

MODULO: 8

Protección y Seguridad

CONTENIDO:

- Conceptos sobre la Protección y Seguridad en un Sistema de Cómputo.
- Mecanismos de integridad y seguridad
- Tipos de Seguridad
- Modelos formales de protección
- Dominios de protección
- Gusanos y Virus
- Seguridad en Procesamiento Distribuido

OBJETIVO DEL MÓDULO: Describir conceptualmente los principios de Protección y Seguridad empleados en Sistema de Cómputos.

OBJETIVOS DEL APRENDIZAJE: Después de estudiar este módulo, el alumno deberá estar en condiciones de:

- Explicar y reconocer las diferencias entre Seguridad y Protección.
- Comprender y explicar la organización y los métodos utilizados en un Sistema de Seguridad.
- Identificar los distintos ataques y amenazas a la seguridad de un Sistema.
- Entender los mecanismos de protección utilizados en los Sistemas de Seguridad.
- Crear las bases de análisis para un Sistema de Seguridad.
- Conocer las funciones del Sistema de Seguridad.
- Conocer y explicar la terminología específica empleada en éste módulo.

Metas:

En este módulo se pretende analizar algunos aspectos sobre la seguridad y la protección que brindan los sistemas operativos tanto para los accesos, la ejecución de programas, la manipulación de datos y el funcionamiento ininterrumpido, sin intromisiones en el sistema. Es obvio que el tema de seguridad abarca, no solo al S.O., sino a las políticas, estrategias, operaciones, etc. que amenazan al normal funcionamiento de un centro de cómputos o una máquina. Una estrategia de seguridad bien diseñada debe abarcar distintos aspectos en forma integral, desde la seguridad física externa como también el abastecimiento de servicios esenciales, el mantenimiento, el normal funcionamiento del sistema, las comunicaciones, los ataques intencionales contra la seguridad, el robo, la violación a reglas, procedimientos y normas preestablecidas, entre otros ingredientes que afectan al normal funcionamiento del sistema. Para construir un buen sistema de seguridad propondremos un modelo genérico en el punto 8.1.10 cuyo objetivo es servir de base de diseño conceptual para implementar protecciones en diversos ámbitos.

8.1. Concepto de Seguridad y Protección

En primer lugar definiremos los dos términos:

* **Protección:**

Es un **mecanismo** que garantiza la integridad de los sistemas y sus recursos, preservándolos contra los usos y accesos indebidos.

Básicamente la palabra protección en cuanto a un sistema computacional, refiere a un mecanismo para controlar el acceso de programas, procesos o usuarios a los recursos definidos por el sistema. Este mecanismo debe proveer un medio para la especificación de los controles a imponer, junto con algunas maneras de aplicarlos. Los sistemas son seguros o inseguros en función de los mecanismos utilizados y de su grado de efectividad para mantener dicha integridad.

* **Seguridad:**

Es una medida de **la confianza** de que la integridad de un sistema y sus datos serán preservados, o sea que los mecanismos de protección implementados respondan a ataques y funcionen correctamente. Por eso, distinguimos entre protección y seguridad en forma tal que seguridad es una medida de la confianza de que la protección actúe bajo la demanda de una acción y preserve la integridad del sistema y sus datos. Seguridad es un tópico mucho más amplio que protección.

En éste módulo examinaremos el problema de la protección en detalle y desarrollaremos un modelo para implementar un sistema de protección.

Los términos seguridad y protección se utilizan a menudo en forma indistinta. Sin embargo, es útil hacer una clara diferenciación o distinción entre los problemas generales relativos a la garantía de que, por ejemplo, los archivos no sean leídos o modificados por personal no autorizado, lo que incluye aspectos técnicos, de administración, normativos, legales y políticos, por un lado y los sistemas específicos del sistema operativo utilizados para proporcionar la protección, por el otro.

Para evitar la confusión, utilizaremos el término seguridad para referirnos al problema general y el término mecanismo de protección para referirnos a los mecanismos específicos del sistema operativo utilizados para resguardar la información de la computadora. Sin embargo, la frontera entre ellos no está bien definida. Primero nos fijaremos en la seguridad; más adelante analizaremos la protección.

Todo sistema de archivos, ya sea en sistemas operativos tradicionales o distribuidos, cuenta con información valiosa con riesgo de ser robada, examinada o extraviada. Por este motivo deben contarse con medidas que aseguren la protección y la seguridad.

La seguridad propiamente dicha pasa por la garantía de que los archivos no sean leídos ni modificados por personal no autorizado.

Las violaciones de seguridad pueden clasificarse en *intencionales o accidentales*, siendo mucho más dificultosa la prevención de las primeras. Las intencionales son esencialmente el robo (lectura no autorizada) y la alteración (por modificación o destrucción) de la información. Con respecto a las accidentales pueden citarse las referidas a actos naturales (incendios, terremotos, etc.), defectos, fallas o errores de hardware o software (mal funcionamiento, bugs, errores de telecomunicación, etc.) y errores humanos (entrada incorrecta de datos, ejecución incorrecta de programa etc.).

Se define como **intruso** a la persona que realiza intencionalmente algún acceso a la información sin estar autorizado.

Para proteger un sistema deben tomarse las medidas a nivel *físico*, en el lugar en donde se encuentra situado el sistema y sus datos, y a nivel *humano* para restringir el acceso de intrusos y a nivel de las *comunicaciones*.

Una debilidad en estos niveles posibilita el acceso al nivel más bajo como resultado de evitar algunas de las medidas propias del sistema operativo.

En la actualidad el hardware posibilita al diseñador el empleo de medidas de seguridad, lo cual trata de ser aprovechado por los sistemas operativos para proteger los recursos, pero no se encuentran en sistemas de computadoras personales como MS-DOS o Macintosh OS. Desafortunadamente, el agregar una característica de seguridad a un sistema funcionando es mucho más difícil que hacerlo en tiempo de diseño previo a la implementación, por lo cual se recomienda incorporarlo en la etapa del diseño.

Por otro lado la seguridad tiene muchas facetas. Dos de las más importantes son el aspecto físico y lógico causales de la pérdida de datos con el consiguiente daño generado por ello y, generalmente relacionado con el equipo computacional y los diferentes tipos de operaciones que realiza el ser humano, intencional o no, que provocan violaciones en la privacidad o robo de información.

Algunas de las causas más comunes de la pérdida de datos son:

1. Actos naturales: Ejemplo: ratas que roen los cables, incendios, inundaciones, terremotos, guerras, revoluciones, choques o movimientos inadecuados que provocan daños materiales en equipos o soportes de información como ser cintas o discos.

2. Defectos o Fallas de hardware o de software: mal funcionamiento de la CPU,

3. La mayoría de estas causas se pueden solucionar con el mantenimiento de los respaldos (back-up) adecuados; de preferencia, en un lugar alejado de los datos originales y un buen mantenimiento periódico en el sistema que contemple los aspectos electromecánicos, electrónicos, servicios, etc. Entre otros.

3. Errores de hardware o de software: errores de telecomunicación o errores en el programa, que generan problemas a la seguridad.

4. **Ataques por Software:** Tales como Virus, Gusanos, etc.

5. Intrusos: El aspecto más complejo es el relacionado con el ser humano. Las acciones y motivaciones para penetrar en un sistema van desde el simple hecho de violar un mecanismo de protección por el solo hecho de violarlo, hasta un interés extremo de robo o hurto de información en beneficio propio. Otro aspecto son los errores propios por descuidos o falta de conocimiento que pueden provocar daños de diversa índole. A continuación analizaremos brevemente, algunos aspectos de los mencionados, sin ser exhaustivos en los mismos.

Los errores humanos pueden ir desde la entrada incorrecta de datos, mal montaje de la cinta o el disco, ejecución incorrecta del programa, pérdida de cintas o discos, operaciones incorrectas, violaciones a procedimientos y normas, etc., de ahí que se trata de un problema interesante. Existen dos tipos intrusos. Los pasivos (solo desean leer archivos o documentos y no están autorizados para hacerlo) y los intrusos activos (son los que desean hacer cambios a los datos y no están autorizados). Si se desea diseñar un sistema seguro contra ellos, es importante determinar el tipo de intruso contra el que se desea disponer de medidas de protección.

Algunas de las acciones comunes son:

A). Torpes: Es impredecible las consecuencias y los daños que puede causar una mala acción de una persona poco habilidosa o deshonesto frente a un sistema. Ejemplo borrar un archivo bajo UNIX.

B). Curiosos (Curiosidad por saber cosas): Generalmente la acción ocurre en las oficinas frente a terminales o computadoras personales conectadas en red, y por la naturaleza humana, alguna persona pretenderá leer documentos reservados o el correo electrónico de los demás o incluso archivos, si no existen barreras que impidan esta acción. Por ejemplo, la mayoría de los sistemas UNIX tienen un mecanismo de protección predefinido que permite restringir la lectura, la escritura y la ejecución de todos los archivos según un conjunto de permisos preestablecidos para cada usuario.

C). Hackers (Conocidos o desconocidos husmeando): Algunos estudiantes inquietos, programadores de sistemas, operadores y demás personal técnico consideran como un reto personal romper la seguridad de algún sistema de cómputo. A menudo son muy calificados y están dispuestos a invertir una gran cantidad de su tiempo en este esfuerzo. Son excelentes investigadores.

D). Delincuentes informáticos (Acciones ilegales): por ejemplo un intento deliberado por hacer dinero. Algunos programadores en bancos han intentado penetrar un sistema bancario con el fin de robarle al banco. Los esquemas han variado desde cambiar el software para truncar y no redondear el interés, para quedarse con una pequeña fracción de dinero, hasta sacar dinero de las cuentas que no se han utilizado en años o introduciendo software (Caballo de Troya) que detecta las claves personales para, luego usarla en provecho propio.

E). Espías (Espionaje comercial o militar): El espionaje indica un intento serio y fundamentado por parte de un competidor o un país por robar programas, secretos comerciales, patentes, tecnología, diseños de circuitos, planes de comercialización, estrategias empresariales, archivos en soportes informáticos o papel, etc. A menudo, este intento implica la captura de información transportada en cables o el uso de un sofisticado equipo de comunicaciones con antenas dirigidas hacia la computadora con el fin de recoger su radiación electromagnética y de esta forma lograr el objetivo de apropiarse indebidamente la información ajena.

Debe quedar claro que el intento por mantener a los servicios secretos de inteligencia lejos de los secretos militares es un poco distinto del intento por evitar que los estudiantes inserten un mensaje gracioso indicando de que penetraron en el sistema y visitaron la cuenta de su docente. La cantidad de esfuerzo que alguien pone en la seguridad y en la protección depende claramente de quien se piensa sea el enemigo o la amenaza.

Otro aspecto del problema de la seguridad es la privacidad: la protección de las personas respecto del mal uso de la información en contra de uno mismo por un tercero. Esto genera conflictos que incluyen aspectos legales y morales en cuanto al uso de información privada.

¿Debería el gobierno tener expedientes de cada persona, con el fin de localizar a los defraudadores X, donde X es el "seguro social" o "el impuesto", según la política?.

¿Debe la policía poder investigar todo de todos para detener al crimen organizado?. ¿Tienen derechos los empleados, las compañías aseguradoras o la Dirección General Impositiva, de investigar las cuentas bancarias privadas u otra información relacionada con la solvencia personal?. Esto se conoce como "cruce de Bases de

Datos" y pueden vulnerar los derechos constitucionales de las personas. ¿Que ocurre cuando estos derechos entran en conflicto con los derechos individuales?.

Todos estos aspectos son importantes pero escapan a los objetivos de este libro.

Requerimientos de Seguridad

Todo sistema que desee tener en cuenta este aspecto, debe apoyarse en cuatro requerimientos básicos:

- **privacidad:** solo las personas autorizadas deben tener acceso a los componentes del sistema
- **integridad:** solo las personas autorizadas pueden modificar las distintas partes del sistema.
- **disponibilidad:** solo las personas autorizadas pueden disponer del sistema
- **autenticidad:** solo las personas autorizadas pueden acceder al sistema

Algunos mecanismos de protección ya estudiados en los diversos módulos previos:

- Instrucciones privilegiadas / comunes.
- Protección de memoria
- Operaciones de E/S realizadas por el S. O.
- Permisos de accesos a archivos
- Deadlock
- Etc.

8.1.1. Concepto de Política y Mecanismo:

Es obvio que las decisiones políticas gobiernan y gobernarán los aspectos de la seguridad sin considerar los aspectos económicos. Siempre se justificará políticamente el empleo de la seguridad para un fin determinado.

Es importante considerar la separación entre **política y mecanismo**:

POLÍTICAS	MECANISMOS
• Aseguran la operación eficiente y ordenada del sistema. Los procesos están sujetos a políticas que gobiernan el uso de recursos.	• Permite aplicar las políticas que gobiernan la utilización de los recursos.
• El rol de la protección es proveer un mecanismo para reforzar las políticas que gobiernan el uso de recursos.	• Los mecanismos refuerzan a las políticas.
• Determinan que es lo que se hará.	• Determinan como hacer algo.
• Pueden cambiar de tiempo en tiempo, y en el peor de los casos, un cambio en la política requerirá un cambio en algún mecanismo.	• Los mecanismos generales podrían ser más deseables, porque un cambio en la política solo requerirá la modificación de una serie de tablas o parámetros.

Tabla 8.1 Políticas y mecanismos.

8.1.4. Política de Seguridad

Se implementan mediante **procedimientos escritos y procesos** que, por ejemplo, especifican:

- Cómo se introduce y saca información en el sistema.
- Quién está autorizado para acceder a un dado tipo de información y bajo que condiciones
- Cuáles son los flujos permitidos de información en el sistema.
- Cuáles son los límites (de restricción, de consulta, de uso, inferencias, estadísticas, etc.)
- etc.

8.1.3. Principios de las Políticas de Seguridad

Los principios, entre otros, se basan en:

- **Mínimo privilegio:** Cada sujeto debería tener permitido el acceso únicamente a la información esencial necesaria para completar las tareas que el sujeto está autorizado a realizar. Por ejemplo: los empleados que manejan la contabilidad del alumnado no deberían tener acceso a los registros de notas de los alumnos, y los docentes no deberían tener permitido acceso a los datos de pagos efectuados por los alumnos.

- **Separación de deberes** (controles cruzados por 2 o más personas): Si hay un conjunto de operaciones que puede poner en riesgo a una organización, debería exigirse que dos o más personas con intereses contrapuestos estuviesen implicadas en ellas.
- **Rotación de roles**: Las operaciones delicadas no deberían ser confiadas permanentemente al mismo personal. Una cierta rotación en las responsabilidades es más probable que descubra incorrecciones.
- etc.

8.1.4. Categorías Básicas de las Políticas de Seguridad

1) **Control de acceso discrecional (CAD):**

- Estas políticas son generalmente definidas por el propietario de los datos, quien puede transferir derechos de acceso a otros usuarios.
- El creador de un archivo puede especificar los derechos de acceso de los usuarios.
- Esta forma de control de acceso es habitual en sistemas de archivos.
- Es vulnerable al ataque del Caballo de Troya (explicado con más detalle a lo largo de este módulo), en donde los intrusos se hacen pasar por usuarios legítimos.

2) **Control de acceso obligatorio (CAO):**

- Las restricciones de acceso obligatorio no están sujetas a la discreción del usuario y por tanto limitan el daño que un Caballo de Troya puede causar.
- Los usuarios se clasifican de acuerdo con niveles importancia, de autoridad o autorización.
- Los datos se clasifican en clases de seguridad según el nivel de confidencialidad, y se definen reglas estrictas respecto a qué nivel de autorización se requiere para acceder a los datos de una clase de seguridad específica.

Las políticas de seguridad que tienen en cuenta amenazas tanto externas como internas son muy importantes en entornos que gestionan datos críticos, ya que la mayoría de las incorrecciones las originan usuarios internos.

8.1.5. Objetivos de la Protección

- Prevenir la violación de una restricción de acceso a usuarios y a los recursos mediante adecuados controles.
- Detectar errores y virus en interfases y componentes de un sistema.
- Preservar la integridad de los recursos compartidos.
- Permitir el uso consistente de los recursos de un sistema de acuerdo a los permisos para ese uso.
- Facilitar medios para el uso y distinguir entre el uso autorizado y no autorizado.
- Controlar el acceso de programas o usuarios a los recursos definidos por un sistema.
- Proveer los medios para especificar los controles a ser impuestos y las medidas de refuerzos necesarias en caso de ataques.

8.1.6. Seguridad a través del Sistema Operativo

Muchas veces nos preguntamos frente a un sistema computacional si existe la seguridad o si estamos seguros de:

- haber logrado un acceso al computador?
- que el S. O. es confiable?
- que el computador hace lo que queremos?. etc.

Las respuestas se las dejamos al lector.

Por otro lado podemos buscar las causas y analizar los siguientes problemas:

Los problemas

- La Computación es cada vez más accesible (masificación).
- El tiempo compartido y acceso remoto como ser redes, accesos y sus problemas, (vulnerabilidad de las comunicaciones).
- Las computadoras administran nuestras actividades profesionales (correspondencia, transferencias de fondos, actividades bancarias, correo, etc.), cada vez dependemos más de los sistemas informatizados.
- Se prestan para el espionaje o el sabotaje.

Todo ello nos lleva a analizar las amenazas a la seguridad en los sistemas.

8.1.7. Niveles de Seguridad en Informática

Una de las primeras preguntas que surgen cuando se comienza a abordar la problemática de la seguridad informática es si existe alguna estandarización de los niveles de seguridad que permita tener una idea clara de cuáles son las necesidades de una determinada empresa.

El estándar de niveles de seguridad más utilizado internacionalmente es el "Orange Book" (libro naranja), desarrollado de acuerdo con los estándares de seguridad en computadoras del Departamento de Defensa de los Estados Unidos. En este estándar se usan varios niveles de seguridad para proteger al hardware, al software y a la información de un ataque.

Los niveles que se enumeran a continuación describen diferentes tipos de seguridad física, autenticación del usuario y confiabilidad del software.

Nivel D1: El nivel D1 es la forma más elemental de seguridad disponible. Este estándar parte de la base de que todo el sistema no es confiable.

No existe protección para el hardware; el sistema operativo es inestable y no hay autenticación con respecto a los usuarios y sus derechos para tener acceso a la información que se encuentra en la computadora. Por lo general, este nivel de seguridad se refiere a los sistemas operativos como MS-DOS, Windows 3.x y System 7.x de Macintosh. Estos sistemas operativos no distinguen entre usuarios y carecen de un sistema definido para determinar quién es el usuario.

Tampoco tienen control sobre la información que puede introducirse en los discos rígidos.

El nivel C tiene dos Subniveles: C1 y C2.

Nivel C1: El nivel C1, o sistema de protección de seguridad discrecional, describe la seguridad disponible en un sistema típico Unix. Considera que existe algún nivel de protección para el hardware, aunque todavía es posible vulnerarlo. Los usuarios deberán identificarse con el sistema por medio de un nombre de registro del usuario y una contraseña. Esta combinación se utiliza para determinar qué derechos de acceso a los programas e información tiene cada usuario. Estos derechos de acceso son permisos para archivos y directorios. Estos controles de acceso discrecionales habilitan al dueño del archivo o directorio, o al administrador del sistema, a evitar que determinadas personas tengan acceso a los programas e información de otras personas. Sin embargo, la cuenta de la administración del sistema está restringida a realizar cualquier actividad. En consecuencia, el administrador de sistema tiene en sus manos gran parte de su seguridad.

Por otro lado, algunas de las tareas de administración del sistema sólo pueden realizarse al registrarse el usuario conocido como root (raíz). Con la centralización de los actuales sistemas de computadoras, no es raro entrar en una organización y encontrar a dos o tres personas que conocen la contraseña para tener los atributos de root.

Nivel C2: El segundo Subnivel, C2, fue diseñado para solucionar las debilidades del C1. Además de las características de C1, el nivel C2 incluye características de seguridad adicional que crean un medio de acceso controlado. Este medio tiene la capacidad de reforzar las restricciones a los usuarios en la ejecución de algunos comandos o en el acceso a algunos archivos basados en permisos y en niveles de autorización.

Este nivel de seguridad requiere auditorías del sistema. La auditoría se utiliza para mantener los registros de todos los eventos relacionados con la seguridad, como aquellas actividades realizadas por el administrador del sistema. La auditoría requiere la autenticación adicional para asegurarse de que la persona que ejecuta el comando es realmente quien dice ser. La desventaja de la auditoría es que requiere recursos adicionales.

Con el uso de autorizaciones adicionales es posible que los usuarios de un sistema C2 tengan la autoridad de realizar tareas de manejo de sistema, sin necesidad de una contraseña root. Esto mejora el rastreo de las tareas relativas a la administración, puesto que cada usuario realiza el trabajo en lugar del administrador del sistema.

Estas autorizaciones adicionales no deben confundirse con los permisos SGID y SUID que se pueden aplicar a un programa. Estas son autorizaciones específicas que permiten al usuario ejecutar comandos específicos o tener acceso a las tablas de acceso restringido.

Nivel B1: El nivel B de seguridad tiene tres niveles. El B1, o protección de seguridad etiquetada, es el primer nivel que soporta seguridad de multinivel, como la secreta y la ultrasecreta. Este nivel parte del principio de que un objeto bajo control de acceso obligatorio no puede aceptar cambios en los permisos hechos por el dueño del archivo.

Nivel B2: El nivel B2, conocido como protección estructurada, requiere que se etiquete cada objeto.

Los dispositivos como discos rígidos, cartuchos o terminales podrán tener asignado un nivel sencillo o múltiple de seguridad.

Este es el primer nivel que empieza a referirse al problema de un objeto a un nivel más elevado de seguridad en comunicación con otro objeto a un nivel inferior.

Nivel B3: El nivel B3, o nivel de dominios de seguridad, refuerza a los dominios con la instalación de hardware. Por ejemplo, el hardware de administración de memoria se usa para proteger el dominio de seguridad de acceso no autorizado o la modificación de objetos de diferentes dominios de seguridad. Este nivel requiere que la terminal del usuario se conecte al sistema por medio de una conexión segura.

Nivel A: El nivel A, o nivel de diseño verificado, es el más elevado. Incluye un proceso exhaustivo de diseño, control y verificación.

Para lograr este nivel de seguridad, todos los componentes de los niveles inferiores deben incluirse; el diseño requiere ser verificado en forma matemática. Además, es necesario realizar un análisis de canales encubiertos y de distribución confiable.

8.1.8. Amenazas a la Seguridad

Los distintos ataques posibles actúan sobre un determinado ámbito, o el sistema. Las herramientas disponibles para solucionarlos también deben actuar sobre el mismo ámbito. Esto es descrito en la figura siguiente.

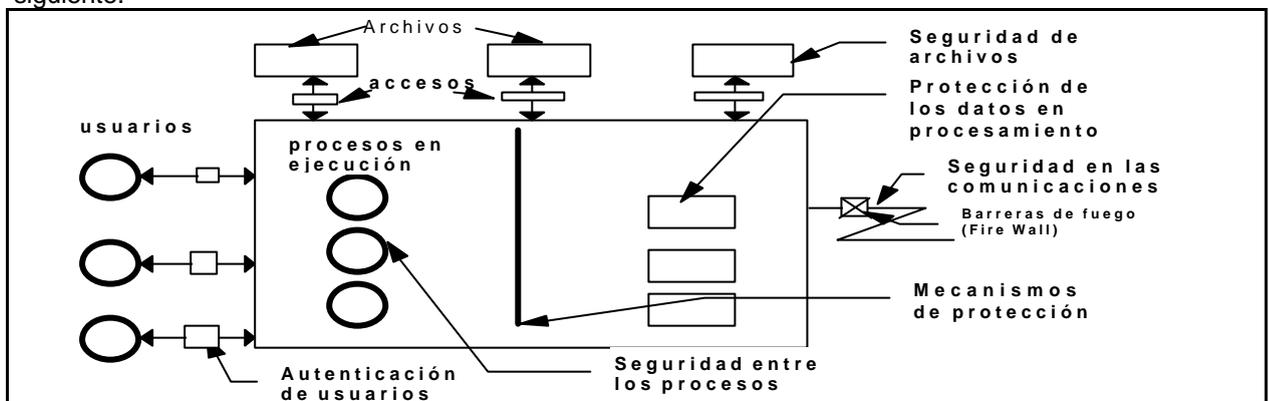


Figura 8.01 ámbitos de la seguridad

Las consecuencias de no aplicar seguridad pueden ser:

1. Revelación no autorizada de la información.

- Puede dar lugar a brechas en la privacidad y a pérdidas tangibles o intangibles para el propietario de la información.
- Dependiendo de la naturaleza de la información en cuestión, las consecuencias del abuso pueden ir desde una simple inconveniente hasta pérdidas catastróficas. Ver Fig. 8.03

2. Alteración o destrucción no autorizada de la información.

- Es potencialmente peligrosa ya que puede afectar a información irrecuperable.

3. Uso no autorizado de servicios.

- Puede dar lugar a pérdida de beneficios para el proveedor del servicio.
- Puede ser explotada (al igual que otras formas de penetración) para obtener acceso ilegal a la información.
- Una penetración sin malas intenciones puede generar mala publicidad y disuadir a clientes potenciales.

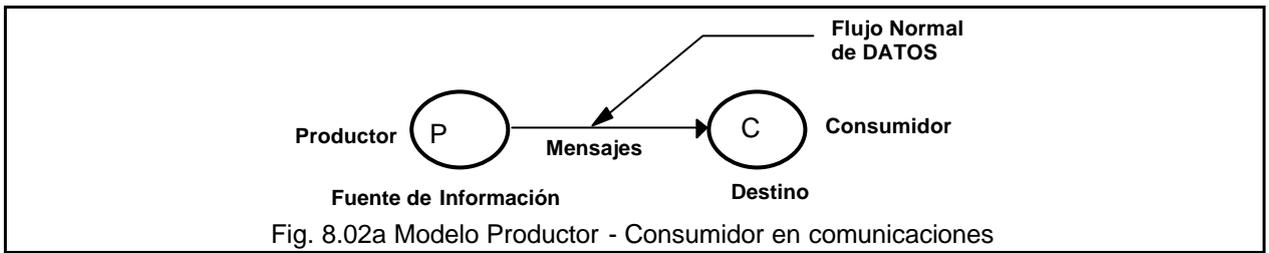
4. Denegación de servicios a usuario legítimos.

- Da lugar a la pérdida parcial o completa del servicio prestado a los clientes legítimos.
- Una forma de denegación de servicio viene representada por los programas que se autoproducen y propagan, llamados gusanos informáticos (que se discuten más adelante en este módulo con más detalle).

Tipos de amenazas

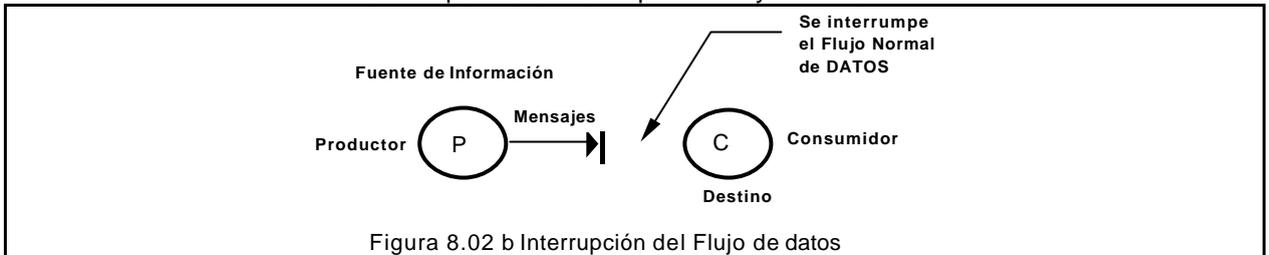
La mejor forma de caracterizar las amenazas de un sistema es verlo a este como un proveedor de información. En general hay un flujo de información entre una fuente (puede ser un archivo o una zona de memoria) y un destino (un usuario u otro archivo) como se indica en la figura 8.2 a.

Las distintas formas en que se da una transacción se muestran en los siguientes esquemas en donde se debe controlar el acceso y el flujo de la información desde un productor (P) a un consumidor (C). Observemos cuando se desplaza la información de un lugar a otro:



1) Primer caso: **Interrupción**

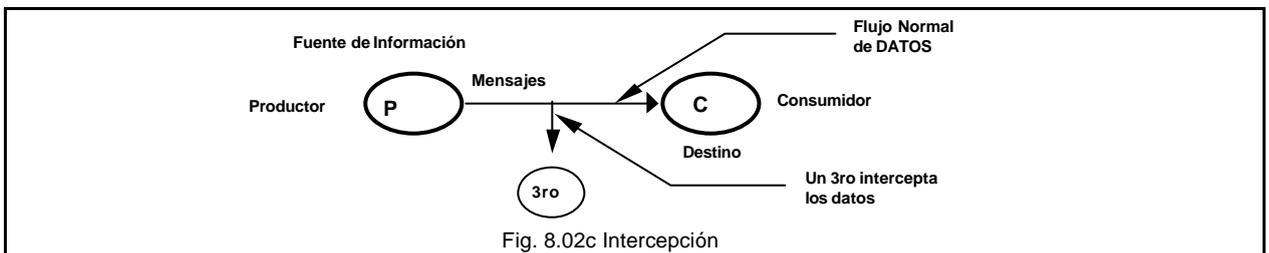
Se trata de la comunicación esperada entre un productor y un consumidor.



Se refiere a la disponibilidad. Uno de los componentes del sistema es destruido o no está disponible por alguna falla. En este caso se amenaza la disponibilidad porque se destruye la información o un modulo del hardware. Ejemplo: destrucción de un disco duro, corte de una línea de comunicación, etc.

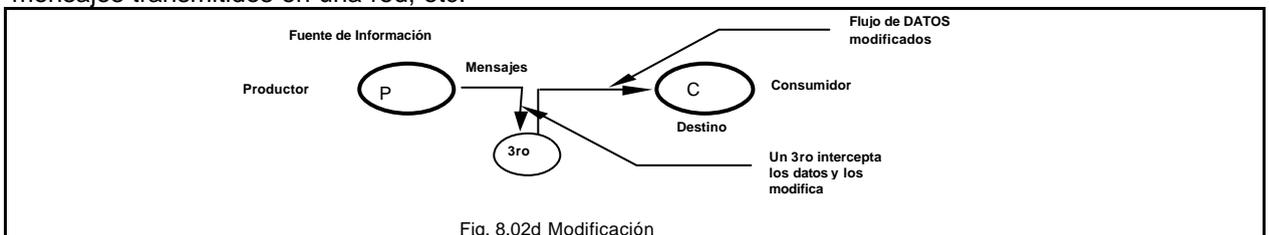
2) Segundo caso: **Intercepción**

Intercepción se produce cuando se logra mediante un acceso no autorizado y afecta a la Privacidad. Un tercero no autorizado accede a un recurso. El tercero puede ser un programa, una persona o una computadora. Ejemplo: intentos de capturar información en una red, copia ilegal de archivos, etc.

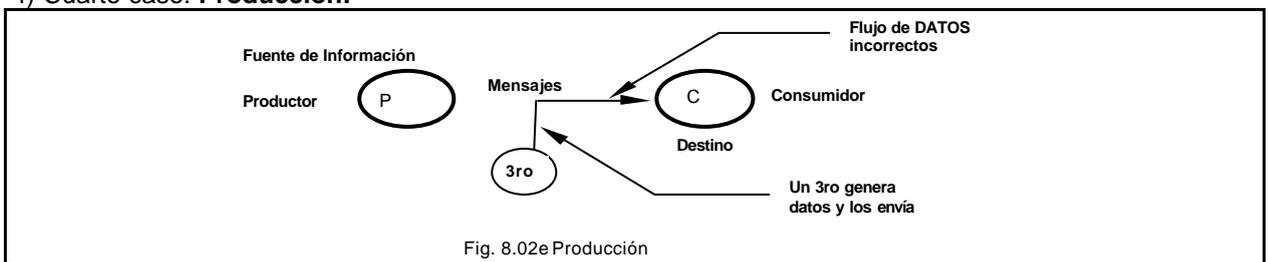


3) Tercer caso: **modificación**

Un tercero no autorizado no solo accede al recurso sino que interfiere en la normal transmisión. Esto amenaza la integridad pues un tercero viola accesos y modifica la información. Ejemplo: Cambio de datos en un archivo, alteración de un programa para que funcione de otra manera, modificación de mensajes transmitidos en una red, etc.



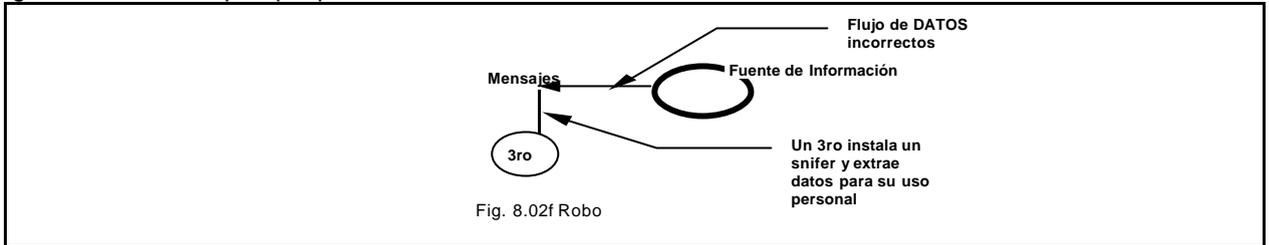
4) Cuarto caso: **Producción:**



Corresponde a otra amenaza a la integridad. El tercero en este caso, introduce objetos en el sistema generando el consiguiente daño al insertar un objeto anormal. Ejemplo: agregado de registros a un archivo o inserción de mensajes no genuinos en una red o el caso de virus, o mensajes espurios.

5) Quinto caso: **Robo:**

Corresponde a una amenaza a la privacidad. El tercero introduce un software (Snifer) en el sistema generando un ataque que permite extraer información del sistema



En base a estos cinco casos la pregunta resultante es ¿Donde aplicar la seguridad?. Genéricamente hablando se puede determinar en un sistema computacional cuatro áreas posibles de recibir ataques: Hardware, Software, datos y las líneas de comunicaciones y en base a las figuras 8.2b a 8.2e podemos observar las siguientes amenazas que se amplían en la tabla 8.2:

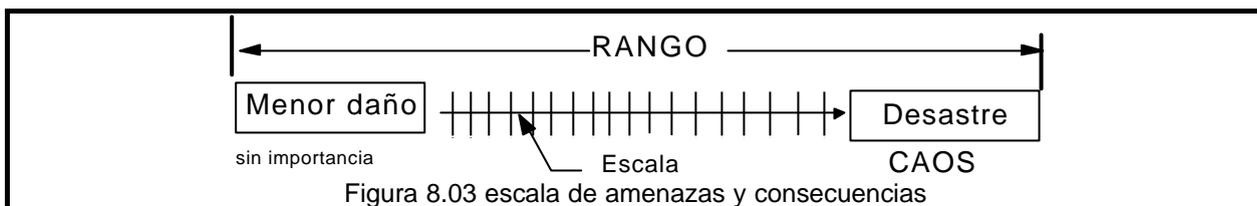
Hardware	Interrupción
Software	Interrupción (borrado) Intercepción Modificación
Datos	Interrupción → pérdidas Intercepción → captura, análisis Modificación Producción
Líneas de comunicación	Interrupción (pérdidas) Intercepción (Captura - análisis) Modificación Producción

Tabla 8.1 Áreas y sus posibles causas de ataque.

ELEMENTO	PRIVACIDAD	DISPONIBILIDAD	INTEGRIDAD
Hardware		- Robo - Desconexiones - Daños - Mantenimientos.	
Software	Copias no autorizadas acceso a objetos (ocultamiento)	- Borrado intencional - Accesos denegados	- Modificación de programas - Virus que dañe
DATOS	- Lectura no autorizada - Análisis estadísticos que revelan direccionamientos.	- Destrucción de mensajes - Borrado de mensajes	- Archivo existentes son modificados o son cambiados por nuevos
LÍNEA DE COMUNIC	- Lectura de Mensajes - Los patrones de Mensajes son observados	- Destrucción de mensajes - Borrado de mensajes	Mensajes: + modificados + demorados + reordenados + duplicados + falseados

Tabla 8.2 Amenazas a la seguridad.

En la tabla 8.2 se observan los cuatro elementos o áreas y los ataques posibles a cada uno de ellos. Las consecuencias van desde un rango de menor daño hasta un desastre generados por los ataques.



Otra forma de caracterizar las amenazas es definir a quién están afectando. Tal es el caso de amenazas a programas o al sistema en sí.

Amenazas a programas (Software)

Esto se da en ambientes en donde un programa escrito por un usuario puede ser usado por otro, lo que conlleva a esperar un uso indebido. Los ejemplos que se describen a continuación serán ampliados más adelante en este módulo por su importancia.

Los siguientes son métodos comunes de conseguir ataques a programas:

Caballo de Troya:

Consiste en modificar un programa normal para que realice cosas adversas además de su función usual y arregle las cosas para que el usuario utilice la versión modificada. Ejemplo: En el sistema operativo MULTICS se podían robar archivos obteniendo una copia del código fuente del editor, modificándolo para que realice esta tarea manteniendo su funcionamiento aparente. Una vez compilado, si se lo leía en el directorio bin de la víctima se podía robar todos sus archivos.

Puertas Traseras (Trap door):

Se denomina de esta manera a un agujero dejado por un programador en el software, lo que posibilita su acceso en un determinado momento. Una puerta trasera es un punto de entrada secreto, dejado por los implementadores del sistema para saltarse los procedimientos normales de seguridad. La puerta trasera puede haberse dejado con fines maliciosos o como parte del diseño; en cualquier caso, son un riesgo.

Esta técnica permite a un intruso volver a entrar en un sistema sin ser detectado teniendo como objetivos del ataque: volver a entrar incluso si se cambian las claves, dejando el menor número de pistas posibles y lo más rápido posible. Esto se vio ejemplificado en la película "Juegos de Guerra", en donde un estudiante ingresaba a una poderosa computadora y creía estar jugando a un juego cuando en realidad se trataba del sistema de defensa de los EE.UU. Han habido casos reales de programadores que han sido arrestados por adjudicarse cierto redondeo en cuentas bancarias. Debido al elevado número de transacciones, las cifras que se manejaban eran millonarias. Un trap door sofisticado puede estar incluido en el compilador, dejando la brecha cada vez que se compila. Para encontrar alguno de ellos es necesario recorrer todos los archivos, cosa que no se hace frecuentemente.

Amenazas a sistemas:

La mayoría de los sistemas operativos permiten que los procesos creen procesos hijos. En esos ambientes es posible crear situaciones en donde se produce el mal uso tanto de los recursos como por ejemplo los archivos. Los métodos para lograr esto son:

Gusanos (worms):

Es un programa que tiene la capacidad de reproducirse por sí mismo valiéndose de las posibles fallas que puede encontrar en algún sistema (ejemplo: UNIX). Utiliza los recursos del sistema e impide a otros usuarios que hagan uso de ellos.

El caso más famoso de un gusano fue el liberado por un estudiante norteamericano en Internet y que se describe en el punto 8.10.2 de este módulo.

Virus:

Se diferencia del gusano en que no es un programa completo sino un fragmento, el cual se ejecuta cada vez que el programa infectado hace lo mismo.

Existen diversas técnicas para hacer esto. Una de ellas consiste en literalmente "engancharse" del programa a infectar por medio de añadirse a su código y desviarse hacia el mismo cada vez que el programa se ejecuta. Al iniciarse el programa infectado inicia la búsqueda de archivos no infectados para hacer lo propio. También existen virus que no solo se contagian sino que destruyen valiosa información. Para eliminar virus se cuenta con programas especialmente diseñados, aunque los casos extremos indiquen que debe formatearse el disco rígido y una posterior adquisición de software seguro (esto se debe a que usualmente los virus se distribuyen en programas de dominio público en boletines y BBSs). Ejemplo: El virus Michelángelo estaba programado para destruir todo el 6 de Marzo de 1992, fecha del cumpleaños 517 del artista.

Este tema lo ampliaremos en el punto 8.10.1 de este módulo.

Ataques genéricos a la seguridad:

Existen equipos especialmente diseñados para probar sistemas y ver si existe alguna falla. He aquí una lista de medidas a tener en cuenta para revisar ataques de otros tipos no solo informáticos:

1. Solicitar páginas de memoria, espacio en disco o cintas en desuso para leerlas y ver si el sistema efectivamente las borra antes de asignarlas nuevamente. Muchos sistemas no las borran después de haber sido usadas entonces podrían contener interesante información escrita por el usuario anterior.
2. Intentar llamadas al sistema inválidas, o bien llamadas validas con parámetros inválidos, o incluso llamadas validas con parámetros validos pero no razonables. Estos intentos son para confundir al sistema y muchos sistemas pueden ser confundidos con facilidad.
3. Iniciar la conexión al sistema y oprimir entonces DEL, REBOOT o BREAK a la mitad de la secuencia de acceso. Tratar de hacer un break del booteo cuando se está en la fase de inicialización en ciertos sistemas, el programa de verificación de la contraseña quedará eliminado y se considerara un acceso exitoso.
4. Intentar modificar las complejas estructuras del sistema operativo que esta en el espacio del usuario. En muchos sistemas, para abrir un archivo, el programa construye una enorme estructura de datos (File Control Block), la cual contiene nombre del archivo y muchos otros parámetros, la que se transfiere al sistema. Al leer o escribir en un archivo, el sistema analiza a veces la propia estructura. La modificación en estos campos puede causar estragos en la seguridad.
5. Tratar de engañar al usuario con un programa que simule el logeo y grabe su contraseña. Un programa que haga aparecer el "login" en la pantalla y que después desaparezca es fácil de construir y grabar la contraseña tipeada en el teclado en un archivo también es muy fácil.
6. Realizar todo lo que en los manuales diga que no debe hacerse (a modo de prueba). Si los manuales dicen "no lleve a cabo X" es porque se ha verificado una debilidad, entonces con intentar tantas variaciones de X como sea posible se podrá descubrir esa debilidad.
7. Convencer a un programador del sistema para que modifique el sistema, con el fin de que evite ciertas verificaciones vitales de seguridad para un dado usuario con nombre de acceso. Este ataque se conoce como una puerta trasera (trap-door) o puertas cepo.
8. Si todo falla, entonces el atacante puede valerse de otras mañas, por ejemplo, encontrar a la secretaria del director del centro de cómputo y engañarla, sobornarla o simplemente conquistarla. Es probable que la secretaria tenga un fácil acceso a todo tipo y clase de información maravillosa y que por lo general se le paga poco. No subestimar los problemas que pueda causar el personal en su centro de cómputos!!!.
9. Alejarse de la Terminal con sesión abierta: Es normal que el usuario deje su puesto de trabajo dejando la terminal conectada al sistema. Por ejemplo atender una llamada telefónica o buscar algo fuera del lugar de trabajo. Esto permite el acceso de cualquier persona a lugares del sistema que pueden brindar información muy importante para lograr posteriores ingresos al sistema.
10. Revolver la basura: Genere un programa que pueda leer los bloques libres de memoria. En muchos sistemas operativos no se eliminan los restos usados, sino que se colocan en puntos importantes.

