UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

ESCUELA NACIONAL DE ESTUDIOS PROFESIONALES

CAMPUS ARAGÓN

ANALISIS DEL SISTEMA DE SEGURIDAD EN UNIX

T E S I S

QUE PARA OBTENER EL TITULO DE

INGENIERO EN COMPUTACION

P R E S E N T A :

PEDRO REYNA FIGUEROA

ASESOR:
ING. JUAN GASTALDI PEREZ

MÉXICO **1998**

A mis padres:

Sr. Pedro Reyna Bruno
Sra. Claudia Figueroa López
Con el más sincero agradecimiento y respeto.

A mis hermanos:

Jesús, Javier y Erika

A mis amigos.

Al Ing. Juan Gastaldi P-rez
Por aceptar dirigir la presente tesis y por su
Apoyo en la realización de la misma.

A mis supervisores:
Ing. Antonio Ortíz Peña
Ing. Ernesto Peñaloza Romero
Ing. José Manuel Quintero
Ing. Manuel Martínez Ortíz
Por sus valiosos comentarios.

OBJETIVOS:

La presente tesis tiene como objetivo principal, el dar a conocer un panorama general de la seguridad en el sistema UNIX, mediante la información que se maneja en su entorno.

Como objetivos particulares se muestran los siguientes:

I.- Dar a conocer los conceptos básicos que se manejan en la aplicación de la seguridad en un sistema y recomendaciones básicas para su protección, a través del estudio de la información que rodea a una red.

II.- Mostrar los métodos con que cuenta el sistema UNIX para su protección y la forma en que se implementan adecuadamente, haciendo mención de los elementos con que cuenta el sistema UNIX para su seguridad.

III.- Dar a conocer y analizar los factores que existen dentro de la administración del sistema UNIX, así como la forma de implementar una política de seguridad confiable y práctica, dando a conocer las herramientas básicas que la conforman.

IV.- Conocer las barreras de protección, implementación y el tráfico de información del sistema UNIX, dando un panorama general de los elementos que la integran.

II.- METODOS DE SEGURIDAD

III.- POLITICAS DE SEGURIDAD

IV.- BARRERAS DE PROTECCION

INTRODUCCION

Actualmente las computadoras son una herramienta imprescindible para un buen número de tareas, y la aparición de las redes fue sin duda, el factor determinante para que las posibilidades de la informática llegaran a todas las empresas y usuarios particulares.

Desde que apareciera la primer computadora su número ha crecido exponencialmente así como sus campos de aplicación. Es fácil imaginar la cantidad de información que se les ha confiado y la importancia que ésta puede tener.

A ellas confiamos una parte importante de nuestro trabajo, valiosa información cuya pérdida sin duda es lamentable. Pese a todo esto, es habitual no dedicar la debida atención a su protección frente a las múltiples amenazas que les acechan.

Hablar de múltiples amenazas tal vez pueda parecer exagerado y hasta cierto punto sensacionalista, sin embargo no es así. Las amenazas son diversas, desde la pérdida de datos por accidentes a causa de errores humanos, motivados en muchos casos por desconocimiento (esta es la amenaza más frecuente y no deja de ser igual de peligrosa que las demás), pasando por los populares virus informáticos, hasta robo de información, fallos de componentes físicos, fallos de alimentación, etc.

La seguridad es algo que cada vez tiene mayor importancia, en este campo lo que se busca es una mayor fiabilidad de los equipos, posibilidades de recuperación de datos, control de accesos mediante claves, cifrado de la información, aplicaciones de copias de seguridad, sistemas tolerantes a fallos, etc.

La importancia creciente que está teniendo todo lo relacionado con la seguridad, es una justificacion importante de la presente tesis, donde se analiza las brechas que se pudieran abrir dentro de un sistema, más en especifico dentro del sistema UNIX, este sistema operativo ha tenido una influencia importante para el desarrollo de otros sistemas operativos, ademas que ha sido un pionero en seguridad.

La presente tesis se ha divido en cuatro capítulos:

El primero habla acerca de los conceptos que se manejan dentro de la seguridad, todo aquello que representa una amenaza para el sistema, como lo pueden ser los virus, las fuentes de infeccion, los intrusos, robos de contraseñas y la forma indicada de implementarla. Tambien se indican los componentes de seguridad de una red. Y por último algunas recomendaciones para protegerse a sí mismo como usuario del sistema UNIX.

El segundo capítulo explica los métodos con los que cuenta UNIX dentro de su propio sistema para la indicada protección, comenzando con los archivos más importantes en el aspecto seguridad; los comandos con los que cuenta el sistema para la protección de la información, el modo de implementar correctamente los permisos dentro de una cuenta de usuario. Se da una explicación del cifrado de contraseñas y archivos, que no deja de ser la forma básica de seguridad en UNIX y en cualquier sistema. Y por último una breve explicación del sistema kerberos, que se menciona como un ejemplo de sistema de autenticación.

El tercer capítulo analiza los factores y asuntos que se deben tomar en cuenta para el diseño de una red segura. Estos factores se formalizan en una política de red que siempre será un valioso apoyo para identificar las amenazas de seguridad, realizar análisis de riesgo y determinar cómo proteger los recursos de la red.

En el último capítulo se da una introducción a los enrutadores de selección y a los filtros de paquetes que conjuntamente conforman las barreras de protección, qué tienen como función principal controlar el tipo de tráfico que puede existir en la red; de esta manera es posible restringir aquellos servicios que puedan comprometer la seguridad de la red.

I.- CONCEPTOS DE SEGURIDAD

El sistema UNIX se diseño de modo que los usuarios pudieran acceder fácilmente a sus recursos y compartir información con otros usuarios. La seguridad era un aspecto importante, pero secundario. No obstante, el sistema ha incluido siempre características para preservarlo de usuarios no autorizados y para proteger los recursos de los usuarios, sin impedimento a los que si lo están. Estas capacidades de seguridad han proporcionado un cierto grado de protección, sin embargo, los intrusos han abusado de esto para acceder a muchas computadoras, debido a una administración poco cuidadosa del sistema o a la existencia de agujeros de seguridad.

Sin embargo en versiones recientes, UNIX ha incluido mejoras a la seguridad que hacen más difícil la obtención de acceso por parte de usuarios no autorizados. Los agujeros de seguridad que han sido identificados se han corregido.

El sistema operativo UNIX ha servido de modelo para el desarrollo de otros sistemas operativos, entre otros aspectos en lo referente a la seguridad. El origen de UNIX se sitúa en un proyecto desarrollado en los laboratorios Bell de AT&T y General Electric en los años sesenta, aunque la primera versión del sistema operativo tardó casi diez años en aparecer. Al principio, sólo era un proyecto experimental y su código fuente era público, utilizándose como materia de estudio en numerosas universidades, pero a partir de la versión 7, que se presentó en 1979, se empezó a pensar en sus posibilidades comerciales y se interrumpió la distribución de su código fuente.

En 1982 apareció la primera versión comercial, el sistema III y al año siguiente se presentó el Sistema V, con importantes mejoras, que obtuvo un gran éxito y que en su versión 4.2 es uno de los sistemas operativos más utilizados en medios y grandes sistemas. En 1993, UNIX pasó a ser propiedad de Novell que adquirió los laboratorios USL (UNIX System Laboratories).

En el entorno del PC, Microsoft desarrollo una versión limitada de UNIX que se comercializó para distintas plataformas Intel alcanzando gran popularidad y a la que llamo XENIX. En 1985, apareció una versión de UNIX Sistema V que se ejecutaba en un 386. En 1987, AT&T y Microsoft trabajaron en la fusión de los sistemas XENIX y UNIX Sistema V lo que dio lugar al nacimiento de la versión 3.2 para PC. Destacable entre las versiones UNIX para PC es la popular Linux.

En el diseño de UNIX no se contemplaba la seguridad como un factor prioritario; sin embargo, en el momento en que comenzó a comercializarse y a utilizarse en buen número de empresas, tanto públicas como privadas (entre ellas el Departamento de los Estados Unidos), la seguridad pasó a ser fundamental y el objetivo prioritario de desarrollo.

La versión 3.2 del Sistema V ya incluía numerosas mejoras en seguridad y la versión 4.0 (SVR4) se adaptaba a los requerimientos del nivel de seguridad C2 del libro naranja. Se encuentra también disponible una versión con características de seguridad avanzada o seguridad

multinivel bajo el nombre UNIX SVR4 MLS (multi-Level Security) que cumple los requisitos del nivel B2, UNIX SVR4 ES (Enhanced Security).

UNIX System V proporciona una variedad de características de seguridad. Entre ellas se incluye la identificación y validación del usuario a través de nombres de presentación y contraseñas, el control de acceso discrecional a través de permisos, las capacidades de cifrado de archivos y las características de auditoría, como el registro de última presentación, sin embargo, la versión 4 no alcanza el nivel de seguridad requerido por aplicaciones sensibles, como las que se encuentran en aplicaciones gubernamentales y militares.

I.1.- NIVELES DE SEGURIDAD

A la seguridad se le puede definir al estar libre y excento de todo daño, todo aquello que permite defenderse de una amenaza. Se considera que algo es o está seguro si ninguna amenaza se cierne sobre ello o bien el riesgo de que las existentes lleguen a materializarse es mínima, lo cual pocas veces se podrá afirmar de forma tajante, sea cual sea la naturaleza de lo que se esté hablando.

En un sistema las amenazas existentes son muy diversas: errores humanos, sabotaje, virus, robo, desastres naturales, etc y con esto la seguridad implica el resguardo de la integridad de software y hardware, ambiente y mantenimiento.

Los riesgos que corre la información son básicamente su pérdida, alteración y robo. La pérdida de información es, en casi todos los casos, el problema más grave y el que afecta a todos los usuarios. Puede producirse por diversos motivos. La alteración de la información puede producirse para cometer un fraude o como una forma de sabotaje provocada para perturbar o confundir.

El Departamento de Defensa de los Estados Unidos ha estandarizado la forma de evaluar los niveles de seguridad de sistemas informáticos. Estos estándares han sido publicados en el documento Trusted Computer System Evaluation Criteria. Este es conocido comúnmente como "El libro naranja", debido a que tiene su cubierta de este color. Los sistemas informáticos son remitidos por los vendedores al National Computer Security Center (NCSC) para su evaluación y calificación.

Hay siete niveles de seguridad informática descritos en el libro naranja. Estos niveles están organizados en cuatro grupos A,B,C, y D, de exigencias de seguridad decrecientes. Dentro de cada división hay uno o mas niveles de seguridad, etiquetados con números. Desde el nivel superior de seguridad hasta el inferior, estos niveles son A1, B3, B2, B1, C2, C1 y D. Todos los requerimientos de seguridad para un nivel inferior también son válidos para los niveles superiores.

I.1.1. NIVEL D1 (PROTECCION MINIMA)

En este nivel prácticamente no existe seguridad alguna, el hardware no recibe protección y el sistema operativo está expuesto fácilmente, este nivel de seguridad se presenta en sistemas operativos como MS-DOS, MS-Windows y System 7.x de apple Macintosh.

Un sistema no tiene que pasar ninguna prueba para ser clasificado como sistema de clase D. si se escuchan historias acerca de piratas que entran en "Computadoras del gobierno", es probable que se refieran a sistemas de clase D que no contienen datos delicados.

I.1.2. NIVEL C1 (PROTECCION DE SEGURIDAD DISCRECIONAL)

Este nivel es usado típicamente en el sistema UNIX. Existe seguridad para el hardware, pero es posible comprometerlo. Existe la forma de identificación y se utiliza para determinar derechos a los usuarios. Sin embargo la cuenta del administrador no tiene restricción alguna y con sólo su registro pueden realizarse algunas de las tareas.

Los usuarios deben ser identificados y validados. Cada usuario tiene control sobre sus objetos y puede limitar el acceso a éstos por parte de otros usuarios. También debe permitirse el acceso a los recursos por grupos de usuarios.

I.1.3. NIVEL C2 (PROTECCION DE ACCESO CONTROLADO)

El nivel C2 soluciona el problema del nivel C1. Tiene la capacidad de reforzar las restricciones a los usuarios en su ejecución de algunos comandos o en su acceso a algunos archivos basados en niveles de autorización.

Además este nivel requiere auditorias del sistema; incluyendo la creación de un registro de auditoria para cada evento que ocurre en el sistema.
La auditoria se utiliza para mantener los registros de todos los eventos relacionados con la seguridad, como las actividades practicadas por el administrador del sistema. Su desventaja es que requiere de autenticación adicional y recursos de disco del subsistema.

Con el uso de las autorizaciones adicionales, es posible que los usuarios tengan la posibilidad de realizar tareas de manejo de sistema sin necesidad de una contraseña raíz.

En este nivel debe haber una clara distinción entre el sistema de seguridad y los ficheros. Debe existir un control de accesos a recursos como ficheros y directorios mediante herramientas de auditoria. También es obligado eliminar todos los restos de cada proceso, ya sea en memoria o en registros temporales en disco. Ejemplos de sistemas operativos que cumplen estos requerimientos son Windows NT, Netware 4.0 y VMS 4.3. UNIX System V versión 4 (SVR4) también cumple tales requerimientos.

I.1.4. NIVEL B1 (PROTECCION DE SEGURIDAD ETIQUETADA)

Es el primer nivel que soporta seguridad de multinivel, como la secreta y la ultrasecreta. Los recursos controlados deben etiquetarse con un nivel de seguridad jerárquicos atendiendo al grado de confidencialidad del recurso: desclasificado, confidencial, secreto y alto secreto. Este nivel parte del principio de que un objeto bajo control de acceso obligatorio no puede aceptar cambios en los permisos hechos ni siquiera por el dueño de un objeto y todas las conexiones al sistema deben ser controladas. UNIX SVR4 MLS (Multi Level Security) cumple las exigencias de este nivel.

I.1.5. NIVEL B2 (PROTECCION ESTRUCTURADA)

Requiere que se etiquete cada objeto. Los dispositivos como discos duros, cintas o terminales podrán tener asignado un nivel sencillo o múltiple de seguridad en comunicación con otro objeto a un nivel inferior.

Debe existir un modelo de seguridad formal y debe comprobarse que el sistema se adapta al modelo. Los canales de transmisión de datos deben estar restringidos. Debe existir una persona encargada de la seguridad, que se encuentre por encima incluso del Administrador del sistema, cuyas funciones, por tanto, quedarán limitadas, Unix SVR4 ES (Enhanced Security) está en proceso de certificación en este nivel.

I.1.6. NIVEL B3 (NIVEL DE DOMINIOS DE SEGURIDAD)

Refuerza a los dominios con la instalación de hardware. El hardware de administración de memoria se usa para proteger el dominio de seguridad de un acceso no autorizado o la modificación de objetos en diferentes dominios de seguridad. Este nivel requiere que la terminal del usuario se conecte al sistema por medio de una ruta de acceso segura.

También debe existir un procedimiento para recoger peticiones de acceso a usuarios, y las acepten o no en base a una política de control de acceso y es necesario un sistema de auditoría que detecte posibles caminos para una violación de la seguridad.

I.1.7. NIVEL A (NIVEL DE DISEÑO VERIFICADO)

Incluye un proceso exhaustivo de diseño, control y verificación. Para lograr este nivel, todos los componentes de los niveles inferiores deben incluirse, el diseño requiere ser verificado en forma matemática, además es necesario realizar un análisis de los canales encubiertos y de la

distribución confiable. Esto significa que el hardware y software han estado protegidos durante su expedición para evitar violaciones a los sistemas de seguridad.

I.2.- IDENTIFICACION PERSONAL

El mantenimiento de la seguridad del sistema empieza por los propios usuarios, a quienes se dirige una serie de medidas y recomendaciones de seguridad que, aunque se citan en gran parte de los libros sobre UNIX y son conocidos por la mayoría de los usuario, a menudo se olvidan, dejando una vía abierta a los posibles intrusos.

Una serie de herramientas de dominio público y comerciales permiten evaluar la seguridad del sistema y descubrir los puntos débiles que puedan dar lugar a problemas en la seguridad. El más conocido entre ellos es COPS.

Las personas responsables de un sistema deben tener listos los elementos necesarios para identificar personal y llevar una bitácora de sus actividades en el sistema.

Las razones más comunes para la identificación del personal son:

- Limitar las actividades del usuario.
- Excluir usuarios no autorizados.
- Responsabilizar a cada usuario de sus acciones.
- Apoyar el control de acceso físico.

Dos características son necesarias para un sistema al establecer la identidad de un individuo. La primera que exista una "identificación" con nombre, numero de empleado, número de tarjeta, etc. y como segunda característica tener "autenticación" privada con contraseña. La autenticación es para dar al sistema la seguridad de que las personas son efectivamente quien dicen ser.

Cuando se selecciona un método de autenticación o identificación en un sistema, los siguientes factores deben ser considerados:

- La probabilidad de que el sistema rechace un usuario genuino o acepte a uno fraudulento.
- Que para burlar la autenticación propuesta se deba necesitar un gran esfuerzo.
- La necesidad de identificación personal para llevar una bitácora de actividades.
- *Que se conozcan las consecuencias de hacerse pasar por otro usuario.*

- La posibilidad de emplear diferentes métodos de identificación, esto dependerá de la sensibilidad de la función o sistema.
- El efecto de cada opción en la función y ejecución del sistema.
- La posibilidad para responder a requerimientos futuros.
- La aceptación del proceso para identificar al personal.
- El costo de implementación, entrenamiento y soporte.
- Los costos relacionados con el cambio de usuarios.
- Conocer las circunstancias en las que el método no es eficiente.

Los métodos de autenticación entran en tres categorías:

- La contraseña.
- Identificación de pruebas (algo distintivo, tarjetas de tiras magnéticas etc.)
- Características biológicas, (Apariencia física, huellas digitales, vasos sanguíneos de la retina, etc.)

I.2.1. CONTRASEÑAS

La forma más común de autenticación son las contraseñas, estas pueden ser difíciles de violar si son implementadas apropiadamente. Un ejemplo de forma mal implementada es darle un nombre común, o el nombre de algún pariente. La seguridad que da la contraseña puede ser comprometida por indiscreciones, o robo.

La contraseña debe elegirse con cuidado y no utilizar las típicas que podrían ser averiguadas en varias horas. Ejemplos de contraseñas típicas y por lo tanto fáciles de adivinar, son las formadas por un nombre seguido de uno o dos dígitos: cadiz93, deporte94; en los que se suele utilizar el nombre de la ciudad natal o de residencia, el de la pareja o el de algún hijo, el último lugar donde se fue de vacaciones, seguido o precedido de un número como el año actual, el de nacimiento, etc.

También es común y se considera como un error, tener dos contraseñas e ir rotándolas cada vez que haya que cambiarlas, por ejemplo: malaga95 y 95malaga. De esta forma, es más difícil que se pueda olvidar pero también es más fácil que alguien pueda llegar a descubrirla.

I.2.2. REVELACION DE CONTRASEÑAS

Las contraseñas son de poca ayuda si el usuario lo revela libremente, si esto sucede se debe reemplazar la contraseña y la antigua no puede ser utilizada nunca más. Esto sólo puede ser implementado si existe un sistema fuerte en responsabilidad individual y contraseñas únicas. El uso de la misma contraseña por un grupo compromete la responsabilidad individual y debe ser evitado.

Una contraseña debe ser siempre de 6 ó más caracteres, de preferencia deben incluir un símbolo y un número, existe poca oportunidad de adivinarla, a no ser que las personas que intentan hacerlo puedan limitar los candidatos posibles. Intentar violar una contraseña es hecho usualmente por personal que conoce al individuo o que tiene un buen diccionario de nombres o palabras. El proceso puede ser muy tedioso para las personas que desean adivinarla y especialmente peligroso para un usuario cuando el sistema permite intentar teclear la contraseña varias veces.

Un buen método para elegir la contraseña es utilizar palabras sin sentido compuestas por las iniciales de una frase fácil de recordar, así como mezclar letras minúsculas con mayúsculas, valores numéricos y/o signos de puntuación. Hay que tener presente que la contraseña mejor elegida no servirá de nada si no se guarda tan en secreto como si fuera el número de identificación de la tarjeta del cajero automático.

Las contraseñas contienen entre seis y ocho caracteres elegidos de los noventa y cinco que se pueden imprimir. Por tanto, las combinaciones de palabras distintas que se podrían generar sería la suma de las combinaciones posibles de palabras de seis caracteres más las combinaciones de las de siete más las de las ocho, es decir: $95^8 + 95^7 + 95^6$, aproximadamente unos seis mil billones (habría también que tener en cuenta que hay algunas restricciones como que debe haber, al menos, dos caracteres alfabéticos y uno numérico o especial; por tanto, habría que restar del total todas las palabras que cumplan dichas normas).

Así que si se parte del dato obtenido y suponiendo que se dispone de un superordenador capaz de cifrar 5,000 palabras por segundo, serían necesarios más de 30,000 años para generar todas las combinaciones posibles (un ordenador de pequeña potencia podría cifrar unas diez o quince palabras por segundo).

Sin embargo, una contraseña típica de cuatro letras minúsculas y los dos últimos dígitos del año, por ejemplo jose98, se podría averiguar en unas horas con un ordenador de mediana potencia. En este caso, se trataría de combinaciones de cuatro caracteres tomados de los 26 del alfabeto, es decir, $26^4 = 456{,}976$ combinaciones. Cifrando diez palabras por segundo se necesitarían menos de trece horas para generar todas las posibles combinaciones. Lo que quiere decir que se tardaría una media de seis horas aproximadamente en encontrar la contraseña en cuestión.

Existen varios programas que pueden ser ejecutados periódicamente por el administrador del sistema para comprobar si las contraseñas utilizadas por los usuarios son fáciles de descubrir y, por tanto, pueden comprometer la seguridad del sistema. Uno de ellos es el programa Crack,

de dominio público, que intenta descubrir las contraseñas de los usuarios consultando un diccionario propio y realizando una serie de permutaciones.

Otra manera de adivinar la contraseña es buscar la contraseña desde adentro del sistema ingresando con identificación y contraseñas de los administradores del sistema ó ingenieros. Esto puede evitarse cambiando las contraseñas preestablecidas en la primera oportunidad

Muchos sistemas tienen un revisador de ortografía y una lista de nombres de usuarios autorizados, como mínimo el sistema debe estar hecho para revisar los cambios por usuarios ausentes de la lista de nombres de usuarios.

El sistema generador de contraseñas puede ser bueno siempre que la gente recuerde la contraseña que el sistema le da. Si la contraseña es muy larga y no puede ser pronunciada entonces se le puede olvidar. Las contraseñas que genera el sistema deben ser pronunciables y no deben de ser de más de 10 caracteres.

Cambiando la contraseña regularmente hace que sea más difícil adivinarla y reduce el valor de intentos para hacerlo. La frecuencia con la que una contraseña debe ser cambiada depende de la aplicación pero una vez por mes es lo más común. Cuando los usuarios cambian su contraseña el sistema revisa que ellos no vuelvan a una que ya usaron antes.

El personal de seguridad registra los intentos fallidos de ingreso en la bitácora del sistema. Si existe evidencia de que un usuario en particular falla al entrar la contraseña correcta entonces el personal de seguridad entrevistará al usuario. Si este no puede explicar las entradas en la bitácora, se cambiará la identificación del usuario en la bitácora del sistema.

I.2.3. ROBO DE CONTRASEÑAS

Una contraseña no puede ser robada si no es revelada, se puede revelar de varias formas: oralmente, material de escritorio, pantallas, VDU, teclados y aparatos electrónicos. Un buen programa de seguridad combinado con medidas técnicas puede defender todas estas formas.

En ocasiones el personal escribe las contraseñas en algún lado (directorios, libretas, trozos de papel), ellos hacen esto si la contraseña es difícil de recordar, si tiene demasiados números, o si su uso es poco frecuente. Los usuarios deben ser ayudados para evitar tener que escribir sus contraseñas, haciendo estas cortas y pronunciables y reduciendo el número de diferentes contraseñas hacen que ellos puedan usarlo. Si el uso de la contraseña es poco frecuente existe la posibilidad de que se conceda su uso a alguien más.

Algunos sistemas producen impresiones que muestran contraseñas sin encriptar. Si las impresiones no son necesarias pueden ser bloqueadas, si son necesarias entonces las impresiones deben ser del dominio del gabinete de seguridad.

Pocos sistemas despliegan contraseñas en la pantalla, porque requieren que estos sean clasificados. El programa de seguridad puede recordarle a los usuarios que deben evitar escribir la contraseña en la presencia de otras personas.

Los aparatos electrónicos pueden ser usados para interceptar contraseñas, VDU's, teclados e impresoras producen señales las cuales pueden ser decodificadas a texto. En sistemas extremadamente sensibles protecciones especiales pueden ser requeridas. Esto concierne a cubrir el equipo con materiales que son sensibles a las señales de radio.

Las líneas de comunicación son vulnerables, si un sistema esta trabajando en red es posible que alguien entre a la línea y registre el tráfico. En muchos sistemas la secuencia de la bitácora inicial incluye a la identificación del usuario y contraseña, esta no es encriptada.

Como ya se explicó la contraseña debe ser cambiada periódicamente, varias versiones de UNIX ofrecen un servicio de caducidad de contraseña. Este mecanismo controla en qué momento pueden los usuarios cambiar sus contraseñas mediante la inserción de un valor en el archivo de contraseña después de la contraseña encriptada. Este valor define el período mínimo de tiempo que debe pasar antes de que los usuarios puedan cambiar sus contraseñas, y el periodo máximo de tiempo que puede transcurrir antes de que la contraseña expire.

La información sobre el control de la caducidad de la contraseña se guarda junto con la contraseña encriptada, como una serie de caracteres susceptibles de impresión. Los controles se incluyen después de la contraseña, precedidos por una coma (,). En general, un número de caracteres después de la coma representa la siguiente información:

- El número máximo de semanas en que la contraseña es válida.
- El número mínimo de semanas que deben transcurrir antes de que el usuario pueda cambiar su contraseña otra vez.
- Cuándo se cambió la contraseña por última vez.

Con versiones más recientes y seguras de UNIX en el mercado, se puede escuchar el término período de vida de una contraseña. Este es un período de gracia después del máximo periodo de tiempo en el cual el usuario puede todavía registrarse en su cuenta con el uso de la contraseña expirada. Al llegar al término del período de vida, la cuenta se inhabilita. Cuando el usuario trata de registrarse en un sistema mediante el uso de una cuenta inhabilitada, se le informa que la cuenta ha sido inhabilitada y que vea al administrador del sistema.

El mecanismo de caducidad de la contraseña no evita que el usuario cambie su contraseña y más tarde la vuelva a cambiar a la original. Sólo algunas versiones del sistema UNIX mantiene el rastro de las contraseñas que ha tenido un usuario. El proceso real de implantar la caducidad en una contraseña depende de la versión.

I.2.4. CARACTERISTICAS BIOLOGICAS

La forma más segura de identificación es usando características induplicables y que no se puedan separar nunca del individuo. el problema esta en que la forma de verificación debe ser económicamente factible, socialmente aceptable y adecuadamente segura.

Las huellas digitales son el uso más común de identificación en muchas circunstancias pero no pueden ser usados para los sistemas. Existen aparatos que pueden examinar y comparar huellas digitales pero el costo por estación es alto. Otras combinaciones para usar características personales para autenticación han sido probados pero ninguno a tenido éxito, excepto en sistemas de alta seguridad.

I.3.- SEGURIDAD FISICA

Los intrusos, el fuego y el agua son las principales amenazas físicas, la guerra, desobediencia civil, huracanes y terremotos pueden representar amenazas físicas dependiendo la zona donde se localice.

Las medidas necesarias para prevenir, detectar y tratar los problemas de la localización física del sitio dependiendo de la importancia y el dominio de la información.

