación de datos. Pueden añadirse algunas medidas de protección adicionales contra el uso no autorizado de manera especifica en redes, como devolución de llamadas, certificados digitales y firewalls.

La devolución de llamadas es una medida común contra el acceso remoto no autorizado. Cuando un módem llama a un sistema, una aplicación especial solicita el número telefónico del cual se esta haciendo la llamada: Si se autoriza el número, el sistema se desconecta y marca es numero. Si este no coincide con las de su lista de números autorizados, el sistema no permite el acceso.

El certificado digital es una encriptación de clave publica. Es el equivalente de una tarjeta de identificación física. Contiene una clave publica y una firma digital. Estos certificados

se obtienen de autoridades certificadas. El receptor de un mensaje encriptado usa la clave publica de la autoridad de certificación para decodificar el certificado digital adjunto al mensaje, verifica que lo emitió la autoridad de certificación y luego obtiene la clave publica del emisor y la información de identificación del certificado.

Un firewall es una herramienta que nos permite proteger nuestra red de posibles ataques externos. Básicamente lo que un firewall es una herramienta que nos permite proteger nuestra red de posibles ataques externos. Lo que conseguimos con un firewall es poder dar acceso a los usuarios de nuestra red privada a Internet, pero en lugar de que cada usuario lo haga desde su computadora (con lo cual tendríamos muchos puntos conflictivos por donde podrían atacar la seguridad de nuestra red) la conexión se realiza a través de un solo host o grupo reducido de ellos con lo que todas las comunicaciones al exterior y las que lleguen de fuera hacia nuestra red pasaran por ese punto, al que denominaremos Zona de Riesgo. Con ello conseguimos monitorear las comunicaciones y solo tenemos que preocuparnos de esa pequeña zona de riesgo en lugar de múltiples puntos de la red.

Un firewall a nivel de red suele ser una barrera de pantalla o una computadora especial que examina las direcciones de los paquetes para determinar si el paquete debe pasar a la red local o se debe impedir el acceso. Un paquete es la información que viaja a través del medio físico y que contiene la información a transmitir, la dirección IP del host emisor, y la dirección IP del host receptor. Los firewals de este tipo utilizan esta información almacenada en los paquetes para controlar su acceso.

En este tipo de firewalls se elaboran una especie de lista negra con las direcciones IP cuyo acceso desea impedir, además puede especificar los servicios que desea restringir su uso. Por ejemplo, puede permitir que los usuarios de Internet accedan a las paginas web de su red privada pero impedir que accedan mediante telnet o al servidor FTP. O bien hacer que se pueda acceder al servidor FTP y descarguen archivos desde él, para impedir que puedan transferir archivos a su servidor FTP. Normalmente estos serán los parámetros que tendrán que tener en cuenta a la hora de diseñar un firewall a nivel de red:

La dirección de origen de la que provienen los datos.
La dirección de destino de los datos.
El protocolo de sesión de los datos; TCP, UDP, ICMP.
El puerto de aplicación de origen y destino del servicio deseado.
Si el paquete es el inicio de una petición de conexión.
Un firewall a nivel de red correctamente configurado será transparente a los usuarios de su red, a no ser que intenten realizar una acción no permitida, como enviar información a una maquina que se encuentra en la lista negra. Además todos los usuarios externos incluidos en la lista negra, no podrán enviar ni recibir paquetes a y desde su red.

El planeamiento de la seguridad en redes es muy importante para su diseño. La seguridad en redes es muy importante para impedir la perdida de datos.

Planeamiento de la seguridad en redes

Es importantísimo en una red el resguardo de datos para impedir el acceso de terceros, como también es importante proteger a una red de daños intencionales que pueden ser provocados por terceros. La seguridad en redes requiere un balance entre facilitarle acceso a los usuarios autorizados, y restringir acceso a los usuarios no autorizados. Las cuatro mayores amenazas en cuanto a la seguridad de los datos en una red son:

Acceso no autorizado
Robo
Acoso digital
Daño intencional o sin intención.

Pese a estas amenazas la seguridad de los datos no siempre es implementada de la mejor manera. La función principal de un administrador es asegurar que la red se mantenga segura y libre de estas amenazas.

Nivel de seguridades

Es muy importante saber que el nivel de seguridad en la red depende del el ambiente que se maneja. Por ejemplo, los datos que guarda un banco comercial tienen que tener una mayor seguridad que los datos guardados en una pequeñas pymes. Esto quiere decir que una red abierta requiere mayor seguridad que una red cerrada.

Políticas de seguridad en redes

Para hacer más segura una red hay que establecer una serie de reglas y políticas para dejar afuera cualquier tipo de improvisto no deseado. Cuando el acceso no autorizado es prevenido los datos están seguros.

Autenticación

Para un usuario pueda acceder a una red tiene que ingresar un user name y un password. El password es un medio de autenticación que permite la defensa contra usuarios no autorizados.
Pese a esto un usuario no autorizado puede llegar a ingresar a una red, y para esto existe una técnica de autenticación llamada handshaking que consiste en formular preguntas privadas que solamente un usuario con acceso autorizados podría contestar.

Capacitación

Es muy importante la capacitación para prevenir cualquier error no intencional que puede llegar a dañar a una red, como ser borrar datos importantes en forma no intencionada.

El administrador debe asegurarse que todos los usuarios de una red sepan su operatoria y cuales son los procedimientos de seguridad que hay que seguir; solamente esto se puede lograr mediante clases de capacitación.

Seguridad en los equipos

El primer paso para mantener la seguridad de los datos, es tener un hardware apropiado, esto consiste en la utilización de equipos de marcas reconocidas que contengan alguna garantía. Cada usuario es responsable de la seguridad de su computadora y de los datos que esta contenga.

Seguridad en los servidores

Es muy importante asegurar los servidores de cualquier tipo de acosos digital accidental o intencional. La forma más simple de proteger a los servidores es encerrarlos en un cuarto con llave con acceso limitado para mantenerlos alejados de posibles sabotajes de terceros.

Seguridad en los cables

Para mantener la seguridad en los cables se debe tenerlos inaccesibles a terceros, desplegando la red de cableado dentro de la estructura del edificio.

LA SEGURIDAD EN LA RED ES UN PROBLEMA CULTURAL MÁS QUE TECNOLÓGICO.

Panelistas participantes de una reunión mensual, coinciden en que el 80 por ciento de las violaciones a la información se da dentro de las organizaciones.

A medida que el comercio de las empresas vía Internet se hace más generalizado, la inseguridad en las transacciones comerciales se vuelve un problema crucial y en constante crecimiento que debe ser contemplado por la alta gerencia en la toma de decisiones y en la implementación de soluciones.

Al hablar sobre la "Seguridad en Internet" nos referimos al gran índice de inseguridad interna de la infraestructura informática de las empresas, así como la falta de una cultura informática necesaria para contemplar estos problemas.

El alto grado de vulnerabilidad de la información transferida por la Internet y la facilidad de ataques externos e internos que se traducen en pérdidas que ascienden hasta miles de dólares en términos de información alterada, robada o perdida.

Según una investigación realizada en 1700 empresas por la empresa, el 75 por ciento de estas han tenido algún problema de seguridad. De éstas el 40 por ciento ha enfrentado problemas de seguridad debido a la falta de apoyo de la alta dirección para invertir en medidas y herramientas de seguridad y sólo el 38 por ciento se debió a la falta de herramientas adecuadas.

Una alternativa es el uso de una llave pública y una privada mediante el protocolo de seguridad Securet Socket Layer (SSL) que autentifica tanto al usuario que envía como al que recibe la información, porque es durante este proceso de transmisión que ocurren la mayor parte de las violaciones en la seguridad.

Más que un problema de tecnología, la seguridad en la transmisión de la información por la Red se debe a la falta de cultura de las organizaciones y de las personas que la integran.
El eslabón más débil de esta cadena en la seguridad la constituye el humano y no el tecnológico, lo cual destaca la importancia de tener una cultura de seguridad, porque no existe en muchas empresas un responsable de la seguridad.

A todos los usuarios se les deben divulgar las políticas de seguridad, además de hacer constantes audítorias para controlar que sean las adecuadas al momento que vive la empresa.

Lo que se necesita no es solamente prevenir un ataque en la seguridad, sino ser capaces de detectar y responder a esta agresión mientras ocurre y reaccionar ante la misma.

Es importante destacar que no existe un control de seguridad único, sino que las empresas deben contar con diversas capas de seguridad en todos los niveles de su información para poder así detectar el problema en algunos de estos puntos antes de que llegue a la información crucial.

LA SEGURIDAD EN LAS REDES: HACKERS, CRACKERS Y PIRATAS

Junto a los avances de la informática y las comunicaciones en los últimos años, ha surgido una hueste de apasionados de estas tecnologías, que armados con sus computadoras y conexiones a redes como Internet, ha logrado humillar a instituciones tan potencialmente seguras como el Pentágono y la NASA. La notoriedad de sus hazañas, su juventud y la capacidad de dejar en evidencia a instituciones muy poderosas, les hace aparecer ante la opinión pública rodeados de un halo de romanticismo

Podemos encontrarnos con diferentes términos para definir a estos personajes: hackers, crackers, piratas, etc., estando normalmente condicionado el calificativo a los objetivos y a los efectos de sus ataques a los sistemas. El término hacker, por ejemplo, se utiliza normalmente para identificar a los que únicamente acceden a un sistema protegido como si

se tratara de un reto personal, sin intentar causar daños. Los crackers, en cambio, tienen como principal objetivo producir daños que en muchos casos suponen un problema de extrema gravedad para el administrador del sistema. En cuanto a los piratas, su actividad se centra en la obtención de información confidencial y software de manera ilícita.

Es muy difícil establecer perfiles de estas personas, porque salvo en los casos en que han saltado a la luz pública como resultado de sus actividades, en su conjunto forman un círculo cerrado e impenetrable. Una aproximación podría ser la de un joven, bastante inteligente, con necesidad de notoriedad, inclinaciones sectarias, y en muchos casos, algo de inadaptación social. Su principal motivación es la de acceder a sistemas protegidos de forma fraudulenta, en una escala que va desde la mera constancia de su éxito, hasta la destrucción de datos, obtención de información confidencial, colapso del sistema, etc. Normalmente los objetivos más apetecibles son los sistemas relacionados con la seguridad nacional, defensa e instituciones financieras, pero ante las posibles consecuencias legales de estos actos optan por otros organismos públicos, las universidades y las empresas.
Existe una serie de grupos que tienen un carácter supranacional, y que se extiende a través de su hábitat natural: Internet. A través de este medio intercambian información y experiencias, al mismo tiempo que logran un cierto grado de organización. Esto ha disparado la alarma en algunos ámbitos gubernamentales, dado que una acción coordinada que afectara a varios sistemas estratégicos de un país puede ser igual de desestabilizadora que las actividades terroristas. En España tenemos ejemplos recientes, como es el caso de Hispahack, que realizó ataques a varios sistemas, incluidos los de algunas universidades. También se ha creado en la Guardia Civil un grupo especializado en todo tipo de delitos informáticos para identificar e investigar a estos modernos delincuentes.

En la ULL, en cambio, hasta este momento no ha existido un riesgo importante ya que, por una parte, había un gran retraso tecnológico en nuestras infraestructuras y, por otro, los sistemas formaban parte de redes que por sus características eran impermeables a dichos ataques. Pero la situación ha cambiado: la ejecución del Plan Integral de Comunicaciones ha elevado tanto nuestras posibilidades que nos permite la integración en una única red de todos nuestros sistemas informáticos, con lo que conlleva a la hora de prestar servicios a los

usuarios. Esto tiene su contrapartida, y es que el número de servicios que se ofrecen es directamente proporcional a los riesgos que se asumen, y sobre todo porque el primer enemigo al que habría que considerar podrían ser los propios usuarios.

De todas formas, el exceso de prudencia es contrario a la innovación y, por tanto, se están adoptando medidas que garanticen una cobertura suficiente: la adquisición de herramientas de software para la gestión de red, firewalls (cortafuegos, programas especializados en la protección de redes y sistemas), y software de auditoria; la elaboración de planes de seguridad tanto física como lógica y de las políticas correspondientes; y, por último, la mentalización de los usuarios para el correcto uso de los servicios que se prestan. De todas formas, la total seguridad nunca se podrá alcanzar, a menos que coloquemos los sistemas detrás de un muro infranqueable. Pero entonces nos encontraríamos con una red que es una auténtica autopista, pero por la que sólo circularían el correo electrónico y las páginas web. Además, esto significa un incentivo para que los administradores de los sistemas y responsables de seguridad seamos mejores en nuestro trabajo, ya que cada ataque con éxito pone en evidencia nuestras deficiencias.

1.3 Diferencias del sistema operativo UNIX Solares en arquitectura RISC y CISC.

En el debate de cual arquitectura es mejor, se puede encontrar diferentes preguntas en funcion de RISC (Reduced Instrition Set Cumputer) o CISC (Complex Instrution Set Computer) se centra en un particular aspecto importante: es mejor completar una tarea de gran tamaño ejecutando muchas tareas pequeñas pero sencillas (RISC) o efectuando unas cuantas tareas de tamaño mediano (CISC). Con la estrategia CISC se reduce el número de instrucciones requeridas para una tarea, pero aumenta la complejidad del procesador. Con el método RISC aumenta el número de instrucciones pero se simplifica la estructura del procesador. Al controlar con cuidado la complejidad del procesador, los arquitectos RISC han logrado optimizar su rendimiento y en la actualidad los componentes favorecen a la estrategia RISC, aunque en el campo de la mercadotecnia se dice otra historia, ya que aun es muy elevado el costo de producción de los procesadores RISC a comparación de los CISC.

Los inicios de la tecnología RISC surge en el ambiente académico, en 1980, el Dr. David A. Paterson inicio un proyecto denominado RSIC I, que obtuvo resultados en tan solo 19 meses, seguido de RISC II, SOAR (Smaltalk on a RISC) y SPUR (Symbolic Processing on a RISC). El resultado directo, fue la creación de una maquina que fuese capaz de mayores velocidades de ejecución a menores velocidades de reloj y que requiriese menores esfuerzos de diseño. Casi simultáneamente, en la Universidad de Stanford , el Dr. John Henney inicio también un proyecto de implementación RISC, denominado MIPS, seguido por el sistema MIPS-XMP, enfocados hacia el proceso simbólico, demostrando las capacidades de velocidad de la arquitectura RISC. Ambos esfuerzos se involucraron rápidamente en proyectos de productos comerciales, en el caso de Henney fue una de los fundadores de MIPS Computer en el caso de Paterson se involucro como asesor en el proyecto de SPARC.

ARQUITECTURA CISC.

Las computadoras que manejan la arquitectura CISC cuentan con algunos privilegios y defectos que heredan del micro código. En el procesador se tiene una velocidad considerablemente más rápida que la memoria principal, esto significa que conviene manejar un amplio abanico de instrucciones complejas cuyo significado se traduce al de varias instrucciones sencillas.

Históricamente la metodología CISC se fundamento en apoyar al programador en lenguaje ensamblador. Como una instrucción CISC puede realizar varias operaciones, los programadores en lenguaje ensamblador no tienen que escribir tantas instrucciones para llevar a cabo una tarea de gran magnitud. Sin embargo, los resultados de pruebas dan otros resultados, en estudios se examino la frecuencia de utilización de las diferentes instrucciones, se observo que el 80% del tiempo era consumido por solo el 20% de las instrucciones, con prioridad de los almacenamientos (STORE), cargas (LOAD) y bifurcaciones (BRANCH).

ARQUITECTURA RISC.

En la implementación directa de instrucciones de una maquina era la estrategia preferida y las maquinas tenían conjuntos de instrucciones muy sencillos. Sin embargo, al aumentar la velocidad de los procesadores, el ancho de banda se convirtió en el cuello de botella de la interpretación de las instrucciones. Para reducir el impacto de este cuello de botella, muchos diseñadotes aumentaron la complejidad de las instrucciones en el conjunto de instrucciones. El fin buscado era fácil de predecir el cual fue reducir el numero de instrucciones en el programa de lenguaje ensamblador mediante un incremento en la cantidad de trabajo realizado por cada instrucción.

Al transcurso del tiempo las instrucciones se hicieron cada vez mas complicadas, los diseñadotes empezaron a usar la microprogramación para reducir la complejidad del diseño de los procesadores, sin embargo estos conjuntos de instrucciones llegaron a ser tan complicados que se torno muy difícil determinar que tan efectivo era el conjunto de instrucciones. Muchas de las instrucciones mas complejas se utilizaban tan poco que su beneficio era dudoso. Tal vez la mayor preocupación era que los compiladores no podían aprovechar las instrucciones tan complejas.

Sin mas, se dio la mirada a la estrategia RISC ya que para el diseño de procesadores se basa en el uso de instrucciones muy sencillas. El conjunto de estas tiene la sencillez necesaria como para permitir una implementación directa con **pipelines** (conductos segmentados). De hecho, gran parte del conjunto de instrucciones esta diseñado para la implementación con pipelines. El conjunto de instrucciones tiene una sencillez suficiente como para permitir estudios cuantitativos de los compromisos en el conjunto de instrucciones. La memoria cache de instrucciones ayuda a superar el problema de ancho de banda de las instrucciones.

CISC vs. RISC

El dilema comparativo surge al evaluar las ventajas netas. En función a las cualidades de cada arquitectura hay diversos mitos que se han producido a través del tiempo y la mercadotecnia como son:

Los procesadores RISC ofrecen peor soporte para los lenguajes de alto nivel o HLL (High Level Lenguaje) que lo CISC. Esta creencia se argumenta en que un conjunto de instrucciones de "alto nivel" (CISC) es mejor soporte para lenguaje de alto nivel. Esto no tiene buen fundamento ya que los lenguajes de alto nivel, implica que el programador solo interacciona con la computadora a través del propio lenguaje de alto nivel (programación, depuración, mensajes del sistema, etc), por lo cual todos los problemas que puedan surgir a "bajo nivel" deben ser transparentes y desconocidos para el programador. Por lo tanto son nulas las consecuencias para el programador y los lenguajes de alto nivel. En la forma de cómo se implementan las funciones, en función del tipo de CPU utilizado.

En gran parte es mas difícil escribir compiladores RISC que CISC. Tomando en cuenta que los procesadores CISC tienen un mayor numero de instrucciones y modos de direccionamiento, existen por tanto mas formas de hacer la misma tarea, lo que implica confundir tanto al compilador como a quien lo usa. Por ello, subjetivamente es posible escoger una forma de hacerlo poco adecuada, por el tipo de instrucciones o por el tiempo de ejecución que requieren. En cambio, en un procesador RISC, hay menos opciones, por lo que el compilador es mas simple, aunque se genere, habitualmente entre un 20-30% mas código; a cambio, se consigue un incremento de la velocidad de hasta un 500% dependiendo del tipo de arquitectura de la computadora y el paralelismo de uso de los CPUs, esto quedara mas claro en los puntos 1.5 y 1.6 de este capítulo.

Un programa es más rápido cuanto mas pequeño es. Esto no es necesariamente verídico, ya que la velocidad a la que un programa puede ser ejecutado no depende en absoluto de su tamaño, sino del tiempo de ejecución de cada una de sus instrucciones. Dado que las instrucciones RISC son más rápidas, y admiten mejor los pipelines, puede haber mayor paralelismo y simultaneidad en la ejecución de pequeñas secciones de código. Dicha sección de código puede ser ejecutada en una fracción del tiempo que requiere una sola instrucción CISC.

Por lo anterior se tienen diferentes puntos por los cuales se pensaría en cual es el problema en la actualidad por el cual no se ha dado el crecimiento suficiente sobre la arquitectura RISC en comparación de la CISC, se puede decir que son dos principalmente:

a) La arquitectura RISC, a tenido diferentes cambios en función de la compatibilidad en aplicaciones soportadas, como ejemplo en los procesadores SPARC II, SPARC IIi con el nuevo SPARC III.

b) Otra razón es aunado al punto anterior debido a la demanda los precios en dicha tecnología no han bajado para ser razonablemente aceptable para un usuario común, esto lo vemos en la actualidad en razón ha que los equipos con procesador RISC (SPARC) están orientado a las medianas y grandes empresas.

1.4 Proyección de toma de decisiones en ambientes de producción críticos.

En la proyección sobre toma de decisiones se debe seguir algún método que sirva como punto base para realizar una metodología sobre acciones, demandas y sucesos de servicios en un ambiente productivo.

El problema de la Decisión, motivado por la existencia de ciertos estados de ambigüedad que constan de proposiciones verdaderas (conocidas o desconocidas), es tan antiguo como la vida misma. Podemos afirmar que todos los seres vivientes, aún los más simples, se enfrentan con problemas de decisión. Por ejemplo, un organismo unicelular asimila partículas de su medio ambiente, unas nutritivas y otras nocivas para él. La composición biológica del organismo y las leyes físicas y químicas determinan qué partículas serán asimiladas y cuáles serán rechazadas.

Conforme aumenta la complejidad del ser vivo, aumenta también la complejidad de sus decisiones y la forma en que éstas se toman. Así, pasamos de una toma de decisiones guiada instintivamente, a procesos de toma de decisiones que deben estar guiados por un pensamiento racional en el ser humano. La Teoría de la Decisión tratará, por tanto, el estudio de los procesos de toma de decisiones desde una perspectiva racional.