Monitoreo de Amenazas (Vigilancias):

Es común que en un sistema se realicen periódicas inspecciones a fin de constatar si ha habido intrusos últimamente que provocaron violaciones a procedimientos. Un ejemplo de esto es registrar la cantidad de veces que una contraseña fue incorrectamente ingresada por un usuario, para determinar si se trataba de meras equivocaciones o si se estaba tratando de adivinar la clave. Otro método consiste en llevar un archivo de registro auditor, en el que se verifican todas las acciones realizadas por todos los usuarios, de manera de individualizar a quien pertenece la cuenta del que originó el desastre.

La siguiente es una lista de aspectos que deberían tenerse en cuenta al verificar un sistema:

- ◆ Contraseñas cortas o fáciles de adivinar
- ◆ Programas que provengan de PIDs (identificación de usuarios) no autorizadas
- ◆ Programas no autorizados en directorios del sistema
- ◆ Procesos que tardan más de lo esperado
- ◆ Protecciones indebidas en directorios de usuario y del sistema
- ◆ Protecciones indebidas en los componentes más importantes del sistema (archivo contraseña, drivers, kernel, etc.)
- ◆ Entradas peligrosas en el camino de búsqueda ejemplo: Caballo de Troya
- ◆ Cambios en los programas del sistema. Esto se puede lograr manteniendo una lista con los valores de checksum de cada archivo y cerciorándose de que es el mismo en cada entrada al sistema cuando el archivo no es modificado.
- ◆ backups o copias de resguardo y su almacenamiento.

8.1.9. Objetivos de la Seguridad y la Protección de un Sistema.

- Funcionamiento correcto ininterrumpido y sin intromisiones en el centro de cómputo.
- Asegurar la integridad de la información del Centro de Cómputo.
- Prevenir y eliminar amenazas potenciales.
- Disponer de un Sistema seguro que mantenga la integridad, Disponibilidad y privacidad de los datos.
- Manipular Datos del sistema, correctos, disponibles y privados.
- Mantener la integridad de datos:
 - 1) Protección frente a modificaciones no autorizadas.
 - 2) Resistencia a la penetración.
 - 3) Protección frente a la modificación no detectada de datos.

8.1.10. ¿Porque el S. O. es el responsable de la Seguridad Interna?

Controla el acceso y uso de los recursos.

- Los programas de aplicaciones solicitan el acceso a los recursos al S.O. y él administra y controla su uso.
- Permite compartir los recursos, pero compartir y proteger son objetivos contradictorios que generan conflictos, lo cual ya fue discutido en módulos anteriores.

8.1.11. Justificación de la Seguridad y Protección.

La justificación se basa en la formulación de una serie de preguntas cuyas respuestas serán los requisitos a tener en cuenta. Por ejemplo:

- ¿Se requiere un sistema seguro? (Principio de la definición)
- ¿ Que se pretende proteger? (Selección de Niveles de seguridad vs. valor de los recursos a proteger)
- ¿Cuáles son los Requisitos de seguridad? (Requerimiento del sistema, mecanismos capaces de satisfacerlos y el Factor humano en cuanto a motivación y niveles de capacidades)
- ¿Cuáles son las amenazas y las consecuencias? (Requisitos de pensar sobre todo en la Identificación de amenazas potenciales y sus Consecuencias)
- ¿Cuáles son las Características del entorno de la instalación?
- ¿ Se justifican los costos? (Fundamentación de la necesidad = Factor determinante para diseñar un sistema de seguridad)
- ¿Y las comunicaciones?. Hoy en día nada funciona sin las comunicaciones...

"Es imposible diseñar medidas de seguridad efectivas si no se intenta primero enumerar y comprender las posibles amenazas contra la seguridad" H.M. Deitel.

8.2. Diseño: Principio de los Mecanismos.

Saltzer y Schroeder (1975) han identificado varios principios generales que se pueden utilizar como una guía para en diseño de sistemas seguros. A continuación daremos un breve resumen de sus ideas (basado en la experiencia con MULTICS).

- **Diseño abierto:** la seguridad del sistema no depende que el diseño del mecanismo sea secreto, sino su uso confidencial.
- **Privilegios - Definición** (procedimientos para otorgar y revocar los permisos, derechos y privilegios de los usuarios).
- **Economía de mecanismos** (pocos, seguros y simples).
- **Aceptabilidad:** no debe interferir con el normal trabajo del usuario y fácil de usar en el instante que se necesite.
- **Inmediato:** los mecanismos tiene que actuar bajo demanda

En primer lugar, **el diseño del sistema debe ser público**. Pensar que el intruso no conocerá la forma de funcionamiento del sistema es engañar a los diseñadores.

En segundo lugar, **el estado predefinido (por default) debe ser el de no acceso**. Los errores en donde se niega el acceso valido se reportan más rápido que los errores en donde se permite el acceso no autorizado.

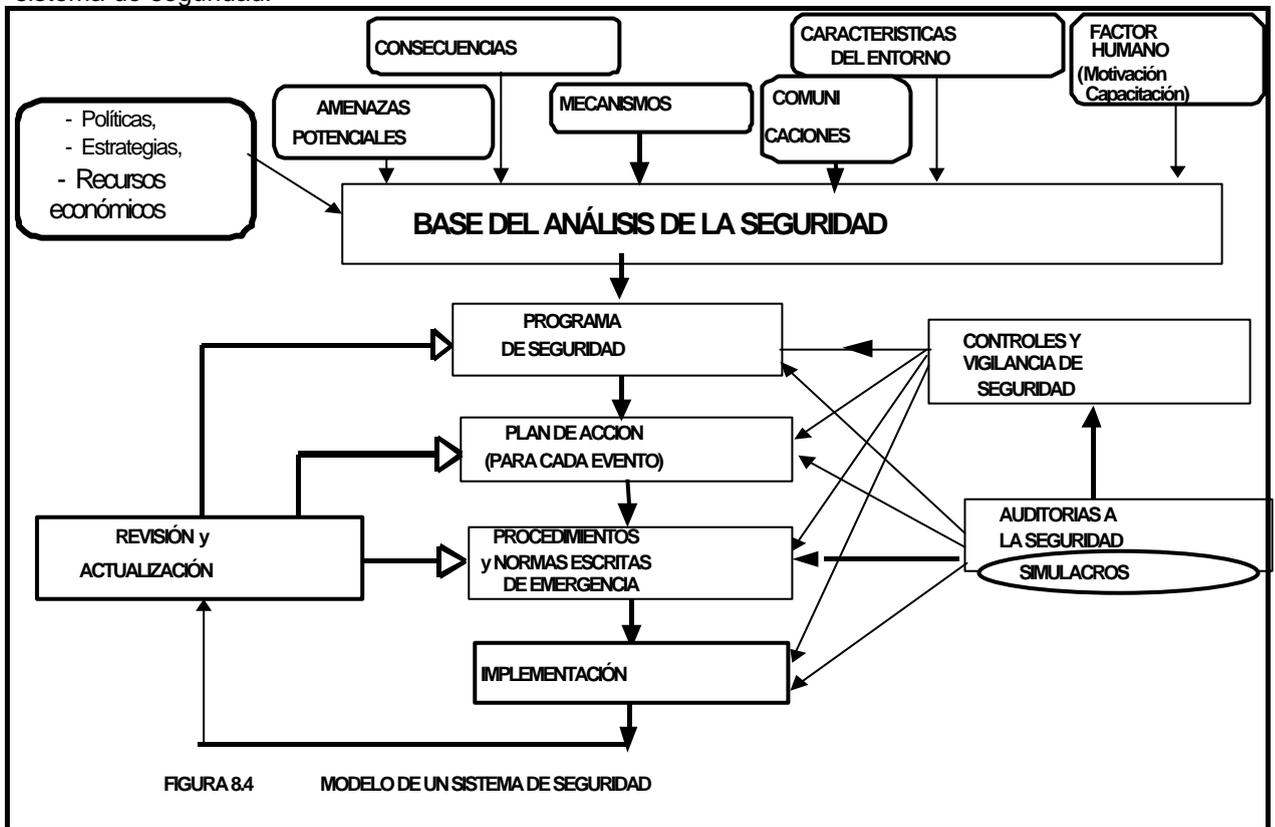
En tercer lugar, **verificar la autorización actual**. El sistema debe verificar el permiso en cada acceso y determinar que el acceso está permitido y después abandonar esta información para su uso posterior aunque esto signifique un gran overhead. Muchos sistemas verifican el permiso solamente al abrir el archivo y no con cada acceso. Esto significa que un usuario que abra un archivo y lo tenga abierto mucho tiempo seguirá teniendo acceso a él, incluso en el caso de que el propietario haya cambiado la protección del archivo.

En cuarto lugar, **dar a cada proceso el mínimo privilegio posible (solo la necesaria para hacer la tarea)**. Si un editor solo tiene la autoridad para tener acceso al archivo por editar, los editores con caballos de Troya no podrán hacer mucho daño. Este principio implica un esquema de protección que se conoce como granulación fina.

En quinto lugar, **el mecanismo de protección debe ser simple, uniforme e integrado hasta las capas más bajas del sistema**. El intento por dotar de seguridad a un sistema inseguro ya existente es casi imposible. La seguridad, al igual que el correcto funcionamiento de un sistema no es una característica que se pueda añadir.

En sexto lugar, **el esquema elegido debe ser sociológicamente aceptable**. Si los usuarios sienten que la protección de sus archivos implica demasiado trabajo, simplemente no los protegerán.

A todas estas ideas se debe agregar, fundamentalmente, el **nivel de capacitación y motivación del ser humano** basado en las características de sus ataques descritos en el punto 8.1 y no debemos olvidar las políticas, estrategias y los recursos económicos que la institución considere adecuados para el sistema de seguridad.



Todo esto forma parte de las **Bases de Análisis para un Sistema de Seguridad** que producirá un **Programa de seguridad** acorde a las necesidades planteadas como justificación. Ver la figura 8.04.

Del Programa de seguridad surgirá un **Plan de acción o Plan de contingencia** que pretenderá cubrir las acciones necesarias para cada evento previsto en los pasos anteriores.

Es necesario documentar a través de **Procedimientos o normas** para encarar cada evento o proceso anormal e incluso emergencias. Estas normas y procedimiento deben ser conocidos por los integrantes de la institución y ser aplicados con la implementación. Durante la implementación surgirán desvíos o modificaciones que deberán ser contemplados en adecuados Procedimientos de revisión y actualización al plan de seguridad. - **TODO DEBERÁ ESTAR DOCUMENTADO!!!** y esta documentación guardada en un lugar seguro.

Cualquier sistema de seguridad requiere de controles y de vigilancias que verificarán el normal funcionamiento del sistema para que no se aparte de los objetivos fijados. Periódicamente se deberá

auditar al sistema con ejercicios que verifiquen la respuesta del sistema a ataques. Estas auditorias son necesarias y se detallan en el punto 8.3.3 de este módulo.

8.3. Tipos de Seguridad

Existen básicamente 2 grandes ámbitos para aplicar los criterios de seguridad:

- **Externa al centro de cómputos:**

- ◇ SEGURIDAD FÍSICA { Desastres
Sabotajes
Intrusos
- ◇ SEGURIDAD LOGÍSTICA { Estado de la instalación
Abastecimiento esencial
Mantenimiento
- ◇ SEGURIDAD OPERATIVA

- **Interna al centro de cómputos:**

- ◇ SEGURIDAD FÍSICA { Acceso de personal al Centro de Computos.
o a áreas controladas: Acceso a cintoteca, etc.
Incendios
- ◇ SEGURIDAD ADM.Y OPERATIVA DEL C.C. { Personal habilitado con división de responsabilidad
Política administrativas
Procedimientos y autorizaciones
Clasificación de los datos
Controles

Desde el punto de vista del Sistema Operativo, la seguridad de la información se basa en:

- seguridad computacional (computer security).
- seguridad de redes computacionales (network security).

8.3.1. Supervisión y Vigilancia

Las tareas en la vigilancia de la seguridad en cuanto a las personas implican determinar los siguientes mecanismos de protección:

- ◇ Control de identidad de usuarios
 - ◇ Patrones de voz vs. sus problemas
 - ◇ Patrones de huellas digitales
 - ◇ Patrones de imágenes
- } características pertenecientes a la persona
- ◇ contraseñas o password
- } característica que conoce una persona
- ◇ tarjetas y llaves
- } característica que posee una persona

Estos puntos se refieren a la autenticación de los usuarios que se trata en el punto 8.8

La supervisión implica que estos mecanismos funcionen adecuadamente cuando se los demanda. Para ello se debe controlar periódicamente su funcionamiento y verificar mediante un procedimiento escrito que el sistema responde bajo una acción de ataque.

8.3.2. Supervisión de Riesgos de Seguridad por el S.O..

El sistema operativo debe controlar las operaciones en lugar del usuario y utilizar **Programas de vigilancia** en caso de una violación o amenaza (concepto de monitor). Para ello pueden emplearse dos tipos de acciones:

- Acción automática
- Acción al supervisor del sistema

En ambos casos se controlan los accesos múltiples con los objetivos de cada usuario y los permisos otorgados. Esto genera una búsqueda mediante los programas de vigilancia que controlan la información de los accesos y los permisos y avisan al supervisor o impiden la acción de acceso.

Estas acciones se refieren a la seguridad de los datos que se trata en el punto 8.4 y a los dominios de protección tratado en 8.5

8.3.3. Auditorias

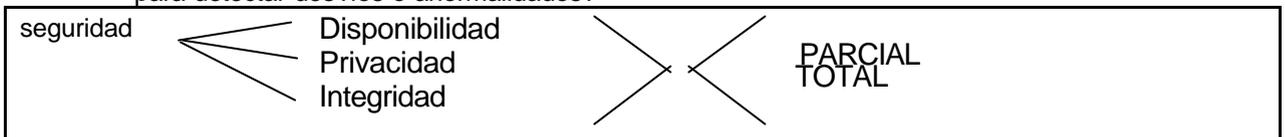
Las auditorias son necesarias para verificar si los sistemas de seguridad funcionan correctamente cuando se los demanda mediante un ataque. Dicho de otra forma, que el sistema cumple con los objetivos bajo el cual fue diseñado y si no lo hace, detectar los puntos débiles que presenta ante estos eventos y luego introducir en el sistema las acciones correctivas adecuadas o modificar las acciones previstas y así restablecer su correcto funcionamiento.

Las Auditorias pueden ser de 2 tipos:

- Sistemas Manuales (generalmente se realizan a posteriori de la ocurrencia de un evento)
- Sistemas Automáticos.

Los Sistemas Interactivos cuentan con el Archivo "Log" o de auditoría que presenta los siguientes problemas:

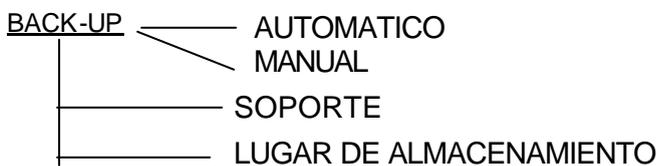
- La protección del archivo log (contra accesos no autorizados. ¿Quién accede y para qué? ¿y el supervisor o Superusuario...?).
- La revisión del contenido del log (¿Quién revisa los contenidos?)
- La frecuencia de revisión (periódico vs. azar)
- La experiencia del auditor para analizar la información que presenta el log (nivel de capacidad para detectar desvíos o anomalías?)



Como ya se explicó, los requerimiento de seguridad en el tratamiento de la información se basa en la disponibilidad, la privacidad y la integridad de la misma. Estas pueden ser totales o parciales.

- ◆ Disponibilidad de los datos o programas mediante adecuados backup o archivo log.
- ◆ Privacidad se ocupa del secreto de los datos que solo deberán estar disponible para los accesos autorizados. Generalmente se utilizan técnicas basadas en Password o criptografía e incluso accesos en que se definen los dominios y permisos o derechos de accesos.
- ◆ La Integridad se ocupa de la modificación de información solo para accesos autorizados.

Por ejemplo la disponibilidad de datos o programas se puede resolver con una adecuada política de copias de resguardos (Backups) realizada ya sea en forma manual o automáticamente. Es importante considerar el tipo de soporte y su capacidad para realizar el copiado puesto que a mayor volumen de datos se requiere mayor capacidad y mayor cantidad de tiempo empleado en la tarea, mayor espacio ocupado para el almacenamiento (cintotecas, etc.) y por supuesto mayores costos en general. En cuanto al lugar de almacenamiento debe ser acondicionado para controlar los accesos y determinar que personas ingresaron y realizaron tal o cual actividad dentro del recinto. Esto requiere adecuados mecanismos de control de accesos y vigilancias. El acondicionamiento también alcanza al ambiente en cuanto a la humedad (relativamente baja, no mayor a 30%) y temperaturas (entre 20 y 23 °C) controladas.



En cuanto al Log no solo sirve para auditar las transacciones realizadas en el sistema, sino permite atacar el problema de integridad de los datos puesto que es el archivo operativo que refleja cada operación o transacción y se puede con el reconstruir la historia con el último backup más el log se recupera la información perdida o dañada. También sirve para verificar la consistencia frente a escrituras (caída del sistema) u operaciones iniciadas y no finalizadas en el log.

8.3.4. Mecanismos y Políticas de Seguridad en Sistemas

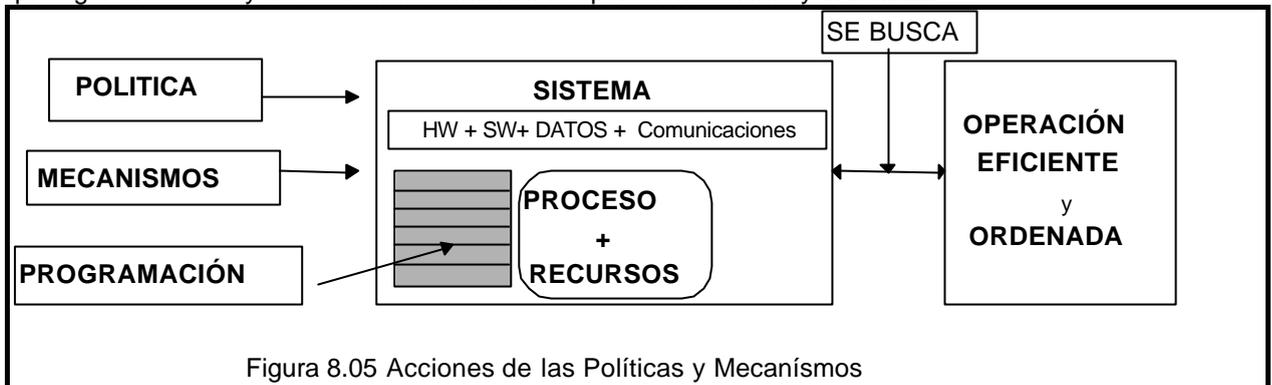
Un sistema puede verse como un conjunto de procesos y recursos (hardware, software, datos y comunicaciones, Ver Figura 8.05) en que se busca una operación eficiente y ordenada.

Para asegurar la operación eficiente y ordenada del sistema, los procesos necesitan estar sujetos a políticas que gobiernan el uso de los recursos, es decir determinan que se hará pero no como se hará y puede variar en el tiempo y de una aplicación a otra.

La protección debe proveer un mecanismo para reforzar las políticas que gobiernan el uso de los recursos. Están más accesibles al programador de aplicaciones y definen el **cómo** se hará el uso de los recursos.

Las políticas definen **qué** se quiere en protección y seguridad.

Los mecanismos de seguridad especifican **como llevar a la práctica** a las políticas de seguridad y como hacerlas cumplir, o sea, que son los que al recibir un ataque generan las acciones necesarias para proteger al sistema y de esta forma continuar la operación eficiente y ordenada.



8.3.5. Funciones de los Sistemas de Protección en el Sistema Operativo

El Sistema Operativo asegura la independencia lógica de los objetos. Ejemplo no establece vinculaciones (bindings) entre ellos o si ya están hechos, no permite las operaciones sobre ellos mediante la negación de acceder a los mismos.

Además el Sistema Operativo debe permitir una protección del uso de la información en función de las operaciones sobre cada objeto. Ejemplo no permitir la lectura cuando el archivo está protegido por privacidad. También debe permitir una protección selectiva de la información compartida en función de los permisos o privilegios de usuarios, grupos, u otros que necesitan usar esa información.

8.4. SEGURIDAD PARA LOS DATOS

Este tema resulta de extremo interés por lo que es necesario tratarlo en los ambientes del Procesamiento centralizado como el distribuido. En particular este último dada su particular óptica, en ese ambiente lo estudiaremos con la visión de los datos almacenados o en movimiento a través de redes, buses o cables. Para ello trataremos de dar algunas técnicas utilizadas para transformar el dato para ocultarlo y que no sea fácilmente legible o entendible para el intruso a simple vista.

8.4.1. Seguridad de Datos en General

En este apartado estudiaremos los ataques que pueden sufrir los datos y como implementar los sistemas de seguridad para mitigar los posibles daños causados por las amenazas.

Los datos son manipulados por programas (software) y en el punto 8.1.8 de este módulo ya se presentaron algunos tipos de amenazas al software y a sistemas. En particular se explicitaron 10 casos de ataques genéricos que también son aplicables a los datos.

Los casos más dañinos son los que se producen por descuido o malicia del personal de la misma instalación (acciones provocados desde los "torpes" hasta los intrusos calificados).

- Ejemplificaremos algunos intentos de penetración mediante los métodos utilizados en los ataques a los datos:
 - **Bomba de tiempo:** cuando el programador es despedido, deja códigos que se destruyen o destruyen información a partir de una cierta fecha.
 - **Redondeo de centavos:** embolsar en una cuenta particular los centavos sobrante por redondeo.
 - **Terminal con sesión abierta:** La terminal queda desatendida por el usuario durante un instante de tiempo. Lo suficiente para que un intruso acceda al sistema con el acceso habilitado a todos los datos y recursos disponibles para el usuario legítimo cuya identidad asume.

- **Contraseñas:** Las contraseñas utilizadas para autorizar a los usuarios pueden ser obtenidas por intrusos con propósitos de acceso ilegal de varios modos, incluyendo la adivinación, el robo, la prueba y error o el conocimiento de contraseñas suministradas por vendedores para generación y mantenimiento de sistemas.
- **Inspección:** Con frecuencia, los usuarios pueden ser capaces de descubrir información a la que no tiene autorización para acceder simplemente inspeccionando los archivos del sistema. En muchos sistemas existen archivos que disponen de controles de acceso inadecuados o demasiado permisivos que son aprovechados por personal no autorizado que “investiga” y obtiene datos confidenciales o protegidos.
- **Puertas traseras o trap doors:** Se trata de puntos secretos de entrada sin autorización de acceso. Los diseñadores de software las preparan a veces, presumiblemente para permitirles a ellos acceder y posiblemente modificar sus programas después de la instalación y puesta en uso. Las puertas cebo pueden ser objeto de abusos por alguien que conozca su existencia y el procedimiento de entrada.
- **Escucha electrónica:** Puede conseguirse mediante conexiones de interceptación pasiva o activa o mediante captura electromagnética de la radiación de cables o circuitos.
- **Mutua confianza:** Una programación demasiado confiada o poco cuidadosa puede llevar a dejar de comprobar la validez de los parámetros transferidos. En consecuencia un invocador puede obtener acceso no autorizado a información protegida. Otros descuidos incluyen el paso de parámetros por referencia en vez de por valor. En este caso, un programa de usuario puede invocar al sistema operativo con punteros a parámetros que residen en el espacio del usuario. Tras pasar la verificación efectuada por el sistema operativo, el usuario puede sustituir rápidamente los valores originales por otros no autorizados. La ejecución consiguiente de la rutina del sistema puede entonces ser aprovechada para obtener acceso no autorizado a información que debería estar protegida.
- **Caballo de Troya:** Un programa puede ocultar intencionalmente parte de su funcionalidad, con frecuencia dañina, con el fin de pasar datos o los derechos de acceso del usuario a alguien más. Es posible escribir un sencillo programa Caballo de Troya para robar contraseñas de usuario imitando el programa legítimo de apertura de sesión (*login*) y reproduciendo fielmente la secuencia y diálogos de presentación formal. Un programa Caballo de Troya es fácil de implantar en sistemas en donde las terminales se hallan en recintos públicos, dejando una copia activa en una terminal y haciendo que simule la pantalla de presentación/despida. Versiones más sofisticadas de programas Caballo de Troya pueden ser muy difíciles de detectar si emulan totalmente al programa utilitario al que están suplantando, con la provisión adicional de enviar los datos de interés a un intruso.
- **Gusanos informáticos:** Estos programas pueden invadir los computadores, generalmente a través de una red, y denegar servicios a los usuarios legítimos utilizando cantidades desproporcionadas de recursos de procesamiento y comunicación para su autopropagación.
- **Virus informáticos:** Los virus son trozos de códigos que infectan a otros programas y con frecuencia realizan actividades dañinas, tales como eliminar archivos o corromper el bloque de arranque de un disco booteable.
- **Prueba y error:** La potencia de procesamiento de un computador puede ser utilizada por un intruso que pretenda entrar al sistema para automatizar repetidamente la apertura de sesión y adivinación de la contraseña mediante prueba y error. En muchos sistemas (UNIX®, por ejemplo), el archivo de contraseñas está almacenado en un directorio públicamente accesible. Las contraseñas están cifradas, pero los nombres de los usuarios y los ID no. Un intruso puede copiar este archivo en un computador aparte y preparar un largo ataque fuera de línea aplicando criptoanálisis y adivinación. Si se obtienen algunas contraseñas, pueden ser utilizadas para irrumpir en el sistema informático y actuar.
- **Búsqueda de basura:** Puede ser utilizada para descubrir contraseñas o escudriñar los archivos, volúmenes y cintas suprimidas. En muchos sistemas, el borrado de los archivos se efectúa actualizando las entradas de los directorios y devolviendo los bloques de datos al espacio de bloques libres. Es posible entonces reconstruir información útil revisando los bloques libres. Los programas utilitarios de recuperación de archivos borrados, proporcionados por el fabricante del sistema operativo, para recuperar archivos eliminados accidentalmente, pueden ser utilizadas para simplificar el proceso. Estos utilitarios sirven para efectuar búsquedas en los directorios, o cestos de basura, de los listados con palabras claves.
- **Descuido:** aprovechar que la terminal quedó conectado mientras el operador se fue a atender el teléfono o hacer otros menesteres, o utilizar palabras claves sencillas de adivinar.

En fin, estos son algunos de la larga lista de acciones que atacan a la Seguridad de los datos en general. Además de estos ataques estudiaremos en particular el aspecto de la integridad de los datos.

El Sistema de Seguridad para los datos

El sistema de Seguridad prevé accesos no autorizados a los sistemas y la consecuente destrucción maliciosa o alteración de los datos. La seguridad determina una correcta autenticación de los usuarios del sistema para proteger la integridad de la información almacenada en el mismo.

Proteger dicha integridad trae aparejada la toma de medidas abarcando dos niveles:

- **FÍSICO:** Controlar el/los lugares donde se hallen dispuestos sistemas informáticos de personas ajenas al mismo, que puedan ser causales de daños maliciosos.
- **HUMANO:** Identificar a la persona que desea acceder al sistema y a sus datos y verificar la identidad de la misma. El control de acceso es una de las líneas de defensa más importantes contra los intrusos indeseados.

• **INTEGRIDAD DE LOS DATOS**

La integridad de los datos busca hacer de estos una “entidad indivisible”, que permita asegurar que están exentos de ataques externos.

Obviamente si se plantea este concepto es porque existe la posibilidad de que se vean amenazados. La siguiente figura muestra las cinco posibles amenazas a la Integridad de los datos más comunes:

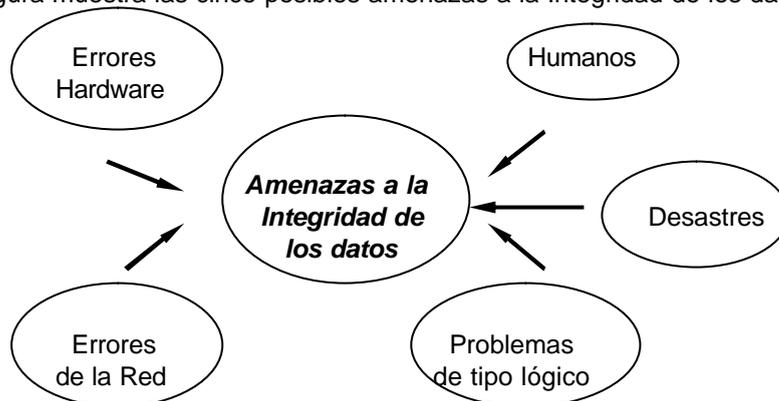


Figura 8.06 Amenazas a la integridad de los datos

Humanos: En este tipo de amenaza entran en juego tanto los errores de los usuarios finales como de administradores de red que por desconocimiento, descuido, estrés, accidentalmente o a propósito borran datos, que a veces puede llevar horas tratar de recuperarlos.

Errores de Hardware: Las personas no son las únicas que fallan, los equipos tienen su vida útil y por ende pueden fallar: problemas de alimentación, de disco, de memoria, mal comunicación con los controladores de E/S, etc.

Errores de la Red: Las señales eléctricas son generadas en una computadora y difundidas en algún tipo de red. Las líneas que conectan las máquinas están expuestas a una variedad de riesgos, incluyendo interferencias y averías físicas, cualquier problema que se pudiese ocasionar en el Cableado, (en algún componente de la red, en los controladores o tarjetas de interfase) darían como resultado la pérdida o la corrupción de los datos.

Problemas de tipo lógico: Otro elemento factible de ocasionar problemas en el la integridad de los datos es el software. Los mismos son producidos por errores de almacenamiento, corrupción de archivos, de S.O. o requisitos mal definidos por el desarrollador de la aplicación.

Desastres: Existen también factores de la naturaleza que no se pueden controlar y que también influyen en la preservación de los datos como ser: incendios, inundaciones, tormentas, accidentes industriales, terrorismo, etc.

• **Seguridad de los Datos**

Veremos ahora las distintas formas en que los datos son puestos en peligro, entendiéndose por Amenazas a la Seguridad a los estados o las actividades que podrían ser explotadas o utilizadas para obtener **accesos no autorizados** a los datos.

Físicas: Se base en mantener de la forma más privada posible los accesos a los datos, no dejando que personas no autorizadas puedan hacer uso de los mismos. Son ejemplo de ello: robo de datos, espionaje, dumpster diving (revolver en la *basura* para tratar de encontrar diskettes o algún tipo de material impreso).

Basadas en los cables: El uso de las redes de computadoras creará amenazas adicionales como ser escuchas del tráfico, marcación del número telefónico, imitación (capacidad de una máquina de parecerse a otra en una red - “gemelo maligno”).

Autenticación: Implica la forma en que los usuarios tienen de identificarse antes de poder acceder a los recursos que les brinde el sistema. Las amenazas relacionadas con este tema son: captura de contraseñas, averiguación de contraseñas y edición de contraseñas.

Programación: Las violaciones contra la seguridad verdaderamente destructivas proceden del código (software o programas), estos ataques la mayoría de las veces destruyen datos intencionalmente. Ejemplo de ello pueden ser: virus - código bomba, Caballos de Troya, Gusanos (Worms).

Trap Doors: Se denomina de esta forma a cualquier tipo de puerta de escape en todo tipo de sistema que posibilite en ingreso indebido a personas no autorizadas por ejemplo: Piggybacking¹ (común en un entorno distribuido), configuración e iniciación.

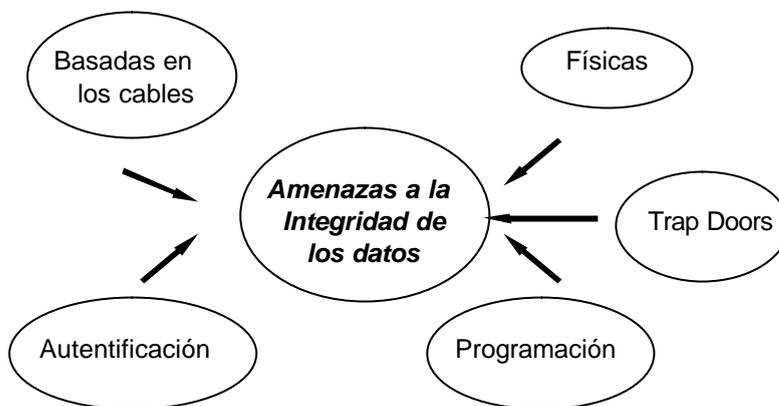


Figura 8.07 Amenazas a la integridad de los datos

• **HERRAMIENTAS PARA MEJORAR LA INTEGRIDAD DE LOS DATOS**

A continuación presentamos una tabla con alguna de las posibles técnicas para recuperar y/o prevenir una falta de integridad de los mismos. Se entiende por “acción preventiva” a la que se debe tomar para mantener la Integridad de los Datos y por “acción correctiva” a la que se debe tomar para recuperar la misma.

Herramientas	Técnica
Copias de Seguridad	Correctiva
Técnicas en espejo	Preventiva
Archivado	Preventiva
Custodia de Backups	Correctiva
HSM (<i>Hierarchical Sotorage Management</i>)	Preventiva
Chequeo de paridad	Preventiva
Planificación de recuperación frente a desastres	Correctiva
Análisis predictivo de fallos	Preventiva
Alimentación ininterrumpida de energía	Preventiva

Tabla 8.03 Técnicas para recuperar o prevenir una falta de integridad de los Datos

Copias de Seguridad: Es indispensable mantener copias de backup actualizadas en toda organización, de modo que si sucede alguna alteración en la información se la pueda recuperar a través de las copias realizadas. Por lo general, el período de conservación de las mismas es corto, puede durar un día, una semana, un mes o quizá algunos años.

Técnicas de Espejo: Esta técnica permite aumentar la disponibilidad de los servidores, implantando componentes redundantes o tolerantes a fallas que puedan entrar en funcionamiento y asumir las responsabilidades de los componentes en fallo.

Archivado: Permite borrar los datos de los sistemas interactivos y llevarlos a sistemas de almacenamiento offline. Los datos archivados pueden ser almacenados por un tiempo indefinido.

Custodia de Backups: Es el resguardo de copias de seguridad en lugares externos a la organización por incidentes que puedan ocurrir en la misma.

HSM (*Hierarchical Sotorage Management*): Es un sistema de gestión jerárquica de almacenamiento y con una estructura similar al proceso de archivado.

¹ Piggybacking = ingreso en cascada

En el proceso de copia de los datos, se deja en el sistema un archivo de resguardo, de modo que si el usuario decide acceder al archivo de resguardo, interviene el sistema HSM y recupera el archivo original del medio HSM apropiado. El concepto de jerarquía se da en el hecho de que se copiarán los archivos en distintos dispositivos (on-line, de uso frecuente u offline) en función de su vejez.

Chequeo de Paridad: Es una característica de los servidores de Head Office. Determina un mecanismo que asegura que ante fallos de memoria inesperados no tengan como resultado el fallo del servidor o la pérdida de integridad de los datos.

Planificación de recuperaciones frente a desastres (Disaster Recovery Planning): Nunca se esta exento de la posibilidad de ocurrencia de alguna contingencia externa que imposibilite el normal desarrollo de las actividades. Es por ello que se debe prever la existencia de algún organismo externo que permita llevar a cabo las actividades mínimas de la empresa hasta tanto ésta se vuelva a rearmar.

Análisis predictivo de fallos: En general algunos dispositivos actuales poseen un mecanismo que posibilita detectar su falla antes de que realmente dejen de funcionar, informando de su estado a través de algún tipo de señal.

Alimentación ininterrumpida de energía: Consiste en brindar por medio de distintos dispositivos el suministro de energía en forma ininterrumpida de forma tal de que los elementos esenciales estén siempre disponibles. Algunos dispositivos también permiten ser usados como estabilizadores de tensión.

• HERRAMIENTAS PARA CONSEGUIR UN NIVEL DE SEGURIDAD ADECUADO

El siguiente cuadro presenta algunas de las herramientas utilizadas para reducir las amenazas contra la Seguridad. En él se indica si la herramienta utilizada puede ser implementada por el sistema o si es una política de la organización que se debe comunicar al personal involucrado.

Recomendaciones	Implementación
Eliminación de las "Trap Doors"	Sistema
Chequeo de Virus	Sistema
Seguridad Física	Política
Determinación de Políticas	Política
Cifrado	Sistema
Obligación de identificación	Sistema
Firewalls	Sistema
Trampas para intrusos	Sistema

Tabla 8.04 Herramientas para reducir las amenazas contra la seguridad de los datos

Eliminación de "Trap Doors": Consiste en eliminar las puertas traseras del sistema para evitar que intrusos al mismo interfieran en la seguridad de los datos.

Chequeo de Virus: El chequeo de virus mediante productos corrientes del mercado permitan prevenir algún daño en el sistema. En la actualidad dichos productos se suelen mantener residentes en memoria principal.

Seguridad Física: Proteger a los equipos en áreas físicas con ingreso restringido.

Determinación de Políticas: Es la definición de políticas internas de cada organización que permitan resguardar la información personal, eliminar aquellos datos que son de desuso para la empresa pero que pueden resultar de interés para otros, fijar medidas para que todos los usuarios accedan a los recursos del sistema o a la red mediante contraseñas, determinar la frecuencia de cambio de contraseñas, etc.

Cifrado: Garantiza que la información no es inteligible para individuos, entidades o procesos no autorizados. Consiste en transformar un texto claro mediante un proceso de cifrado en un texto codificado, mediante la utilización de claves de cifrado.

Obligación de identificación: Requiere una identificación correcta del origen del mensaje asegurando que la entidad no es falsa.

El término Identificación hace referencia al proceso de verificar la identidad de alguien. Generalmente existen tres formas de probar la propia identidad: mediante algo que se tiene, que se sabe o se es. Más adelante en este módulo explicaremos en detalle estas técnicas.

Que se tiene	llaves del auto, tarjetas de identificación
Que se sabe	contraseña (password)
Que se es	Huellas digitales, DNI, patrones de voz

Firewalls: Utilización de dispositivos para la protección de la LAN² como consecuencia de posibles ataques producidos a través de una conexión a Internet con que cuente la organización.

² LAN = Local Area Network (Red de Área Local)

Trampas para intrusos: La idea es hacer creer a los intrusos que se encuentran dentro del sistema mientras que, en forma simultánea se intenta localizarlos.

8.4.2. Seguridad de Datos en Bases de Datos

Generalmente se plantea la pregunta: ¿Por dónde empezar?

- Utilizando los mecanismos de seguridad propios de los S.O. de red modernos (UNIX, Novell, Windows NT) los cuales ofrecen algún tipo de seguridad en forma de administración de usuarios, comprobación de contraseñas, seguimiento.
- Estableciendo procedimientos que especifiquen el tipo de contraseñas a utilizar a través de un mecanismo de gestión de claves que tenga en cuenta los siguientes ítems:
- **Espacio de claves reducido** → Esta relacionado a las restricciones impuestas en el número de bits o en la clase de bytes (caracteres ASCII, alfanuméricos o imprimibles) permitidos en una clave.
- **Elección pobre de la clave** → Esta referido a la elección de claves evidentes por parte del usuario (por ejemplo su propio nombre, sobrenombre o el de algún familiar)
- **Claves aleatorias** → Claves buenas son las cadenas de bits aleatorios generadas por medio de algún proceso automático.
- **Frases de paso** → Esta solución al problema de la generación de contraseñas seguras y fáciles de recordar por parte del usuario consiste en utilizar una frase suficientemente larga que posteriormente es convertida en una clave aleatoria por medio de un algoritmo (key-crunching).
- Si se supone un acceso público a ciertas Bases de Datos, habría que mantener ciertos controles sobre su actualización.
- Por ejemplo:
 - **Dependiente de los datos:** solo es posible borrar registros que no hayan sido actualizados en por lo menos más de tres meses.
 - **Dependiente del tiempo:** el campo de sueldo solo es posible de ser cambiado en horarios determinados.
 - **Dependiente del contexto:** solo se pueden listar nombres o sueldos, pero no ambos a la vez.
- Generalmente las bases de datos comerciales chequean con una Lista de Control de Acceso (ver punto 8.5) asociada a una relación o conjunto de campos o registros.
- Es necesario un control de consistencia de dato constante y antes de permitir un acceso o modificación.
- En los casos que se permite el acceso a Bases de Datos con fines estadísticos no debería permitirse el acceso a datos individuales.
- Habría que bloquear el acceso a la información cuando los datos recuperados son pocos.

Esto son solamente algunos puntos a tener en cuenta desde el aspecto de la seguridad en las bases de datos.

SEGURIDAD DE LOS SERVIDORES DE BASES DE DATOS

Dado que los servidores son el medio de almacenamiento más común de las Bases de Datos es necesario contar con algún mecanismo que les brinde protección. Esta puede ser tanto física para evitar su robo como lógica en lo que hace al robo de información no autorizada. También será necesario implementar un óptimo control de virus, para poder controlar toda información que tenga acceso al servidor en cuestión.