Un buen personal entrenado contra los desastres puede reducir la frecuencia de amenazas físicas y el porcentaje de daños causados sin que esto represente un desperdicio de tiempo y dinero. Planeación y práctica es la clave del éxito para responder a una emergencia.

I.3.1. HACKERS

Por supuesto no sólo la naturaleza representa un peligro en la integridad física de los sistemas, las personas también pueden serlo, no sólo para el hardware sino también, incluso con más frecuencia para el software. A estas personas se les denomina "hacker" este término se le daba a la gente que era persistente, que trataba d romper cosas y averiguar cómo funcionaban. Como resultado de esta reputación, y debido a que la mayoría de la gente que hacía destrozos eran sabios de la ciencia de la computación, este término desarrollo una connotación negativa.

Una persona de este tipo desea ingresar a un sistema por algunas de las siguientes razones:

- Sólo por diversión.
- Sólo para mirar (curiosidad).

- Para robar recursos de cómputo, como tiempo de CPU.

- Para robar secretos del oficio u otra información propietaria.

Aunque no todas las razones para ingresar a un sistema son con intensiones dañinas, la mayoría de los casos se deberán tratar como si lo fueran. En UNIX la pieza de información más codiciada para los vándalos es el archivo /etc/passwd. Cuando se tiene la lista de nombres de cuentas de usuarios que es válida, resulta trivial crear un programa para adivinar las contraseñas. Sin embargo, muchos programas de registro modernos incluyen una demora de tiempo entre los indicadores de registro que se alargan más con cada intento frustrado. También se puede incluir también un código de programas para inhabilitar el puerto de acceso en caso de registrarse demasiados intentos fracasados.

Es correcto decir que después de que una contraseña ha sido encriptada no puede ser desencriptada. Pero eso no significa que la contraseña ya esté segura. Un buscador de contraseñas, es un programa que utiliza el vándalo que intenta adivinar las contraseñas en el archivo /etc/passwd mediante la comparación de éstas con las palabras de un diccionario. El éxito del programa buscador depende de los recursos de la CPU, la calidad del diccionario y el hecho de que el usuario tenga una copia de /etc/passwd.

Para tratar de dar mejores contraseñas en un sistema, un administrador podría escribir uno, sin embargo se deben tratar como peligrosos y que no vale la pena el potencial de problemas que estos crean, por mencionar un ejemplo, que consecuencias traería si este programa es robado, no se debe tomar esto a la ligera, habrían asuntos legales en caso de que hubiera daño directo como resultado del uso de ese programa.

I.3.2. TCB (BASE COMPUTACIONAL CONFIABLE)

La base computacional confiable (TCB) es parte del sistema de seguridad para los sistemas valorados como C2 de UNIX. Este sistema agrega un alto nivel de complejidad a la operación del sistema y a la administración del mismo. Este sistema trabaja mediante el traslado de bits en el archivo /etc/passwd hacia otros lugares, así como la adición de información sobre la información original. Los archivos que constituyen las bases de datos para la base computacional confiable se encuentran diseminados en varias jerarquías de directorio. Sin embargo, no es una buena idea editar esos archivos, ya que podría causar severos daños al sistema.

En un sistema que utiliza el TCB, se coloca un asterisco en el campo de contraseña de /etc/passwd. Esto porque la contraseña de usuario real se guarda junto con otra información de usuario en la base computacional confiable (TCB). El usar TCB no cambia la operación del sistema tanto como cuando UNIX proporciona los mismos servicios al emplear TCB. En algunas versiones de UNIX, como SCO UNIX, incluso si no está utilizando la seguridad C2, la base computacional confiable todavía estará en uso para proporcionar servicios de seguridad.

I.3.3. INTEGRIDAD EN LAS COMUNICACIONES

La integridad en las comunicaciones es crítica para todos las organizaciones. Esto requiere que todos y cada uno de los mensajes autorizados sean transferidos propia y adecuadamente. Si las líneas de comunicación son inseguras entonces los mensajes pueden ser revelados y modificados además el mensaje puede perderse o retardarse. Si los mensajes no son encriptados entonces revelaciones sin autorización pueden ocurrir si la autenticación es débil y no hay realimentación, entonces los mensajes no autorizados pueden ser admitidos.

Los canales de comunicación necesitan ser tolerantes en todas las aplicaciones triviales. La red de trabajo puede ser diseñada así que ningún componente puede fallar sin afectar al resto de la red. Pueden ser requeridas conexiones dobles para aplicaciones críticas.

Parte del trabajo del manejo de la red es guardar todas las conexiones autorizadas del sistema de información. Los diagramas son un buen punto de comienzo para diagnosticar fallas y determinar conexiones críticas las cuales necesitan ser duplicadas.

Las líneas de teléfono público no son seguras, estas pueden ser fácilmente violadas e intervenidas. Usar líneas privadas son una forma segura para comunicarse a alta velocidad, los canales de conexión son la principal forma de entrar de los "hackers".

I.3.4. VIRUS, GUSANOS Y CABALLOS DE TROYA

La amenaza para la integridad del sistema que pueden suponer los virus es mínima o prácticamente inexistente en un sistema operativo avanzado como UNIX que incorpora en su diseño los mecanismos de protección adecuados para impedirlo.

Con los caballos de Troya o programas que bajo una apariencia normal destruyen información cuando el usuario los ejecuta, no debe haber problemas si los permisos del sistema están correctamente asignados y se siguen algunas de las recomendaciones como por ejemplo la relativa al valor de la variable PATH o evitar ejecutar programas desconocidos de directorios donde cualquiera puede introducir un fichero.

VIRUS

Aunque es habitual pensar que los virus informáticos son algo relativamente nuevo, sus orígenes se remontan a los años cuarenta. Fue por aquel entonces cuando el célebre matemático John Von Neumann presentó el documento Theory and Organization of Complicated Automata donde se consideraba por primera vez la posibilidad de un programa capaz de reproducirse. Sus estudios tuvieron poco impacto por la imposibilidad de llevarlos a la práctica (las máquinas de cálculo electrónico llegarían varios años después de su muerte).

Veinte años después, John Conway desarrollo un programa que podría ser considerado como el primer virus informático: el juego de la vida (Game of Life) que simulaba el comportamiento de un virus biológico cuando ataca al cuerpo humano y entran en acción las defensas de éste.

Años más tarde, un grupo de jóvenes programadores de AT&T crearon un programa al que dieron el nombre de Core Wars (Guerra por la memoria) y cuya misión era reproducirse y apoderarse de la mayor cantidad de memoria posible del ordenador, compitiendo por ella con otros programas. Las pruebas se realizaron en equipos aislados, por lo que no existía la posibilidad de que los programas pudieran propagarse a otros sistemas.

Las claves del programa Core Wars se mantuvieron en secreto hasta que en 1983 Ken Thompson, autor del sistema operativo UNIX, las reveló en una conferencia de miembros de la Association for Computing Machinery (ACM). El año siguiente, la revista Scientific American publicó un articulo con abundante información sobre los virus, ofreciendo a los lectores la posibilidad de recibir más información sobre el desarrollo de estos programas por la cantidad de dos dólares.

El primer programa que llegó a infectar varios sistemas fue el monstruo de las galletas (Cookie Monster) y su único efecto era visualizar el mensaje I want a Cookie (Quiero una galleta) bloqueando el ordenador hasta que el usuario tecleaba la palabra Cookie.

De entre los cientos de virus que existen, algunos llegan a propagarse ampliamente en una zona, en un país o incluso en todo el mundo. Esto ha ocurrido con los que empleaban nuevas formas de propagación o nuevos métodos de ocultación, pero en un buen número de ocasiones la causa ha sido la distribución del virus en disquetes que acompañaban a algunas publicaciones o iban incluso, en programas comerciales.

El término virus es apropiado como término genérico para todas las formas de programación dañina Sin embargo, la definición básica es: "programa que tiene la capacidad de multiplicarse por sí mismo", existen definiciones aún más elaboradas por ejemplo la de Fred Cohen quién dijo que un virus era "programa que puede infectar a otros programas modificándolos para incluir una copia de sí mismo un poco alterada".

Los virus son programas ocultos que trabajan insertando copias de ellos mismos en otros programas. Los virus en microcomputadoras han recibido más publicidad, pero todas las computadoras de un sistema pueden ser infectadas. Los efectos de la infección pueden ser desde inconveniente hasta desastroso. Cualquier fuente de programas es una fuente potencial de infección de virus Así que un buen detector y recuperador son esenciales en un sistema.

CABALLOS DE TROYA

"Caballo de Troya" son programas disfrazados de aplicaciones atractivas pero dañan a los sistemas cuando se accionan por algunos eventos, el evento puede ser día, fecha, contador ó algún otro cambio en el sistema, como el dar de baja el nombre del programa de la nómina. Un caso muy famoso fue el del disco AIDS distribuido en 1989, que pretendía ser un programa de información sobre el SIDA pero pronto empezó a estropear los archivos de las computadoras que habían cargado el programa. Este troyano en concreto también pretendía obtener dinero de los usuarios, ofreciendo recuperar, por un cierto precio, los datos que se habían borrado u ocultado.

BOMBAS LÓGICAS

"Bombas lógicas" ó "Bombas de tiempo". Estos nombres se derivan del evento que activa la acción secreta del programa. Un programa puede ser virus o caballo de Troya, en ambos casos puede ser entregado como parte de una aplicación atractiva, duplicándose para infectar a otros programas y destruyendo cuando se acciona un evento predeterminado.

Los eventos que las activan pueden ser una fecha determinada o un cierto número de ejecuciones del sistema.

Ejemplificando un caso de "bomba lógica", Donald Burleson, un programador de la compañía aseguradora USPA, en septiembre de 1987 fue despedido, para él injustamente, dos días más tarde, se borraron a sí mismos aproximadamente 168,000 registros vitales de las computadoras de la compañía. Una de las divisiones más poderosas en contra de los delitos informáticos, la de Texas, retrocedieron un equivalente de dos años en los archivos del sistema y descubrieron que dos años antes de que Burleson fuera despedido había colocado una bomba lógica, un virus destructivo que estaba aletargado hasta el día de su despido. Se convirtió en la primera persona de América en ser encarcelada por acceso perjudicial a las computadoras.

GUSANOS

Se toma el término "gusano" de la traducción del acrónimo WROM (Write-One Read-Many) que significa una escritura-múltiples lecturas, de acuerdo con algunos expertos, un gusano es un programa que expande parte de sí mismo a través de diferentes computadoras que están conectadas en red.

Son diseñados para asegurar su propia supervivencia y robar recursos del sistema. Difieren de los virus en que ellos pueden existir y duplicarse por sí mismos sin atacar al programa anfitrión, en resumen los virus se multiplican y los gusanos crecen.

Los gusanos se duplican ellos mismos en progresión geométrica. Cada copia usa recursos del sistema y crea más copias hasta que eventualmente el sistema se dedica a copiar y ejecutar el gusano.

El caso más publicado sobre el destrozo de archivos de las computadoras tuvo lugar en noviembre de 1988, cuando Robert Morris Jr., de 24 años, liberó un virus gusano a través de internet. Básicamente, Morris escribió un programa que examinaba el archivo de claves de acceso de las computadoras en red, el programa tenía un gran éxito al encontrarlas. En un momento, más de un 80% de claves fueron descubiertas por el gusano, por si fuera poco el gusano era capaz de acciones mucho más sofisticadas, por ejemplo, tenia la capacidad de implantar programas troyanos que podían activarse más tarde y virus locales con un carácter propio que no podía asociarse a los gusanos.

Al final el gusano infectó a más de 6000 computadoras y bloqueó la mayor parte de internet durante varios días mientras se lograba el programa intruso. Nos podemos hacer una idea del nivel al que trabajo Morris por el hecho de que se necesitaron seis horas de un equipo de 18 UNIX de U.C. Berkeley y los laboratorios Lawrence Livermore para entender que era lo que causaba el problema. Las consecuencias para los usuarios de internet y el programador fueron desastrosas, para internet y la gente que controla los estándares del sistema operativo UNIX un considerable estrechamiento en la seguridad.

I.3.5. FUENTES DE INFECCION

Los distintos tipos de programas dañinos pueden encontrar acceso a la computadora por diversos caminos.

- Discos: flexibles, intercambiables, etc.
- Puertos: puertos de comunicación, interfaces de red.
- Dispositivos de entrada: el teclado.

La utilización de un dispositivo de entrada para infectar la computadora personal necesita que alguien se siente frente al teclado y escriba el programa virus. Esto no es imposible, algunos programas virus son extraordinariamente compactos y no se tardaría demasiado en introducirlos.

La infección a través del software transmitido por teléfono es más común que la infección directa Se carga el software desde otra computadora quizás un boletín electrónico, y el software sale infectado.

Un ejemplo es ARC513.EXE, una utilidad que se oferta para la compresión de archivos y su almacenamiento, con funciones similares a ARC.EXE, un legítimo y muy útil software de dominio compartido. Cuando se ejecuta ARC513.EXE no se comprimen archivos, ¡se destruye la pista 0 del disquete o del disco fijo!.

Algún aparato de almacenamiento puede ser usado para infectar un sistema. La ROM de una estación de trabajo puede tener un virus codificado por el fabricante, algún disco formateado puede traer programas ocultos. Los virus han sido propagados en toda clase de fuentes incluyendo revistas, discos promocionales, correo electrónico a usuarios y aplicaciones directas del proveedor.

Un canal de comunicación puede ser usado para introducir virus. Abrir estaciones de trabajo y permitir copias de programas de una organización a otra son fuentes de infección, todos los días nuevos archivos son automáticamente copiados a una red. Este sistema permite los manejos.

El personal de mantenimiento viaja de lado a lado insertando discos de diagnósticos en las máquinas. Algunos ingenieros de mantenimiento también distribuyen utilidades y software para el personal de apoyo en organizaciones que visitan. Los casos de infección viral han sido atribuidos a los ingenieros.

Los virus son más frecuentes en instituciones educativas y fundaciones de investigación; universidades y colegios son vulnerables particularmente porque las máquinas son compartidas, existe una larga lista de usuarios, muchos tienen la habilidad de escribir programas maliciosos y tienen la motivación y oportunidad del intercambio libre de programas con otras organizaciones.

I.3.6. PREVENCION

La responsabilidad de una buena administración y control de calidad debe contar con personal con suficiente conocimiento para seguir la lógica de un programa. Esto asegura que los programadores no pongan bombas de tiempo o lógicas en su código. Programación a bajo nivel, lenguajes obscuros y programación de sistemas son todos los medios donde esto puede suceder.

El riesgo de introducir software dañino de fuera de la organización puede ser reducido importando la menor cantidad de software como sea posible, usando sólo fuentes confiables de hardware y software y probar a fondo todas las unidades de almacenamiento al entrar a las organizaciones, cada maquina, programa y unidad de formateo debe ser revisado si tiene sello del distribuidor autorizado Teniendo cuidado especial para probar algún disco introducido por ingenieros o personal que uso computadoras en colegio y algún programa que viene de bbs's o redes de trabajo abiertas

A continuacion veremos algunos métodos ortodoxos para adquirir software:

- Hacerlo uno mismo
- Contratar a un programador.
- Empresas de software
- Paquetes comerciales

I.3.7. DETECCION

Muchos virus pueden ser detectados por una combinación de exploradores de software y un probador del medio. Los programas exploradores almacenan la búsqueda para atribuir características de virus conocidos. Existen varios exploradores disponibles de marcas comerciales y bbs´s. Estos programas deben ser actualizados cada que un nuevo virus es identificado.

Al menos dos exploradores deben usarse para mayor protección, también un explorador puede ser utilizado para revisar el otro. Un explorador siempre debe ser aplicado y ejecutado con protección de escritura para evitar que sea infectado.

El centro de competencia para establecer una prueba para explorar un sistema, si es posible debe consistir de una máquina que este física y lógicamente separada de otras máquinas. Antes de explorar una nueva unidad de almacenamiento la máquina debe ser inicializada por un disco sistema protegido contra escritura.

Podría ser útil conocer la mayor cantidad de síntomas en caso de infección, desafortunadamente, pocos son claros:

- Menos memoria de lo habitual.
- Pérdida de espacio en disco.
- Un cambio en el tamaño de los archivos del sistema operativo.
- Un cambio en la apariencia del directorio.
- Pérdida de archivos que sabía que no había borrado.
- Dificultades al leer archivos o discos.

Actualmente, algunas computadoras personales son suficientemente potentes para ejecutar versiones del sistema operativo UNIX. Algunas empresas lo están contemplando como alternativa al sistema operativo OS-2 y como modo de evitar los problemas de las redes locales basadas en PC/MS-DOS.

UNIX incorpora muchas prestaciones de seguridad, y se utiliza ampliamente en universidades e instituciones de investigación. Los sistemas UNIX más dispersos pueden conectarse en red.

I.4.- COMPONENTES DE SEGURIDAD DE UNA RED.

Para un intruso que busque acceder a los datos de la red, la línea de ataque más prometedora será una estación de trabajo de la red. Estas se deben de proteger con cuidado. Debe habilitarse un sistema que impida que usuarios no autorizados pueden conectarse a la red y copiar información fuera de ella, o incluso imprimirla.

Por supuesto, una red deja de ser eficiente si se convierte en una fortaleza inaccesible. El administrador de la red puede que tenga que clasificar los usuarios de la red con objeto de adjudicarles el nivel de seguridad adecuado. A continuación se sugiere un sistema en tres niveles

- Nivel de administración: aquellos que diseñan, mantienen o ponen en marcha la red.
- Usuarios fiables: aquellos competentes, que cumplen las normas y cuyo trabajo se pueda beneficiar de una mayor libertad de acceso a la red.
- Usuarios vulnerables: aquellos que muestran falta de competencia, son excesivamente curiosos o beligerantes, o en los que por alguna razón no se puede confiar.

Se requiere también de control de información, sobre todo se debe evitar su robo y mal uso, los siguientes componentes son indispensables para lograrlo:

- Control de acceso a la red: aquí se pueden utilizar cualquiera de las formas de control de acceso mencionadas anteriormente, todo en función del valor de los datos que se protegen y la sofisticación de los posibles atacantes.
- Estaciones de trabajo sin disco: claramente existe necesidad de impedir la copia de programas y datos fuera de la red en disquetes y de eliminar la posibilidad de que se puedan copiar virus y otros programas dañinos. Por otro lado, la persona responsable de mantener en marcha la red requiere de un acceso amplio a todas las unidades, una posible solución a esto es dotar a los usuarios vulnerables con estaciones de trabajo sin unidad de disco.
- Seguridad del servidor: Los servidores no son grandes sistemas del tamaño de una habitación. Se puede llevar fuera, o al menos se puede trasladar. Aunque no se halla considerado la posibilidad de realizar un anclaje físico de las estaciones de trabajo, se debe considerar para el servidor y sus periféricos inmediatos. La dependencia en que se encuentre el servidor no debe ser accesible para nadie excepto para el administrador de la red.

- Copias de seguridad del servidor: las copias de seguridad del servidor de archivos son un elemento especialmente valioso, debiendo quedar guardados en un lugar cerrado. Un conjunto de copias de seguridad se debe trasladar regularmente a otro lugar seguro. Se debe tener cuidado con los sistemas de copia de seguridad en cinta más lentos que han de trabajar de noche, en muchos casos sin vigilancia. Además, en quién se delega la tarea de hacer las copias de seguridad.

I.5.- RECOMENDACIONES DE SEGURIDAD EN UNIX

I.5.1. CONEXIÓN EN RED

La conexión en red necesaria para compartir información entre usuarios de distintos sistemas, también provoca que aumenten los riesgos.

El protocolo TCP/IP, muy extendido en UNIX al igual que en otros sistemas operativos de redes, cuenta con una serie de programas que se ejecutan sobre él y que permite transferir ficheros entre sistemas, ejecutar órdenes remotamente, entrar en una cuenta en un sistema remoto, ver que usuarios están conectados en el sistema remoto, etc. Por ejemplo, el programa rcp (remote copy) permite copiar ficheros desde un sistema remoto a otro también remoto. El programa rsh (remote shell) permite ejecutar órdenes remotamente y rlogin (remote login) entrar en una cuenta del sistema remoto.

Los mencionados programas requieren que en el sistema remoto exista un fichero con el nombre de los sistemas y de los usuarios que tienen permiso para ejecutarlos, no requiriéndose ninguna contraseña para poder hacerlo. Este fichero se denomina .rhost y debe residir en el directorio de entrada (HOME) del usuario remoto con permisos de lectura y escritura únicamente para el propietario.

```
$ cat .rhost
phoebe miguel
tetys csainz
zipi raul
```

El contenido de este fichero debe guardarse con cautela puesto que alguien que lo conociese podría suplantar a un usuario utilizando el nombre de alguno de los sistemas que aparecen en él. El suplantador conseguiría esto fácilmente disponiendo de un sistema propio y asignándole alguno de los nombres que aparecen, creando a continuación el usuario correspondiente.

Otros programas que corren sobre TCP/IP, como rexec (remote execution), que tiene la misma función que rsh, utilizan un método más seguro para ejecutar órdenes remotamente, siempre que se emplee correctamente. En este caso, en un fichero del sistema local de nombre

.netrc aparece la información sobre los usuarios de los sistemas remotos a los que se podrá acceder, con la contraseña sin cifrar incluida.

```
$ cat .netrc
machine tetys login czainz password npmmamt.
Machine zipi login raul password mvtqn94
```

Con este método, el único peligro es que alguien pueda ver el contenido del fichero; por ello, sólo el propietario deberá tener permiso de lectura.

El programa FTP (file transfer protocol) es también parte de las aplicaciones TCP/IP y permite entrar en una cuenta para transferir o borrar ficheros. A veces, se crean cuentas con el único propósito de que cuando el usuario entre se ejecute automáticamente un programa, para consultar datos por ejemplo; sin que tenga posibilidad de acceder al sistema operativo, inhabilitando incluso las teclas de interrupción para que no salga del programa. Se introducen, entonces en el fichero .profile las líneas apropiadas para que se ejecute directamente el programa adecuado. Sin embargo, si se accede a la cuenta con ftp, .profile no se ejecutará, obteniéndose acceso a los ficheros de la cuenta.

I.5.2. PUERTAS TRASERAS

Un usuario no autorizado puede acceder al sistema aprovechando un punto débil del sistema operativo, un descuido o desconocimiento de un usuario o una mala administración del sistema. Los puntos débiles abren vías de acceso al sistema saltándose el proceso de identificación -popularmente se conocen como puertas traseras - y son uno de los entretenimientos de los piratas informáticos (hackers) el descubrirlos.

Las versiones primitivas de UNIX contenían numerosos fallos que daban lugar a la existencia de otras tantas puertas traseras. Una de ellas consistía en pulsar Control-S, después un gran número de teclas y, al final, Control-Q. Esto provocaba un error de desbordamiento de la memoria intermedia de teclado y se interrumpía el programa de acceso lo que permitía entrar en el sistema. Otro parecido consistía en introducir un nombre de acceso y después pulsar una tecla suprimir (Del) repetidas veces, lo que provocaba en algunos sistemas que la comprobación de la contraseña fuera también interrumpida y se accediera al sistema directamente.

Todos esos fallos se fueron corrigiendo, sobre todo cuando se pensó en la posibilidad de comercializarlo y, en la actualidad, se puede decir que es uno de los sistemas más seguros que hay, existiendo incluso algunas versiones especiales con características de seguridad avanzada.

I.5.3. RESTRICCIONES DE USO

Cuando un usuario entra en el sistema se ejecuta un programa que interpreta sus órdenes y que es conocido como shell. Para restringir las acciones a determinados usuarios se les puede asignar un intérprete de órdenes restringido cuyo nombre es rsh (restricted shell). No se debe confundir con remote shell de TCP/IP) que limita en gran medida sus movimientos en el sistema. Algunas de las restricciones que ésta impone son la posibilidad de cambiar de directorio; es decir, inhabilita la orden cd, o la ejecución de programas que no se encuentren en los directorios del usuario o en los especificados en la variable PATH que, lógicamente, no puede ser modificada por el usuario como ocurre en la variable shell.

I.5.4. INICIO DE SESION.

Antes de iniciar una sesión es aconsejable apagar y volver a encender la terminal cuando alguien la haya dejado encendida, para evitar caer en la trampa de introducir la contraseña en un programa que simule la entrada al sistema y que un determinado usuario puede haber dejado ejecutándose.

I.5.5. ORDEN DE BUSQUEDA DE PROGRAMAS

En la variable PATH se indican los directorios donde deberán buscarse los programas que se ejecute, produciéndose la búsqueda en el orden que aparece en la variable. Si ésta se asigna de la forma PATH=:/usa/bin:/usa/lbin o PATH=:/usr/bin:/usr/lbin primero se buscará en el directorio actual y después en el resto de los que aparecen en la variable. Esto puede ser peligroso puesto que si se está en un directorio ajeno, se podría ejecutar inconscientemente algún programa de desconocidos efectos al que se le hubiera puesto el mismo nombre que alguna orden o utilidad del sistema.

Por ejemplo, si alguien creara el siguiente fichero en un determinado directorio:

```
$ cat > ls
rm $HOME/*
rm ls
ls $*
 D
$chmod 555 ls
```

Un usuario que tuviera asignada la variable PATH de la forma citada, al entrar en ese directorio y teclear ls para ver su contenido, ejecutaría este pequeño fichero de órdenes o shell script que borraría todos los ficheros del directorio del usuario; después se borraría a sí mismo

para no dejar rastro alguno y, por ultimo, visualizaría el contenido del directorio con lo que se dará la impresión de que la orden se ejecutó normalmente. Para evitar esto, dicha variable debe asignarse como PATH=/usr/bin:/usr/lbin:, es decir, con los dos puntos al final en lugar de al principio. De esta forma, primero se buscarán las ordenes en los directorios del sistema y sólo si no se encuentran se buscarán en el directorio actual.

I.5.6. BLOQUEO DE LA SESION

Dejar la terminal desatendida teniendo una sesión abierta puede ser peligroso porque cualquiera podria utilizar la cuenta durante ese tiempo y borrar o extraer información. También, aprovechando esta circunstancia, otro usuario podría crear un intérprete de órdenes que, posteriormente, le serviría para utilizar la identidad del usuario descuidado lo que lógicamente, sería mucho más grave si le ocurriera al administrador del sistema.

El procedimiento para conseguir esto es sencillo, el saboteador copiaría uno de los interpretes de órdenes o shells (ficheros sh, ksh, csh, etc.) en uno de sus directorios activándole el permiso setuid de la siguiente forma:

copiaría shell a su directorio con otro nombre, por ejemplo:

$ cp /usr/lbin/ksh /users/sabot/prog

si el propietario del shell no es ya **root** lo pondría como tal

$ chown root /users/sabot/prog

activaría el setuid y el permiso de ejecución para otros

$ chmod u+s /users/sabot/prog

Cualquier usuario que ejecute este programa con posterioridad pasará a tener el identificador efectivo (euid) del administrador del sistema lo que implica que tendrán todos sus permisos.

Si se deja la terminal desatendido con una cuenta abierta es recomendable utilizar un programa que lo bloquee, como lock, que bloquea la terminal hasta que se introduce una contraseña, pudiendo en algunos casos combinar esta función con la de salva-pantallas.

Los usuarios que utilizan la **korn** shell (ksh) tienen disponible la variable del entorno TMOUT (time out) mediante la cual es posible especificar un número de segundos de inactividad, transcurridos los cuales la sesión abierta terminará automáticamente. El valor de esta variable no está marcada como de sólo lectura. Por ejemplo:

$ TMOUT = 1800

terminará la sesión actual tras media hora de inactividad.