Los modelos en la toma de decisiones que a continuación se describirán caerán en una de las cuatro categorías generales siguientes:

Categorías	Consecuencias
Certidumbre	Deterministas
Riesgo	Probabilísticas
Incertidumbre	Desconocidas
Conflicto	Influidas por un oponente

TABLAS DE DECISIÓN BAJO CERTIDUMBRE

En los procesos de decisión bajo certidumbre se supone que el verdadero estado de la naturaleza es conocido por el decidor antes de realizar su elección, es decir, puede predecir con certeza total las consecuencias de sus acciones. Esto es equivalente a considerar n=1 en la descripción de la tabla de decisión, dando lugar a siguiente tabla trivial:

	Estado de la Naturaleza
Alternativas	e_1
a_1	x_{11}
a_2	x_{21}
. . .	. . .
a_m	x_{m1}

Conceptualmente, la resolución de un problema de este tipo es inmediata: basta elegir la alternativa que proporcione un mejor resultado, es decir se selecciona como alternativa óptima aquella alternativa ak tal que:

xk1 = max {xi1 : 1£i£m}

El problema de decisión se reduce, por tanto, a un problema de optimización, ya que se trata de escoger la alternativa que conduzca a la consecuencia con mayor valor numérico asociado.

Básicamente, un problema de optimización puede expresarse en forma compacta como sigue:

$$\max \{ f(x) : x \in S\}$$

donde:

S es el conjunto de alternativas o conjunto factible. Se trata de un subconjunto del espacio euclídeo $\Re n$, que puede contener un número finito o infinito de elementos.
f: S a $\Re$ es la denominada función objetivo, que asigna a cada alternativa una valoración, permitiendo su comparación.
x representa el vector n-dimensional que describe cada elemento del conjunto factible. Cada una de sus componentes recibe el nombre de variable de decisión.

TABLAS DE DECISIÓN BAJO INCERTIDUMBRE

En los procesos de decisión bajo incertidumbre, el decidor conoce cuáles son los posibles estados de la naturaleza, aunque no dispone de información alguna sobre cuál de ellos ocurrirá. No sólo es incapaz de predecir el estado real que se presentará, sino que además no puede cuantificar de ninguna forma esta incertidumbre. En particular, esto excluye el conocimiento de información de tipo probabilística sobre las posibilidades de ocurrencia de cada estado.

REGLAS DE DECISIÓN

A continuación se describen las diferentes reglas de decisión en ambiente de incertidumbre.

Criterio de Wald
Criterio Maximax
Criterio de Hurwicz
Criterio de Savage
Criterio de Laplace

Los criterios descritos anteriormente no son los únicos que pueden utilizarse en ambiente de incertidumbre; muchas otras reglas de decisión son válidas en este contexto, por lo que parece preciso determinar propiedades que hagan un criterio preferible a otro.

Con este propósito vamos a describir los axiomas o principios de racionalidad basados en la propuesta realizada por Milnor en 1954, y que pueden ser considerados propiedades razonables para ser verificadas por toda regla de decisión.

Axioma 1: Orden
El criterio debe proporcionar una ordenación total de las alternativas del problema. Esta propiedad es deseable, pues en caso de no darse existirían alternativas no comparables, siendo preciso un nuevo criterio para dilucidar entre elementos.
Axioma 2: Simetría
El criterio debe ser simétrico, es decir, independiente del orden fijado a priori en el conjunto de alternativas y del orden en que se definan los estados de la naturaleza.
Axioma 3: Linealidad
La relación de orden establecida por el criterio no debe cambiar si los resultados xij son reemplazados por otros yij tales que
$yij = lxij + m$ con $l>0$
Axioma 4: Dominancia fuerte
Si en una tabla de decisión existen dos alternativas ai y ak tales que $xij>xkj$ para todos los

estados de la naturaleza ej, entonces el criterio debe asignar valores a las alternativas de modo que T(ai)>T(ak).

Axioma 5: Independencia de alternativas irrelevantes

El criterio debe ser abierto, es decir, el valor asignado por dicho criterio a una alternativa no debe variar al ser definido en otro conjunto de alternativas que contenga al primero con las mismas valoraciones (el orden entre dos alternativas no cambia por la adición de una nueva alternativa).

Esta propiedad es muy importante, ya que garantiza que al aumentar el conjunto de alternativas, los cálculos efectuados con anterioridad siguen siendo válidos.

Axioma 6: Linealidad de columnas

La relación de orden establecida por el criterio no debe cambiar si se añade una constante a todos las valoraciones correspondientes a un estado de la naturaleza.

Axioma 7: Independencia de permutación de filas

Si en una tabla de decisión existen dos alternativas ai y ak tales que el conjunto de valoraciones de la alternativa ak es una permutación del conjunto de valoraciones correspondiente a la alternativa ai, entonces el criterio debe asignar idéntico valor a ambas, es decir, T(ai)=T(ak).

Axioma 8: Independencia de duplicación de columnas

El criterio debe ser invariante por extensión, es decir, el orden establecido por el criterio no debe cambiar si se añade una nueva columna (estado de la naturaleza) idéntica a alguna columna ya existente.

La siguiente tabla resume la compatibilidad de los diferentes criterios analizados con los axiomas anteriores. El carácter S indica que el criterio satisface el correspondiente axioma, mientras que N indica que no lo verifica.

	Wald	Hurwicz	Savage	Laplace	
Axioma 1	S	S	S	S	Orden
Axioma 2	S	S	S	S	Simetría
Axioma 3	S	S	S	S	Linealidad
Axioma 4	S	S	S	S	Dominancia fuerte
Axioma 5	S	S	N	S	Independencia de alternativas irrelevantes
Axioma 6	N	N	S	S	Linealidad de columnas
Axioma 7	S	S	N	S	Independencia de permutación de filas
Axioma 8	S	S	S	N	Independencia de duplicación de columnas

TABLAS DE DECISIÓN BAJO RIESGO

Los procesos de decisión en ambiente de riesgo se caracterizan porque puede asociarse una probabilidad de ocurrencia a cada estado de la naturaleza, probabilidades que son conocidas o pueden ser estimadas por el decidor antes del proceso de toma de decisiones.

REGLAS DE DECISIÓN

Los diferentes criterios de decisión en ambiente de riesgo se basan en estadísticos asociados a la distribución de probabilidad de los resultados. Algunos de estos criterios se aplican sobre la totalidad de las alternativas, mientras que otros sólo tienen en cuenta un subconjunto de ellas, considerando las restantes peores, por lo no que están presentes en el proceso de toma de decisiones.

Representaremos por R(ai) los resultados asociados a la alternativa ai, y por P(ai) la distribución de probabilidad correspondiente a tales resultados, esto es, el conjunto de valores que representan las probabilidades de ocurrencia de los diferentes estados de la naturaleza:

R	xi1	xi1	. . .	xi1
P	p1	p2	. . .	pn

Los principales criterios de decisión empleados sobre tablas de decisión en ambiente de riesgo son:

Criterio del valor esperado
Criterio de mínima varianza con media acotada
Criterio de la media con varianza acotada
Criterio de la dispersión
Criterio de la probabilidad máxima

Todos estos criterios serán aplicados al problema de decisión bajo riesgo cuya tabla de resultados figura a continuación:

Decisión bajo riesgo: Ejemplo

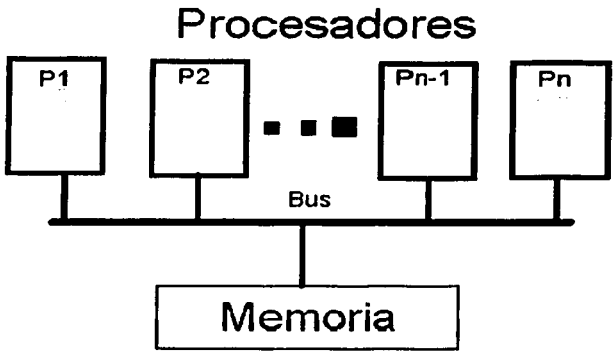
	Estados de la Naturaleza			
Alternativas	e_1	e_2	e_3	e_4
a_1	11	9	11	8
a_2	8	25	8	11
a_3	8	11	10	11
Probabilidades	0.2	0.2	0.5	0.1

Adicional a las metodologías anteriores para el apoyo de toma de decisiones, un factor que siempre estará presente es el sentido de costo-beneficio sobre el producto que se este trabajando.

1.5 Sistemas con arquitectura SMP, MPP y SPP.

Arquitectura SMP (Multiprocesamiento Simétrico).

El Multiprocesamiento simétrico (symmetric multiprocessing) tiene un diseño simple pero aun así efectivo. En SMP múltiples procesadores comparten la memoria RAM y el bus del sistema. Este diseño es también conocido como estrechamente acoplado (tightly coupled), o compartido todo (shared everything).



Debido a que SMP comparte globalmente la memoria RAM, tiene solamente un espacio de memoria, lo que simplifica tanto el sistema físico como la programación de aplicaciones.

Este espacio de memoria único permite que un sistema operativo con multiconexión (multithreaded operating system) distribuya las tareas entre varios procesadores, o permite que una paliación obtenga la memoria que necesita para una simulación compleja. La memoria globalmente compartida también vuelve fácil la sincronización de los datos. Este

tipo de diseño es una de los mas maduros debido a su tiempo de desarrollo, este apareció en la década de los ochenta (1983) con la supercomputadora Cray X-MP. Ahora veamos la desventaja de tener una memoria global, pensemos que conforme se van añadiendo procesadores, el tráfico en el bus de memoria se satura. Al agregar mas memoria cache a cada procesador se puede reducir algo el tráfico en el bus, pero el bus generalmente se convierte en un cuello de botella al manejarse alrededor de ocho o mas procesadores. SMP es tomada como una tecnología poco escalable, enseguida se muestra una grafica sobre el rendimiento al sumar CPUs a este tipo de arquitectura. El la figura 1.a se muestra el comportamiento de la arquitectura SMP en función de desempeño de CPUs.

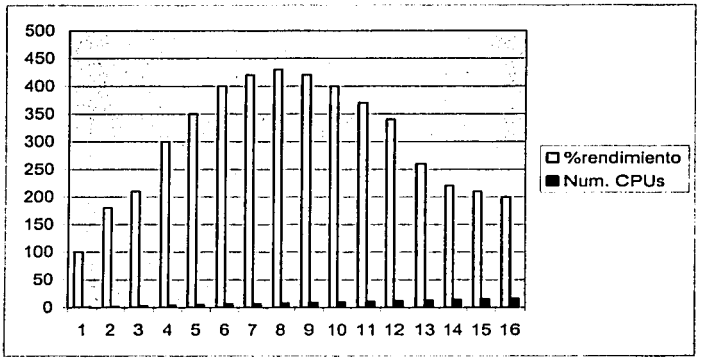
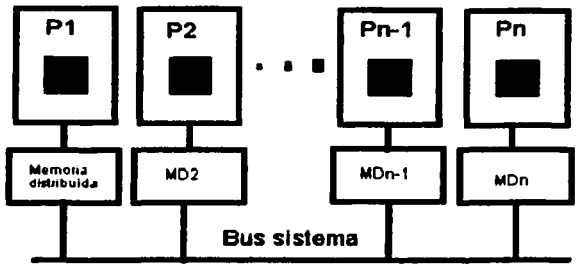


Figura 1.a

Arquitectura MPP (Masivamente paralelo).

El procesamiento masivamente paralelo (Massively parallel processing). Debido a los sistemas SMP y sus inconvenientes sobre los cuellos de botella en el bus de memoria, los sistemas MPP no utilizan memoria compartida. En lugar de una memoria global, se realiza una distribución de la memoria RAM entre los procesadores de modo que se parezca a una red (cada procesador con su memoria distribuida asociada es similar a una computadora dentro de una red de procesamiento distribuido), debido a este tipo de distribución en los

recursos de la RAM, este tipo de arquitectura es también conocida como dispersamente acoplada (loosely coupled), o compartiendo nada (shared nothing).



El paso de mensajes mueve datos a traves del sistema

Solo en el caso de tener que tomar memoria fuera de la asignada, los procesadores utilizan un esquema de paso de mensajes análogo a los paquetes de datos en redes. Este tipo de sistemas lógicamente reducen el tráfico del bus, debido a que cada sección de memoria observa únicamente aquellos accesos que le están destinados, en lugar de observar todos los accesos, como ocurre en los sistemas SMP. Únicamente cuando un procesador no dispone de la memoria RAM suficiente, utiliza la memoria RAM sobrante de los otros procesadores. Esto permite sistemas MPP de gran tamaño con cientos y aun miles de procesadores. MPP es una tecnología escalable. En seguida se muestra una grafica del rendimiento al sumar CPUs. La figura 1.b muestra el comportamiento de la arquitectura MPP en función de desempeño de CPUs.

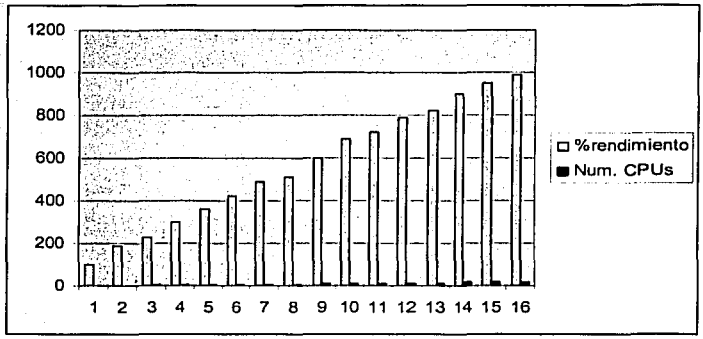


Figura 1.b

El RS/6000 de IBM es un claro ejemplo de un sistema MPP que presenta una ligera variante respecto al esquema anteriormente mencionado. Los procesadores del RS/6000 se agrupan en nodos de 8 procesadores, los que utilizan una única memoria compartida (tecnología SMP). A su vez estos nodos se agrupan entre si utilizando memoria distribuida para cada nodo (tecnología MPP). De este modo se consigue un diseño más económico y con mayor capacidad de crecimiento.

Ahora bien tendremos que comentar lo malo de la tecnología MPP es que a nivel de programación se vuelve difícil, debido a que la memoria se rompe en pequeños espacios separados. Sin la existencia de un espacio de memoria global compartido, correr y escribir una paliación que requiere una sincronización de datos entre tareas ampliamente distribuidas también se vuelve difícil, particularmente si un mensaje debe pasar por muchas fases hasta alcanzar la memoria del procesador destino.

Escribir y ejecutar una paliación MPP también requiere estar al tanto de la organización de la memoria manejada por el programa. Donde sea necesario o se requiera insertar comandos de paso de mensajes dentro del código del programa. Adicional a esto se complica el diseño del programa, tales comandos pueden crear dependencias de hardware en las aplicaciones. Sin embargo, mayoría de los vendedores de computadoras han adoptado estándares de dominio publico para este tipo de problema como el conocido PVM (maquina virtual paralela) o en fase de desarrollo otro llamado interfaz de paso de mensajes (MPI), para implementar el mecanismo de paso de mensajes.

Arquitectura SPP (Procesamiento paralelo escalable)

Las últimas preguntas de como superar las dificultades de SMP y MPP se plantearon desde sus comienzos, por lo cual surge la arquitectura paralela SPP. Esta es un híbrido de SMP y MPP la cual utiliza una memoria jerárquica de dos niveles para alcanzar la escalabilidad. La primera capa de memoria consiste de un nodo que es esencialmente un sistema SMP completo, con múltiples procesadores y su memoria globalmente compartida.
Se construyen sistemas SPP grandes interconectados dos o más nodos a través de la segunda capa de memoria, de modo que esta capa aparece lógicamente, ante los nodos, como una memoria global compartida.

La memoria de dos niveles reduce el tráfico de bus debido a que solamente ocurren actualizaciones para mantener coherencia de memoria. Por tanto, SPP ofrece facilidad de programación del modelo SMP, a la vez que prevé una escalabilidad similar a la de un diseño MPP. En la figura 1.c se muestra como funciona la arquitectura SPP en función de desempeño de CPUs.

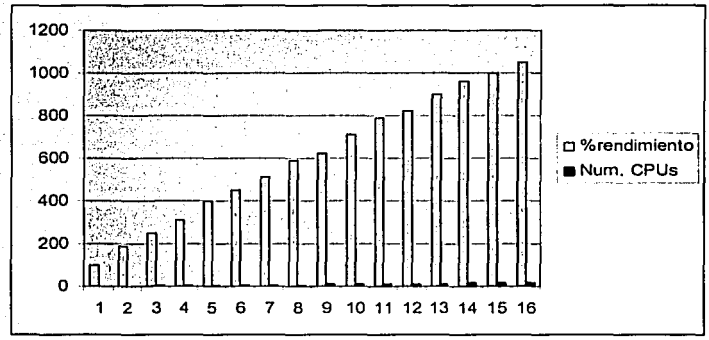


Figura 1.c

CAPÍTULO II SISTEMAS DE MONITOREO ACTUALES PARA AMBIENTES UNIX.

2.0 Sistemas de monitoreo para sistemas locales y remotos.

Entre sistemas locales y remotos de monitoreo hay que nombrar que hay diferencias fundamentales desde la generación de datos hasta la manipulación, e interpretación de estos, toda la logística de cada sistema se va haciendo mas trabajoso; su aplicación cada vez que se va extendiendo en toda la red, los puntos medulares en las diferencias son:

- Seguridad del medio de comunicación.
- Obtención de datos.
- Procesamiento de datos.
- Presentación de datos.
- Agentes externos para monitoreo.
- Precio comercial.

En el caso de los sistemas de monitoreo para UNIX en la actualidad se basan en utilizar agentes preinstalados en clientes, los cuales entregan la información mediante un servicio de cifrado. La seguridad en fundamental en sistemas remotos, no cumpliéndose por razones lógicas en sistemas locales, ya que la información no viaja en canales de comunicación sin embargo, cada vez son mas las necesidades de tener sistemas remotos de monitoreo debido a diferentes puntos de alcances tecnológicos. El procesamiento y presentación se realizan normalmente en tiempos distintos, también por obvias razones, aunque en la actualidad encontramos muchos sistemas en sitios de software freeware (software gratis) o shareware (software con un tiempo de prueba definido), en el caso particular de equipos Unix a comparación de sistemas windows se podría decir que se encuentra una relación muy marcada 1 a 100 si bien nos va, desfavoreciendo al sistema Unix.

Desgraciadamente estos programas se van reduciendo más y más si los enfocamos a plataformas en donde la arquitectura es RISC, en donde en definitiva los sistemas de monitoreo se cuentan con los dedos de las manos. Adicional a esto los precios normalmente son muy elevados y la palabra freeware queda en el olvido.

También es bueno nombrar que un sistema adecuado de monitoreo debe tener puntos de toma de decisión, es decir tener un histórico en el cual uno pueda consultar referencias en distintos intervalos de tiempos, sin esto simplemente el sistema servirá en el momento de su uso, pasando al olvido datos de gran importancia que tal vez, en su momento de generación, pasaron desapercibidos, no dándole la importancia necesaria. Se debe tener una base de datos de decisiones.

Otro punto: el sistema debe procesar y presentar los datos de forma muy clara, en la cual se pueda interpretar rápidamente dichos resultados de monitoreo, y por último, los datos, aunque se tenga la propiedad de almacenamiento, debe tener el punto de generación de datos en tiempo real, con control de intervalos de tiempos con alcance mínimo de 1 minutos.

2.1 Análisis de costos implícitos en sistemas de monitoreo actuales.

Todo sistema de monitoreo informático, contrae gastos implícitos o los llamados gastos ocultos, estos egresos se orientan principalmente a cuatro puntos:

- Mantenimiento del sistema (Software y Hardware)
- Auditoria del sistema
- Capacitación de personal (actualizaciónes y renovación de personal)
- Renovación de servicios con proveedor (garantía, actualizaciones software, y soporte técnico)

En el punto de mantenimiento debe saberse clasificar que es lo que se buscará al pedir un mantenimiento del sistema; la asesoria puede ser llevada como asesoria externa o interna. En cada caso se debe saber que es lo que busca, el punto mas destacado de estos es el de mantenimiento. La clasificacion se da de la siguiente manera:

***Mantenimiento** correctivo.*

Independientemente de cuán bien diseñado, desarrollado y probado está un sistema o aplicación, ocurrirán errores inevitablemente. Este tipo de mantenimiento se relaciona con la solución o la corrección de problemas del sistema. Atañe generalmente a problemas no identificados durante la fase de ejecución. Un ejemplo de mantenimiento correctivo es la falta de una característica requerida por el usuario, o su funcionamiento defectuoso.

***Mantenimiento** para fines específicos.*

Este tipo de mantenimiento se refiere a la creación de características nuevas o a la adaptación de las existentes según lo requieren los cambios en la organización o los usuarios, por ejemplo, los reglamentos internos de la organización.

Mantenimiento para mejoras.

Se trata de la extensión o el mejoramiento del desempeño del sistema, ya sea mediante el agregado de nuevas características, o el cambio de las existentes. Un ejemplo de este tipo de mantenimiento es la conversión de los sistemas de texto a GUI (interfaz gráfica de usuarios), en caso de no contar con esto.

Mantenimiento preventivo.

Este tipo de mantenimiento es probablemente uno de los más eficaces en función de los costos, ya que si se realiza de manera oportuna y adecuada, puede evitar serios problemas en el sistema.

En el punto de mantenimiento debe también evaluarse el impacto que se tendrá por parte del hardware que se tiene y que se debe considerar en un crecimiento de infraestructura.

Otro tipo de costo que se ha propagado cada vez más, es el de realizar audítoria sobre los sistemas ya establecidos, lo cual incluye tratar desde la gestión de equipos, gestión de costos en administrativos y personal, análisis y diagnostico de infraestructura utilizada, el mismo ejercicio de audítoria produce una logística de gastos dependiendo de que tan seguido se realice.