8.4.3. Seguridad en Telecomunicaciones o Redes de Computadoras

Desde el punto de vista de la seguridad informática, una red debe entenderse como un entorno de cómputo con más de un computador independiente. No obstante, la introducción de las redes informáticas no modifica tan solo cuantitativamente el problema de la seguridad, sino que supone un incremento cualitativo del mismo. No se trata tan solo de que debamos proteger un mayor número de computadores de un mayor número de atacantes potenciales, sino que se introducen toda una nueva serie de vulnerabilidades y amenazas, y se hacen necesarias toda una nueva serie de técnicas y herramientas para protegernos de ellas.

En este tema no se pretende recorrer exhaustivamente todos los aspectos de la seguridad en redes informáticas, sino solo exponer algunos de los principales problemas de seguridad que plantean e introducir algunos de los principales mecanismos utilizados para evitarlos, tales como las listas de acceso.

Comenzaremos por citar algunas de las principales ventajas y desventajas introducidas con el uso de redes informáticas.

Ventajas:

En primer lugar y fundamentalmente, la introducción de las redes informáticas supone **compartir** una enorme cantidad de recursos, tanto hardware, como software y de información. Con la conexión a Internet, el usuario puede acceder a nuevas máquinas situadas en cualquier parte del mundo para desarrollar y ejecutar sus programas, puede acceder a toda una serie de aplicaciones que no podría utilizar en su PC, y sobretodo tiene acceso a una gran cantidad de información abarcando todo tipo de temas imaginables y en continua actualización.

A la hora de acceder a los recursos, el uso de una red informática incrementa la **fiabilidad** de los mismos, puesto que su replicación permite seguir disponiendo de recursos alternativos en caso de fallo.

En aplicaciones que exijan una gran cantidad de recursos es posible **distribuir el trabajo** entre varias máquinas distribuidas en red.

Desventajas:

Como contrapartida a todas las ventajas anteriores, la conexión a la red y la expansión de Internet, ha incrementado muy sustancialmente los problemas de seguridad con los que debemos enfrentarnos. Desde un punto de vista general algunos de los principales inconvenientes introducidos son los siguientes:

El hecho de **compartir los recursos** a través de la red incrementa el número de usuarios involucrados y por tanto el número de atacantes potenciales.

La complejidad del sistema: La combinación de distintos tipos de nodos con distintos sistemas operativos a través de redes que pueden ser heterogéneas, incrementa la complejidad del sistema. Los controles de seguridad se hacen entonces más difíciles de implementar.

Perímetro desconocido. La continua expansión de la red convierte en incierta la identidad de los integrantes de la misma. No conocemos qué nodos pueden conectarse ni qué nuevos problemas suponen.

Múltiples puntos de ataque: La información a proteger ya no se encuentra restringida a la memoria o los dispositivos de almacenamiento de un nodo. La introducción de la red y la transmisión de la información hace que sea necesario establecer mecanismos de protección, tanto en los nodos origen y destino, como en los dispositivos de encaminamiento y transmisión y en todos los puntos intermedios por los que circula la información.

Privacidad de la información: se hace más difícil mantenerla en cada nodo debido al aumento de posibles usuarios que pueden penetrar en el mismo. Por otro lado al transferirse la información, en muchos casos sin ningún tipo de protección, aumenta el número de puntos en los que puede ser interceptada y desvelada.

Integridad de la información: La transmisión de la información es un claro peligro para su mantenimiento. Los mensajes pueden ser interceptados, modificados, borrados, e incluso pueden insertarse mensajes falsos.

Autenticidad de Usuarios: al utilizar redes informáticas no tan solo es necesario autenticar a los usuarios, sino también a los nodos. Para ello estudiaremos los métodos de accesos más adelante.

Disponibilidad de Recursos: son muchos los ataques que pueden lanzarse contra una máquina con el fin de saturar sus recursos o de aislarla de la red.

Entonces se deben resolver básicamente dos Problemas:

- **PRIVACIDAD:** Es necesario mantener los datos en secreto de un punto al otro de la comunicación.
- **AUTENTICIDAD:** Es necesario conocer si los datos son auténticos o si fueron modificados antes de llegar.

Por ejemplo:



En la figura 8.08 "X" podría ser un espía industrial o financiero o cualquier Hacker. También podría hacer cosas como generar o repetir mensajes de depósitos en cuentas corrientes (esto se podría evitar teniendo en cuenta el número de mensaje, la hora, la identificación de la terminal, etc.). Una solución para este tipo de problemas es la **CRIPTOGRAFIA** que explicaremos en el próximo punto.

En general se usan "barreras de Fuego" (firewall) para proteger el acceso de intrusos a un área local de una red. Estas barreras se realizan mediante un sofisticado software.

Firewalls - Conceptos Básicos

En sus primeros años Internet era un ambiente en el cual se privilegiaba el intercambio de información y la facilidad de conexión. Es por ello que los protocolos que allí se usaban privilegiaban varios aspectos, pero no la seguridad.

Con el correr de los años Internet se vuelve más comercial y de fácil acceso, por lo que el nivel de seguridad ofrecido por estos protocolos se volvió insuficiente. La comunidad de Internet respondió a estas amenazas con nuevos protocolos más seguros como el IPv6, la aparición de organismos de control de incidentes y asesoría de seguridad como el CERT (Computer Emergency Response Team) y de nuevos dispositivos de mejora de seguridad. Entre estos últimos se encuentran los firewalls cuya misión es limitar el acceso a una red desde Internet.

El Concepto de Firewall

En la construcción de edificios, un muro o pared contra incendios (firewall) está diseñado para evitar que se propague el fuego de una parte a la otra. Este mismo concepto de propagación se utiliza en las redes.

Análogamente, un firewall para una red evita que los peligros externos de una red o internet se extiendan a la red interna.

En la práctica abarca los siguientes propósitos:

- Restringe el acceso a un punto cuidadosamente controlado.
- Evita que los atacantes se acerquen más a las defensas.
- Restringe a los usuarios para que salgan en un punto cuidadosamente controlado.

El punto donde se instala es donde la red interna protegida se conecta con Internet, y controla todos los pedidos de servicios que entran desde el exterior y salen desde los usuarios, tales como, correo electrónico, transferencia de archivos, inicio de sesiones remotas, etc.

Por lógica, un firewall es un separador, un limitador, un analizador. Su implementación física varía de una instalación a otra. Con mayor frecuencia consta de un conjunto de componentes de hardware (un enrutador, una computadora anfitrión, o cierta combinación de enrutadores, computadoras y redes con software apropiado).

Existen tres tipos de ataques en Internet:

La penetración: donde el atacante trata de acceder a alguna estación en la red bajo ataque y obtener de ella información no autorizada;

La utilización de recursos ajenos: un caso típico de este tipo de ataque es el uso de servidores de mail ajenos para enviar correo basura. A este ataque se lo conoce como Spam.

La negación de servicio: este ataque no consiste en penetrar una red o un servidor sino en dejarlos inoperativos. Es uno de los ataques más comunes, ya que puede ser la respuesta exasperada de un atacante ante la falla de algún ataque previo de los tipos anteriores.

LA SEGURIDAD Y LOS FIREWALLS

El firewall puede protegernos muy bien de los ataques desde Internet, pero es un hecho muy conocido que más del 90% de los ataques a redes provienen desde dentro de la misma red, ya sea por empleados insatisfechos, o por falta de seguridad interna en la misma. En otras palabras el firewall soluciona muchos inconvenientes, pero su sola instalación no es excusa para no desarrollar un esquema interno de seguridad informática en la red.

Un firewall además de protegernos de ataques externos, nos permite implementar una política de acceso interno a Internet, es decir:

- ¿Quiénes en la red protegida?,
- ¿De qué manera? y
- ¿Cuándo podrán hacer uso de Internet?.

Algunos firewalls poseen listas de sitios que resultan inapropiados visitar.

Limitaciones del firewall

La limitación más grande que tiene un firewall sencillamente es el hueco que no se tapa y que coincidentalmente o no, es descubierto por un hacker. Los firewalls no son sistemas inteligentes, ellos actúan de acuerdo a parámetros introducidos por su diseñador, por ende si un paquete de información no se encuentra dentro de estos parámetros como una amenaza de peligro lo dejara pasar. Pero este no es lo

más peligroso, lo verdaderamente peligroso es que ese hacker deje "back doors" es decir abra un hueco diferente y borre las pruebas o indicios del ataque original.

Otra limitación es que el firewall "no es contra humanos", es decir que si un hacker logra entrar a la organización y descubrir passwords o se entera de los huecos del firewall y difunde la información, el firewall no se dará cuenta.

Es claro que el firewall tampoco provee de herramientas contra la filtración de software o archivos infectados con virus.

Tipos de Firewalls

Como dijimos anteriormente, un firewall es un dispositivo que restringe el acceso desde y hacia Internet. Básicamente separa la red en dos porciones, la llamada red "insegura" también conocida como "red roja" sin ningún tipo de protección, y la "red protegida" por el firewall y conocida como "red azul".

A esta división se le suele agregar una zona intermedia donde se ubican los servidores de acceso público: DNS³, Mail Exchanger, Web (www⁴), etc. A esta zona intermedia, protegida por el firewall pero con acceso permitido desde el exterior, es conocida como zona desmilitarizada (DMZ).

El sentido de poner una zona desmilitarizada es el siguiente: para poder acceder a un servidor público es necesario permitir el acceso externo a través del firewall pues si dicho servidor estuviera ubicado en la zona segura, el posible atacante tendría acceso a toda la red protegida, dado que en esta instancia ya hubiera pasado a través del firewall. En cambio al estar en una DMZ, el intruso sólo podrá acceder a los servidores ubicados en dicha zona, pero no a la red segura.

Existen dos tipos básicos de firewall:

Firewall a nivel de red o filtros de paquete: Restringen las conexiones basadas en dirección origen y destino y ports de servicio TCP/IP⁵ de los paquetes que circulan a través de ellos. Esto se concreta en las listas de acceso en las cuales se permite o se niega acceso a determinadas direcciones y ports de servicio TCP/IP. Ejemplo de ello es un router efectuando filtrado de conexiones.

Firewall a nivel de aplicación: En este caso los firewalls no permiten el paso de tráfico entre las redes sino que efectúan conexiones a nivel de aplicación. A este tipo de firewall se lo conoce como Proxy Server. Ejemplo de este tipo es una conexión de Internet al servidor WWW que se conecta en primer lugar con el firewall y luego éste a su vez se conecta con el server de WWW efectuando el paso de datos entre ambos. Otra ventaja de estos firewalls es que permiten la traducción de direcciones, es decir, ocultan la numeración IP de la red interna utilizando solamente los números de la red pública.

Un aspecto muy importante de todo firewall es su capacidad y facilidad de actualización frente a nuevos métodos de ataque sin tener que esperar a un nuevo release de software. En este sentido, el firewall debe comportarse como un programa antivirus.

REDES LOCALES:

- Estas redes tienen problemas de seguridad, pues generalmente están pensadas como "broadcast" (todos escuchan). Inclusive existe la posibilidad de daño físico, ya que si alguien conecta el cable de transmisión al toma corriente eléctrica, es posible dañar la interfase de nodos y hacerla inoperante.

Políticas internas:

Estas políticas aunque no lo parezca, influyen mucho a la hora de prevenir ataques, porque según estadísticas, la mayoría de los ataques sufridos por empresas son de "internos". Con una buena política interna de la empresa, se podrían prevenir estos ataques con un a mayor efectividad.

8.4.4 Métodos de ocultamiento de los Datos

Los datos o mensajes se transforman para su almacenamiento o transmisión. Luego para volver a usarlos se descifran⁶ a su forma original.

- La base de la *criptografía* es la que se indica en la siguiente figura 8.9.

³ DNS = Domain Name Server (Servidor de Nombres de Dominio)

⁴ www = World Wide Web

⁵ TCP/IP Transmission Control Protocol / Internet Protocol.

⁶ Usaremos el término encriptar y cifrar como sinónimos

Se aplica según el siguiente ejemplo en que las entradas son el texto y la clave, ambas en binario y se obtiene el criptograma, luego con el criptograma y la clave se reconstruye el texto:

TEXTO	CLAVE	CRIFTOGRAMA	CLAVE	TEXTO
1	1	0	1	1
1	0	1	0	1
0	1	1	1	0
1	0	1	0	1
0	1	1	1	0

Tabla 8.08 Método DES

- En 1976 se estableció el DES (Data Encryption Standard) como norma para mensajes no relacionados con la seguridad nacional de los EE.UU., por la NBS (National Bureau of Standards), y fue desarrollado por IBM. Consiste en dividir los datos en bloques de 64 bits, utilizando 56 para datos, y los otros 8 como paridad. Luego existen $2^{56} (\cong 7.2 * 10^{16})$ claves posibles.

Ejercicio: Descifre el siguiente texto, sabiendo que la clave es SEGURO, que realiza el orden de acuerdo a la posición alfabético de sus letras: (E=1, G=2, etc.). El cifrado es por transposición y el texto cifrado es:

NSEEEIPCPOCNUNRICEAODOEXOETRDOTNUEJDDMECRSUTNIOO

Solución: El texto tiene 48 letras, si la cantidad de columnas (total de letras de SEGURO), son 6, entonces cada columna posee 8 letras.

De acuerdo a ello, tomamos grupos de 8 letras del texto cifrado y luego ordenamos alfabéticamente SEGURO (queda la palabra EGORSU) y asignamos a cada grupo de letras su letra correspondiente de la clave:

NSEEEIPC ---> E
 POCNUNRI ---> G
 CEAOODEX ---> O
 OETRDOTN ---> R
 UEJDDMEC ---> S
 RSUTNIOO ---> U

Ahora se ordena SEGURO con sus letras encolumnadas y nos queda:

```

S E G U R O
=====
U N P R O C
E S O S E E
J E C U T A
D E N T R O
D E U N D O
M I N I O D
E P R O T E
C C I O N X
    
```

El texto llano es: UN PROCESO SE EJECUTA DENTRO DE UN DOMINIO DE PROTECCIÓN

ONE-TIME PAD (BLOQUE DE USO ÚNICO):

- Es un método muy seguro. Trabajo como el método de Cesar, pero el entero a sumar varía para cada carácter del texto en forma aleatoria.
- Es seguro, porque el criptograma "SECRETO" podría provenir de una palabra como ser:

MENTIRA con la clave 6 0 15 2 22 23 12 o de:
 REALEZA con la clave 1 0 2 6 0 6 12.

El problema es que la clave es tan larga como el mensaje, y nunca debe ser reutilizada.

8.4.5 Algunos problemas en CRIPTOGRAFÍA.

Las Claves:

- Métodos simétricos:** Son aquellos en los que la clave de cifrado coincide con la de descifrado. Lógicamente dicha clave tiene que permanecer secreta, lo que presupone que emisor y receptor se han puesto de acuerdo previamente en la determinación de la misma, o bien que existe un centro de distribución de claves que se la ha hecho llegar a ambos por un canal seguro.

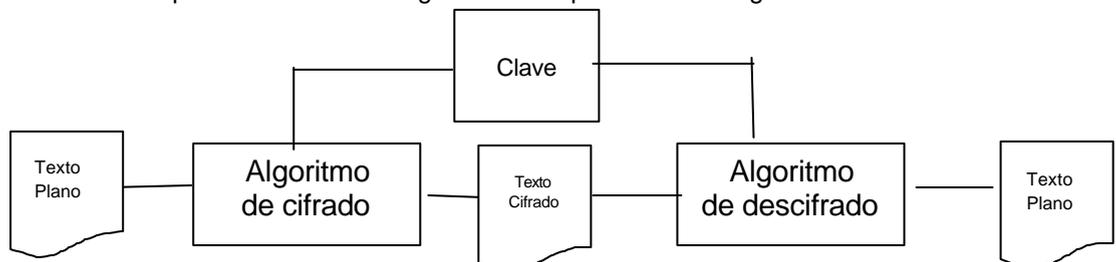


Figura 8.10 Método de claves simétricas

- **Métodos asimétricos:** Son aquellos en los que la clave de cifrado es diferente a la de descifrado. En general, la clave de cifrado es conocida libremente por el público mientras que la de descifrado es conocida únicamente por el usuario.

Los métodos simétricos son propios de la criptografía clásica o *criptografía de clave secreta*, mientras que los métodos asimétricos corresponden a la *criptografía de clave pública*.

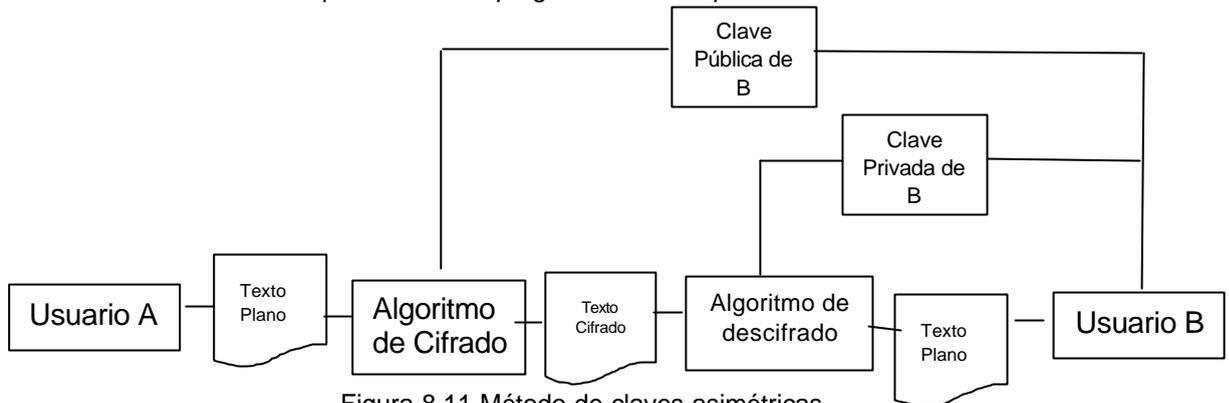


Figura 8.11 Método de claves asimétricas

CLAVES PÚBLICAS:

- Los métodos tradicionales se denominan simétricos, pues usan la misma clave para encriptar y desencriptar. Los denominados asimétricos tienen dos claves: una para encriptar y otra distinta para desencriptar.
- Si se supone dos interlocutores A y B. A usa una clave E para encriptar, y B usa una clave D para desencriptar en un tiempo razonable.
- Esto permite que A y B publiquen sus claves de encriptamiento E_a y E_b , y mantengan en secreto sus claves de desencriptamiento D_a y D_b . Con esto, el problema de distribución de claves desaparece.
- El problema radica en generar claves E y D.

Uno de los métodos más importantes es el RSA (por sus autores **Rives, Shamir, Adleman**). La seguridad de este método depende de ciertas propiedades de los números primos, y de la dificultad de encontrar divisores de números muy grandes (cientos de dígitos).

DISTRIBUCIÓN DE CLAVES:

- Los métodos criptográficos (DES y One-time Pad) tienen el problema de la distribución de las claves. Todos los que participan deben conocerlas.

El problema reside en que, por razones de seguridad, las claves deben ser cambiadas con cierta frecuencia; en consecuencia se necesita un canal de distribución de claves. No debería usarse la misma red para transmitirlos, por problemas de seguridad antes mencionados (punto 8.3.5); entonces hay que buscar otra forma de hacerlo que sea equivalente. Esto puede parecer un problema insoluble de nunca acabar. En general se llega a la conclusión de que el cambio será esporádico con una muy baja frecuencia de ocurrencia y se determina utilizar un mensajero de confianza, pues suele ser lo más económico, seguro y confiable frente a sofisticados métodos o tortuosos senderos para distribuir las dichas claves.

Ventajas de la criptografía de clave pública

- No necesita un canal seguro ya que el mensaje sólo lo podrá descifrar quien posea la clave secreta.
- Permite identificar al remitente unívocamente mediante su firma.
- Mediante la clave pública no se puede deducir la clave privada por lo que se asegura la intimidad.

Una de las diferencias fundamentales entre criptografía clásica y moderna radica en el concepto de seguridad. Antes, los procedimientos de cifrado, tenían una seguridad probable; hoy, los procedimientos de cifrado han de tener una seguridad matemática demostrable. Esto lleva a una primera clasificación de seguridad criptográfica:

- Seguridad Incondicional (Teórica): El sistema es seguro frente a un ataque con tiempo y recursos computacionales ilimitados..

- Seguridad computacional (Práctica): El sistema es seguro frente a un ataque con tiempo y recursos limitados. (Ej. Sistema de clave pública basados en problemas de alta complejidad de cálculo)
- Seguridad probable: No se puede demostrar su integridad, pero el sistema no ha sido violado. (Ej. DES)
- Seguridad condicional: Todos los demás sistemas, seguros en tanto que el enemigo carece de medios para atacarlo.

AUTENTICIDAD (FIRMA):

- Si se piensa el RSA al revés, por ejemplo, si encriptamos con una clave D secreta, se puede desencriptar con una clave E pública, y nadie podría falsificar el mensaje, ni siquiera el receptor. Esto podría usarse como "firma" de documentación electrónica.

8.5. Dominios de protección

Mecanismos de Autorización

Un sistema computacional es una colección de procesos y objetos (Hardware, Software, Datos y comunicaciones) en que, cada objeto tiene un nombre único en el sistema y puede ser accedido por un juego de operaciones definidas para ese objeto. Entonces cada objeto tiene un número finito de operaciones que los procesos pueden efectuar sobre él (leer y escribir en archivos, P y V en semáforos). Podemos ver a estos objetos como tipos abstractos de datos.

Los procesos acceden a los recursos que le son permitidos y en cualquier momento solo deben poder acceder a los recursos que necesitan para sus tareas y no a otros. Obviamente, un proceso no debe poder acceder a los recursos que necesitan para sus tareas y no a otros. Obviamente, un proceso no debe poder acceder a los recursos que necesitan para sus tareas y no a otros. Obviamente, un proceso no debe poder acceder a los recursos que necesitan para sus tareas y no a otros. Obviamente, un proceso no debe poder acceder a los recursos que necesitan para sus tareas y no a otros.

Para autorizar a los procesos acceder a los objetos se crearon los Dominios de Protección que es un conjunto de pares (objeto, operaciones) donde cada par identifica un objeto y las operaciones permitidas sobre él.

Entonces un proceso opera con un dominio de protección que especifica los recursos a los que pueden acceder y usar.

Cada dominio define un conjunto de objetos y los tipos de operaciones que se pueden realizar en cada uno de ellos como se indica en la figura 8.12.

La posibilidad de ejecutar una operación en un objeto es un **derecho de acceso**.

Un **dominio** es una colección de derechos de accesos, cada uno de ellos es un par ordenado **í nombre del objeto, conjunto de derechos** Ejemplo: {Archivo Toto, lectura y escritura} el proceso que ejecuta en el dominio D puede solo leer y escribir el archivo Toto y no hacer otra cosa.

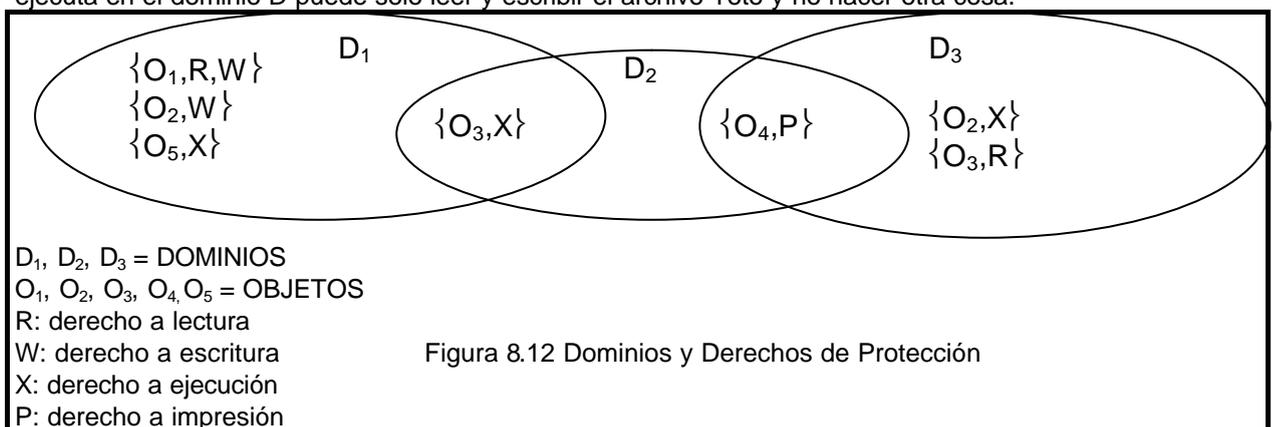


Figura 8.12 Dominios y Derechos de Protección

Cualquier proceso ejecutando en D₂ o D₃, puede imprimir al objeto O₄.

Un proceso en D₃ solo puede ejecutar al objeto O₄ mientras que el mismo objeto solo puede ser leído o escrito por un proceso en D₁.

En cada instante, cada proceso ejecuta dentro de un dominio de protección. Los procesos pueden cambiar de un dominio a otro en el tiempo. El cómo depende mucho del Sistema Operativo. En UNIX por ejemplo, se asocia un dominio a cada usuario + grupo; dado un usuario y el grupo al cual pertenece, se puede construir una lista de todos los objetos que puede acceder y con qué operaciones. Cuando un usuario ejecuta un programa almacenado en un archivo de propiedad de otro usuario B, el proceso puede ejecutar dentro del dominio de protección de A o B, dependiendo del bit de dominio o SETUSERID bit del archivo.

Cuando se hace una llamada al sistema también se produce un cambio de dominio, puesto que la llamada se ejecuta en modo protegido.

8.6. MATRIZ DE ACCESOS

El modelo de protección puede verse de forma abstracta como una matriz en que las filas representan los dominios y las columnas objetos. Cada entrada representa un conjunto de derechos u operaciones que un proceso ejecutando en el dominio D_i pueda evocar sobre el objeto O_j .

DOM \ OBJ.	Archivo 1	Archivo 2	CINTA	IMPRESORA
D_1	Read			Print
D_2		Read	Read	
D_3	Read Write			

Tabla 8.09 Matriz de accesos.

Una política de protección involucra decidir cómo se va a llenar esta matriz.

Normalmente el usuario que crea un objeto es quién decide cómo se va a llenar la columna de la matriz correspondiente a ese objeto. La matriz de acceso es suficientemente general como para apoyar diversas políticas. Por ejemplo: La capacidad para copiar o transferir un derecho de un objeto a otro dominio. Capacidad de un dominio para modificar los derechos en otros dominios (todos, o para un recurso específico). El problema es cómo almacenar esta matriz. Como es una matriz poco densa (muchos de los elementos son vacíos), no resulta práctico representarla como matriz propiamente. Podríamos usar una tabla con triples (dominio, objeto, derechos).

Si un proceso dentro de un dominio D intenta efectuar una operación M sobre un objeto O , se busca (D, O, C) , y se verifica si M pertenece a C . De todas maneras, la tabla es grande, y el esquema no es muy eficiente. Además, si un objeto puede ser, por ejemplo, leído por todo el mundo, debe tener entradas para cada dominio.

8.6.1. Implementación de la Matriz de Accesos.

Se implementa por:

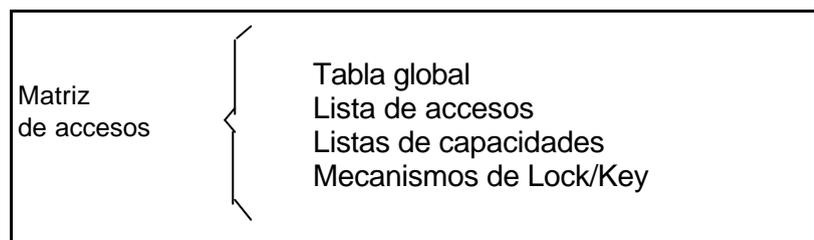


Tabla global

- Compuesto por un conjunto de ternas: **{dominio, objeto, conjunto de derechos}**
- Cuando se ejecuta una operación M en un objeto O_j dentro del dominio D_i se busca en la tabla $\{D_i, O_j, R_x\}$ donde M pertenece a R_x . Si se encuentra la terna se permite la operación sino ocurre una excepción.
- Ventajas:
Sencilla de implantar.
- Desventajas:
 - Tabla muy grande para guardarla en memoria central.
 - Si está en disco hay E/S (con frecuencia se emplean técnicas de memoria virtual para manipular esta tabla).
 - Difícil agrupar objetos o dominios con características similares. Por ejemplo, si todo el mundo puede leer un objeto, debe contar con una entrada en cada dominio.

Lista de accesos

- Es un modo para registrar los derechos de acceso en un sistema.
- Se utiliza frecuentemente en sistemas de archivos.

- Es una enumeración exhaustiva de los derechos de acceso específicos de todas las entidades (dominios o sujetos) que tienen acceso autorizado a un objeto dado.
- Cada columna es la matriz de accesos se puede implementar con una lista de accesos para un objeto.
- Una lista de accesos para un objeto específico es una lista que contiene todas las celdas no vacías de una columna de la matriz de accesos asociada con un objeto dado.
- La lista resultante para cada objeto consiste en pares ordenados í **dominio, conjunto de derechos** que define todos los dominios con un conjunto no vacío de derechos de acceso para ese objeto.
- Cuando se intenta una operación M sobre un objeto O_j en el dominio D_i , consultamos la lista de acceso del objeto O_j en busca de una entrada $\langle D_i, C_k \rangle$, donde $M \in C_k$. Si se localiza la entrada, permitimos la operación; si no está, consultamos el conjunto por omisión. Si M se encuentra allí, permitimos el acceso; de lo contrario, se niega el acceso y se produce una condición de excepción.
- En sistemas que emplean listas de accesos, se mantiene una lista separada para cada objeto. El propietario tiene derecho exclusivo a definir y modificar la lista de accesos asociada.
- Para almacenar la información de acceso en sistemas de archivo se emplean muchas variantes del esquema de lista de accesos. La lista de accesos o un puntero a ella se almacena en el directorio de archivos.
- Ventajas:

Este enfoque puede extenderse fácilmente para definir una lista más un conjunto *por omisión* de derechos de acceso. Por cuestiones de eficiencia se puede consultar primero el conjunto por omisión y luego buscar en la lista de acceso.

El propietario de un objeto puede revocar los derechos de acceso concedidos a un sujeto particular o a un dominio modificando o eliminando simplemente la entrada adecuada en la lista de accesos. Las listas de accesos pueden ser combinadas con otros esquemas para reforzar la protección.
- Desventajas:

Recargo de búsqueda impuesto por la necesidad de verificar la autoridad de un sujeto para acceder al objeto solicitado.

De acuerdo con el principio de mediación completa, toda petición de acceso a un archivo debería ser verificada. Con el fin de mejorar la eficiencia, algunos sistemas comprueban la autoridad del solicitante solo durante la apertura del archivo. Esto debilita la protección abriendo la puerta a la penetración después que el archivo está abierto y haciendo no efectivas las revocaciones de privilegio en tanto el usuario tenga el archivo abierto, lo cual puede ser indefinidamente en algunos sistemas.
- Con el fin de evitar almacenamiento y búsqueda de listas potencialmente muy largas de usuarios autorizados, especialmente para archivos públicos, algunos sistemas dividen a los usuarios en clases (grupos) y sólo almacenan los derechos de acceso de los grupos. Este esquema ahorra almacenamiento y acelera el procesamiento a costa de reducir flexibilidad y de limitar el número de dominios de protección de archivos distintos a un pequeño número de clases de usuario disponibles distintas. En UNIX, por ejemplo, las listas de accesos están reducidas a tres entradas por archivo, una para el propietario, otra para el grupo y otra para todos los usuarios (el mundo).

Lista de capacidades (capability)

- El orden está dado por las filas de la matriz, donde cada fila es para un dominio.
- Una lista de capacidades para un dominio es una lista de objetos, y las operaciones permitidas sobre éste.
- Un objeto, en general, está representado por su nombre o dirección (llamado capacidad).
- Para ejecutar la operación M , especificando como parámetro la capacidad (apuntador o puntero) para el objeto O_j . Basta la posesión de la capacidad para que se permita el acceso.
- La lista de capacidades está asociada a un dominio, pero un proceso que se ejecuta en ese dominio no puede acceder a ella directamente.
- La lista de capacidades es un objeto protegido, mantenido por el sistema operativo y al cual el usuario sólo puede tener acceso indirecto.
- Omitiendo las entradas vacías, el resultado es una lista exhaustiva de los objetos a los que un sujeto específico está autorizado a acceder (una fila en la matriz de acceso).
- Conceptualmente una **capacidad** es una cédula o billete que da al usuario que lo posee permiso para acceder a un objeto específico en la manera especificada. Las capacidades proporcionan un mecanismo único y unificado para:
 1. Direccional memoria primaria y secundaria.
 2. Acceder a recursos hardware y software.
 3. Proteger objetos en memoria primaria y secundaria.
- Una capacidad puede representarse como una estructura de datos formada por dos elementos de información:

1. Un identificador de objeto único (normalmente un puntero).
 2. Los derechos de acceso para ese objeto.
- La protección basada en capacidades se apoya en que nunca se permite que las capacidades se muevan a un espacio de direcciones accesible directamente por un proceso de usuario (donde podría modificarse). Si todas las capacidades están seguras, el objeto que protegen está también seguro frente al acceso no autorizado.
 - Generalmente las capacidades se distinguen de otros datos de una de estas dos maneras:
Cada objeto tiene una **etiqueta** que especifica su tipo como capacidad o dato accesible. Los programas de aplicación no deben tener acceso directo a las etiquetas; debe usarse hardware o firmware para aplicar esta restricción. Aunque sólo se requiere un bit para distinguir entre las capacidades y otros objetos, con frecuencia se emplean más bits. Esta extensión permite que el hardware etiquete todos los objetos con su tipo; así el hardware puede distinguir enteros, números de coma flotante, apuntadores, valores booleanos, caracteres, instrucciones, capacidades y valores sin inicializar, utilizando sus etiquetas.

Otra manera consiste en dividir en dos partes el espacio de direcciones asociado a un programa. Una parte es accesible al programa y contiene sus datos e instrucciones normales; la otra parte, que contiene la lista de capacidades, sólo es accesible por sistema operativo. Para apoyar esta estrategia es útil un espacio segmentado de memoria.

- Cuando se utilizan para acceder a la memoria central, las capacidades funcionan de manera análoga a la segmentación. Esto ocurre porque ambos, segmentos y capacidades, direccionan entidades lógicas. En vez de los descriptores de segmentos almacenados en tablas de mapas de descriptores, los sistemas basados en capacidades utilizan capacidades almacenadas en listas de capacidades. Una diferencia importante es que las capacidades son un mecanismo mucho más general que puede direccionar tanto objetos hardware como software y tanto en memoria central como secundaria.

Las propias listas de capacidades son objetos protegidos sólo accesibles al sistema operativo. Los usuarios pueden manipular las capacidades sólo por medio de las funciones suministradas por el sistema que típicamente incluyen operaciones para:

- Trasladar una capacidad a un lugar diferente dentro de una lista.
 - Eliminar una capacidad.
 - Restringir la parte de derechos de acceso de una capacidad.
 - Transferir una capacidad como parámetro.
 - Transmitir una capacidad a otro usuario.
- **Ventajas:**
 - Los sistemas basados en capacidades combinan las funciones de direccionamiento y protección en un solo mecanismo unificado que se utiliza para acceder a todos los objetos del sistema.
 - La lista de capacidades de un proceso puede ser implementada de una manera jerárquica, dando así a diferentes partes del programa diferentes derechos de acceso. Diferentes partes de un programa pueden tener derecho de acceso a diferentes objetos.
 - Flexibilidad y facilidad de uso tanto para el sistema como para los objetos.
 - **Desventajas:**
 - Posibilidad de falsificaciones.
 - Dificultad de revocación de privilegios de acceso.
 - Una primitiva implementación software de la protección basada en capacidades fue el sistema operativo Hydra.

Mecanismo lock/key

- El mecanismo de Lock/key es un compromiso entre listas de acceso y listas de capacidad.
- Cada objeto tiene una lista de bits de **candado** (locks), y cada dominio tiene otra lista de bits de **llave** (keys).
- Un proceso solo puede ejecutar (acceder) al objeto si el dominio al cual pertenece tiene llaves que coincidan con los candados del objeto.
- Las llaves del dominio son manejadas solo por el sistema operativo.
- A un sujeto S se le da una llave K_i para la cerradura L_i , sólo si tiene el derecho de acceso R_i para el objeto asociado.
- Una lista de cerraduras es una columna de la matriz de accesos en donde las entradas no vacías idénticas pueden representarse con un único par (L_i, R_i) .
- Una llave para una cerradura es una forma de capacidad que da derecho al propietario a acceder al objeto, suponiendo que al llave encaje en la cerradura correspondiente.
- El propietario del objeto puede revocar los derechos de acceso de todos los objetos que comparten la llave K_i , suprimiendo la entrada de la cerradura L_i .

- **Ventajas:**

Las llaves pueden pasarse libremente de un dominio a otro y, además, es posible cancelar eficazmente los privilegios de acceso con sólo cambiar algunas de las llaves asociadas al objeto.

Comparación de las implementaciones

Las Listas de Acceso corresponden a las necesidades de usuario. Cuando se crea un objeto específica a qué dominio pertenece. En sistemas grandes, la búsqueda puede llegar a ser muy grande o muy costosa. El cuadro comparativo de la Tabla 8.10 muestra las ventajas y desventajas de cada una de las implementaciones.

- Las Listas de Capacidades no son sencillas de implementar por el usuario. Es eficiente una vez implementado, pues solo es necesario verificar que la capacidad es válida. La revocación es muy complicada, porque las capacidades están distribuidas por todo el sistema.

	VENTAJAS	DESVENTAJAS
Tabla global	-Sencilla Implementación	-Tabla muy grande para guardarla en memoria central. -Dificultad para agrupar objetos o dominios con características similares.
Lista de accesos	-Puede combinarse con otros esquemas para reforzar la protección. -El propietario de un objeto puede revocar los derechos de acceso concedidos a un sujeto particular o a un dominio.	-Recargo de búsqueda impuesto por la necesidad de verificar la autoridad de un sujeto para acceder al objeto solicitado. -En algunos sistemas se debilita la protección cuando comprueban la autoridad del solicitante sólo durante la apertura del archivo.
Listas de capacidades	-Combinan las funciones de direccionamiento y protección en un solo mecanismo. -Diferentes partes de un programa pueden tener acceso a diferentes objetos. -Flexibilidad y facilidad de uso tanto para el sistema como para los objetos.	-Posibilidad de falsificaciones. -Dificultad de revocación de privilegios de acceso.
Mecanismos lock/key	-Las llaves pueden pasarse libremente de un dominio a otro, y además, es posible cancelar eficazmente los privilegios de acceso con sólo cambiar algunas de las llaves asociadas al objeto. -Es efectivo y flexible.	

Tabla 8.10 Comparación de las ventajas y desventajas de cada una de las implementaciones

- El mecanismo de Llave/Candado es una solución de compromiso. Es efectivo y flexible, y cualquier cambio solo requiere cambiar la configuración de unos pocos bits.

La mayoría de los sistemas utilizan una combinación de listas de acceso y capacidades. Cuando un proceso trata por primera vez de acceder a un objeto, se consulta la lista de acceso. Si se niega el acceso, se produce una condición de excepción; de lo contrario, se crea una capacidad y se une al proceso. Las referencias posteriores utilizan la capacidad para demostrar rápidamente que está permitido el acceso; después del último acceso se destruye la capacidad. Esta estrategia fue empleada en los sistemas MULTICS.

Utilización de Listas de Acceso en Redes:

Las primeras redes enrutadas conectaban una pequeña cantidad de LANs y de hosts. Luego, luego se extendieron las conexiones del router a redes heredadas y a redes de socios externos. Con el incremento del uso de Internet se plantearon nuevos desafíos al control de acceso. La tecnología más reciente- desde backbones ópticos a servicios de banda ancha y a switches LAN de alta velocidad - aumentó nuevamente los desafíos de control.

Nos enfrentamos al siguiente dilema: ¿cómo denegar las conexiones no deseadas y a la vez permitir un acceso apropiado? Si bien otras herramientas ayudan como las contraseñas, los equipos de respuesta de

llamada y los dispositivos físicos de seguridad, éstas suelen carecer de la expresión flexible y de los controles específicos que prefieren la mayoría de los administradores.

Las listas de acceso ofrecen otra herramienta poderosa para el controlar red. Estas listas agregan la flexibilidad de filtrar el flujo de paquetes que entran o salen de las interfases del router. Las listas de acceso ayudan a proteger la expansión de los recursos de la red sin impedir el flujo de comunicaciones legítimas. Las listas de acceso diferencian el tráfico de paquetes en categorías que permiten o deniegan otras características. También se pueden utilizar las listas de acceso para:

Identificar los paquetes para colas de prioridad o determinadas por el cliente.

Restringir o reducir el contenido de las actualizaciones de enrutamiento.

Las listas de acceso también procesan los paquetes para otras características de seguridad a fin de:

Brindar un control de acceso dinámico al tráfico IP con una autenticación de usuario mejorada utilizando las características de bloqueo (*lock*) y clave (*key*).

Identificar los paquetes para la encriptación

Identificar el acceso Telnet permitido a las terminales virtuales del router.

Comparado con una LAN o con el networking interdepartamental, el tráfico que utiliza el enrutamiento por llamada **telefónica bajo demanda** (DDR) en general tiene un volumen escaso y periódico. El DDR inicia una llamada WAN a un sitio remoto sólo cuando hay tráfico para transmitir. A fin de identificar este tráfico, se especifican los paquetes que los procesos DDR en el router interpretarán como tráfico "interesante".

Cuando se realiza la configuración para DDR, se debe ingresar comandos de configuración que indiquen qué paquetes de protocolo constituyen tráfico interesante para dar inicio a la llamada. Para hacerlo, se ingresa instrucciones de la lista de acceso para identificar las direcciones de origen y destino, y se eligen ciertos criterios de selección específica de protocolos para iniciar la llamada..