II.- METODOS DE SEGURIDAD

II.1.- ARCHIVOS DE SEGURIDAD

II.1.1. EL ARCHIVO PASSWD

La seguridad en UNIX está disponible a través de simples comandos que forman bases para un sistema seguro que puede ser adaptado como sea conveniente.

Existe un archivo en UNIX llamado **/etc/passwd** que contiene toda la información que necesita el sistema para conocer a cada usuario, es la primera línea de defensa. Por desgracia también es el punto más débil ya que incluye la contraseña. Estos archivos tienen la particularidad de que pueden imprimirse por alguien en el sistema. La razón por la que el sistema es tan confiable es la siguiente, la contraseña es cifrada usando un código que hace que nadie pueda decifrarlo. Un fragmento típico de **/etc/passwd** se ve con el ejemplo siguiente:

```
$cat /etc/passwd
  root:xyDfccTrt180,m.y8:0:0:admin:/:/bin/sh
  console:101ndT0ee0Map,M.y8:1:1:admin:/:/bin/sh
  pat:xmotTvoyUmjls:10:10:p wood:/usr/pat:/bin/sh
  steve:J9exPd97Ftlbn,M.28:15:10:s kochar:/usr/steve:/bin/sh
  restrict:PomJk109JKy41,.:16:16::/usr/restrict:/bin/rsh
$
```

Tomemos al usuario steve y mediante la siguiente tabla se explicarán los contenidos y valores típicos de este comando.

Ejemplo:

steve:J9exPd97Ftlbn:15:10:s kochar:/usr/steve:/bin/sh
forma general:

username:encrypted passwd:UID:GID:comment:home directory:login shell

NOMBRE DEL CAMPO	DESCRIPCION	VALORES TIPICOS	VALORES EN EL EJEMPLO
Username	identifica al usuario con el sistema. Es en primer término para beneficio humano. Deben ser únicos en una máquina dada y de manera ideal, dentro de una organización	chare rceh brb markd	steve
Encrypted passwd	este campo contiene, o puede contener, la contraseña encriptada.	U7mHuh5R4UmVo X * NOLOGIN	J9exPd97Ftlbn
UID	Esta es la representación numérica del usuario en el sistema.	0-60,000	15
GID	Esta es la representación numérica del grupo login al cual pertenece el usuario.	0-60,000	10
comment	Este contiene la información respecto al usuario. Con frecuencia lista un nombre completo de usuario, número de teléfono u otra información	Chris Hare Ops Manager, x273	s kochar
home directory	Este es el directorio donde se ubica al usuario con base al login.	/u/chare /usr/lib/ppp/ppp-users	/usr/restrict
login shell	Este es el shell de registro que inicia para que los usuarios queden habilitados para interactuar con el sistema. Recuerde, no todos los shells de registro son interactivos	/bin/sh /bin/csh /bin/ksh /bin/tcsh /bin/lib/uucp/uucico /usr/lib/ppp/ppd/	/bin/sh

La tabla anterior proporciona información importante; no es necesario que la contraseña encriptada real se coloque en el campo de contraseña cifrada. Este campo puede contener otros valores. Por ejemplo, para evitar que alguien se registre con una cuenta especifica, esa contraseña puede cambiarse a un valor no repetido, como NOLOGIN. Este campo, sin embargo, contiene por lo general una x o un asterisco, que indica que la contraseña se guarda en otro lado.

En esas situaciones, la contraseña se guarda ya sea en la base computacional confiable (TCB), o en el archivo **shadow passwd,** del que se hablará más adelante.

Los permisos en el archivo **passwd** son de sólo lectura, lo que significa que nadie puede editarlo. De manera similar, el archivo **shadow passwd** y los archivos TCB son, en general, de sólo lectura también.

Cada vez que se entra, la contraseña es cifrada y en el archivo **/etc/passwd** se compara con las registradas, si son iguales, se tiene acceso al sistema; si no, sale un mensaje indicando que el login es incorrecto y que se debe intentar otra vez.

Si se desea cambiar la contraseña, no se puede modificar el archivo **/etc/passwd**, esto no lo permite el sistema, si así fuera tarde o temprano alguien podría entrar y cambiar las contraseñas; como consecuencia nadie podría tener acceso.

Para poder cambiar la contraseña el usuario debe usar el comando **passwd**. Al utilizar este comando el sistema desplegará los mensajes como en el siguiente ejemplo:

```
$ passwd
 changing passwd for pat
Old passwd:cont_vieja
New passwd:Nueva_cont
Re-enter new passwd:Nueva_cont
```

Antes de cambiar una contraseña, el comando **passwd** pide la contraseña (actual y nueva). Esto es para asegurarse que realmente se trata del usuario a la que le pertenece la clave. Si se teclea la contraseña actual equivocada, el sistema responde que no puede cambiar la contraseña actual y que se debe intentar otra vez. Si la contraseña es correcta, el comando **passwd** pide la nueva contraseña. Esta contraseña no se ve en pantalla, el comando hace que se escriba la contraseña por segunda vez. Si las dos nuevas contraseñas no son iguales, el comando **passwd** no realiza la operación y pide la nueva contraseña otra vez:

El comando **passwd** es como muchos otros comandos UNIX, no imprimen nada en pantalla cuando ha sido utilizado con éxito. **passwd** simplemente termina y regresa al shell.

También se usa la orden **passwd** para especificar el número de días mínimo y máximo que una contraseña puede estar funcionando. El envejecimiento evita que un usuario utilice la

misma contraseña durante largos periodos, y cuando se fuerza a cambiarla se evita que el usuario vuelva a ella forzando una duración mínima, por ejemplo:

$ **passwd -x30 -n7 terminal1**

pedirá a terminal1 que cambie su contraseña cada 30 días y que mantenga la contraseña durante al menos una semana.

Al establecer el envejecimiento de contraseñas, hay variables en **/etc/default/passwd** que ponen los valores por defecto de envejecimiento. La orden **passwd** puede usarse para cambiar estos valores por defecto en una base de usuario individual.

- MAXWEEKS= número, donde número es el valor máximo de semanas que una contraseña puede estar en uso.

- MINWEEKS= número, donde número define el valor mínimo de semanas que una contraseña tiene que estar en uso antes de poder ser cambiada.

- WARNWEEKS= número, donde número es la cantidad de semanas antes de que la contraseña expire en que el usuario será avisado.

II.1.2. BLOQUEO DEL ACCESO A UN USUARIO

Se puede bloquear el acceso al sistema a un usuario de diversas maneras. Puede usarse esta orden para cerrar una presentación de forma que se niegue el acceso a un usuario:

$ **passwd -l abc**

Si el usuario abc quiere acceder a su cuenta y sus archivos, el superusuario tendrá que ejecutar **passwd** de nuevo para esta presentación.

Puede limitarse o bloquearse el acceso de un usuario cambiando el shell del usuario. Por ejemplo, la orden:

$ **usermod -s /usr/bin/rsh abc**

Modificará la definición de la presentación del usuario en el sistema y cambiará el shell de abc al shell restringido, lo que limita el acceso del usuario a ciertas órdenes y archivos. Si se pone el shell por defecto a alguna otra orden, como esta, por ejemplo:

$ **usermod -s /bin/true abc**

Entonces abc será desconectado inmediatamente después de cada intento de presentación. UNIX pasará por el proceso de presentación, **exec /bin/true** en lugar del shell, y cuando true finalice, el usuario será desconectado.

II.1.3. ELIMINACION TOTAL DE UN USUARIO

Si no se quiere que un usuario y sus archivos estén más en el sistema, puede usarse la orden userdel:

```
$ userdel -r abc
```

El ejemplo anterior eliminará al usuario abc del sistema y borrará el directorio propio de abc (-r). Una vez que se elimina un usuario, cualquier archivo o propiedad de ese usuario que esté todavía en el sistema será todavía propiedad del número de ID de usuario. Si se realizara un ls -l de esos archivos el ID de usuario se listaría en lugar del nombre de usuario.

II.1.4. ELIMINACION BLANDA DE UN USUARIO

La orden **userdel** elimina a un usuario de los archivos **/etc/passwd** y **/etc/shadow**, y elimina su directorio propio Puede que no se quiera llegar hasta ese extremo. Frecuentemente los usuarios comparten archivos en un proyecto y otros usuarios pueden necesitar recuperar material del directorio de algún usuario eliminado. El siguiente procedimiento es útil, para bloquear cualquier acceso posterior al sistema se debe utilizar lo siguiente:

```
$ passwd -l abc
```

se debe encontrar cualquier otro usuario que esté en el mismo grupo que abc, caso del ejemplo utilizado, y se enviara un correo informándoles que la presentación abc se va a cerrar:

```
$ grep abc /etc/group
abc::568:abc,lsb,oca,gxl
$ mailx lsb oca gxl
Subject. abc login
cc:
se eliminara el directorio propio de abc.
Informar si se tiene necesidad de alguno de los archivos.

El administrador del sistema
```

A continuacion se deben hacer los permisos del directorio propio del usuario 000 de forma que sea inaccesible a todo el mundo.

```
$ chmod 000 /home/abc
```

y después preparar una orden at para eliminar el directorio propio del usuario en un mes.

```
$ at now + 1 month 2 >/dev/console <<%%
rm -r /home/abc
%%
```

II.1.5. EL ARCHIVO SHADOW PASSWD

Las versiones de UNIX que no incluyen las opciones de seguridad avanzada de Secure Ware pueden soportar al archivo **shadow passwd**. Cuando aparece el carácter * en el campo de contraseña, entonces la contraseña real del usuario se guarda en el archivo **shadow passwd**, **/etc/shadow**. A diferencia del archivo **/etc/passwd**, deberá ser legible sólo para root , como se ilustra en el siguiente ejemplo:

```
$ ls -l /etc/shadow
-r-------- 1 root          auth 861 Oct 24 11:46 /etc/shadow
$
```

El formato del archivo **/etc/shadow** es similar al formato del archivo **/etc/passwd**, ya que ambos tienen siete campos delimitados por dos puntos (:), sin embargo, el primero sólo contiene el nombre del usuario, la contraseña encriptada y la contraseña de información caduca, como se muestra en el siguiente ejemplo:

```
$cat /etc/shadow
root:DYQC9rXCioAuo:8887:0:0: :
daemon:*: :0:0: :
ftp:b80lug/921MeY:8842:0:0: :
chare:7xqmj9fj3bVw2:9009: : : :
$
```

El archivo **/etc/shadow** se crea mediante el comando **pwconv**, sólo el superusuario puede crear el archivo **shadow passwd**. La principal ventaja de utilizar el archivo **shadow passwd** es que coloca la contraseña cifrada en un archivo al que no tienen acceso los usuarios normales, así se reducen las posibilidades de que un intruso sea capaz de robar la información de la contraseña cifrada.

II.1.6. EL ARCHIVO DIALUP PASSWD

La capacidad de instalar contraseñas adicionales en líneas de puertos en serie no está presente en todas las versiones de UNIX. Estas se encuentran con más frecuencia ahora en los *sistemas basados en* SCO. La protección con contraseñas de marcación telefónica para estas líneas en serie se controla por medio de dos archivos: **/etc/dialups y /etc/d_passwd.** El primero contiene una lista de los puertos en serie protegidos por una contraseña de marcación telefónica. Ese archivo se ilustra en el siguiente ejemplo:

```
$cat dialups
/dev/tty2A
$
```

El archivo **/etc/d_passwd** se utiliza para identificar el shell de registro que corresponde a una contraseña especifica. A diferencia de la contraseña normal, que emplea el nombre de registro del usuario, la contraseña de marcación telefónica va unida al shell de registro que un usuario determinado tiene uso. Eso significa que cada usuario que utiliza el shell Bourne tiene la misma contraseña de marcación telefónica. El siguiente texto ilustra la contraseña típica de marcación telefónica.

```
$cat /etc/d_passwd
/bin/sh: :
/usr/lib/uucp/uucico
$
```

Cada línea o registro en el archivo consiste de campos delimitados por la repetición de dos puntos. El primer campo identifica el shell que es para la contraseña especificada y el segundo campo lista la contraseña. En el ejemplo anterior, ningún shell tiene una contraseña. Esto significa que no se indicará al usuario que introduzca una contraseña cuando se registre.

La contraseña de marcación telefónica se agrega mediante el uso de la opción -m en el comando **passwd**. Esta opción le informa a **passwd** que la contraseña que se busca es para un shell específico en el archivo de contraseña de marcación telefónica. La sintaxis para esta forma de comando es la siguiente:

```
passwd      -m   shell_name
```

la ejecución de este comando se ilustra en el ejemplo siguiente:

```
$passwd -m /bin/sh:
Setting modem passwd for login shell: /bin/sh
Please enter new passwd:
Modem passwd: testing
Re-enter passwd:testing
$
```

En el ejemplo anterior, el administrador del sistema agrega una contraseña de marcación telefónica al shell **/bin/sh.** Esto significa que a cualquier usuario que se registre en el sistema y tenga el shell Bourne como shell de registro se le indicará que presente la contraseña de marcación telefónica. En el ejemplo también se muestra la contraseña que debe introducirse como lo habría tecleado el administrador del sistema. Igual que el comando **passwd** normal, la contraseña en cuestión no se despliega al momento de escribirse. Aquí se muestra sólo con el propósito de ilustrarla. Observe que sólo el administrador del sistema puede cambiar la contraseña de marcación telefónica para un shell.

El comando **passwd** modifica el contenido del archivo **d_passwd** para incluir la nueva contraseña, como en el siguiente ejemplo:

```
$cat d_passwd
/bin/sh:Ora691.Na1jjQ:
/usr/lib/uucp/uucico: :
$
```

Como se observa en el ejemplo anterior, los usuarios del shell Bourne tienen que proporcionar la contraseña adicional al registrarse en los puertos terminales especificados en el archivo **/etc/dialups.**

El siguiente fragmento ilustra el proceso que ahora realiza un usuario al registrarse con la contraseña de marcación telefónica.

```
gateway
gateway!login: chare
passwd.
Dialup passwd.
Last successful login for chare: Fri Oct 28 22:52:02 EDT 1994 on tty2a
Last unsuccessful login for chare: Tue Oct 18 16:27:56 EDT 1994 on ttyp1
SCO Unix System V/386 Release 3.2
Copyright © 1976-1989 UNIX System Laboratories, Inc.
Copyright © 1980-1989 Microsoft Corporation
Copyright © 1983-1992 The Santa Cruz Operation, Inc.
All Rights Reserved
gateway
TERM = (ansi)
$
```

Como se ilustró, el usuario va a través del procedimiento normal de registro hasta que introduce el nombre de usuario y la contraseña. Después de que éstos han sido validados, se verifica el archivo de control de contraseña de marcación telefónica, **/etc/dialups**. Cuando la

terminal a la cual el usuario está conectado se localiza en el archivo y el shell de registro es un shell Bourne, se le indica al usuario que introduzca la contraseña de marcación telefónica. si esta contraseña se introduce correctamente, al usuario se le permite el acceso al sistema, como se ilustra en el ejemplo anterior. Sin embargo, si la contraseña de ,marcación telefónica es incorrecta, se aborta el registro y el usuario se ve forzado a comenzar de nuevo. Esto se muestra en el siguiente ejemplo:

```
gateway
gateway!login: chare
passwd:
Dialup passwd:
Login incorrect
Wait for login retry: .
Login: chare
Passwd:
Dialup Passwd:
Last successful login for chare: Fri Oct 28 22:52:02 EDT 1994 on tty2a
    1  unsuccessful login for chare: Tue Oct 18 16:27:56 EDT 1994 on tty2a
SCO Unix System V/386 Releasa 3.2
Copyright © 1976-1989 UNIX System Laboratories, Inc.
Copyright © 1980-1989 Microsoft Corporation
Copyright © 1983-1992 The Santa Cruz Operation, Inc.
All Rights Reserved
gateway
TERM = (ansi)
$
```

II.1.7. EL ARCHIVO GROUP

El archivo **/etc/group** se utiliza para formar grupos de trabajo y permitir que puedan compartir información entre ellos libremente. Se debe recordar como funcionan los permisos: si el usuario no es el dueño del archivo, entonces el grupo al cual pertenece se verifica para saber si el usuario es miembro del grupo que posee el archivo. La lista de membresía de grupo se encuentra en **/etc/group**. El formato del archivo se muestra a continuación:

```
tech:*:103:andrewg,patc,chare,erict,lorrainel,lens
group name:passwd:GID:userlist
```

La siguiente tabla explica cada uno de los campos del archivo **/etc/group**

NOMBRE DEL CAMPO	DESCRIPCION	EJEMPLOS
group name	Este es nombre del grupo. Tal nombre se utiliza en primer lugar con fines humanos	tech sales group
passwd	Esta es la contraseña a usar cuando se regresa a este grupo	*
GID	este es el número de ID del grupo numérico en todos los archivos	0-30,000
userlist	Es una lista de usuarios separados por comas que son miembros del grupo	chare, andrewg

La contraseña para el archivo group no se utiliza, y no hay mecanismos sencillos que estén disponibles para instalar una contraseña en el archivo. Si un usuario trata de cambiar hacia un grupo del cual no es miembro, entonces se le indica que introduzca una contraseña, como se ilustra a continuación:

```
$newgrp tech
newgrp: Passwd
newgrp: Sorry
$ grep tech /etc/group
tech:*:103:andrewg, patc
$
```

Si el usuario que ejecutó este comando hubiera sido miembro del grupo tech, entonces el comando **newgrp** habría sido exitoso. Varias versiones actuales de UNIX, sin embargo, permiten al usuario ser un miembro de más de un grupo al mismo tiempo, para reducir o eliminar la necesidad de un comando **newgrp.**

Es importante considerar que UNIX Berkeley extiende la protección de la cuenta root con el grupo wheel. Sólo los usuarios miembros del grupo wheel pueden utilizar el comando su para convertirse en root.

II.2.- ATRIBUTOS DE ARCHIVOS

Si se usa el comando ls con la opción –l, este imprime alguna información cifrada con el tamaño del archivo, la última vez que fue modificado y el nombre del archivo.

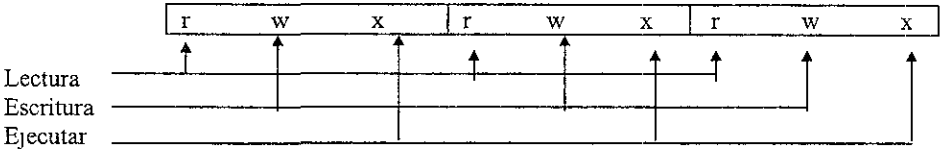
```
$ls –l nombre_de_archivo
 -rwxrwxrwx 1 pat group1 70 jul 28  21:12 nombre_de_archivo
$
```

Omitiendo el -rwxrwxrwx y el número siguiente por ahora, se puede ver que hay 2 palabras antes del tamaño del archivo (el cual es de 70 bytes) esas palabras indican el nombre del propietario y grupo del archivo. Todos los archivos lo tienen. El propietario del archivo es el usuario, usualmente es el que crea el archivo, el grupo es una clave para varios usuarios quienes tienen que ser lógicamente miembros del grupo y tienen que dar un nombre, por ejemplo: mucha gente trabaja en el mismo proyecto y son puestos en un grupo así que ellos pueden tener libre acceso para ver archivos de otros usuarios, mientras restringen el acceso para extraños (por ejemplo, los usuarios que no pertenecen al grupo).

En el ejemplo previo, **pat** es el propietario de nombre_de_archivo, el cual está asociado con el grupo group1. Retomando a –rwxrwxrwx, que esta al principio. Esto significa que es un archivo regular y no un directorio. El guión (-) que está antes de rwxrwxrwx, indica que se editarán los permisos del archivo. Esta sección indica quién puede hacer qué con nombre_de_archivo. En este ejemplo el patrón rwx es repetido 3 veces. Cada una de ellas indica que un tipo particular de usuario puede hacer con el archivo.

El primer rwx nos dice que el propietario de nombre_de_archivo (en este caso pat) puede leer (r), escribir (w), y ejecutar (x) el archivo. El segundo grupo rwx nos dice que algún usuario que sea miembro del grupo group1 puede hacer lo mismo con el archivo. El tercer rwx dice que cualquier usuario puede leer, escribir y ejecutar el archivo.

Si uno de los permisos es denegado, un guión (-) se pone en el lugar de la letra correspondiente. Por ejemplo, rw- significa que se puede leer y escribir, pero no ejecutar; r-x significa que se puede leer y ejecutar pero no escribir.



II.2.1. CAMBIANDO PERMISOS DE ARCHIVOS

Ya que se tiene una manera de reconocer los permisos que se tienen para el acceso de los archivos para cada uno de los tres tipos de usuarios: propietarios, grupo y otros, se debe conocer cómo cambiarlos. por ejemplo, cuando algún usuario es el dueño de algún archivo y no se quiere que otro usuario escriba en sus archivos, por ser un programa valioso, y también tiene que estar disponible para que todos los usuarios lo examinen y lo ejecuten, y además habilitar a los usuarios del grupo y otros a leer (o copiar) y ejecutar el archivo, pero no a modificarlo.

Para alterar el modo del archivo, se debe utilizar el comando **chmod** con el nuevo modo y nombre del archivo como argumentos. El nuevo modo no es especificado para **chmod** como rwxr-xr-x, sino como un dígito de 3 números que es procesado para añadir el equivalente numérico de los permisos.

II.2.2. PROCESANDO NUEVO PERMISO

r	w	x	r	w	x	r	w	x
4			4			4		
	2			2			2	
		1			1			1
	dígito			dígito			dígito	

En este caso, el nuevo modo es 755 donde el 7 es rwx para el propietario, el primer 5 es r-x para el grupo y el segundo 5 es r-x para otros de esta forma queda de la siguiente manera.

r	w	x	r	w	x	r	w	x
4			4			4		
	2			-			-	
		1			1			1
	7			5			5	

Para cambiar el modo de algún archivo a -rwxr-xr-x, se debe usar el comando chmod con el número de 3 dígitos:

```
$chmod 755 nombre_de_archivo
$
```

de esta forma se indicará

```
$ls –l nombre_de_archivo
-rwxr-xr-x 1 pat group1 70 jul 28 21:12 nombre_de_archivo
$
```

El propietario o administrador del sistema son los únicos que pueden cambiar el modo de un archivo particular, esto es, si otros usuarios no pueden acceder al archivo, ellos no pueden usar **chmod** para cambiar atributos.

Existe otra forma de cambiar el modo del archivo con **chmod**. Esto permite utilizar un modo simbólico en lugar de uno numérico. El formato de modo simbólico es:

[**who**] operador permisos

donde **who** es una combinación de letras "u" para los permisos del propietario o usuario, "g" para permisos de grupo, "y" para otros permisos. "a" puede ser usado para especificar todos estos (ugo) y es el que se usa por default. El operador puede ser un + para añadir un permiso, un – para retirar el permiso, o un = para asignar permiso al modo. Permisos es una combinación de letras r,w y x que significan lectura, escritura y ejecución respectivamente.

Así que para cambiar el modo de un archivo a -rw-rw-rw-, se debe escribir:

```
$chmod a=rw nombre_de_archivo
$ls –l nombre_de_archivo
-rw-rw-rw- 1 pat group1 70 jul 28 21:12 nombre_de_archivo
$
```

en este ejemplo la a pudo ser omitida. Para añadir permisos de ejecución se debe teclear

```
$chmod +x nombre_de_archivo
$ls –l nombre_de_archivo
-rwxrwxrwx 1 pat group1 70 jul 28 21:12 nombre_de_archivo
$
```

Finalmente, para remover permisos de escritura de otros, se debe usar o-w

```
$chmod o-w nombre_de_archivo
$ls –l nombre_de_archivo
-rwxrwxr-x 1 pat group1 70 jul 28 21:12 nombre_de_archivo
```

Una de las principales ventajas de esta forma de añadir y quitar permisos es que se puede hacer sin tener que conocer que tan viejos son, por ejemplo cuando se crea un programa shell, los permisos no incluyen el ejecutable. Esto puede ser añadido rápidamente con **chmod** +x sin conocer que permisos de r/w tienen.

Los permisos pueden ser usados para prevenir accidentes de escritura o borrado de un archivo importante. Todo lo que se debe hacer es cambiar el modo a –r-xr-xr-x y sólo podrá alterar el archivo el propietario.

Se puede cambiar el modo para dar permiso de escritura antes de poder alterar el archivo

```
$chmod 555 nombre_de_archivo
```

```
$ls –l nombre_de_archivo
-r-xr-xr-x 1 pat group1 70 jul20 16:57 nombre_de_archivo
$echo hi  there > nombre_de_archivo
nombre_de_archivo:cannot create
$
```

aquí el shell no puede redireccionar la salida del comando echo en el archivo porque no tiene el permiso de escritura.

rm no remueve el archivo inmediatamente. Este pide la confirmación

```
$ls –l nombre_de_archivo
-r-xr-xr-x 1 pat group1 70 mar 18 16:57 nombre_de_archivo

$rm nombre_de_archivo
nombre_de_archivo:555 mode ¿
```

Si se teclea una "Y" cuando rm pide confirmación, el archivo será removido sin tomar en cuenta el modo, alguna otra entrada causa que rm salga sin remover el archivo

```
$rm nombre_de_archivo
nombre_de_archivo:555 mode ¿ n
$
```

II.2.3. CAMBIANDO GRUPO Y PROPIETARIO

Como el propietario de un archivo, se puede cambiar el grupo asociado usando el comando chgrp.

```
$chgrp group2 nombre_de_archivo

$ls –l nombre_de_archivo
-rwxrwxr-x 1 pat group2 70 jul 28 21:12 nombre_de_archivo
$
```

Ahora el grupo 2 tiene permisos de lectura, escritura y ejecución y el grupo 1 no los tiene todos. (los miembros del grupo 1 ahora entran en la categoría otros).

También es posible cambiar el propietario, para esto se utiliza el comando **chown:**

$chown steve nombre_de_archivo

$ls –l nombre_de_archivo
-rwxrwxr-x 1 steve group2 70 jul 28 21:12 nombre_de_archivo

$

Como se puede ver, steve es nuevo propietario de -rwxrwxr-x 1 pat group2 70 jul 28 21:12 nombre_de_archivo, para cambiar el grupo, propietario o permisos, se debe tener el login de steve.

Los directorios se trabajan en forma similar a los archivos ordinarios.

II.2.4. LOS COMANDOS su Y newgrp.

El comando su se usa cuando algún usuario desea obtener el status de superusuario. En un sistema UNIX Multiusuario, el status de superusuario corresponde exclusivamente al programador encargado de la administración y funcionamiento del sistema. Por tanto, dicho status no puede lograrse sin la autorización específica del programador del sistema, que se conoce a través de una palabra clave (**passwd**),

$su usuario
enter passwd:
$

Si el passwd no es correcto para el usuario, el status no será cambiado y el mensaje de la advertencia será desplegado. Si se escribe la contraseña correcta de cualquier modo, se tendrá todos los privilegios asociados con el superusuario y mientras tanto se pierden todos los privilegios asociados con el usuario. Cuando se quiere regresar a ser el usuario, se tiene que presionar ctrl+d o teclear "exit" , esto pone al usuario de regreso a los privilegios que tenia antes de correr su.

Igual que el comando su es el comando **newgrp**, en algunas versiones de UNIX sólo se puede pertenecer a un grupo, así que para cambiar a otros grupos el comando **newgrp** es usado de la siguiente forma:

$newgrp group2
$

esto cambia la estancia del grupo al que se pertenece al grupo 2. Ahora se puede tener acceso a los archivos del grupo2 como si fuera usuario de este grupo. Ahora no se tiene ningún privilegio asociados con el grupo 1.

Por su puesto estar en todos los grupos es absurdo sin un mecanismo para controlarlos. El archivo **/etc/group** contiene una lista de grupos y miembros elegibles.