Por último dos puntos son básicos en cada empresa, los costos que se realizarán sobre la capacitación del personal y la relación que se lleve con los proveedores de servicios, estos dos puntos que aunque no necesariamente son constantes, en cada caso que se produzcan, pueden dar altos gastos en un plazo muy corto, por lo cual sé deben estimar siempre en la compra de algún bien para mejorar la productividad de la empresa.

Los costos implícitos en los sistemas de informática por parte de servicios ha tenido un crecimiento, en los últimos años, de estos solo del 2.1-3% en sistemas de monitoreo.

MERCADO MEXICANO DE TECNOLOGÍAS DE INFORMACIÓN Y TELECOMUNICACIONES,1995-2002
(Millones de dólares)

Concepto	1995	1999	2000	2001 e/	2002
Total	**22 975**	**19 599**	**22 219**	**24 625**	**26 929**
Tecnologías de la Información	**2 126**	**4 664**	**5 716**	**5 929**	**6 186**
Equipo	1 247	2 513	3 328	3 444	3 600
Software	234	522	608	632	631
Servicios	645	1 629	1 780	1 853	1 955
Telecomunicaciones	**9 362**	**14 935**	**16 503**	**18 696**	**20 743**
Equipo	735	2 041	2 449	2 484	2 538
Servicios	8 627	12 895	14 054	16 212	18 205

FUENTE: INEGI a partir del 2000, datos del evento "Tendencias 2000", organizado por Select-IDC, octubre 2001. Para años anteriores, cifras elaboradas con datos de eventos previos.

2.2 Parámetros de desempeño en dispositivos de hardware.

Los parámetros de desempeño se basan principalmente en los siguientes dispositivos que tienen normalmente cualquier equipo de cómputo, independientemente del sistema operativo o marca de proveedor, los cuales son:

- CPU
- Disco
- Memoria
- Dispositivo de red

Con los anteriores parámetros correspondientes se puede dar una opinión bastante amplia del desempeño que puede tener algún sistema residente en el equipo de cómputo que se quiera analizar, y por lo tanto saber cual serian los posibles problemas o recomendaciones en el crecimiento de la demanda de servicios del sistema tratado.

Enseguida se tocarán a fondo los parámetros de desempeño sobre los puntos anteriormente nombrados, los cuales se basan las estadísticas sobre monitoreo de los sistemas UNIX, así mismo se darán ejemplos de los datos arrojados por comandos comúnmente usados en la administración de estos sistemas.

CPU

El CPU como siempre se ha nombrado es el cerebro de todo sistema, es el dispositivo por definición principal, primordial y fundamental. Los parámetros más comunes de monitoreo del CPU son porcentajes de ocupación sobre procesos, clasificados principalmente en tres rubros, los cuales son: % de ocupación por procesos de usuarios, % de ocupación por procesos del sistema (kernel) y % de ocupación por procesos de I/O. Enseguida se muestra un ejemplo de la salida del comando "**sar**" en el caso del sistema operativo Solares 2.9:

```
bash-2.05# sar 1 10

SunOS dns1 5.9 Generic_112233-01 sun4u    03/25/2003

13:29:53   %usr   %sys  %wio  %idle
13:29:54     98      2     0      0
13:29:55     99      1     0      0
13:29:56    100      0     0      0
13:29:57    100      0     0      0
13:29:58     99      1     0      0
13:29:59     99      1     0      0
13:30:00     99      1     0      0
13:30:01    100      0     0      0
13:30:02    100      0     0      0
13:30:03     88     12     0      0
```

Un último parámetro en el cual se puede dar un panorama general del % sin uso del CPU es el parámetro nombrado como %idle. Los parámetros anteriores cambiarán en función de la arquitectura del CPU (RISC o CISC ya nombradas en el capítulo anterior) y de la optimización de la paliación que se pueda hacer.

DISCO

El caso del disco es uno de los dispositivos con más demanda, debido ha esto debe ser planeado a conciencia, en la actualidad se tienen de diferentes tecnologías (IDE, SCSI, y Fibra) y de diferentes velocidades normalmente medidos por RPM (Revoluciones por minutos), entre las nombradas detallaremos las diferentes tecnologías para saber sus ventajas o inconvenientes. Un punto muy importante en el esquema de desempeño de los discos es el tipo de arreglo RAID (*redundant array of independent [inexpensive] disks*) que se maneje ya que este puede mejorar drásticamente dicho desempeño o empobrecerlo en demasía.

Tipos de tecnologías en discos:

· **IDE**: Es un interface a nivel de sistema que cumple la norma *ANSI* de acoplamiento a los *AT* y que usa una variación sobre el bus de expansión del *AT* (por eso también llamados discos tipo *AT*) para conectar una unidad de disco a la *CPU*, con un valor máximo de transferencia de 4 Mbytes por segundo. En principio, *IDE* era un término genérico para cualquier interface a nivel de sistema. La especificación inicial de este interface está mal definida. Es más rápida que los antiguos interfaces *ST506* y *ESDI* pero con la desaparición de los *ATs* esta interfase desaparecerá para dejar paso al *SCSI* y el *SCSI-2*.

Íntimamente relacionado con el *IDE*, tenemos lo que se conoce como *ATA*, concepto que define un conjunto de normas que deben cumplir los dispositivos. Años atrás la compañía *Western Digital* introdujo el standard *E-IDE* (*Enhanced IDE*), que mejoraba la tecnología superando el límite de acceso a particiones mayores de 528 Mb. y se definió *ATAPI*, normas para la implementación de lectores de *CD-ROM* y unidades de cinta con interfaz *IDE*. *E-IDE* se basa en el conjunto de especificaciones *ATA-2*. Como contrapartida comercial a *E-IDE*, la empresa *Seagate* presento el sistema *FAST-ATA-2*, basado principalmente en las normas *ATA-2*. En cualquier caso a los discos que sean o bien *E-IDE* o *FAST-ATA*, se les sigue aplicando la denominación *IDE* como referencia.

Para romper la barrera de los 528 Mb. las nuevas unidades *IDE* proponen varias soluciones:

- El **CHS** o traducción entre los parámetros que la *BIOS* contiene de cilindros, cabezas y sectores (ligeramente incongruentes) y los incluidos en el software de sólo lectura (*Firmware*) que incorpora la unidad de disco.

- El **LBA** (dirección lógica de bloque), que estriba en traducir la información *CHS* en una dirección de 28 bits manejables por el sistema operativo, para el controlador de dispositivo y para la interfaz de la unidad.

Debido a la dificultad que entraña la implementación de la compatibilidad *LBA* en *BIOS*, muchos de las computadoras personales de fabricación más reciente, continúan ofreciendo únicamente compatibilidad con *CHS*. El techo de la capacidad que permite las solución *CHS* se sitúa en los 8,4 Gb, que por el momento parecen suficientes.

· **SCSI**: Es un interface a nivel de sistema, diseñado para aplicaciones de propósito general, que permite que se conecten hasta siete dispositivos a un único controlador. Usa una conexión paralela de 8 bits que consigue un valor máximo de transferencia de 5 Mbytes por segundo. Actualmente se puede oír hablar también de *SCSI-2* que no es más que una versión actualizada y mejorada de este interface. Es la interfase con más futuro, si bien tiene problemas de compatibilidad entre las diferentes opciones de controladoras, discos duros, impresoras, unidades de *CD-ROM* y demás dispositivos que usan esta, interfase debido a la falta de un estándar verdaderamente sólido.

Las mejoras del *SCSI*-2 sobre el *SCSI* tradicional son el aumento de la velocidad a través del bus, desde 5 Mhz a 10 Mhz, duplicando de esta forma el caudal de datos. Además se aumenta el ancho del bus de 8 a 16 bits, doblando también el flujo de datos. Actualmente se ha logrado el ancho de 32 bits, consiguiendo velocidades teóricas de hasta 40 Mbytes / seg.

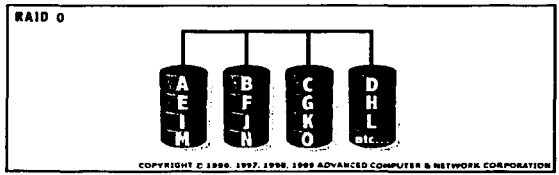
· **Fibra**: Estos discos cumplen con el estándar SCSI-3. A diferencia de los discos de SCSI-1, SCSI2, y Ultra SCSI, que usan interfaces paralelas, los discos de canal de fibra utilizan una interfaz serial, y pueden transferir información en ráfagas de hasta 100 MB por segundo. Estos discos, junto con los discos de 40 MB por segundo de SSA (Arquitectura de Almacenamiento en Serie) desarrollada por IBM, son principalmente para uso en servidores poderosos.

Uno de los parámetros, de fundamental importancia en los discos, es la velocidad de rotación de los discos medida por RPM. Enseguida se muestra una tabla en donde se clasifica este tipo de parámetro debido la velocidad que maneje el disco:

RPM	1 Vuelta cada	Latencia
3600	16,66 mseg.	8,33 mseg.
4500	13,33 mseg.	6,66 mseg.
5400	11,11 mseg.	5,55 mseg.
7200	8,33 mseg.	4,16 mseg.
10000	6,00 mseg.	3,00 mseg.

Tipos de RAID.

RAID 0: Este tipo de arreglo utiliza una técnica llamada "striping", la cual distribuye la información en bloques entre los diferentes discos. Es el único nivel de RAID que no duplica la información, por lo tanto no se desperdicia capacidad de almacenamiento. Se requieren mínimo dos discos.



RAID 0[3]

[3] Las imágenes de RAID 0, 1 3, 5,10 fueron obtenidas de la fuente de Advanced Computer Corporation.

Ventajas: RAID-0 permite acccesar más de un disco a la vez, logrando una tasa de transferencia más elevada y un rápido tiempo de acceso. Por no utilizar espacio en información redundante, el costo por Megabyte es menor.

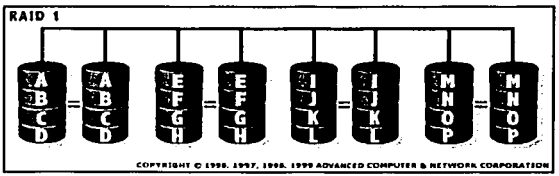
Desventaja: No existe protección de datos. No existe información en cuanto a Paridad.

Ambientes donde implementarlo: una buena alternativa en sistemas donde sea más importante el rendimiento que la seguridad de los datos. Es decir, en ambientes que puedan soportar una pérdida de tiempo de operación para poder reemplazar el disco que falle y reponer toda la información.

m= información duplicada

RAID 1: Este nivel de RAID usa un tipo de configuración conocido como "mirroring", ya que la información de un disco es completamente duplicada en otro disco. Así mismo, también se puede duplicar el controlador de disco (duplexing). Se desperdicia el 50% de la capacidad y sólo maneja dos discos.

RAID 1

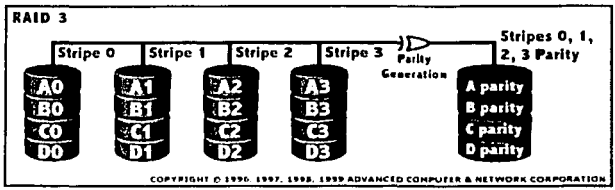


Ventajas: Se protege la información en caso de falla tanto del disco como del controlador (en caso de duplex), ya que si un disco suspende su operación, el otro continua disponible.

De este modo se evita la pérdida de información y las interrupciones del sistema debido a fallas de discos.

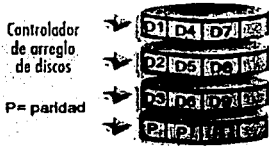
Desventajas: Gran consumo de necesidades hardware, 100% paridad y alto costo, pues es necesario el doble de discos.

Ambientes donde implementarlo: RAID-1 está diseñado para sistemas donde la disponibilidad de la información es esencial y su reemplazo resultaría difícil y costoso (más costoso que reponer el disco en sí). Típico en escrituras aleatorias pequeñas con tolerancia a fallas. El problema de este tipo de arreglos es el costo que implica duplicar los discos.



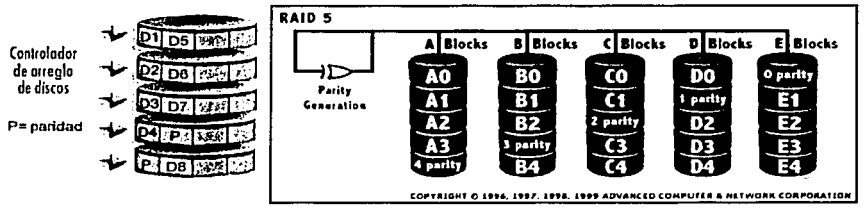
RAID 3: Conocido también como "striping con paridad dedicada", utiliza un disco de protección de información separado para almacenar información de control codificada. Esta información de control codificado o paridad proviene de los datos almacenados en los discos y permite la reconstrucción de la información en caso de falla. Se requieren mínimo tres discos y se utiliza la capacidad de un disco para la información de control.

RAID 3



Ventajas: RAID-3 proporciona una alta disponibilidad del arreglo, así como una tasa de transferencia elevada, mejorando de ese modo el rendimiento del sistema.
Desventajas: Un disco de paridad dedicado puede convertirse en un cuello de botella porque cada cambio en el grupo RAID requiere un cambio en la información de paridad. No plantea una solución al fallo simultáneo en dos discos. Está especialmente recomendado para aplicaciones que requieran archivos de datos de un gran tamaño (vídeo, imágenes, DataWare House).
Ambientes donde implementarlo: Es típico para transferencia larga de datos en forma serial , tal como aplicaciones de imágenes o video

RAID 5: Este nivel de RAID es conocido como "striping con paridad distribuida", ya que la información se reparte en bloques como RAID-0, pero un bloque de cada disco se dedica a la paridad. Es decir los datos codificados se añade como otro sector que rota por los discos igual que los datos ordinarios. Se requieren mínimo tres discos.

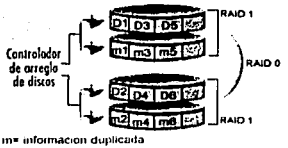


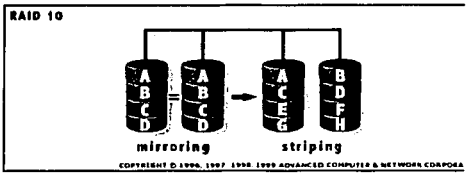
RAID 5

Ventajas: Es el esquema de protección de información más comúnmente usado, ya que proporciona un buen rendimiento general con una mínima pérdida de capacidad. Además el sistema tiene suficiente redundancia para ser tolerante a fallos.
Desventajas: Menores prestaciones que en RAID 1. No plantea una solución al fallo simultáneo en dos discos.

Ambientes donde implementarlo: Es recomendable para aplicaciones intensas de entrada/salida y de lectura/escritura, tal como procesamiento de transacciones.

RAID 10: Es un nivel de arreglo de discos, donde la información se distribuye en bloques como en RAID-0 adicionalmente, cada disco se duplica como RAID-1 , creando un segundo nivel de arreglo. Se conoce como "striping de arreglos duplicados". Se requieren, dos canales, dos discos para cada canal y se utiliza el 50% de la capacidad para información de control. También se le conoce como RAID 0&1





RAID 10

Ventajas: Este nivel ofrece un 100% de redundancia de la información y un soporte para grandes volúmenes de datos, donde el precio no es un factor importante.

Desventajas: Costo elevado, gran overhead y 100% de redundancia

Ambientes donde implementarlo: Ideal para sistemas de misión crítica donde se requiera mayor confiabilidad de la información, ya que pueden fallar dos discos inclusive (uno por cada canal) y los datos todavía se mantienen en línea. Es apropiado también en escrituras aleatorias pequeñas.

Como se ha visto, el tipo de RAID se basa dependiendo de la aplicación, desempeño y seguridad que se de implícitamente a la información manejada. En el caso del sistema de monitoreo nos basaremos en parámetros de escritura y lectura en Kr/s (Kilobytes lectura por segundo), Kw/s (Kilobytes de escritura por segundo) y el parámetro promedio de transacciones activas para ser atendidas en la cola de espera. Con los parámetros anteriores podremos saber perfectamente si el desempeño del disco es bueno o en su caso poder realizar modificaciones de RAID.

MEMORIA

En el caso de la memoria, hay diferentes variedades de memoria las cuales se pueden clasificar de la siguiente forma:

- **DRAM:** acrónimo de "Dynamic Random Access Memory", o simplemente RAM ya que es la original, y por tanto la más lenta.

Usada hasta la época del 386, su velocidad de refresco típica es de 80 ó 70 nanosegundos (ns), tiempo éste que tarda en vaciarse para poder dar entrada a la siguiente serie de datos. Por ello, la más rápida es la de 70 ns. Físicamente, aparece en forma de **DIMMs** o de **SIMMs**, siendo estos últimos de 30 contactos.

- **FPM (Fast Page Mode)**: a veces llamada DRAM, puesto que evoluciona directamente de ella, y se usa desde hace tanto que pocas veces se las diferencia. Algo más rápida, tanto por su estructura (el modo de " Página Rápida ") como por ser de 70 ó 60 ns. Es lo que se da en llamar la RAM normal o estándar. Usada hasta con los primeros Pentium, físicamente aparece como SIMMs de 30 ó 72 contactos (los de 72 en los Pentium y algunos 486).

Para acceder a este tipo de memoria se debe especificar la fila (página) y, seguidamente, la columna. Para accesos sucesivos de la misma fila, sólo es necesario especificar la columna, quedando la columna seleccionada desde el primer acceso. Esto hace que el tiempo de acceso en la misma fila (página) sea mucho más rápido. Era el tipo de memoria normal en las computadoras 386, 486 y los primeros Pentium y llegó a alcanzar velocidades de hasta 60 ns. Se presentaba en módulos SIMM de 30 contactos (16 bits) para los 386 y 486 y en módulos de 72 contactos (32 bits) para las últimas placas 486 y las placas para Pentium.

- **EDO o EDO-RAM:** Extended Data Output-RAM. Evoluciona de la FPM. Permite empezar a introducir nuevos datos mientras los anteriores están saliendo (haciendo su Output), lo que la hace algo más rápida (un 5%, más o menos). Mientras que la memoria tipo FPM sólo podía acceder a un solo byte (una instrucción o valor) de información de cada vez, la memoria EDO permite mover un bloque completo de memoria a la caché interna del procesador para un acceso más rápido por parte de éste. La estándar se

encontraba con refrescos de 70, 60 ó 50 ns. Se instala sobre todo en SIMMs de 72 contactos, aunque existe en forma de DIMMs de 168.

La ventaja de la memoria EDO es que mantiene los datos en la salida hasta el siguiente acceso a memoria. Esto permite al procesador ocuparse de otras tareas sin tener que atender a la lenta memoria. Esto es, el procesador selecciona la posición de memoria, realiza otras tareas y cuando vuelva a consultar la DRAM los datos en la salida seguirán siendo válidos. Se presenta en módulos SIMM de 72 contactos (32 bits) y módulos DIMM de 168 contactos (64 bits).

- **SDRAM:** Sincronic-RAM. Es un tipo síncrono de memoria, que, lógicamente, se sincroniza con el procesador, es decir, el procesador puede obtener información en cada ciclo de reloj, sin estados de espera, como en el caso de los tipos anteriores. Sólo se presenta en forma de DIMMs de 168 contactos; es la opción para las computadoras nuevas. SDRAM funciona de manera totalmente diferente a FPM o EDO. DRAM, FPM y EDO transmiten los datos mediante señales de control, en la memoria SDRAM el acceso a los datos esta sincronizado con una señal de reloj externa.

La memoria EDO está pensada para funcionar a una velocidad máxima de BUS de 66 Mhz, llegando a alcanzar 75MHz y 83 MHz. Sin embargo, la memoria SDRAM puede aceptar velocidades de BUS de hasta 100 MHz, lo que dice mucho a favor de su estabilidad y ha llegado a alcanzar velocidades de 10 ns. Se presenta en módulos DIMM de 168 contactos (64 bits). El ser una memoria de 64 bits, implica que no es necesario instalar los módulos por parejas de módulos de igual tamaño, velocidad y marca

- **PC-100 DRAM:** Este tipo de memoria, en principio con tecnología SDRAM, aunque también la habrá EDO. La especificación para esta memoria se basa sobre todo en el uso no sólo de chips de memoria de alta calidad, sino también en circuitos impresos de alta calidad de 6 o 8 capas, en vez de las habituales 4; en cuanto al circuito impreso, este debe cumplir las tolerancias mínimas de interferencia eléctrica; por último, los ciclos de memoria también deben cumplir altas especificaciones muy exigentes. De cara a evitar posibles confusiones, los módulos compatibles con este estándar deben estar identificados así: PC100-abc-def.

- **BEDO (burst Extended Data Output):** Fue diseñada originalmente para soportar mayores velocidades de BUS. Al igual que la memoria SDRAM, esta memoria es capaz de

transferir datos al procesador en cada ciclo de reloj, pero no de forma continuada, como la anterior, sino a ráfagas (bursts), reduciendo, aunque no suprimiendo totalmente, los tiempos de espera del procesador para escribir o leer datos de memoria.