Las listas de acceso son instrucciones que especifican condiciones que se definen para que el router maneje el tráfico cubierto por la lista de acceso de manera extraordinaria. Las listas de acceso brindan un control adicional para el procesamiento de paquetes específicos de una forma única. Los dos tipos principales de listas de acceso son:

Listas de acceso estándar: Las listas de acceso estándar para IP revisan la dirección de origen de los paquetes que pueden enrutarse. El resultado permite o deniega la salida de una suite de protocolos completa en base a la dirección de red/subred/host.

Por ejemplo, se revisan la dirección y el protocolo de los paquetes que entran por E0. Si están permitidos, los paquetes salen a través de S.0., que está agrupado en la lista de acceso.

Si la lista de acceso estándar deniega los paquetes, todos estos paquetes de la categoría determinada son desechados.

Listas de acceso extendidas: Las listas de acceso extendidas revisan tanto la dirección de origen como la dirección de destino. También pueden revisar protocolos específicos, números de puerto y otros parámetros. Esto permite una mayor flexibilidad a los administradores para describir qué revisión debe realizar la lista de acceso. La salida de los paquetes se puede permitir o denegar en base a su origen o destino.

La lista de acceso extendida también permite o deniega con mayor granularidad. Por ejemplo, puede permitir el tráfico de correo electrónico entre E0 y destinos específicos de S.0. mientras deniega logins remotos o transferencia de archivos.

Funcionamiento de las listas de acceso en redes

Las listas de acceso expresan el conjunto de reglas que brindan un control adicional sobre los paquetes que ingresan por las interfaces de entrada, sobre los paquetes que pasan a través del router y sobre los paquetes que salen por las interfaces de salida del router. Las listas de acceso no actúan sobre los paquetes que se originan en el router mismo.

El comienzo del proceso es el mismo, independientemente de que se utilicen o no listas de acceso: Cuando un paquete ingresa a una interfase, el router verifica si éste es enrutable (o apto para ser tratado por un bridge). Si alguna de estas situaciones es falsa, el paquete es desechado. Una entrada de la tabla de enrutamiento indica una dirección de destino, alguna métrica o estado de enrutamiento y la interfase a utilizar.

A continuación el router revisa para ver si la interfase de destino está agrupada con una lista de acceso. Si no es así, el paquete puede enviarse al buffer de salida; por ejemplo, si va a utilizar To0, que no tiene listas de acceso activadas, el paquete utiliza directamente To0.

La interfase E0 se ha agrupado con una lista de acceso extendida. Se emplearon expresiones precisas y lógicas para definir la lista de acceso. Antes de que un paquete pueda proceder hacia esa interfase, se lo prueba mediante una combinación de instrucciones de la lista de acceso asociadas a dicha interfase.

En base a las pruebas de la lista de acceso extendida, el paquete se puede autorizar. Para las listas de entrada esto significa continuar procesando el paquete después de recibirlo en una interfase de entrada. Para las listas de salida, esto significa enviarlo al buffer de salida para E0; de lo contrario, los resultados de las pruebas pueden denegar el permiso. Esto significa descartar el paquete. La lista de acceso del router brinda control de firewall para denegar el uso de la interfase E0. Al descartar los paquetes, algunos protocolos devuelven un paquete especial al emisor. Este paquete notifica al emisor que el destino es inalcanzable.

Estructuras de protección dinámicas

- La asociación entre un proceso y un dominio puede ser **estática** si los recursos disponibles para ese proceso lo serán para toda su ejecución, o **dinámico** si existen cambios de accesos o dominios. Si se permiten estos cambios, posiblemente sean violadas las protecciones.
- Un mecanismo que permite implementarlo es incluir a los mismos dominios como objetos de la matriz de accesos, y cuando sean necesarios cambios en los accesos incluimos a la propia matriz como objeto. Como lo que se quiere es que cada entrada pueda ser modificada en forma individual, se considera cada entrada como un objeto.

8.6.2. Cambio de Dominio - Switch

D \ O	F ₁	F ₂	F ₃	Lector Tarjetas	Impresora	D ₁	D ₂	D ₃	D ₄
D ₁	Read		Read				switch		
D ₂				Read	Print			switch	switch
D ₃		Read	Execute						
D ₄	Read Write		Read Write			switch			

D₍₂₎ puede cambiar a D₍₄₎ o D₍₃₎

D₍₄₎ puede cambiar a D₍₁₎ pero no a D₍₂₎ Tabla 8.11

D₍₁₎ puede cambiar a D₍₂₎ pero no a D₍₃₎.

- Un proceso puede cambiar del dominio D_(i) al dominio D_(j) si el derecho de acceso SWITCH pertenece a ACCESO_(i,j).
- Ejemplo en la tabla 8.11 se cambia de dominio del 1 al 4 , del 2 al 1, del 3 al 2 y del 4 al 2.

8.6.3. Cambio de contenido de la Matriz de Accesos

- El permitir cambios controlados en la matriz de acceso requiere de tres operaciones adicionales: copy, owner y control.

COPY:

- La capacidad de copiar un derecho de acceso de un dominio a otro (fila) de la matriz de accesos se indica agregando un asterisco (*) al derecho de acceso.
- El derecho COPY solo permite copiar el derecho de acceso dentro de la misma columna (es decir, para el mismo objeto) en la cual está definido tal derecho.
- Ejemplo:

D \ O	F ₍₁₎	F ₍₂₎	F ₍₃₎
D ₁	READ		* WRITE
D ₂		* READ	EJECUTAR
D ₃	READ		

Tabla 8.12.a.

Se transforma en:

D \ O	F ₍₁₎	F ₍₂₎	F ₍₃₎
D ₁	READ		* WRITE
D ₂		* READ	EJECUTAR
D ₃	READ	READ	

Tabla 8.12.b

Tabla 8.12 Cambio de contenidos en la matriz..

* Habilita el copiado del "acceso" en la misma columna (objeto).

Un proceso ejecutando en D₍₂₎ puede "copiar" el acceso Read en cualquier entrada asociada a F₍₂₎.

- Algunas variantes de esto podría ser:

- Transfer: se pasa de **acceso (i,j)** al **acceso (k,j)** pero se quita el acceso (i,j) (Removed).
- Copia limitado: se pasa solo el acceso **R** y no la capacidad **R***.

⇒ **DUEÑO (OWNER):**

- Se necesita además del copiado algún mecanismo para agregar nuevos derechos y eliminar otros, el derecho de acceso OWNER controla estas operaciones.
- Si en el acceso (i,j) tenemos Owner, significa que un proceso ejecutando en D_(i) puede agregar o quitar accesos en toda la columna j.
- Ejemplo:

D \ O	F ₍₁₎	F ₍₂₎	F ₍₃₎
D ₁	READ OWNER		WRITE
D ₂		* READ OWNER	* READ OWNER
D ₃	READ		

Tabla 8.13a.

Se transforma en:

D \ O	F ₍₁₎	F ₍₂₎	F ₍₃₎
D ₁	READ OWNER		
D ₂		* READ/WRITE OWNER	* READ OWNER
D ₃		WRITE	WRITE

Tabla 8.13b.

Tabla 8.13 Cambio de contenidos para Dueño.

⇒ **CONTROL:**

- El Copy y el Owner permiten que un proceso altere entradas en una columna. Un mecanismo para alterar entradas en las filas es asimismo necesario. Tal mecanismo es el derecho CONTROL que habilita cambios en otras filas (dominios).
- Ejemplo:

D \ O	F ₍₁₎	D ₍₁₎	D ₍₂₎	D ₍₃₎
D ₁	READ		SWITCH	
D ₂	READ			SWITCH CONTROL
D ₃	READ/WRITE	SWITCH		

Tabla 8.14.a

Se transforma en:

D \ O	F ₍₁₎	D ₍₁₎	D ₍₂₎	D ₍₃₎
D ₁	READ		SWITCH	
D ₂	READ			SWITCH CONTROL
D ₃	READ	SWITCH		

Tabla 8.14.b

Tabla 8.14 Cambio de contenidos para control.

O sea que D₍₂₎ puede cambiar el modo de acceso a F₍₁₎ de D₍₃₎ de R/W a R.

Lo importante de esto es que con estos esquemas y los conceptos de objetos y capacidad, es posible crear un nuevo tipo de monitor, llamado **manager**, que es usado para cada recurso, el cual planifica y controla el acceso a ese recurso. Cuando un proceso necesita un recurso, llama al manager, el cual le devuelve la capacidad para ese recurso. El proceso debe presentar la capacidad cuando usa el recurso. Cuando el proceso finaliza el uso del recurso devuelve la capacidad al manager, quien lo asignará a otro proceso, de acuerdo a su planificador (scheduler).

8.6.4. Revocación de permisos

- En protección dinámica, cuando se revocan los accesos, se puede tener varias opciones:
 - Inmediata / Postergada.
 - Selectiva / General (para algunos usuarios o para todos).
 - Parcial / Total (todos los accesos a un objeto, o solo algunos).

- Temporario / Permanente (se podrá obtener nuevamente o no).
- En la Lista de Accesos la revocación es fácil. Se busca la Lista de Accesos y se hacen los cambios, luego cualquier combinación anterior es posible.
- En el esquema de Listas de Capacidad, la revocación (cancelación) es más dificultosa, pues las capacidades están distribuidas por todo el sistema. Los distintos sistemas de revocación en Listas de Capacidades son los siguientes:
 - **Readquisición:** las capacidades son borradas periódicamente. Si un proceso intenta readquirirla, y la capacidad se revocó, NO puede hacerlo ni se lo debe permitir algún intento.
 - **Back-pointers (punteros inversos):** una lista de punteros asocia a cada objeto con todas las capacidades asociadas. Siguiendo los punteros es posible cambiarlos. La implementación es general pero muy costosa. Fue utilizado en MULTICS.
 - **Indirección:** las capacidades no apuntan al objeto en forma directa, sino a una única entrada en una tabla global, la cual, a su vez, apunta al objeto. La revocación se implementa buscando en la tabla global la entrada y borrándola. Puede ser reutilizada por otra capacidad. No se puede hacer revocación selectiva. Deben concordar el objeto designado por una capacidad y su entrada en la tabla.
 - **Llaves:** la revocación se hace cambiando la llave. Si un objeto tiene asociadas varias llaves es posible hacer revocación selectiva.

8.6.5. Algunos Sistemas de Seguridad en Sistemas Centralizados.

UNIX:

- Cada archivo tiene asociado 3 campos (dueño, grupo, universo). Cada campo tiene 3 bits: R (read), W (write), X (execution).
- Un dominio está asociado con el usuario.
- Hacer switch de dominio corresponde a cambiar la identificación del usuario temporariamente. Cuando un usuario A comienza ejecutando un archivo de dueño B, el usuario es "seteado" a B, y cuando termina, es "reseteado" a A.

MULTICS:

- Su sistema de protección está sobre sus sistemas de archivos (todo es archivo), y cada archivo tiene asociada una lista de acceso, como el acceso del dueño y del universo.
- Tiene una estructura de anillos que representan los dominios, tal que dado los dominios $D_{(i)}$ y $D_{(j)}$, con $i < j$, $D_{(i)}$ tiene mayores privilegios que $D_{(j)}$. Dado solo dos dominios, nos encontraríamos frente a un sistema maestro/esclavo. $D_{(0)}$ tiene el mayor privilegio de todos.
- El espacio de direcciones es segmentado, y cada segmento tiene asociado su número de anillo y sus bits de acceso (R, W, X). Cuando un proceso ejecuta en $D_{(i)}$ no puede acceder a un segmento de $D_{(j)}$ si $j < i$, pero sí a uno de $D_{(k)}$ si $i < k$, respetando, desde ya, los bits de acceso.
- Para llamar a procesos de anillos inferiores, se produce un trap (system call o SVC), entonces es más controlado.
- Protecciones de la forma "capacidad" fueron implementados en los sistemas como el Hydra y el Cambridge CAP System.

8.6.6. Planteo de un problema clásico en un Sistema Operativo.

Sean:

- Un conjunto de objetos $R = \{V_1\}$ a proteger y cuyo uso debe controlarse (Ejemplos: archivos, semáforos, segmentos, páginas, procesos, periféricos, etc.)
- Un conjunto de usuarios $U = \{u_i\}$ que son los que pueden producir errores o violaciones.
- Una matriz $||M||$ donde las líneas representan a los usuarios u_i , y las columnas a los objetos r_j .
- Cada elemento m_{ij} de la matriz representa el conjunto de permisos de accesos a r_j por u_i que depende del usuario y del recurso (lectura de un archivo, creación de procesos, ejecución, etc.)

La matriz M define las reglas de protección que se tiene que respetar.

R	Arch. A	B	Proc. X	Proc. Y	Impresora	Objeto 2
Proc 1						
Proc 2						
Proc 3						
Proc 4						

Derechos (capabilities) de un usuario u_i sobre el objeto v_j al elemento m_{ij} .

Poder de un usuario u_i a la línea m_{ij} . matriz derechos (acceso matriz) a la matriz M.

Tabla 8.15.

8.7. SEGURIDAD EN EL KERNEL

- Es necesario introducir el concepto de seguridad en el diseño del Kernel desde el origen de la idea o de la concepción.
- Debe ser intrínsecamente seguro.
- Debe asegurar las funciones de:
 - Control de accesos.
 - Las operaciones de entradas y salidas al y del sistema
 - La supervisión de la administración de:
 - almacenamiento real en Memoria Central y soportes
 - almacenamiento virtual
 - el Sistema de Archivos (file system)
 - Debe ser simple y pequeño (pequeño es hermoso: UNIX) facilita la revisión para detectar fallas y demostrar que funciona correctamente.

Algunos sistemas utilizan un concepto de seguridad de los datos basado en transacciones. Este es el caso de los servidores de bases de datos por lo que lo detallamos a continuación.

8.7.1. Transacciones

- La mutua exclusión de regiones críticas aseguran que ellas son ejecutadas atómicamente, es decir, el resultado es equivalente a sus ejecuciones en forma secuencial en algún orden desconocido. Esta propiedad no siempre es útil, hay muchos casos donde nos gustaría asegurarnos de que una región crítica forme una unidad lógica de trabajo que es realizada totalmente, o no se ejecute nada.

Modelo de sistema transaccional:

- **Transacción:** es una colección de instrucciones que realizan una función lógica. Un gran tema en el proceso de transacciones es la preservación de la atomicidad a pesar de la posibilidad de fallas en el sistema.
- Una transacción es una unidad de programa que accede y posibilita actualizar datos que pueden residir en archivos en un disco.
- Desde nuestro punto de vista es una secuencia de operaciones de **read** y **write**, terminadas por una operación **commit** o una operación **abort**.
- Una operación **commit**, significa que la transacción ha terminado su ejecución exitosamente, mientras que una operación **abort** significa que la transacción ha cesado su ejecución por causa de algún error lógico.
- Una transacción puede terminar su ejecución a causa de una falla en el sistema; en este caso, como una transacción abortada pudo modificar datos que accedió el estado de esos datos pueden no ser los mismos a que si la transacción se hubiera ejecutado atómicamente (consistencia de los datos). Tal que la propiedad de atomicidad se asegura, una transacción abortada no debe tener efecto sobre el estado de los datos que se modificó. Entonces, el estado de los datos accedidos por una transacción abortada debe ser restauradas al estado anterior a que se empieza la ejecución. Esto se conoce como **rolled back**.

Recuperación basada en un log:

- Una manera de asegurar la atomicidad es grabar, en almacenamiento estable (disco), información acerca de todas las modificaciones hechas por la transacción a los datos que ha accedido.
- El método más usado para realizar esto se denomina **write-ahead logging**. El sistema mantiene en almacenamiento estable, una estructura llamada **log (Archivo log)**. Cada registro del log describe una operación de una transacción de escritura (**write**) y generalmente tiene los siguientes campos:
 - ◇ **Nombre de la transacción:** El nombre único de una transacción que realizó una operación de write.
 - ◇ **Nombre del dato:** El nombre único del dato escrito.
 - ◇ **Valor anterior:** El valor del dato anterior a la escritura.
 - ◇ **Nuevo valor:** El valor del dato que tendrá después de la escritura.
- Existen otros registros especiales para guardar eventos significantes durante el proceso de la transacción, tal como el comienzo, el **commit** o **abort** de una transacción.

- Antes de la transacción T_i comience su ejecución, el registro $\langle T_i \text{ start} \rangle$ se escribe al log. Durante su ejecución, cualquier operación de **write** de T_i es precedida por la escritura de los nuevos registros al log. Cuando T_i **commit**, el registro $\langle T_i \text{ commits} \rangle$ se escribe al log.
- Esta información se usa para reconstruir el estado de los datos accedidos por las transacciones.
- Requerimos que antes de que una operación **write (x)** se ejecute, los registros del log correspondientes a X deben ser escritos a un almacenamiento estable.
- Se produce una pérdida de rendimiento en el sistema al realizar estas escrituras.
- Los algoritmos de recuperación usan dos procedimientos:
 - **undo** (T_i), el cual restaura el valor de todos los datos actualizados por la transacción T_i a los valores anteriores.
 - **redo** (T_i), el cual fija el valor de todos los datos actualizados por la transacción T_i a los nuevos valores.
- El conjunto de datos actualizados por T_i , y sus respectivos valores anteriores y nuevos pueden ser encontrados en el log.
- Si una transacción T_i aborta, podemos restaurar el estado de los datos que han sido modificados ejecutando una operación de **undo** (T_i). Si ocurre una falla del sistema, restauraremos el estado de todos los ítems actualizados consultando en el log para determinar cual transacción necesita ser rehecha o cual debe ser deshecha.
- Transacción T_i necesita ser deshecha si el log contiene el registro $\langle T_i \text{ start} \rangle$ pero no contiene el registro $\langle T_i \text{ commits} \rangle$.
- Transacción T_i necesita ser rehecha si el log contiene los registros $\langle T_i \text{ start} \rangle$ y $\langle T_i \text{ commits} \rangle$.

Una importante observación: El log crece en función de las transacciones por lo que se debe dimensionar adecuadamente.

Checkpoints

- Cuando en un sistema ocurre una falla, debemos consultar al log para determinar cuales datos de las transacciones son necesarios para ser rehecha y aquellas que necesita para ser deshecha. Entonces se necesita buscar la entrada al log para hacer estas determinaciones.
- Hay dos importantes causas para ese aprovechamiento (drawbacks):
 - 1) El proceso buscado es tiempo consumido
 - 2) Muchas transacciones necesitan ser rehechas y ya tienen actualizados los datos que el log necesita para modificar. Si bien rehacer las modificaciones de los datos no causan daño, sin embargo la recuperación toma mucho tiempo.

Para reducir el overhead, se introduce el concepto de **checkpoints**.

- Durante la ejecución, el sistema mantiene el write-ahead log.
- El sistema periódicamente chequea los checkpoints, los cuales requieren la siguiente secuencia de acciones:
 - a) Salida de todos los registros de log que residen en el almacenamiento volátil (usualmente memoria central) sobre un almacenamiento estable (disco).
 - b) Salida de todos los datos modificados que residen en almacenamiento volátil sobre un almacenamiento estable.
 - c) Salida de un registro de log **checkpoint** sobre un almacenamiento estable.
- La presencia de un registro **checkpoint** en el log permite que el sistema actualice sus procedimientos recuperados.
- Si se considera una transacción T_i cometida previa al checkpoint. El registro $\langle T_i \text{ commits} \rangle$ aparece en el log antes del registro **checkpoint**. Cualquiera de las modificaciones hechas por T_i tienen que haber sido escrito al almacenamiento estable previo al checkpoint o como parte del checkpoint mismo. El tiempo de recuperación no necesita chequear la operación **redo** sobre T_i .
- Esta observación nos permite refinar nuestro algoritmo de recuperación previo.
- Después que ha ocurrido una falla, la rutina de recuperación examina el log para determinar la transacción T_i más reciente que comenzó ejecutando antes del checkpoint más reciente.

8. 8. AUTENTICACIÓN DEL USUARIO

Este es uno de los problemas de seguridad más importantes. Se trata de identificar a los distintos usuarios del sistema y a partir de allí contar con un mecanismo de protección que identifique que procesos y programas que están ejecutándose. La autenticación se basa en uno o la combinación de algunos de tres conjuntos de ítems: posesión del usuario (clave o tarjeta), conocimiento del usuario (identificación y contraseña), y atributos del usuario (huella dactilar, pattern ocular o grabado o firma).

La forma más común es pedirle al usuario el password o contraseña. Si lo tecleado por él es idéntico a lo almacenado por el sistema, se considera al usuario como válido y se le permite el acceso al sistema. En UNIX, por ejemplo, existe un archivo de contraseñas en donde cada usuario se identifica con el login y su password.

Existen diversas formas tanto de presentar la contraseña al usuario como de adivinarla por parte del intruso. La tendencia del usuario es elegir nombres obvios de contraseña unida a una cantidad de caracteres reducida lo que posibilita una rápida adivinación de la misma. Para contrarrestar esto se cuenta con la posibilidad de agregar caracteres especiales y mayúsculas-minúsculas diferenciadas. Otra forma es aumentar el número de caracteres, o combinar la contraseña con un número antes de ocultarla o usar contraseñas con una palabra del sistema a la que debe el usuario agregar una parte, o una función entera de la que se da el argumento y el usuario debe responder el resultado, cambio de contraseña con cada sesión o programas que ayudan al usuario a elegir contraseñas mediante palabras sin sentido pero de fácil adivinación son algunas de las formas de dificultar la adivinación. Ciertos sistemas informan al usuario de la caducidad de la contraseña forzándolo a cambiarla.

El mayor riesgo de la contraseña es que se coloque un programa residente que registre lo tecleado. Además, si las contraseñas que obliga el sistema a usar son largas y complicadas, puede que este método sea más perjudicial que beneficioso, por el ocasional olvido.

Muchos de los sistemas de protección se basan en la hipótesis de que el sistema conoce la identidad de cada usuario. El problema de identificar a los usuarios cuando están conectados al sistema y se conoce como la autenticación del usuario.

Muchos de los métodos de autenticación se basan en la identificación de algo conocido por el usuario.

- **Situaciones:**

- 1) - Accesos no permitidos
- 2) - Uso compartido de información
- 3) - Modificación de Información
- 4) - Falla de Hardware - Corte de luz

- **Herramientas:**

1. Nombres claves
 - Password - palabra clave
 - Lista de acceso (tabla con usuario + modo de acceso + permiso)
 - Niveles de usuario, grupo, universo (user - owner - group - universe)
2. Permiso de operaciones sobre Archivos o Directorios:
READ, WRITE, EXECUTE, OPEN, CLOSE, UPDATE, NOUPDATE, CREATE, DELETE, ERASE.
 - Uso compartido de ARCHIVOS:
 - Accesos protección a escritura - lectura - ejecución - simultáneo FCB (File Control Block) y Tablas de Permisos
3. Fallas o errores:
 - BACK UP - CIERRE

Es importante tener en cuenta que existen varias formas para ingresar a los sistemas de cómputos y que la mayoría son por dejadez de los usuarios del sistema.

Veremos más en detalle el tema de la validación del usuario.

8.8.1. Validación

- Objetivo: Permitir acceso a los usuarios legítimos del sistema y denegar a los no autorizados.
- Principales medidas de la validación:
 - 1) **La tasa de falsas aceptaciones:** el porcentaje de usuarios legítimos erróneamente admitidos.
 - 2) **La tasa de falsos rechazos:** el porcentaje de usuarios legítimos a los que se deniega acceso debido a un fallo en el mecanismo de validación.

El objetivo es minimizar ambas tasas.

- La validación unidireccional se basa en:
 - Posesión de un secreto (contraseña).
 - Posesión de un artefacto (tarjeta, llave, etc.).
 - Características fisiológicas o de comportamiento específicas del usuario.

Contraseñas (passwords):

- La contraseña es el mecanismo de validación más común utilizado en ambientes computacionales y su base se sustenta en compartir un secreto.
- Cada usuario tiene una contraseña, que puede ser inicialmente asignada por el sistema o por el administrador.
- El sistema almacena todas las contraseñas de los usuarios en un archivo especial y la utiliza para validarlos mediante la coincidencia de la consulta ante sus ingresos o accesos a recursos.

- **Ventajas:**

Las contraseñas son populares ya que no requieren un hardware especial y son relativamente fáciles de implementar.

- **Desventajas:**

Ofrecen protección limitada, ya que pueden ser relativamente fáciles de adivinar o de obtener:

- * Los archivos de contraseñas no cifrados almacenados en un sistema son presa fácil.
 - * La inspección y búsqueda de basura puede descubrir algunas contraseñas intercambiadas por usuarios mediante correo electrónico.
 - * Las contraseñas son generalmente palabras del diccionario o nombres propios, que son fáciles de recordar pero también de adivinar.
 - * Las contraseñas elegidas por el sistema son combinaciones aleatorias de letras y números que son difíciles de adivinar pero también difíciles de recordar.
- Se han propuesto varias técnicas para reforzar el nivel de protección de las contraseñas. La mayoría de ellas tienen desventajas que reducen su efectividad o su aceptación por el usuario:
 1. Los esquemas de contraseña pueden ser multinivel, y se puede requerir que los usuarios suministren contraseñas adicionales a petición del sistema a intervalos aleatorios durante el uso del computador. Esto fastidia a los usuarios legítimos lo que va contra uno de los principios de la seguridad (Aceptación del usuario)
 2. Otra técnica es hacer que el sistema genere un reto dinámico al usuario después del inicio de la sesión. Este reto puede tomar la forma de un número aleatorio generado por el computador, al cual se supone que el usuario aplica una transformación secreta, como por ejemplo hallar la raíz cuadrada e incrementar el valor. El error o acierto en la obtención del resultado puede ser utilizado para detectar usuarios no autorizados.
 3. Un modo de tratar el problema es limitar el número de intentos consecutivos de aperturas de sesión desde un destino dado y proceder a la consiguiente desconexión de la línea. Desgraciadamente, la desconexión de la línea no protege ni detiene a los marcadores automáticos. Algunos sistemas confían en la retrollamada, en donde el usuario se identifica a sí mismo y el sistema informático marca el número conocido asociado con el ID del usuario. Los sistemas con retrollamada son caros tanto en equipos adicional como en costo de llamada.
 4. El número de intentos de apertura de sesión puede controlarse desactivando la cuenta del usuario después de un cierto número de intentos sin éxito. El usuario puede posteriormente reinstaurar la cuenta identificándose ante el director o el administrador del sistema. El peligro de esta técnica es la potencial denegación de servicio.

Validación basada en artefactos u Objetos:

- Los artefactos que se usan comúnmente incluyen bandas legibles por máquina y varias variantes de las tarjetas electrónicas inteligentes. Estos pueden instalarse en o cerca de las terminales.
- En muchos sistemas la validación por artefacto va acompañada por una contraseña. Esto es habitual en los cajeros automáticos.

- **Ventajas:**

Reducen la tasa de falsas aceptaciones.

- **Desventajas:**

Añaden un costo adicional.

Técnicas biométricas:

- Se basa en características específicas que posee cada usuario.
- Hay dos categorías básicas:
 1. **Características fisiológicas:** tales como huellas, patrones capilares, diámetro de la retina, color, geometría de la mano, características faciales, peso, etc..
 2. **Características de comportamiento:** tales como dinámica de la firma, patrón de voz, temporización de pulsaciones de tecla, etc..
- **Ventajas:**

Los dispositivos de detección son generalmente autocontenidos e independientes de sistema informático. Aumenta su resistencia a métodos de penetración informática y mejora la detección de falsificaciones.

Aumento en la precisión de la validación del usuario.

Reducción de las falsas aceptaciones.

- **Desventajas:**

Las características de comportamiento pueden variar con el estado del usuario y pueden ser susceptibles de producir tasas más altas de falsas aceptaciones o rechazos.

Aumento del costo.

Potencial invasión de privacidad.

Rechazo por parte de algunos usuarios.

8.8.2 Los problemas de la identidad: sus puntos débiles

- * **Algo característico de la persona**

- **Huellas dactilares:** La identificación se dificulta con quemaduras, suciedad, lectores sucios con polvo del ambiente, etc.

- **Patrones de voz:** Se modifican con los resfríos, disfonías, etc.

- **Fotografías:** los cambios físicos como ser: corte de cabello, barbas y bigotes, pinturas, operaciones estéticas, etc. producen problemas.

- **Firmas:** los cambios en la traza, apuros que modifican las velocidades de la traza, etc.

- * **Algo que posee la persona**

- **Credenciales o Tarjetas de identificación:** Las perdidas, alteraciones, robos, etc. producen serios problemas.

- **Cables y llaves:** perdidas, alteraciones, etc.

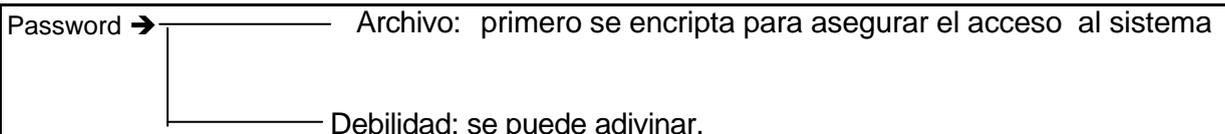
- * **Algo que sabe la persona**

- **Contraseñas (Password):** los problemas mencionados de longitud vs. Memorización, elección de la contraseña (fácil de recordar)

- **Accesos:** Los intentos repetidos de accesos para lograr violar la seguridad, se utiliza la restricción en un número determinado de accesos.

- **Acceso a archivos de contraseña del sistema operativo:** se evita con el encriptado y se asegura la probabilidad de dos problemas de violación: el conocimiento de la password y la clave de encriptado o el método y la clave.

Seguridad de la perspectiva del usuario: Contraseñas:



El tipear la contraseña sobre el teclado y luego transferirlo a memoria central para su encriptamiento puede ocasionar la captura de lo digitado en el módulo de entrada - salida, antes de ser transformada y de esta forma ser desviada a un archivo oculto en forma desenscriptada. Esto se resuelve colocando un circuito encriptador sobre el teclado.

Reglas para un buen password

- Debe tener seis caracteres mínimos (uno no alfabético)
- Debe cambiarse una vez cada tres meses por lo menos.
- Debe ser diferente en cada sistema o cuenta
- Debe ser generado al azar en lo posible (dificultad: recordarlo)
- No debe ser una palabra de diccionario.
- No debe ser un apodo de un familiar, mascota, número telefónico, o algo que se puede obtener con facilidad.
- No debe ser almacenado en ningún soporte ni usar teclas de función para alargar el password
- No debe dejarse ver (surfing)
- No tipearlo frente a otros

En el sistema, se deben controlar los accesos:

- Permiso de accesos a archivos

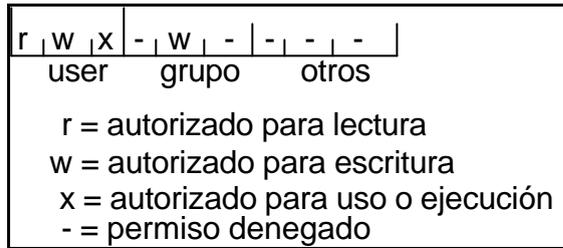


Fig. 8.13 Permisos en UNIX

Identificación en acceso por huellas dactilares digitalizadas.

Se basa en imágenes “scanneadas” de la huella dactilar de un usuario que se autoriza a acceder, ya sea a un área restringida o a un sistema como se observa en las figuras 8.14 a 8.16.

Este sistema se integra con otro sistema basado en una clave numérica llamada PIN (personal identification number = número de identificación personal) que incluso puede ser ampliada con otros sistemas, como ser, un registro de firma en un solo periférico con las tres funciones: huella dactilar, PIN y firma.

La identificación por las huellas dactilares se basa en un patrón registrado en una tarjeta o en memoria central y es comparada, en el instante que el usuario intenta acceder al sistema, con su huella personal. En el caso de tener las huellas almacenadas en una tarjeta procede así:

- 1) Introduce la tarjeta en un lector (de tarjeta) y coloca su dedo sobre un scanner que lee la imagen de la huella dactilar y la envía al procesador.
- 2) El procesador compara ambas imágenes y si concuerda autoriza el acceso, caso contrario almacena la huella digital del scanner como evidencia de que una persona ha intentado un acceso indebido utilizando la tarjeta de otro usuario.

En el caso de tenerlo almacenado en el sistema, el primer paso se carga en memoria y se compara con lo obtenido del segundo paso.

Primer Método:

El primer método consiste en dos pasos basado en la construcción de un registro de la huella que puede ser soportada sobre una tarjeta y el segundo en un archivo en el disco:

Primer paso:

Ejemplo: construcción de la huella dactilar como patrón en una tarjeta.

Para eso se produce de acuerdo a la siguiente figura:

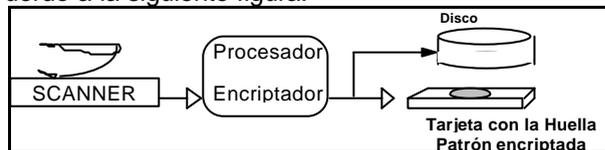


Fig. 8.14 Construcción de la huella patrón

segundo paso: uso de la identificación:

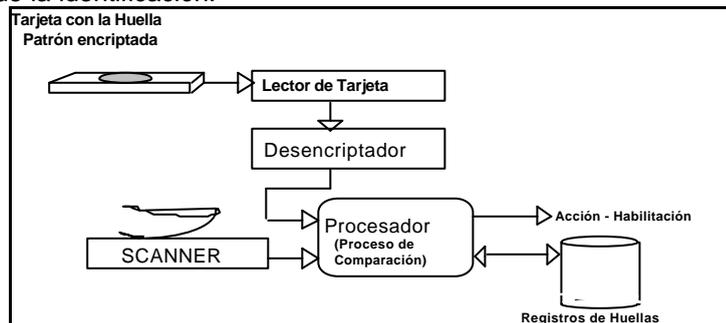


Fig. 8.15 Proceso de identificación mediante la huella patrón

Método dos:

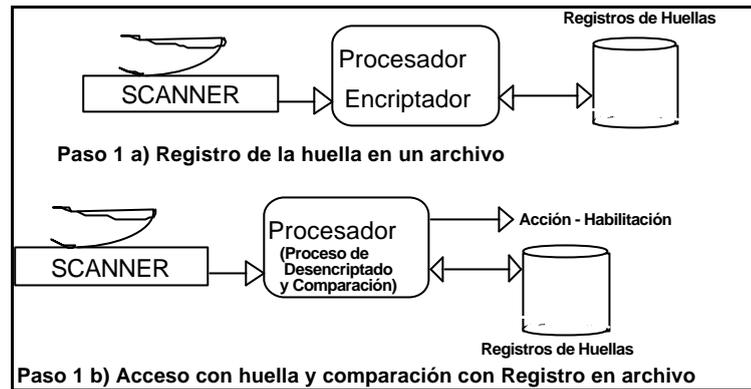


Fig. 8.16 Proceso de Scanner para identificación

8.9. GUSANOS Y VIRUS

8.9.1. Concepto de Virus

- Los virus informáticos, realmente no son propiamente programas sino un segmento de código, un trozo de programa, que se intercala dentro de otros programas que puede tomar el control del sistema operativo del computador o no, pero son capaces de programarse creando duplicados de sí mismos e intercalarse dentro de otros programas que se ponen a su alcance.
- Los virus llegan al computador dentro de un programa contaminado contenido en algún disco flexible. A continuación se instalan infectando el sistema y modifican todo programa que se utiliza a partir de entonces. Una vez pasado el tiempo durante el cual ha estado reproduciéndose, conocido como tiempo de latencia del virus, o bien cuando se cumple una determinada fecha, el virus se activa y comienza la acción destructora.
- Los tipos de acciones destructoras son diversas: desde el virus que borra todos los archivos, a los que hacen aparecer objetos por pantalla, pasando por los que realizan acciones que disminuyen la velocidad de ejecución de los programas.
- La contaminación, entonces, se produce cuando se copia un programa contaminado y se lo ejecuta. A partir de ese momento, el virus hace una copia de sí mismo en algún lugar del disco que contenga algún programa ejecutable.
- La existencia de un virus informático recorre cuatro etapas:
 - 1) Letargo.
 - 2) Propagación.
 - 3) Activación.
 - 4) Daño.

Tipos de contaminación:

- Existen 3 tipos básicos de contaminación:
 - 1) Virus que contaminan el sector 0 (partition table) o el Boot sector del disco.
 - 2) Virus que contaminan el sistema operativo (COMMAND.COM en el MS-DOS)
 - 3) Virus que contaminan los archivos ejecutables (.EXE o .COM)
- Los que contaminan el sector 0 o el Boot-sector (sector 1), sustituyen el programa de carga del DOS existente en dicho sector por un programa de carga del virus. El programa del virus, así como el programa de carga del DOS original, se localizan en sectores libres del disco que son posteriormente marcados en la FAT como sectores defectuosos para que no puedan ser detectados fácilmente por el usuario.
- Al arrancarse el computador, este lee el sector cero y uno del disco, el programa contenido en estos sectores localiza donde se encuentra el programa del virus, lo carga y posteriormente va al sector donde se guarda el programa de carga del DOS para terminar de poner en marcha el computador.
- Los virus que contaminan los archivos .COM (Ejemplo COMMAND.COM u otros) añaden a estos archivos de programa del virus al principio o al fin del mismo y sustituyen los 3 primeros bytes del archivo por la llamada al código del virus. El inconveniente que tienen este tipo de virus es que no pueden ser muy complicados, ya que un archivo .COM no puede tener más de 64 Kbytes. Sin embargo, muchos virus lo hacen es ocultar el grueso del código en un archivo oculto y utilizar el archivo .COM para hacer una llamada a este.

- Los virus que contaminan los archivos .EXE son los más difíciles de realizar y también de detectar, ya que estos archivos tienen una cabecera que necesita el DOS para el cálculo de las direcciones correctas de algunas instrucciones, así como los datos para localizar la primera instrucción de un programa. Los códigos del virus se colocan a continuación del código del programa y realizan en la cabecera los cambios necesarios para que la primera instrucción que se ejecute sea la del código del virus. A continuación da paso al código del programa.

Medidas básicas:

Después de lo que hemos analizado acerca de los virus, podemos afirmar que aunque no se pueda proteger al computador al 100% del posible ataque de un virus, si se puede reducir al mínimo la probabilidad de contagio si se toman las siguientes medidas:

- Utilizar Programas originales
- Proteger los discos flexibles contra escritura.
- Utilizar el antivirus antes de cargar o copiar cualquier programa externo.
- Efectuar copias de seguridad (Back-up)

Programas antivirus

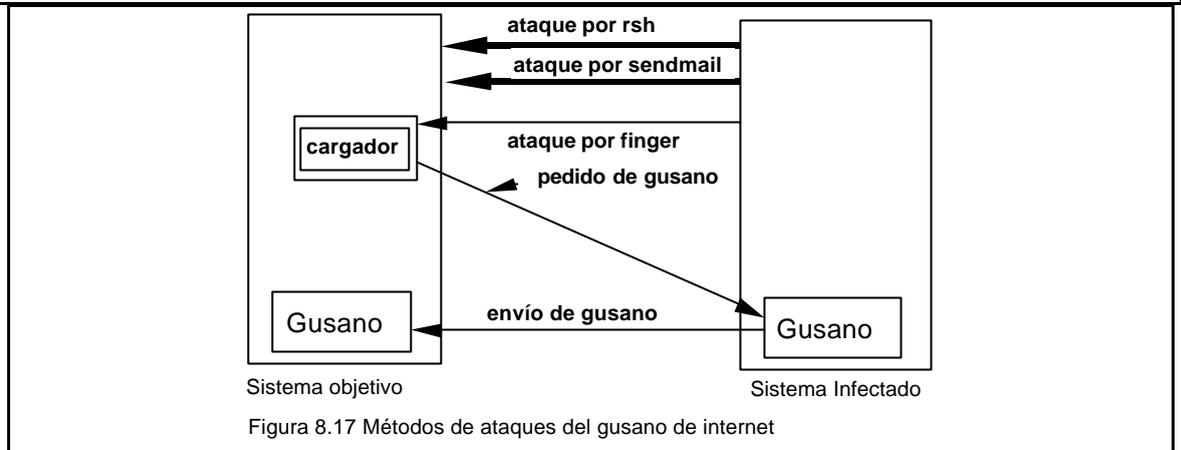
- La principal defensa que tienen los usuarios contra los virus es la precaución en utilizar programas originales y, en todo caso, utilizar programas antivirus que detecten, eliminen y protejan el computador del contagio.
- Existen 4 tipos de programas antivirus:
 1. **Detectores**, los cuales detectan la existencia de un virus conocido, alertando al usuario para que tome las medidas adecuadas.
 2. **Reparadores**, los cuales no solamente detectan la existencia del virus, sino que lo eliminan dejando los programas en su estado original.
 3. **Protectores**, los cuales se instalan en el computador permaneciendo residentes y vigilando las operaciones de los programas para impedir que el computador sea contagiado por algún virus, dando un mensaje de alarma en caso de que se detecte la presencia de alguno.
 4. **De vacunación**, los cuales añaden un código a los archivos ejecutables de modo que este se autorevisa al ejecutarse para impedir ser atacado por un virus.
- No hay una “vacuna” general conocida para los virus informáticos. **La prevención es la mejor protección.** Las medidas preventivas comunes incluyen el filtrado preventivo de todo software de reciente adquisición, las copias de respaldo frecuentes y una combinación de utilidades de las recién mencionadas.
- Cada uno de estos programas antivirus tiene fortalezas y debilidades. Su combinación junto como la filtración y adquisición cuidadosa de software sólo de fuentes reputadas puede conseguir una protección contra virus bastante buena. Las copias de respaldo frecuentes son importantes para ayudar a prevenir la pérdida de trabajos e información y ayudar a restaurar un estado sano del sistema después de una infección causada por algún virus.