Como el archivo **/etc/passwd**, la formación es separada por colonias, con el nombre de grupo primero, una contraseña opcional después, sigue un número que identifica el grupo, y una coma separando la lista de los miembros del grupo por ultimo.

Se puede ser listado casi como miembro de un grupo al cambiar al grupo particular sin una contraseña encriptada es dada, por lo tanto ningún usuario puede entrar con **newgrp** al grupo si este no conoce la contraseña. Como se puede ver en el siguiente ejemplo, un grupo, "anyone" tiene una contraseña encriptada. En orden al cambiar a este grupo, **newgrp** pide un usuario para que escriba la contraseña correcta para el grupo.

```
$newgrp anyone
enter passwd:******
$
```

por su puesto si la contraseña no es bien escrita, el cambio no se realiza. A diferencia de su, no se presiona ctrl+d cuando se quiere regresar al grupo anterior, **newgrp** sin un nombre de grupo cambia al grupo por default.

II.2.5. ORDENES REMOTAS

La versión 4 del sistema V de UNIX incorpora las órdenes remotas Berkeley, que fueron originalmente desarrolladas como parte del sistema BSD. Generalmente se las conoce como las ordenes r*, ya que sus nombres comienzan con r, de modo que r* se corresponde con todos sus nombres cuando el * se considera que es el comodín shell.

Podemos utilizar las ordenes remotas para llevar a cabo muchas tareas diferentes sobre máquinas conectadas a nuestra máquina a través de una red con TCP/IP. Las más comúnmente utilizadas de estas órdenes son rcp (copia remota), utilizada para transferir archivos; rsh (shell remoto), utilizada para presentarse en una máquina remota.

Las ordenes remotas permiten utilizar recursos en otras máquinas. Esto permite tratar a una red de computadoras como si fuera una única máquina.

Cuando usuarios remotos tienen permitido el acceso a un sistema, usuarios no autorizados pueden obtener acceso a recursos restringidos. Existen varios modos en que UNIX controla qué usuarios remotos tienen acceso a un sistema.

La seguridad para las ordenes remotas se gestiona a nivel usuario y a nivel anfitrión. A nivel usuario, el administrador de sistema de una máquina remota puede conceder acceso añadiendo una entrada en los archivos de contraseña del sistema. Además el administrador del sistema en la máquina remota puede crear un directorio propio en esa máquina para el usuario.

II.2.6. SEGURIDAD A NIVEL ANFITRION

A nivel anfitrión, cada sistema en la red TCP/IP contiene un archivo llamado /etc/host.equiv, este archivo contiene una lista de las máquinas que son de confianza para ese anfitrión. Los usuarios en máquinas remotas listadas en este archivo pueden abrir sesión remotamente sin suministrar una contraseña.

Por ejemplo si la computadora terminal1 confía en las máquinas remotas terminal2 y terminal4, el archivo /etc/host.equiv en terminal1 será similar a este:

```
$ cat /etc/host.equiv
terminal2
terminal3
```

si el archivo /etc/host.equiv contiene una línea con sólo un signo (+), esta máquina confia en todos los anfitriones remotos.

II.2.7. SEGURIDAD A NIVEL USUARIO

Existe otra facilidad utilizada para forzar la seguridad a nivel usuario. Un usuario que tenga un directorio propio en una máquina remota puede tener un archivo de nombre .rhost en el directorio propio de usuario en esa máquina. Este archivo se utiliza para permitir o denegar acceso al nombre de presentación (login) de ese usuario, dependiendo de en qué máquina y qué usuario esté tratando de obtener acceso a esta presentación. El archivo .rhost define usuarios "equivalentes", a los que se proporcionan los mismos privilegios de acceso.

Una entrada en .rhost es o bien un nombre de máquina, indicando que este usuario es de confianza cuando accede al sistema desde la máquina especificada, o un nombre de máquina seguido de un nombre de presentación, indicando que el nombre de presentación listado es de confianza cuando accede al sistema desde la máquina especificada. Por ejemplo, si khr tiene el siguiente archivo .rhost en /home/khr sobre el sistema local,

```
$ cat .rhost
terminal2
terminal3
terminal4   rrr
```

terminal5 jmf
terminal6
terminal6 rrr

entonces los únicos usuarios de confianza son khr, cuando se presenta desde terminal2, terminal3 o terminal6; rrr, cuando se presenta desde terminal4 o terminal6, y jmf, cuando se presenta desde terminal5.

Cuando la seguridad esta relajada en un sistema, los archivos .rhost son propiedad de los usuarios remotos, para facilitar el acceso. Sin embargo, cuando la seguridad es estricta, root (en la máquina local) será el propietario de todos los archivos .rhost y denegará permiso de escritura a los usuarios remotos.

II.3.- ARCHIVO DE CIFRADO

II.3.1. CRIPTOGRAFIA:

Un buen sistema de control de acceso a los archivos supone la capacidad de hacer que los archivos no tengan utilidad para aquellos que no tienen acceso autorizado. Normalmente esto significa que los datos del archivo se encuentran desordenados o cifrados, utilizando algún tipo de clave de acceso (cifrar viene del griego *kryptos*, que significa clave). Sin esta clave los datos que están desordenados no se pueden ordenar o *descifrar*. La ciencia que se ocupa de los diseños de cifrado y descifrado se denomina *criptografía,* en términos criptográficos, el contenido del texto antes del cifrado se denomina texto abierto, mientras el que está ordenado o codificado se conoce como texto cifrado. Esta es una materia que se remonta a varios siglos, datando el primer tratado europeo del siglo XIV. Este tema tuvo una inmensa importancia histórica durante la segunda guerra mundial. Los esfuerzos concentrados (y en parte exitosos) de los británicos en violentar los códigos utilizados por los alemanes para proteger sus comunicaciones militares tuvieron un factor importante tanto en la victoria en la guerra como en el desarrollo de los primeros sistemas de computación electrónicos.

En nuestros tiempos, con la llegada de las computadoras, especialmente las multiusuario, la seguridad y la existencia de códigos indescifrables se ha convertido en algo muy importante. No sólo es frecuente la necesidad de mantener secretos ciertos archivos, sino que el mismo acceso a la computadora debe ser controlado y regulado. Se han desarrollado numerosos métodos de cifrado de archivos de datos y el algoritmo DES (Estándar de Encriptación de Datos), aceptado por la oficina Nacional de Estándares de Estados Unidos, generalmente se considera seguro contra intentos de descifrarlo. Sin embargo, el DES es muy difícil de implementar y puede no ser adecuado en todas las ocasiones

II.3.2. COMO SE CIFRAN LAS CONTRASEÑAS

En alguna ocasión las contraseñas se guardaban en un formato de texto simple, y sólo el administrador y el software del sistema tenían acceso a tal archivo. Sin embargo, sucedían varios problemas al editar el archivo de contraseñas (**/etc/passwd**). La mayoría de los editores crearon un archivo temporal, el cual es el archivo editado en realidad. En este punto, el archivo sería leído por todos, al facilitar las contraseñas para todas las cuentas.

Como resultado, se desarrollo un método de encriptación de contraseña que utiliza un algoritmo de encriptación de una vía. Así, los valores de encriptación se guardan en lugar del texto. Sin embargo, la seguridad del sistema es sólo tan buena como el método de encriptación elegido.

Cuando un usuario se registra en un sistema UNIX, el programa getty le pide al usuario su nombre_de_usuario y luego ejecuta el programa de registro, indicando la contraseña pero sin descifrarla. En realidad, el programa de registro descifra la contraseña y luego compara el valor reciente con el que está guardado en **/etc/passwd**. Si coincide, entonces el usuario proporcionó el valor correcto.

El método de cifrado de UNIX para las contraseñas se introduce por medio de un mecanismo de kernel nombrado crypt(3). Gracias a los asuntos de licencias federales de Estados Unidos, las rutinas **crypt** quizá no están disponibles en su computadora. La razón es que mientras que estas rutinas que se necesitan para cifrar información están disponibles, las rutinas para descifrar no se encuentran fuera de Estados Unidos.

El valor de la contraseña real guardado en **/etc/passwd** es el resultado de emplear la contraseña del usuario para cifrar un grupo de 64 bits de ceros al utilizar la llamada crypt(3). Clear text es la contraseña del usuario, la cual es la clave para la operación de encriptación. El texto a ser cifrado es de 64 bits de ceros y el texto cifrado resultante es la contraseña cifrada.

El algoritmo crypt(3) se basa en el estándar de encriptación de datos (DES) desarrollado por el instituto Nacional de Estándares y Tecnología o NIST. En operación normal de acuerdo con el DES, una clave de 56 bits, como ocho caracteres de siete bits, se utiliza para cifrar el texto original, el cual es llamado texto llano, este es por lo general, de 64 bits de largo. El texto cifrado resultante no puede descifrarse con facilidad sin saber la clave original.

La llamada crypt(3) de UNIX utiliza una versión modificada de este método, al establecer que el texto llano se cifra en un grupo de ceros. El proceso es complicado al tomar el texto cifrado resultante y descifrandolo de nuevo con la contraseña del usuario como clave, este proceso se realiza 25 veces, cuando finaliza, los 64 bits resultantes se dividen en 11 caracteres y luego se guardan en el archivo de contraseña.

A pesar del hecho de que la fuente para **crypt** puede obtenerse de varios vendedores, aunque su distribución comercial se encuentra limitada fuera de E.U., no hay método conocido disponible para traducir el texto cifrado o el valor cifrado del regreso a su texto llano original.

Robert Morris, padre y Ken Thompson, quienes implantaron la tecnología crypt(3) en UNIX en sus inicios, temían que con el advenimiento del hardware de chips DES, la seguridad del sistema UNIX sería traspasada con facilidad. Mediante el uso de un "grano de sal", se las arreglaron para evitar esta amenaza.

El "grano de sal", es el número de 12 bits que se utiliza para modificar el resultado de la función DES. El valor de este número de 12 bits va de cero a 4095. Así que para cada contraseña posible, existen 4096 formas de encriptación y almacenamiento. Es posible para varios usuarios en la misma máquina utilizar la misma contraseña.

Cuando un usuario ejecuta el programa /bin/passwd para establecer una nueva contraseña, el programa /bin/passwd elige una sal basada en la hora del día. Esta sal se utiliza para modificar la contraseña del usuario.

El problema viene después, al cifrar la contraseña cuando el usuario se registra. Es posible, pero no probable, que el usuario se registre y la sal sea la misma. Para que las cosas marchen bien, UNIX también guarda la sal en **/etc/passwd**. En realidad, crea los dos primeros caracteres de la contraseña cifrada. A continuación se muestra un valor cifrado.

2eLNss48eJ/GY

En este ejemplo, los dos caracteres iniciales (2e) son la sal para esta contraseña. Cuando el usuario se registra en el sistema, el programa de registro extrae la sal de la contraseña guardada y la utiliza para cifrar la contraseña que proporciona el usuario. Si la nueva contraseña cifrada y la almacenada son iguales, entonces la contraseña introducida por el usuario es correcta y estará registrado en el sistema. Si los valores no se igualan, entonces el usuario observará el mensaje login incorrect, lo que indica que deberá intentarlo otra vez.

II.3.3. COMO CIFRAR ARCHIVOS

Como se ha observado, la encriptación de contraseñas mediante el uso de un mecanismo que es difícil de descifrar, ofrece un método relativamente seguro para evitar que usuarios no autorizados tengan acceso al sistema. Pero para la encriptación de archivos se utiliza un método de encriptación poco seguro, esto es por medio del comando crypt(1). Es interesante que algunos comandos UNIX soporten la manipulación directa de estos archivos cifrados sin tener que descifrarlos primero.

Cifrar archivos con el comando **crypt** es bastante simple. Si no se proporcionan argumentos en la línea de comando, entonces **crypt** indica la clave, lee los datos a cifrar de una

entrada estándar e imprime la información cifrada en la salida estándar. Sin embargo, idealmente, la información a usar se proporciona en la línea de comando, como en el siguiente ejemplo:

crypt key <clear> cipher

el comando anterior lee el archivo clear, cifra el texto con la clave de contraseña y guarda el texto cifrado resultante en el archivo cipher. Los archivos cifrados pueden observarse o descifrarse mediante el uso de una línea de comando similar, como se muestra a continuación:

crypt key <cipher> clear
crypt key <cipher> | pr | lp

en el primer comando del ejemplo anterior, el texto cifrado en cipher se descifra mediante el uso de key y se guarda en el archivo llamado clear. El segundo ejemplo de línea de comando utiliza **crypt** para descifrar el texto y enviar el resultado de texto llano a pr para ser formateado y luego a lp para impresión. Los archivos generados por **crypt** pueden ser editados por ed o vi, tomando en cuanta que la versión de estos editores que tiene en el sistema soporta la edición de archivos cifrados.

El mecanismo exacto que utiliza **crypt** está documentado, y varias versiones de este comando están disponibles en internet. Crypt(1) no usa las mismos métodos de encriptación que crypt(3), el cual se emplea para la encriptación de contraseñas. El mecanismo es una máquina de 256 elementos.

La clave de encriptación que se utiliza es el factor limitante en la determinación del nivel de esfuerzo para descifrar los datos. Mientras más larga es la contraseña, más complejos son los patrones de encriptación y más tiempo toma transformar la clave a los arreglos internos utilizados por la máquina. Por ejemplo, el proceso de transformación es para que tome cerca de un segundo, pero si la clave se restringe a tres letras minúsculas, los archivos pueden ser leídos en sólo una fracción sustancial de cinco minutos del tiempo real de la máquina.

Como la clave de **crypt** podría ser vista por otro usuario que utiliza **ps**, **crypt** destruye cualquier rastro de la llave real luego de hacer la entrada. En consecuencia, como cualquier otro sistema de seguridad, la contraseña empleada para cifrar los datos es el componente más crítico y el más sospechoso.

II.4.- SISTEMA KERBEROS

Muchos sistemas pueden ser modificados para que utilicen el mecanismo de autenticación kerberos. Kerberos, nombre del perro que en la mitología griega se dice guardaba las puertas del Hades, es una conexión de software que se emplea en una red grande para establecer la identidad

declarada de un usuario, utiliza una combinación de encriptación y bases de datos distribuidas de tal forma que un usuario en el campus universitario puede registrarse y empezar una sesión desde cualquier computadora localizada en ese campus.

Kerberos es un sistema de autenticación que fue desarrollado por el proyecto athena del instituto Tecnológico de Massachusetts. Desde entonces, kerberos ha sido adoptado por otras organizaciones para sus propias necesidades. Muchos desarrolladores de aplicaciones de otras firmas incluyen el soporte para la autenticación del sistema kerberos en sus productos.

Este sistema valida la identidad de un principal (puede ser un usuario o un servicio). En cualquier caso, el principio se define por cualquiera de los componentes siguientes:

- Nombre primario.
- Instancia
- Reino.

En la terminología kerberos, a esto se le llama un trío y se ilustra a continuación:

<nombre primario, instancia, reino>

Donde nombre primario, en el caso de una persona genuina, es el identificador de registro. La instancia es o nula o contiene información particular respecto al usuario. Para un servicio, el nombre primario es el nombre del servicio y el nombre de la máquina se utiliza como la instancia, como en rlogin.mimáquina. en cualquier caso , el reino se emplea para distinguir entre diferentes dominios de autenticación. Por medio del reino, es posible tener un servidor kerberos distinto para cada unidad pequeña dentro de una organización en lugar de uno grande. Esta situación sería un objetivo fácil para los intrusos, porque tendría que ser confiable de manera universal a toda la organización. En consecuencia, ésta no es la mejor opción para configurar.

Los principales kerberos obtienen boletos para servicios de un servidor especial conocido como servidor despachador de boletos. Cada boleto consiste en información diversa que identifica al principal que esta cifrado en la clave privada para ese servicio. Puesto que sólo kerberos y el servicio conocen esta clave, se considera auténtica. El boleto otorgado por el servidor despachador de boletos contiene una nueva clave de sesión.

La principal ventaja con el sistema kerberos es que cada boleto tiene un tiempo de vida específico Despues de que dicho tiempo termina, debe solicitarse un nuevo boleto, el cual será emitido por el servidor despachador de boletos.

Como se menciono antes, el sistema fue desarrollado originalmente por el MIT durante el desarrollo del proyecto athena La desventaja aquí es que la configuración del ambiente en el MIT es diferente a otra organización. En otras palabras, kerberos fue diseñado para autenticar al usuario final para determinados servidores.

Varios asuntos importantes conciernen al sistema de autenticación kerberos. El primero, y más importante, la mayoría de los sistema de cómputo no tienen un área segura en donde guardar las claves. Puesto que la clave de un texto llano debe ser guardada en el diálogo inicial para obtener un boleto del despachador, deberá existir un lugar seguro para guardar esa información. En el caso de que un usuario no autorizado obtenga esta clave de texto llano, el servidor de autenticación kerberos en ese reino puede comprometerse con facilidad.

Otros de los problemas es cómo maneja kerberos las claves en las computadoras de multiusuarios. En este caso, las claves en la memoria pueden ser obtenidas por otros usuarios registrados en el sistema. En un medio de estación de trabajo soporta varios usuarios, entonces es posible para otro usuario del sistema obtener las claves.

III.- POLITICAS DE SEGURIDAD

III.1.- PLANEACION DE LA SEGURIDAD DE LA RED

La política de seguridad es una forma de protección para una inversión y recursos de información. La mayoría de los sistemas tienen información que debe ser protegida contra el vandalismo del mismo modo que otros bienes valiosos.

La mayoría de diseñadores de redes, por lo general, empiezan con la implantación de soluciones de barreras de protección antes que ningún otro problema de seguridad haya sido identificado en forma acertada. Tal vez una razón para esto sea que establecer una política de red efectiva, puede tener consecuencias indeseables; los usuarios de red podrán encontrar formas de ignorar su política de red, convirtiéndola en algo inútil.

Una política de seguridad de red efectiva es algo que todos los usuarios de la red y administradores pueden aceptar, y están dispuestos a reforzar.

Una organización puede tener muchos sitios y cada uno contar con sus propias redes. Si la organización es grande, es probable que los sitios tengan diferentes administradores de red, con diferentes metas y objetivos. Si estos sitios no están conectados por medio de una red interna, cada uno de ellos podrá tener sus propias políticas de seguridad de red. Sin embargo, si los sitios están conectados por una red interna, la política de red deberá agrupar las metas de todos los sitios que están interconectados.

En general, un sitio es cualquier parte de una organización que posee computadoras y recursos relacionados con la red. Dichos recursos incluyen, pero no se limitan a lo siguiente:

- Estaciones de trabajo.
- Computadoras anfitrión y servidores.
- Dispositivos de interconexión: compuertas, enrutadores, puentes, repetidoras.
- Servidores de terminal.
- Software para red y aplicaciones.
- Cables de red.
- Información en archivos y bases de datos.

La política de seguridad del sitio debe tomar en cuenta la protección de todos sus recursos Puesto que el sitio estará conectado con otras redes, la política de seguridad del sitio debe considerar las necesidades y requerimientos de seguridad de todas las redes interconectadas. Esto es un punto importante, por que se puede lograr una política de seguridad de red que salvaguarde sus intereses pero que puede ser perjudicial para otros. Un ejemplo de esto podría ser un uso deliberado de direcciones IP, detrás de la compuerta de la barrera de protección, que ya sean utilizadas por alguien más. En este caso, los ataques en contra de la red mediante la imitación de las direcciones IP de la red serán desviadas hacia la organización de la cual se emplean las direcciones IP. Esta situación debería evitarse, porque va contra los intereses de ser un "buen ciudadano" en la red.

III.1.1. RESPONSABILIDAD DE UNA POLITICA DE SEGURIDAD

Un aspecto importante de la política de seguridad de red es asegurar que todos saben cuál es su responsabilidad para mantener la seguridad. Es difícil para una política de seguridad de red anticipar todas las amenazas posibles. La política puede, sin embargo, garantizar que cada tipo de problema tiene alguien que puede manejarlo de manera responsable. Asimismo, pueden existir varios niveles de responsabilidad asociados con una política de seguridad de red. Cada usuario de la red, por ejemplo, deberá ser responsable de guardar su contraseña. Un usuario que pone en riesgo su cuenta aumenta la probabilidad de comprometer otras cuentas y recursos. Por otro lado, los administradores de red y de sistema son responsables de mantener la seguridad general de la red.

III.1.2. PLANTEAMIENTO DE LA POLITICA DE SEGURIDAD

Definir una política de seguridad de red significa desarrollar procedimientos y planes que salvaguarden los recursos de la red contra pérdidas y daños. Un planteamiento posible para desarrollar esta política es el análisis de lo siguiente:

- ¿Qué recursos se quieren proteger?
- ¿De qué personas se protegerán los recursos?
- ¿Qué tan reales son las amenazas?
- ¿Qué tan importante es el recurso?
- ¿Qué medidas se pueden implantar para proteger los bienes de una manera económica y oportuna?

- Se debe examinar con frecuencia la política de seguridad de red para verificar si los objetivos y circunstancias en la red han cambiado.

La siguiente figura muestra una hoja que se puede usar para ordenar las ideas respecto a estas preguntas:

RECURSOS DE LA RED			Tipo de usuarios indeseables	Posibilidad de una amenaza	Medidas a implantar para proteger los recursos de red.
Número	Nombre	Importancia del recurso			

La columna "Número de recursos de la red" es una número de red de identificación interna de los recursos (si es que se aplica).

La columna "Nombre de los recursos de la red" es una descripción de los recursos. La importancia del recurso puede estar en una escala numérica del cero al diez, o en expresiones "confusas" de lenguaje nativo como bajo, alto, medio, muy alto, etc.

La columna "Tipo de usuarios indeseables" puede tener calificativos como interno, externo, huésped, o nombres de grupo como usuarios de cuenta, asistentes corporativos, etc.

La columna "Posibilidad de una amenaza" puede estar en una escala numérica de cero a diez, o en expresiones similares a la columna "Nombre de los recursos de la red".

La columna "Medidas a implantar para proteger recursos de red" pueden tener valores como "Permisos del sistema operativo" para archivos y directorios; "pistas/alertas de auditoria"

para servicios de red;" enrutadores de selección" y "barreras de protección" para anfitriones y dispositivos de red; o cualquier otra descripción del tipo de control de seguridad.

En la práctica, el costo de proteger las redes de una amenaza debe ser menor que el costo de la recuperación, si es que se ve afectado por la amenaza de seguridad. Es importante involucrar al tipo adecuado de personas en el diseño de la política de seguridad de red. Técnicos y administrativos, los primeros para sugerir las medidas a implementarse y los segundos para avalarlas, estas medidas deben ser tanto preventivas como correctivas.

III.2.- ANALISIS DE RIESGOS

Al crear una política de red, se deben identificar los recursos que se protegerán, tener en cuenta que los recursos invertidos en la seguridad son costeables. Esto significa que se debe identificar que recursos queremos proteger y también identificar cuales son más importantes que otros.

El análisis de riesgo implica determinar lo siguiente:

- Qué necesita proteger.
- De quién debe protegerlo.
- Cómo protegerlo.

Los riesgos se clasifican por el nivel de importancia y por la severidad de la pérdida. En el análisis de riesgos es necesario determinar los siguientes factores:

1.- Estimación del riesgo de pérdida del recurso (Ri).

2.- Estimación de la importancia del recurso (Wi).

Como un paso hacia la cuantificación del riesgo de perder un recurso, es posible asignar un valor numérico. Por ejemplo, al riesgo (Ri) de perder un recurso se le asigna un valor de cero a diez, donde cero indica que no hay riesgo y diez es la importancia más alta. La evaluación general del riesgo será entonces el producto numérico del valor del riesgo y su importancia (también, llamado peso). Esto puede escribirse como sigue:

$WRi = Ri * Wi$

Wri = Peso del riesgo del recurso "i"
Ri = Riesgo del recurso "i"
Wi = Importancia de recurso "i"

La siguiente figura muestra una hoja de trabajo que se puede utilizar para registrar los cálculos previos.

Recursos de red		Riesgos para los recursos de red (Ri)	Peso (importancia) del recurso (Wi)	Riesgo evaluado (Ri*Wi)
Número	Nombre			

La columna "Número de recursos en red" es un número para identificación interna del recurso en la red (si aplica).

La columna "Nombre de los recursos de red" es una descripción de los recursos.

La columna "Riesgo para los recursos de red (Ri)" puede estar en una escala numérica del cero al diez, o en expresiones como bajo, alto, muy alto, etc.

De manera similar, la columna "Importancia del recurso (Wi)" debe estar en una escala numérica del cero al diez, o en expresiones similares a la columna anterior. Si se utilizan valores numéricos para las columnas de riesgo y peso, se puede calcular el valor en la columna "riesgo evaluado (Ri*Wi)" como el producto de los valores de riesgo y peso.

Con la siguiente fórmula es posible calcular el riesgo general de los recursos de la red:

$$WR=(R1*W1 + R2*W2 + ...+Rn*Wn)/(W1 + W2 + ... + Wn)$$

Otros factores que se deben considerar para el análisis del riesgo de un recurso de red son su disponibilidad, su integridad y su carácter confidencial. La disponibilidad de un recurso es la medida de qué tan importante es tenerlo disponible todo el tiempo. La integridad de un recurso es la medida de qué tan importante es que éste o los datos del mismo sean consistentes. Esto es de particular trascendencia para los recursos de bases de datos. El hecho de ser confidenciales se aplica a los recursos, como archivos de datos, a los cuales se desea restringir el acceso.

III.2.1. RECURSOS BAJO RIESGO EN UN SISTEMA:

Al realizar un análisis de riesgo, debe identificar todos los recursos cuya seguridad está en riesgo de ser quebrantada.

El RFC 1244 lista los recursos de red que deben ser considerados al estimar las amenazas a la seguridad general:

1.- **Hardware:**

- Procesadores.
- Tarjetas.
- Teclados.
- Terminales.
- Estaciones de trabajo.
- Impresoras.
- Unidades de disco.
- Líneas de comunicación.
- Servidores de terminal.
- Enrutadores.

2.- **Software:**

- Programas fuente.
- Programas objeto.
- Utilerías.
- Programas de diagnóstico.
- Sistemas operativos.
- Programas de comunicación.

3.- **Datos:**

- Durante la ejecución.
- Almacenados en línea.
- Archivos fuera de línea.
- Apoyos.
- Bitácora de auditoría.
- Bases de datos.
- En tránsito sobre medios de comunicación.

4.- **Gente:**

- Usuarios.
- Personas para operar sistemas.

5.- **Documentación:**

- Sobre programas.
- Hardware.
- Sistemas.
- Procedimientos administrativos locales.

6.- **Accesorios:**

- Papel.
- Formas.
- Cintas.
- Información grabada.

III.2.2. ACCESO NO AUTORIZADO

Sólo se permite el acceso a los recursos de red a usuarios. A esto se le llama acceso autorizado. Una amenaza común es el acceso no autorizado a los recursos de cómputo, el cual puede ser de diversas formas, como utilizar la cuenta de otro usuario para obtener acceso a la red y sus recursos. En general, el uso de cualquier recurso de red sin el permiso previo se considera como acceso no autorizado.

La gravedad de un acceso no autorizado depende del sitio y la naturaleza de la pérdida potencial. Para algunos sitios, el sólo hecho de permitir acceso a un usuario no autorizado podrá causar daños irreparables, ante la falta de seguridad para cubrir los medios.

III.3.- RESGUARDO DE INFORMACION.