- **RDRAM:** (Direct Rambus DRAM). Es un tipo de memoria de 64 bits que puede producir ráfagas de 2ns y puede alcanzar tasas de transferencia de 533 MHz, con picos de 1,6 GB/s. Es el componente ideal para las tarjetas gráficas AGP, evitando los cuellos de botella en la transferencia entre la tarjeta gráfica y la memoria de sistema durante el acceso directo a memoria (DIME) para el almacenamiento de texturas gráficas. Hoy en día la podemos encontrar en las consolas NINTENDO 64.
- **DDR SDRAM:** (Double Data Rate SDRAM o SDRAM-II). Funciona a velocidades de 83, 100 y 125 MHz, pudiendo doblar estas velocidades en la transferencia de datos a memoria. En un futuro, esta velocidad puede incluso llegar a triplicarse o cuadriplicarse, con lo que se adaptaría a los nuevos procesadores. Este tipo de memoria tiene la ventaja de ser una extensión de la memoria SDRAM, con lo que facilita su implementación por la mayoría de los fabricantes.
- **SLDRAM:** Funcionará a velocidades de 400 MHz, alcanzando en modo doble 800 MHz, con transferencias de 800 MB/s, llegando a alcanzar 1,6 GHz, 3,2 GHz en modo doble, y hasta 4 GB/s de transferencia. Se cree que puede ser la memoria a utilizar en los grandes servidores por la alta transferencia de datos.
- **ESDRAM:** Este tipo de memoria funciona a 133 MHz y alcanza transferencias de hasta 1,6 GB/s, pudiendo llegar a alcanzar en modo doble, con una velocidad de 150 MHz hasta 3,2 GB/s.

En el caso particular de la memoria, se monitoreará en función del parámetro de la partición llamada SWAP (Parte del disco que simula una cantidad de memoria adicional a la que se tiene físicamente) en un sistema UNIX, normalmente cuando la memoria física (ejemplos anteriores), ha sido ocupada en su totalidad, el sistema tiene la parte del disco llamada SWAP y realiza un intercambio de páginas de memoria física a el SWAP y viceversa, sabiendo la cantidad de SWAP que se está utilizando se puede llegar a estimar un aproximado de las necesidades adicionales sobre este recurso para el sistema.

RED

Hablar de los parámetros para monitoreo sobre la red, puede ser muy extenso, solamente nos centraremos a los requerimientos y necesidades de los puertos de red que estén en el servidor de servicios, aunque con esta restricción se puede tener también algunos puntos de opinión sobre la red en general.

Hay diferentes tecnologías ocupadas en comunicación de redes, sin embargo nos basaremos en el modelo TCP/IP, debido la extensa variedad que van desde el común Ethernet 10/100 pasando por ATM (Ya casi en desuso) hasta la IP encapsulado en cableado de Fibra. En el monitoreo nos basaremos en cinco parámetros ha estudiar los cuales serán:

- Porcentaje de Colisiones
- Número de Paquetes de entrada
- Número de Paquetes de Salida
- Errores Paquetes de Entrada
- Errores Paquetes de Salida

El tamaño de paquetes está dado por el parámetro de cada sistema llamado MTU (maximum Transmission Unit), el cual dependerá de la tecnología que se esté ocupando en el dispositivo de red.

Con los puntos anteriores resumimos que se tendrá un control completo sobre el monitoreo para saber cuales serán los posibles alcances del sistema en cuestión de servicio y disponibilidad.

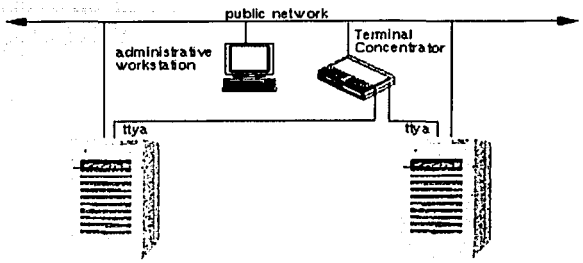
2.3 Análisis de requerimientos para ambientes de alta disponibilidad.

Para satisfacer las necesidades de un sistema de Alta Disponibilidad HA (HIGH AVALIBILYTY) se necesita llevar a la practica toda una metodología sobre los parámetros técnicos, procedimientos y necesidades ambientales para cada sistema en particular, dependiendo de sus funcionalidades, servicios y disponibilidad que se requiera, en general se tiene que contemplar los siguientes puntos para alcanzar un nivel básico en un modelo HA:

- Esquema funcional del sistema HA Cluster.
- Definir el % de disponibilidad.
- Definición de procedimientos de restauración.
- Definición de tiempos de servicio en hardware.
- Definición de tiempos de servicio en software.
- Definición de servicios en red.
- Tiempos de recuperación en agentes de HA.
- Políticas de monitoreo en ambiente HA.

Es bueno comentar que estos puntos son el inicio de una serie de parámetros para tratar de alcanzar el llamado DRP (Disaster Recovery Planning), que en el punto 2.6 del presente capítulo se tratará. Los puntos anteriores enseguida, los aplicaremos a la funcionalidad de un sistema de monitoreo en HA sobre el sistema operativo Solaris con este ejemplo se podrá concluir que se puede extender fácilmente a otro tipo de aplicación ya que se debe de cumplir en forma general los puntos ya nombrados.

El esquema funcional de un sistema HA se forma con un mínimo de dos nodos hasta 2+N donde N es desde 1 hasta 256 nodos, el crecimiento de nodos dependerá del proveedor. Enseguida se mostrará un esquema de dos nodos:



En el esquema funcional se describirán los componentes redundantes que debe tener la arquitectura del cluster como son, discos, tarjetas de red, tarjetas seriales, CPUs, Memoria, Fuentes de poder, conexiones de fibra, etc. Este es el primer panorama de un sistema en HA cada componente debe venir en el esquema por su MTBF (Mean Time Between Failures) el cual nos proporciona el promedio de horas que se tiene estimado para que ocurra un fallo por dispositivo a nivel de hardware.

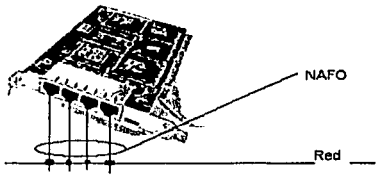
La disponibilidad del servicio se define como el porcentaje de tiempo en que el servido contratado está operativo, es decir, el servicio de información del cliente que está accesible (respuesta a una petición) según el mecanismo de monitorización establecido.

Y se calculará la disponibilidad mensualmente según la siguiente fórmula:
% = (Ttotal - Tparada)/ Total

Dónde *%* es la disponibilidad expresada en tanto por ciento; *Ttotal* es el período total de cálculo de disponibilidad (un mes); y *Tparada* es el tiempo durante el que el servicio contratado no se ha prestado. Los actuales clusters dan un rango de 99.98% a 99.99% de servicio disponible al cliente.

Por otro lado otros parámetros importantes sin menos preciar son los servicios de soporte en software y hardware en función de actualizaciones de sistema y servicios de mantenimiento sobre el mismo cluster, dichos servicios para el funcionamiento de un sistema de HA se darán normalmente sobre tiempos de 24x7x365 es decir disponibles en cualquier momento del año.

En el caso de la red, debe haber doble ruta de enlaces en servicios de rute adores, switches y en el caso del sistema de alta disponibilidad, se crearán grupos llamados NAFO (Network Address Fauli Over) estos grupos son normalmente trajetas Quad Fast 10/100 aunque no hay restricciones sobre otras tipo de tarjetas como son Gigabit Ethernet o ATM, según sea el medio de comunicación, dichos grupos dan servicio de red en los nodos del cluster y en caso de falla en algún nodo de red entra el respaldo dirigido por el monitoreo de agentes de alta disponibilidad, enseguida se muestra un esquema de un grupo NAFO:



Grupo NAFO de 4 puertos Fast Ethernet (Quad Fast), solo uno esta en servicio, y los tres restante estan en standby.

Por último en los sistemas de alta disponibilidad, se deben saber cuales son los tiempos de recuperación en caso de falla de los servicios en agentes de HA y lo que involucra la caída de estos, hay casos en los cuales debido a este tipo de fallas se tiene que construir totalmente un nuevo nodo o el cluster en su totalidad, esto se da principalmente en sistemas con base de datos debido a que no se realizó una sincronización adecuada en los nodos del cluster, sin embargo normalmente estos errores son debido a descuidos humanos en su poco entrenamiento sobre estos sistemas.

El sistema de monitoreo se centrará en tener una funcionalidad de alta disponibilidad en función a un esquema de grupo NAFO, en el capítulo III y IV se describirá a detalle sobre el funcionamiento de estos grupos en el sistema de monitoreo.

2.4 Protocolo para monitoreo de redes SNMP.

Protocolo SNMP

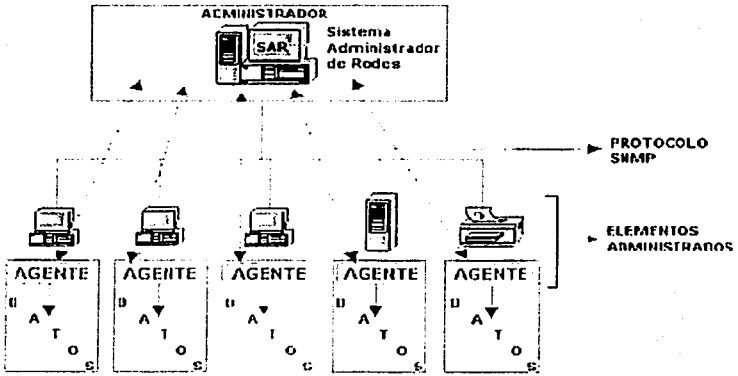
En la primera etapa de ARPANET se comprendió que cuando había problemas con la red, la única forma de identificar el problema, era ejecutando comandos muy simples como el ping, el cual no brinda suficiente información para resolver rápidamente dichos problemas. En el año de 1990 surge un nuevo estándar llamado: SNMP (Simple Network Managenement Protocol, Protocolo Simple de Administración de Redes), definido en el RFC 1157 aunque ya han salido RFC complementarios. Este protocolo muestra una manera de administrar y supervisar las redes de cómputo para identificar y resolver problemas, así como para planear su crecimiento. Se encuentra implementado en la capa de aplicación y pertenece al grupo de protocolos de TCP/IP.

Hasta el momento existen tres versiones del protocolo: SNMPv1 (versión 1), SNMPv2 (versión 2) y SNMPv3 (versión 3). Las tres son muy parecidas, sólo que SNMPv2 tiene algunas mejoras sobre la primera versión, y de la misma forma SNMPv3 tiene ciertas ventajas sobre la segunda versión.

Componentes de SNMP

Podemos decir que SNMP cuenta con 4 componentes principales: Sistema de Administración de Redes (SAR), elementos administrados, un agente SNMP y el protocolo SNMP.

Los elementos administrados son cualquier nodo de la red que contiene un agente SNMP, son elementos como: servidores, ruteadores, impresoras, etc., los cuales recopilan información administrable para el SAR, tal que es accesada por medio del protocolo SNMP. El agente SNMP es un software que reside en el elemento administrado, el cual toma la información de administración recopilada por este elemento y la traduce para que sea compatible con el SAR. Y por ultimo, SNMP es el protocolo por medio del cual el elemento administrado proporciona la información de administración al SAR.



2.5 ¿Cual es la importancia de un DRP?

Cual es la forma en que un director, gerente o dueño de toda empresa quisiera saber que todas sus actividades de negocio siempre estarán en funcionamiento, pase lo que pase, como son; administración, costos, logística de producción, etc. Aunque sabemos que en este mundo no podemos dar una seguridad de 100%, si podemos acercarnos lo más posible a este número. Para alcanzar esta seguridad trataremos lo que llamamos DRP (Disaster Recovery Planning), plan de recuperación de desastres.

EL DRP es toda una metodología, la cual consiste en realizar todo un plan de contingencia de desastres para el ambiente esencialmente productivo de una empresa, si bien esto esta ya manejado por algunas políticas de respaldo y sistemas de alta disponibilidad, estos puntos no llegan al porcentaje óptimo en caso de riesgos como temblores, inundaciones o índoles de descuido o poco conocimiento en el sentido humano.

El plan debe cumplir con las siguientes premisas:

- ✓ Alcance de disponibilidad de servicios.
- ✓ Modelo de respaldos y recuperación
- ✓ Escenarios de contingencia
- ✓ Servicios profesionales
- ✓ Infraestructura requerida antes y después del siniestro
- ✓ Centro de cómputo alterno
- ✓ Configuración de equipo de computo y telecomunicaciones
- ✓ Servicios logísticos asociados
- ✓ Plan de recuperación en caso de desastre
- ✓ Resguardo de medios de información en medios magnéticos
- ✓ Pruebas de recuperación en escenario de contingencia
- ✓ Documentación aprobada según pruebas de recuperación
- ✓ Planes de pruebas anuales en ambiente de contingencia.

Todos los puntos deben ser cumplidos, documentados y probados. Desafortunadamente este tipo de metodologías normalmente tienen un costo muy elevado, debido a los requerimientos en servicios de personal, software y hardware, lo que sólo pocas empresas son capaces de costear estos egresos, el nivel de disponibilidad está dado como mínimo en un 99.99998%

CAPÍTULO III SEGURIDAD DE SISTEMAS DE MONITOREO.

3.0 Seguridad en redes LAN y WAN.

Criptografía en redes LAN y WAN.

En los procesos de almacenamiento y transmisión de la información normalmente aparece el problema de la seguridad. En el almacenamiento, el peligro lo representa el robo del soporte del mensaje o simplemente el acceso no autorizado a esa información, mientras que en las transmisiones lo es la intervención del canal.

La protección de la información se lleva a cabo variando su forma. Se llama cifrado (o transformación criptográfica) a una transformación del texto original (llamado también texto inicial o texto claro) que lo convierte en el llamado texto cifrado o criptograma. Análogamente, se llama descifrado a la transformación que permite recuperar el texto original a partir del texto cifrado.

Cada una de estas transformaciones está determinada por un parámetro llamado clave. El conjunto de sus posibles valores se denomina espacio de claves K. La familia de transformaciones criptográficas se llama sistema criptográfico T={Tk/k K}.
Para cada transformación criptográfica Tk se definen las imágenes de cada una de las palabras de n letras. Es decir, el sistema criptográfico se puede describir como T={Tk^n:1<=n<i}., siendo Tk^n(x)=y, donde y es la palabra cifrada que corresponde a la palabra original x. En adelante se usará el termino n-palabra en lugar de palabra de n letras.
Para evitar ambigüedades, se hacen las siguientes suposiciones:
Tk^n es biyectiva.
Se usa el mismo alfabeto para ambos textos, original y cifrado.
Se define el cifrado de todas las posibles palabras, independientemente de si existen o no.
Cada n-palabra se cifra en una n-palabra, teniéndose así que el cifrado no cambia la longitud del texto original.

En general no es necesario imponer simultáneamente todas estas condiciones aunque en algunos casos, como el procesamiento de información digital, si es recomendable, porque:
Se trabaja únicamente con alfabeto binario.
Debe existir el cifrado de todas las palabras posibles.

Si el cifrado cambiara la longitud del texto, sería necesario usar un nuevo formato para el texto cifrado.

En particular, si el texto se alargara al cifrarse:

El programador tendría que prevenirlo reservando suficiente memoria. En una base de datos organizada por campos, la expansión de uno de ellos obligaría a reformatearlos todos.

La teoría de la información estudia el proceso de la transmisión ruidosa de un mensaje:

canal ruidoso

mensaje transmitido ----------> mensaje recibido

Por otro lado, la criptografía estudia el proceso de cifrado de un texto.

transformación criptográfica

texto original ----------> texto cifrado

En el primer caso, la distorsión del mensaje transmitido no es intencionada, sino que se debe al canal. Además, el receptor trata de recuperar el mensaje transmitido, usando para ello el mensaje recibido y una descripción probabilística del canal ruidoso. En el otro caso, la distorsión si es intencionada y el interceptor intenta obtener el texto original, usando para ello el criptograma y una descripción parcial del proceso de cifrado.

En cuanto a objetivos, en la teoría de la información se intenta transmitir el mensaje lo más claro posible, mientras que en criptografía se trata de lo contrario; es decir, hacer el

mensaje incompresible para el enemigo. Sin embargo, aunque opuestos, estos propósitos se pueden combinar para proteger los mensajes contra el enemigo y el ruido a la vez. Para ello, primero hay que cifrar el mensaje y luego hay que aplicar al criptograma resultante un código corrector de errores.

Shannon, padre de la teoría de la información, ya mencionó en 1949 la relación existente entre ésta y la criptografía. Desde entonces se usan su teoría y nomenclatura para el estudio teórico de la criptografía.

Como ya se ha dicho, la criptología representa una lucha entre el criptógrafo, que trata de mantener en secreto un mensaje usando para ello una familia de transformaciones, y el enemigo, que intenta recuperar el texto inicial. Como en toda lucha, es necesario establecer unas reglas.

Reglas de Kerckhoffs

Kerckhoffs (s. XIX), en su trabajo titulado "La criptografía militar", recomendó que los sistemas criptográficos cumpliesen las siguientes reglas, que efectivamente han sido adoptadas por gran parte de la comunidad criptográfica:

No debe existir ninguna forma de recuperar mediante el criptograma el texto inicial o la clave. Esta regla se considera cumplida siempre que la complejidad del proceso de recuperación del texto original sea suficiente para mantener la seguridad del sistema.

Todo sistema criptográfico debe estar compuesto por dos tipos distintos de información.

Pública, como es la familia de algoritmos que lo definen.

Privada, como es la clave que se usa en cada cifrado particular. En los sistemas de clave pública, parte de la clave es también información pública.

La forma de escoger la clave debe ser fácil de recordar y modificar.

Debe ser factible la comunicación del criptograma por los medios de transmisión habituales. La complejidad del proceso de recuperación del texto original debe corresponderse con el beneficio obtenido.

Tipos de ataques

Para el estudio de los sistemas criptográficos es conveniente conocer la la situación del enemigo. Se tienen los siguientes ataques posibles:

Ataque sólo con texto cifrado. Esta es la peor situación posible para el criptoanalista, ya que se presenta cuando sólo conoce el criptograma.
Ataque con texto original conocido. Consiste en que el criptoanalista tiene acceso a una correspondencia del texto inicial y cifrado. Se de este caso, por ejemplo, cuando conoce el tema del que trata el mensaje, pues eso proporciona una correspondencia entre las palabras más probables y las palabras cifradas mas repetidas.
Ataque con texto original escogido. Este caso se da cuando el enemigo puede obtener, además del criptograma que trata de descifrar, el cifrado de cualquier texto que él elija, entendiéndose que no es que él sepa cifrarlo, sino que lo obtiene ya cifrado.
Ataque con texto cifrado escogido. Se presenta cuando el enemigo puede obtener el texto original correspondiente a determinados textos cifrados de su elección.

Retomando la primera regla de Kerckhoffs, se pueden distinguir dos tipos de secreto: el secreto teórico o incondicional y el secreto práctico o computacional. El primero se basa en que la información disponible para el enemigo no es suficiente para romper el sistema. Por ejemplo, se da este caso cuando el enemigo sólo conoce una cantidad de criptograma insuficiente para el criptoanálisis. Por el contrario, el secreto práctico se mide de acuerdo con la complejidad computacional del criptoanálisis. Según las necesidades actuales, y debido principalmente a la gran cantidad de información que se transmite habitualmente, los diseñadores de sistemas criptográficos deben suponer que el enemigo puede hacer al menos un ataque del segundo tipo, luego deben intentar conseguir al menos secreto práctico.

Como ya se ha mencionado, existen muchos puntos en común entre la teoría de la información y la criptografía. En concreto, entre codificación y criptografía se tiene que la

codificación representa una forma alternativa de esconder un mensaje. La diferencia esencial entre un cifrado y un código estriba en que este último es un diccionario fijo; es decir, a cada palabra le corresponde siempre la misma palabra código. Las principales desventajas del código cuando se utiliza como cifrado son:

- Sólo se pueden transmitir aquellas palabras que tengan traducción asignada en el diccionario del código.
- El receptor debe tener el diccionario para poder decodificar; es decir, el código completo constituye la clave.
- Su implementación, sobre todo a la hora de cambiar el código, es muy costosa.
- El criptoanálisis se puede basar en un análisis de frecuencias.
- La ventaja de este sistema es la compresión de la información siempre que las palabras de códigos usadas sean más cortas que las palabras originales.

3.1 Niveles de seguridad en sistemas operativos basados en el Orange Book.

Para tratar los niveles de seguridad en los sistemas operativos se realizo en forma exhaustiva sobre pruebas hechas por la Defensa Nacional de los Estados Unidos en el ano 1983 publicado con el nombre de "Defense *Trusted Computer System Evaluation Criteria*" en el cual se manifiesta las cualidades y propiedades que deben cumplir los sistemas operativos para alcanzar los distintos niveles de seguridad los cuales se describen a continuación:

Hay 7 niveles de seguridad definido por el departamento de defensa de los Estados Unidos usados para la protección de hardware, software, y almacenamiento de información los cuales se han nombrado como D1, C1, C2, B1, B2, B3 y A1.
D1: Es el nivel bajo de seguridad (no tiene seguridad), solo es un sistema de trabajo sin autorizaciones.

C1: Es el nivel de seguridad sin autorización (no hay autenticación a nivel de usuario), el sistema tiene archivos y directorios con control de lectura y escritura, sin embargo, root es

considerado inseguro ya que el sistema de auditoria no esta disponible así como en la totalidad de usuarios no cuenta con un control de cambios en el sistema.

C2: Es el nivel de seguridad que tiene funciones de auditoria para archivos y para relativos eventos del sistema (login, servicios remotos, análisis de recursos) además da una fuerte protección para archivos esenciales del sistema, como ejemplos: /etc/passwd y /etc/shadow.

B1: Es el nivel de seguridad que soporta multi-niveles de seguridad, acceso de control restringido, se puede desarrollar la autentificación a través de encriptación (DES, SKIP), así como propiedad de archivos y directorios. Control de procesos únicamente por root o delegados.