Acciones en caso de tener un virus:

- Si a pesar de las medidas de seguridad adoptadas, se detecta la existencia de un virus en el computador, se debe eliminarlo.
- En caso de haberse perdido información, para recuperarla se debe utilizar las copias de seguridad o algún programa comercial utilitario como Pctools o Norton.
- En cualquier caso los pasos a seguir son los siguientes:
 1. Apagar el computador con el interruptor. No basta con reinicializar el sistema con las teclas CTRL+ALT+DEL en el caso de una P.C..
 2. Introducir en la unidad A: un disco booteable con el sistema operativo. Se debe estar seguro de que el disco no contiene virus y de que está protegido contra escritura.
 3. Encender de nuevo el computador, asegurándose de que se carga el Sistema Operativo desde la unidad A:
 4. Luego utilizar un programa detector de virus para identificar los archivos contagiados y el tipo de virus de que se trata.
 5. Utilizar un programa antivirus para eliminar el virus de los archivos contagiados.
 6. Hacer una comprobación para asegurar de que se consiguió eliminar el virus.
 7. Si después de lo anterior algunos archivos continúan contagiados, se debe borrarlos completamente. A continuación se intenta restaurarlos desde los discos flexibles o de las copias de resguardo. Si no se disponía de copias de ellos, se lo da por perdidos.

En caso de que no se disponga de un programa antivirus, o de que los programas antivirus disponibles no sean los adecuados entonces solamente queda la trágica solución de formatear el disco rígido, previa salvaguarda de los archivos no ejecutables.

8.9.2. El Gusano de Internet



Existían **3 métodos** para intentar infectar las nuevas máquinas como se ve en la figura 8.17.

Mediante la detección de 2 errores en el UNIX, Morris consiguió un acceso no autorizado a máquinas conectadas a la red Internet y mediante un programa llamado gusano, que se reproducía a sí mismo accedía a todas las máquinas, infectó miles de miles de ellas.

Desde el punto de vista técnico el gusano constaba de 2 programas, el "arrancador" y el gusano propiamente dicho. El arrancador era un archivo de 99 líneas en C llamado l1.c. Se compilaba y ejecutaba en el sistema por atacar. Una vez ejecutado, se conectaba de la máquina de la cual provenía, cargaba el gusano principal y lo ejecutaba. Después de ciertos problemas para ocultar su existencia, el gusano buscaba en las tablas de rutas del nuevo anfitrión para saber las máquinas que estaban conectadas con este e intentaba diseminar el arrancador a estas máquinas.

El gusano infectó sistemas SUN y VAX.

El método 1 intentaba ejecutar un Shell remoto mediante el comando rsh. Algunas máquinas confían en otras, por lo que podrían ejecutar rsh sin autenticación alguna. Si esto funcionaba, el Shell remoto cargaba el programa del gusano y seguía infectando las demás máquinas a partir de ese punto.

El método 2 utilizaba un programa presente en todos los sistemas BSD, llamado finger, el cual permite a cualquier usuario de Internet escribir y poder exhibir la información relativa a una persona en una instalación dada. Esta información incluye por lo general el nombre real de la persona, clave de acceso, direcciones de su casa y su oficina, nombre de la secretaria y número telefónico, número de Fax, etc. (un montón de datos)....

Finger funciona de la siguiente manera: En cada instalación BSD, un proceso secundario llamado **finger daemon** se ejecutaba todo el tiempo con el fin de enviar y contestar las solicitudes de todas las partes de Internet. Lo que hizo el gusano fue llamar a finger con una cadena especialmente diseñada de 536 bytes como parámetro. Esta cadena sobrepasaba la capacidad del buffer de Daemon y escribía sobre su pila. El error que se utilizaba aquí era el hecho que Daemon *no verificaba* si se excedía la capacidad del buffer (¿Ustedes verifican?...).

Cuando Daemon regresaba del procedimiento en el cual se encontraba cuando recibió la solicitud, no regresaba al main, sino al procedimiento dentro de la cadena de 536 bytes de la pila. Este procedimiento intentaba ejecutar /bin/sh. Si lo lograba, el gusano disponía de un Shell y entonces se ejecutaba en la máquina atacada.

El método 3 dependía de un error en el sistema de correo, Sendmail, que permitía al gusano enviar una copia del arrancador además de la ejecución de éste.

Una vez establecido, el gusano intentaba romper las contraseñas del usuario. Cada contraseña rota permitía que el gusano penetrara en otras máquinas donde el poseedor de la contraseña tuviera cuentas.

Cada vez que el gusano lograba acceso a una nueva máquina, verificaba si en ella existían copias activas del mismo. En tal caso, la nueva copia salía, excepto una vez por cada siete que entraba a la máquina, tal vez en un intento de mantener al gusano en propagación. El uso de 1 por cada 7 creó un número enorme de gusanos y fue la razón por la cual las máquinas infectadas fueron obligadas a parar: *estaban totalmente infectadas por gusanos*.

Propagación y consecuencias del gusano de internet:

- ❖ Propagación y Consecuencias de la presencia de un gusano de internet en un sistema:
 - Los programas gusanos informáticos no causan generalmente ningún daño directo a los otros programas y archivos del sistema que invaden.
 - Tienen a consumir grandes cantidades de recursos para su propia diseminación, denegando servicio a los usuarios legítimos.
 - Operan enviando copias de sí mismo a otras máquinas en una red.
 - La línea habitual de ataque son las listas de *correo de usuarios* que contienen los nombres y direcciones de otras máquinas alcanzar. Al obtener acceso a una lista de correo, envía copias de sí mismo (un programa ejecutable) alguna o todas las máquinas listadas.
 - El gusano puede o no comprobar si el nuevo destino está ya infectado. Si no está infectado, o si no realiza la comprobación, el gusano intentará obtener acceso a las listas de correo local para seguir propagándose.
 - La infección se extiende generalmente con rapidez y ahoga al sistema entero al consumir el tiempo de proceso y el ancho de banda de la red de tal modo que prácticamente no puede realizarse ningún otro trabajo ni comunicación.
 - Causas comunes de fallos y bloqueo en tales circunstancias incluyen alcanzar el límite del tamaño de la lista de procesos de tal modo que no pueda crearse ningún proceso más, el desbordamiento de los buffers de red y la inundación de los nodos de comunicación.
- ❖ Dos importantes salvaguardas frente a gusanos de Internet:
 1. *Hacer que la penetración de cada máquina sea difícil.* Los intentos de penetración pueden ser detectados reforzando la seguridad del sistema, especialmente en las áreas de utilidades de correo, contraseñas y control de acceso a las listas de correo.
 2. *Estorbar la propagación de los gusanos.* La diseminación de programas ejecutables puede ser restringida por medio de puntos de comprobación en el sistema de comunicación.

8.9.3. Esteganografía:

La esteganografía o empleo de canales subliminales consiste en ocultar en el interior de una información, aparentemente inocua, otro tipo de información (cifrada o no). Este método ha cobrado bastante importancia últimamente debido a que permite burlar diferentes sistemas de control. Supongamos que alguien quiere enviar un mensaje fuera de su país, burlando la censura. Si lo codifica, las autoridades jamás permitirán que el mensaje atraviese las fronteras independientemente de que puedan acceder a su contenido, mientras que si ese mismo mensaje viaja camuflado en el interior de una imagen digital para una inocente felicitación navideña, tendría muchas más posibilidades de llegar a su destino. Como es de suponer, existen tantos mecanismos para llevar a cabo este camuflaje como nuestra imaginación nos permita.

Mención especial merece el uso de la esteganografía para exportar información sin violar las leyes restrictivas que, con respecto a la Criptografía fuerte, existen en algunos países. El mensaje se envía como texto plano, pero entremezclado con cantidades ingentes de basura.

El destinatario empleará técnicas esteganograficas para separar la información útil del resto.

Esta técnica se conoce como *chang and winnowing*, que vendría a traducirse como llenar de paja y separar el grano de la paja. En consecuencia, tenemos un mecanismo para transmitir información no cifrada, pero que sólo puede ser reconstruida por el destinatario, con lo que en realidad hemos logrado protegerla sin usar en ningún momento ningún algoritmo de codificación. Este sistema surgió en marzo de 1998, propuesto por Ronald L. Rivest [uno de los creadores de RSA], como desafío a la política restrictiva del Gobierno de los EE.UU. con respecto a la Criptografía.

8.10. MODELOS FORMALES DE PROTECCIÓN

- Los modelos formales de protección proporcionan un aparato para la formulación precisa y para el razonamiento respecto a la corrección y complejidad de las diferentes políticas y mecanismos de seguridad.
- Los dos primeros modelos, la matriz de accesos y Tomar-Conceder, formalizan modelos de protección basados en control de acceso discrecional (**CAD**). La principal preocupación es el ataque de caballo de Troya.

- Los otros dos modelos, Bell-Lapadula y el del retículo, se ocupan del control de flujo y del control de acceso obligatorio por lo que se presentan a continuación.

8.10.1. Modelo de Matriz de Control de Accesos

- Este modelo permite la demostración de varias propiedades globales de los sistemas de protección (fue presentado en el punto 8.6 y 8.7).
- El modelo es una formalización de la matriz de control de accesos.
- Su objetivo principal es resolver el problema de la *seguridad* del sistema, es decir, determinar si un sujeto puede obtener acceso a un objeto dado.
- El sistema de protección se modela como un conjunto de *sujetos* S cuyos derechos de acceso al conjunto de *objetos* O se expresa mediante una *matriz de accesos* A. Los sujetos son las entidades activas del modelo. Los sujetos también se consideran objetos. Los objetos son las entidades protegidas del sistema.

8.10.2. Modelo Tomar-Conceder

- Es un modelo basado en grafos que describe una clase restringida de sistemas de protección. Como ventaja, la seguridad de los sistemas Tomar-Conceder es decisivo incluso si es posible crear un número de sujetos y objetos ilimitados. Utilizando reglas de transformación de grafos, la seguridad del sistema de protección es decisivo en un tiempo que crece linealmente con el tamaño del grafo de protección inicial.
- Considera al sistema de protección formado por:
 - 1) un conjunto de sujetos (los cuales no son objetos),
 - 2) un conjunto de objetos,
 - 3) un conjunto de derechos genéricos (leer, escribir, ejecutar, tomar, conceder).
- Describe la transferencia de derechos de acceso en sistemas.
- Captura únicamente los estados relativos a la seguridad, el modelo simplifica así la determinación de la seguridad.
- Describe muchos aspectos de sistemas existentes, especialmente los basados en capacidades. Sin embargo no pretende representar ningún sistema en particular.
- La principal importancia está en la demostración de que las decisiones de seguridad pueden ser relativamente sencillas es sistemas un tanto restringidos.

8.10.3. Modelo Bell-Lapadula

- Bell y Lapadula han ideado un modelo de protección que combina el modelo de matriz de acceso con la jerarquía de ordenación.
- El sistema de protección se visualiza como un conjunto de sujetos, un conjunto de objetos y una matriz de accesos.
- Cada entrada de la matriz puede contener un conjunto de derechos de acceso genéricos que registran derechos de accesos discrecionales, tales como leer, escribir y ejecutar. Cada sujeto tiene asignada una autorización, y cada objeto un nivel de autoridad.
- El modelo define un conjunto de operaciones primitivas.
- El mecanismo de flujo de información implementa las siguientes propiedades:
Seguridad simple: requiere que un sujeto pueda leer únicamente de objetos cuya clasificación de seguridad sea igual o inferior a la de su propia autorización.
La **propiedad** exige que un sujeto s puede tener acceso de escritura sólo a los objetos o cuyo nivel de clasificación sea igual o superior al de su propia autorización.
- Este modelo proporciona los medios para evitar flujos de información no autorizados en sistemas basados en él.
- El modelo es incapaz de especificar algunas políticas de protección útiles para casos en donde el flujo de información requiere ordenación no lineal.

8.10.4. Modelo Retículo de Flujo de Información

- Visualiza el sistema de protección como un conjunto de sujetos, objetos y clases de seguridad.
- Se consideran asignaciones de seguridad estática y dinámicamente variables.
- Proporciona una abstracción que puede ser utilizada para expresar una variedad de políticas de seguridad práctica.

- Proporciona fundamento para la verificación automática del flujo de información en tiempo de compilación, de ejecución o en ambos.

La seguridad del sistema se fuerza asegurando que todos los flujos de información estén autorizados.

8.11. SISTEMAS DE PROTECCIÓN Y SEGURIDAD EN ARCHIVOS.

- **El uso compartido y los problemas de seguridad**
- **Especificar** que **privilegios** de acceso deben tener los otros usuarios.
- **Asociar** con cada Archivo un conjunto de privilegio de acceso.

Clasificar los Usuarios como Clases de user: El propietario, Sus socios o grupo y los otros como lo hace UNIX

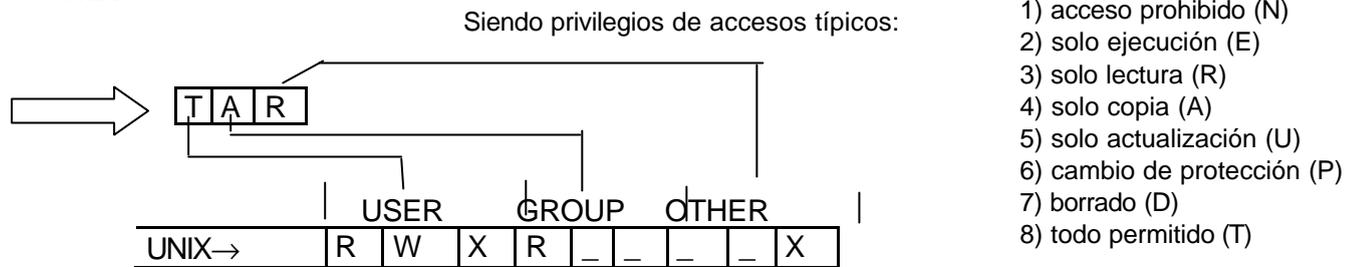


Figura 8.18 Métodos de Protección en UNIX

El Sistema de Gestión de Archivos visto en el módulo 7, debe permitir cambiar los privilegios.

El problema de seguridad ligada al uso compartido de los archivos cuanto más flexible sea el método adoptado, mayor espacio ocuparán en los User File Directory (UFD) y más tiempo se emplearán para llevar a cabo un acceso.

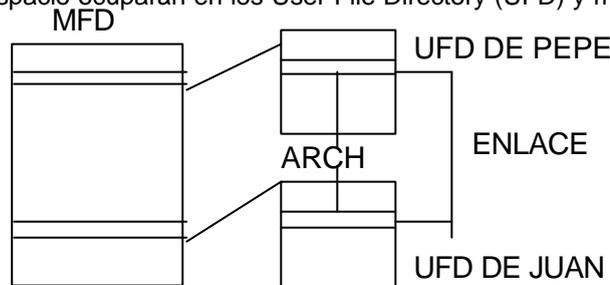


Figura 8. 19 enlaces entre MFD y UFD

Hay distintas soluciones:

- 1º) el propietario guarde en su UFD los usuarios a los cuales ha concedido privilegios y el tipo de privilegio.
- 2º) que el propietario permita a otro propietario crear enlaces en sus respectivos UFD.

PROBLEMA: Cuando se borra un Archivo se debe borrar el enlace. UNIX contador de enlace solo se borra cuando está en cero.

8.12. Normas y Procedimientos en un Sistema de Seguridad

Normas y procedimientos para riesgos de origen físico

Las prevenciones para catástrofes climáticas e incendios para un sistema informático son las mismas que se deben tomar desde el punto de vista de la seguridad edilicia. Deben estar en funcionamiento si el edificio en que se encuentra el sistema está correctamente habilitado, según las normas de urbanismo vigentes.

Normas y procedimientos para riesgos de origen humano

La forma más fácil de acotar los problemas ocasionados por las persona ajenas al sistema es restringir su acceso. En caso de ser necesario su ingreso (como por ejemplo el personal de limpieza) debe tratar de que el sistema esté desactivado (al menos parcialmente) y protegido ante cualquier tipo de acción dañina, o de no ser esto posible, se debe poner una persona que supervise las acciones que se realicen.

Normas y procedimientos para riesgos de origen técnico

Control de intrusos: partiendo de la base de que, debido a que las personas externas al sistema son controladas cuando ingresan al entorno, o ni tienen posibilidad de interactuar con los componentes si no

son controladas, los principales problemas de intrusión estarán reducidos a las personas que operan el sistema.

El procedimiento debe ser determinado y llevado adelante por lo responsables de la estrategia de seguridad.

Los responsables de la estrategia de seguridad informática deberán ajustar los privilegios de cada empleado en el programa que limita el control de acceso a la información del sistema de acuerdo al siguiente parámetro: "dar acceso a cada uno sólo a la información que necesita para trabajar". Como lo más probable es que los responsables de la estrategia de seguridad del sistema no hayan podido prever en forma exacta las necesidades de acceso a la información de cada persona, se tendrán que hacer ajustes sobre la marcha (que deberá ser correctamente estudiados y autorizados) hasta poner a punto el control de accesos.

Backup: al ser la parte más importante de toda estrategia de seguridad, es necesario crear una cadena de responsables que se encarguen de que el procedimiento de backup se cumpla siempre sin excepciones. El almacenamiento debe ser realizado a buen recaudo de los intrusos y protegido de las contingencias a las que esté expuesto el sistema.

Control de virus: como siempre, los responsables de el determinar mecanismo y llevarlo adelante son quienes se ocupan de la estrategia de seguridad.

8.12.1. Cómo llevar adelante la estrategia de seguridad

Una vez que se han determinado cuáles son los componentes del sistema, los riesgos a los que están expuestos, hasta dónde se deben minimizar y cuáles son las herramientas, normas y procedimientos con que se van a minimizar se procede a montar la estrategia de seguridad.

En este punto los pasos a seguir son los siguientes:

- Montar los sistemas antivirus y de control de acceso a la información durante un fin de semana y sin comunicarlo a los empleados. De esta manera se evita el riesgo de que una persona descontenta ingrese un virus o copie información al saber que luego no podrá acceder.
- Antes de comenzar a trabajar (luego de montados los mecanismos de seguridad) se le comunicará a los empleados, claramente y por escrito, cuáles son las normas y procedimientos a seguir y cuáles son los privilegios de acceso a la información de cada uno y cómo presentar pedidos de aumento de privilegios.
- Como la mayoría de los sistemas de administración de información funcionan a partir de passwords, también se deberá instruir a los empleados sobre su correcto uso. Puede llegar a exigirse que el empleado se haga responsable de la confidencialidad y buen uso de password bajo la pena de ser sancionado.
- Se debe dejar en claro que se parte de la base de que lo que no está claramente permitido en la documentación de la estrategia de seguridad está prohibido. Esto permite ir ajustando la estrategia a medida de que aparezcan necesidades nuevas.

8.12.2. Plan de contingencia

Una estrategia de seguridad puede fallar porque fue mal diseñada o porque actuó un factor de riesgo fuera del límite prefijado para su contención. Recordemos que no podemos manejarnos con un 100% de seguridad en cada uno de los elementos del sistema y que la estrategia de seguridad implica la toma de decisiones sobre el grado de inseguridad con el que puede convivir el sistema de acuerdo a la información que maneja y la misión que cumple.

Cuando la seguridad de un sistema cae se pasa a manejar el plan de contingencia de la estrategia de seguridad, compuesto también por una serie de normas y procedimientos.

Normas y Procedimientos del Plan de Contingencia

Catástrofe climática, incendio y hurto

La primera norma de contingencia con la que se debe contar ante estos riesgos es un seguro adecuado que facilite el volver a condiciones operativas.

En el caso de que la destrucción o hurto del sistema sea parcial se puede contar con una cantidad mínima de computadoras de respaldo para continuar operando mínimamente cuanto antes.

En éste como en todos los casos es fundamental la existencia del backup actualizado y que se haya conservado en un lugar a salvo de la contingencia.

Sabotaje

Si el sabotaje es de tipo físico, como por ejemplo vandalismo, caben las mismas consideraciones que en el punto anterior. Si el sabotaje es de tipo lógico sobre los programas y los datos, se deben recuperar inmediatamente las condiciones operativas desde el backup actualizado. En todos los casos corresponde una investigación interna que puede llegar a requerir intervención policial y judicial.

Virus informáticos

El objetivo también es recuperar rápidamente la operatividad.

Intrusión

Si la intrusión ha provocado algún tipo de daño lógico en programas o datos se aplican los criterios del punto de sabotaje lógico.

Si se detectó la intrusión pero no se está seguro si existió algún daño, se compare la información existente en el sistema con el backup actualizado y se analiza si las diferencias observadas responder al trabajo posterior a la actualización. Antes de seguir este procedimiento conviene revisar el sistema de virus o programas dañinos que podrían haber sido dejados por el intruso.

Se debe hacer una investigación interna y corregir la falla en la estrategia de seguridad (ésta puede no ser técnica sino, por ejemplo, consistir en la sanción a un empleado que facilitó la intrusión en forma intencional o por descuido).

Quién debe llevar adelante el plan de contingencias

El plan de contingencias debe ser llevado a cabo por los responsables del área de sistemas correctamente especificados. Debe existir una cadena de estos responsables para el caso de ausencias por vacaciones o enfermedad.

Todas las personas del sistema deben saber que sólo los responsables indicados deben actuar en una contingencia.

TIPOS DE PLAN DE CONTINGENCIA

Hay dos tipos de contingencia a ser dirigidos por cada Plan de Contingencia: Programático y Operativo.

El primer tipo haría necesario el uso de desvíos previos al Horizonte de Tiempo como para superar una falta percibida de tiempo para completar los correctivos.

El segundo tipo, también solicitado, haría necesario pasos para hacer frente desde el comienzo los problemas operacionales a algún Horizonte de Tiempo. Cada sección correspondiente al Plan Programático y Operativo, tiene un enfoque apropiado de gerenciar riesgos, y los dos se integran en un único Plan de Contingencia coherente.

CONSIDERACIONES ADICIONALES DEL PLAN DE CONTINGENCIA:

- El alcance de los planes debe establecerse claramente, incluyendo duración de contingencia esperada, para ambas contingencias Programática y Operativa.
- Deben definirse y establecerse las prioridades cuidadosamente y deben estar de acuerdo con todas las partes involucradas.
- Desde temprano, la estimación de los costos para preparar y llevar a cabo los planes se deben desarrollar y considerar posibles cambios.
- Deben establecerse las responsabilidades por desarrollar y mantener cada Plan de Contingencia, así como el período de tiempo entre las revisiones mayores.
- El mantenimiento de Plan de Contingencia es continuo y se rastreará a través del proceso de aprobación de cada organización.
- La existencia de un Plan de Contingencia aceptado es un requisito a ser considerado certificado en muchas organizaciones.
- Finalmente, debido a su importancia, los planes de contingencia deben fecharse, firmarse y promulgarse a un nivel alto dentro de la organización.
- También se deben considerar un plan de capacitación dentro de la institución.

PLANES DE CONTINGENCIA PROGRAMÁTICOS DE DETALLES GENERAL:

Pre contingencia (Fase de Planeamiento)

Esta fase cubriría los elementos siguientes:

1. Determinar las políticas y recursos para la seguridad (Alta Dirección).
2. Evaluar los riesgos potenciales. Listar los tipos de fallos, sus efectos, y sus probabilidades.
3. Evaluar la fiabilidad de la certificación de sistemas.
4. Desarrollar los Planes de Clasificación.
5. Desarrollar una matriz para los pasos y evaluar los impactos probables de fracasos del sistema
6. Identificar recursos críticos que pueden afectarse
7. Asignar prioridades operacionales por definiciones acordadas
8. Fijar Centro de Crisis
9. Considerar alternativas
10. Identificación de fechas de acción para las contingencias Programáticas y Operativo.
11. Evaluar los modos de fracaso potenciales de alternativas.
12. Establecer los acuerdos de apoyo como requerimientos de servicios (Bomberos, Médicos, ambulancias, electricistas, etc.). Equipos de comunicaciones (teléfono, radio) Apoyo en-sitio (personal adicional, apoyo del vendedor, equipo de reserva)
13. Identificar las maneras de conservar y proteger datos del sistema.
14. Establecer los métodos para descubrir y corregir datos corruptos.
15. Designar los roles, responsabilidades, y autoridad de: los gerentes, mantenedores, y diseñadores y como localizarlos
16. Identificar los procedimientos de emergencia a realizar durante la contingencia, fijar umbrales y límites.
17. Desarrollar procedimientos de identificación de contingencia para las contingencias Operativo
18. Considerar brechas de seguridad potenciales
19. Preparar un plan de Capacitación.
20. Establecer y realizar programa de entrenamiento
21. Identificar los procedimientos de recuperación de contingencia para las contingencias Programáticas y Operativo
22. Poner en práctica y probar el Plan de Contingencia Programático y operativo.
 - Tomar en cuenta la planificación de contingencia que esta siendo efectuada, la cual se interconecta con sus sistemas
 - Posibilidad de hacer "equipo" con otra organización con similar función o capacidad

DURANTE LA CONTINGENCIA (FASE DE EJECUCIÓN):

1. Monitoreo para los eventos desencadenadores de contingencia; serios y repetidos retrasos del horario
2. Verificar el sistema de correctivo y progreso de aplicaciones y juzgar resultados, para determinar los próximos pasos
3. Notificar a las autoridades apropiadas la naturaleza y alcance del problema.
4. Ejecutar los planes de contingencia, incluyendo la aplicación de principios de clasificación como sea requerido.
5. Ejecutar los planes de contingencia de riesgo específica como se requiera.
6. Modificar y mejorar los planes de contingencia como sea requerido, es decir, que sea hábil y flexible
7. Minimizar el daño al equipo.
8. Minimizar el daño a los datos y comunicaciones.
9. Documentar
 - Durante el evento, mantener registros cronológicos manuales, como se requiera
 - Supervisar la activación de riesgo de contingencia específica
 - Verificar las funciones del sistema y juzgar resultados, usados para determinar los próximos pasos
 - Notificar a las autoridades apropiadas la naturaleza y alcance de problema
 - Minimizar las amenazas a la vida y a la propiedad.

POST CONTINGENCIA (FASE DE RECUPERACIÓN)

Esta fase cubre de los siguientes ítems:

1. Ejecutar los planes de post-contingencia como sean requeridos.
2. Probar las funciones del sistema y revisar los resultados.
3. Restaurar y reiniciar los sistemas para la implementación como sea determinado en la prueba.
4. Notificar a las autoridades apropiadas la resolución del problema.
 - Verificar las funciones y resultados del sistema.
 - Corregir y restaurar los datos perdidos o viciados.
 - Revisar los procedimientos y actualizarlos.

Los Planes de Contingencia Operativo (operacional) son desarrollados por las organizaciones que los usan en consulta con los diseñadores de Tecnología de Información (TI), operadores y mantenedores, basado en la misión de las funciones críticas como sean elegidas por la organización.

Estos Planes los considerarían equipos TI y no TI (es decir, ítems de la infraestructura), tales como sistemas de misión y de apoyo de decisión, banco de datos y sistemas de comunicaciones, logísticas y sistemas financieros, el control del proceso de los sistemas industriales, calentamiento y enfriamiento, fuego y de seguridad, etc. Las consideraciones también incluirían planes para los ítems de infraestructura de la misión crítica, los sistemas y proveedores. Es muy importante la participación de todos en el diseño y revisión de los planes de contingencias.

Los planificadores aplicarían análisis en el Riesgo Operacional Gerencial para identificar los riesgos, impactos, probabilidades y procedimientos de la mitigación para las contingencias Operativo. Estar consciente de que los planes de contingencia (operacional) se ocupan de la continuidad y aplicación de misiones en las situaciones "en el peor caso". Finalmente, habría informes con respecto a los objetivos y la duración deseada del Plan.

8.13. RESUMEN

- La **seguridad** consiste en un conjunto de medidas y procedimientos para prevenir frente a robos, ataques, delitos, espionaje y sabotajes.
- El **objetivo de la seguridad informática** es mantener la integridad, disponibilidad y privacidad de la información confiada al sistema.
- **Amenazas** importantes a la seguridad incluyen la **revelación** no autorizada de información, la **alteración o destrucción** no autorizadas de la información, **el uso** no autorizado de servicios y la **denegación de servicio** a usuarios legítimos.
- Los **mecanismos de seguridad** especifican como realizarlas políticas de seguridad y proporcionan los medios para supuesta en acción.
- La **protección** se refiere a un mecanismo para controlar el acceso de programas, procesos o usuarios a los recursos definidos por un sistema.
- La protección es un problema interno, mientras que la seguridad debe considerar tanto el sistema de computación como el entorno (personas, edificios, actividades comerciales, objetos de vándalos y amenazas) donde se usa.
- Los sistemas de computación contiene muchos objetos, los cuales se deben proteger de la mala utilización. Los objetos pueden ser hardware o software.
- Un **derecho de acceso** es el permiso para efectuar una operación sobre un objeto.
- Un **dominio** es un conjunto de derechos de acceso.
- Los procesos se ejecutan en dominios y pueden usar cualquiera de los derechos de acceso en el dominio para acceder a los objetos y manipularlos.
- La **matriz de acceso** contempla al sistema como un conjunto de objetos, sujetos y dominios de protección que especifican grupos de objetivos y operaciones permitidas sobre ellos. .Está dispersa y normalmente se implanta como **listas de acceso** asociadas a cada objeto o como **listas de capacidades** asociadas a cada dominio. Se puede incluir la protección dinámica en el modelo de la matriz de acceso considerando como objeto a los dominios y a la propia matriz de acceso.
- Las **listas de accesos** se centran en los objetos y registran los derechos de acceso de todos los usuarios legítimos, correspondientes a las columnas en la representación de la matriz de acceso.
- Las **capacidades** registran los derechos de acceso completos de los sujetos individuales. Tratan con dominios o filas de la matriz de acceso.
- Los sistemas reales son muchos más limitados y tienden a ofrecer protección sólo para los archivos. UNIX es un caso representativo que para cada archivo proporciona protección de lectura, escritura y ejecución para el dueño, grupo y público en general. MULTICS utiliza una estructura anular además del acceso a archivos. Hydra y el sistema CAP de Cambridge son sistemas de capacidades que extienden la protección hasta los objetos definidos por los usuarios.
- Las **contraseñas** se emplean habitualmente para resolver el problema de la validación.
- La **criptografía** proporciona un método para aumentar la confianza en el secreto de la información en tránsito y en almacenamiento. Los sistemas criptográficos descansan en el secreto de la clave, no del algoritmo.
- Los **gusanos informáticos** son programas ejecutables autocontenidos que se diseminan a través de redes informáticas. Pueden provocar infecciones que consumen cantidades enormes de recursos y niegan servicio a los usuarios legítimos.

- Los **virus informáticos** se adhieren a otros programas y después de un período de letargo y propagación, provocan algún tipo de daño en la máquina anfitrión.

8.14. Bibliografía recomendada para este módulo

1. Distributed System, edited by Sape Mullander. (Second Edition),; Addison Wesley, 1994, 601 pages.
2. Operating Systems. (Second Edition), Stallings Willams; Prentice Hall, Englewood Cliff, NJ. 1995, 702 pages.
3. Operating Systems Concepts (Fifth Edition), Silberschatz, A. and Galvin P. B; Addison Wesley 1998, 850 Pages
- 4.. Operating Systems Concepts and Design (Second Edition), Milenkovic, Millan; Mc Graw Hill 1992
5. Modern Operating Systems, Tanenbaum, Andrew S.; Prentice - Hall International 1992, 720 pag
- 6.. Sistemas Operativos Conceptos y diseños.(Segunda Edición); Milenkovic Milan; Mc Graw Hill; 1994. 827 páginas
7. Operating Systems Review (ACM Press), Volumen 28 Número 1 Enero 1994
8. ACM Transactions on Computer Systems, Volumen 7 Número 3 Agosto 1989
8. Security in distributed computing: Did you Lock the door, Bruce Gleen, Dempsey Rob, Prentice Hall, Englewood Cliff, NJ. 1996, 464 pages.
10. Network Security:Private communication and Public Work, Kaufman/ Perlman/ Speciner, Prentice Hall, Englewood Cliff, NJ. 1995, 640 pages.
11. Network an internetwork Security, Stallings Willams; Prentice Hall, Englewood Cliff, NJ. 1996, 480 pages.

Bibliografía Complementaria recomendada sobre Seguridad y Protección

MILLEN J.

- SECURITY KERNEL VALIDATION IN PRACTICE
COMMUNICATION OF THE ACM VOL 19 NUM 5 MAYO 1976

SCHROEDER M.D.; CLARK D.D.;SALTZERT J.H.

- THE MULTICS KERNEL DESIGN PROYECT, OPERATING SYSTEM REVIEW VOL 11,NUM 5 1977
PAG.43 A 56
- FINAL REPORT OF THE MULTICS KERNEL DESING PROYECT, MIT - LCS - TR - 96. (1977).

POPEK, G.J. KLINE C.S.

- ISSUES IN KERNEL DESIGN
AFIP CONFERENCE PROCEEDINGS 1978 NCC VOL 47 PAG.1079-1086.

MC COULEY E.J.

- KSOS: THE DESIGN OF A SECURE OPERATING SYSTEM
AFIP CONFERENCE PROCEEDINGS 1978 VOL 48 -1979- PAG.345-353

WALKER B., KLEMMERER R.A., POPEK G.J.

- SPECIFICATION AND VERIFICATION OF THE UCLA UNIX SECURITY KERNEL
COMMUNICATION OF THE ACM VOL 23 Nº 2 FEBRERO 1980.

AMES S.R. HIJO Y GASSER M.

- SECURITY KERNEL DESIGN AND IMPLEMENTATION: AN INTRODUCCION
COMPUTER VOL 16, Nº 7 JULIO 1983 PAG.47-53.

SILVERMAN J.M.

- REFLECTIONS ON THE VERIFICATION OF THE SECURITY SYSTEM KERNEL
ACM VOL 17, Nº 5 OCT.83 PAG 143-154 (PROCEEDING OF THE SYMPOSIUM ON OPERATING SYSTEM PRINCIPLES.

GLOSARIO DE TÉRMINOS EN IDIOMA INGLÉS

| Write

| Lock

| Network security

| System call

Update	Set up	Data Encryption Standard	Pattern
Sendmail	Finger	Cluster	Offset
Sniffer	Send mail	Display	Manager
Relay	End to end	Any User	Stream
Read	Back-up	Computer security	Trap
Prompt	Traffing padding	Switchboard	Anonymous
Print	Rebout	Password	Rolled back
Partition table	Relay	Traffig Padding	Switchboard
Overhead	Drivers	Default	Backpointer
Open	Start	On Pad	redo
No update	Display	Capability	Check point
Lookup	Zoom	User	Surfing
Kernel	Scheduler	Hardware	commit
Insert	Random	Group	Command
Hackers	Break	Firmware	Execute
File System	Local Area Network	Daemon	Host
File	Link	Deadlock	Mainframe
Erase	Tokens	Owner	Personal Identification number
Create	Stream	Lock / Key	Scanner
Copy	Checksum	Overhead	Execution
Close	Abort	Fire wall	Write - ahead login
Caching	Random	Front end	Manager
Bugs	Checksume	Software	Switch
Administer	Drawbacks	Universe	Shell
UNIX	Usenet	Internet	Intranet
Archy	Gopher	Proxy	Telnet

American Standard Code for information interchange

GLOSARIO DE TÉRMINOS EN CASTELLANO

Seguridad	Protección
Causas de Pérdida de datos	Requerimiento de seguridad
Concepto de políticas de seguridad	Concepto de mecanismos de protección
Objetivos de protección	Seguridad a través del S.O.
Amenazas a la seguridad - Tipos	Caballo de Troya
Gusanos	Virus
Ataques a la seguridad	Monitoreo de las amenazas - Vigilancia
Objetivos de la seg. y protección de un sistema	Justificación de la seguridad y protección
Principio de los mecanismos	Tipo de seguridad
Supervisión y vigilancia	Supervisión de riesgos seguridad por el S.O.
Auditorías	Back - ups
Funciones de un sistema de protección en el S.O.	Seguridad en los datos
Seguridad en las bases de datos	Seguridad en las telecomunicaciones o redes
Métodos de ocultamiento de datos	Criptografía - sus problemas
Dominios de protección	Matriz de acceso - Implementación
Cambios de dominios	Revocación
Seguridad en el Kernel	Transacciones
Modelo de sistema de Seguridad	Recuperación basado en el archivo log
Check points	Autenticación del usuario - Problemas
Anfitrión	Anfitrión Bastión
Modelos formales de protección	Sistemas de Protección y Seguridad
Seguridad en Sistemas Distribuidos	Seguridad en red - Criptografía
Seguridad en multimedia.	Enrutador

ACRÓNIMOS USADOS EN ESTE MÓDULO

E/S	Entrada / Salida	I/O	Input / Output
CPU	Central Processing Unit	LAN	Local Area Network
S.O.	Sistema Operativo	internet	inter Network
RPC	Remote Procedure Call	CAD	Control de Acceso Discrecional
R/w/x	Read / Write/Execute	CAO	Control de Acceso Obligatorio
USR	User	DEL	DELeTe
SW	SoftWare	FCB	File Control Block

HW	HardWare	BBS	Bolletin Board System
vs	versus	ID	IDentificador
pdeves	Phisical devices	ldevs	Logical devices
cdevs	Constructor device	HTTP	Hiper Text Transfer Protocol
DES	Data Encryption Standard	NBS	National Bureau of Standard
IBM	International Busines Machine Company	RSA	Rives Shamir Adleman
SVC	SuperVisor Call	PIN	Personal Identification Number
FAT	File Allocation Table	PC	Personal Computer
rsh	remote shell	sh	shell
UFD	User File Directory	MFD	Master File System
FS	File System	U	Unclassified
SMTP	Simple Mail Tranfer Protocol	C++	Lenguaje C++
C	Confidential	S	Secret
TS	Top Secret	Seg.	Seguridad
FTP	File Transfer Protocol	www	Word Wide Web
Prot.	Protección	mod	modulo
FEP	Front End Processor	TIU	Trust Interface Unit
BBS	Boletin Board System	vs	Versus
ACC	Access Control Counter	KDC	Key Distribution Center
RTE	Real Time Environment	ASCII	American Standard Code for Information Interchange

Anexo 8.a. Diseño de un sistema de seguridad para redes

Las políticas de seguridad para redes

Antes de construir un firewall para conectar una red al resto de Internet, es importante conocer exactamente qué recursos y servicios de la red se quieren proteger. Una política de seguridad para redes es el documento que describe las consideraciones de seguridad de una organización. Este documento constituye el primer paso en la construcción de un firewall efectivo.

Planeamiento de la seguridad en la red

Es importante tener una política de seguridad bien definida y efectiva para proteger los recursos de información de la empresa. Muchos diseñadores de redes comienzan a implementar soluciones con firewalls antes de que un problema particular de política de seguridad haya sido debidamente identificado. Tal vez una de las razones de que esto sea así es que llevar adelante una política de seguridad efectiva significa hacer preguntas difíciles acerca de qué tipos de servicios de Internet y qué recursos tendrán permitido el acceso los usuarios, y cuáles tendrán acceso restringido por riesgo en la seguridad.

Si actualmente los usuarios disfrutan de un acceso irrestricto a la red, será difícil implementar una política de seguridad que restrinja el acceso.

Una política de seguridad para red efectiva será aquella en la que todos los usuarios y administradores de la misma puedan y quieran ponerla en práctica.

Los mecanismos de protección de una red: FIREWALLS

Desde que, en 1981, I.B.M. presentó la "Computadora Personal" (P.C.) para complacer a quienes deseaban sus propias computadoras, para correr sus propios programas y manejar sus propios archivos, separándose así de las minis y grandes computadoras que estaban bajo el estricto control de los Departamentos de Informática, ocurrió un fenómeno que I.B.M. no previó con su creación: el éxito y su masificación, y la interconexión mediante técnicas de comunicaciones.

A partir de ese momento comenzaron a enlazarse en redes para compartir los recursos que en el principio de su creación querían mantener separados (personales) en sus computadoras, avalando la teoría que dice que el hombre cuando obtiene lo que quiere extraña lo que tenía.

Con el transcurrir de los años dichas redes se fueron interconectando con otras aprovechando principalmente las experiencias del Ministerio de Defensa de los EE.UU. y de las Universidades, masificándose debido al bajo costo de dicho invento.

En la actualidad dicha explosión nos encuentra con grandes posibilidades de comunicación y de generación de negocios a través de INTERNET, la gran red de redes.

De la misma manera nos encontramos con una gran exposición a ambientes mucho menos controlables que el ambiente físico, en el que tenemos nuestra PC con nuestra más preciada información personal (dentro de nuestra casa), o donde tenemos nuestra red corporativa con nuestra más importante información comercial (la empresa).

Por lo que se deben tener en cuenta nuevas medidas de seguridad que se complementen con los tradicionales métodos de back-up, detección de virus informáticos, protección de soportes magnéticos y de listados para hacer frente a las nuevas situaciones de peligro.

Amenazas Potenciales de una Conexión a INTERNET o INTRANET

Existen distintos tipos de ataques que se pueden sufrir desde la exposición al ambiente de redes INTERNET o INTRANET, los mismos guardan una relación directa con el tipo de servicio que uno preste, la importancia en el mercado de la empresa, los enemigos que uno pueda tener, y a los servicios de la red a los que cada usuario pretenda acceder, entre otros, y dichos ataques suelen estar ligados con diferentes motivaciones y características que se detallan a continuación en tres categorías básicas: Intrusión, Negación del Servicio y Robo de Información.

1. Intrusión

Son los ataques más comunes que reciben los sistemas, la finalidad de éste tipo de ataques es el uso de la computadora atacada como si fuera propia del atacante.

Existen múltiples formas de obtener acceso, van desde ataques mediante **manipulación social** (encuentran la forma de saber el nombre de alguien con un puesto importante en la empresa y llaman a un administrador del sistema diciendo que son esa persona y necesitan cambiar su contraseña, en ese preciso instante, para hacer un trabajo urgente) hasta **adivinación** (prueban combinaciones con el nombre y contraseña de la cuenta hasta que una funcione, incluyendo o no el uso de programas para facilitar ésta práctica).

2. Negación del Servicio

Un ataque de negación del servicio está dirigido en su totalidad a evitar que el usuario atacado utilice sus propias computadoras.