La divulgación de información, ya sea voluntaria o involuntaria, es otro tipo de amenaza. Se deberá determinar el valor o sensibilidad de la información guardada en sus computadoras. En el caso de vendedores de hardware y software, código de fuente, detalles de diseño, diagramas e información específica de producto representa un hito competitivo. Los hospitales, compañías de seguros e instituciones financieras mantienen información confidencial, su divulgación podría ser dañina a sus clientes y a la reputación de la compañía. Por otro lado, los laboratorios farmacéuticos pueden guardar aplicaciones de patente y no pueden arriesgarse

III.3.1. USO CORRECTO DE UN RECURSO

Cuando se han determinado, a cuales usuarios se les permite ingresar a los recursos de la red, se deberá proveer guías para el uso aceptable de estos recursos. Las guías dependerán de la clase de usuario, como desarrolladores de software, estudiantes, programadores, usuarios externos, etc. Para cada uno de estos usuarios se requieren guías diferentes. La política debe establecer qué tipos de uso de red es aceptable e inaceptable, y que tipo de uso será restringido. La política que se desarrolle se llamará Política de Uso Aceptable (AUP) para la red. Si el acceso a un recurso de red se restringe, se deberá considerar el nivel de acceso que tendrán las diferentes clases de usuarios.

Aunque pueda parecer obvio, la AUP deberá decir con claridad que no se permite irrumpir en las cuentas o ignorar la seguridad. Esto puede ayudar a evitar aspectos legales concernientes a empleados que ignoren la seguridad de la red y después digan que no se les informó o entrenó acerca de la política de red. Esta es una lista de aspectos que es necesario considerar al desarrollar una AUP:

- ¿Se permite introducir a las cuentas?
- ¿Se permite quebrantar las contraseñas?
- ¿Esta permitido interrumpir servicios?
- ¿Deberán los usuarios suponer que un archivo de lectura mundial les otorga la autorización a leerlo?
- ¿Podrán los usuarios compartir cuentas?

Sólo si las necesidades son inusuales, la respuesta a casi todas estas preguntas para la mayoría de las organizaciones deberá ser no.

Además, tal vez se desee incorporar en las políticas un enunciado concerniente al software con derechos de autor y con licencia. En general, los procedimientos de uso de la red deberán ser de una manera que dificulten a los usuarios copiar software no autorizado de la red. Copiar software ilegalmente es un acto castigado por la ley en la mayoría de los países. Las grandes organizaciones con frecuencia tienen políticas muy estrictas con relación a las licencias, dado el riesgo de problemas legales y al daño causado por la publicidad de estos incidentes.

Es importante incluir información con respecto al software con derechos de autor y/o licencia en la AUP. Los siguientes son algunos ejemplos de los puntos que se deberían mencionar:

- El software con derechos de autor y licencia no podrá ser duplicado, salvo que se acuerde de manera explícita.

- Indicar métodos que contengan información acerca del estado de derechos de autor y licencia del software.

- Si existen dudas sobre las copias del software, no se debe hacer.

Una AUP que no dice con claridad lo que está prohibido, sólo dificulta el poder comprobar que un usuario ha violado la política. Las exenciones a la política podrán ser miembros de los llamados equipos tigre, que tienen la responsabilidad de sondear los puntos débiles en la seguridad de la red. Los usuarios que conforman estos equipos deberán identificarse con claridad. Habrá ocasiones en que algunos usuarios se autonombran miembros del equipo tigre y desean sondear las debilidades de seguridad con propósitos de investigación o para demostrar algo. La AUP deberá mencionar los siguientes aspectos acerca de explorar la seguridad:

- ¿Se permite de alguna forma el vandalismo a nivel usuario?
- ¿Qué tipo de actividades de sondeo de seguridad se permite?
- ¿Qué controles deben emplazarse para asegurar que el sondeo de seguridad no se sale de control?
- ¿Qué controles deben emplazarse para evitar que otros usuarios de la red sean víctimas de las actividades de sondeo de seguridad?
- ¿Quién deberá tener el permiso para realizar sondeos de seguridad, y cuál es el proceso para la obtención del permiso para realizar esas pruebas?

Si se desea permitir el sondeo de seguridad legítimo, deberá tener segmentos de red separados y anfitriones en su red para esas pruebas. En general, es muy peligroso probar worms y virus. Si se tiene que realizar estas pruebas, sería ingenuo hacerlas en una red viva. Se deberá, en cambio, aislar de manera física los anfitriones y segmentos de red que se utilicen para la prueba, y hacer una recarga completa y cuidadosa de todo el software al final de cada prueba.

Evaluar las debilidades de seguridad y tomar medidas correctas puede ser efectivo para evitar ataques vandálicos. Algunas organizaciones utilizan consultores externos para evaluar la seguridad de sus servicios. Como parte de esta evaluación, podrán realizar un "vandalismo" legítimo. Su política deberá permitir este tipo de situaciones.

III.3.2. OTORGAR ACCESO Y APROBAR EL USO DE LA RED

La política de seguridad de la red deberá identificar a quién esté autorizado a otorgar el acceso a los servicios. Se deberá también determinar qué tipo de acceso podrán otorgar estos individuos. Si no se puede controlar a quién se otorga el acceso al sistema, será difícil controlar quién estará utilizando la red. Si se puede identificar a las personas a cargo de otorgar el acceso a la red, se podrá averiguar qué tipo de acceso o control ha sido otorgado. Esto es útil en la identificación de la causa de las lagunas de seguridad como resultado del exceso de privilegios brindado a los usuarios.

Para determinar quién otorgará el acceso a los servicios de una red se deben considerar los siguientes factores:

- ¿Se permitirá el acceso a los servicios desde un punto central?
- ¿Que métodos se utilizarán para crear cuentas y finalizar el acceso?

Si la red es grande y descentralizada, es posible tener varios puntos centrales, uno para cada departamento, que es responsable por la seguridad de la red departamental. En este caso se deberán tener guías globales de qué tipos de servicios se permiten para cada clase de usuario. En general, mientras más centralizada es la administración de la red, más fácil es mantener la seguridad. Por otro lado, las administraciones centralizadas pueden generar problemas cuando los departamentos desean un mayor control sobre sus recursos de red. La cantidad correcta de centralización o descentralización dependerá de factores más allá del ámbito de este análisis.

Los administradores de sistemas requieren de acceso especial a la red, pero algunos usuarios tal vez necesitarán ciertos privilegios también. La política de seguridad de la red deberá mencionar este aspecto Una política universal que restrinja todos los privilegios especiales, aunque sea segura, evitará que los usuarios legítimos hagan su trabajo. Por tanto, es necesario un enfoque más balanceado.

En la práctica algunos administradores de sistemas toman el camino más sencillo y asignan más privilegio de los que necesita el usuario, para que el usuario no se queje por lo mismo. También el administrador del sistema deberá entender con propiedad los puntos finos de asignar seguridad y no se deberá equivocar al asignar más privilegios de los indcados.

Las personas que tengan privilegios especiales deberán ser responsables, además de tener cierta personalidad legal de la autoridad identificada dentro de la política de seguridad. Algunos sistemas podrán tener mecanismos de auditorías que pueden utilizarse para que los usuarios privilegiados no abusen de esa confianza.

Si hay un gran numero de administradores de sistemas y de red, es difícil mantener la pista de los permisos que han sido otorgados para los recursos de la red. Hay una manera formal de otorgar pedidos Despues de que el usuario hace el pedido y éste es autorizado por el supervisor del usuario, el administrador del sistema deberá registrar y documentar las restricciones de seguridad o acceso que se le otorgaron al usuario.

La siguiente figura muestra una hoja de trabajo que se puede utilizar para mantener una pista en papel de los permisos otorgados en la red.

RECURSOS DE RED		TIPO DE ACCESO	PERMISO DE UNIX
NUMERO	NOMBRE		

La siguiente es una descripción de las columnas utilizadas en esta hoja de trabajo.

- La columna del número de recursos de red es un número de identificación interna para cada recurso de la red.
- la columna de nombre de recurso es una descripción de los recursos.
- la columna de tipo de acceso puede utilizarse para una descripción del recurso, como "leer y ejecutar acceso al directorio".
- la columna de permisos para el sistema operativo contiene las banderas del sistema operativo utilizado para implantar el acceso de seguridad. Para los sistemas UNIX, estas señales son leer (r), escribir (w) y ejecutar (x).

También se deberá examinar el procedimiento que se habrá de usar para crear cuentas de usuario y asignar permisos. En el caso menos restrictivo, la gente autorizada para otorgar acceso deberá ser capaz de entrar al sistema de manera directa y crear una cuenta a mano o por medio de mecanismos suplidos por el proveedor. Estos mecanismos enfatizan la confianza en la persona que los ejecuta, y esta persona por lo general tiene una gran cantidad de privilegios, como el usuario root en UNIX. Bajo esas circunstancias, necesita seleccionar a alguien muy confiable para realizar esa tarea.

Se deberá desarrollar procedimientos específicos para la creación de cuentas. Bajo UNIX, hay varios métodos que se pueden utilizar para crear cuentas. Sin importar por cual procedimiento se emplee, es mejor que se este bien documentado para evitar confusiones y reducir posibles errores.

Otro aspecto a considerar es la selección de un procedimiento de creación de cuenta de usuario que es más simple y fácil de entender. Esto asegura que se cometerán menos errores, y que los administradores del sistema serán más propensos a cumplir (como debería suceder regularmente).

También se debe tener una política en la selección de una contraseña inicial. Al otorgar una contraseña inicial es un momento donde la cuenta del usuario se encuentra vulnerable. Las

políticas como aquellas donde la contraseña inicial sea la misma que el nombre del usuario o que se deje en blanco, pueden dejar al descubierto las cuentas. También lo sería evitar establecer la contraseña inicial como una función del nombre de usuario, o parte del nombre de usuario, o alguna contraseña generada en forma de algoritmo que pueda adivinarse con facilidad. La selección de una contraseña inicial no debería ser obvia.

Algunos usuarios utilizan su cuenta hasta mucho tiempo después de creada; otros nunca se registran, en estas circunstancias, si la contraseña inicial no es segura, la cuenta y el sistema son vulnerables. Por tal razón, se deberá tener una política para inhabilitar las cuentas que nunca se han introducido durante cierto tiempo. Con esto el usuario es forzado a pedir que se habilite su cuenta de usuario.

También es un error permitir que los usuarios continúen utilizando la contraseña inicial por tiempo indefinido. Si el sistema lo permite, se deberá forzar a los usuarios a cambiar las contraseñas en el primer registro. Muchos sistemas tienen una política de caducidad de contraseña. Esto puede ser útil para proteger las contraseñas. También hay utilerías UNIX como password+ y npasswd que pueden emplearse para probar la seguridad de la contraseña.

III.3.3. DETERMINAR RESPONSABILIDADES DEL USUARIO

La política de seguridad en la red debe definir los derechos y las responsabilidades de los usuarios al utilizar los recursos y servicios de la red. La siguiente es una lista de aspectos que se deben mencionar con respecto a las responsabilidades del usuario:

- Guías respecto al uso de recurso de red en caso de que los usuarios estén restringidos y cuáles son las restricciones.
- Lo que constituye abuso en los términos del uso de los recursos de red que afectan el desempeño del sistema y la red.
- ¿Podrán los usuarios compartir sus cuentas o permitir a otros utilizar sus cuentas?
- ¿Deberían los usuarios revelar sus contraseñas de manera temporal para permitir a quienes trabajan en un proyecto el acceso a sus cuentas?
- Política de contraseña del usuario, *¿Con qué frecuencia deberían cambiar sus contraseñas* los usuarios y cualesquiera otras restricciones de contraseñas o requerimientos?
- ¿Son responsables los usuarios de brindar respaldos de sus datos o es responsabilidad del sistema?

- Consecuencias para los usuarios que divulgan información que podría ser propietaria. ¿Qué acciones legales u otro castigo podrían implementarse?.

- Una declaración sobre privacidad de correo electrónico (acta de privacidad en las comunicaciones electrónicas).

- Una política respecto a correo controversial o adicciones a las listas de correo o grupos de discusión.

- Una política sobre comunicaciones electrónicas como falsificación de correo.

La asociación de correo electrónico (EMA) recomienda que cada sitio tenga una política sobre la protección a la privacidad del empleado. Las organizaciones deberían establecer políticas de privacidad que no se limiten al correo electrónico, sino que abarquen otros medios como discos duros, cintas y documentos en papel, EMA sugiere cinco criterios para evaluar cualquier política:

1 - ¿Cumple la política con la ley y con los deberes a otras compañías?

2.- ¿Compromete la política de manera innecesaria los intereses del usuario o de otras personas?

3.- ¿La política es aplicable y viable para ser cumplida?

4.- ¿Abarca la política de manera apropiada todas las formas de comunicación y archivo de registro en la oficina?

5.- ¿Se ha anunciado la política por adelantado y están de acuerdo todos los implicados?

III.3.4. DETERMINAR RESPONSABILIDADES DE LOS ADMINISTRADORES DEL SISTEMA

El administrador del sistema con frecuencia necesita recabar información de los archivos en los directorios privados de los usuarios para diagnosticar problemas del sistema. Los usuarios, por otro lado, tienen el derecho de mantener su privacidad. Hay entonces, un intercambio entre los derechos de un usuario a la privacidad y las necesidades de un administrador del sistema. Cuando ocurren las amenazas a la seguridad de la red, el administrador del sistema podrá tener una mayor necesidad de recabar información de los archivos del sistema, incluso en los directorios base del usuario

La política de seguridad de red deberá especificar el límite hasta donde los administradores del sistema podran examinar los directorios y archivos privados del usuario para el diagnóstico de problemas del sistema y para investigar violaciones a la seguridad. Si la seguridad de la red está en riesgo, la política deberá permitir una mayor flexibilidad para que los administradores del

sistema corrijan los problemas de seguridad. Otras preguntas con relación a esto son las siguientes:

- ¿El administrador del sistema puede revisar o leer los archivos de un usuario por alguna razón en particular?
- ¿Puede el administrador borrar información que considere pertinente, en caso de necesitar espacio en el disco?
- ¿ Los administradores de red tienen el derecho de examinar el tráfico de la red o del anfitrión?
- ¿Cuáles son las contingencias legales de los usuarios, administradores del sistema y organizaciones por acceso no autorizado a los datos privados de otras personas?

III.3.5. MANEJO DE INFORMACION DELICADA

Se deberá determinar qué tipo de datos delicados tienen que ser guardados en un sistema específico. Desde el punto de vista de seguridad, los datos delicados en extremo, como una nómina y los planes futuros, deberían restringirse a algunos anfitriones y administradores del sistema. Antes de otorgar el acceso a los usuarios a un servicio en un anfitrión, es necesario considerar qué servicios e información existen y a los cuales pueden ingresar un usuario. Si el usuario no tiene necesidad de manejar los datos delicados, entonces no debería tener una cuenta en un sistema que contiene dicho material.

También es necesario considerar la existencia de la seguridad adecuada en el sistema para proteger los datos delicados. En general, no es conveniente que los usuarios guarden información muy delicada en un sistema que no planea la seguridad con la importancia que esta requiere. Por otro lado, asegurar un sistema puede incluir hardware adicional, software y costos de administración, y podría no ser costeable asegurar datos en un anfitrión que no es muy importante para la organización o los usuarios.

La política también debe considerar que es necesario decirle a los usuarios quiénes podrían guardar información delicada y qué servicios son apropiados para ello.

III.4.-ACCIONES CUANDO LA SEGURIDAD HA SIDO VIOLADA

Cada vez que se viola la política de seguridad, el sistema se abre a amenazas de seguridad. Si no ocurre un cambio en la seguridad de la red después de esto, entonces la política de seguridad deberá ser modificada para retirar aquellos elementos que no estén asegurados.

Sin importar qué tipo de política se implante, existe la tendencia a ser violada por algunos usuarios. A veces es obvio cuando una política de seguridad ha sido quebrantada; otras veces

estas infracciones pasan inadvertidas. Los procedimientos de seguridad que se implanten deberán minimizar la posibilidad de que una infracción de seguridad pase inadvertida.

Cuando se detecte una violación a la política de seguridad, se debe clasificar si la violación ocurrió por una negligencia personal, un accidente o error, ignorancia de la política actual o ignorancia deliberada a la política. En este último caso, la violación pudo haber sido realizada no sólo por un individuo, sino por un grupo, que realizan conscientemente un acto de violación directo de una política de seguridad. En cada una de esas circunstancias, la política de seguridad debe ofrecer guías sobre las medidas a tomar de inmediato.

III.4.1. RESPUESTA A LAS VIOLACIONES DE LA POLITICA

Cuando una violación sucede, la respuesta depende del tipo de usuario que causó la violación. Las violaciones a la política podrían ser cometidas por una amplia variedad de usuarios, tanto locales como externos. Los usuarios locales son llamados con frecuencia internos y los foráneos, externos. La diferencia entre ellos se basa más que nada en límites de red, administrativos, legales o políticos. El tipo de límite determina cómo deberá ser la respuesta a la violación de seguridad. Los ejemplos de respuestas pueden ir desde una reprimenda verbal o advertencia, una carta formal o incluso cargos legales.

Es necesario definir acciones basadas en el tipo de violación. Se requiere definir estas acciones con claridad, tomando como base el tipo de violación del usuario a la política de seguridad. Los usuarios internos y externos de su red deben conocer la política de seguridad. Si existen usuarios foráneos que utilizan la red de manera legal, es responsabilidad de el administrador del sistema verificar que estos usuarios tengan una conciencia general de las políticas que se hayan establecido. Esto es de particular importancia si se necesitara acción legal en contra de los transgresores. Si ha ocurrido una pérdida importante, tal vez se quiera iniciar un acción más drástica. Si urge con esto una publicidad adversa para la organización, tal vez se prefiera arreglar la falla en la seguridad y no iniciar acción legal.

El documento de política de seguridad también deberá incluir los procedimientos para el manejo de cada incidente de violación a la seguridad. Un registro adecuado para esas violaciones de seguridad deberá mantenerse y revisarse periódicamente para observar las tendencias y tal vez ajustar la política de seguridad para que se tome en cuenta cualquier tipo nuevo de amenazas.

III.4.2. RESPUESTA A LAS VIOLACIONES POR USUARIOS LOCALES

Tal vez haya una violación a la política de seguridad en la cual el transgresor sea un usuario interno. Esto ocurre en las siguientes situaciones:

- Un usuario local viola la política de seguridad del sitio local.
- Un usuario local viola la política de seguridad de un sitio remoto.

En el primer caso, puesto que la política de seguridad ha sido violada, se podrá tener mayor control sobre el tipo de respuesta a dicha violación de seguridad. En el segundo, un usuario interno viola la política de seguridad de otra organización. Esto puede suceder por medio de una conexión por internet. La situación se complica porque está implicada otra organización y cualquier respuesta que tome tendrá que discutirla con la organización cuya política de seguridad fue violada por un usuario. También se deberá consultar a los abogados de la corporación o aquellos especializados en seguridad legal de computadoras.

III.4.3. ESTRATEGIAS DE RESPUESTA

Hay dos tipos de estrategias de respuesta a incidentes de seguridad:

- Proteger y proceder.
- Perseguir y procesar.

Si los administradores de las políticas de seguridad sienten que la compañía es bastante vulnerable, podrán elegir la estrategia de proteger y proceder. La meta de esta política es proteger de manera inmediata la red y restaurarla a su estado normal para que los usuarios puedan seguir utilizándola. Para hacer esto, quizá se tenga que interferir en forma activa con las acciones del intruso y evitar mayor acceso. Esto deberá ser seguido por un análisis de la cantidad de daño causado.

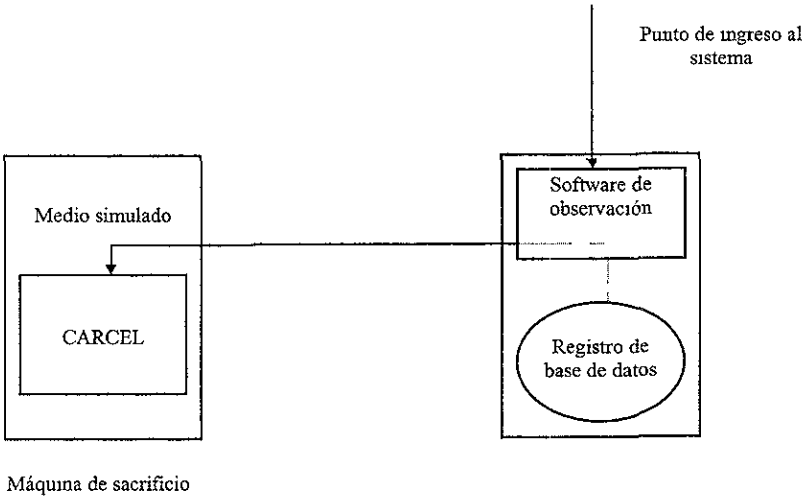
A veces no es posible restaurar la red de manera inmediata a su operación normal; tal vez se deberán aislar segmentos en la red y cerrar sistemas, con el objeto de prevenir mayor acceso no autorizado al sistema. Una desventaja de esto es que los intrusos saben que han sido detectados e iniciarán acciones para evitar ser rastreados. También, el intruso podrá reaccionar a su estrategia de protección mediante el ataque al sitio con una estrategia diferente; por lo menos, el intruso continuará su destrucción en otro sitio.

El segundo enfoque (perseguir y procesar) adopta la estrategia de que la mejor meta es permitir a los intrusos seguir con sus acciones mientras se observan sus actividades. Esto deberá hacerse tan disimuladamente como sea posible para que los intrusos no se den cuenta de que

están siendo observados. Las actividades del intruso deberán registrarse, para que existan pruebas disponibles en la fase de acusación de la estrategia. Este procedimiento es el que recomiendas las agencias de la ley y fiscales, porque esto genera pruebas que esas agencias pueden usar en el procesamiento de los intrusos. La desventaja de esto es que el intruso continuará robando información o haciendo otros daños, y el administrador del sistema será vulnerable a las demandas que resulten del daño al sistema y la pérdida de información.

Una forma posible de vigilar a los intrusos sin causar daño al sistema es construir una cárcel. En este caso, define un medio simulado para que lo utilice el intruso, para que sus actividades puedan ser observadas. El medio simulado presenta datos falsos, pero el sistema se prepara de tal forma que las actividades del intruso sean detectadas.

La siguiente figura muestra la idea general para construir una cárcel:



Para construir una cárcel, es necesario tener acceso al código fuente del sistema operativo y talento de programación de la compañía que pueda simular este medio. Es más seguro construir la cárcel con el uso de una máquina de sacrificio en un segmento aislado de la red para minimizar el riesgo de contaminar otros segmentos de la red y sistemas por las actividades del intruso

En el sistema UNIX el mecanismo chroot puede ser muy útil para hacer una cárcel. Este mecanismo confía de manera irrevocable un proceso para una ramificación sencilla del sistema del archivo. Con fines prácticos, la root de este ramal del sistema de archivo aparece como la root del sistema de archivo al proceso. Este mecanismo evita el acceso hacia los archivos de los dispositivos y al archivo real de contraseñas /etc/passwd.

Si no se desea que haya otros usuarios registrados en la máquina de sacrificio, se deberá actualizar en forma periódica el archivo utmp, el cual contiene los datos de los usuarios registrados para que la cárcel tenga realismo. También se deberá cancelar el acceso a aquellas utilerías que puedan revelar que la cárcel es un medio simulado. Algunas de estas utilerías son netstat, ps, who, w. Alternadamente, se podrá ofrecer versiones falsas de estas utilerias para hacer que este medio aparente ser real.

Una vez que se tenga suficiente evidencia en contra del intruso, tal vez se desee procesarlo. Sin embargo, esto no siempre es la mejor solución. Si el intruso es un usuario interno o un usuario huésped como un estudiante, las acciones disciplinarias podrán ser igual de efectivas sin el costo adicional del proceso legal y la publicidad dañina que esto conlleva. La política de seguridad de la red debería listar estas opciones y ofrecer guías de cuándo deben ejecutarse.

La estrategia de proteger y proceder puede usarse bajo las siguientes condiciones:

- Si los recursos de la red no están bien protegidos contra los intrusos.
- Si la continua actividad del intruso pudiera resultar en gran daño y riesgo financiero.
- Si el costo del proceso es demasiado alto o si la posibilidad o los deseos de procesar no existen.
- Si hay un riesgo considerable para los usuarios existentes en la red.
- Si los tipos de usuarios de una red interna grande no se conocen en el momento del ataque.
- Si el sitio es vulnerable a acciones legales por los usuarios. Esto es cierto para compañías de seguros, bancos, formas de seguridad, proveedores de red, etc.

Se puede utilizar la estrategia de perseguir y procesar bajo las siguientes condiciones:

- Si los recursos de la red y sistemas están bien protegidos. Esto significa que el ataque quedó sólo en un intento, registrado en las bitácoras del sistema.
- Si el riesgo para la red es incrementado por los disturbios causados por las intrusiones presentes y futuras potenciales.
- Si es un ataque concentrado y ha ocurrido antes.
- Si el sitio es muy visible y ha sido el blanco de ataques anteriores.
- Si no perseguir y procesar invita a nuevas incursiones o ataques.
- Si el sitio pone en riesgo los recursos de la red al permitir al intruso continuar.
- Si el acceso del intruso puede controlarse.

- Si las herramientas para observación están bien desarrolladas para crear registros aptos y recabar evidencia para el proceso legal.

- Si se cuenta en la organización con talentosos programadores para construir con rapidez herramientas especializadas.

- Si los programadores, administradores del sistema y red son tan listos y conocedores del sistema operativo, utilerías del sistema para que valga la pena la persecución.

- Si la gerencia desea un proceso legal.

- Si los administradores del sistema saben que tipo de evidencia conduciría al proceso legal, y pudieran crear registros adecuados de las actividades del intruso.

- Si hay contactos con agencias legales conocedoras.

- Si hay un representante versado en los asuntos legales relevantes.

- Si el sitio está preparado para una posible acción legal de sus usuarios si los datos o sistemas se encontraran comprometidos durante la persecución.

- Si se cuenta con respaldos adecuados.

III.4.4. ORGANIZACIONES EXTERNAS

La política de seguridad de red deberá definir los procedimientos para interactuar con organizaciones externas. Estas podrían incluir agencias de observación de la ley, expertos legales, otros sitios afectados por el incidente de la violación de seguridad, organizaciones de equipo de respuesta externa como CERT (equipo de respuesta a emergencia de computadoras), CIAC (capacidad en asesoría de incidentes de computadora), y si fuera necesario, a agencias noticiosas.

Es necesario que la gente con autorización para establecer contacto con estas organizaciones esté identificada. Se sugiere identificar a más de una persona para cada área, a fin de prevenir situaciones en las que los individuos designados no estén disponibles. Los temas a considerar pueden incluir los siguiente:

- Identificar "los tipos de relación pública" de quienes están versados para hablar con la prensa.

- ¿Cuándo ponerse en contacto con agencias de observación de la ley local y federal e investigadoras?

- ¿Qué tipo de información puede ser divulgada?

III.5.- INTERPRETACION Y PUBLICACION DE LA POLITICA DE SEGURIDAD

Es importante identificar a la gente que interpretará la política. No es buena idea contar con un sólo individuo, pues podría darse el caso de que esta persona esté ausente en un momento de crisis. Se puede identificar un comité, pero tampoco es buena idea tener demasiados miembros. De vez en cuando, el comité de política de seguridad será llamado para interpretar, revisar y analizar el documento.

Una vez escrita y acordada la política de seguridad del sitio, deberá asegurarse que la declaración de la política ha sido diseminada y discutida con amplitud. Podrán utilizarse listas de correo. La nueva política también puede ser observada mediante educación interna como seminarios de entrenamiento, juntas de grupo, talleres, pláticas con administradores de persona a persona o todo esto de acuerdo con el tamaño de la institución y las necesidades del momento.