B2: Es el nivel de seguridad que tiene implementado en cada objeto y archivo claves de seguridad con cifrado de algún algoritmo de nivel de seguridad, estas claves para cada objeto o archivo cambia dinámicamente dependiendo de los eventos del sistema.

B3: Es el nivel de seguridad extendido a los niveles de seguridad en acción directa a dispositivos físicos del Sistema Operativo que se este utilizando, es decir dentro del hardware, por ejemplo terminales que puedan solo conectarse a través de un único cable con una ruta especifica y sistemas de rastreó en dicha trayectoria.

A1: Es el nivel de seguridad mas alto que es validado a través de las políticas del Orange Book (Políticas y Parámetros de seguridad dados por la defensa de los Estados Unidos). El desarrollo de las políticas de seguridad deben ser verificado matemáticamente en el hardware y software. También se debe de implementar protección durante el envío de procesos y en prevenir formas de violación del sistema a niveles de seguridad en procesos de CPU, Disco, Terminales, Memoria RAM y secuencia de inicialización del sistema.

Los niveles de seguridad crean un fuerte impacto para la relación del desempeño de usuarios y de aplicaciones residentes en el servidor.

3.2 Algoritmos de autentificación DES, 3DES, RSA y MD5.

Sin duda el cifrado en bloque más conocido es el llamado DES. Este sistema se puede catalogar como un cifrado en bloque que es a la vez un cifrado producto de transposiciones y sustituciones.

A finales de los años cuarenta, Shannnon sugirió nuevas ideas para futuros sistemas de cifrado. Sus sugerencias se referían al uso de operaciones múltiples que mezclaran transposiciones y sustituciones. Estas ideas fueron aprovechadas por IBM en los años setenta, cuando desarrolló un nuevo sistema llamado LUCIFER. Poco después en 1976, el gobierno de EEUU adoptó como estándar un sistema de cifrado basado en el LUCIFER y denominado DES (Data Encryption Standard). En consecuencia casi todos los gobiernos del mundo aceptaron el mismo cifrado o partte de él como estándar en las comunicaciones de las redes bancarias y comerciales.

En el DES, el bloque de entrada M en primer lugar sufre una transposición bajo una permutación denominado IP, originando To=IP(M). Después de pasar To dieciséis veces por una función f, se transpone bajo la permutación inversa IP', obteniéndose así el resultado final.

Criptografía de clave pública
Propiedades de los algoritmos
Sistemas de clave pública
En los cifrados asimétricos o de clave pública la clave de descifrado no se puede calcular a partir de la de cifrado.

En 1975, dos ingenieros electrónicos de la Universidad de Stanford, Whitfield Diffie y Martin Hellman, sugieren usar problemas computacional mente irresolubles para el diseño de criptosistemas seguros. La idea consiste básicamente en encontrar un sistema de cifrado computacional mente fácil (o al menos no difícil), de tal manera que el descifrado sea, por el contrario, computacional mente irresoluble a menos que se conozca la clave.

Para ello, hay que usar una transformación criptográfica Tk de fácil aplicación, pero de tal forma que sea muy difícil hallar la transformación inversa Tk' sin la clave de descifrado. Dicha función Tk es, desde el punto de vista computacional, no invertible sin cierta información adicional (clave de descifrado) y se llama función de una vía o función trampa.

En estos esquemas se utiliza una clave de cifrado (clave pública) k que determina la función trampa Tk, y una clave de descifrado (clave secreta o privada) que permite el cálculo de la inversa Tk'.

Cualquier usuario puede cifrar usando la clave pública, pero sólo aquellos que conozcan la clave secreta pueden descifrar correctamente.
En consonancia con el espíritu de la criptografía moderna, y tal cómo sucedía en los sistemas simétricos, los algoritmos de cifrado y de descifrado son públicos, por lo que la seguridad del sistema se basa únicamente en la clave de descifrado.

Propiedades de los algoritmos de clave pública
Según Diffie y Hellman, todo algoritmo de clave pública debe cumplir las siguientes propiedades de complejidad computacional:
Cualquier usuario puede calcular sus propias claves pública y privada en tiempo polinomial.

El emisor puede cifrar su mensaje con la clave pública del receptor en tiempo polinomial.
El receptor puede descifrar el criptograma con la clave privada en en tiempo polinomial.
El criptoanalista que intente averiguar la clave privada mediante la pública se encontrará con un problema intratable.
El criptoanalista que intente descifrar un criptograma teniendo la clave pública se encontrará con un problema intratable.

En la práctica, el diseñador de algoritmos asimétricos se encuentra con cinco problemas numéricos distintos. Los tres primeros, correspondientes a las condiciones 1,2 y 3, deben pertenecer a la clase polinomial. Los otros dos, correspondientes a las condiciones 4 y 5,

son problemas complejos, preferiblemente NP-completos. Hay que señalar que las condiciones 4 y 5 no exigen sólo la simple pertenencia a la clase de los problemas NP-completos, ya que aunque un problema pertenezca a esta clase siempre puede darse algún ejemplo concreto que se resuelva en tiempo polinomial.

En líneas generales, un esquema a seguir para la construcción de un criptosistema de clave pública es el siguiente:

- Escoger un problema difícil P, a ser posible intratable.
- Escoger un subproblema de P fácil, Pfacil, que se resuelva en tiempo polinomial preferiblemente en tiempo lineal.
- Transformar el problema Pfacil de tal manera que el problema resultante Pdificil, no se parezca al inicial, pero sí al problema original P.
- Publicar el problema Pdificil y la forma en que debe ser usado, constituyendo este proceso la clave (pública) de cifrado. La información sobre cómo se puede recuperar el problema Pfacil a partir del problema Pdificil se mantiene en secreto y constituye la clave (secreta) de descifrado.

Los usuarios legítimos utilizan la clave secreta para llevar a cabo el descifrado convirtiendo el problema Pdificil en el problema Pfacil, mientras que, por el contrario, el criptoanalista debe enfrentarse forzosamente a la resolución del problema Pdificil.
Es más difícil diseñar un sistema de clave pública seguro contra un ataque con texto original escogido que un sistema de clave secreta seguro frente al mismo tipo de ataque.
En la construcción de criptosistemas se pueden observar diferencias entre los algoritmos para sistemas simétricos y los usados en clave publica. En primer lugar, existen mayores restricciones de diseño para un algoritmo asimétrico que para uno simétrico, debido a que la clave pública representa información adicional que potencialmente un enemigo puede usar para llevar a cabo el criptoanálisis. Normalmente, el algoritmo de clave pública basa su seguridad en la dificultad de resolver algún problema matemático conocido, mientras que algunos algoritmos simétricos, cómo el DES, se diseñan de tal manera que las ecuaciones matemáticas que los describen son tan complejas que no son resolubles analíticamente.

En segundo lugar, existen grandes diferencias en la generación de claves. En los algoritmos simétricos, en los que el conocimiento de la clave de cifrado es equivalente al de la de descifrado, y viceversa, la clave se puede seleccionar de forma aleatoria. Sin embargo, en los algoritmos asimétricos, cómo la relación entre clave de cifrado y de descifrado no es pública, se necesita un procedimiento para calcular la pública a partir de la clave privada que sea computacional mente eficiente y tal que el cálculo inverso sea imposible de realizar.

Hace algunos años, este tipo de sistemas no parecía tener ninguna ventaja en el mundo criptográfico, porque tradicionalmente la criptografía se usaba sólo con propósitos militares y diplomáticos, y en estos casos el grupo de usuarios es lo suficientemente pequeño cómo para compartir un sistema de claves. Sin embargo, en la actualidad, las aplicaciones de la criptografía han aumentado progresivamente, hasta alcanzar muchas otras áreas donde los sistemas de comunicación tienen un papel vital. Cada vez con mayor frecuencia se pueden encontrar grandes redes de usuarios en las que es necesario que dos cualesquiera sean capaces de mantener secretas sus comunicaciones entre sí. En estos casos, el intercambio continuo de claves no es una solución muy eficiente.

Por otro lado, hay que resaltar la ventaja que representa en los sistemas asimétricos la posibilidad de iniciar comunicaciones secretas sin haber tenido ningún contacto previo.

Sistemas de clave pública más transcendentes

A continuación, nombramos algunos de los sistemas de clave pública que han tenido más trascendencia.

Sistema RSA. Se basa en el hecho de que no existe una forma eficiente de factorizar números que sean productos de dos grandes primos.

Sistema de Rabin. Se basa también en la factorizacion.

Sistema de ElGamal. Se basa en el problema del logaritmo discreto.

Sistema de Merkle-Hellman. Esta basado en el problema de la mochila.

Sistema de McEliece. Se basa en la teoría de la codificación algebraica, utilizando el hecho de que la decodificación de un código lineal general es un problema NP-completo.

Sistemas basados en curvas elípticas. En 1985, la teoría de las curvas elípticas encontró de la mano de Miller aplicación en la criptografía. La razón fundamental que lo motivó fue que las curvas elípticas definidas sobre cuerpos finitos proporcionan grupos finitos abelianos, donde los cálculos se efectúan con la eficiencia que requiere un criptosistema, y donde el cálculo de logaritmos es aún más difícil que en los cuerpos finitos. Además, existe mayor facilidad para escoger una curva elíptica que para encontrar un cuerpo finito, lo que da una ventaja más frente a su predecesor, el sistema de ElGamal.

Sistema probabilístico. Aunque la criptografía de clave pública resuelve el importante problema de la distribución de claves que se presenta en la criptografía de clave secreta; en clave pública se presenta otro problema, el texto cifrado C=Ek (M) siempre deja escapar alguna información sobre el texto original porque el criptoanalista puede calcular por sí mismo la función de cifrado con la clave pública sobre cualquier texto que quiera. Dado cualquier M' de su elección, puede fácilmente descubrir si el mensaje original M=M', pues esto se cumple si, y sólo si Ek(M')=C. Incluso aunque recuperar M a partir de C fuera efectivamente infactible, no sabemos cómo medir la información que deja escapar sobre M. El propósito de la criptografía probabilística (noción ideada por Golwaser y Micali) es cifrar mensajes de manera que no exista cálculo factible que pueda producir información en lo que respecta al texto original correspondiente (salvo con una probabilidad ínfima). Hay que decir que estos sistemas no ofrecen verdadero secreto perfecto, son totalmente inseguros contra criptoanalistas con poder de cálculo ilimitado.
La principal diferencia técnica entre el cifrado probabilístico y los criptosistemas de clave pública es que los algoritmos de cifrado son probabilísticos en lugar de determinísticos: el mismo mensaje original puede dar lugar a un gran número de criptogramas distintos. En consecuencia, un criptoanalista que tenga un candidato para el texto original no podría verificar su suposición cifrándolo y comparando el resultado con el criptograma interceptado.

Sistema RSA

Tipos de ataques al RSA.

La seguridad del RSA se basa en el hecho de que no existe una forma eficiente de factorizar números que sean productos de dos grandes primos.
Fue desarrollado en 1977 por Ronald Rivest, Adi Shamir y Leonard Adleman, de ahí el nombre de RSA, que corresponde a las iniciales de los apellidos de sus autores.

Sistema RSA

En la siguiente descripción del algoritmo señalamos entre paréntesis que partes del sistema se consideran publicas y cuales secretas.
Encontrar dos grandes números primos, p y q (secretos), y calcular el número n (publico) mediante su producto, n=p*q.
Encontrar la clave de descifrado constituida por un gran número entero impar, d (secreto), que es primo con el número F(n) (secreto), obtenido mediante F(n)=(p-1)*(q-1). Siendo F(n) la función de Euler.
Calcular el entero e (publico) tal que 1 ó e ó F(n), mediante la formula: e*d-1(mod F(n)).
Hacer publica la clave de cifrado (e,n).

Para cifrar un texto, es necesario previamente codificar el texto en un sistema numérico, bien decimal o bien binario, y dividir en bloques Mi de tamaño j o j-1 de forma que, según sea el alfabeto usado el decimal o el binario cumpla en cada caso: 10^(j-1) < n < 10^j o 2^(j-1) < n < 2^j. Cuando se toma como tamaño j, el descifrado del texto puede no ser único, por tanto esta elección se hace solo cuando la unicidad del descifrado no es importante.
Cifrar cada bloque Mi transformándolo en un nuevo bloque de números Ci de acuerdo con la expresión: Ci-Mi^e(mod n).

Para descifrar el bloque Ci, se usa la clave privada d según la expresión: Mi-Ci^d(mod n).

Analicemos la base numérica que hace que si M se cifra en C entonces C se descifra en M. Con las claves publica y privada (e,n) y d descritas, dado cualquier mensaje original M representado por un entero entre 0 y n-1 se tiene que, efectivamente, si C-M^e(mod n), entonces C^d-M(mod n).

Si se considera el mensaje descifrado D-C^d(mod n) como C-M^e(mod n), se tiene que D-(M^e+kn)^d(mod n), siendo k algún entero no negativo. Si se desarrolla el binomio, se obtiene que D-M^(e*d)(mod n), pero dado que e*d-1(mod F(n)), se tiene que D-M^[t*(p-1)*(q-1)+1](mod n) para algún entero t no negativo. Como p es primo y p-1-0(mod F(p)), la identidad de Euler-Fermat confirma que M^(p-1)-1(mod p), luego existe algún entero r tal que M^(p-1)=r*p+1 y por tanto M^[t*(p-1)*(q-1)+1]=[(r*p+1)^(t*(q-1))]*M- M(mod p). De la misma forma, se llega a que M^[t*(p-1)*(q-1)+1]-M(mod q). A partir de ambas ecuaciones congruenciales y gracias al corolario de la identidad de Euler-Fermat que afirma que: "Para dos primos distintos cualesquiera p y q y cualquier par de enteros positivos x y u, si x^u-x(mod p) y x^u-x(mod q), entonces x^u-x(mod p*q)", se llega finalmente a la identidad M^[t*(p-1)*(q-1)+1]-M(mod p*q).

Los procesos de cifrado y descifrado pueden ser implementados mediante algunos algoritmos conocidos.

Las dos principales dificultades en la implementación del RSA son:

Potencias modulares.

Búsqueda de números primos.

Para aumentar las dificultades, la seguridad del RSA estriba en que los primos p y q elegidos han de ser muy grandes. Para encontrar los grandes primos p y q se pueden utilizar varios algoritmos, como el de Solovay-Strassen, y el de Lehman y Peralta, que sirven para comprobar la primalidad.

En el caso del RSA puede encontrarse el entero d, primo con F(n)=(p-1)*(q-1), tomando simplemente un número primo mayor que max{p,q}.

Para calcular el entero e tal que e*d*-1(mod F(n)), se puede utiliza el algoritmo euclideo por ser d primo con F(n).

Por ultimo, las operaciones de cifrado y descifrado requieren el calculo de potencias modulares, lo que se puede llevar a cabo en tiempo polinomial. Exactamente son necesarias 2*(log2 (n)) multiplicaciones modulares, de manera que, por ejemplo, para el calculo de 2^18=(((22) 2) 2) 2 * (22) son necesarias 5 multiplicaciones modulares.

Tipos de ataques al RSA

Para analizar la seguridad del sistema, se supone que el criptoanalista tiene una cantidad ilimitada de pares (M,C) de mensajes originales y sus correspondientes criptogramas. Las posibles maneras que tiene de atacar el sistema son las siguientes:

Factorizar n.

Calcular F(n).

Ataque por iteración.

Ataque de Blakley y Borosh.

Vamos a analizar estos posibles ataques:

1.- Factorizar n:

De esta forma obtiene el número F(n)=(p-1)*(q-1), y con el la clave privada d, puesto que e es publica y se cumple: e*d-1(mod F(n)).

Al ser n el producto de solo dos números primos, un algoritmo de factorización requiere como máximo 'n pasos, pues uno de los dos factores es necesariamente un primo menor que 'n. Sin embargo, si n fuera el producto de N>2 primos, un algoritmo de factorización necesitaría como máximo n^(1/N) pasos, que es una cota menor que 'n, por lo que se concluye que es adecuada la obtención de n como producto de solo dos números primos.

Con respecto al estudio del problema de la factorización, hay que mencionar al precursor de la moderna factorización, el algoritmo de fracciones continuas de Morrison-Brillhart, ya que es uno de los más rápidos. Sin embargo, los dos algoritmos de factorización que resultan más prácticos para grandes enteros corresponden al de factorización con curvas elípticas de Hendrik Lenstra y al de factorización con filtro cuadrático de Carl Pomerance.

Ambos algoritmos convierten el problema de la factorización de un entero n en el problema de encontrar soluciones no triviales ('x <> y(mod n)' y 'x <> -y(mod n)') de la ecuación: x3-y3 (mod n). Si se supone que ni (x+y) ni (x-y) son múltiplos de n enteros, se deduce que el m.c.d.(x+y,n) o bien el m.c.d.(x-y,n) es con seguridad un factor no trivial de n, por lo que se resuelve el problema de la factorización.

Por ejemplo, si n=97343, entonces la ecuación x2-y2 (mod 97343) es fácilmente resoluble por ser los factores de n dos números primos muy cercanos. Como 3122-1(mod 97343) y ni 313 ni 311 son múltiplos de 97343, se concluye que 313 y 311 son los factores de n.

2.- Calcular F(n):

Como se ve a continuación, esta manera es equivalente a la anterior. Si se tiene F(n), dado que p+q=n-F(n)+1 y a partir de la suma se puede calcular (p-q) 2 por coincidir con (p-q) 2-4*n, luego se consigue la factorización mediante las formulas q=[(p+q)-(p-q)]/2 y p=[(p+q)+(p-q)]/2.

3.- Ataque por iteración:

Si un enemigo conoce (n,e,C), entonces puede generar la secuencia: C1-C^e(mod n), ..., Ci-[C(i-1)]^e(mod n), con lo que si existe algún Cj tal que C=Cj se deduce que el mensaje buscado es M=C(j-1) pues [C(j-1)]^e=Cj=C. Ahora bien, en cuanto la igualdad Cj=C se cumple solo para un valor de j demasiado grande, este ataque se vuelve impracticable. Con respecto a esto, Rivest demostró que si los enteros p-1 y q-1 contienen factores primos grandes, la probabilidad de éxito mediante este procedimiento es casi nula para grandes valores de n.

4.- Ataque de Blakley y Borosh:

El sistema RSA, además, tiene una característica muy peculiar, advertida por Blakley y Borosh, y es que no siempre esconde el mensaje. A continuación vemos un ejemplo que lo muestra.

Si e=17, n=35 y los mensajes a cifrar son M1=6 y M2=7, entonces se obtiene que 6^17-6(mod 35) y 7^17-7(mod 35). Una situación más peligrosa para el sistema aparece, por

ejemplo, con los valores p=97 q=109 y e=865, ya que el criptosistema resultante no esconde ningún mensaje, pues M^865-M(mod 97*109)
En general, lo ocurrido en el ultimo ejemplo ocurre siempre que e-1 es múltiplo de p-1 y q-1, pues en ese caso M^e-M(mod p*q). Además, se tiene que para cualquier elección de n=p*q siempre existen al menos 9 mensajes M que no se cifran en realidad, ya que verifican la ecuación M^e-M(mod n). De esos 9 mensajes hay tres fijos, que son M perteneciente a {0,1,-1}. Para hacer que el sistema RSA sea resistente contra ataques basados en este hecho, es conveniente elegir como claves privadas números primos de la forma p=2*p'+1, donde p' es un primo impar.

Por ultimo trataremos el cifrado a través de MD5 que implica una función HASH descripta a continuación:

Función HASH

Un valor hash se genera por una función H de la forma:
H = H(M) Donde M es un mensaje de longitud variable, H(M) es el valor hash de longitud fija. El valor hash se añade al mensaje en la fuente, cuando el mensaje se conoce, y es correcto. El receptor autentica ese mensaje recalculando el valor hash. Dado que la función hash, por si misma no es secreta, se necesita algún mecanismo para proteger el valor hash.
En primer lugar, examinaremos los requerimientos de una función hash, para que pueda ser usada en autenticación. Dado que las funciones hash son, normalmente, bastante complejas, es útil examinar al principio algunas funciones hash simples, para observar los aspectos relacionados. Posteriormente veremos algunas aproximaciones para el diseño de funciones hash.
La longitud del valor que genera la función influye en la seguridad. 64 bits son insuficientes para resistir determinados ataques, por lo que se suele optar por tamaños mayores, usualmente 128 bits. Existe un método para obtener un valor aleatorio de longitud mayor, a partir de otro más pequeño:

- Generar un valor aleatorio usando una función unidireccional.

- Añadir el valor generado al mensaje inicial.
- Generar un nuevo valor aleatorio, a partir de la concatenación anterior.
- Unir los valores generados en el paso 1) y 3).
- Repetir los pasos anteriores, hasta lograr un valor de longitud suficiente

El diseño de este tipo de funciones no es sencillo. Debe aceptar una entrada de longitud arbitraria, y obtener una salida de longitud fija. Básicamente, esto indica un proceso de compresión. Para ello se diseña un función cuya entrada será un bloque de texto y la salida del bloque de texto previo; de esta forma, la salida en un instante determinado hi, será el resultado de todo el bloque de texto ya procesado:

$hi = f(M, hi\text{-}1)$

El proceso anterior se repetiría con el siguiente bloque de entrada, para obtener el siguiente valor de la función. El valor final que devuelve la función es justamente, el valor de la última operación.

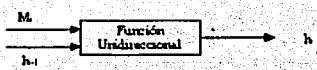


Figura 1 Función unidireccional.

Un checksum criptográfico, también conocido como un Código de Autenticación de Mensaje (MAC), se genera por una función C de la forma:

$MAC = Ck(M)$.