Ejemplos de ésta práctica son:

- ✓ Inundación de mensajes a un servidor de correo,
- ✓ El Gusano de Internet,
- ✓ Re-enrutamiento de servicios hacia otro usuario (por ej. los llamados telefónicos en vez de llegarles al usuario original le llegan a otro, logrando de ésta manera incluso entorpecer el trabajo de dos blancos atacados),
- ✓ Bloqueo de una cuenta de cualquier usuario, que esté configurada de manera de inhabilitarse luego de un número de intentos fallidos, de manera sencilla al intentar iniciar varias veces la sesión.

Como vemos, es casi imposible evitar todos los ataques de negación del servicio, aunque por fortuna, no es una práctica habitual ya que es demasiado simple de realizar y muchas veces fácil de rastrear.

3. Robo de Información

El atacante, comúnmente llamado Espía o Delincuente Informático, intenta tener acceso para obtener información que luego pueda convertir en dinero (números de tarjetas de crédito, fórmulas, teléfonos o información para tener acceso a redes).

Las formas en las que suelen intentar el acceso a dichas redes son mediante:

- ✓ El uso de sniffers (analizadores), muy efectivos para encontrar claves interceptando los primeros paquetes que circulan por el cable en la red.
- ✓ La instalación de Caballos de Troya en busca del mismo objetivo.

Consecuencias de la Exposición a los Ataques

Las actividades descritas se traducen en la pérdida del secreto de la información, de la propiedad de la información, e inclusive si los fines del atacante son vandálicos o si intenta borrar las huellas de su accionar puede llegarse a la pérdida de la información.

En todos los casos el resultado se verá reflejado en pérdidas económicas, ya sea por, gastos generados por la necesidad de recuperar la información perdida, el uso por parte del atacante de servicios prestados por la empresa, la pérdida de confianza por parte del cliente en el servicio que la empresa presta si se hace pública la vulnerabilidad del sistema, o por la pérdida de posición frente a los competidores que aprovechan la información robada o simplemente el tiempo que permanece el atacado fuera del mercado recuperándose de las pérdidas sufridas.

Por todo lo anterior se debe tener en cuenta la necesidad de definir una Política de Seguridad para dar respuesta a tales ataques.

Limitaciones Legales

Respecto de la actividad que uno desarrolla puede que la falta de aplicación de ciertas normas mínimas de seguridad esté penada por la ley comercial, confiriéndole a la institución la responsabilidad legal de salvaguardar cierto tipo de información acerca de los empleados (historiales, domicilios, sueldos, etc.); O bien, puede tener obligaciones contractuales que lo obligue a proteger información de clientes o consumidores. Es altamente recomendable estar asesorado sobre este aspecto y tenerlo en cuenta cuando se diseña un sistema de seguridad para redes.

Conceptos que entran en la protección

Con los peligros enumerados estamos en condiciones de discutir cuan protegidos vamos a estar en el nuevo ambiente.

La presente no es una discusión simple ya que,

- ❖ Se debe lograr comprometer a todos los **usuarios** en la conclusión de la necesidad de tal protección puesto que la puesta en práctica de cualquier medida de seguridad lo tendrá como protagonista principal y le provocará ciertas “molestias” (como el uso de passwords) que deberá encarar con responsabilidad,
- ❖ Se debe lograr una presentación sólida y fuerte ante la **Gerencia** puesto que de los recursos económicos obtenidos dependerá también el nivel de protección alcanzado y por otra parte es necesario contar con el apoyo de la Gerencia para llevar a cabo posteriormente la implementación del proyecto que debe incluir la posibilidad de llevar a cabo las acciones

correctivas respecto del mal uso o incumplimiento de las pautas fijadas para el manejo de los respectivos sistemas y de las libertades propias de cada usuario.

Posibles Políticas de Seguridad

En relación con el resultado de la gestión del Administrador del Sistema respecto del punto anterior, los posibles modelos de seguridad son los siguientes:

I. Ninguna Seguridad

La medida más simple posible es no poner ningún esfuerzo en la seguridad y tener la mínima, cualquiera que sea, que proporcione el proveedor de manera preestablecida.

II. Seguridad a través de ser desconocido

Muchas personas piensan que por el hecho de ser “desconocido” o por ser una empresa pequeña nadie se va a tomar el trabajo de atacarlo.

De hecho, muchos de los intrusos no tienen interés en ningún blanco en particular, solo les interesa ingresar en la mayor cantidad de máquinas posible sin interesarles si son hogares, empresas pequeñas o grandes, y seguramente si intentan borrar su rastro a la hora de entrar puede ocasionar un grave daño.

III. Seguridad para Anfitrión (Host)

Este método consiste en la implementación de refuerzos en la seguridad de cada una de las máquinas que tienen contacto con el mundo exterior.

Es una solución compleja debido a que cada máquina según el fabricante y la versión del sistema operativo que corre tiene sus propios problemas de seguridad, y aunque se logre homogeneizar lo anterior, diferentes configuraciones (distintos servicios activados, etcétera) pueden llamar a diferentes subsistemas (y conflictos) que llevan a distintos problemas de seguridad.

Es un modelo que puede llegar a funcionar en sitios pequeños pero conforme aumenta el número de máquinas y usuarios se torna inmanejable.

De todas maneras se debe tener en cuenta como complemento de algún otro método que reúna a todos los anfitriones (PC's conectadas), por ejemplo, en las máquinas que tendrán contacto directo con Internet.

IV. Seguridad para Redes

Conforme los ambientes se tornan más grandes y diversos, y conforme se vuelve más difícil asegurarlos anfitrión por anfitrión surgen como opción principal las medidas de seguridad para redes, consistentes en la construcción de **Firewalls** para proteger sus sistemas y redes internas, utilizando estrictas medidas de autenticación (como las contraseñas de una sola vez o múltiples veces) y el uso de encriptación para proteger datos que son muy sensibles cuando transitan por la red.

La gran ventaja que tiene este modelo es la posibilidad de proteger con el mismo firewall, cientos, miles, o hasta cientos de miles de máquinas contra un ataque a las redes que están más allá del firewall.

Los servicios de INTERNET y los peligros asociados

Existen múltiples servicios a los que se puede tener acceso o brindar desde una conexión de Internet, los que se pueden resumir en:

1. Correo Electrónico

Es el servicio de red más popular y básico. Es de riesgo relativamente bajo, pero no invulnerable.

Los ataques más comunes que se pueden sufrir son:

i. Por falsificación de Correo Electrónico:

(a) Ataques contra su Reputación.

(b) Ataques de Manipulación Social (atacante envía correo electrónico a usuarios haciéndose pasar por el administrador aconsejándoles cambiar sus contraseñas de forma específica o por alguna autoridad).

i. Ataques de negación de Servicio.

ii. Correo incluyendo Caballos de Troya (en sistemas de correo multimedia modernos)

El Protocolo de Transferencia de Correo Simple (SMTP) es el protocolo estándar para enviar y recibir correo electrónico y su punto más débil radica en su servidor SMTP, quien tiene la capacidad de ejecutarse como cualquier usuario con amplios poderes lo que lo hace un blanco tentador para los atacantes.

El servidor SMTP más popular es el Sendmail de Unix, el que ha recibido un gran número de ataques (incluyendo el Gusano de Internet)

2. Transferencia de Archivos

El Correo Electrónico transfiere datos de un lugar a otro, pero está diseñado para archivos pequeños.

El protocolo más usado para la transferencia de archivos es el File Transfer Protocol (FTP).

En teoría, permitir que sus usuarios obtengan archivos no incrementa más el riesgo que permitir el correo electrónico. En la práctica, sin embargo, las personas realizarán más transferencias de archivos cuando FTP esté disponible, y es más probable que obtengan programas y datos indeseables, inclusive Caballos de Troya.

El FTP anónimo (anonymous FTP) es un mecanismo extremadamente popular para dar acceso a usuarios remotos a los archivos sin tener que darles un acceso completo a su máquina.

Si ejecuta un servidor FTP, puede permitir que los usuarios obtengan archivos colocados en un área pública separada de su sistema sin dejarlos iniciar una sesión y, potencialmente, tener acceso a todo su sistema.

Para muchas empresas, el establecimiento de un sitio FTP es el primer paso para realizar negocios por Internet, por lo que se debe protegerlo.

3. Acceso de Terminal Remota y ejecución de comandos

Los programas que proporcionan acceso de terminal remota permiten que utilice un sistema remoto como si su máquina fuera una terminal conectada directamente.

Telnet es el estándar para acceso de terminal remota en Internet.

Telnet se consideró en un tiempo un servicio más o menos seguro porque requiere que los usuarios se autentiquen por ellos mismos. Por desgracia, Telnet envía toda su información sin codificar, lo que lo hace muy vulnerable a ataques de espionaje (utilizando analizadores de protocolo) y es muy vulnerable desde el punto de vista del robo.

4. Noticias de Usenet

Los grupos de noticias (newsgroups) están diseñados para la comunicación de muchos a muchos, mediante suscripción a un grupo de interés particular.

Los riesgos de seguridad de las noticias son bastante bajos, lo único que hay que tener en cuenta es que generalmente se recibe un gran volumen de información, lo que no es un problema si está bien configurado el servicio.

5. World Wide Web

El correo, FTP, y las noticias de Usenet han existido desde los primeros días de Internet, en realidad, son extensiones de servicios proporcionados mucho antes de que existiera esa red. El World Wide Web (WWW) es un concepto, basado totalmente en Internet y, en parte en servicios existentes y en un protocolo: el Protocolo de Transferencia de Hipertexto (HTTP).

El WWW es una colección de servidores de HTTP y es el responsable, en gran medida, de la reciente explosión de actividad dentro de Internet.

Por desgracia, los navegadores Web y los servidores son difíciles de proteger.

La utilidad del Web se basa, en gran medida, en su flexibilidad, pero ésta dificulta su control.

Esta lista no son todos los servicios entre los que se debe decidir si pueden y deben ser instalados en cada uno de los anfitriones, pero son una muestra importante de tipos y peligros acarreados para la seguridad de la red interna.

Qué puede hacer y Qué no puede hacer un Firewall?

- ⇒ Es una fuente de decisiones de seguridad, un cuello de botella que permite concentrar sus medidas de seguridad en ese punto controlado de inspección.
- ⇒ Puede almacenar su relación con Internet de manera eficiente y proporcionar un buen lugar para reunir información sobre el uso de sistemas y redes ya que todo pasa por el firewall.
- ⇒ Limita la exposición evitando que los problemas de seguridad se propaguen por toda la red.
- ⇓ No puede protegerlo contra personas maliciosas internas, éstas amenazas requieren de medidas de seguridad internas, como seguridad para anfitrión y educación del usuario.
- ⇓ No puede protegerlo contra conexiones que no pasen por él.
- ⇓ No puede protegerlo contra amenazas antes desconocidas.
- ⇓ No puede protegerlo contra virus.

Estrategias de Seguridad para redes

De acuerdo a lo estudiado en puntos anteriores se recomiendan las siguientes estrategias:

1. Menor Privilegio

El principio de menor privilegio significa que cualquier objeto (usuario, administrador, programa, sistema, o lo que sea) debe tener sólo los privilegios que necesita para llevar a cabo sus tareas asignadas (no más). Recordar que por default, nada.

Si se entregan más permisos de los necesarios corre serios riesgos de seguridad y si por el contrario, se entregan menos permisos de los necesarios se ganarán enemigos entre los usuarios e intentaran evitar las medidas de seguridad para lograr sus objetivos.

2. Defensa por niveles

No se debe depender de un solo método de seguridad por más fuerte que éste parezca, se debe adoptar mecanismos que se den respaldo y redundancia entre sí para que el trabajo del atacante sea más costoso.

3. Accesos controlados

Se debe obligar a los atacantes a utilizar un canal angosto que se pueda monitorear y controlar. Es probable que parezca como poner todos los huevos en una canasta y, por lo tanto, mala idea, pero la clave es que se trata de una canasta que se puede proteger con sumo cuidado. La alternativa es dividir la atención entre muchos posibles frentes de ataque, más difícil de encarar.

4. Búsqueda del eslabón más débil

El eslabón más débil es donde se corta la cadena.

Se debe conocer los puntos débiles de la defensa para eliminarlos y para que pueda monitorear con cuidado lo que no pueda eliminar.

Es razonable, por ejemplo, preocuparse más por los atacantes a través de la red que por los que van a su sitio a atacarlo físicamente, por lo que se puede permitir que la seguridad física sea el eslabón más débil, lo que no implica olvidarse de la amenaza.

5. Postura de Falla Segura

Si un sistema de seguridad va a fallar debe hacerlo de tal forma que nieguen el acceso a un atacante en lugar de dejarlo entrar.

Si se descompone un enrutador para filtrado de paquetes, no debe dejar pasar ningún paquete. Si un programa proxy falla, no debe proporcionar servicio.

6. Participación universal aceptada

Para la efectividad de los sistemas de seguridad se requiere de la participación universal o por lo menos de la falta de oposición activa por parte del personal de un sitio.

Se necesita que se seleccionen buenas contraseñas y que todos notifiquen ocurrencias extrañas que pueden estar relacionadas con la seguridad.

7. Diversificación de Defensa

La idea que sirve de apoyo a la diversificación de defensa por niveles es que se puede requerir utilizar sistemas de seguridad de diferentes proveedores, lo que puede reducir las posibilidades de un problema o error de ataque pero puede generar nuevos problemas de configuración común que pueda comprometerlos a todos, o sea que, las mezcla de productos y de fabricantes dificulta la configuración.

8. Simplicidad

Mantener las cosas sencillas las hace más fáciles de comprender, lo que ayuda a saber si se tiene seguridad o no.

Los programas grandes y complejos suelen tener más problemas, lo cual evita distinguir entre problemas de seguridad y fallas del programa.

Definiciones y Componentes de un Firewall

Firewall: Un componente o conjunto de componentes que restringen el acceso entre una red interna protegida (intranet) y una red externa o Internet, o entre otros conjuntos de redes.

Anfitrión (Host): Un sistema de cómputo conectado a una red.

Anfitrión Bastión: Sistema de cómputo encargado de la conexión a Internet por lo que debe ser altamente seguro por su exposición a ataques.

Paquete: Unidad fundamental de comunicación en Internet.

Red de Perímetro: Es una red agregada entre una red de protección y una red externa a fin de proporcionar una capa adicional de seguridad.

Filtrado de paquetes: Acción de un dispositivo que controla el flujo de datos desde y hacia una red.

El filtrado puede darse en un enrutador, en un puente o en un anfitrión individual.

Los sistemas para filtrado de paquetes enrutan los paquetes entre anfitriones internos y externos, en forma selectiva, permitiendo o bloqueando ciertos tipos de paquetes.

Servidor Proxy: Programa que trata con servidores externos en nombre de clientes internos (representante).

Los servidores proxy proporcionan conexiones substitutas y actúan como compuerta a los servicios.

Los servicios proxy son efectivos cuando se emplean junto con un mecanismo que restringe las comunicaciones directas entre los anfitriones internos y externos.

Arquitecturas de Firewalls

1. Arquitectura de Anfitrión con Doble Acceso: Se construye alrededor de una computadora anfitrión que tiene por lo menos dos interfases de red. Tal anfitrión trabaja como enrutador entre las redes a las que están conectadas sus interfases; es capaz de enrutar paquetes IP de una red a otra. Sin embargo en tal arquitectura se suele desactivar ésta función de enrutamiento.

Así, los paquetes IP de una red (por ej. Internet) no se enrutan de forma directa a la otra red (la red interna protegida).

Los anfitriones con doble acceso pueden proporcionar un alto nivel de control.

El mismo permite que se rechacen conexiones que pretenden ser para un servicio específico pero que, en realidad, no contienen el tipo correcto de datos, (un sistema de filtrado de paquetes tiene dificultades con éste nivel de control).

Un anfitrión con doble acceso solo puede proporcionar servicios tipo proxy, o hacer que sus usuarios inicien una sesión directa con él y él es el intermediario.

Las cuentas de usuario, en dicha arquitectura, presentan problemas de seguridad importantes, donde pueden, inesperadamente, activar servicios que se consideran inseguros.

2. Arquitectura de Anfitrión de Protección: Se proporciona servicios en un anfitrión conectado sólo a la red interna, utilizando un enrutador independiente.

La seguridad principal la proporciona el filtrado de paquetes (por ej., el filtrado evita que las personas evadan los servidores proxy para hacer conexiones directas).

El anfitrión bastión se coloca en la red interna. El enrutador de protección está configurado de manera tal que el anfitrión bastión es el único sistema en la red interna con el que los anfitriones de Internet pueden abrir conexiones, siempre pasando por el enrutador de protección.

La ventaja en ésta arquitectura radica en que es más fácil defender un enrutador, que proporciona un número limitado de servicios, que defender un anfitrión.

Pero si un atacante logra penetrar el anfitrión bastión, no queda nada en la ruta de seguridad entre ese anfitrión y el resto de los anfitriones internos, situación mejor manejada en la próxima arquitectura.

El enrutador también presenta un solo punto de falla; si éste se halla en peligro, toda la red estará a merced del atacante.

3. Arquitectura de Subred de Protección: Se agrega respecto de la anterior arquitectura una red de perímetro que aísla más la red de Internet.

Hay dos enrutadores de protección, cada uno conectado a la red de perímetro, uno colocado entre ella y la red interna, y el otro entre ella y la red externa (por lo general, Internet).

Los servicios más vulnerables y menos confiables se colocan en las redes de perímetro exteriores, más lejos de la red interior.

Existen luego combinaciones de las arquitecturas expuestas y multiplicidad de posibilidades de acentuar la seguridad con las características de cada una de ellas, con mayores o menores posibilidades de éxito, como por ejemplo:

- ✓ Uso de múltiples Anfitriones Bastión (por rendimiento, redundancia, complementación y necesidad de separar datos o servidores).
- ✓ Fusión del enrutador exterior e interior en uno solo.
- ✓ Fusión del anfitrión bastión y el enrutador exterior (se complementen).
- ⇓ Fusión del anfitrión bastión y el enrutador interior. (se puede perder el respaldo que es el enrutador interior para el exterior y el anfitrión bastión).
- ⇓ Uso de múltiples enrutadores interiores (peligroso, configuración compleja).
- ✓ Uso de múltiples enrutadores exteriores.

- ✓ Configuración de múltiples redes de perímetro.
- ✓ Combinación de anfitriones con doble acceso y subredes de protección.

4. Red de Perímetro: La red de perímetro es otra capa de seguridad, una red adicional entre la red externa y la red interna protegida. Si un atacante entra con éxito a los límites externos del firewall, la red de perímetro ofrece una capa adicional de protección entre el atacante y los sistemas internos.

Por ejemplo, en muchas configuraciones de red, es posible para cualquier máquina monitorear el tráfico de cada máquina en la red, por lo cual es interesante mantener a los atacantes alejados del ámbito del "diálogo" de las computadoras de la red interior.

Es obvio que el tráfico que viene o va hacia el anfitrión bastión, o el mundo exterior, excede el ámbito de la red interior, por lo que todavía estará visible.

Parte del trabajo de diseñar un firewall consiste en asegurar que éste tráfico no sea, en sí, lo suficientemente confidencial como para que leerlo comprometa su sitio por completo, teniendo en cuenta que siempre se puede acudir a la criptografía.

5. Anfitrión Bastión: Es el principal punto de contacto para las conexiones que entran desde el mundo exterior:

- Sesiones de correo electrónico (SMTP) para entregar mensajes al sitio.
- Conexiones FTP que entran al servidor FTP anónimo del sitio.
- Consultas que entran al servicio de nombres de dominio (DNS) del sitio.

La mayoría de las tareas del anfitrión bastión es actuar como servidor proxy para varios servicios, ya sea ejecutando software servidor proxy especializado para protocolos específicos (como HTTP o FTP), o ejecutando servidores estándar para protocolos basados en proxy (como SMTP).

6. Enrutador Interior: El enrutador interior (o de choque) protege la red interna tanto de Internet como de la red de perímetro; es quien realiza la mayor parte del filtrado de paquetes para manejar servicios tales como Telnet, FTP, WAIS, Archie, Gopher, y otros.

Se debe limitar los servicios entre el anfitrión bastión y su red interior para reducir el número de máquinas que pueden ser atacadas desde el anfitrión bastión, en caso de verse comprometido.

7. Enrutador Exterior: La principal tarea del enrutador exterior (o de acceso) es el bloqueo de cualquier paquete que entra de Internet con direcciones fuentes falsificadas. Tales paquetes dicen venir desde la red interna, pero en realidad entran de Internet.

En Resumen

Sobre la base de las políticas de Seguridad, las posibilidades económicas y las necesidades particulares de cada usuario ya se está en condiciones de elegir el esquema de seguridad que más convenga al sitio, resta evaluar la metodología de seguimiento y respuesta a los posibles ataques a sufrir o sea definir el Plan de contingencia.

Vigilancia del Sistema

Se debe realizar un seguimiento permanente de las actividades sobre el sistema en busca de:

- Intentos de inicio de sesión con nombres de cuenta comunes (guest).
- Varios intentos de inicio de sesión con usuarios existentes en el sistema.
- Inicios de sesión desde un sitio inesperado.
- Archivos de inicio de sesión eliminados o modificados.
- Sondeos o ataques aparentes provenientes de máquinas propias.
- Inicios de sesión inesperados como usuario administrador o usuarios inesperados que de pronto se convierten en usuarios privilegiados.

Manejo de Incidentes de Seguridad en redes

▪ **Planificación de Detección de Incidentes**

Se deben realizar normas de procedimiento para el tratamiento de incidentes que incluyan el reporte inmediato de los accidentes detectados y a quién deben ser informados a los niveles de decisión correspondientes.

▪ **Planificación de la Evaluación de Incidentes**

Se debe definir quién decidirá si una situación es sospechosa o es realmente un problema de seguridad.

▪ **Planificación de la Desconexión o Cierre de las Máquinas**

El plan de respuesta debe especificar qué tipo de situación requiere desconectar o cerrar y quién puede tomar la decisión para hacerlo.

En éste punto se debe ser muy claro para evitar malas interpretaciones y que por ejemplo en un incidente menor a alguien se le ocurra bajar la llave de la electricidad.

▪ **Planificación para Notificar a las personas sobre el Incidente**

Se debe planificar quién, qué y a quienes debe informar sobre un incidente. Este problema se debe estudiar y debe encuadrarse en el Sistema de comunicaciones que posee la empresa.

Por ejemplo se debe informar a los usuarios:

- Que está fuera de servicio.
- La razón por la cual está haciendo sus vidas miserables.
- Cuáles son, con exactitud, las cosas que normalmente hacen que no van a funcionar.
- Qué deben hacer (incluyendo dejarlo en paz para que se pueda concentrar en la respuesta).
- Que, después, se les informará con todos los detalles.

▪ **Planificación para Restablecer y Recuperar**

Se debe prever que en un caso de éste tipo puede ser necesario realizar reinstalaciones por lo que se debe tener a mano la posibilidad de reinstalar todos los sistemas que corren en la red de manera inmediata.

▪ **Planificación de la Documentación**

Se deben incluir instrucciones básicas sobre los métodos de documentación de los incidentes, le servirá sobre todo si en un futuro se decide iniciar acciones legales. Este aspecto se contemplará en las normativas o procedimientos.

▪ **Revisión Periódica de los Planes - Auditorías**

Tan importante como la elección de los métodos de seguridad son las respuestas que se deben dar en caso de incidentes y más importante que éstas dos, sobre todo por las nuevas formas de ataque, es la revisión permanente de los sistemas de seguridad y sus respuestas.

Tal revisión puede manejarse mediante auditorías, seguimiento del sistema y ataques modelo para verificar la seguridad.

Hasta aquí se presentó un pequeño resumen sobre seguridad en redes. Esta problemática es inagotable y siempre presenta una nueva faceta en cuanto a proteger la información y los datos en un ambiente distribuido. Lo importante es prevenir o sea pensar en los posibles ataques y generar respuestas estudiadas para mitigar las consecuencias.

ANEXO 8. B. SEGURIDAD EN SISTEMAS DISTRIBUIDOS.

Protección de Objetos en Amoeba

Amoeba es un sistema operativo distribuido, permite que una colección de CPU y equipo de E/S se comporten como una sola computadora y proporciona elementos de programación en paralelo. Su arquitectura está conformada por una pila de procesadores, terminales, servidor de archivos y servidor de impresión (esta configuración es un ejemplo, puede expandirse).

Los objetos son el concepto más importante que se maneja en este sistema. Un objeto es una parte encapsulada en la que se pueden llevar a cabo ciertas operaciones bien definidas. En lugar de contener procesos o métodos que hagan cosas son controlados por un proceso servidor. La protección de cada objeto depende de sus *posibilidades* (serie de campos que incluyen: puerto del servidor, objeto, derechos y verificación). El campo puerto del servidor sirve para localizar la maquina que contiene a dicho servidor. Para identificar al objeto se utiliza el campo del mismo nombre. Los derechos son un campo representado por un mapa de bits que indica las operaciones permitidas al poseedor de una posibilidad. Por último la verificación se utiliza para darle validez a la posibilidad mediante los procesos del usuario.

La protección se realiza de la siguiente manera: Cuando se crea un objeto, el servidor elige un campo Verificación al azar y lo almacena en la nueva posibilidad y dentro de sus propias tablas. Se activan todos los bits de una nueva posibilidad y esta posibilidad del poseedor regresa al cliente. Cuando

la posibilidad regresa al servidor en una solicitud para llevar a cabo una operación, se verifica el campo verificación.

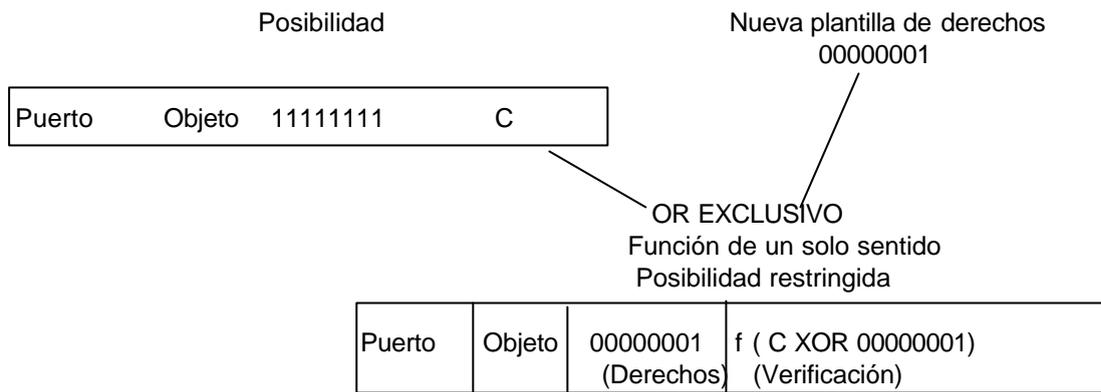


Figura 8.B. 1 Creación de objetos con protección en Amoeba.

Para crear una posibilidad restringida, un cliente debe regresar una posibilidad al servidor, junto con los bits de una plantilla para los nuevos derechos. Mediante un OR exclusivo con los nuevos derechos se ejecuta el resultado a través de una función de un solo sentido. Esa función debe tener la característica de ser fácil de encontrar dado un valor, pero de muy difícil inversa. La nueva posibilidad es creada, con sus nuevos valores en el campo derechos y la salida de la función en el campo verificación más el campo objetos ya existente anteriormente. El cliente puede enviar esta posibilidad, si lo desea, a otro proceso. La forma de realizar lo antes descrito se ve en la figura B.1.

Todo usuario que intente añadir derechos que no le son permitidos se verá imposibilitado. Las posibilidades se utilizan en Amoeba para dar nombres a todos los objetos y para su protección. Además, sirve para realizar una operación sobre un objeto sin tener que saber en qué lugar se encuentra.

Sistemas Confiables

Un requerimiento aceptable a la hora de proteger un sistema es el encontrado generalmente en ámbitos militares. Allí la información es clasificada en no procesada (U) confidencial (C), secreta (S), super secreta (TS), etc. Este concepto puede ser aplicable en otras áreas, donde los usuarios pueden tener acceso a diferentes rangos de áreas. Por ejemplo, el más alto nivel de seguridad pueden ser los documentos de planeamiento estratégico, solo accesibles por personal del staff de directivos, luego pueden venir datos de personal o finanzas, accesibles por personal de administración, etc.

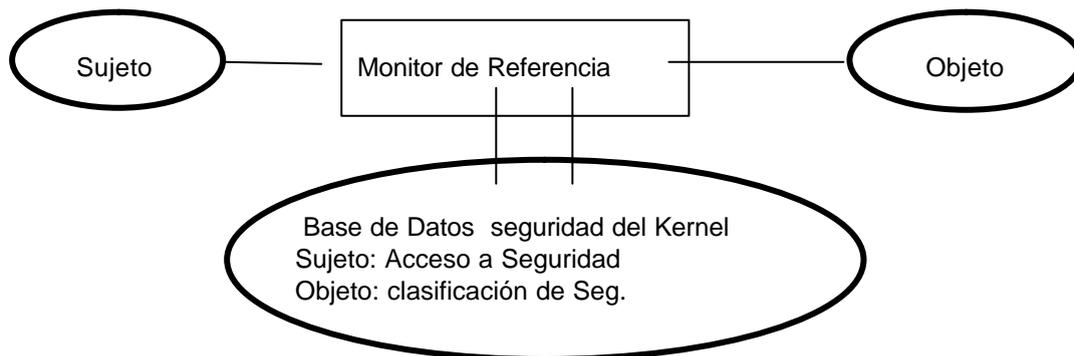


Figura 8. B. 2 Esquema de monitor de referencia.

Cuando se define un sistema de este tipo la seguridad en cuestión se denomina *multinivel*. Primero haré una breve descripción en sistemas simples para pasar a discutir conceptos de sistemas en red, donde se manejan los sistemas distribuidos.

Un sistema multinivel debe garantizar dos aspectos fundamentales: Que un sujeto con una determinada prioridad no acceda a objetos de más alta prioridad que la de él, a no ser que sea autorizado a tal efecto por un superior y que no escriba objetos de más baja prioridad.

Para llevar a cabo estas reglas se emplea el concepto de *monitor de referencia*. Este es un elemento del hardware y del sistema operativo de una computadora que se encarga de regular el acceso de sujetos a objetos según sus parámetros de seguridad. Tiene acceso a un archivo conocido como *base*

de datos de seguridad del kernel, en la cual están expresados los privilegios de acceso y los atributos de protección de cada objeto.

Las siguientes son tres propiedades que debe tener un monitor de referencia:

Mediación completa: las reglas son reforzadas con cada acceso.

Aislamiento: La base de datos y el monitor deben estar protegidos contra accesos indeseados

Verificabilidad: Debe poder demostrarse matemáticamente que el monitor cumple con las propiedades anteriores.

Un sistema que garantice las propiedades se conoce como sistema confiable. La figura 8.B.2 muestra un esquema de monitor de referencia.

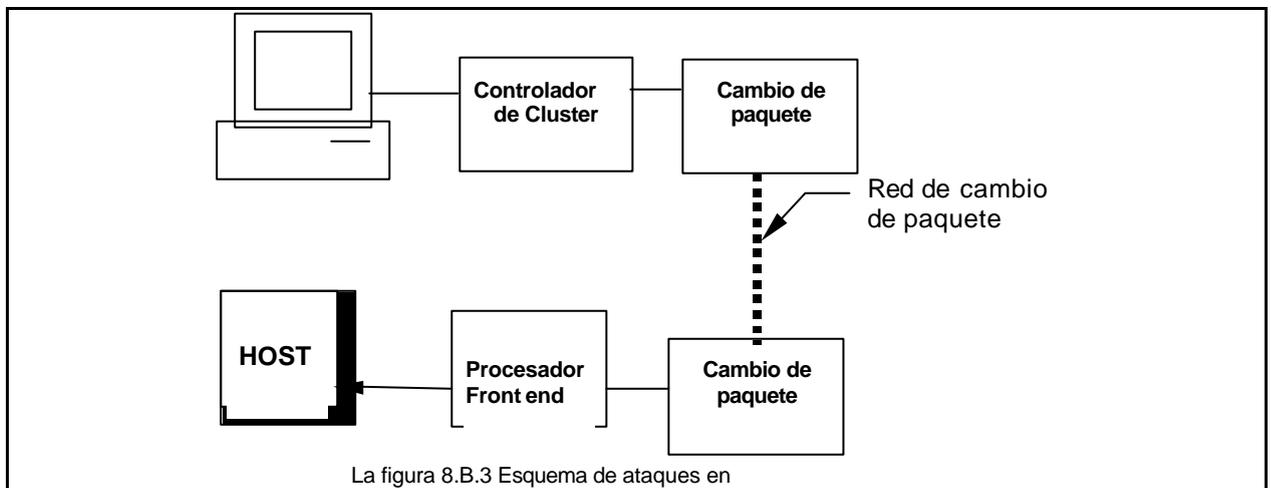
Seguridad en la Red (Sistemas Distribuidos)

Al aumentar la complejidad a una red, se agregan problemas a los ya vistos. Los esquemas usados para solucionar los problemas consisten en aplicar la solución como parte del sistema distribuido o como una utilidad adicional.

Existen distintos lugares en donde pueden ocurrir ataques, durante todo el trayecto de la comunicación entre el servidor y el cliente.

El proceso que se sigue es el siguiente: La información ingresada en la terminal por el usuario debe pasar a través de un link de comunicaciones por un controlador de cluster. Desde allí entra en una red de cambio de paquete por otro link que conecta al controlador con uno de los nodos de dicha red. Dentro de la red, la información pasa a través de un número de nodos y links hasta que llega al nodo al cual está conectado el host de servicio. Dicho host está conectado a un procesador front end y después a través del link al host.

Un ataque puede ocurrir en cualquier punto de los links. Si el ataque es activo, el atacante debe tomar control de la red e interceptar las transmisiones. En un ataque pasivo, el atacante solo monitorea las transmisiones y logra ver el contenido, sin embargo no puede modificarlas. Otro tipo de ataques puede concentrarse sobre el/los procesadores, como intentos de modificar el software o hardware o interceptar emanaciones electromagnéticas. La figura 8.B.2 muestra el esquema anteriormente explicado.



Ejemplo de Seguridad en un Sistema Distribuido: Andrew

Andrew es un sistema de computación distribuido desarrollado en la universidad de Carnegie Mellon. Su característica más importante es su tamaño, el cual puede alcanzar hasta 10.000 nodos. En la fecha de su puesta en marcha, en 1986, estaba compuesto por 400 estaciones de trabajo que utilizaban 1200 usuarios activos. Su file system almacenaba 15 gigabytes de datos esparcidos en 15 servidores.

La estructura del sistema se basa en dos tipos de componentes: Vice y Virtue. Una estación de trabajo Virtue provee el poder y la capacidad de una computadora personal dedicada, mientras que una Vice es el soporte de la abstracción de tiempo compartido. Las Virtue manejan el sistema operativo UNIX 4.3 BSD.

Los mecanismos de seguridad en Andrew aseguran que la información es liberada y modificada solo en forma autorizada. Un aspecto importante es asumir quién lleva a cabo la seguridad. Los lugares son los servidores Vice, por ser su número mucho menor que las miles de estaciones de trabajo. Estos

servidores se ubican físicamente en cuartos seguros y se les suministra software confiable, es decir, no se ejecuta nunca software de usuario. Por razones operacionales es necesario proveer utilidades que manipulen directamente los datos del file system de Andrew. Estas utilidades pueden ser ejecutadas solo por superusuarios en los servidores. El acceso a los servidores y la capacidad de convertirse en superusuario deben ser privilegios celosamente guardados.

Un superusuario podría instalar tranquilamente un caballo de Troya en el sistema.

Como las estaciones de trabajo pueden ser públicas o privadas los mecanismos de copropietarios y usuarios individuales son establecidos.

Debido a la gran cantidad de nodos y a la imposibilidad de garantizar la integridad física de los datos en la red es que se utilizan mecanismos de encriptación end to end. Una persona que acceda a un dato encriptado puede borrarlo pero no enterarse de que se trata. Agregando el beneficio del manejo de replicas, lo anterior no presenta mayores inconvenientes.

Finalmente, el diseño del file system de Andrew postula el uso de un canal independiente de comunicación segura entre los servidores Vice.

El dominio de protección

La pregunta fundamental radica en el siguiente hecho: Puede el agente X realizar una operación Y sobre el agente Z?

En Andrew el dominio está dividido en grupos y usuarios. Un usuario es una entidad, usualmente un humano, que puede autenticarse en un servidor Vice, ser responsable de sus propias acciones y ser contabilizado su consumo de recursos. Un grupo es un conjunto de otros grupos y usuarios asociados a un usuario llamado *dueño*.

El nombre del dueño es prefijo del nombre del grupo. Los servidores Vice utilizan identificadores únicos de 32 bits a tal efecto.

Un usuario distinguido llamado "System" es omnipotente; los servidores Vice no le aplican ningún tipo de protección. El dominio de protección incluye otros dos tipos de entidades: el grupo "System: Any User", el cual contiene a todos los usuarios autenticados y el usuario "Anonymous", que corresponde a un usuario no autenticado.

Autenticación y comunicación segura

En Andrew, las dos partes que interactúan son un usuario en una estación de trabajo Virtue y un servidor Vice. Los posibles intrusos están en la red como espías o como hardware o software que alteran los datos a ser transmitidos.

El mecanismo de autenticación es una variante del de Needham y Schroeder, que usa llaves de encriptación privadas. La función principal es desglosada en tres componentes:

- Un mecanismo de llamada a procedimiento remoto (RPC) que provee soporte de seguridad.
- Un esquema para obtener y usar tokens de autenticación (objetos que vienen en pares y cuya posesión es prueba de autenticidad)
- Un servidor de autenticación que es un repositorio de la información de contraseña. Este corre en una máquina Vice confiable y es responsable para restringir el acceso de Vice y para determinar si el acceso por un usuario es válido.

Estos componentes funcionan de la siguiente manera. En respuesta a un prompt de login estándar UNIX en una estación de trabajo, el usuario provee su nombre y contraseña. La contraseña se usa para establecer una conexión RPC segura con el servidor de autenticación. De este se obtienen un par de tokens de autenticación y son salvados en la estación de trabajo. Estos tokens son usados para establecer conexiones seguras RPC con los servidores de archivos. El establecimiento de una comunicación es totalmente transparente al usuario, el cual no tiene que ingresar su contraseña cada vez que se establece una nueva conexión.

Protección en Vice

Vice refuerza la política de protección especificada por los usuarios, como custodio de la información compartida.

La revocación de los privilegios de acceso es una operación importante y común. Para ello Andrew implementa el mecanismo de *listas de acceso*. Estas se implementan como un paquete disponible a cualquier servicio de Vice, aunque solo lo usa el file system distribuido. Una entrada en la lista de acceso equivale a una posición de 32 bits que significan una lista de derechos que varían según el usuario. La lista se divide en 2: derechos positivos (posesión) y negativos (no posesión).

Este esquema se usa para evitar la pérdida de tiempo que generaría una doble búsqueda en los permisos que se posee para saber los que no se posee.

Vice asocia una lista de acceso con cada directorio, y le aplica a cada archivo el correspondiente status de protección. El usuario dispone en todo momento de las diferentes protecciones que le recaen

sobre archivos accedidos. Cuando un archivo necesita una protección diferente al resto de los archivos del directorio dado, se procede a ubicar el mismo en otro directorio con la protección adecuada y se le hace un link simbólico al directorio que era su origen.

Los derechos que se asocian con un directorio son: read, write, lookup, insert, delete, manager (modifica la lista de acceso del directorio), y lock. Por ejemplo si se quiere tener completo acceso la combinación será **rwlidka**.

El dominio de protección de la información es mantenido en una base de datos que es replicada en cada servidor de archivos Vice. La base consiste en datos en disco y un índice que es cacheado en memoria virtual, el cual contiene información para efectuar traducciones y el offset en el archivo de datos. Cada entrada en la base de datos corresponde a un sólo usuario o grupo.

Protección en Virtue

Como multiusuario UNIX, Virtue refuerza la seguridad entre los usuarios que usan concurrentemente una estación de trabajo. Su rol en Andrew se relaciona con las siguientes funciones de seguridad:

- Emula semántica de UNIX en los archivos Vice. Así como Vice se encarga de los directorios, Virtue crea el ambiente de permisos ya conocidos para los archivos individuales.
- Asegura que el caching es consistente con la protección en Vice.
- Permite a los usuarios total control sobre sus estaciones de trabajo, sin comprometer la seguridad en Vice.
- Provee interfases de usuario y programa para usar explícitamente los mecanismos de seguridad de Vice.

Uso de Recursos

La ausencia de puntos focalizados de asignación de recursos hace que el control de los mismos sea dificultoso en un sistema distribuido, ya que no existe un agente como el algoritmo de planificación con el que cuentan los sistemas de tiempo compartido. Otra diferencia es que un proceso en un sistema de tiempo compartido tiene que ser autenticado antes de que pueda consumir una cantidad apreciable de recursos.

En contraste, cada estación de trabajo en Andrew puede ser modificada para consumir anónimamente ancho de banda de la red y ciclos de CPU del servidor.

Aunque Andrew no está diseñado para ser inmune a las violaciones de seguridad por negación de recursos, provee control sobre el almacenamiento en disco y los ciclos de CPU del servidor y el almacenamiento en disco y los ciclos de CPU de la estación de trabajo. Sin embargo existen ciertas violaciones no factibles de ser detectadas inmediatamente.

Conclusión sobre Seguridad en Sistemas Distribuidos:

La creciente evolución de los sistemas operativos distribuidos hace que se generen paralelamente dos tendencias. La primera, de parte de usuarios maliciosos o interesados que buscan formas de obtener beneficios del sistema, mediante el acceso a información no autorizada o el daño premeditado de datos importantes. La segunda, de parte de los desarrolladores de los sistemas, para garantizar la seguridad e integridad de los datos.

Canales seguros, encriptación, estándares y otras medidas de protección son empleadas y de su correcto uso depende el éxito de los sistemas operativos distribuidos, dado que a nadie le agrada tener su información administrada por un sistema que sea inseguro, ya sea por la pérdida de información o por el conocimiento de ella por parte de intrusos.