Implantar una política de seguridad efectiva es un esfuerzo colectivo. Por lo tanto, se debe permitir a los usuarios de la red comentar la política durante cierto tiempo. Tal vez se desee mantener reuniones para recabar comentarios y asegurarse de que la política ha sido entendida de manera correcta. Esto también ayudará a clarificar el lenguaje de la política y evitar ambigüedades e inconsistencias en la política.

Las reuniones serán abiertas a todos los usuarios en la red y a los niveles más altos de la gerencia, quienes podrán tomar desiciones globales cuando surjan preguntas importantes. La participación y el interés del usuario ayuda a asegurar que la política será mejor entendida y más propensa a ser cumplida.

III.5.1. IDENTIFICACION Y PREVENCION

La política de seguridad define lo que es necesario proteger, pero la política quizá no explíque cómo deberán protegerse los recursos y el enfoque general para manejar los problemas de seguridad. Una sección adicional a la política de seguridad debe discutir los procedimientos generales sobre su implantación para prevenir problemas de seguridad. La política de seguridad podría referirse a la guía del administrador del sistema del sitio para los detalles adicionales sobre la implantación de los procedimientos de seguridad.

Antes de establecer los procedimientos de seguridad se deberá evaluar el nivel de importancia de los recursos de la red y su grado de riesgo de seguridad. El desarrollo de una política de seguridad efectiva requiere un esfuerzo considerable. Conlleva un esfuerzo considerar todos los asuntos y cierta disposición para establecer las políticas en papel y hacer todo lo necesario para estar seguros que la política ha sido entendida por los usuarios de la red.

Además de realizar un análisis de riesgo de los recursos de la red, se requiere identificar otros puntos débiles. La siguiente lista es un intento de describir algunas de las áreas de problemas más comunes. Esta lista pude guiar al administrador en la dirección correcta, sin embargo, de ningún modo es la más completa, debido a que cada sitio está propenso a tener debilidades únicas.

- Puntos de acceso: Son puntos de entrada (también llamados ingresos) para los usuarios no autorizados. Tener varios puntos de acceso aumenta los riesgos de seguridad en la red.

- Sistemas configurados de manera incorrecta: si los intrusos penetran la red, tratarán de subvertir al anfitrión del sistema. Los anfitriones que actúan como servidores telnet son el blanco más común. Si la configuración del anfitrión es deficiente, el sistema puede ser fácilmente subvertido. Los sistemas mal configurados son responsables de gran número de problemas de seguridad en las redes.

- Problemas de software: conforme aumenta la complejidad del software, también crece el número y complejidad de problemas en cualquier sistema. Tal vez el software nunca estará libre de problemas, a menos que surjan métodos revolucionarios para su creación. Los problemas de seguridad conocidos por el público son métodos comunes de entrada no autorizada. Si la implantación de un sistema es abierto y conocido con amplitud, el caso de UNIX, un intruso puede utilizar las debilidades en el código de software que trabajan en un modo privilegiado para ganar acceso al sistema.

- Amenazas de usuarios internos: Por lo general, los usuarios internos tienen un acceso más directo al software de la computadora y de la red que al del hardware real. Si un usuario interno decide subvenir la red, puede representar una amenaza considerable a la seguridad de la red. Si alguien tiene acceso físico a los componentes de un sistema, éste es más fácil de subvenir. Por ejemplo, muchas estaciones de trabajo pueden ser manipuladas fácilmente para otorgar un acceso privilegiado. Es sencillo ejecutar decodificación de protocolo y captura de software para analizar el tráfico de protocolo. La mayoría de servicios de aplicación TCP/IP estándar como telnet, rlogin y ftp tienen mecanismos muy débiles de autenticación, donde las contraseñas se envían a la vista de todos. El acceso a los servicios que utilizan cuentas privilegiadas debería evitarse, pues esto podría comprometer las contraseñas para esas cuentas.

- Seguridad física Si la computadora, por sí misma, no tiene una seguridad física, los mecanismos de seguridad del software pueden ser ignorados con facilidad. En el caso de las estaciones de trabajo DOS/Windows, no hay siquiera un nivel de contraseña de protección. Si una estación UNIX queda desatendida, sus discos físicos pueden ser suplantados, o si la estación de trabajo se deja en modo priviligiado, queda abierta del todo. Además, el intruso puede parar la maquina y regresarla a este modo. Y luego colocar caballos de troya o hacer todo lo necesario para dejar la máquina abierta para futuros ataques.

Todos los recursos de red críticos como estructuras de apoyo, enlaces de comunicación, anfitriones, servidores y máquinas clave deberán colocarse en áreas con seguridad física. El mecanismo de autenticación kerberos, por ejemplo, requiere que el servidor esté seguro

físicamente. Esto significa que la máquina esté encerrada en un cuarto o ubicada de tal manera que restrinja el aceso físico a los datos de la máquina.

A veces no es fácil asegurar físicamente las máquinas. En este caso se deberá tener cuidado de no descuidar demasiado esas máquinas. Se deberá limitar el acceso de máquinas que no esten seguras a aquellas que lo esten mejor. En particular, no se debe debe permitir el acceso hacia anfitriones mediante el uso de mecanismos de acceso confiable como las utilerías Berkeley-r* (rsh,rlogin,rcp,rwho,ruptime,rexec).

Aún cuando la máquina esté en un lugar seguro, es necesario vigilar quién tiene acceso a ella. El uso de tarjetas electrónicas "inteligentes" para tener acceso a la sala en donde las máquinas se encuentran, puede limitar el número de gente con acceso y también ofrecer un registro de la identidad y la hora en que los individuos entren a ella. También se deberá contar con una política que evite que los usuarios introduzcan a otras personas a la sala al abrirse la puerta, aun cuando su identidad sea conocida. Si el administrador permite que entre otra gente, no tendrá un registro preciso de quién y cuando entró a la sala.

III.5.2. CONFIDENCIALIDAD

La confidencialidad puede definirse como el acto de mantener las cosas ocultas o secretas. Esta es una consideración importante para varios tipos de datos delicados.

Las siguientes son algunas situaciones en las que la información es vulnerable de ser divulgada:

- Cuando la información se guarda en el sistema de cómputo.
- Cuando la información está en tránsito hacia otro sistema de red.
- Cuando la información se guarda en cintas de respaldo.

El acceso a la información que se guarda en una computadora se controla con permisos de archivo, listas de control de acceso (ACL) y otros mecanismos similares. La información que esté en tránsito puede protegerse mediante encriptación o compuertas de barreras de protección. Es posible utilizar la encriptación para proteger las tres situaciones. El acceso a la información almacenada en cintas puede controlarse mediante seguridad física como guardarla en una caja fuerte o en un área inaccesible.

Es necesario seleccionar los mecanismos de control y protección de tal forma que puedan contrarrestar efectivamente las amenazas que surgen durante la evaluación del riesgo. Estos controles deberán implantarse de tal manera que sean costeables. Tiene poco sentido gastar grandes sumas de dinero, y sobreproteger y restringir el uso de un recurso si el riesgo de exposición es mínimo.

Con frecuencia, el sentido común es una herramienta muy eficaz para establecer la política de seguridad. Aun cuando los esquemas y mecanismos de seguridad elaborados pueden ser impresionantes, también pueden ser demasiado costosos. A veces los costos de estas implantaciones están escondidos. Por ejemplo, se podrá implantar una solución de seguridad de software disponible en forma gratuita pero sin tomar en cuenta el costo de administrarla y mantenerla al día. Además , cuando la solución de seguridad es muy elaborada, podrá ser difícil implantarla y administrarla. Si la administración es una instalación de un paso, los comandos para administrar tal sistema pueden olvidarse con facilidad.

III.5.3. SELECCION DE LA POLITICA DE CONTROL

Los controles que se seleccionen srán la primera línea de defensa en la protección de la red. Estos controles deberían representar con precisión lo que se intenta proteger como fue definido en la política de seguridad. Por ejemplo si la mayor amenaza al sistema fuera el empleo no autorizado de recursos de computadora por usuarios internos, tal vez se deseará establecer procedimientos de contabilidad automatizada. Si la mayor amenaza a la red son los usuarios externos, se querrá construir enrutadores de selección y soluciones de barreras de protección.

III.6.- DETECCION Y VIGILANCIA DE ACTIVIDAD NO AUTORIZADA

Cualquier intento de intrusión deberá ser detectado lo más rápido posible. Es factible implantar varios procedimientos simples para detectar los usos no autorizados de un sistema de cómputo. Algunos procedimientos se basan en las herramientas suministradas por el proveedor en el sistema operativo.

III.6.1. USO DEL SISTEMA

El administrador del sistema puede realizar periódicamente la vigilancia del sistema. De la misma manera, es posible utilizar el software creado para vigilancia del sistema. La vigilancia de un sistema incluye la observación de varias de sus partes y la búsqueda de algo inusual. Algunos de estos métodos se explican a continuación.

La vigilancia debe hacerse en forma regular. No es suficiente hacerla cada mes o cada semana, pues esto dejaría una brecha en la seguridad sin ser detectada por un largo tiempo. Algunas brechas de seguridad pueden ser detectadas unas horas después de que suceden, en cuyo caso la vigilancia semanal o mensual no servirá de mucho. La meta de la vigilancia es detectar la brecha en la seguridad con oportunidad para poder responder de manera apropiada.

Si se utilizan herramientas de vigilancia, es recomendable examinar con frecuencia la salida de estas. Si los registros son voluminosos, tal vez se deba emplear los scripts awk o perl para analizar la salida. Estas herramientas también se encuentran disponibles para sistemas diferentes a UNIX.

III.6.2. VIGILANCIA DE LOS MECANISMOS

Muchos sistemas operativos guardan información acerca de los registros en archivos especiales de registro. El administrador del sistema deberá examinar estos archivos de registro con regularidad para detectar el uso no autorizado del sistema. La siguiente es una lista que se puede utilizar en un sitio.

- Comparar las listas de usuarios registrados con historiales de registro anteriores. La mayoría de los usuarios de red que trabajan con cierto horario y registran su ingreso y salida casi a la misma hora todos los días. Una cuenta que muestre actividad de registro fuera de las horas "normales" deberá ser vigilada de cerca. Tal vez un intruso utiliza esa cuenta. También se puede advertir a los usuarios para que verifiquen el último mensaje registrado que aparece al momento de hacer su primer registro. Si detectan horarios inusuales, deberán alertar al administrador del sistema.

- Muchos sistemas operativos pueden utilizar historiales de cuentas con propósito de cobranza. Estos historiales pueden ser examinados también para detectar patrones de uso inusual en el sistema. Dichos historiales de cuentas inusuales podrían indicar la penetración ilegal al sistema.

- El sistema operativo podría tener recursos de registro al sistema, como syslog utilizado en UNIX. Es necesario verificar los registros que producen estas herramientas para detectar errores inusuales en los mensajes desde el software del sistema. Por ejemplo, un gran número de intentos fallidos en un corto periodo de tiempo podría indicar que alguien trata de adivinar contraeñas. Se debería también de vigilar el número de intentos de registro en las cuentas delicadas como root, sisadm, etc.

- Muchos sistemas operativos tienen comandos, como ps de UNIX, para listar los procesos de ejecución que se estén llevando a cabo. Esto puede servir para detectar a los usuarios que ejecutan programas no autorizados, así como para detectar programas no autorizados que hayan sido iniciados por un intruso.

- Si se tienen recursos especiales que se deseen vigilar, se podrán construir las propias herramientas de vigilancia con el uso de utilierías estándares del sistema operativo. Por ejemplo, se puede combinar el ls de UNIX y encontrar comandos en un script de shell para verificar las pertenencias de archivo privilegiado y permisos establecidos. Es posible guardar la salida de información de la actividad de vigilancia en listas que pueden ser comparadas y analizadas con el uso de herramientas UNIX ordinarias como diff, awk o perl. Las diferencias

en los permisos para archivos críticos pueden indicar modificaciones no autorizadas al sistema.

III.6.3. HORARIO DE VIGILANCIA

Los administradores del sistema deben realizar una vigilancia regular y frecuente al sistema a lo largo del día. Si la vigilancia se hace con horario fijo, podría resultar muy tediosa, pero los comandos para vigilancia pueden ejecutarse a cualquier hora del día durante los momentos de reposo, como cuando se atienden negocios por teléfono.

Al ejecutar comandos de vigilancia en forma frecuente, se sabrá con rapidez cuál es la salida normal de las herramientas de vigilancia. Esto es útil para detectar salidas inusuales en la vigilancia. Es posible intentar automatizar este proceso al ejecutar herramientas de búsqueda sobre salida, y se pueden buscar cieros patrones establecidos, pero generalmente es difícil anticipar todas las salidas inusuales que causan una intrusión en el sistema.

Si se ejecutan varios comandos de vigilancia varias veces a lo largo del día, será difícil para un intruso anticipar las acciones del administrador del sistema. El intruso no podrá adivinar cuándo podría ejecutar el comando de vigilancia para desplegar a los usuarios registrados, de esta manera corre un riesgo mayor de ser detectado. Por otro lado, si el intruso sabe que el sistema se revisa todos los días, por ejemplo, a las 6:00 pm, cuando todos han registrado su salida, esperará que el sistema complete su inspección antes de registrarse.

III.6.4. PROCEDIMIENTOS DE INFORMACION

En caso de detectar acceso sin autorización, se debe contar con procedimientos para informar este acceso y a quién sería informado. Además, es necesario que la política de seguridad cubra los siguientes puntos

- Procedimientos para manejo de cuentas: Al crear cuentas de usuarios, se deberá tener cuidado en no dejar ninguna laguna de seguridad. Si el sistema operativo es instalado por los medios de distribución, se requiere examinar el archivo de contraseña para las cuentas privilegiadas que no se necesiten

Las cuentas sin contraseña son peligrosas, incluso si carecen de un intérprete de comando, como las cuentas que existen sólo para observar quién está registrado en el sistema. Si estas cuentas no se preparan bien, la seguridad del sistema puede verse comprometida. Por ejemplo, si la cuenta del usuario anonimo utilizada por FTP no se establece de manera correcta, podría permitir que cualquier usuario entre al sistema y retire archivos. Si se cometieran errores al establecer esta cuenta y el acceso de escritura al sistema de archivo se otorgara en forma inadvertida, un intruso podría cambiar el archivo de contraseña o destruir el sistema.

El recurso de contraseña sombra fue introducido primero con System V, pero otros sistemas UNIX como SunOS 4.0 y más recientes, y 4.3BSD UNIX Tahoe, ofrecen esta característica. Este archivo permite que la forma cifrada de las contraseñas se oculte a los usuarios no privilegiados. El intruso, de esta manera, no puede copiar el archivo de contraseñas e intentar adivinar las contraseñas.

La política deberá también incluir los procedimientos para mantener la pista de quienes tienen cuentas de usuario privilegiado, como root en UNIX. En UNIX, si se conoce la contraseña root se puede utilizar el comando su para usurpar los privilegios root. Si la contraseña es descubierta de manera inadvertida, el usuario puede registrarse con sus propias cuentas personales y usurpar privilegios root. Entonces se deberá implantar una política que obligue el cambio de contraseñas para cuentas de usuario privilegiado en intervalos periódicos.

Los intrusos recaban información acerca de las debilidades de las versiones de sistema particulares. A veces circulan sus hallazgos en revistas clandestinas. Los administradores se suscriben en ocasiones a estas revistas para conocer a los intrusos.

• Configuración de procedimientos de administración: Es recomendable mantener versiones actualizadas del sistema operativo y de las utilerías críticas. Las debilidades de seguridad en los sistemas más viejos son por lo general, bien conocidas, y es muy probable que el intruso esté pendiente de los problemas de seguridad. Las nuevas ediciones de software, aunque solucionan los viejos problemas, frecuentemente introducen nuevos. Por esta razón, es importante evaluar los riesgos de no actualizarse hacia un nuevo sistema y dejar las lagunas en la seguridad sin ser descubiertas, contra el costo de actualizarse con un nuevo software.

En general, se puede confiar en la mayoría de los vendedores para que las nuevas ediciones de software solucionen viejos problemas y no generen mayores problemas de seguridad. Otra complicación es que la nueva edición puede romper el software existente de aplicación del que dependen sus usuarios. También es positivo coordinar los esfuerzos de actualización con más de un vendedor.

También se puede recibir soluciones por medio de las listas de correo en la red. Se deberá tener personal competente que pueda examinar estas soluciones a los problemas con cuidado y sólo implantarlas si son seguras.

• Procedimientos de recuperación: Siempre que se instale una nueva versión del sistema operativo, no sólo se debe respaldar la imagen binaria del kernel del sistema operativo sino también los archivos utilizados para compilar y configurar el sistema operativo. Esto es válido para las otras aplicaciones y software de la red.

Los respaldos del sistema de archivo son como una póliza de seguro. No sólo protegen en caso de que falle el disco duro u otro hardware, sino también contra eliminación accidental y como una medida de reserva si el sistema ha sido penetrado. Si se sospecha de una intrusión, quizá se tenga que restaurar el sistema desde un respaldo para protegerse de los cambios hechos por el intruso. Si no se puede detectar cuando sucedieron los cambios no autorizados, se debe

examinar varios respaldos. Si se carece de una buena copia del software del sistema, será difícil determinar lo que los datos de este y archivos se supone que sean.

Los respaldos diarios, así como los respaldos de incremento, pueden ser útiles al proveer un historial de las actividades del intruso. Mediante la revisión de respaldos más viejos, se podrá determinar cuando fue penetrado el sistema por primera vez. Aunque se hayan borrado los archivos del intruso, es posible observarlos en los respaldos.

Es necesario elegir una estrategia de respaldo. Las estrategias de respaldo incluyen usualmente la combinación de los siguientes métodos:

- Respaldo completo.
- Respaldo nivel 1.
- Respaldo nivel 2.
- Respaldo personalizado.

En los sistemas UNIX, el respaldo completo se llama respaldo nivel 0; un respaldo nivel 1 respalda todos los archivos que hayan sido modificados desde el último respaldo nivel 0, en general, un respaldo nivel N respalda a todos los archivos modificados desde el último respaldo N-1. En el caso de las utilerías de respaldo como dump, un respaldo nivel N respalda todos los archivos modificados desde el último respaldo N-1 o menor.

Se puede utilizar un número arbitrario de niveles, pero en general esto no tiene sentido, pues se vuelve difícil mantener la pista de los respaldos. Los niveles de respaldo numéricos son compatibles en los comandos de respaldo de estilo BSD desde el nivel 0 hasta el 9, pero el concepto puede usarse en cualquier sistema y tal vez tenga que llevar alguna compatibilidad manual. En BSD UNIX el programa de respaldo es dump, y los archivos que son respaldados a un nivel específico se mantienen en el archivo /etc/dumpdates.

En el respaldo completo (nivel 0), todos los datos se respaldan sin importar cuando fue la última vez que se modificó, o si no ha sido modificado. Un ejemplo de esto son todos los directorios y archivos en un sistema de archivos. Después de que los datos fueron respaldados, el bit de archivo se limpia para todos los archivos que se están respaldando.

Un respaldo de nivel 1 respalda a todos los archivos que han sido modificados desde el último respaldo completo nivel 0. Esto significa que todos los archivos que fueron respaldados en el primer respaldo nivel 1 también lo fueron en el segundo respaldo nivel 1, junto con cualquier archivo que hubiera sido modificado desde el primer respaldo nivel 1. Este proceso continúa con cada respaldo nivel 1, y es de esperarse que más archivos se respalden con cada respaldo de nivel 1.

Existe una diferencia para describir el respaldo de nivel 1. En los sistemas UNIX, el respaldo de nivel 1 se conoce como respaldo incremental. En muchos sistemas diferentes a UNIX, el respaldo nivel 1 se llama respaldo diferencial. El término respaldo incremental en muchos sistemas que no son UNIX significa algo por completo diferente. Para evitar confusión, la política debe establecer cuál definición se usará.

Para obtener un registro completo de las versiones más actuales de los archivos, se tendría que comenzar con el más reciente respaldo completo (nivel 1), y agregarlo a los archivos en el más reciente respaldo nivel 1. Esto es:

respaldo más reciente = Ultimo respaldo completo + d

donde d es el respaldo más reciente nivel 1.

Puesto que el último respaldo nivel 1 tiene todos los archivos que han sido modificados desde el último respaldo completo, se podrá reintegrar los datos con sólo dos juegos de respaldo de cinta: uno para el respaldo completo y otro para el último respaldo nivel 1.

Si los datos en uno de los últimos respaldos nivel 1 están corrompidos, se deberá tener apoyo en el penúltimo respaldo diferencial. Por otro lado, si cualquier dato en otra cinta de respaldo nivel 1 está corrompido, no importará mientras sean correctos los datos en el más reciente respaldo diferencial.

Si por algún tiempo no se ha realizado un respaldo completo y han habido muchos cambios en el archivo, el tamaño de los datos que se necesitan ser respaldados tiende a crecer con cada respaldo nivel 1. Si todos los archivos han sido modificados, la sesión de respaldo nivel 1 será la misma que las sesiones de respaldo completo. Esto no tiende a ser el caso, puesto que la mayoría de los sistemas de archivos contienen una mezcla de programas y datos, y los archivos de programa no se modifican en forma usual.

• Procedimientos de informe de problemas para los administradores del sistema: Los administradores del sistema deben tener un procedimiento definido para señalar los problemas de seguridad. En redes grandes, esto puede hacerse mediante la creación de una lista de correo electrónico que contenga las direcciones de todos los administradores del sistema en la organización. Algunas organizaciones establecen un equipo de respuesta que ofrece un servicio de emergencia.

III.7.- INFORMACION ACTUALIZADA

La política de seguridad, además de identificar las agencias con las que se debe establecer contacto en caso de un incidente de seguridad, también se debe designar a las personas que se mantendrán actualizadas en asuntos de seguridad y problemas.

Una conexión a internet es muy útil para unirse a las listas de correo o grupos de noticias que discuten temas de seguridad de interés para un administrador de red.

III.7.1. LISTAS DE CORREO

Las listas de correo se mantienen por servidores de listas en internet. Cuando se usa una lista de correo, podrá haber comunicación con usuarios en esta lista por el correo electrónico. Para enviar la respuesta o puntos de vista sobre un tema, se puede enviar el correo electrónico en una lista y todos los inscritos a ella recibirán el mensaje. La petición para unirse a una de estas listas se envía a otra dirección de correo. Esta dirección es diferente de la dirección de la lista. Se debe enviar el pedido de suscripción a la dirección de pedido de correo y no a la dirección de la lista. Algunos administradores de lista de correo ostentan una lista especial de preguntas frecuentes (FAQ). Las FAQ son, con frecuencia, un buen lugar para encontrar más información.

Las listas de correo pueden ser moderadas o inmoderadas. En las primeras, el dueño de la lista con frecuencia actúan como un moderador y filtra aquellas respuestas de correo que nada tienen que ver con los objetivos de la lista de correo.

III.7.2. LISTAS DE CORREO DE SEGURIDAD EN UNIX

El objetivo de estas listas es notificar a los administradores del sistema los problemas de seguridad antes de que sean del dominio público, y proporcionan información sobre aspectos relacionados con la seguridad. Puesto que este tipo de información pudiera ser dañino si llega a manos equivocadas, las listas de correo de seguridad UNIX, tienen acceso restringido, esta lista está cubierta sólo a las personas que puedem comprobar que son administradores de sistemas de un sitio.

Para unirse a esta lista, los pedidos deben originarse desde el contacto del sitio listado en la base de datos WHOIS del centro de información sobre redes de la red de la Defensa (DDN NIC). o desde una cuenta root en alguna de las principales máquinas de sitio. Se debe incluir la dirección de correo electrónico que se desea en la lista. También se debe incluir la dirección de correo electrónico y el número telefónico del correo local del contacto del sitio.

III.7.3. LISTA DEL FORO DE RIESGOS

El foro de riesgos forma parte del comité de computadoras y política pública (ACM). El foro es una lista moderada y discute los riesgos con el público acerca de computadoras y sistemas relacionados. También analizan aspectos de seguridad de interés particular, incidentes internacionales de seguridad de computadoras, problemas en los sistemas de control aéreos y de ferrocarril, ingeniería de software, etc.

III.7.4. LISTA VIRUS-L

La lista VIRUS-L discute las experiencias con virus en computadoras, software de protección y temas afines. La lista está abierta al público y se implanta como un compendio moderado. La mayoria de la información está relacionada con computadoras personales, aunque algo de esto podría ser aplicable a sistemas más grandes. Para suscribirse, se envía un correo electrónico a la dirección:

Listserv%lehiibm1.bitnet@mitvma.mit.edu

O

Listser@lehiibm1.bitnet

En el cuerpo de la lista se incluye la siguiente línea:

Subscribe virus-L Nombre Apellido

IV.- BARRERAS DE PROTECCION

IV.1.- ENRUTADORES DE SELECCION

IV.1.1. DEFINICION DE ENRUTADORES DE SELECCIÓN

Los enrutadores de selección pueden discriminar entre el tráfico de la red basado en el tipo de protocolo y los valores de los campos de protocolo en el paquete. La capacidad de estos enrutadores para discriminar entre los paquetes y restringirlos en sus puertos, tomando como base un criterio de protocolo específico, se llama filtración de paquetes. Por esta razón, los enrutadores de selección se llaman también enrutadores de filtro de paquetes.

IV.1.2. ZONAS DE RIESGO

Las estadísticas de internet hasta 1997 indicaban que consta de más de 30,000 redes con un total de más de 2.5 millones de anfitriones. Con tantos usuarios de red, existe, por desgracia, un pequeño segmento de usuarios que son vándalos. Esta situación se compara a la acción de mudarse a una gran ciudad que tiene su cuota de criminales. En este caso, es prudente proteger nuestro hogar con puertas seguras. La prudencia de nuestra parte también exige que si alguien toca la puerta se debería tener la habilidad de analizar a la persona antes de permitirle la entrada a la casa. A las personas con apariencia peligrosa o dañina no debe permitírseles la entrada. De manera parecida, los enrutadores de selección examinan los paquetes que ingresan para determinar cuáles podrían ser peligrosos.

En un sistema que este conectado a otra red externa, el límite de la red local se le llama perímetro de seguridad. A causa de que podrían abundar los vándalos en las redes externas, resulta muy útil definir una zona de riesgo. La zona de riesgo incluye a todas las redes compatibles con TCP/IP que son accesibles de manera directa por medio de las redes externas. Compatible con TCP/IP significa que el anfitrión soporta este protocolo y toda su familia. Accesible de manera directa significa que no existen medidas fuertes de seguridad (puertas abiertas) entre la red externa y los anfitriones en la red local.

Desde el punto de vista del administrador de una red local, regional y nacional, representan una zona de riesgo. Los anfitriones dentro de esta zona son vulnerables a los ataques. Colocar las redes y anfitriones fuera de la zona de riesgo es preferible. Sin embargo, sin un dispositivo que pueda bloquear los ataques contra la red, la zona de riesgo se extenderá hacia ella. El enrutador de selección es el dispositivo que puede reducir la zona de riesgo para que ésta no cruce el perímetro de seguridad.

No todos los anfitriones de la red de una empresa u organización podrán ser compatibles con TCP/IP, y aún así, estos anfitriones no compatibles con TCP/IP podrán volverse vulnerables aunque no sean de manera técnica parte de la zona de riesgo. Esto puede ocurrir si los anfitriones no compatibles con TCP/IP se conectan con este anfitrión. El intruso puede utilizar un protocolo común para este tipo de anfitriones, compatibles y no compatibles, para tener acceso al anfitrión no compatible con TCP/IP desde el que sí es compatible. Si los anfitriones se encuentran en el mismo segmento por ejemplo, Ethernet, el intruso puede llegar al anfitrión no compatible con TCP/IP mediante este protocolo. Los enrutadores de selección por sí mismos tal vez no sean capaces de eliminar la zona de riesgo. Sin embargo, son en extremo efectivos para reducirla.