Donde M es un mensaje de longitud variable, K es la clave secreta compartida únicamente por el emisor y el receptor, y CK(M) es el autenticador de longitud fija. El MAC se añade al mensaje en el emisor, cuando se sabe que el mensaje es correcto. El receptor autentica el mensaje recalculando el MAC.

REQUERIMIENTOS PARA UNA FUNCION HASH

El propósito de una función hash es construir una "huella dactilar" de un fichero, un mensaje, o de un bloque de datos. Para que sea útil para la autenticación, la función hash H debe poseer las siguientes propiedades:

1. H debe poder aplicarse a un bloque de datos de cualquier tamaño.
2. H debe producir una salida de longitud fija.
3. H(x) debe ser fácil de calcular para cualquier x dado, de forma que tanto la implementación hardware como la de software sean prácticas.
4. Para cualquier código m, debe ser imposible computacional mente encontrar un valor x, tal que H(x)=m
5. Para cualquier bloque x, debe ser imposible computacional mente encontrar y ® x, con H(y)=H(x)
6. Debe ser imposible computacional mente encontrar un par (x , y), tal que H(x)=H(y)

Las tres primeras características se refieren a la aplicación práctica de la función hash para autenticación. La cuarta propiedad es la propiedad "unidireccional". Es fácil generar un código a partir de un mensaje, pero es virtualmente imposible obtener el mensaje a partir del código. Esta propiedad es muy importante si la técnica de autenticación implica el uso de un valor secreto. El valor secreto no se envía; sin embargo, si la función hash no fuera unidireccional, un atacante podría fácilmente descubrir el valor secreto: Si el atacante puede observar o interceptar una transmisión, el atacante obtiene el mensaje M y el código hash C = H (SAB || M). El atacante puede invertir la función hash para obtener SAB || M = H-1(C). Como ahora el atacante tiene M y SAB || M, puede recuperar SAB fácilmente.

La quinta propiedad garantiza que, dado un mensaje, no se puede encontrar otro que genere el mismo código hash. Esto previene la falsificación cuando se utiliza un código hash cifrado. En estos casos, el oponente puede leer el mensaje y por tanto generar el código hash, Pero, dado que el oponente no tiene la clave secreta, no podrá alterar el mensaje sin ser detectado. Si esta propiedad no se cumpliese, el atacante podría realizar los siguientes pasos: Primero, observa o intercepta un mensaje más su código hash cifrado; segundo,

genera un código hash no cifrado a partir del mensaje; tercero, genera un mensaje alternativo con el mismo código hash.

Una función hash que satisfaga las cinco primeras propiedades, se dice que es una función hash débil. Si además satisface la sexta propiedad, entonces se dice que es una función hash fuerte. La sexta propiedad protege frente a una clase de ataque sofisticado, conocido como el ataque del cumpleaños, que veremos posteriormente.

CLASIFICACION DE FUNCIONES HASH

Las funciones resumen se pueden clasificar en dos grandes grupos:

1. Funciones resumen con clave. Las funciones tienen como uno de sus argumentos la clave del sistema de cifrado utilizado para transmitir la información. El paradigma de esta clase de funciones resumen es el MAC (message Authentic Code).
2. Funciones resumen sin clave. Estas funciones tienen como argumento tan sólo el mensaje a compactar, al cual le aplican técnicas de compresión. El resumen producido por estas funciones se denomina MDC (Manipulation Detection Code). Los códigos MDC pueden ser igualmente utilizados para garantizar la integridad de los mensajes firmados digitalmente. este grupo se puede dividir a su vez en dos subgrupos:

- Las Funciones OWHF (One Way Hash Functions). En estas se busca una entrada cuyo valor hash, sea difícil de calcular contra un valor hash predeterminado. A estas funciones se les denomina también débiles, ya que solamente satisfacen tres de las cuatro condiciones necesarias de las funciones hash.
- Las Funciones CRHF (Collision Resistant Hash Functions). En estas, encontrar dos valores que tengan el mismo valor hash es difícil. Se les llama también fuertes, ya que aparte de cumplir las tres condiciones que cumplen las débiles, también cumple la cuarta es decir, es libre de colisiones.

Función MD5

Tras algunos procesos iniciales, MD5 procesa el texto de entrada en bloques de 512 bits, divididos en 16 sub-bloques de32 bits. La salida del algoritmo es un conjunto de cuatro bloques de 32 bits, que se concatenan para formar el valor hash de 128 bits.
En primer lugar, el mensaje se rellena de forma que su longitud 64 bits menos que un múltiplo entero de 512. Para ello, se añade una cadena de bits iniciada por un 1, y el resto 0s. Después, se representa la longitud del mensaje con un valor binario de 64 bits, y se añaden al final del bloque. Estos dos pasos permiten que la longitud del mensaje sea múltiplo entero de 512, y aseguran que dos mensajes diferentes no aparezcan iguales después del relleno. Se inician cuatro variables de 32 bits, denominadas variables de encadenamiento, a los valores:

$$A = 0x01234567$$
$$B = 0x89abcdef$$
$$C = 0xfedcba98$$
$$D = 0x76543210$$

Después comienza el ciclo principal del algoritmo. Este ciclo continua para cada uno de los bloques de 512 bits del mensaje. Las cuatro variables se copian en otras cuatro variables diferentes, a, b, c, y d. El ciclo principal del algoritmo tiene cuatro rondas (MD4 tiene sólo tres rondas), todas muy parecidas. Cada ronda usa una operación diferente 16 veces. Cada operación realiza una función no lineal en tres de las cuatro variables a, b, c, d. Entonces, añade el resultado a la cuarta variable, un subloque del texto y una constante. Entonces rota el resultado a la derecha un número variable de bits y añade el resultado a una de las cuatro variables. Por último, el resultado reemplaza a una de las variables. Esto se observa en las Figura 7 y Figura 8.

Hay cuatro funciones no lineales, que se utilizan en cada operación (una cada vez).

$$F(x,Y,Z) = (X \wedge Y) \vee ((\neg X) \wedge Z)$$
$$G(X,Y,Z) = (X \wedge Z) \vee (Y(\neg Z))$$
$$H(X,Y,Z) = X \oplus Y \oplus Z$$
$$I(X,Y,Z) = Y \oplus (X \vee (\neg Z))$$

Estas funciones están diseñadas de forma que si los bits correspondientes de X, Y y Z son independientes y no relacionados, entonces también lo serán los bits del resultado. La función F es la condicional a nivel de bits: Si X entonces Y sino Z. La función H es el operador de paridad a nivel de bit.

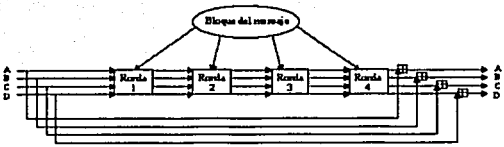


Figura 7. Ciclo principal de MD5.

Si Mj representa el j-ésimo bloque del mensaje (de 0 a 15), y <<<s representa un desplazamiento circular de s bits, las cuatro operaciones son:

$$FF(a,b,c,d,M_j,s,t_i) \text{ denota } a = b + ((a + F(b,c,d) + M_j + t_i) <<< s)$$
$$GG(a,b,c,d,M_j,s,t_i) \text{ denota } a = b + ((a + G(b,c,d) + M_j + t_i) <<< s)$$
$$HH(a,b,c,d,M_j,s,t_i) \text{ denota } a = b + ((a + H(b,c,d) + M_j + t_i) <<< s)$$
$$II(a,b,c,d,M_j,s,t_i) \text{ denota } a = b + ((a + I(b,c,d) + M_j + t_i) <<< s)$$

Las constantes ti se eligen como sigue:
En el paso i, ti es la parte entera de 232 D abs(sen(i)), con i en radianes.
Después de todo esto, a, b, c, y d se añaden a A, B, C, y D, respectivamente, y el algoritmo continua con el siguiente bloque de texto. La salida final es la concatenación de A, B, C, y D.

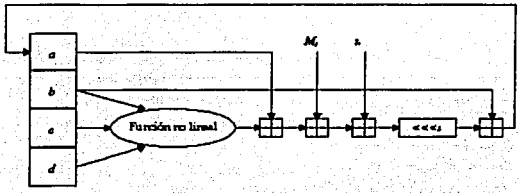


Figura 8. Una operación de MD5.

3.3 Seguridad en el sistema operativo UNIX.

Hablado de niveles de seguridad, en los sistemas UNIX, solo algunos llegan a tener un nivel muy restringido para llegar a alcanzar un grado de B1 o superior, normalmente los sistemas comerciales manejan niveles como tope máximo C2 ya que para subir se necesita agregar software de auditoria y de restricción mas rigurosa en seguridad de usuarios así como de dispositivos físicos, en la actualidad solo hay uno que se encuentra en forma comercial llegando al nivel B1 por lo cual nos centraremos en este, el cual es el Trusted Solares versión 8 y 9.

Niveles de seguridad en Solaris.

El sistema operativo Solaris se clasifica dependiendo del nivel de seguridad de la versión que se implemente, en la actualidad se tienen las siguientes versiones: 2.6, 2.7 ,8 y 9.

Versión de Solares	Modo de Aplicaciones	Nivel de Seguridad
2.6	32 bits, parcialmente 64bits	C2
2.7	32, 64 bits	C2
8, 9	32, 64 bits	C2
Trusted 8, 9	64 bits	B1

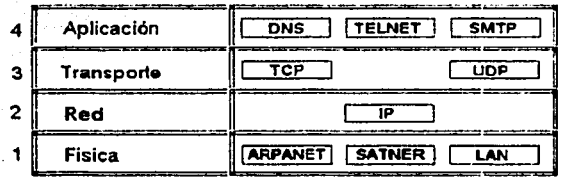
Herramientas para seguridad dentro de Solaris 2.7, 8, y 9

- EEPROM: seguridad a nivel de hardware.
- Permisos dados a través de herramientas normales para dar permisos como: chmod, chown, chgrp, umask, y setuid.
- ASET (Automated Security Enhancement Tool): es un programa para monitorear o restringe acceso a archivos y directorios. Checa ambientes específicos de usuarios como variables, eeprom, firewall IP packets. Este ayuda a dar un reporte en diferentes archivos rpt para tomas de decisión.
- Tipos de shells no restringidos para usuarios: sh, csh, ksh, zsh, bash.
- Tipos de shells restringidos para usuarios: rsh, rksh, keysh
- NIS (Network Information Services): utiliza la encriptación de DES por default, punto de falla role de root.
- NIS+ (Network Information Services Plus): utiliza DES, se pueden hacer grupos de trabajo-dominios, mas seguro debido a que las tablas de NIS no se implementa el role de root.
- Ipsec (IP Security Arquitecture): Es una autentificación a nivel de IP-layer, cubre una IP de autentificación Header, IP de encapsulamiento y una Llave maestra para cliente/servidor.
- SEAM (Sun Enterprise Authnetication Mechanism (Kerberos): Para la autentificación de procesos de login, privacidad de datos e integridad de estos. Se basa en Kerberos V5 compatible con Windows 2000.
- PAM (Pluggable Authentication Module): interfase para colocar una API que pueda ser usada por aplicaciones de terceros así como por el propio Sistema Operativo.
- RBAC (Role-Based Access Control): es una aplicación para desempeñar roles específicos esta propiedad es exclusiva de un nivel B1 de seguridad, ya que es implementado a niveles de puertos, sockets, file, log, impresoras, y los normales roles del sistema de usuarios.
- SKIP (Simple Key management for Internet Protocols): modo de encriptación de trafico en IP a nivel de stream. soporta Wndows 95, Windows 98 y Windows NT. Llave 128 y 512 bytes.

- Firewall SunScreen 3.1 (Autentificación de usuarios, NAT, modo de routeo o sin IP stealth.
- ACL (Access Control List): Creación de listas de usuarios y sus asociados derechos de acceso dando un control más estricto de cada archivo o directorio. Este servicio da la facilidad de ocultar permisos en derechos normales, es decir crear control de seguridad por grupos relacionados a un usuario.
- AUDIT: Procesos de auditoria a través de eventos asociados a cada usuario, se dan procesos de revisión de lectura, escritura, borrar archivos, comandos de atributos, eventos de red, file systems. Se puede implementar niveles de seguridad de usuarios y directorios específicos. La auditoria se ayuda a través de dos métodos: sistema de log, y por medio de CAPP (Controlled Access Protection Profile).

3.4 Seguridad en los protocolos TCP/IP y SNMP.

En el caso de estos dos protocolos en sus inicios fueron muy pobres los recursos para el control de seguridad en la actualidad el protocolo TCP/IP es inseguro por naturales debido a la cantidad de servicios que soporta por lo cual es necesario tener software y hardware adicional para dar soporte a este protocolo, esto dependerá del tipo que se quiera en las diferentes capaz que lo componen:



Capas del Modelo de Referencia TCP/IP

Capa Física. Esta capa se encarga de transmitir bits por un canal de comunicación y entre sus funciones se encuentran: definir las características físicas y eléctricas, manejar los voltajes y pulsos eléctricos, además de especificar cables, conectores y componentes de interfaz con el medio de transmisión.

Capa de Red. Esta capa es la encargada de que cualquier nodo pueda transmitir paquetes en cualquier red y que estos lleguen a su destino. Estos paquetes pueden tomar caminos diferentes a su destino y llegar de manera desordenada pero las capas superiores se encargan de reordenarlos. Esta capa se encarga también del ruteo de los paquetes y de evitar la congestión. Define un formato de paquete y un protocolo llamado IP (Protocolo de Internet).

Capa de Transporte. En esta capa es donde se lleva a cabo una conexión entre dos nodos y se definen dos protocolos TCP y UDP, los cuales se encargarán de transportar los datos. Podemos decir que dentro de esta capa y sobre los protocolos TCP y UDP se encuentran los SOCKETS (que son creados para comunicar procesos).

Capa de Aplicación. El Modelo TCP/IP no cuenta con las capas de sesión ni de presentación, ya que no se pensaron necesarias. Así que sobre la capa de transporte se encuentra la de aplicación, que es donde se incluyen protocolos destinados a proporcionar servicios, tales como correo (SMTP), transferencia de ficheros (FTP), conexión remota (TELNET) y otros más recientes como el protocolo HTTP y el servidor de nombres de dominio (DNS).

En el caso del protocolo SNMP proporciona mecanismos para el acceso a un almacén de información jerárquica compuesta por un conjunto de variables. Se distinguen dos tipos distintos de acceso a dicha información: un acceso para lectura que permite consultar los valores asociados a cada una de las variables y un acceso para escritura que permite modificar dichos valores.

Los mensajes de la primera versión del protocolo incluyen una cadena de caracteres denominada nombre de comunidad que se utiliza como un sencillo mecanismo de control de acceso a la información. Los agentes que implementan dicha versión del protocolo disponen generalmente de dos comunidades, o conjuntos de variables (no necesariamente disjuntos), identificadas por un nombre de comunidad configurable por el administrador del sistema. Una de dichas comunidades recibe el nombre de comunidad pública, y sus variables pueden ser accedidas sólo para lectura.

Por el contrario los valores asociados a las variables que componen la otra comunidad, denominada comunidad privada, pueden ser modificados.

Toda la seguridad proporcionada por el sistema se basa en el hecho de que es necesario conocer el nombre asignado a una comunidad para conseguir el acceso a la información proporcionada por sus variables. El nivel de protección ofrecido por la versión original del protocolo es, por tanto, muy débil. Más aún si se tiene en cuenta que los nombres de comunidad incluidos en los mensajes del protocolo SNMP viajan por la red en texto plano y por consiguiente pueden ser obtenidos como resultado de ataques pasivos (escuchas malintencionadas).

Además, y sobre todo en el caso de la comunidad pública, está muy extendido el uso del nombre de comunidad configurado por defecto (public) por lo que un usuario ajeno al sistema puede obtener gran cantidad de información acerca del mismo utilizando el protocolo SNMP.

Con el fin de aumentar la seguridad del protocolo es necesario realizar cambios en su modelo administrativo para introducir los conceptos de autentificación, integridad y privacidad así como para mejorar el control de acceso a la información.

En primer lugar se identifican las posibles amenazas a las que dicho protocolo se encuentra sometido. Las más importantes son las siguientes: modificación de los mensajes en tránsito o de su orden, suplantación y ataques pasivos (escuchas). En el caso concreto del protocolo SNMP no se consideran relevantes las amenazas de los tipos negación de servicio y análisis de tráfico.

Una versión segura del protocolo debería impedir en la medida de lo posible ataques de los tipos mencionados. El apartado siguiente describe la evolución que ha sufrido el protocolo a través de sus distintas versiones así como las principales mejoras aportadas por cada una de dichas versiones en relación a la seguridad.

Evolución de la seguridad proporcionada por el protocolo

Cronológicamente hablando, el primer intento serio de dotar al protocolo SNMP de un cierto grado de seguridad se corresponde con la versión denominada SNMPsec, cuyos fundamentos se definen en los RFC 1351 y 1352. Los elementos introducidos en dicha versión para mejorar la seguridad del protocolo forman la base de todas las versiones posteriores y se siguen utilizando en la actualidad.

Las principales innovaciones propuestas en la versión SNMPsec son la identificación unívoca de las entidades que participan en las comunicaciones SNMP, lo que permitirá grandes mejoras y mayor flexibilidad en cuanto al control de acceso, así como la utilización de mecanismos criptográficos para conseguir autentificación, integridad de los mensajes y privacidad.
Dicha versión introduce los siguientes conceptos:
Party SNMP. Es un contexto virtual de ejecución cuyas operaciones se pueden encontrar restringidas a un subconjunto del conjunto total de operaciones permitidas por el protocolo. Un party involucra un identificador, una localización en la red utilizando un protocolo de transporte determinado, una vista MIB sobre la que opera, un protocolo de autentificación y un protocolo de privacidad.

Vista sub-árbol y vista MIB. Una vista sub-árbol es un conjunto de variables de un MIB (Management Information Base) que tienen como prefijo un identificador de objeto común. Una vista MIB no es más que un conjunto de vistas sub-árbol.

Política de control de acceso. Es el conjunto de clases de comunicación autorizadas entre dos parties SNMP o lo que es lo mismo el conjunto de mensajes del protocolo SNMP cuyo uso se permite entre dos elementos participantes en una comunicación de gestión.
Protocolo de autentificación. Sirve al mismo tiempo para autentificar los mensajes y para poder comprobar su integridad. Se suele utilizar un mecanismo de firmas digitales, como por ejemplo el algoritmo MD5 que calcula un digest del mensaje. El valor obtenido se incluye entre los datos transmitidos a la hora de llevar a cabo una comunicación.

Protocolo de privacidad. Sirve para proteger las comunicaciones contra escuchas malintencionadas. Se utiliza, por ejemplo, el algoritmo simétrico de encriptación DES (Data Encryption Standard).
La versión SNMPsec se adopta inicialmente con la introducción de la versión 2 del protocolo SNMP y pasa a denominarse SNMPv2p (Party-based SNMPv2).
Posteriormente el marco de trabajo SNMPv2, cuya definición no contiene ningún estándar en cuanto a seguridad, se asocia con otros modelos administrativos referentes a seguridad, y aparecen tres nuevas versiones del protocolo: SNMPv2c, SNMPv2u y SNMPv2*.

La versión SNMPv2c (Community-based SNMPv2) utiliza el mismo modelo administrativo que la primera versión del protocolo SNMP, y como tal no incluye mecanismos de seguridad. Las únicas mejoras introducidas en la nueva versión consisten en una mayor flexibilidad de los mecanismos de control de acceso, ya que se permite la definición de políticas de acceso consistentes en asociar un nombre de comunidad con un perfil de comunidad formado por una vista MIB y unos derechos de acceso a dicha vista (read-only o read-write).

La versión SNMPv2* proporciona niveles de seguridad adecuados, pero no alcanzó el necesario nivel de estandarización y aceptación por el IETF (Internet Engineering Task Force).
Por último, la versión denominada SNMPv2u (User-based SNMPv2) reutiliza los conceptos introducidos en la versión SNMPsec, introduciendo la noción de usuario. En este caso, las comunicaciones se llevan a cabo bajo la identidad de usuarios en lugar de utilizar el concepto de party existente en las versiones precedentes. Un mismo usuario puede estar definido en varias entidades SNMP diferentes.

La seguridad en la versión 3 del protocolo

La principal novedad introducida en la versión 3 del protocolo SNMP es la modularidad. En dicha versión una entidad SNMP se considera compuesta por un motor y unas aplicaciones. A su vez el motor se divide en cuatro módulos: dispatcher, subsistema de proceso de mensajes, subsistema de seguridad y subsistema de control de acceso.

Se observa, por tanto, que en la versión SNMPv3 se independizan los mecanismos utilizados para la seguridad (autentificación y privacidad) y para el control de acceso. De este modo, una misma entidad puede utilizar diferentes modelos de seguridad y control de acceso simultáneamente, lo que incrementa notablemente la flexibilidad y la interoperabilidad.

Se define un modelo estándar para seguridad basada en usuarios, USM (User Security Model) y otro para control de acceso basado en vistas, VACM (View-based Access Control Model). Se aprovechan los conceptos definidos en las versiones previas y al mismo tiempo la modularidad del protocolo permite la introducción de futuros modelos independientes de los actuales.

CAPÍTULO IV DISEÑO DE UN SISTEMA DE MONITOREO CENTRALIZADO.