Seguridad Multinivel

Es la extensión del concepto de sistema confiable a un entorno de red. Las mismas políticas vistas se aplicarán a cada uno de los componentes.

Para lograr esto se tenderá a tratar que cada uno de los sistemas individualmente sean confiables. Esta alternativa es muy costosa, por lo que se utiliza un producto conocido como *unidad de interfase confiable* (TIU=Trust Interface Unit). Una TIU se conecta a la red y las terminales hacen lo propio con la TIU. Su función principal es la de aceptar datos y transmitirlos por la red. Para llevar a cabo su función se requiere que la TIU etiquete cada paquete con una etiqueta de seguridad y que acepte solo paquetes que estén etiquetados con su propia o menor prioridad. La figura 8.B.4. muestra el esquema.

La TIU asegura que los datos recibidos por el host son solo esos datos que están en el rango de la clasificación de los que se permite tener. Todo dato transmitido por el host es etiquetado por la TIU con su nivel de seguridad. Las terminales también se conectan a la TIU y deben operar todas al mismo nivel.

Los dispositivos que no están clasificados requieren una TIU que opere en ese nivel. La función de filtro es la más importante para los dispositivos no clasificados.

La TIU es una forma de aplicar la tecnología de sistema confiable a un sistema distribuido. Cualquier función crítica de seguridad puede ser implementada como una pieza separada de hardware confiable o como una pieza aislada de software confiable con un sistema de propósito general.

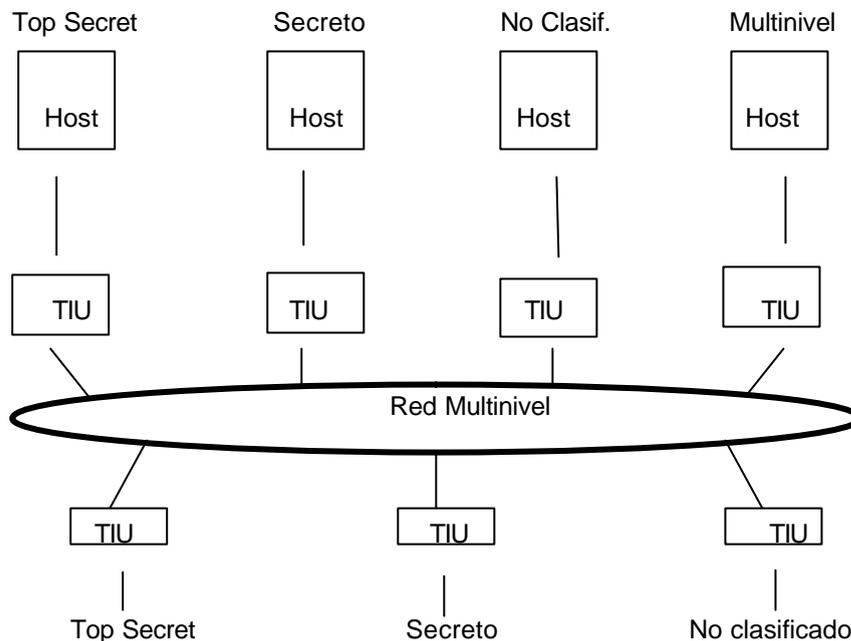


Figura 8.B.4. : Aplicación de la unidad de interfases confiable.

ANEXO 8.C. Criptografía en Redes

Antes de hacer una descripción de este proceso en la red conviene definir algunos conceptos básicos.

La encriptación es un método que permite transmitir la información por rutas que no son confiables. El mecanismo básico es el siguiente:

1. La información (texto) es encriptada (codificada) y pasa de una forma legible a otra interna (cifrada), la cual no tiene sentido.
2. La forma cifrada puede guardarse en un archivo legible o transmitirse por los canales inseguros
3. El que recibe la información debe desencriptarla (descodificarla) para poder interpretar su contenido.

Lo más importante de un esquema de encriptado es lograr que sea imposible de desencriptar para un usuario que no esté autorizado. Ello garantiza que no haya más de una forma de descifrarlo.

Para realizar criptografía se cuenta con dos funciones de encriptado y desencriptado E y D, cada una es una función de dos variables llamadas llave y texto. La operación de encriptación se llama E(K,T) y la desencriptación D(K,T). Se denomina *cognatos* al par de llaves que cumplen

$$D(K_2, E(K_1, T)) = T$$

Si $K_1=K_2$ el sistema de criptografía se llama *simétrico*, de otra forma es *asimétrico*. En este último conviene que conocida una clave sea muy difícil adivinar la otra. Además es conocida como encriptación de llave pública porque quien encripta publica una de las llaves como si fuera una guía telefónica. Dado un mensaje encriptado, cualquier intento de desencriptarlo usando la clave pública habilitará al desencriptador para inferir quien lo encriptó. Otra de las garantías es que si se encripta un texto ASCII y la llave usada para desencriptar no es la correcta, el texto obtenido no se parecerá en nada al texto ASCII original.

Implementación

Para encriptación de llave pública se utiliza el algoritmo RSA.

Sea N un número grande con dos factores primos p y q de similar magnitud. $\phi(N)$ se define como $(p-1)(q-1)$. Se elige un número d que sea coprimo con $\phi(N)$ y se computa e tal que $e*d \equiv 1 \pmod{\phi(N)}$.

De esta manera un mensaje puede ser encriptado elevándolo a la potencia e (mod $\phi(N)$) y descryptado elevando el texto cifrado a la potencia d . La seguridad de este algoritmo radica en la capacidad de factorizar grandes números.

Para encriptación simétrica el algoritmo más conocido es el DES. DES es un bloque cifrado que toma 64 bits de texto y produce 64 bits de texto cifrado bajo el control de una llave de 56 bits. Está construido con la siguiente idea. Supóngase que la cantidad de 64 bits a ser encriptada puede ser dividida en dos partes, L_0 y R_0 . Entonces $G(X,Y)$ es una función de X y una llave K de 32 bits (dado un valor de G uno no puede ser capaz de determinar X). Luego se determina:

$$L_1 = R_0 \text{ y } R_1 = \text{XOR}(L_0, G(K, R_0)).$$

Para que el sistema sea realmente efectivo conviene realizar el proceso varias veces. Esto no genera mayor problema para el poseedor de la llave K , quien puede realizar el proceso inverso tantas veces como sea necesario.

Canales de encriptación

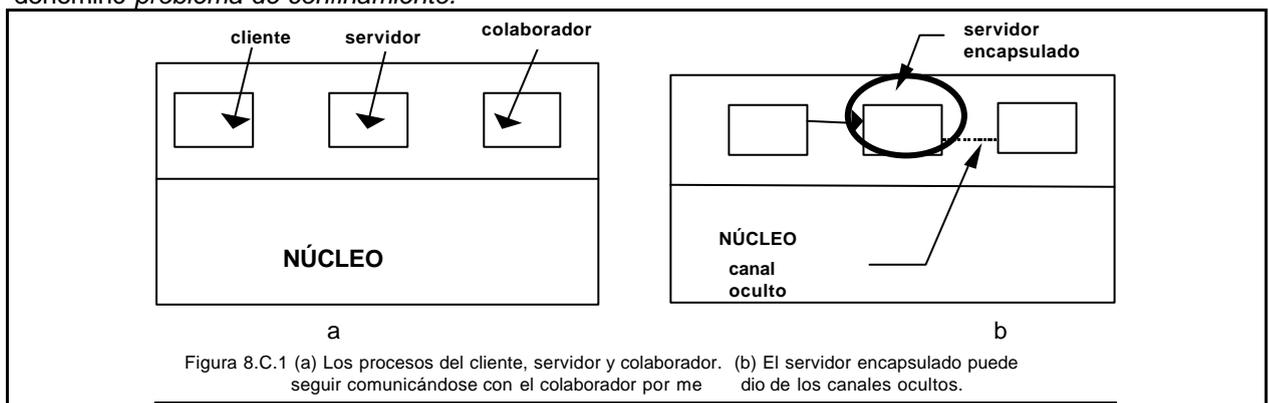
El propósito fundamental de la encriptación en sistemas distribuidos es hacer aseveraciones acerca de la fuente o destino de los mensajes. Un ejemplo de canal es un conjunto de mensajes a ser encriptados.

Debe ser posible para el usuario dar nombre a sus canales. Es evidente que si se usa encriptación simétrica no se le puede dar el nombre de la llave, ya que se haría pública. Los tres tipos de llave de referencia para identificar una llave son: Un offset dentro de una tabla de llaves mantenida en el que recibe el paquete, la llave encriptada con otra llave solo conocida por el receptor o una combinación de ambas.

Para ayudar a un sistema de encriptación simétrico se cuenta con *relays*, máquinas seguras que toman un único estado igual a la llave que no son compartidos por nadie excepto un servicio de autenticación. La idea es hacer pública una versión de la llave simétrica encriptándola con una llave de referencia que provee el relay. El relay garantiza integridad, no así seguridad.

Canales ocultos

El modelo de Lamson es empleado en grandes sistemas distribuidos de tiempo compartido. Sirve para demostrar la ineficacia de ciertas matrices de protección. Cuenta con tres procesos, el *cliente*, el cual desea que el segundo proceso el *servidor* realice un trabajo. Como no confían plenamente el uno del otro para realizar dicha tarea, se toma un tercer proceso, llamado *colaborador*, que conspira con el servidor para robar los datos confidenciales del cliente. El objetivo de este modelo es que en el sistema a diseñar, sea imposible para el servidor obtener la información legítimamente del cliente. Este problema se denominó *problema de confinamiento*.



En la figura 8.C.1 se esquematiza el modelo.

Se debe tratar que el servidor, mediante el encapsulamiento no pueda transferir información al colaborador. Esto se logra con una matriz de protección, en cuyas filas se tienen los distintos dominios de protección y en las columnas se tienen los distintos procesos en cuestión. También puede garantizarse que el servidor no se pueda comunicar con el colaborador mediante el mecanismo de comunicación entre procesos del sistema. Sin embargo el servidor puede comunicarse mediante un *canal oculto*. Este se produce si el servidor discrimina los bits, realizando demoras ya sea calculando o durmiendo durante cierta cantidad de tiempo. La mejor respuesta se producirá cuando el servidor envíe un 0.

El canal oculto es un canal con ruido, pero puede corregirse si se utiliza algún código corrector (por ejemplo: el código de redundancia de Hamming). Ningún otro modelo de protección basado en matrices puede detectar dichos canales.

Entre otros métodos de canales ocultos se cuentan: la velocidad de paginación (un fallo se cuenta como 1 y ninguna falla como 0), el servidor al bloquear procesos en sistemas que lo permiten, la apropiación y liberación de recursos (unidades de cinta, plotters; 1 para apropiarse, 0 para liberar), etc.

Encriptación del link y end to end:

En un sistema distribuido, que funcione en un entorno de red, se debe determinar la ubicación de los dispositivos de encriptación, ya que existen diversas capas en las comunicaciones. Actualmente se cuenta con dos formas de aplicar la encriptación: *encriptación del link* y *encriptación end to end*.

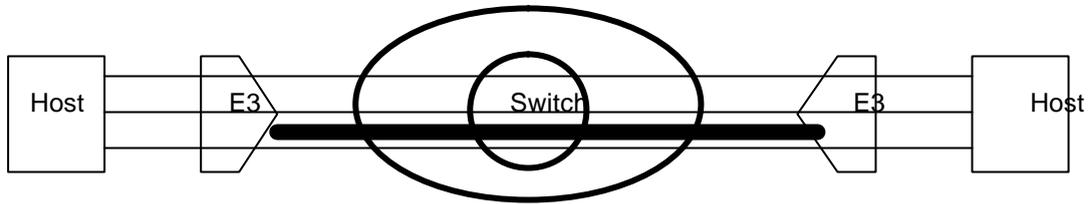


Figura 8.C.2 Encriptación End to End

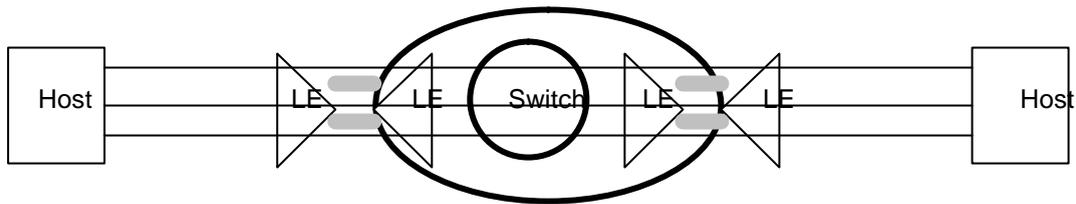


Figura 8.C.3. Encriptación de Link

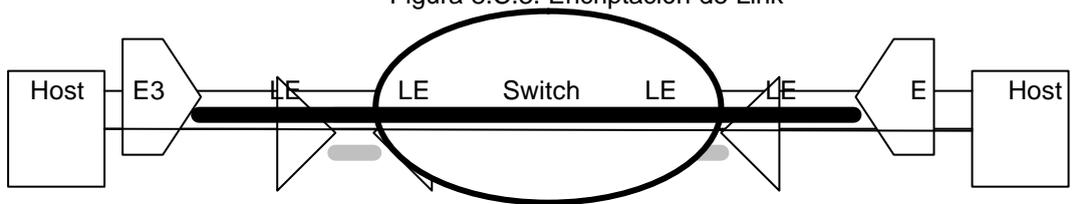


Figura 8.C.4 Ambos Métodos

En la encriptación del link cada extremo de un link que se considera inseguro cuenta con un dispositivo de encriptación. Una desventaja de esto es el gran número de dispositivos si se cuenta con gran número de links. Otra radica en el hecho de que cada vez que entra en un cambio de paquete, el mensaje debe ser descryptado. Esto debe realizarse para que en el cambio se obtenga la dirección a donde debe ir; es evidente que si el paquete está encriptado, no se podrá extraer dicha dirección. En cada nodo del cambio del paquete, el mismo será vulnerable.

La encriptación end to end se lleva a cabo en los sistemas emisor y receptor. Los datos se encriptan en origen y no son alterados a través de la red hasta que llegan al receptor del paquete, quien debe descryptarlos. Este receptor comparte con la fuente la llave con la que descrypta.

El punto débil de este esquema es presentado con la siguiente situación. Un host se conecta a una X.25 red de cambio de paquetes, establece un circuito virtual con otro host y está preparada para transmitir datos de uno a otro por medio de encriptación end to end. Los paquetes a transmitir están conformados por un encabezado y la parte de datos de usuario. Aplicando end to end a todo el paquete no podrá realizarse el cambio dentro de la red por estar encriptada la dirección. Si se encripta solo la parte de datos de usuario, dejando el encabezado sin encriptar se puede enviar a través de la red, pero acecha el peligro de que la parte de encabezado puede ser modificada. Por esto es que se emplea un método combinado de encriptación de link y end to end. De esta manera el host que envía el paquete encripta la parte de datos usando end to end. Luego hace lo propio con todo el paquete, usando encriptación de link. Cada vez que se debe hacer un cambio de paquete se debe descryptarlo usando descryptación de link. Esto garantiza que en cada nodo se conocerá el encabezado y por lo tanto el próximo destino del paquete. No debe olvidarse el volver a encriptar una vez obtenida la dirección.

Se presentan en la figuras 8.C.2, 8.C.3 un esquema del uso de cada método y por último en la figura 8.C.4, la combinación de ambos.

Distribución de llaves

Como ya se ha visto las dos partes que intervienen en una comunicación deben tener la misma llave, y esta tiene que estar protegida para que otros no puedan acceder a ella. Ocurre lo mismo que con una contraseña, sería conveniente que la llave se renovara periódicamente. Para lograr este objetivo se utiliza el esquema de distribución de llave, de manera tal que los interesados intercambien la llave sin que otros la conozcan.

Existen diversas maneras de lograr el esquema. Por ejemplo, si la comunicación es entre A y B son las siguientes.

1. A selecciona la llave y se la envía físicamente a B
2. Un tercero selecciona la llave y la envía a A y B
3. Si A y B usaron previamente una clave, pueden usarla para transmitir la nueva.
4. Si A y B por separado están conectados con C con un determinado método de encriptación, C puede enviar una llave por los links con A y B.

Las opciones 1 y 2 sirven si se trata de encriptación de link, en donde cada nodo se comunica con su adyacente. Sin embargo no es conveniente en caso de que se use encriptación end to end. En los sistemas distribuidos cada host o terminal necesita saber las llaves dinámicamente, cosa no posible si cada cambio es enviado manualmente como sugieren 1 y 2.

ACC: Centro de Control de Acceso. KDC: Centro de distribución de llave. FEP: Procesador Front end

La opción 3 puede darse en ambos casos, pero si un atacante gana acceso a una llave, todas las demás le serán reveladas. Por ello la opción 4 es la más adecuada para un esquema de encriptación end to end.

Un ejemplo de un esquema de encriptación end to end será presentado a continuación. A dicho esquema se le puede agregar sin variar el esquema de link.

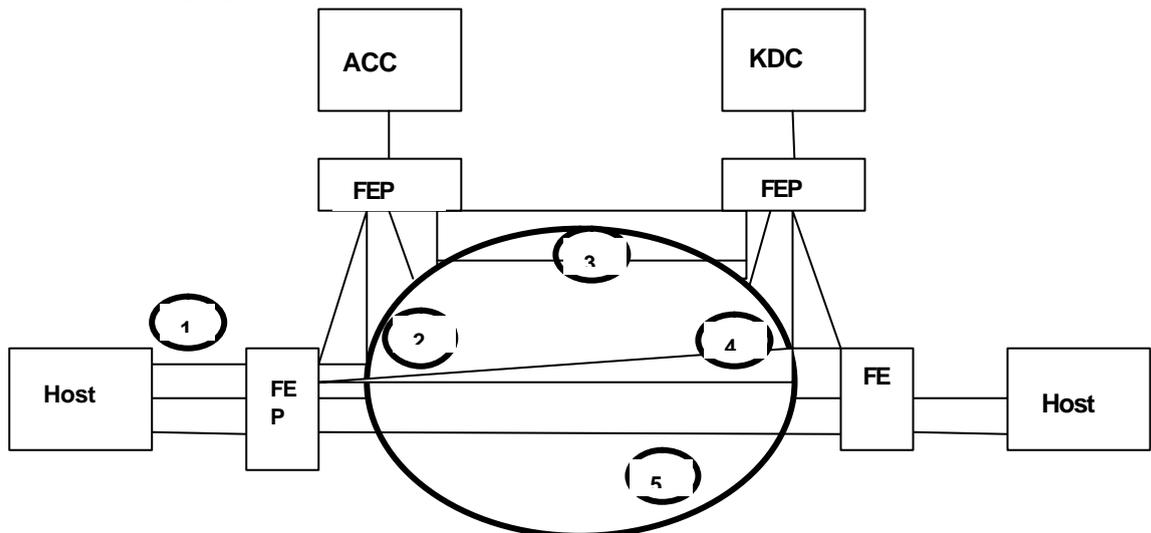


Figura 8.C.5 Encriptación End to End en una Red

Las llaves de este esquema son:

- Llave de Sesión: Dura mientras dure la conexión lógica entre dos sistemas
- Llave Permanente: es usada con motivo de la distribución de otras llaves
- La configuración cuenta con los siguientes elementos.
- Centro de Control de Acceso: determina que sistemas pueden comunicarse
- Centro de Distribución de llaves: Provee la llave de sesión cuando puede realizarse la comunicación.

Procesador Front end (FEP): realiza la encriptación end to end y obtiene del host o terminal la llave de sesión.

La figura 8.C.5 muestra un esquema de este tipo y también muestra los pasos para establecer una conexión. Primero se transmite un paquete de pedido de conexión (1). Luego el FEP graba el paquete y consulta al centro de control de Acceso el permiso para establecer la conexión (2). La comunicación entre el FEP y el centro de control de acceso es encriptada con una llave permanente solo conocida por ellos (3). El centro de control de acceso aprueba la comunicación envía un mensaje al centro de distribución de llave, pidiéndole que genere una llave de sesión. El centro de distribución de llave genera la misma y se la envía a los FEPs adecuados, usando una única llave permanente para cada uno (4). El FEP que solicitó la

comunicación puede liberar al paquete de pedido de conexión y la conexión se establece entre los dos sistemas (5).

Existen diversas variantes a este esquema. La diferencia está dada por una mayor concentración de los dispositivos y una menor liberación de tareas del host.

Traffic Padding

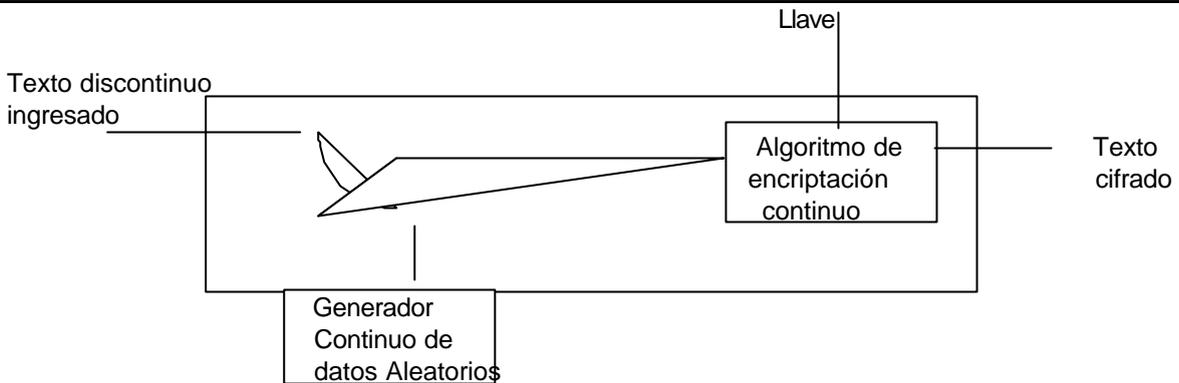


Figura 8.C.6 Dispositivo de encriptación con Traffic Padding

Se denomina de esta manera a la forma de agregar mediante una función ruido, de manera tal que sea imposible para un atacante distinguir entre información y basura. El ruido es agregado en forma de texto cifrado aún cuando no se está transmitiendo información. Cuando está disponible el texto útil, es cifrado y transmitido. Cuando no está presente, el cifrado es el texto random (ruido).

ANEXO 8.D. El Esquema Switchboard para hacer Sistemas Distribuidos Multimedia seguros

Los sistemas distribuidos multimedia surgen a partir del mejoramiento de sistemas con ventanas, estilo X Windows, con elementos multimedia o imágenes digitalizadas de vídeo. Desde ya que la seguridad en un sistema de este tipo constituye un punto crucial. No es deseable que un intruso obtenga información al conectarse a la red sin estar autorizado mediante el acceso a nuestra cámara de vídeo. Tampoco deseáramos que este intruso evaporara datos importantes como grabaciones o imágenes.

El esquema Switchboard

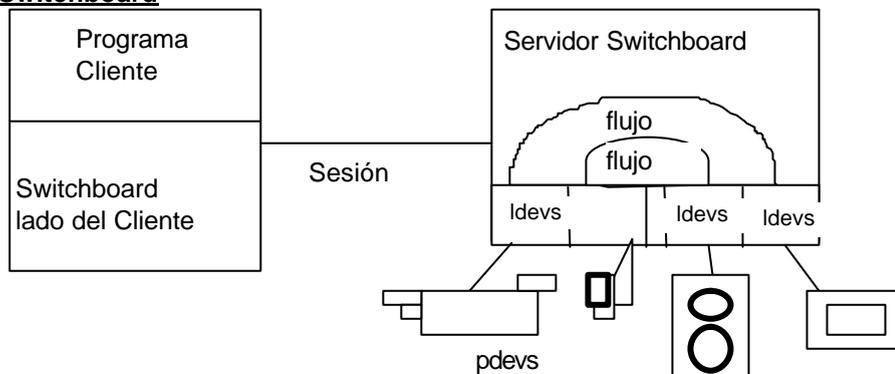


Figura 8.D.1 : un típico escenario switchboard

La estructura se basa en un modelo cliente/servidor que presenta una estructura de objetos. El término multimedia se refiere a la representación de la información de audio y vídeo. En switchboard se definen dos tipos de abstracciones: visibles al cliente del switchboard y visibles solo a través del servidor. La figura 8.D.1. muestra un típico escenario switchboard

Abstracciones visibles a la aplicación

La capa de cliente define abstracciones del hardware actual que controla el lado del servidor del switchboard. Estas abstracciones pueden ser usadas para controlar el hardware multimedia y el flujo de datos entre los dispositivos de hardware (micrófonos, parlantes, etc.). Antes de la definición formal de las abstracciones principales se describe el escenario de la figura 8.D.1. En este caso un programa cliente ha interconectado los siguientes elementos multimedia que forman parte de la estación de trabajo en la que está presente el servidor: un micrófono a un parlante y una cámara a una ventana (para salida digital de vídeo).

Las abreviaturas usadas en la figura 8.D.1. significan lo siguiente:

- Dispositivos físicos (pdevs): se representan por un objeto que es una pieza de hardware multimedia en ambos lados (cliente y servidor). Pueden dividirse en dos clases: pdevs de hardware (cámaras, micrófonos) y pdevs representadas por software del otro lado (secuencias almacenadas de lo filmado con la cámara). Además tienen asociado un conjunto de posibles operaciones y un conjunto de atributos que los caracterizan. Por ej.: una cámara las operaciones serían hacer zoom, hacer foco, rotar y los atributos son el máximo valor de zoom o la resolución de la toma. También se subdividen en pdevs de entrada, salida y ambos.
- Dispositivos lógicos (ldevs): son el espejo del siguiente nivel de abstracción. Son estáticos ya que su vida depende de la vida del lado del servidor. Esta abstracción trata de que el cambio entre pdevs con características similares no tenga que liberar la conexión entre los pdevs involucrados.
- Sesiones: son las conexiones del lado del cliente al servidor del switchboard. La sesión se usa para la transmisión de datos de control entre el cliente y el servidor del switchboard.
- Flujos: ldevs y cdevs (Dispositivos de Constructor) son conectados vía flujos, los cuales se encargan de la transferencia de datos multimedia desde la fuente. Dos son los tipos de flujo, flujos de red, que cruzan las fronteras de los nodos simples de la red y flujos internos al nodo que llevan datos solo adentro de un nodo.
- Dispositivos de Constructor (cdevs): además de los ldevs, la clase de los cdevs es otro tipo de bloques de construcción de aplicaciones. Son usados para conversiones de formato, combinación y división de streams (flujo) de datos. Un ejemplo sencillo de cdev es un duplicador, el cual produce dos copias idénticas de un flujo que recibe.

Usando ldevs, cdevs y flujos, puede usarse un grafo acíclico para representar el flujo de información multimedia en un sistema distribuido.

Desde el punto de vista del programador, estas abstracciones son representadas por objetos. Esto puede representarse en C++ por ejemplo. Un problema es el carácter distribuido de los objetos: para ser accedido por una aplicación, un parlante ser visible del lado del cliente del switchboard, mientras el objeto actual (el parlante) reside del lado del servidor, posiblemente en otro nodo. Para hacer la distribución transparente a las aplicaciones se utilizan los objetos próximos, que desde la capa del cliente hacen llamadas a procedimientos para transmitir pedidos al servidor y obtener los resultados.

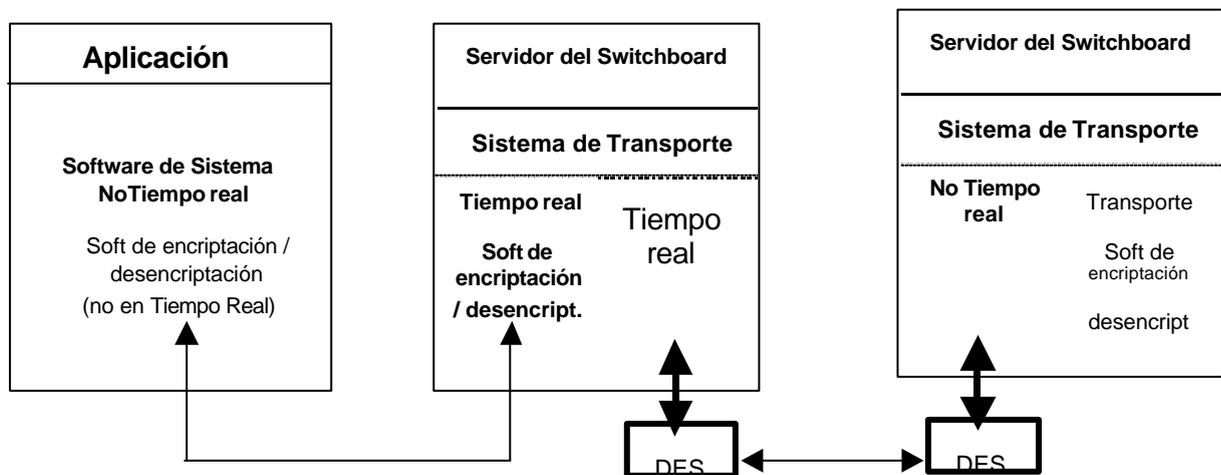


figura 8.D.2.. Modelo del sistema para comunicación server para cliente

Modelo del sistema

El sistema de seguridad que utiliza es de dos partes: una atiende las conexiones entre el cliente y el servidor del lado del switchboard. En este caso puede usarse software de encriptación lento y barato debido a que estas conexiones no son parte del llamado Ambiente de Tiempo Real (RTE). En contraste con esto el flujo multimedia entre dos servidores switchboard es parte del y tiene que funcionar con elementos de tiempo real. Aquí una solución sería hardware rápido como la encriptación de datos standard (DES). Son chips que encriptan y desencriptan datos en tiempo real. Son usados porque una solución puramente de software sería muy lenta para este propósito. La figura 8.D.2. muestra el esquema.

Seguridad en el switchboard

Los ataques a prevenir son: acceso no autorizado a dispositivos multimedia, conexión no autorizada de dos dispositivos multimedia por un flujo y simular una falsa identificación (intruso). Una

persona que posea los medios adecuados de hardware puede transformarse en un espía dispuesto a romper la seguridad del sistema.

La protección en este caso apunta a dos clases: estática (quien puede realizar qué operación con un dispositivo multimedia) y dinámica (a quién se le permite conectar qué dispositivos por medio de un flujo).

Protección Estática

Las operaciones posibles en un dispositivo multimedia son guardadas en una matriz extendida de control de acceso. Las filas representan usuarios y las columnas muestran las cdevs y pdevs pertenecientes a la estación de trabajo controlada por el servidor del switchboard. Una entrada única a esta matriz consta de dos partes: atributo y operación.

Por ejemplo, un dispositivo de audio que tiene la capacidad de almacenar datos en una variedad de formatos de muestreo, con calidad CD e inferiores. Sería deseable que unos pocos usuarios tuvieran acceso a la calidad CD, porque ocupa mucho espacio y es más costoso grabarlo.

La otra entrada de la matriz de control de acceso es una lista de las posibles operaciones que se permiten con una pdev. Por ej.: un usuario puede estar autorizado a capturar información con una cámara, pero no a moverla. Este puede ser el caso en que la cámara está instalada en un cuarto seguro al cual el usuario no tiene acceso.

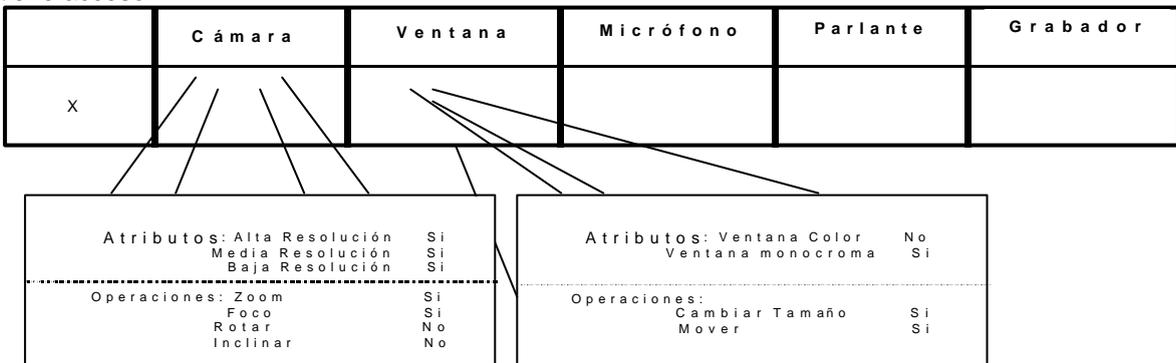


Figura 8.D.3. matriz de acceso en protección estática

La figura 8.D.3. muestra un ejemplo de protección estática. El usuario X tiene acceso a una cámara y a una ventana como parte de un sistema de ventanas capaz de mostrar imágenes de vídeo. X puede usar la cámara en tres modos posibles: alta, media y baja resolución. Puede hacer zoom y enfocarla, pero no puede moverla. Puede mostrar solo secuencias monocromo en la ventana porque es más barato para el sistema y puede mover y modificar el tamaño de las ventanas.

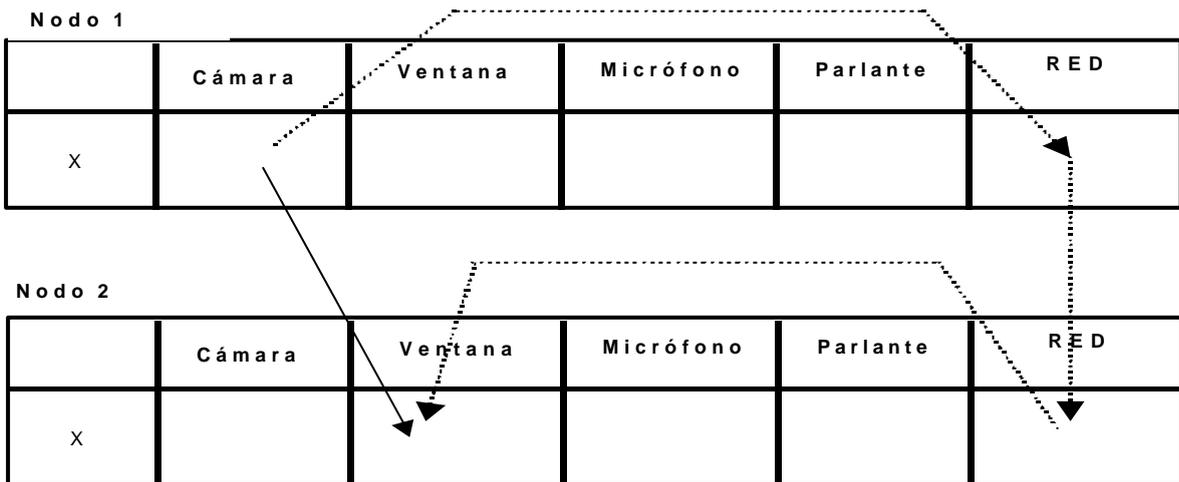


FIGURA 8.D.4. Protección dinámica en la red.

Seguridad Dinámica

Es la que se encarga de conexiones seguras entre los pdevs, para que puedan compartir información entre si. Las operaciones posibles de la matriz se extienden con permisos de conexión.

Debido a esta definición, el atributo conexión refleja el acceso combinado de lectura/escritura para un pdev pero como cada uno tiene diferentes características el servidor debe interpretarlas correctamente. En el ejemplo anterior se debe incluir la operación de conexión para mostrar en la ventana los datos captados por la cámara.

Como en la matriz solo se muestra si los elementos están conectados unos a otros, el servidor debe chequear la semántica, es decir si un pdev está conectado a otros y la conexión es válida (cámara a display, micrófono a parlante, etc.). Todo flujo inválido es rechazado por el sistema.

Para el traslado de la información en la red se hace que cada nodo se comporte como un pdev y la capa de transporte se encarga de esta tarea.

En la figura se ve lo siguiente, la cámara que está en el nodo 1 envía datos a la ventana que los muestra por pantalla cuya ubicación es el nodo 2. Físicamente esto se hace conectando los pdev de los nodos 1 y 2 en la red. La línea continua representa el flujo de datos lógico y la línea de trazos representa el flujo de datos actual. Este consiste en tres flujos: dos internos a los nodos y un flujo de red entre los dos nodos. Es tarea de cada servidor el chequear si el usuario X tiene los permisos para acceder a los datos de cada nodo. Cabe destacar que en las figuras 8.D.3 y 8.D.4. estamos hablando de un sistema multiusuario.

Autenticación

Además de los esquemas mencionados, un mecanismo adecuado de autenticación contribuye a la seguridad del switchboard. Si se está en una teleconferencia es una medida correcta el no permitir el ingreso a curiosos.

La autenticación se hace al comienzo de la sesión. Luego de la validación se puede arrancar el set-up de la sesión. La identificación del lado del cliente es el nombre de usuario que empezó la comunicación y el servidor se identifica con el nombre del nodo en donde se encuentra.

La autenticación de dos servidores en el establecimiento de uno o más flujos de red es disparada por el lado del cliente del switchboard en el primer intento de conexión entre ambos. Si los identificadores son correctos comienza la comunicación, ya que existe confianza mutua entre los servidores. Luego de que el último flujo entre los servidores ha sido cerrado, debe autenticarse nuevamente si se desea comenzar otra comunicación.

ANEXO 8.E. PASOS ESENCIALES PARA EL HACKING

Nota: este anexo se introduce a los efectos de protegerse de las acciones de un hackeo.

El intruso o Hacker conseguirá el objetivo mediante la siguiente serie de pasos:

1º) PASO: RECOPIACION DE INFORMACION

Un intruso para poder llegar a nuestra red deberá recopilar suficiente información de la misma para que el ataque sea concreto y preciso (es decir, que no sea detectado inmediatamente), para ello usaran herramientas que busquen información sobre los aspectos relativos a la política de seguridad seguida por nuestra organización.

Por ejemplo debe identificar:

EN EL CASO DE INTERNET:	EN EL CASO DE UNA INTRANET
<ul style="list-style-type: none"> • Nombre de dominio • Bloques de Red • Direcciones IP • Servicios • Arquitectura del Sistema • Mecanismo de Control de Acceso • Sistema de Detección de Intrusos • Nombres de Administradores del Sistema, usuarios y grupos. 	<ul style="list-style-type: none"> • Protocolos de Red • Nombres de Dominio interno • Bloques de Red • Direcciones IP específicas de Sistemas disponibles vía Internet. • Arquitectura del Sistema • Etc.

2º) PASO: EXPLORACION

El segundo paso que realizara un intruso para introducirse a nuestro sistema será la exploración del mismo, es decir obtendrá cuales son los sistemas activos y cuales son accesibles a través de Internet, mediante el uso de una variedad de técnicas y herramientas tales como barridos de ping, exploración de puertos y herramientas de búsqueda automatizada.

3º) PASO: ENUMERACION

Lo siguiente que hará un intruso o Hacker será intentar identificar cuentas de usuarios validas o recursos compartidos mal protegidos. Existen varias formas de extraer nombres de recursos exportados o cuentas validas de un sistema, a este proceso se lo denomina enumeración.

La principal diferencia existente entre las técnicas de recopilación de información vista previamente y la enumeración es el grado de intrusión; la enumeración implica la conexión activa a los sistemas y la

realización de consultas dirigidas. Como consecuencia, estas operaciones deberían ser registradas o al menos detectadas.

La información que el intruso querrá obtener será la siguiente:

- Recursos de red y recursos compartidos
- Usuarios y Grupos
- Aplicaciones y Mensajes

Las técnicas de enumeración son específicas de cada S.O y utilizarán la información recopilada en el paso 2.

MAPA DE VULNERABILIDADES

La generación del mapa de vulnerabilidades es el proceso por el que se relacionan ciertos atributos de la seguridad de un sistema con una determinada vulnerabilidad asociada o potencial. Se trata de una fase crítica del ataque real sobre un sistema objetivo que no debería pasarse por alto. Los atacantes necesitan conocer ciertos atributos tales como servicios de escucha, números específicos de versión de los servidores en ejecución (por ejemplo: Apache utilizando en HTTP y sendmail utilizado para SMTP), arquitectura del sistema e información sobre los nombres de los usuarios, para detectar posibles agujeros en la seguridad del sistema. Existen varios métodos que los atacantes pueden utilizar:

- Identificar manualmente los atributos específicos a partir de fuentes públicamente disponibles de información sobre vulnerabilidades, tales como Bugtraq, consejos de Computer Emergency Response Team (www.cert.org) y alertas sobre seguridad emitidas por los fabricantes. Aunque es un proceso tedioso, puede proporcionar un completo análisis de potenciales vulnerabilidades, sin necesidad de violentar realmente el sistema.
- Usar programas públicos de explotación publicados en diversas listas de correos sobre seguridad, en sitios WEB o desarrollar sus propios programas. Este procedimiento determinará, con un elevado grado de certeza, la existencia de una vulnerabilidad real.
- Utilizar herramientas automáticas de rastreo para identificar vulnerabilidades verdaderas. Entre las herramientas comerciales de este tipo, con prestigio, se incluyen: Internet Scanner de Internet Security System o CyberCop Scanner de Network Associates.

Todos estos métodos tienen sus ventajas e inconvenientes; sin embargo, es importante recordar que solamente los atacantes con poca información evitarán esta fase de generación del mapa de vulnerabilidades, lanzando un ataque a ciegas contra el sistema para intentar introducirse en él.

ACCESO REMOTO FRENTE A ACCESO LOCAL

El acceso remoto se define como la obtención de acceso a través de la red o mediante otro canal de comunicación. El acceso local es el acceso que se realiza mediante un Shell de comandos real o un inicio de sesión directo en el sistema.

Es importante entender la relación existente entre acceso remoto y local. Existe una progresión lógica cuando los atacantes, de manera remota, explotan una vulnerabilidad en un servicio de escucha y, posteriormente, obtienen acceso al Shell local. Una vez obtenido el acceso al Shell se considera que los atacantes están en "Local" frente al sistema.

Acceso remoto

Como se ha mencionado anteriormente, el acceso remoto implica acceso directo a la red o acceso a través de otro canal de comunicaciones, tal como un módem de acceso telefónico unido a un sistema Unix. Para ello, limitaremos nuestro análisis al acceso al sistemas UNIX mediante redes TCP/IP. Después de todo TCP/IP, es la piedra angular de Internet y, por ello, resulta de la máxima importancia en nuestro análisis sobre la seguridad de UNIX.