IV.1.3. OSI Y LOS ENRUTADORES DE SELECCIÓN

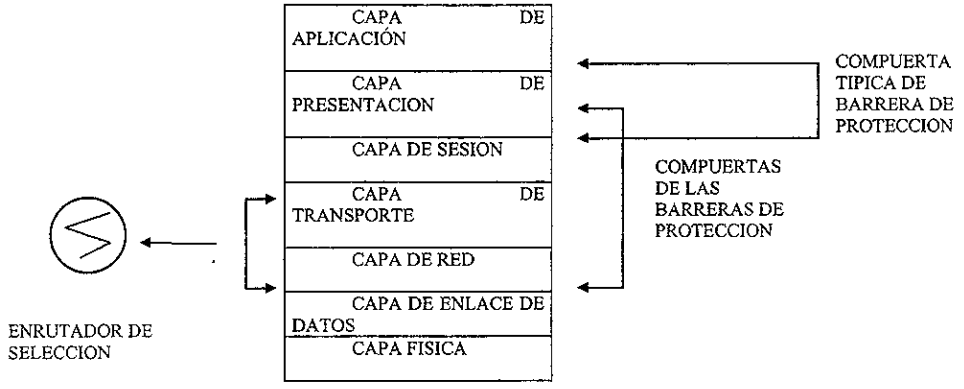
El término enrutador de selección implica un nivel de desempeño en lo que se refiere a los asuntos de enrutamiento. Entender el papel general del enrutador en la provisión de comunicaciones de red ayudará a comprender qué tipo de acciones de filtración son provistas por tales enrutadores. El modelo predominante en las funciones de comunicación es el modelo de referencia OSI. Este modelo puede utilizarse para describir el desempeño de un enrutador, y ayuda a explicar la magnitud de la capacidad de filtración de paquetes de un enrutador de selección y sus limitaciones.

El modelo de referencia OSI fue desarrollado en 1978 por la Organización Internacional de Estándares (ISO) para especificar un estándar que pudiera ser utilizado en el desarrollo de sistemas de red diseñados de acuerdo con el diseño y especificaciones OSI, que hablan el mismo lenguaje, esto es, usan métodos de comunicación similares o compatibles. Este tipo de sistemas de red permite que los sistemas de diferentes compañías interactúen entre sí.

Al inicio de las redes de cómputo (antes del modelo OSI), la arquitectura de red de computadora propietaria dominaba. Cualquier organización interesada en instalar una red examinaba la opciones disponibles de vendedores como IBM, DEC, HP, Honeywell y Sperry y Burroughs (ahora UNISYS). Cada uno de ellos tenía su propia arquitectura, la capacidad de interconectar redes de diferente firma era casi inexistente.

Una vez que se compraba equipo de un proveedor específico, la organización virtualmente se encasillaba. Las actualizaciones o modificaciones al sistema las suministraba el proveedor, y puesto que éste tenía una arquitectura propietaria cerrada, nadie podía competir con dicho vendedor para surtir servicios equivalentes.

IV.1.4. BARRERAS DE PROTECCION Y ENRUTADORES DE SELECCIÓN CON RELACION AL MODELO OSI



La figura anterior compara a los enrutadores de selección y barreras de protección con relación al modelo OSI. Esta figura muestra que las funciones del enrutador de selección corresponden a las capas de red (protocolo IP) y de transporte (protocolo TCP) en el modelo OSI. Las barreras de protección se llaman así porque operan en las capas superiores del modelo OSI y tiene información completa sobre las funciones de la aplicación en la cual basan sus decisiones; con frecuencia son descritas como compuertas. Las compuertas pueden realizar procesos en las siete capas del modelo OSI. Por lo general, las compuertas realizan procesos en la séptima capa del modelo OSI (de aplicación). Esto es verdad para la mayoría de las compuertas de barrera de protección.

En la figura también se observa que así como las barreras de protección cubren las capas de red y transporte, también pueden realizar funciones de filtración de paquetes. Algunos proveedores, tal vez por razones de mercadotecnia, enturbian la diferencia entre un enrutador de selección y una barrera de protección, al extremo de decir que sus enrutadores de selección son barreras de protección. Para aclarar esto último, aquí se hace la distinción entre estos dos términos con base al modelo OSI.

En ocasiones a los enrutadores de selección se le llaman también compuertas para filtración de paquetes. Tal vez una justificación del uso del término compuerta para el dispositivo de filtro de paquetes sea que la filtración basada en las banderas TCP que se realiza en la capa de transporte no es una función del enrutador que opera en la capa de red del modelo OSI. Los dispositivos que operan por encima de la capa de red se llaman compuertas.

IV.2.- FILTRACION DE PAQUETES

Los enrutadores de selección pueden utilizar la filtración de paquetes con el objetivo de aumentar la seguridad de la red. La función de filtración también puede ser realizada por muchos productos comerciales de barreras de protección y por productos basados en software, como por ejemplo, los filtros basados en PC Karlbridge. Sin embargo, muchos enrutadores comerciales pueden programarse para realizar filtración. Los vendedores de enrutadores ofrecen un producto que puede programarse para realizar funciones de filtración de paquetes.

La filtración de paquetes puede ser utilizada para implantar variedad de políticas de seguridad de red. La política de seguridad de red deberá dejar en claro qué tipos de recursos y servicios se están protegiendo, su nivel de importancia y la gente de la cual se protegen.

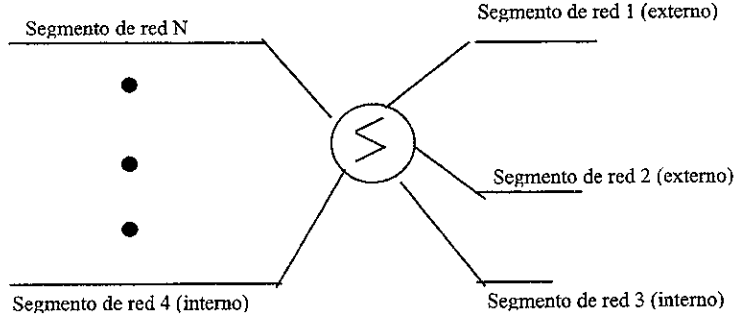
De manera general, las guías de la política de seguridad de red se concentran más en mantener fuera a extraños, que en tratar de perseguir a los usuarios de confianza. Por ejemplo, es más importante evitar que los usuarios externos irrumpan y expongan de manera intencional los datos sensibles o desorganicen los servicios que evitan que los usuarios de confianza utilicen servicios externos de la red. Este tipo de política de seguridad de la red determina dónde deben colocarse los enrutadores de selección y cómo deberán ser programados para llevar a cabo la filtración de paquetes. Una buena implantación de seguridad en la red. Con frecuencia éste no es el mejor procedimiento en los esfuerzos de seguridad.

Una de las metas de una política de seguridad en la red es ofrecer un mecanismo transparente, de manera que la política no sea un estorbo para los usuarios. Puesto que la filtración de paquetes opera en las capas de red y de transporte del modelo OSI, y no en la de aplicación, este enfoque tiende a ser más transparente que el de las barreras de protección. Es importante recordar que las barreras de protección operan en la capa de aplicación del modelo OSI, y las implantaciones de seguridad en esta capa tienden a no ser transparentes.

IV.2.1. MODELO DE FILTRACION DE PAQUETES.

Los segmentos de la red se clasifican como externos o internos. Los segmentos externos conectan a su red con redes externas. Los segmentos de red internos se utilizan para conectar a los anfitriones con otros recursos de la red.

En la siguiente figura se muestra como se coloca un filtro de paquetes comúnmente.



Cada uno de los puertos del dispositivo para filtro de paquetes puede utilizarse para implantar políticas que describan el tipo de servicio de red que es accesible por medio del puerto. Si el número de segmentos de red que se conectan con el dispositivo para filtro de paquetes es grande, las políticas que ese dispositivo implanta pueden volverse complejas. En general, las soluciones complejas a los problemas de seguridad deberían evitarse por las siguientes razones:

- Son más difíciles de mantener.
- Es fácil cometer errores al configurar el filtro de paquetes.
- Tienen un efecto adverso en el desempeño del dispositivo en el que se implantan.

En muchos casos el modelo sencillo que podría ser el de un filtro de paquete colocado entre dos segmentos de red, casi siempre, uno de estos segmentos de red es externo y el otro es interno. La filtración de paquetes se hace para restringir el tráfico en la red para los servicios que serán denegados. Puesto que la política de la red se escribió para favorecer que los usuarios de confianza (internos) establezcan contacto con anfitriones externos, el filtro en cada lado de los puertos enrutadores de selección deberán comportarse de manera diferente. En otras palabras los filtros son asimétricos.

IV.2.2. OPERACIONES DEL FILTRO DE PAQUETES

Casi todos los dispositivos actuales de filtro de paquetes (enrutadores de selección o compuertas de filtro de paquetes) operan de la siguiente forma:

1.- El criterio de filtro de paquetes debe almacenarse para los puertos del dispositivo para filtro de paquetes. El criterio de filtro de paquetes se conoce como reglas del filtro de paquetes.

2,- Cuando el paquete llega al puerto, entonces los encabezados de los paquetes se analizan, la mayoría de los filtros de paquetes examinan los campos sólo en los encabezados IP, TCP, o UDP.

3.- Las reglas del filtro de paquetes se almacenan en un orden específico. Cada regla se aplica al paquete en el mismo orden en el que la regla del filtro de paquetes se almacenó.

4.- Si una regla bloquea la transmisión o recepción de un paquete, el paquete no se acepta.

5.- Si una regla permite la transmisión o recepción de un paquete, a dicho paquete se le permite proceder.

6.- Si un paquete no satisface ninguna regla, es bloqueado.

En las reglas 4 y 5 , se observa que es importante colocar las reglas en el orden correcto Un error común al configurar las reglas del filtro de paquetes es hacerlo en desorden. Si las reglas del filtro de paquetes se colocan en un orden equivocado, podría terminar denegando servicios válidos, mientras que permitirán los servicios que deseaba denegar.

La última regla sigue esta filosofía:

Aquello que no esté expresamente permitido se prohibe.

Esta es una filosofía a prueba de fallas que deberá seguir en el diseño de redes seguras. Es lo opuesto a una filosofía permisiva:

aquello que no esté expresamente prohibido se permite.

Si esta filosofía permisiva se utilizara para el diseño de filtros de paquetes, tendría que pensar en cada caso que no fuera cubierto por las reglas de filtro de paquetes para hacer segura la red. y mientras se agreguen nuevos servicios, podrá llegar con facilidad a situaciones en las cuales ninguna regla se iguale. En lugar de bloquear este servicio y escuchar quejas de los usuarios a quienes habrá bloqueado un servicio legítimo, podría terminar permitiendo un servicio que pusiera en riesgo la seguridad de la red.

Los enrutadores de selección, en general, pueden filtrar con base en cualquiera de los valores de campo en los encabezados de protocolo TCP o IP. Para que la mayoría de las políticas de seguridad en red puedan implantarse mediante los enrutadores de selección, se necesitará especificar sólo las banderas TCP, opciones IP y valores de dirección del destinatario y la fuente. La siguiente tabla muestra una hoja de trabajo que se puede utilizar para el diseño de las reglas del filtro de paquetes.

Número de regla de filtro	Acción	Fuente	Puerto de la fuente	Destino	Puerto destino	Opciones de bandera de protocolo	Descripción
1							
2							
3							
4							
5							
6							
7							
8							

Si se examina cada fila en la hoja de trabajo, se observa que esta describe de manera completa la conexión TCP. Formalmente, una descripción completa de una conexión se llama asociación completa.

Cuando se diseñan las reglas de filtro de paquetes es útil tener en mente las de asociación completa, media asociación y puntos terminales. Esto ayuda a entender mejor las reglas de la filtración de paquetes.

Una asociación completa se puede describir mediante la siguiente información:

- Tipo de protocolo.
- Dirección local.
- Número de puerto local.
- Dirección remota.
- Número de puerto remoto.

IV.3.- IMPLANTACION DE REGLAS DE LOS FILTROS DE PAQUETES

Después de diseñar las reglas de filtro de paquetes y describirlas en la hoja de trabajo adecuada, se deberán implantar en el enrutador de selección o la barrera de protección (siempre y cuando se permita especificar las reglas de filtro de paquetes).

Se debe señalar que cada tipo de dispositivo para filtro de paquetes tiene su propio grupo de reglas y sintaxis acerca de cómo programar las reglas. Por lo tanto, se debe leer la documentación proporcionada para comprender la sintaxis del dispositivo para filtro de paquetes.

Para lograr cumplir el objetivo sobre este tema es necesario proporcionar un ejemplo práctico para construir barreras de protección en redes externas, a fin de mejorar la seguridad de la red, por esto y sobre lo ya mencionado anteriormente con relación a la diferencia de reglas y sintaxis, se especificará la marca del distribuidor del enrutador de selección, que en este caso será Cisco Inc; simplemente por ser quién domina el mercado de enrutadores. Las reglas de los filtros de paquetes para los enrutadores de otros distribuidores son en general similares a las reglas Cisco, pero difieren en términos de sintaxis.

IV.3.1. DEFINICION DE LAS LISTAS DE ACCESO.

Las listas de acceso se definen como una selección secuencial de condiciones de permiso y de negación que se aplican a las direcciones internet. Estas condiciones sirven para establecer las reglas de los filtros de paquetes.

Cuando el enrutador de selección esté programado con listas de acceso, probará los paquetes contra las condiciones que se encuentran en esa lista, una por una. El primer paquete que se ajuste a una de las condiciones determinará si el enrutador aceptará o rechazará el paquete. Puesto que el enrutador de selección detiene la prueba de las condiciones en la lista de acceso después de que corresponden un paquete y una condición, el orden de tales condiciones es muy importante. Si no hay condiciones que correspondan a los paquetes, éstos serán rechazados.

Los enrutadores CISCO tienen dos tipos de lista de acceso:

- Listas de acceso estándares.
- Listas de acceso extendidas.

Las listas de acceso estándares tienen una sola dirección para las operaciones de correspondencia, mientras que las listas extendidas tienen dos direcciones con información opcional de tipo protocolo para las operaciones de correspondencia. Para varias operaciones de filtración prácticas, se necesitan ambas listas de acceso.

IV.3.2. LISTAS DE ACCESO ESTANDARES

La sintaxis para las listas de acceso estándares es la siguiente:

acces-list lista (permit | deny) dirección máscara de comodín no access-list lista

La lista es un número entero que tiene un valor desde 1 hasta 99, el cual se utiliza para identificar una ó más condiciones de permiso y (o) negación. La lista de acceso cero se encuentra predefinida y permite cualquier acceso de la lista de acceso predeterminada para las líneas de terminal.

Las palabras "permit" y "deny" corresponden a los términos "permiso" y "negación" en las reglas de filtros de paquetes mencionados con anterioridad. Cualquier dirección IP destinataria en el paquete se compara al valor de dirección especificado en el comando de lista de acceso. Si se utiliza la palabra "permit", una correspondencia entre condición y paquete causará que éste se acepte. Si se usa la palabra "deny", una correspondencia del mismo tipo provocará que se rechace el paquete.

Los valores de dirección y máscara de comodín son de 32 bits cada uno y se escriben por medio de la notación decimal punteada. La máscara de comodín no debe confundirse con las máscaras de subred que subdividen un número de asignación de red IP. Los bits de dirección correspondientes a 1 en el valor de máscara de comodín se ignoran en la comparación; mientras que los bits de dirección correspondientes a 0 en dicho valor se utilizan en la comparación. Por ejemplo:

```
acces-list 1 permit 199.245.180.0 0.0.0.255
acces-list 1 permit 132.23.0.0 0.0.255.255
```

En el ejemplo, se especifican dos valores de dirección y máscara de comodín, los cuales se aplican al número 1 de la lista de acceso. El primer comando de la lista de acceso permite el acceso de anfitriones en la red de clase C 199.245.180.0 y el segundo comando permite el acceso de anfitriones de la red de clase B 132.23.0.0.

Si no se especifica el valor de la máscara de comodín, se supone que este valor será de 0.0.0.0; es decir, todos los bits en la dirección serán comparados y por lo tanto, los siguientes dos comandos de la lista de acceso tendrán un efecto similar:

```
acces-list 2 permit 132.23.1.3 0.0.0.0
acces-list 2 permit 132.23.1.3
```

Los dos comandos anteriores sólo aceptan los paquetes para el anfitrión con dirección IP 132.23.1.3. Si el valor de la máscara de comodín es diferente de cero, se debe especificar un rango de direcciones IP

IV.3.3. LISTAS DE ACCESO EXTENDIDAS

Las listas de acceso extendidas permiten filtrar el tráfico de interfaz con base en direcciones IP fuente y destino, además de la información del protocolo.

La sintaxis es la siguiente.

acces-list lista (permit . deny) protocolo fuente destino de máscara fuente máscara destino (operador operando)

La lista es un valor entero de 100 a 199 y se utiliza para identificar una o más condiciones de permiso y/o negación Los números 100 a 199 están reservados para las listas de acceso extendidas y se encuentran fuera del rango de los números 1 a 99, empleados para las listas de acceso estándares

Si se utiliza la palabra "permit", una correspondencia del paquete y la condición causará que el paquete se acepte. Esta acción equivale a la regla "permiso", usada en las reglas de diseño del filtro de paquetes. Si se ocupa la palabra "deny", una correspondencia causará que el paquete se rechace. Esta acción equivale a la regla "negación", empleada en el diseño del filtro de paquetes. El resto de la lista extendida no se procesará después de ocurrir una correspondencia.

El protocolo puede representar cualquiera de los siguientes valores que corresponden a los protocolos IP, TCP, UDP e ICMP:

- ip
- tcp
- udp
- icmp

El protocolo IP encapsula los paquetes TCP, UDP e ICMP, así que puede utilizarse para igualar a cualquiera de estos protocolos.

La fuente y la máscara de fuente son valores de 32 bits y están escritos con la notación decimal punteada. Estos valores se utilizan para identificar la dirección IP fuente. La máscara de fuente no debe confundirse con las máscaras de subred empleadas para subdividir un número de asignación de una red IP. Los bits de dirección correspondientes a 1 en la máscara de fuente se ignoran en la comparación, mientras que los bits de dirección correspondientes a 0 se usan en la comparación.

Los valores de destino y máscara de destino sirven para igualar la dirección IP destinataria. Estos valores también se escriben por medio de la notación decimal punteada y la máscara de destino se utiliza de la misa manera como se usa la máscara de fuente para las direcciones fuente.

Con los valores de operador y operando se comparan los números de puerto, puntos de acceso de servicio o nombres de contacto. Estos valores son significativos para los protocolos TCP y UDP. Para los valores clave tcp y udp, el operador puede tener cualquiera de los siguientes valores:

- it (menor que)
- eq (igual que)
- gt (mayor que)
- neq (diferente de)

El operador es el valor decimal del puerto destinatario del protocolo especificado. Para mostrar el uso de los comandos de la lista de acceso se mostrará un ejemplo:

La política de red requiere que se impida la entrada de las conexiones SMTP desde el anfitrión 132.124.23.55 hacia la red 199.245.180.0; por medio de la siguiente lista de acceso se puede implantar esta política:

```
no acces-list 101
acces-list 101 deny tcp 132.124.23.55 0.0.0.0 199.245.180.0 0.0.0.255
eq 25
```

El primer comando elimina cualquier lista de acceso extendida 101 anterior. El segundo comando no permite que se envíe un paquete TCP desde el anfitrión 132.124.23.55 a la red 199.245.180.0 con un puerto destinatario en 25 (SMTP).

IV.3.4. FILTRACION DE LLAMADAS ENTRANTES Y SALIENTES

Para restringir las conexiones entrantes y salientes entre una línea dentro del enrutador Cisco y las direcciones en una lista de acceso, se usará el comando de configuración de acceso y clase.

```
acces-class lista (in | out)
```

La lista es el número de la lista de acceso. Al utilizar el valor "in" al final del comando, se restringe el tráfico entrante entre el dispositivo Cisco y las direcciones de la lista de acceso. Al emplear el valor "out" al final del comando, se restringe el tráfico saliente entre el dispositivo Cisco y las direcciones en la lista de acceso. Si se desea eliminar las restricciones de acceso especificada, se usará el siguiente comando:

```
no acces-class numero de lista de acceso (in | out)
```

En el siguiente ejemplo, se define una lista de acceso que sólo permite la conexión de anfitriones en la red 199.245.75.0 con los puertos del 1 al 5 de la terminal virtual en el enrutador:

```
acces-list 18 permit 199.245.75.0   0.0.255
line 1  5
acces-class   18 in
```

En el siguiente ejemplo, se bloquean las conexiones a las redes diferentes de la red 156.233.0.0 en las líneas terminales 1 a 3:

```
acces-list  19  permit  156.233.0.0  0.0.255.255
linea  1  3
acces-class  19  out
```

IV.3.5. OPCION DE SEGURIDAD IP PARA LOS ENRUTADORES CISCO

Los enrutadores Cisco soportan tanto a las opciones de seguridad básica como a las extendidas. Este soporte es útil para las redes que aplican la seguridad IP. La opción de seguridad IP (IPSO) tiene las siguientes características:

- Define el nivel de seguridad en una base de interfaz individual.
- Define las interfaces de nivel sencillo o de multinivel.
- Proporciona etiquetado para datagramas entrantes.
- Elimina las etiquetas en una base de interfaz individual.
- Reordena las opciones para colocar cualquier opción básica de seguridad al comienzo.
- Acepta o rechaza los mensajes con opciones de seguridad extendidas.

IV.3.6. COLOCACION DE FILTROS DE PAQUETES Y LA SUPLANTACION DE DIRECCION

Al diseñar las reglas de los filtros de paquetes, se debe especificar si se desea realizar la filtración en los paquetes entrantes o salientes. Asimismo, se puede determinar el lugar donde es preciso prevenir la suplantación de direcciones.

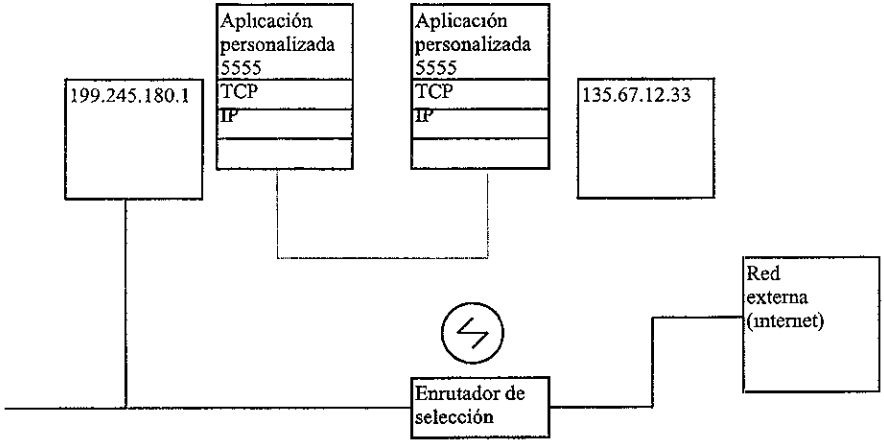
Muchos distribuidores de enrutadores ejecutan la filtración en los paquetes salientes por razones de eficacia Las reglas de filtro pueden aplicarse en los paquetes salientes, cuando el enrutador consulta las tablas correspondientes para determinar el destino del paquete. Si el paquete no puede enviarse con una ruta dada o no corresponde a las reglas de filtro, el paquete se rechaza y se envía el mensaje ICMP destination unreachable (destino no localizado).

Si los enrutadores filtran los paquetes en el momento en que estos se envían fuera de un puerto de enrutador, se perderá información. Como el enrutador no sabe a cual interfaz ha llegado el paquete, puede dejar a la red resentida para cualquier tipo de ataque, lo cual se conoce como una suplantacion de dirección. Así la filtración de los paquetes entrantes puede prevenir este tipo de ataque, en general la filtración se debe realizar tan rápido como sea posible.

IV.3.7. FILTROS EN PUERTOS DE ENTRADA Y SALIDA

No todos los enrutadores filtran en los puertos de fuente y en los puertos de destino, sino que la mayoría sólo filtra en este último, debido a que las conexiones TCP requieren un flujo de datos en ambas direcciones, el puerto donde se quiera colocar el filtro aparecerá como el puerto destinatario cuando se envíen los datos o se reciba la confirmación de recepción. Sin embargo, si no se es capaz de facilitar en forma simultánea los puertos de fuente y de destino, esto puede causarle problemas.

Como ejemplo debemos considerar que la política de seguridad de una red permite que haya conexiones TCP para una aplicación personalizada entre un anfitrión interno y uno externo. Suponemos que este servicio personalizado utiliza el número de puerto TCP 5555 en ambos lados. Este caso se muestra en la siguiente figura. Se puede diseñar la regla del filtro de paquetes por medio de la hoja de diseño de las reglas de paquetes:



Y la siguiente es el diseño de la regla del filtro de paquetes para una aplicación personalizada, cuando los puertos fuente y de destino pueden especificarse.

Número de regla de filtro	Acción	Fuente	Puerto de la fuente	Destino	Puerto destino	Opciones de bandera de protocolo	Descripción
1	acepta	199 245 180.1	5555	135.67.12.33	5555	TCP	Acepta una sesión TCP para un puerto 5555 de anfitrión externo desde anfitriones externos
2							
3							

en la tabla anterior se muestra que sólo se necesita una regla del filtro de paquetes. Sin embargo, si el enrutador de selección sólo permite especificar el puerto destino, la única regla que hay en la tabla tiene que escribirse como dos:

Número de regla de filtro	Acción	Fuente	Puerto de la fuente	Destino	Puerto destino	Opciones de bandera de protocolo	Descripción
1	acepta	199.245.180.1	-	135 67 12 33	5555	TCP	permite una sesión TCP para un puerto 5555 de anfitrión externo desde anfitriones internos.
2	acepta	135.67 12.33	-	199 245 180 1	5555	TCP	permite una sesión TCP para un puerto de anfitrión interno 5555 desde anfitriones externos
3							

en conclusión, para escribir con efectividad las reglas para los filtros de paquetes, los enrutadores de selección deben aceptar la especificación tanto de los puertos fuente como de los puertos de destino, dentro de una sola regla correspondiente.

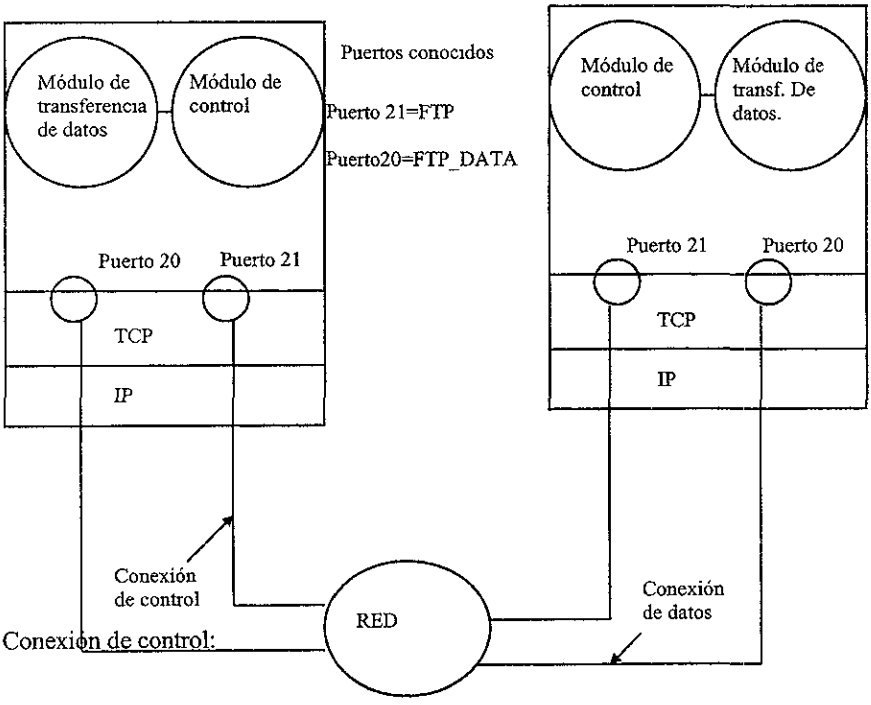
IV.4.- PROTOCOLOS Y LA FILTRACION DE PAQUETES

IV.4.1. FILTRAR EN EL TRAFICO EN UNA RED FTP

Cuando se diseñan las reglas de un filtro de paquetes, se debe comprender primero el comportamiento del servicio de aplicación que se trata de filtrar. Algunos servicios de aplicación, como FTP, requieren un mecanismo de respuesta donde un anfitrión externo pueda necesitar iniciar una conexión con el anfitrión interno dentro de un puerto desconocido al momento de especificar las reglas del filtro de paquetes.