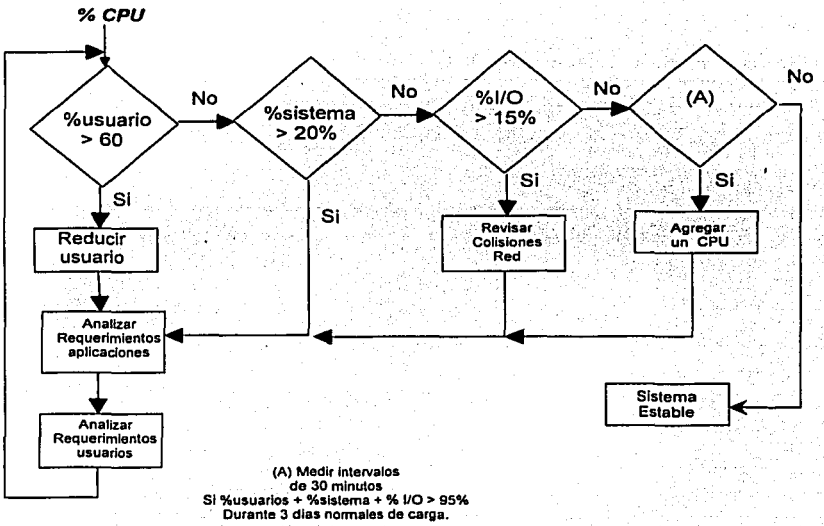
4.0 Análisis de los parámetros a monitorear (discos, procesadores, memoria, red, usuarios locales y usuarios remotos).

Daremos por hecho que los parámetros a analizar ya fueron definidos en el Capítulo II por lo cual nos centraremos al análisis de cómo se presentara la información a través de la herramienta centralizada de monitoreo.

La recopilación de la información se realizará en función de módulos que engloban los parámetros en cada dispositivo que se han nombrado (discos, procesadores, memoria, red, usuarios locales y usuarios remotos). Sin embargo se como se comentó en el Capítulo II se tiene que realizar una acotación de los parámetros que se analizarán ya que en este estudio puede ser tan intenso y completo como uno lo desee en función de la información dada por cada dispositivo.

Pero lo importante es obtener estadísticas que cumplan una relación de tiempo-desempeño aceptables, ya que mientras más información se maneje mayor será el desgaste del equipo en función principalmente de procesamiento y disco, lo cual puede llegar a ser perjudicial en estos casos se debe tener una media de rendimiento, para saber cual será el máximo de ocupación permitido del sistema.

Enseguida se mostrará el análisis basado en cada parámetro nombrado en el Capítulo II mediante algunos diagramas y fórmulas para sacar el rendimiento de cada dispositivo:



En el caso de la memoria se tiene que analizar solamente cual es la frecuencia en la que el sistema utiliza la memoria swap (memoria virtual), o el proceso más conocido como paginación, el cual se realiza cuando el sistema se ha quedado sin memoria física y no puede cargar programa alguno en memoria principal (memoria física), por lo cual el sistema empieza a bajar segmentos de memoria física a la memoria virtual (parte del disco asignada para este proceso), sin embargo el inconveniente es que la memoria virtual no es tan rápida como la memoria física y por otro lado hay programas que necesitan ser cargados en su totalidad en memoria física, en estos casos se tiene dos opciones:

Analizar requisitos de memoria para el sistema, y tratar de mejorara el desempeño de las aplicaciones.

Agrandar la memoria virtual, aplicando la siguiente regla:

Memoria Virtual = 2 x Memoria Física del sistema.

En caso de haber crecido el espacio de memoria virtual y el sistema sigue con el proceso de paginación en intervalos constantes (un día de prueba) y de haber analizado los requerimientos de las aplicaciones y en caso de seguir con el sistema con bajo rendimiento, se tendrá que adicionar más memoria física, esto dependerá de la cantidad del proceso de paginación del sistema en particular.

En el caso del disco se tendrá que analizar Kilobytes por segundo de entrada y los Kilobytes por segundo de salida, esto va de acuerdo con el tipo de tecnología que se ocupe SCSI, IDE o Fibra, y por otro lado la transferencia dependerá de la búsqueda de información en disco que está en función de RPM (revoluciones por minuto) del disco, con lo cual podemos calcular la latencia, la cual es el promedio de tiempo para que el disco una vez en la pista correcta encuentre el sector deseado, es decir el tiempo que tarda el disco en dar media vuelta. Velocidad de transferencia: velocidad a la que los datos (bits) pueden transferirse desde el disco a la unidad central. Depende esencialmente de dos factores: la velocidad de rotación y la densidad de almacenamiento de los datos en una pista.

Con esto podemos calcularla como:
3600 rpm = 1 revolución cada 60/3600 segundos (16,66 milisegundos)
Si calculamos el tiempo de ½ vuelta --> Latencia Promedio 8,33 milisegundos

RPM	1 Vuelta cada	Latencia
3600	16,66 mseg.	8,33 mseg.
4500	13,33 mseg.	6,66 mseg.
5400	11,11 mseg.	5,55 mseg.
7200	8,33 mseg.	4,16 mseg.
10000	6,00 mseg.	3,00 mseg.

En el caso de la red se analizará mediante el porcentaje de colisiones que se tiene en la red basándose en la siguiente fórmula:

Porcentaje de colisiones = (Núm. de colisiones / Paquetes enviados) * 100

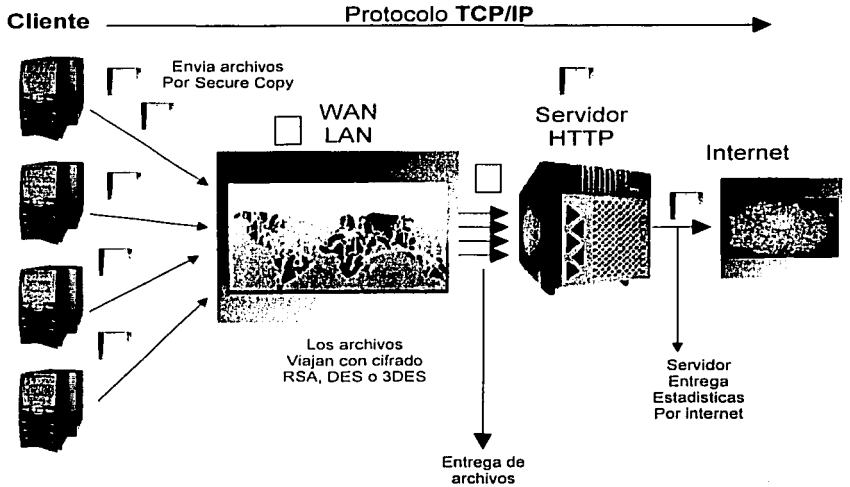
4.1 Análisis de la logística de comunicaciones entre el sistema centralizado y equipos remotos.

La logística de comunicaciones entre los equipos a monitorear y el servidor central se realizará ocupando el protocolo TCP/IP dentro de éste, tomaremos los siguientes servicios que implementan la comunicación y seguridad (RSA, DES y 3DES):

Servicios:

Secure Shell (Para administración de equipos remotos)
Secure Copy (Para transferencia de archivos)
http (Servicio para equipo centralizado)
netstat (Servicio para revisión de colisiones de red)
imap (Servicio para mandar alarmas mediante protocolo SMTP)
ntp (Servicio para sincronizar tiempo en servidor remoto y clientes)

Los servicios anteriores describen su funcionamiento en el siguiente diagrama de comunicaciones:



4.2 Creación del sistema centralizado.

El análisis estadístico de datos se fundamenta, principalmente, utilizando modelos gráficos con relación de uno a uno, ocupando programas creados en shell, php y utilizando la librería para crear imágenes en tiempo real llamada phplot; para más detalles de versiones de software y hardware ver el punto 4.6 en este capítulo.

La generación de graficas se desarrolló en tres etapas las cuales son:

- Obtención de datos de cada dispositivo por medio de programas shell.
- Manipulación de los datos (dar formato e introducir datos a la Base de Datos)
- Creación dinámica de programa php para generación de grafica.

Ejemplo: Generación de grafica para CPU %usuarios.

Programa para recavar y para dar formato a la información de clientes:

```
#!/bin/sh
set -x
ruta=/var/apache/htdocs/phplot-4.4.6
seg=1
muestreo=4
> $ruta/grafica.php
> $ruta/crea_php_b
> $ruta/datos.sar
> $ruta/inserta.sql
/usr/sbin/sar -u $seg $muestra > $ruta/datos.sar (Recava Información CPU)
usr=`tail -1 $ruta/datos.sar | awk '{ print $2 }'`
sys=`tail -1 $ruta/datos.sar | awk '{ print $3 }'`
io=`tail -1 $ruta/datos.sar | awk '{ print $4 }'`(Da formato a los datos)
idle=`tail -1 $ruta/datos.sar | awk '{ print $5 }'`
suma=`/usr/bin/perl -e "{ print ( $usr + $sys + $io ) }"`
hora=`date '+%H%M'`
hora_form=`date '+%H:%M:%S'`
horap=`date '+%H %M %S' | awk '{ print $1 }`
horam=`date '+%H %M %S' | awk '{ print $2 }`
fecha=`date '+%d/%m/%y'`
if [ $hora -ge 1200 ]; then
        meridiano=12
      ampm=pm
else
        meridiano=0
      ampm=am
fi
minutom=`/usr/bin/perl -e "{ print ( ($horap - $meridiano) * 60 ) }"`
muestram=`/usr/bin/perl -e "{ print ( $minutom + $horam ) }"`
/usr/bin/echo "USE learndb" > $ruta/inserta.sql
/usr/bin/echo "INSERT INTO cpudns1              (insertar datos en BD Mysql)
VALUES('0','$ampm','$suma','$muestram','$hora_form','$fecha');" >>
$ruta/inserta.sql
/usr/local/mysql/bin/mysql < $ruta/inserta.sql

/usr/local/mysql/bin/mysql < $ruta/exequery.sql | grep -v "cpu" | awk '{
print "array(\"\"," $2 "," $1 ")," }' > /var/apache/htdocs/phplot-
4.4.6/crea_php_b

/usr/bin/cat $ruta/crea_php_a >> $ruta/grafica.php
/usr/bin/cat $ruta/crea_php_b >> $ruta/grafica.php
/usr/bin/cat $ruta/crea_php_c >> $ruta/grafica.php
```

Programa para modificar rangos de gráfica en Base de Datos (Mysql):

```
<?php
$archivo = "/var/apache/htdocs/phplot-4.4.6/exequery.sh";
if ($dianoche=="")
{
$fp = fopen("exequery.sql","w+");
fputs($fp,"use learndb;\n");
fputs($fp, "select cpu,minuto from cpudns1 where fecha=\"14/02/03\" and
ampm=\"pm\" and minuto>\"0\" and minuto<\"720\" order by id;");
fclose($fp);
}
else {
$fp = fopen("exequery.sql","w+");
fputs($fp,"use learndb;\n");
fputs($fp, "select cpu,minuto from cpudns1 where fecha=\"$dma\" and
ampm=\"$dianoche\" and minuto>\"$menor\" and minuto<\"$mayor\" order by
id;");
fclose($fp);
}
system("sh ".escapeshellarg($archivo));
?>
```

Programas para generar gráfica dinámica en PHP:

En esta fase, se manejan tres archivos que son concatenados en uno solo los cuales llevan el esqueleto del programa en PHP que utilizara la librería phplot para realizar la grafica, enseguida se muestra el esqueleto de cada programa:

Archivo crea_php_a

```
<?
include ( "phplot.php");
$graph = new PHPlot;
$graph->SetDataType( "linear-linear");
// Specify some data
$data = array(
```

Archivo crea_php_b

Estos datos fueron extraídos con un query de la base de datos y formateados desde el scripts de shell.

```
array("",375,2),
array("",376,2),
array("",377,5),
array("",378,1),
.....
.....
.....
array("",382,2),
array("",383,1),
array("",384,1),
array("",385,3),
```

Archivo crea_php_c

```
);
$graph->SetDataValues($data);
$nameserver = `hostname`;
$hora=`date '+%H:%M:%S'`;
$horab=`date '+%H%M'`;
$Xvalorini="$menor";
$Xvalorfin="$mayor";
if ($horab>1200 )
{
$hora_dis="       12:00  13:00  14:00  15:00  16:00  17:00  18:00
19:00  20:00  21:00  22:00
 23:00";
}
else {
$hora_dis="       00:00  01:00  02:00  03:00  04:00  05:00  06:00
07:00  08:00  09:00  10:00
 11:00";
}
if ($Xvalor=="" || $Xvalorfin=="" || $Yvalor=="")
{
$Xvalorini="0";
$Xvalorfin="720";
$Yvalor="100";
}
//Specify plotting area details
$graph->SetImageArea(600,400);
$graph->SetPlotType( "lines");
$graph->SetTitleFontSize( "2");
$graph->SetTitle( "Muestreo de servidor $nameserver ARANEA-DF hora:
$hora");
$graph->SetPlotAreaWorld($Xvalorini,0,$Xvalorfin,$Yvalor);
$graph->SetPlotBgColor( "white");
$graph->SetPlotBorderType( "left");
$graph->SetBackgroundColor( "white");

//Define the X axis
$graph->SetXLabel("$hora_dis");
$graph->SetHorizTickIncrement( "60");
$graph->SetXGridLabelType( "plain");

//Define the Y axis
$graph->SetYLabel( "% Ocupacion CPU");
$graph->SetVertTickIncrement( "10");
$graph->SetPrecisionY( "0");
$graph->SetYGridLabelType( "right");
$graph->SetLightGridColor( "blue");

$graph->SetDataColors( array( "red"), array( "black") );

$graph->DrawGraph();
?>
```

Los esqueletos anteriores se concatenan dejándolos en un archivo para ser utilizado por la librería phplot, en nuestro caso se llama grafica.php. Con esto hemos terminado el programa para las estadísticas del CPU, hay que señalar que el servicio de servidor http debe estar configurado e instalado. Este proceso se realizará para las estadísticas de disco, memoria y red.

4.3 Creación de agente para sistema de alta disponibilidad.

Se realizaron tres programas para dar alta disponibilidad en el sistema de monitoreo, los cuales cubren disponibilidad en red, disponibilidad en CPU y disponibilidad de servicios de monitoreo, estos programas podrán ser instalados en una arquitectura de cluster, y en su caso de tener solo un servidor se podrán instalar en función de la duplicidad de dispositivos de hardware que se necesitan, los cuales serian; dos tarjetas de red y tener dos CPUs mínimo. Estos requisitos se nombrarán más a detalle en el punto 4.5 ,enseguida se muestran los programas para dar alta disponibilidad en los servicios anteriormente mencionados.

Programa para switcheo de nodo de red.

```
#!/bin/ksh
set -x
ADMINDIR=/scripts
upservers=`/usr/bin/cat $ADMINDIR/srvnum`
echo "0" > $ADMINDIR/srvnum
while read -r IP SRVNM
do
    if test `/usr/sbin/ping $IP | grep -c "is alive"` -eq 0; then
        # Wait 3 second before checking again
 sleep 1
     if test `/usr/sbin/ping $IP | grep -c "is alive"` -eq 0; then
upservers=`/usr/bin/cat $ADMINDIR/srvnum`
nuevo_serial=`/usr/bin/perl -e "{ print ( $upservers + 1 ) }"`
/usr/bin/echo "$nuevo_serial" > $ADMINDIR/srvnum
 fi
    fi
upservers=`/usr/bin/cat $ADMINDIR/srvnum`
echo $upservers
if [ $upservers -eq 2 ]
then
/usr/sbin/ifconfig hme0 unplumb
/usr/sbin/ifconfig hme1 plumb
/usr/sbin/ifconfig hme1 inet 13.52.2.85 netmask 255.255.255.0 broadcast
13.255.255.255 up      (IP y Broadcast variables)
fi
done < $ADMINDIR/mon_srv.dat
exit 0
```

En el programa anterior se debe crear un archivo llamado mon_srv.dat en el cual debe contener las IP de los servidores que estén el la misma red, estos servirán para saber si la red está en funcionamiento.

Programa para limitar y monitorear porcentaje de CPU en proceso definido.

Los parámetros a modificar son el proceso ha controlar (proid) y porcentaje máximo de uso en CPU que se le dará. En este caso esta dado en el rango de 2 a 3 por ciento.

```
#!/bin/sh
set -x
while (true)
do
proid=`/usr/bin/pgrep find` (cambiar find por proceso deseado)
proc=`/usr/ucb/ps -aux | grep find | cut -d" " -f7,8`
if [ $proc -ge 3 ]; then
/usr/bin/kill -23 $proid
fi
if [ $proc -le 2 ]; then
/usr/bin/kill -25 $proid
fi
done
```

Programa para mantener siempre en función el programa de monitoreo.

```
#!/bin/ksh
set -x
ruta=/var/apache/htdocs/phplot-4.4.6
cad="crearray.sh"
flag="no"
master=/opt/admin/scripts
ruta=/bita/files
hd=`date '+%d%m%y:%H%M%S'`
pgrep $cad && flag=si; export flag

if [ $flag == "no" ]; then
/$ruta/crearray.sh &
/usr/sbin/touch /$ruta/servicio.fallo.$hd
fi
```

4.4 Análisis de desempeño del sistema centralizado.

Se realizaron pruebas de carga en el sistema centralizado en diferentes equipos (Enterprise 250, Ultra 10 y Ultra 5) concluyendo que el sistema se utiliza de un 2.4% un 3.9% de los recursos. Esta medición se realizó a través de la herramienta llamada TOP, que cuenta el sistema operativo Solaris.

Ejemplo de medición con TOP.

```
last pid: 20563;  load averages:  0.01,  0.03,  0.04       12:59:58
119 processes: 118 sleeping, 1 on cpu
CPU states: 99.6% idle,  0.1% user,0.3% kernel,0.0% iowait,0.0% swap
Memory: 2048M real, 1192M swap in use, 4459M swap free

  PID USERNAME THR PRI NICE  SIZE   RES STATE   TIME    CPU COMMAND

 2565 oracle    11  59    0 1115M 1085M cpu1    0:02  3.01% crearray.sh
20561 root       1  58    0 2664K 1720K cpu1    0:00  0.16% top
20563 root       1  52    0 1680K 1280K sleep   0:00  0.05% rcp
  398 root      12  58    0 2456K 1064K sleep   0:00  0.00% mibiisa
 2594 oracle    11  30    0 1157M 1109M sleep   0:11  0.00% oracle
 4909 oracle     1  31    0 1115M 1091M sleep   0:07  0.00% oracle
  311 ctxsrvr    9  58    0 5160K 2336K sleep   0:05  0.00% ctxfm
 2408 oracle    10  58    0 8000K 7088K sleep   0:03  0.00% ctxXtw
12643 oracle    11  10    0 1116M 1091M sleep   0:02  0.03% oracle
    1 root       1  58    0  816K  256K sleep   0:02  0.00% init
 2700 oracle     1  58    0 1114M 1089M sleep   0:01  0.00% oracle
 2567 oracle     1  58    0 1114M 1087M sleep   0:01  0.00% oracle
  272 root       6  58    0 2704K 1144K sleep   0:01  0.00% vold
  287 root       1  58    0 1696K  344K sleep   0:01  0.00% prngd
 2510 oracle     8  59    0 9384K 7456K sleep   0:01  0.00% dtwm
```

4.5 Requisitos mínimos de hardware y software para la implementación.

Los requisitos mínimos para el sistema centralizado son:

Hardware:

1 CPU de 400 MHz (arquitectura RISC o CISC)

2 discos de 36 GB

128 GB de memoria

2 puertos de red 10/100 Fast Ethernet

***Software:**

Sistema Operativo Solaris 6,7,8 o 9 (Para arquitectura RISC o CISC).

Aplicación NSCPcom	Netscape Communicator
Aplicación SMCautoc	autoconf
Aplicación SMCautom	automake
Aplicación SMCbison	bison
Aplicación SMCdb3	db
Aplicación SMCegd	egd
Aplicación SMCexpect	expect
Aplicación SMCflex	flex
Aplicación SMCgcc	gcc
Aplicación SMCglib	glib
Aplicación SMCgtk	gtk+
Aplicación SMCjpeg	jpeg
Aplicación SMClibgcc	libgcc
Aplicación SMClibt	libtool
Aplicación SMClpng	libpng
Aplicación SMCm4	m4
Aplicación SMCmysql	mysql
Aplicación SMCncurs	ncurses
Aplicación SMCossl	ossl
Aplicación SMCperl	perl
Aplicación SMCprngd	prngd
Aplicación SMCsnort	snort
Aplicación SMCtcl	tcl
Aplicación SMCtk	tk
Aplicación SMCtop	top
Aplicación SMCxpm	xpm
Aplicación SMCzlib	zlib
Aplicación Apache	apache
Aplicación PHP	php
Aplicación Librería	phplot

*Las versiones que están manejando son a la fecha 14 Enero 2003.

CAPÍTULO V IMPLEMENTACION DEL SISTEMA DE MONITOREO CENTRALIZADO.

5.0 Implementación del sistema centralizado.

La implementación en el sistema principal, se realiza en dos pasos los cuales son:

- Compilación e instalación del lenguaje PHP y librería para gráficas PHPLOT
- Configuración de servidor Web
- Configuración e instalación de scripts realizados en shell y php.

Pasos para la compilación e instalación de lenguaje PHP versión 4.3.1

Primero se debe obtener la versión desde el sitio www.php.net, el archivo está comprimido y tiene la siguiente nombre con la extensión gz, php-4.3.1.tar.gz en este caso debe ser descomprimido de la siguiente forma:

```
# gunzip php-4.3.1.tar.gz
# tar xvf php-4.3.1.tar
```

Posteriormente se creará un directorio llamado php-4.3.1 en donde se almacenará todos los archivos fuentes para realizar la compilación. Se debe tener de espacio libre mínimo 200 MB para la compilación y el espacio para descomprimir el archivo.