Existen tres métodos principales para burlar de forma remota la seguridad de un Sistema Unix:

1. Explotación de un servicio de escucha (por ejemplo, TCP/UDP).
2. Introducirse a través de un dispositivo UNIX (enrutamiento) que proporcionan seguridad entre dos o más redes.
3. Ataque de ejecución remota iniciados por el usuario (caballos de Troya, sitios web hostiles).

Veamos unos cuantos ejemplos que nos permitirán conocer los diferentes tipos de ataque que entran dentro de las categorías anteriores.

- Explotación de un servicio de escucha: Es imperativo recordar que se necesita de un servicio a la escucha para conseguir el acceso.
- Introducirse a través de un sistema Unix: En muchos casos los atacantes burlan al Firewall mediante el envío de paquetes de enrutamiento que acceden a través del Firewall a los sistemas internos.
- Ejecución Remota iniciada por el usuario: Esto se debe a que el usuario pueda navegar por sitios hostiles al sistema.

Ataques de fuerza bruta

La forma de ataque mas básica es a través de la adivinación de la contraseña por la fuerza bruta. Este sistema consiste en simplemente en adivinar la contraseña y el ID del usuario para acceder al sistema.

Los servicios más comunes son:

- Telnet
- Protocolo FTP
- Los comandos "r" (rlogin,rsh, etc)
- Secure Shell (ssh)
- Protocolo de oficina de correos (POP)
- Protocolo de Transporte de Hipertexto (HTTP/HTTPS)

Una vez que los atacantes han conseguido una lista de cuentas de usuario, intentaran acceder al Shell del sistema adivinando la contraseña asociada con una de las ID. Desgraciadamente, muchas cuentas de usuarios tienen contraseñas sencillas o no tienen contraseñas en absoluto.

Aunque es posible tratar de adivinar las contraseñas manualmente existen herramientas que lo hacen en forma automática.

CONTRAMEDIDA PARA LA FUERZA BRUTA

La mejor defensa contra este tipo de ataque es utilizar una buena contraseña que sea difícil de adivinar.

Por ejemplo:

- Asegúrese de que todos los usuarios tengan contraseñas.
- Obligue a un cambio de contraseñas cada 30 días para las cuentas privilegiadas y cada 60 días para los usuarios normales.
- La longitud mínima de la contraseña debería ser de 6 caracteres alfanuméricos o preferiblemente 8.
- Registre los intentos fallidos múltiples de autenticación.
- Configure los servicios para proceder a la desconexión del sistema después de tres intentos fallidos.
- Desactive los servicios que no use.
- Utilice herramientas de generación de contraseñas que impidan que el usuario escoja una contraseña fácil de adivinar.
- No utilice las mismas contraseñas en todos los sistemas en los que pueda iniciar una sesión.
- No escriba nunca su contraseña en una hoja.
- No diga su contraseña a otros.
- Verifica que ciertas cuentas predeterminadas, tales como Administrador, Admin etc. No utilicen contraseñas predeterminadas.

ATAQUES ORIENTADOS A DATOS

Un ataque orientado a Datos se ejecuta enviando datos a un servicio activo que produce resultados indeseados o inesperados. Estos ataques se dividen en ataques de desbordamiento de buffers y ataques con validación de entrada.

1) ATAQUES CON DESBORDAMIENTO DE BUFFERS

Se produce un desbordamiento del buffer, cuando un usuario o un proceso intenta introducir en el buffer (o array fijo) mas datos de los originalmente permitidos. Este tipo de conductas esta asociada a determinadas funciones de C tales como strcpy(), strcat(), sprintf(), entre otras. Una condición de desbordamiento de buffer provocara, normalmente, una violación de la segmentación. Sin embargo, se puede sacar partido a este comportamiento para conseguir acceder fraudulentamente al sistema ya que al producirse el desbordamiento los atacantes pueden crear un ejecutar un código de su elección (Por ejemplo: provocar un desbordamiento al comando VRFY del sendmail mientras se este ejecutando la cuenta de administrador, luego puede acceder a /bin/sh ejecutando un pequeño programa).

Contramedida para el ataque de desbordamiento de buffers

b) practicas de codificación seguras

La mejor contramedida para el desbordamiento de buffer es utilizar prácticas de programación seguras. Aunque es imposible diseñar y escribir el código de un programa que este completamente libre de errores, existen algunas medidas que ayudan a minimizar las condiciones que producen el desbordamiento del buffer.

- Desde el principio, diseñe los programas teniendo en cuenta la seguridad.
- Emplee compiladores seguros.
- Valide los argumentos cuando se reciben de un usuario o de un programa.
- Utilice rutinas seguras tales como fgets(), strncpy() y strncat() y compruebe los códigos de retorno de las llamadas al sistema.
- Reduzca la cantidad de código que se ejecuta con privilegios de la cuenta root.

- Especialmente, aplique los parches del fabricante, relacionados con la seguridad.

b) Probar y auditar cada programa

Unos de los mejores ejemplos para probar y auditar código UNIX es el proyecto OPENBSD. OPENBSD audita continuamente su código fuente y ha solucionado cientos de condiciones de desbordamiento de buffer, sin mencionar muchos otros tipos de problemas relacionados con la seguridad.

c) Desactivar servicios sin uso o peligrosos

Desactive los servicios sin uso o peligrosos si no son esenciales para el sistema UNIX. Los intrusos no se pueden infiltrar a través de un servicio que no se encuentre en ejecución.

d) Desactivar la ejecución de la pila

Este procedimiento tiene pocos efectos secundarios y protege a muchos sistemas de todo tipos de ataques.

2) ATAQUES DE VALIDACION DE ENTRADA

Un ataque con validación de entrada se produce cuando:

- Un programa no reconoce una entrada sintácticamente incorrecta.
- Un modulo acepta una entrada extraña.
- Un modulo no puede manejar campos de entrada perdidos.
- Se produce un error de correlación de valor de campo.

Contra medida para la validación de entrada

Como se menciona anteriormente, la mejor medida de seguridad preventiva es la utilización de prácticas de codificación seguras y este concepto es también valido para los ataques con validación de entrada. Es absolutamente crítico asegurarse de que los programas y los archivos de comandos acepten únicamente los datos que se supone que tienen que recibir, y que rechazan todo los demás.

TELNET INVERSO y CANALES TRASEROS

Definimos canal trasero (back channel) como un mecanismo donde el canal de comunicación se origina en el sistema objetivo en lugar de originarse en el sistema atacante. Como los atacantes tienen que originar una sesión desde el servidor Unix vulnerable hacia el sistema de los atacantes mediante la creación de un canal trasero deben utilizar un método, el método mas conocido es el de Telnet inverso. Se denomina Telnet inverso porque la conexión telnet se origina en el sistema al que los atacantes están intentando acceder, en lugar de originarse en el sistema de los atacantes.

3) CONTRAMEDIDA PARA EL CANAL TRASERO

Es muy difícil protegerse contra los ataques de canal trasero. Preferentemente se debe desactivar los servicios innecesarios y aplicar los parches del distribuidor y de los programas complementarios relacionados.

Otro punto a tener en cuenta son: eliminar X de cualquier sistema que requiera un alto nivel de seguridad, de esta forma los atacantes no pueden usar un ataque usando xterm.

SERVICIOS FRECUENTEMENTE ATACADOS

Algunos de los servicios que se atacan con mayor frecuencia son:

- TFTP
- SENDMAIL
- NFS
- SISTEMAS DNS
- FTP
- RPC
- X

TFTP (Trivial Protocolo de Transferencia de Archivos), se utiliza típicamente para arrancar estaciones de trabajos sin disco o dispositivos de red tales como routers. TFTP es un protocolo basado en UDP que escucha en el puerto 69 y es muy poco seguro. En muchas ocasiones los atacantes localizaran un sistema con un servidor TFTP activado e intentaran conseguir mediante TFTP una copia del archivo /etc/passwd. Luego de estos los atacantes dispondrán de una lista de los nombres de los usuarios que pueden ser objeto de un ataque de fuerza bruta.

Las ultimas versiones de TFTP se configuran en forma predeterminada para prohibir el acceso a cualquier directorio excepto a /tftpboot. Esto es una muy buena medida pero sin embargo los atacantes pueden obtener cualquier archivo de ese directorio (como ser los archivos de la configuración del router).

Contra medidas para TFTP

Asegúrese de que el servidor TFTP este configurado para restringir el acceso a determinados directorios como /tftpboot. De esta forma evitara que los atacantes se apoderen de los archivos de configuración del sistema.

FTP o File Transfer Protocol (Protocolo de Transferencia de archivos), es uno de los protocolos más usado actualmente. El mismo permite bajar o subir archivos desde y hacia sistemas remotos. Muchos servidores

permiten un acceso anónimo, permitiendo a cualquier usuario iniciar una sesión en un servidor FTP sin necesidad de autenticación. Normalmente el sistema de archivos al que se le permite el acceso se limita a una rama en particular del árbol de directorios. Sin embargo, en otras ocasiones, el servicio de FTP anónimo permite al usuario moverse por la estructura de directorios completas. Por lo tanto sin seguridad, los atacantes pueden colocar un archivo `.rhosts` en el directorio `home` de un usuario, permitiendo a los atacantes hacer un rlogin al sistema atacado.

Contra medidas para FTP

Aunque FTP es muy útil, permitir acceso anónimo a FTP puede resultar peligroso para nuestro servidor. Evalúe la necesidad de ejecutar un servidor FTP y decida si tiene que permitir el acceso anónimo a su servicio de FTP. Si realmente lo necesita reduzca el número de directorios de escritura disponibles.

SENDMAIL es un agente de transferencia de correo (MTA) que se utiliza en muchos sistemas Unix. Sendmail es uno de los programas más malignos que puede encontrar. Es extensible, configurable y realmente muy complejo. Aunque la seguridad de sendmail ha mejorado en estos últimos tiempos todavía queda algunos problemas de seguridad.

Además, de los ataques de desbordamiento de buffer y de validación de entrada es muy posible aprovechar la funcionalidad de sendmail para conseguir un acceso privilegiado. Un ataque común sería crear o modificar un `~/forward` del usuario mediante FTP o NFS, suponiendo que los atacantes disponen de privilegios de escritura en el directorio `home` de la víctima.

Normalmente, un archivo `~/forward` dirige el correo a una cuenta diferente o ejecuta algún programa cuando llega un correo. Obviamente los atacantes pueden modificar el archivo `~/forward` para conseguir sus propósitos.

Contra medidas para sendmail

La mejor defensa contra los ataques con sendmail es desactivarlo, si no lo está usando para recibir correo en una red. Si está utilizando sendmail asegúrese que se la última versión con todos los parches disponibles. También puede considerar un MTA más seguro como qmail, un moderno sustituto de sendmail.

SERVICIOS DE LLAMADAS A PROCEDIMIENTOS REMOTOS

Remote Procedure Calls (RPC), llamada a procedimiento remoto es un mecanismo que permite a un programa que este corriendo en un equipo ejecutar código, de forma no perceptible, en un sistema remoto. En los sistemas Unix muchos de estos servicios se ejecutan en modo root y además son muy complejos. Por ello, un ataque con éxito de desbordamiento de buffer o con validación de entrada producirá un acceso directo al sistema como administrador.

Contra medidas para los servicios de RPC

La mejor defensa contra los ataques RPC remotos es desactivar todos los servicios RPC que no sean absolutamente necesarios. Considere también la utilización de Secure RPC si lo admite su sistema Unix, Secure RPC intenta proporcionar un nivel de autenticación adicional utilizando cifrado basada en clave pública.

NFS Network File System, es uno de los sistemas de archivos con capacidad de red más populares. NFS permite acceder a de forma transparente a los archivos y directorios de sistemas remotos como si estuvieran almacenados localmente. Un ataque contra un NFS es del tipo más común en Sistemas Unix. En primer lugar, se han descubierto muchas condiciones de desbordamiento de buffer que guardan relación con `mountd`, el servidor NFS. Además, NFS se apoya en los servicios RPC que se puede engañar con facilidad permitiendo que los atacantes monten un sistema de archivos remotos. La mayor parte de la seguridad proporcionada por NFS está relacionado con un objeto de datos conocido como el manejador de archivos (`file handle`), el manejador de archivos es una señal utilizada para identificar de forma única cada archivo y directorio contenido en el servidor remoto. Si el manejador de archivos puede ser adivinado, los atacantes remotos podrían acceder fácilmente a los archivos que se hallan en el sistema remoto.

El tipo más común de vulnerabilidad NFS se relaciona con una mala configuración que exporta el sistema de archivos a `everyone` (todos). Es decir, cualquier usuario remoto puede montar el sistema de archivos sin tener que autenticarse, este error puede provenir por un descuido del administrador del sistema.

Contra medidas NFS

Si no son necesarios debe desactivarse tanto NFS como los servicios relacionados (por ejemplo `mountd`, `statd` y `lockd`). Utilice controles de acceso de aplicaciones clientes y usuarios para permitir únicamente el acceso a los archivos a los usuarios autorizados.

INSEGURIDADES X

El sistema X Window dispone de una gran variedad de funciones que permiten que muchos programas compartan una única pantalla gráfica. El problema más importante con X es que todo su modelo de seguridad se basa en todo o nada. Una vez que se le ha otorgado el permiso a un cliente para acceder a un servidor X, todo se le permite. Los clientes X pueden capturar pulsaciones de teclas de las consolas, cerrar ventanas, capturar ventanas para mostrarla en cualquier sitio e, incluso, reasignar el teclado. La forma más sencilla, y más popular, de controlar el acceso X es la autenticación xhost, muchos sistemas ejecutan en forma predeterminada xhost + (el + es un carácter comodín que equivale a cualquier dirección IP) permitiendo de esta forma a cualquier usuario local o remoto acceder al servidor X. Por lo tanto de esta forma puede comprometer seriamente la seguridad del servidor.

Contramedidas para X

No usar el comando xhost +. Otras medidas de seguridad incluyen utilizar mecanismo de autenticación más avanzados como MIT-MAGIC-COOKIE-1, XDM-AUTHORIZATION-1 y MIT-KERBEROS-5. Estos mecanismos proporcionan un nivel de seguridad adicional cuando se conectan a un servidor X. Si utiliza xterm o un terminal similar, active la opción secure Keyboard (teclado seguro). De esta forma, impedirá que cualquier otro proceso intercepte sus pulsaciones de teclas. Finalmente, piense seriamente en instalar un Firewall en los puertos 6000-6063, para evitar que los usuarios no autorizados se conecten a sus puertos de servidor X.

SISTEMAS DNS

DNS es uno de los servicios más populares utilizados en Internet y en la mayoría de las intranets empresariales. Como se puede imaginar, la ubicuidad de DNS le hace muy vulnerable a los ataques. La mayoría de los atacantes buscan de forma rutinaria las vulnerabilidades existentes en la implantación más frecuentes de DNS para Unix, el paquete Berkeley Internet Name Domain (BIND). Además, DNS es uno de los pocos servicios que es casi siempre necesario y que se ejecuta en cualquier red perimetral de Internet de cualquier empresa.

Contramedidas para el DNS

En primer lugar y, sobre todo, desactive y elimine BIND en cualquier sistema que no se utilice como servidor DNS. En segundo lugar, deberá asegurarse que la versión de BIND que este utilizando se encuentre actualizada y que se utilicen los parches adecuados para corregir los defectos de seguridad. En tercer lugar ejecuta named como un usuario sin privilegios.

AUTOEVALUACIÓN DEL MODULO 8:

Preguntas:

1. Explique la diferencia entre protección y seguridad
2. Describa política de seguridad
3. ¿Cuáles son las diferentes amenazas a la seguridad?
4. ¿Cuál es el concepto de Auditoria referida a la seguridad?
5. Describa métodos de ocultamiento de datos
6. Defina el concepto de Dominio de Seguridad
7. Describa brevemente el concepto de seguridad de los datos basado en transacciones
8. ¿En que consiste la validación de usuarios?
9. Explique brevemente el concepto de virus.
10. ¿Cuáles son los diferentes modelos formales de protección?
- 11.- ¿Cuáles son las causas y acciones más comunes de la perdida de datos?
- 12 ¿Que es un firewall?
- 13.- ¿Qué tipos de seguridad existen?
- 14.- ¿Cómo reconoce a una persona un dispositivo biométrico que identifica por huellas digitales? Beneficios.
- 15.- ¿Cuáles son las etapas de un virus?

Múltiple Choice:

<p>1.- Dentro de los Requerimientos de Seguridad, en cuanto a la privacidad, solo las personas autorizadas....</p> <p>a) pueden modificar las distintas partes del sistema. b) pueden disponer del sistema c) pueden acceder al sistema d) deben tener acceso a los componentes del sistema e) todas las anteriores f) ninguna de las anteriores</p>	<p>2.- Los ataques por Software son perpetrado por....</p> <p>a) Virus b) Mal funcionamiento de la CPU c) Gusanos d) Discos o cintas ilegibles. e) Falta de un adecuado mantenimiento. f) Todas las anteriores g) Ninguna de las anteriores</p>
<p>3.- Las políticas de seguridad se caracterizan por....</p> <p>a) Determinar como hacer algo. b) Determinar que se hará. c) Poder cambiar de tiempo en tiempo. d) Asegurar la operación eficiente y ordenada del sistema. e) Aplicar las políticas que gobiernan la utilización de los recursos. f) Todas las anteriores. g) Ninguna de las anteriores</p>	<p>4.- El S. O. es el responsable de la Seguridad Interna debido a que....</p> <p>a) los mecanismos actúan bajo demanda b) permite compartir los recursos. c) provee una gran variedad de mecanismos d) interfiere con el normal trabajo del usuario e) es difícil de usar en el instante que se necesite. f) Todas las anteriores g) Ninguna de las anteriores</p>
<p>5.- La Intercepción afecta a</p> <p>a) La disponibilidad. b) La autenticidad c) La integridad d) La privacidad. e) Todas las anteriores</p>	<p>6.- La Seguridad física externa al centro de procesamiento se ocupa</p> <p>a) Del mantenimiento. b) Del abastecimiento esencial. c) De los intrusos. d) De los sabotajes.</p>

f) Ninguna de las anteriores	e) De los desastres. f) Todas las anteriores. g) Ninguna de las anteriores.
7.- Los ataques de intrusos en la red o Internet se conoce como.... a) Falsificación. b) Penetración. c) Modificación de archivos sin protección. d) Robo de contraseñas e) Bomba de tiempo f) Todas las anteriores. g) Ninguna de las anteriores.	8.- El Sistema de Seguridad para los datos se implementa para... a) Prever accesos no autorizados a los sistemas b) Prevenir la destrucción maliciosa o alteración de los datos. c) Determinar una correcta autenticación de los usuarios del sistema d) Proteger la integridad de la información almacenada en el mismo. e) Todas las anteriores. f) Ninguna de las anteriores.
9.- La denegación de servicios a usuario legítimos..... a) Una forma de denegación de servicio viene representada por los programas que se autoproducen y propagan, llamados virus informáticos b) Una forma de denegación de servicio viene representada por los programas llamados caballo de troya c) Una forma de denegación de servicio viene representada por los programas que se autoproducen y propagan, llamados gusanos informáticos d) Da lugar a la pérdida parcial o completa del servicio prestado a los clientes legítimos. e) Todas las anteriores. Ninguna de las anteriores.	10.- Un dominio de protección se caracteriza porque.... a) Un proceso opera con un dominio de protección que especifica los recursos a los que pueden acceder y usar. b) Cada dominio define un conjunto de objetos y los tipos de operaciones que se pueden realizar en cada uno de ellos. c) La posibilidad de ejecutar una operación en un objeto es un derecho de acceso. d) Un dominio es una colección de derechos de accesos, cada uno de ellos es un par ordenado (nombre del objeto, conjunto de derechos) e) Todas las anteriores. f) Ninguna de las anteriores.
11.- Los mecanismos de seguridad se caracterizan por... a) Determinan como hacer algo. b) Pueden cambiar de tiempo en tiempo. c) Aseguran la operación eficiente y ordenada del sistema. d) Permite aplicar las políticas que gobiernan la utilización de los recursos. e) Refuerzan a las políticas. f) Todas las anteriores. g) Ninguna de las anteriores.	12.- Cuál de los siguientes principios generales para los mecanismos de sistemas seguros han sido propuesto por Saltzer y Schroeder... a) Diseño Cerrado. b) Gran cantidad de mecanismos c) Inaceptabilidad d) Acciones post demanda e) Todas las anteriores f) Ninguna de las anteriores
13.- El Control de acceso obligatorio (CAO) se caracteriza por... a) Las políticas son generalmente definidas por el propietario de los datos quien puede transferir derechos de acceso a otros usuarios. b) El creador de un archivo puede especificar los derechos de acceso de los usuarios. c) Esta forma de control de acceso es habitual en sistemas de archivos. d) Es vulnerable al ataque del Caballo de Troya e) Todas las anteriores. f) Ninguna de las anteriores.	14.- La criptografía proporciona un método a) Que se emplean habitualmente para resolver el problema de la validación. b) Para aumentar la confianza en el secreto de la información en tránsito y en almacenamiento. c) Que trata con dominios o filas de la matriz de acceso. d) Basado en el secreto de la clave y no del algoritmo. e) Basado en el secreto del algoritmo y no de la clave. f) Todas las anteriores g) Ninguna de las anteriores
15.- El Control de Virus se previenen mediante normas y procedimientos para riesgos de origen..... a) Físico b) Humano c) Técnico d) Catástrofes climáticas e) Personas externas al sistema f) Todas las anteriores g) Ninguna de las anteriores	16.- La(s) Ventaja(s) de la Tabla global a) Tabla es muy grande para guardarla en memoria central. b) Si está en disco hay E/S c) Difícil agrupar objetos o dominios con características similares. d) Sencilla de implantar. e) Todas las anteriores f) Ninguna de las anteriores

<p>17.- El Cifrado de los datos ...</p> <ul style="list-style-type: none">a) Garantiza que la información no sea inteligible para personas no autorizados.b) Garantiza que la información sea inteligible entidades o procesos no autorizados.c) Consiste en transformar un texto claro en un texto codificado mediante un proceso de cifradod) Consiste en transformar un texto claro en un texto codificado mediante la utilización de claves de cifrado.e) Todas las anterioresf) Ninguna de las anteriores	<p>18.-Una Lista de capacidades (capability) se caracteriza porque....</p> <ul style="list-style-type: none">a) El orden está dado por las filas de la matriz, donde cada fila es para un dominio.b) El orden está dado por las columnas de la matriz, donde cada columna es para un dominio.c) Para un dominio es una lista de objetos, y las operaciones permitidas sobre éste.d) Para un dominio es una lista de dominios, y los procesos permitidos sobre éste.e) Un objeto, en general, está representado por su nombre o dirección (llamado capacidad).f) Todas las anteriores.g) Ninguna de las anteriores.
<p>19.- Las contraseñas (password) tienen como ventajas....</p> <ul style="list-style-type: none">a) Que no requieren un hardware especialb) Que son relativamente fáciles de implementar.c) Ofrecen protección limitada porque pueden ser relativamente fáciles de adivinar o de obtenerd) Si son palabras del diccionario o nombres propios son fáciles de recordar pero también de adivinar.e) Todas las anterioresf) Ninguna de las anteriores	<p>20.- Los gusanos informáticos ...</p> <ul style="list-style-type: none">a) Se adhieren a otros programasb) Son programas ejecutables autocontenidos que se diseminan a través de redes informáticas.c) Después de un período de letargo y propagación provocan algún tipo de daño en la máquina anfitrión.d) Pueden provocar infecciones que consumen cantidades enormes de recursose) Niegan servicio a los usuarios legítimos.f) Todas las anterioresg) Ninguna de las anteriores

Respuestas a las preguntas**1. Explique la diferencia entre protección y seguridad**

Cuando hablamos de protección hacemos referencia a un mecanismo por el cual se intenta asegurar la integridad del sistema y sus recursos. El mecanismo controla los accesos al sistema tanto de procesos, como de usuarios.

Por otra parte la seguridad es el nivel de confianza en los mecanismos de protección. Y la capacidad que estos tienen para responder a los ataques y fallas manteniendo la integridad del sistema y sus datos.

2. Describa política de seguridad

Una política determina que es lo que se hará, dejando el como a los mecanismos. Aseguran la correcta operación del sistema. Pueden variar a lo largo del tiempo, y los cambios en las políticas pueden conllevar cambios en los mecanismos.

Las políticas son implementadas a través de procedimientos escritos, en los que se establece entre otras cosas, la manera de introducir y sacar información del sistema, los niveles de autorización para acceder a los datos, como fluye la información en el sistema, y los límites establecidos.

Las políticas se basan en principios como, el mínimo privilegio, en el que cada usuario accede a la información mínima requerida para su trabajo; separación de deberes, que establecen controles cruzados para el manejo de información; rotación de roles, en el que las operaciones mas sensibles de un sistemas no deben ser realizadas siempre por el mismo grupo de personas.

3. Describa brevemente diferentes amenazas a la seguridadRevelación no autorizada de información

Constituyen una violación a la privacidad, su efecto depende de la naturaleza de la información, pudiendo llegar a constituir una gran amenaza.

Alteración o destrucción no autorizada de información

Puede derivar en pérdidas de información irrecuperable, por lo que resulta potencialmente peligroso.

Uso no autorizado de servicios

Las principales consecuencias de esta penetración, son la posibilidad de llegar a acceder a la información, lo que puede redundar en pérdidas de prestigio del proveedor de los servicios.

También puede generar pérdidas económicas al proveedor, debido a la utilización no controlada de sus servicios.

Denegación de servicios a usuarios legítimos

Esto genera una pérdida total o parcial de funcionalidad en los usuarios legítimos, un ejemplo concreto de esto son los denominados gusanos, que cuentan con la posibilidad de reproducirse y propagarse.

4. ¿A qué se refiere el concepto de Auditoria referida a la seguridad?

Las auditorias son pruebas o ataques intencionales al sistema para verificar que este responda de la manera esperada, o para validar una corrección introducida en los niveles de seguridad.

Las auditorias pueden llevarse a cabo de manera manual o automática.

Una de las herramientas utilizadas para las auditorias es el uso de un archivo log que registra los eventos que se produjeron en el sistema, lo que presenta un nuevo problema de seguridad, ya que hay que protegerlo de accesos no autorizados.

5. Describa métodos de ocultamiento de datos

La criptografía se basa en la idea de tomar los mensajes generados y aplicarles algún tipo de transformación, para luego ser almacenados o transferidos.

Principalmente se trata a los datos con un método y una clave que no pueda ser identificada por el intruso.

Algunos de los métodos de sustitución mencionados son los que siguen:

Enciframiento de Cesar

Este método incrementa cada letra del alfabeto en una cantidad entera n , con lo que al recibir los datos se debe aplicar el método inverso para poder utilizarlos.

Sustitución con palabra clave

Consiste en asignar a cada letra del alfabeto una nueva letra, de un nuevo alfabeto reordenado, que comienza por las letras de la palabra clave.

Transposición

Tanto el primer como el segundo método son de fácil violación.

Un método más complejo consiste en la permutación de las letras de la palabra, que puede llegar a hacerse a nivel de bits.

La ventaja de este método reside en que puede conocerse el método, pero al no conocerse la clave, resulta difícil su desciframiento. Este sistema ha llegado a convertirse en un standard de seguridad.

6. Defina el concepto de Dominio de Seguridad

Un dominio de seguridad se conoce como un conjunto de derechos de acceso, estos derechos representan las operaciones permitidas sobre un objeto determinado. Por lo cual, un usuario, dentro de un dominio determinado, sólo puede realizar sobre un objeto dado, las operaciones definidas en este dominio para ese objeto.

Este tipo de esquema puede ser implementado a través de una matriz de accesos.

La matriz de accesos define a través de sus filas y columnas, los dominios y objetos existentes en el sistema, y en cada celda de la matriz de establecen los derechos de accesos.

Una matriz de accesos puede ser implementada de diferentes formas, entre las que podemos contar:

Tabla Global

Una tabla global se establece por medio de una entrada por cada relación Dominio-Objeto, en la cual se especifican los derechos de accesos de ese dominio al objeto en cuestión.

Cuando un proceso perteneciente a un dominio, intenta acceder a un objeto, se busca en la tabla la entrada correspondiente y se verifica que el proceso cuente con los permisos adecuados.

Lista de accesos

La lista de accesos se define por cada objeto en el sistema (tantas listas como objetos haya), y se agrega en cada una de ellas, una entrada por cada dominio, en esta entrada se enumeran los derechos de acceso al objeto que ese esté tratando.

Lista de Capacidades

Este mecanismo es similar al de lista de accesos, pero se define una lista por cada dominio, en la que se registran todos los objetos, y los derechos del dominio sobre el mismo. Los derechos que el dominio tiene sobre un objeto determinado, se denominan capacidades.

Mecanismos Lock/Key

Consiste en otorgarle a cada objeto una serie de candados (bits), a los cuales se puede acceder por medio de las llaves (bits) adecuadas. Como contra partida se le otorga a cada dominio una serie de llaves que los habilitan para acceder a los objetos.

Un proceso perteneciente a un dominio determinado, sólo puede acceder a un objeto, si su dominio, cuenta con las llaves correspondientes a los candados del objeto.

7. Describa brevemente el concepto de seguridad de los datos basado en transacciones

Nos referimos a transacción como una serie de operaciones (read, write), sobre un archivo, que debieran de ser ejecutadas en forma atómica, para asegurar la integridad de los datos.

Por diversos motivos, como pueden ser fallas del sistema, puede que una transacción no complete su ejecución, siendo abortada. Esto puede producir datos inconsistentes, lo que genera un problema que hay que solucionar. Existen diferentes opciones para hacer esto, algunas de ellas son la utilización de un archivo log, o el establecimiento de múltiples check point, a los cuales se pueda retornar en caso de fallas, por medio de un roll back.

Archivo Log:

Este mecanismo requiere de la escritura en un archivo log, almacenado en disco, de cada operación de una transacción, antes de que la misma se ejecute. Esta entrada en el archivo guarda datos que permiten la recuperación del estado anterior de los datos. De esta manera se guarda el nombre del dato modificado, su valor actual y el anterior. Se guardan entradas en el log también por las operaciones de comienzo y fin de una transacción, así como el commit y el abort que puedan generar.

En el caso de un error se puede recuperar el estado anterior del sistema. Esto se paga con una baja en el rendimiento del sistema producto de las escrituras en el archivo log.

La utilización del archivo log permite operaciones de deshacer y rehacer.

Check Point:

Los métodos que utilizan check points, optimizan parte del overhead producido por la recuperación del sistema mediante el simple uso de un log.

El método requiere que cada determinada cantidad de tiempo, se baje a un almacenamiento no volátil todos los datos modificados, junto con una entrada "check point" en un log. De esta manera, si se produce una falla en el sistema, se ejecuta un roll back, que vuelve hasta el último check point ingresado en el archivo log. Se puede asegurar que todas las operaciones anteriores al checkpoint generaron datos consistentes.

8. Defina en que consiste la validación de usuarios

La validación de usuarios se refiere a la posibilidad del sistema de permitir o no el acceso según estos se identifiquen correctamente o no.

Existen diferentes medios para establecer esta validación, siendo el más común la proporción de una contraseña.

El principal problema de este método reside en los usuarios, estos muchas veces utilizan contraseñas fáciles de recordar, que a su vez resultan fáciles de adivinar por los intrusos.

Por otro lado la utilización de contraseñas proporcionadas por el mismo sistema resulta difícil de recordar, por lo que se genera una resistencia a su utilización.

Una técnica posible es aumentar a cantidad de caracteres que requiere una contraseña, dando la posibilidad de utilizar caracteres especiales, o mayúsculas y minúsculas de forma diferenciada.

Se pueden establecer validaciones adicionales una vez que el usuario logro entrar al sistema, pero esto genera un deterioro en la relación de los usuarios con el sistema.

Existen otros medios de validación de usuarios, sin contraseñas, por ejemplo la utilización de tarjetas electrónicas, o cintas magnéticas de diversa índole. Incluso se puede implementar una combinación de estos objetos, con claves de identificación.

Una tercer técnica consiste en validar rasgos personales del usuario, como puede ser una huella digital, la pupila del ojo, el tono de voz, etc.

9. Explique brevemente el concepto de virus.

Un virus es un fragmento de código, que logra mezclarse entre el código de un programa, de esta manera consigue generar copias de si mismo, y mezclarse dentro de otros programas que le permitan acceso. En un principio el virus se dedica solo a reproducirse, infectando todos los programas que se ejecutan después de su instalación. En determinado momento el virus comienza su acción destructiva. Estas acciones son muy variadas, y van desde las más inofensivas. A las que pueden causar una perdida total.

Las etapas por las que atraviesa un virus son: Letargo - Propagación - Activación - Daño.

Los ataques de los virus pueden estar dirigidos al sector de arranque del disco, al sistema operativo o a los archivos ejecutables, estableciendo distintos tipos de contaminación.

En general los virus más complejos son los que se alojan en archivos .EXE, y son también los más, difíciles de detectar. Los virus que atacan al sector de arranque, lo modifican para que se ejecute el código del virus y a continuación inicie el sistema operativo.

10. Describa diferentes modelos formales de protección

Los modelos formales de protección proporcionan los medios necesarios para una formulación precisa de políticas y mecanismos de seguridad. Entre estos modelos formales, podemos mencionar los siguientes:

Modelo de matriz de control de acceso

Este modelo es la formalización del esquema de matriz de accesos explicado más arriba. Se representa en una matriz a todos los objetos del sistema (entidades protegidas), y los sujetos con sus correspondientes derechos de acceso. Estos sujetos son considerados también objetos.

Modelo Tomar-Conceder

El modelo tomar-conceder, representa a través de un grafo la seguridad del sistema; representa un conjunto de sujetos (que a diferencia del anterior modelo, no los considera objetos), otro conjunto representando a los objetos, y un tercer conjunto para derechos de acceso genérico. A partir de estos elementos se establecen las relaciones necesarias.

El modelo resulta efectivo aun cuando el sistema experimenta un crecimiento ilimitado.

Modelo Bell-Lapadula

Este modelo de protección combina el modelo de matriz de accesos con la jerarquía de ordenación. se representa al sistema por un conjunto de sujetos, uno de objetos y una matriz de accesos. Las entradas de la matriz guardan derechos de accesos, y se le asigna a los sujetos un nivel de autorización y a los objetos un nivel de autoridad. Este modelo define un conjunto de operaciones primitivas.

Se implementan para el flujo de información las siguientes propiedades:

- Seguridad simple, que establece que un sujeto puede acceder para leer un objeto solo si este tiene una clasificación igual o inferior a su nivel de autorización.
- Propiedad, determina que un sujeto solo puede escribir en objetos cuya clasificación sea igual o superior a su autoridad.

Modelo retículo de flujo de información

Este modelo ve al sistema de seguridad como un conjunto de sujetos, objetos y clases de seguridad. Las asignaciones de seguridad se realizan en forma estática y pueden ser modificadas dinámicamente. Permite expresar abstractamente diferentes políticas de seguridad prácticas. Contribuye al control del flujo de información.

Se establece la seguridad del sistema exigiendo que todos los flujos estén autorizados.

11.- ¿Cuáles son las causas y acciones más comunes de la pérdida de datos?

Las causas más comunes de la pérdida de datos son:

- a) Actos naturales: como por ejemplo terremotos, incendios, movimientos inadecuados que provocan daños materiales en equipos o soportes de información.
- b) Defectos o fallas de hardware o de software: errores de telecomunicación o errores en el programa, que generan problemas a la seguridad.
- c) Intrusos: las acciones y motivaciones para penetrar en un sistema van desde el simple hecho de violar un mecanismo de protección por el solo hecho de violarlo, hasta un interés interno de robo o hurto de información en beneficio propio. Otro aspecto son los errores propios por descuidos o falta de conocimiento que pueden provocar daños de diversa índole.

Las acciones más comunes son:

- a) Torpes: Es imprescindible las consecuencias y los daños que pueden causar una mala acción de una persona poco habilidosa o deshonesto frente a un sistema.
- b) Curiosos: Generalmente ocurre cuando las computadoras están en red. Alguna persona pretenderá leer documentos o el correo electrónico de los demás o incluso archivos, si no existen barreras que impidan esta acción.
- c) Hackers: Son muy calificados y están dispuestos a invertir gran parte de su tiempo en este esfuerzo. Son excelente investigadores.
- d) Delincuentes informáticos (acciones ilegales): por ejemplo un intento deliberado por hacer dinero.
- e) Espías: Implica la captura de información transportada por cable o el uso de un sofisticado equipo para lograr el objetivo de apropiarse indebidamente de la información.

12 ¿Que es un firewall?

Un firewall para una red evita que los peligros de una red externa o internet se extiendan a una red interna.

Abarca los siguientes propósitos:

- Restringe el acceso a un punto cuidadosamente controlado.
- Evita que los atacantes se acerquen más a las defensas.
- Restringe a los usuarios para que salgan en un punto cuidadosamente controlado.

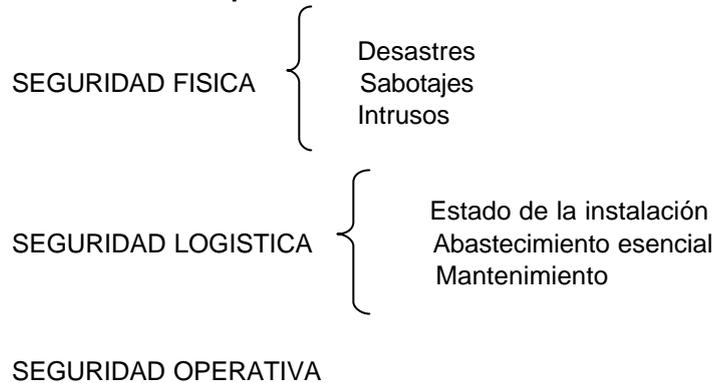
El punto donde se instala es donde la red interna protegida se conecta con internet, y controla todos los pedidos de servicios que entran desde el exterior y salen desde los usuarios, tales como, correo electrónico, transferencia de archivos, inicio de sesiones remotas, etc.

Por lógica un firewall es un separador, un limitador, un analizador. Su implementación física varía de una instalación a otra. Con mayor frecuencia consta de un conjunto de componentes de hardware (un enrutador, una computadora anfitrión, o cierta combinación de enrutadores, computadoras y redes con software apropiado).

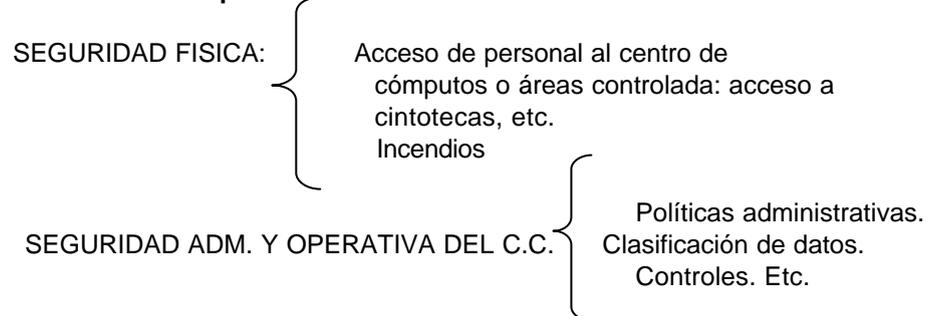
13.- ¿Qué tipos de seguridad existen?

Existen básicamente 2 grandes ámbitos para aplicar los criterios de seguridad:

- **Externa al centro de cómputos:**



- **Interna al centro de cómputos:**



Desde el punto de vista del Sistema Operativo, la seguridad de la información se basa en: - Seguridad computacional (computer security)

- Seguridad de redes computacionales (network security)

14.- ¿Cómo reconoce a una persona un dispositivo biométrico que identifica por huellas digitales? Beneficios.

Cuando un usuario se carga al sistema se toman y combinan múltiples imágenes de sus huellas dactilares, que se convierten en formato binario y el registro biométrico se almacena en memoria.

Luego de la conversión de la imagen original, esta es destruida. Una vez convertida en formato binario, el código no puede reconstruir la imagen dactilar con el fin de proteger la identidad de los usuarios.

Cada vez que un usuario intente acceder al sistema, coloca su dedo en el dispositivo que, toma y compara sus datos dactilares contra los almacenados anteriormente permitiéndole o no el acceso.

15.- ¿Cuáles son las etapas de un virus?

Durante su vida un virus típico pasa por las siguientes cuatro etapas:

1. Una fase latente, en la que el virus está inactivo. El virus será finalmente activado por algún suceso, como una fecha, la presencia de otro programa o archivo o que la capacidad de disco exceda de cierto límite. No todos los virus pasan por esta etapa.
2. Una fase de propagación, durante la cual el virus sitúa una copia idéntica suya en otros programas o en ciertas zonas del sistema en el disco. Cada programa infectado contendrá ahora un clon del virus, que entrará a su vez en la fase de propagación.
3. La fase de activación, en la que el virus se activa para llevar a cabo la función para la que está propuesto. Como en la fase latente, la fase de activación puede ser causada por una variedad de

sucesos del sistema, incluyendo la cuenta del número de veces que ésta copia del virus ha hecho copias de sí mismo.

4. La fase de ejecución, en la que se lleva a cabo la función. La función puede ser no dañina, como dar un mensaje en la pantalla, o dañina, como la destrucción de archivos de programas y datos.

<i>Respuestas del múltiple choice.</i>

- | | | | | |
|---------------|---------------|---------------|------------|---------------|
| 1.- d. | 2.- a, c. | 3.- b, c, d. | 4.- a, b. | 5.- d. |
| 6.- c, d, e. | 7.- b. | 8.- e. | 9.- c, d. | 10.- e. |
| 11.- a, d, e. | 12.- f. | 13.- f. | 14.- b, d. | 15.- c. |
| 16.- d. | 17.- a, c, d. | 18.- a, c, e. | 19.- a, b. | 20.- b, d, e. |