El protocolo X11, utilizado en las aplicaciones Unix X-Windows, también necesita una llamada "entrante" para un anfitrión interno desde un anfitrión externo. Los anfitriones internos deben estar protegidos contra este tipo de llamadas.

Ahora se analizará el comportamiento normal de FTP y los problemas que este presenta para los enrutadores de selección. Además se analizará las soluciones a la filtración de paquetes para las sesiones FTP.



- Conexión de control:

I.- Creada cuando se establece la conexión al servidor FTP

II.- Utilizada sólo para los comandos y/o respuestas FTP

- Conexión de transferencia de datos:

I.- Creada por solicitud para cada transferencia de datos

II.- Destruida al final de cada transferencia de datos

En la figura anterior se muestra el modelo FTP. El cliente FTP realiza una conexión al servidor FTP en el puerto 21, que esta asignado a dicho servidor. Esta conexión se conoce como la conexión de control, la cual envía comandos FTP y recibe respuestas del servidor FTP.

Cuando un archivo se recupere o almacene en el servidor FTP, se establecerá una conexión de datos por separado. Esta conexión se establece en el puerto 20 del servidor FTP. Además, dicha conexión existe sólo mientras dura la transferencia de datos y se destruye después.

IV.4.2. FILTRAR EL TRAFICO EN LA RED TELNET

El tráfico TELNET no requiere ningún mecanismo de contestación hacia un puerto desprotegido en el cliente TELNET; por lo tanto, filtrar sesiones TELNET en el puerto estándar 23 es relativamente sencillo.

TELNET puede utilizarse como un mecanismo general para conectarse a cualquier puerto por medio del siguiente comando:

telnet host [número de puerto]

Si el número de puerto no está especificado, se usa el puerto TELNET predeterminado: el 23. Si la política de seguridad no permite que los usuarios internos establezcan contacto con *servicios como Gopher o WWW*, esta comunicación se controlará con dificultad mediante un número de puerto que no sea el estándar.

Los usuarios externos también pueden utilizar TELNET para probar los servicios especiales que usted proporciona en los puertos TCP, en las máquinas internas que se tienen. Se deben proteger todos esos servicios por medio de enrutadores de selección o barreras de protección.

IV.4.3. FILTRACION DE PAQUETES ICMP

El protocolo de control de mensajes de internet (ICMP) es parte de la capa Internet. El ICMP debe establecerse en todos los módulos de protocolo IP, lo cual significa que todos los anfitriones TCP/IP tienen soporte de ICMP.

ICMP se utiliza para informar acerca de errores en los datagramas IP. No hace más confiable la capa de protocolo IP, sino que señala errores en la capa Internet y deja a un protocolo de capa superior, como TCP, la tarea de hacer más confiable la capa Internet. ICMP emite información acerca de parámetros y errores en la red; también puede utilizarse para hacer un diagnóstico en la red. Asimismo, están definidos los siguientes servicios ICMP:

- Prueba de eco, utilizada para verificar la disponibilidad de un anfitrión TCP/IP (ping).
- Mensajes impresos de tiempo (time stamp), para medir el retraso causado por la red.
- Mensajes de expiración Time to Live (tiempo de vida).
- Mensajes para indicar que el anfitrión o la red destinataria no está disponible.
- Mensajes para señalar errores en los parámetros IP de los encabezados IP.
- Redirigir mensajes para determinar mejores rutas.
- Determinar la máscara de dirección para subred en la red a la que se conecta el anfitrión.
- Mensajes para informar a la fuente que disminuya el envío de paquetes. Esto es un intento de aplicar el control de flujo.

En general los enrutadores emiten mensajes de redirección ICMP hacia otros dispositivos, para informarles acerca de nuevas rutas. Si se permite filtrar estos mensajes dentro de la red interna, un sitio externo puede enviar falsos mensajes de redirección ICMP hacia anfitriones internos y causar estragos en las tablas de enrutamiento de la red interna. Este es un ejemplo de un ataque de "negación de servicio", porque interrumpe las operaciones normales. En general, no hay ninguna razón por la cual una red interna capte los mensajes de redirección ICMP que surgen de una red externa, en especial si se generan desde una red no confiable. Por esta razón, se debería considerar filtrar los paquetes de redirección ICMP que se difunden desde una red externa.

Algunos anfitriones son susceptibles a los mensajes de respuesta de una subred ICMP, aun cuando no hayan realizado ninguna solicitud de subred ICMP. Esto es un error en la implantación de TCP/IP y debe identificarse para prevenir que ocurra.

A pesar de que el servicio de eco ICMP es eficaz para verificar conexiones, si se permite que los sitios externos hagan prueba de eco en la red interna estos sitios pueden obtener un mapa lógico de la red interna que usted tiene.

IV.4.4. FILTRACION DE PAQUETES RIP

Muchas redes internas usan RIP (Protocolo de Información de Enrutamiento), que intercambia información "de lapso" acerca de los destinos de red y de anfitrión en intervalos periódicos de 30 segundos. Estos intercambios RIP se basan en la confianza entre los enrutadores, pues no existe ninguna autenticación sobre estos mensajes. Si un enrutador se equivoca en una ruta, este error puede propagarse con facilidad hacia otros enrutadores, lo que producirá problemas como ciclos de enrutamiento, rutas ineficaces y destinos inalcanzables.

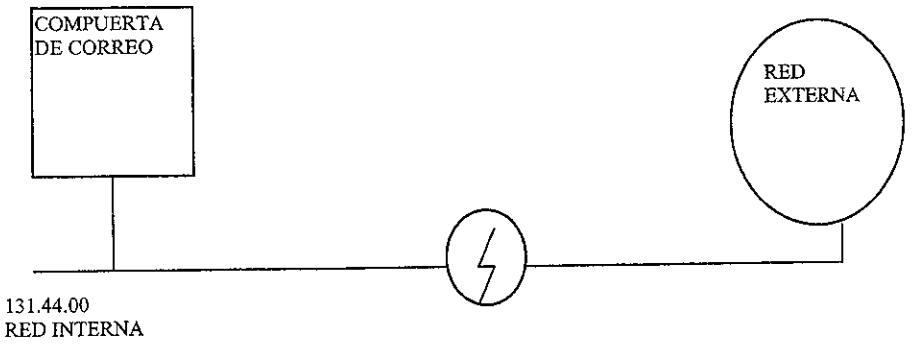
Qué pasaría si alguien proporcionara información falsa sobre el enrutamiento dentro de una red interna, de manera deliberada. Por ejemplo, alguien podría cambiar la información de rutas predeterminadas para los anfitriones, a fin de desviar el tráfico en la red interna hacia un anfitrión de un ataque.

También se necesita inhabilitar el enrutamiento fuente en el enrutador de selección, porque un intruso puede utilizarlos para forzar el envío y la recepción de paquetes mediante el enrutador de selección. Por ejemplo si se usan enrutadores Cisco, se debe inhabilitar el enrutamiento fuente con el siguiente comando:

no ip source-route

IV.4.5. EJEMPLO SOBRE LOS ENRUTADORES DE SELECCIÓN

Ahora se analizará una configuración para enrutadores de selección, la cual tiene como objetivo proporcionar una guía para diseñar e implantar las reglas de los filtros de paquetes.



La figura anterior muestra una red con la siguiente política de seguridad:

1.- Todos los anfitriones de la red interna 131.44.0.0 pueden tener acceso a cualquier servicio TCP en internet.

2.- Los anfitriones externos no pueden conectarse a la red interna, excepto mediante la compuerta de correo en 131.44.1.1. donde pueden tener acceso sólo al servicio de correo SMTP.

3.- Los mensajes ICMP para Internet deben bloquearse.

Se puede expresar esta política de seguridad como una serie de reglas para los puertos internos como sigue:

Número de regla de filtro	**Acción**	**Fuente**	**Puerto fuente**	**Destino**	**Puerto de destino**	**Opciones de banderas de protocolo**	**Descripción**
1	Acepta	131.44.0	*	*	*	TCP	Acepta las conexiones TCP de salida
2	Bloquea	131.44.0	NA	*	NA	ICMP	Bloquea los mensajes ICMP hacia la red externa.
3							

NA = No aplicable

o externos de la siguiente forma:

Número de regla de filtro	**Acción**	**Fuente**	**Puerto fuente**	**Destino**	**Puerto de destino**	**Opciones de banderas de protocolo**	**Descripción**
1	Acepta	*	*	131.44.1.1	25	TCP	Acepta las conexiones TCP de entrada para el anfitrión SMTP
2	Bloquea	*	*	131.44.1.1	*	TCP	Bloquea todos las demás conexiones TCP externas.
3							

Con los enrutadores Cisco, las reglas de los filtros de paquetes para el puerto externo pueden implantarse como se expone a continuación:

regla 1 del filtro de paquetes para el puerto externo:

```
acces-list 101 permit tcp 131.44.0.0  0.0.255.255  0.0.0.0   255.255.255.255
```

regla 2 del filtro de paquetes para el puerto externo:

```
acces-list 101 deny icmp 131.44.0.0  0.0.255.255   0.0.0.0   255.255.255.255
```

Al utilizar los enrutadores Cisco, las reglas de los filtros de paquetes para el puerto interno pueden implantarse como se muestra:

Regla 1 del filtro de paquetes para el puerto interno:

```
acces-list 102 permit tcp 0.0.0.0  255.255.255.255  131.44.1.1   0.0.0.0 eq 25
```

Regla 2 del filtro de paquetes para el puerto interno:

```
acces-list 102 deny tcp 0.0.0.0 255.255.255.255  131.44.0.0.0.0  0.255.255
```

IV.5. ARQUITECTURA Y TEORIA DE LAS BARRERAS DE PROTECCION

En vista de la falta de información de contexto, ciertos protocolos como UDP y RPC no pueden filtrarse con efectividad. Además, en muchas implantaciones de filtración de paquetes, faltan los mecanismos de intervención y alerta. Muchas de estas implantaciones también pueden sufrir malas interfaces de administración y de usuario. La implantación de filtros de paquetes puede requerir un alto nivel de comprensión de los protocolos de comunicación y de comportamiento, cuando se utilizan por diferentes aplicaciones.

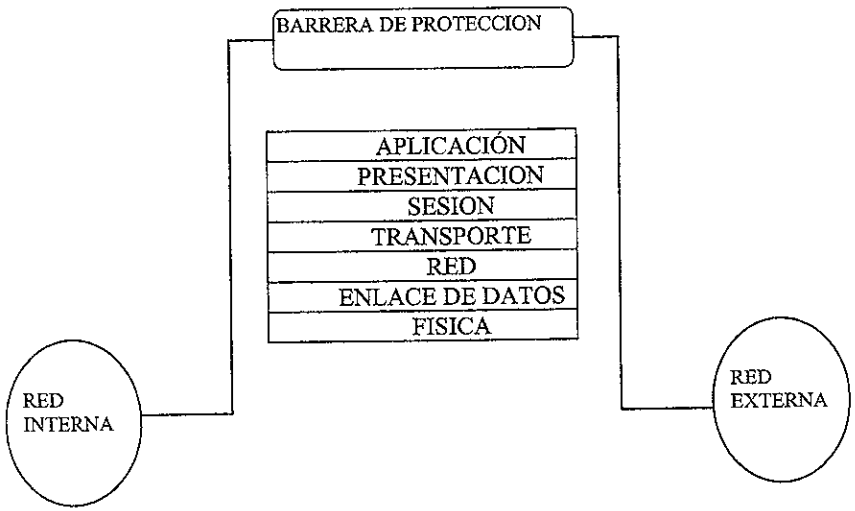
Existen varios métodos para construir una barrera de protección. Las organizaciones con talento en la programación y recursos financieros suficientes, en general prefieren usar un método de "envoltura propia", el cual implica construir soluciones personalizadas de barreras de protección para proteger la red de la organización. Si se ejecuta de manera adecuada, tal vez éste sea el método más eficaz, aunque también el más costoso.

Otras organizaciones prefieren usar los productos comerciales existentes, así como personalizarlos y configurarlos para cumplir la política de seguridad de red de esas organizaciones.

IV.5.1. ANALISIS DE LOS COMPONENTES DE LAS BARRERAS DE PROTECCION

El objetivo de una barrera de protección es proteger una red de otras. En general, la red que se protege es propiedad del usuario (o es su responsabilidad) y la red contra la que se protege es externa, en la que no puede confiarse y desde la cual puede violarse la seguridad. Proteger la red es prevenir que los usuarios no autorizados tengan acceso a datos delicados y permitir que los usuarios legítimos tengan libre acceso a los recursos de la red.

Anteriormente se presentó el modelo OSI como un medio para diferenciar los enrutadores de selección de las barreras de protección. A pesar de que el modelo OSI es un medio para diferenciar la arquitectura de las comunicación y sus capacidades, no siempre se esta consciente de este hecho o del uso del modelo. El término barrera de protección se utiliza comúnmente como un concepto genérico que describe un amplio rango de funciones y la arquitectura de los dispositivos que protegen la red. En ocasiones se emplea dicho término para referirse a casi cualquier dispositivo de seguridad en la red, como un dispositivo de encriptación de programas, un enrutador de selección, etc. En general, una barrera de protección se coloca entre la red interna confiable y la red externa no confiable. La barrera actúa como un punto ahogador que monitorea y rechaza el tráfico en la red a nivel de aplicación, como se indica en el siguiente diagrama:

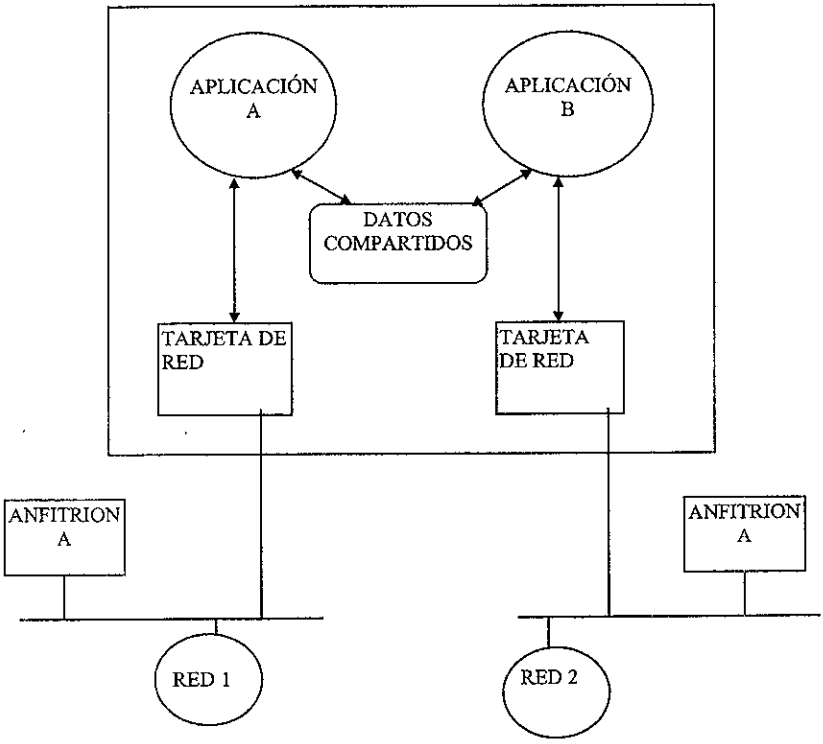


Las barreras también pueden operar en las capas de transporte y de red, aquí revisan los encabezados IP y TCP de los paquetes de entrada y de salida, además de rechazar o pasar los paquetes con base en las reglas programadas del filtro de paquetes.

IV.5.2. ANFITRION DE DOS BASES

En las redes TCP/IP, el término anfitrión de bases múltiples describe un anfitrión que tiene tarjetas múltiples de interfaz de red. En general, cada tarjeta se conecta a una red. Históricamente, este anfitrión de bases múltiples también podía enrutar tráfico entre los segmentos de red. El término compuerta fue utilizado para describir la función de enrutamiento realizada por estos anfitriones. El término enrutador se refiere a esta función, mientras que la palabra compuerta se ha reservado para las funciones correspondientes a las capas superiores del modelo OSI.

Si la función de enrutamiento en el anfitrión de bases múltiples está inhabilitada, el anfitrión puede aislar el tráfico en la red, entre la red a la cual se conecta; aún así, cada red procesará las aplicaciones en los anfitriones de bases múltiples. Además si las aplicaciones lo permiten, las redes también pueden compartir datos. Un anfitrión de dos bases es un ejemplo especial de un anfitrión de bases múltiples con dos interfaces de red y las funciones de enrutamiento inhabilitadas. En la siguiente figura se ejemplifica un anfitrión con dichas características.



El anfitrión A de la red 1 puede tener acceso a la aplicación A en el anfitrión de dos bases. Asimismo, el anfitrión B puede tener acceso a la aplicación B en el anfitrión de dos bases. Además, las dos aplicaciones en los anfitriones pueden compartir datos. Es posible que los anfitriones A y B intercambien información mediante los datos compartidos en los anfitriones de dos bases; sin embargo, el tráfico en la red no se intercambia entre los dos segmentos de red conectados al anfitrión de dos bases.

IV.5.3. SEGURIDAD DE UNA BARRERA DE PROTECCION DE DOS BASES

La mayor amenaza ocurre cuando un intruso obtiene el acceso directo de conexión al anfitrión de dos bases. La conexión siempre se da mediante una aplicación apoderada en el anfitrión de dos bases Las conexiones desde redes externas no confiables requieren una autenticación más rigurosa.

Si el usuario obtiene acceso al anfitrión, la red interna puede ser invadida. Estas invasiones pueden tener cualquiera de las siguientes fuentes:

- Autorizaciones débiles en el sistema de archivos.
- Volúmenes montados en NFS en la red interna.
- Autorizaciones permitidas a utilerías Berkeley r* mediante archivos equivalentes a anfitriones, como .rhost, en directorios base de usuarios para cuentas de usuario que hayan sido comprometidas.
- Programas de respaldo de red que puedan restituir autorizaciones excesivas.
- El uso de scripts administrativos, que no se hayan asegurado de manera adecuada.
- Compresion del sistema a partir de antiguos niveles de revisión del software y notas que no se hayan asegurado de manera adecuada.
- La instalacion de antiguos kernels de sistema operativo que activen el envío IP o la instalacion de versiones de antiguos kernels de sistema operativo con problemas de seguridad conocidos.

Si el anfitrión de dos bases falla, la red interna no tendrá defensa ante futuros intrusos, a menos que el problema se detecte y corrija con rapidez. La variable ipforwarding del kernel de UNIX controla el desempeño del enrutamiento IP. Si el intruso obtiene suficientes privilegios del sistema, podrá cambiar el valor de esta variable y habilitar el envío IP, con lo cual se ignorará el mecanismo de la barrera de protección.

IV.5.4. PRECAUCIONES Y SOFTWARE PARA FORMAR UNA BARRERA DE PROTECCION DE DOS BASES

Además de inhabilitar el envío IP, también se debe eliminar de la barrera de protección de dos bases todos los programas, utilerías y servicios que puedan ser peligrosos en las manos de un intruso. A continuación, se señalan algunos puntos de verificación para las barreras de protección de dos bases en UNIX:

- Eliminar las herramientas de programación: compilación, enlazadores, etc.
- Eliminar los programas con autorizaciones SUID y SGID que no se necesiten o no se comprenda. Si los programas no funcionan, siempre es factible colocar de nuevo aquellos que son esenciales.
- Utilizar particiones en el disco para que, en caso de haber una invasión para llenar todo el espacio del disco en la partición, la invasión quede confinada en dicho espacio.
- Eliminar todas las cuentas especiales y del sistema que no se necesiten.
- Eliminar los servicios de red que no se requieran. Utilizando el comando netstat -a para verificar que sólo se tienen los servicios precisos. Editando los archivos de servicio **/etc/inetd.conf** además de eliminar las definiciones de servicio innecesarias.

A continuación se mencionan varios paquetes disponibles de manera comercial, que permiten construir una barrera de protección.

TCP Wrapper: Es un software gratuito de control de acceso para los sistemas UNIX que desempeña las siguientes funciones básicas:

- Las solicitudes para entrar en el servicio Internet se efectúan mediante el archivo **/etc/inetd.conf.**
- Proporciona un mecanismo para adecuar el acceso a los servicios.

Ambas capacidades pueden utilizarse para dar una sencilla solución a la barrera de protección.

Fire Wall-1: es un producto comercial de compuerta que esta disponible y pertenece a Internet Security Corporation. En la actualidad, el producto se ejecuta en SUN SparcStations.

FireWall-1 utiliza los siguientes métodos para establecer la seguridad en la red:

- Compuerta de aplicación.
- Filtración de paquetes.

La configuración de la compuerta FireWall-1 se realiza por medio de interfaces gráficas como OpenLook para los sistemas operativos SunOS. La compuerta FireWall-1 proporciona estas características:

- Filtración de paquetes segura.
- Poder agregar información de contexto a conexiones sin estado.
- Auditorías y alertas.
- Capacidad para definir y agregar nuevos protocolos y servicios.
- Sesiones Telnet y FTP autenticadas.
- Poder crear canales encriptados.

Un aspecto único de FireWall-1 es que, a pesar de utilizar la filtración de paquetes como su mecanismo básico, esta tarea se realiza en las capas 2 a 7 del modelo OSI. Los protocolos que carecen de información de contexto, como UDP, se manejan al construir información de contexto dentro de la compuerta FireWall-1.

Network Objects Manager: (administrador de objetos en red) se utiliza para definir los objetos específicos en la política de seguridad, los cuales son de los siguientes tipos:

- Redes y subredes.
- Servidores y estaciones de trabajo.
- Anfitriones y compuertas FireWall-1.
- Enrutadores.
- Demonios Internet.

Services Manager: (administrador de servicios) define los servicios que conoce el sistema y que están especificados en la política de seguridad de la red. Todos los recursos de red están filtrados y controlados. FireWall-1 incluye definiciones precargadas para más de 40 servicios TCP/IP e Internet, entre los cuales destacan:

- Servicios estándares: Telnet, FTP, SMTP, etc.

- Servicios Berkeley r*: rlogin, rsh, rcp, rexec, rwho, ruptime, etc.
- Servicios sunRPC: NIS, NFS.
- Herramientas Internet de búsqueda: HTTP (protocolo de transferencia de hipertexto), Gopher, Archie, WAIS (servicios de información de área amplia).
- Servicios IP: ICMP y RIP.
- Servicios de administración: SNMP.

Services Manager puede utilizarse para definir un nuevo servicio al seleccionar el tipo de servicio. En los tipos de servicio se incluyen:

- TCP.
- UDP
- RPC.
- Others (otros): permite definir otros servicios y protocolos que no sean los estándares.

Los servicios pueden agruparse en familias y jerarquías, como el grupo NFS, que incluye al servidor NFS, a Lock Manager y a Mount Programa. Otro ejemplo es Mosaic (WWW), que contiene HTTP, Archie, Gopher y otros.

CONCLUSIONES

Existen puntos básicos en la conservación de la seguridad en una red, no sólo es responsabilidad del administrador del sistema, sino también del usuario, el administrador tiene la responsabilidad de manejar adecuadamente los recursos del sistema, sin embargo el usuario debe saber aprender a manejar sus recursos y seguir las reglas básicas de seguridad.

El administrador debe tener los siguientes mecanismos:

- Examen periódico del sistema para problemas comunes.
- Uso de las herramientas que UNIX ofrece para el resguardo de la información.
- Educación de los usuarios del sistema.
- El administrador debe estar preparado lo mejor posible, debido a que la ignorancia es el problema más común, cuando se presentan problemas en un sistema.
- Hacer una adecuada selección de niveles dentro de los usuarios del sistema.

El usuario por su parte debe:

- Seguir las recomendaciones del administrador.
- Nunca atentar contra la seguridad del sistema.
- Mantener su contraseña en secreto.

El sistema UNIX contiene suficientes herramientas para mantener segura la red, si es que se administra apropiadamente. Para diseñar una red segura se debe analizar todos los factores que forman parte del sistema. Estos factores deben formalizarse en una política de red que ayude a identificar las amenazas, realizar el análisis de riesgo y cómo proteger los recursos.

Es importante entender con exactitud cada elemento que se encontrará dentro de la red, con la política de seguridad se pueden escribir necesidades de seguridad. Este documento es el primer paso dentro de la construcción de las barreras de protección. Sin embargo una seguridad extrema, que no deje trabajar a los usuarios también puede resultar dañina, ya que no se explotarían adecuadamente los recursos del sistema.

Para tener un mecanismo eficaz que controle el tráfico en la red se debe implementar de forma adecuada algún enrutador de selección. Al controlar el tipo de tráfico que existe, se controlará el tipo de servicios que podrían haber dentro de ella. De esta manera se pueden restringir todos aquellos servicios que puedan comprometer la seguridad dentro de la red.

La filtración de paquetes a pesar de tener configuraciones muy variadas, debido a que pueden aplicarse desde una pc, hasta las torres y estaciones de trabajo UNIX, que

utilizan plataformas UNIX basadas en RISC o con los productos enrutadores especiales, aunque estas últimas tengan un elevado costo, siempre serán indispensables para el resguardo de la seguridad en el sistema.

Sin duda alguna UNIX es uno de los sistema más seguros que existen, incluso ha servido como modelo para el desarrollo de otros sistemas operativos, entre otros aspectos en lo referente a la seguridad.

Como conclusión final se debe mencionar que un sistema como UNIX fue hecho principalmente para la comunicación mediante una red, sin olvidar esto, no se debe dejar a la ligera la importancia de la seguridad. UNIX es un sistema que manejado adecuadamente puede resultar una herramienta muy poderosa, sin embargo si la seguridad es descuidada puede ser extremadamente peligrosa, principalmente para la información que se almacena en el sistema. Aunque es cierto que el sistema más seguro es aquel que no esta conectado a ningún otro, no se deben olvidar los beneficios que tenemos al estar comunicados.

BIBLIOGRAFIA

Curry, D.A., **UNIX SYSTEM SECURITY, A GUIDE FOR USERS AND SYSTEM ADMINISTRATORS**, Addison-Wesley, 1992

H. Rosen Kenneth, **UNIX SISTEMA V VERSION 4**, 2da. Edición, McGraw Hill, Madrid España, 1997

Nombela Juan José, **SEGURIDAD INFORMATICA**, Parainfo, Madrid España, 1997

Siyan Karanjit, **INTERNET FIREWALLS AND NETWORK SECURITY**, New Riders Publishing, U.S.A., 1994.

Wood, Patrick, **UNIX SYSTEM SECURITY**, Hayden Book Company, 1986

Garkinkel, S., **PRACTICAL UNIX SECURITY**, O´Reilly & Associates, 1991