También se debe obtener la librería de PHPLOT versión 4.4.6 que se puede bajar del sitio; www.phplot.com, esta librería se tendrá que descomprimir de la siguiente forma:

```
# gunzip  phplot-4.4.6.tar.gz
# tar xvf phplot-4.4.6.tar
```

Dicha librería creará un directoria llamado phplot-4.4.6 esta librería debe ser compilada al mismo tiempo que se está realizando la compilación del lenguaje php.

Enseguida se tendrá que crear las siguientes variables de ambiente:

```
# LD_LIBRARY_PATH=/usr/local/lib
# export LD_LIBRARY_PATH
# PATH=$PATH:/usr/local/bin:/usr/ccs/bin:/php-4.3.1
# export PATH
```

Es bueno recordar que para este momento debe haberse instalado todos los paquetes necesarios nombrados en el capítulo anterior, ya que en la compilación de php y sus librerías para gráficos se harán llamadas a varios de estos paquetes. Enseguida se procederá a la compilación, primero se creará la estructura de cómo se realizará la compilación.

```
# cd /php-4.3.1
# ./configure
# make
# make install
```

En el ultimo paso (install) se creará las ligas para el servidor de apache que introducirá el lenguaje php para que funcione en el servidor de web apache, así como la librería de phplot. Con esto finalizaremos la compilación por lo que resta realizar las prueba del buen funcionamiento de php en el browser, esto lo haremos con el siguiente programa en lenguaje php, para este punto ya debe estar configurado el servidor de Web Apache.

Programa de prueba en PHP (prueba.php).


```
<html>
    <head>
        <title>Example</title>
    </head>
    <body>

        <?php
        echo "Hola, Esta es una prueba de PHP script!";
        ?>

    </body>
</html>
```

El programa anterior lo abriremos desde el browser de nuestro servidor de Web en donde se compilo PHP y debemos ver el anuncio de echo que dice "Hola, Esta es una prueba de PHP script", teniendo esto sabremos que ya esta instalado y funcionando php en nuestro servidor.

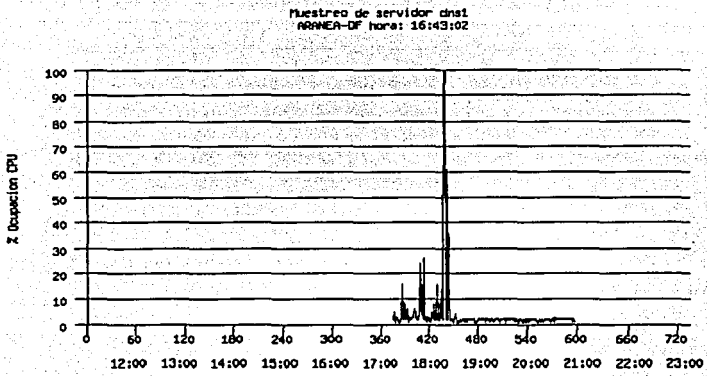
Por otro lado debemos probar que la librería PHPLOT debe estar funcionando también, por lo cual tendremos que ejecutar el programa siguiente para generar una gráfica dinámica de prueba.

Programa de prueba para librería PHPLOT.

```
<?
include ( "phplot.php");
$graph = new PHPlot;
$graph->SetDataType( "linear-linear");

// Specify some data
$data = array(
array("",375,2),
array("",376,2),
array("",377,5),
array("",378,1),
array("",379,2),
array("",380,3),
array("",381,2),
array("",382,2),
array("",383,1),
array("",384,1),
array("",385,3),
);
$graph->SetDataValues($data);
$nameserver = `hostname`;
$hora=`date '+%H:%M:%S'`;
$horab=`date '+%H%M'`;
$Xvalorini="$menor";
$Xvalorfin="$mayor";
if ($horab>1200 )
{
$hora_dis="     12:00 13:00 14:00 15:00 16:00 17:00 18:00 19:00 20:00 21:00 22:00
23:00";
}
```

```
else {
$hora_dis="      00:00 01:00 02:00 03:00 04:00 05:00 06:00 07:00 08:00 09:00 10:00
11:00";
}
if ($Xvalor=="" || $Xvalorfin=="" || $Yvalor=="")
{
$Xvalorini="0";
$Xvalorfin="720";
$Yvalor="100";
}
//Specify plotting area details
$graph->SetImageArea(600,400);
$graph->SetPlotType( "lines");
$graph->SetTitleFontSize( "2");
$graph->SetTitle( "Muestreo de servidor $nameserver ARANEA-DF hora: $hora");
$graph->SetPlotAreaWorld($Xvalorini,0,$Xvalorfin,$Yvalor);
$graph->SetPlotBgColor( "white");
$graph->SetPlotBorderType( "left");
$graph->SetBackgroundColor( "white");

//Define the X axis
$graph->SetXLabel("$hora_dis");
$graph->SetHorizTickIncrement( "60");
$graph->SetXGridLabelType( "plain");

//Define the Y axis
$graph->SetYLabel( "% Ocupacion CPU");
$graph->SetVertTickIncrement( "10");
$graph->SetPrecisionY( "0");
$graph->SetYGridLabelType( "right");
$graph->SetLightGridColor( "blue");
$graph->SetDataColors( array( "red"), array( "black") );
$graph->DrawGraph();
?>
```

El programa anterior nos dará la siguiente gráfica:



Revisado lo anterior y funcionando, se tendrá que proceder a la configuración e instalación de scripts en el servidor los cuales ya se nombraron a detalle en el capítulo IV, estos se deben ubicar en el directorio de configuración del Web Apache, así como tener los siguientes permisos:

chmod 700 /ruta_de_archivos_de_configuracion_servidor

Por ultimo se creará un proceso mediante el comando cron para estar ejecutando el monitoreo en el servidor principal con un intervalo de 1 minuto. También se debe crear un directorio donde se este almacenando todos los archivos que llegaran al sistema principal a través de los clientes que se estén monitoreando, este directorio lo podemos crear en raíz con el nombre de monitorclientes de la siguiente forma:

mkdir /monitorclientes
chmod 700 /monitorclientes
La importancia de este último directorio se remarcará más en el punto 5.1

5.1 Implementación de logística entre equipos remotos y central.

En la logística de intercambio de archivos de información de monitoreo a través del equipo central y los equipos remotos está basada en el cifrado de paquetes como se describe en el punto 4.1 del capítulo anterior, en este punto se describirá el proceso desde como se realiza la transferencia de los archivos de clientes al lugar de almacenamiento centralizado.

Cada cliente contiene prácticamente una estructura idéntica en función de archivos de control en cada cliente, así mismo cada cliente contiene su propia base de datos de donde se realiza la extracción de datos de monitoreo, con los cuales se genera el archivo que es enviado a través de la LAN o WAN hacia el servidor central, estos archivos son enviados por medio de un cifrado y depositados en el servidor central en el directorio **/monitorclientes**, desde este directorio se crea el archivo master llamado monitorgeneral.php en el servidor en donde se generan todas la graficas para ser presentadas por medio del protocolo http.

5.2 Implementación de seguridad en protocolos de comunicación.

Cuando establecemos la comunicaciones a través de los clientes y del sistema central se implementa el protocolo de seguridad llamado secure shell y secure copy, el cual realiza un canal entre los clientes y el servidor de manera privada ya sea por medio de LAN o WAN, este sistema de seguridad se utiliza para la administración remota y trasmisión de archivos de cada cliente, hay sistemas que no traen configurado o implementado el sistema, en dado caso, se tendrán que seguir los siguientes pasos para su implementación y configuración, en el caso particular del sistema Solaris sólo las versiones 8 y 9 contienen este sistema, en versiones anteriores hay que instalarlo.

Primeramente se tiene que obtener el software para su instalación, afortunadamente en el caso de solaris ya está compilado y empaquetado para ser instalado fácilmente, este paquete se obtiene del sitio; www.sunfreeware.com de este sitio tendremos que bajar el programa llamado openssh-3.6.1pl.

Antes de realizar la instalación debemos asegurarnos que el sistema contiene los siguientes paquetes, en dado caso de no tenerlos, los podemos bajar del mismo sitio.

Archivos (paquetes) requeridos antes de la instalación de secure shell.

openssl-0.9.7a-sol7-sparc-local.gz
tcp_wrappers-7.6-sol7-sparc-local.gz (opcional, pero recomendado)
zlib-1.1.4-sol7-sparc-local.gz
libgcc-3.2.2-sol7-sparc-local.gz
gcc-3.2.2-sol7-sparc-local.gz
perl-5.8.0-sol7-sparc-local.gz (opcional)
prngd-0.9.25-sol7-sparc-local.gz
egd-0.8-sol7-sparc-local.gz

La instalación de los paquetes debe realizarse con la siguiente secuencia de comandos:

Primero se tendrá que descomprimir con el comando # gunzip nombre_paquete.tar.gz
Segundo se abre el paquete con el comado # tar xvf nombre_paquete.tar
Tercero instalarlo en el sistema con el comado # pkgadd –d nombre_paquete

Después de la instalación de paquetes se debe agregar nuevos parámetros a la variable # LD_LIBRARY_PATH=/usr/local/lib:/usr/local/ssl/lib, debemos exportar la variable # export LD_LIBRARY_PATH, y debemos asegurarnos que en la variable PATH esten las rutas de /usr/local/bin y /usr/bin.

Adicional a esto, crearemos algunos archivos de control de la siguiente forma:

cat ***alguno_archivo_grande*** > usr/local/etc/prngd/prngd-seed
mkdir /var/spool/prngd
/usr/local/sbin/prngd /var/spool/prngd/pool

Posteriormente se generan las llaves de cifrado para los paquetes que serán transportados via WAN oLAN.

ssh-keygen -t rsa1 -f /usr/local/etc/ssh_host_key -N ""
ssh-keygen -t dsa -f /usr/local/etc/ssh_host_dsa_key -N ""
ssh-keygen -t rsa -f /usr/local/etc/ssh_host_rsa_key -N ""

Este proceso debe realizarse en todos los sistemas que contengan versiones de Solaris menores a 8 o 9. Posteriormente se probará el sistema desde los clientes al servidor de la siguiente forma:

ssh **IP_servidor** o **Nombre_servidor**
En este caso de pedirá crear una llave de confirmación para comunicación, esta confirmación solo se hará la primera vez. Se pedirá login y password del sistema central, se puede ocupar cualquier cuenta ya dada de alta.

En caso de fallo se puede consultar la siguiente página para volver a seguir el procedimiento paso a paso, http://www.sunfreeware.com/openssh26-7.html.

Para ver la funcionalidad completa del sistema centralizado se tendrá que hacer referencia al servidor por medio del browser con el siguiente URL, http://dominio.prueba/phplot-4.4.6/monitorgeneral.php.

CAPÍTULO VI

RESULTADOS Y CONCLUSIONES.

6.0 Resultados.

En la creación de la tesis, se llegó a la creación de un sistema en el cual se puede ver que los recursos necesarios para su implementación en función de costos es muy económicos, este punto aunado al desempeño que se puede ver, generó una relación de costo-desempeño muy importante en comparación con sistemas que hay en la actualidad y que por sus costos son definitivamente no rentables para la micro y pequeña empresa.

Por otro lado se a conseguido, a su vez, dar una fácil implementación y administración de forma segura y confiable en toda la infraestructura que implica el sistema, otro punto que cabe destacar lo fácilmente portable que puede ser el código de implementación refiriéndose a sistemas basados en UNIX, aunque cabe la posibilidad de realizar implementaciones en sistemas Windows con algunas modificación y utilizando programas sin costo para crear funciones para la obtención de estadísticas, es bueno comentar que todas las aplicaciones y librerías existen para el sistema windows. Sin embargo hay que remarcar la necesidad de utilizar programas externos que son gratuitos para realizar toda la logística en la obtención de datos.

Sin salir del punto, en alcances y resultados, se creó un sistema con los parámetros establecidos desde el inicio de la tesis, obteniendo un monitoreo confiable para la toda de decisiones en una empresa. Este punto de toda de decisiones fue una meta que se alcanzó debido a que la información se puede consultar de forma cualitativa y cuantitativa en función de la base de datos que se tiene, sin mas los resultados fueron alcanzados en su totalidad sobre el tiempo estimado de desarrollo, pruebas e implementación.

6.1 Estado Actual.

El sistema actual cuenta con el soporte suficiente para la implementación en equipos de producción, ya que está creado con estándares de comunicaciones por medio del protocolo TCP/IP, así como para tener un sistema de seguridad en el transporte de archivos entre clientes y servidor central. El sistema ha sido empaquetado para ser instalado de forma automática lo cual implica sólo pequeños cambios en clientes y servidor.

6.2 Trabajo a futuro

El sistema en la actualidad está disponible para que en futuras versiones se pueda extender a diferentes plataformas en función de los sistemas operativos como son Windows, Mac y Linux., en la actualidad únicamente se ocupa el agente nativo en Solaris (MIB) para la comunicación entre switches y concentradores que soporten el protocolo, este proceso es para el intercambio de información en MAN o LAN .

Se tiene pensado también crear otros agentes de alta disponibilidad para cluster que contengan más de dos nodos, ya que en el sistema actual sólo está creado para soportar dos nodos. Para futuras mejoras, cualquier persona interesada podrá obtener los códigos fuentes, e información al email: vidalgvg@yahoo.com esto es porque cada vez que se llegue a involucrar más gente en dicho proyecto se podrán realizar correcciones y aportaciones más precisas sobre el tema de monitoreo y necesidades de cada caso.

6.3 Conclusiones

Se puede concluir que el sistema realizado en comparación con los sistemas actuales de monitoreo específicamente para sistemas UNIX es de un alto nivel de competencia y funcionalidad debido a las consideraciones de costos y beneficios que se obtienen de este, los sistemas de vanguardia posiblemente contendrán otras opciones sobre el sistema expuesto aquí, pero sin duda, esto se podrá igualar con la implementación progresiva de dichas propiedades que no se tienen actualmente, como podría ser distintas formas de graficación. Lo sobresaliente del sistema que se plantea en la tesis son los siguientes puntos:

- Costos nulos del sistema.
- Servicio de monitoreo en cualquier punto del mundo.
- Administración desde cualquier punto del mundo.
- Estadísticas a través de servicio estándar http
- Seguridad implementada.
- Fácil crecimiento en características, implementación y configuración.
- Código abierto para llevarlo a diferentes plataformas.

Los puntos anteriores hace al sistema una de los pocas opciones que hay en el mercado, considerándo la línea de sistemas UNIX. Por ultimo otro punto sobresaliente del sistema es su composición de elementos que lo forman, esto implica que debido a que todos los paquetes que lo constituyen tiene desarrollos adicionales disponibles en internet siendo freeware, los cuales se pueden implementar con totalidad facilidad para la administración, con esto se reducen costos de administración en lo futuro, un ejemplo seria la base de datos mysql que tiene el sistema de administracion AdminPHP facil de implementar y conseguir en Internet para la administración de las tablas base de dicha implementación..

GLOSARIO

Anillo.- Topología de red, conecta a las computadoras con un solo cable en forma de circulo.

Arbol.- Topología de red, pueden ser combinadas en una variedad de topologías híbridas más complejas. (ej. Anillo, Malla, Bus, etc).

Bus.- Topología de red, las computadoras están conectadas por un canal de comunicación en línea recta. Esta red es la más común y la más simple.

CISC.- (Complex Instrution Set Computer) Modelo para construír fisicamente un microprocesador, en el cual se ocupan un numero mayor de instrucciones por ciclo de reloj para realizar un programa en comparación con RISC.

COBIT.- (Control Objectives for Information and related Technology), las cuales son el una serie de procedimientos para el resultado de auditoria informática.

Cluster .- Sistema de alta redundancia para sistemas en producción.

Cracker.- Persona con un alto conocimiento a nivel informatico, que entra en los sistemas sin tener autorización usando estrategias diversas par poder romper la seguridad de dichos sistemas.

DES.- Algoritmo de cifrado para sistemas para creacion de passwords

DNS.- Sistema para resolver nombres o mejor conocidos como dominios en la Internet.

DRP.- Sistema de recuperación de desastres. Este se implanta con una serie de procedimientos ha seguir para la recuperación del sitio de producción en uno laterno.

Estrella.- En esta topología todos los cables de todas las computadoras son conectados a un dispositivo central llamado hub.

HTTP.-

Hub.- Dispositivo de para conectar y crear redes, se puede clasificar por el numero de puertos.

IP.- Dirección de red, con la cual se puede comunicar una computadora a traves de internet o redes locales a otros equipos.

ICPM.- Es un protocolo de control de mensajes y reporte de errores a traves de TCP, entre comunicaciones de diferentes equipos de computo.

ISACF.- Organización Mundial de Auditores de Sistemas de Información.

LAN.- Son redes de propiedad privada , de hasta unos cuantos kilómetros de extensión.

MAN.- Son redes que se extienden sobre un área geográfica extensa.

Mainframe.- Equipos de computo con propiedades de procesar gran cantidad de información y tener disponibilidad de servicios similares a los clusters.

MD5.- Es un metodo de algoritmo para realizar una autentificación en un sistema de computo.

MPP.- Es un sistema de procesamiento masivo en el cual tiene la propiedad de realizar una distribución de la memoria RAM entre los procesadores de modo que se parezca a una red y asi optimizar el tiempo de acceso a memoria, pudiendo dar más velocidad en la operaciones por ciclo de reloj.

MIB.- Es un programa que se utiliza a traves de la red sacar datos del estado de diferentes sistemas en la red. Este es utilizado por el protocolo SNMP.

Malla.- Topología de red, todas las computadoras están interconectadas entre sí por medio de un tramado de cables. Esta configuración provee redundancia porque si un cable falla hay otros que permiten mantener la comunicación.

Orage Book.- Documento integrado por politicas de seguridad, dadas por los Estados Unidos de America.

PHP.- Lenguaje para programar aplicaciónes en Internet, el tiene la facilidad de tomar datos en linea del usuario.

Pipelines.- Este termino se puede aplicar a diferentes esquemas en el tema de computación, en este caso se dará el sentido sobre el diseño de CPU a nivel de circuitos llamados conductos segmentados, en los cuales pueden repartir la carga del CPU en función de cada ciclo de reloj.

RAID.- Es un concepto referido a dar un nivel más alto de disponibilidad en los disco, o para crear espacios de almacenamiento superiores a las propiedades fisicas de los discos actuales. Este concepto es basicamente una agrupación de disco para que se pueda ver como un solo disco.

RPM.- Se refiere a la velocidad de rotación por minuto de los disco (Revoluciónes por minuto).

RISC.- (Complex Instrution Set Computer) Modelo para construír físicamente un microprocesador, en el cual se ocupan un numero menor de instrucciones por ciclo de reloj para realizar un programa en comparación con CISC.

RSA.- Algoritmo de cifrado para proporcionar password al sistema operativo.

SMP.- . En SMP múltiples procesadores comparten la memoria RAM y el bus del sistema.

SPP.- Esta es un híbrido de SMP y MPP la cual utiliza una memoria jerárquica de dos niveles para alcanzar la escalabilidad. La primera capa de memoria consiste de un nodo que es esencialmente un sistema SMP completo, con múltiples procesadores y su memoria globalmente compartida.

SNMP.- Simple Network Managenement Protocol, Protocolo Simple de Administración de Redes.

TI.- Tecnología Informática

TCP.- Protocolo de comunicación entre computadoras, es el más popular y usado.

Topología.- Arreglo de computadoras en el cual pueden transferir datos o compartir información.

UDP.- Protocolo de comunicación sin aviso de conexión entre computadoras, a diferencia del TCP no tiene bloques de control de mensajes recibidos.

WAN.- Son una versión mayor de la LAN y utilizan una tecnología muy similar. Actualmente esta clasificación ha caído en desuso.

3DES.- Algoritmo de cifrado la diferencia entre DES y este es que maneja una llave más grande y por lo tanto es menos probable el problema de Hackers en el sistema.

BIBLIOGRAFÍA

Enterprise Security Solaris Operating Enviroment
Sun Bluprinnts
Autor: Alex Noordergraaf, et al
Editorial: Prentice Hall
Edición: 2002

Tecnicas criptograficas de protección de datos
Autor: Hernandez Luis
Editorial: Alfaomega

Toma de desiciones por medio de investigación de operacipnes
Autor: Thieraf, Robert J.
Editorial: Limusa-Noriega

Herramientas de programación patra shell de UNIX.
Autor: Medinets, David
Editorial: McGraw Hill

Administración Solaris I.
Sun Services
Editorial: Sun Services Educational.
Edición: 1998

Administración Solaris II.
Sun Services
Editorial: Sun Services Educational.
Edición: 1998

Administración de Redes TCP/IP Solaris
Sun Services
Editorial: Sun Services Educational.
Edición: 1998

http://docs.sun.com
http://www.whatis.com
http://www-cse.stanford.edu/classes/sophomore-college/projects-00/risc/riscciscl/
http://www.azc.uam.mx/publicaciones/enlinea2/num1/1-2.htm
http://www.desarrolloweb.com/manuales/12/
http://es.tldp.org/Manuales-LuCAS/manual_PHP/manual_PHP/
http://www.phpiot.com/
http://spisa.act.uji.es/~peralta/os/
http://www.testking.com/
http://www.apache.org/index2.html
http://httpd.apache.org/docs/
http://www.demiurgo.org/doc/shell/shell.html
http://www.aulafacil.org/CursoHtml/temario.htm
http://www.chiark.greenend.org.uk/~sgtatham/putty/