

Capítulo 1- Seguridad, ¿Para qué?

1.1 Introducción

Cuenta la mitología griega acerca de un perro de tres cabezas, llamado Kerberos (en español Cancerbero), que vigilaba las puertas del infierno. La función de Kerberos era la de dejar entrar con facilidad a las almas que radicaban ahí y de aniquilar a aquellos intrusos que no tuviesen autorización de penetrar en el infierno. Kerberos permanecía atento de cualquier invasor y mientras una cabeza dormía las otras continuaban su labor.

Este concepto del perro mitológico ha permanecido hasta nuestros días y es interesante ver como el mantener la información segura se puede convertir en un verdadero infierno.

Desde que aparecieron las primeras redes de computadoras a principios de los 80's, el tema de la seguridad es de los mas controvertidos. Lo que preocupa al usuario que no conoce de redes es que otras personas puedan acceder a su información personal, pues los archivos de toda una empresa residen en el mismo dispositivo. Este era básicamente el único obstáculo que ponían los usuarios cuando comenzaban a utilizar un sistema de redes.

Con el crecimiento de las redes y las facilidades de interconexión hacia cualquier red del mundo, la preocupación por la seguridad ha aumentado de manera considerable, pues ahora ya no es solo que un compañero de trabajo acceda a nuestra información sino que ahora, cualquier persona que tenga acceso a la intranet en la que se trabaja puede ver su información y peor aun, dañarla o contaminarla por virus.

Dado el tremendo crecimiento de las redes de computadoras y las facilidades de interconexión a redes publicas o de otras empresas, tanto los fabricantes de sistemas operativos de redes como los fabricantes de software de administración de redes, han desarrollado varios mecanismos para ofrecer al usuario toda la seguridad que requiera.

Hace varios años se veían películas de niños genios que burlaban la seguridad de los sistemas computacionales del Pentágono y creaban guerras nucleares. Hoy esto es mucho más que solo ciencia-ficción, podemos ver en revistas de interés general información acerca de las guerras cibernéticas y podemos leer en los diarios acerca de fraudes informativos multimillonarios. Los "vándalos" (muchas veces llamados de manera errónea "hackers") han encontrado muchas formas de traficar con la información ajena, desde la transferencia directa de fondos, hasta la venta de información confidencial de una compañía a otra. Inclusive, existen "vándalos profesionales", los cuales son contratados por compañías para que intenten romper sus sistemas de seguridad y con ello saber si efectivamente están protegidas o no contra este tipo de ataques.

El manejo de la seguridad no es únicamente para las agencias militares o para evitar que espías o "vándalos" penetren en nuestra red. Son para situaciones tales como usuarios no autorizados que se entrometen a revisar documentos, copiar o robar archivos y contraseñas, modificar datos o leer y utilizar documentos de correo electrónico sin autorización.

Nuevas tecnologías, nuevos métodos.

Lo que es un hecho es que el ambiente cliente / servidor y la tendencia a la utilización del ciberespacio traen consigo nuevos requerimientos de seguridad que no existían en los ambientes no distribuidos. En los sistemas no distribuidos se puede confiar en la autorización de acceso a los sistemas operativos. En el caso de los sistemas distribuidos el reto es mucho mayor ya que la información fluye a través de una red y esta puede ser observada e interceptada. Se requieren nuevas formas de seguridad para resguardar a los servidores del acceso no autorizado. El reto es poder lograr un alto nivel de protección y seguridad con el mayor grado de transparencia de estos mecanismos de control para el usuario autorizado.

Software Antivirus.

Hoy en día los ataques de virus informáticos son parte de nuestra vida cotidiana y así como aparecen nuevos virus cada día, aparecen nuevos antivirus. Y entonces usted se preguntara, ¿cual es el mejor antivirus? NINGUNO.

Cada antivirus tiene pros y contras, pero definitivamente aquellos que incluyen un programa residente en memoria cuya función es detectar el virus antes de que ataque, son los más recomendables. Existen versiones para computadoras personales, versiones para computadoras que están en red, versiones para servidores dentro de la red, etc. Aunque lo realmente importante es tener protegidas todas las computadoras de la empresa y no solo aquellas conectadas a la red. Cabe aclarar que por mas cuidado que se tenga, ninguna maquina esta exenta de ser atacada, por lo que es necesario revisar cada disco que se vaya a introducir al equipo, aunque sean discos de reconocidas marcas de software. Nunca es demasiado lo que se hace por proteger el equipo contra virus.

Llave publica / Llave privada*Proceso de Acceso al servidor de archivos.*

Cuando se trabaja en un ambiente de red, en donde hay un servidor de archivos central y varias estaciones de trabajo conectadas a este, es necesario que cada usuario que va a acceder a dicho servidor tenga lo que se conoce como: cuenta de acceso. Esta cuenta opera de manera similar a una cuenta de tarjeta de crédito, en donde el dueño de la tarjeta tiene una cuenta única que le sirve para acceder al banco. Asociada a esta cuenta esta una contraseña o clave secreta (password) que solo el usuario conoce. Cuando el usuario teclea su cuenta y su contraseña para acceder a la red, el sistema evalúa que la cuenta exista y que la contraseña sea correcta, algo similar a lo que sucede con el NIP de las tarjetas de crédito. Si todo es correcto, el usuario tiene acceso a la red y puede empezar a trabajar.

Llaves de acceso únicas

Este era el proceso de verificación inicial que hacían los sistemas de red. La información de "cuenta" y contraseña, no se volvían a validar a lo largo de la sesión.

En la actualidad, este proceso tiene una variante, que consiste en validar constantemente la vigencia y veracidad de la cuenta, así como de la contraseña. Sin entrar

en demasiado detalle, el proceso funciona de la siguiente manera: cuando el usuario teclea su cuenta y contraseña, se aplica a estos dos valores un algoritmo y se obtiene un dato conocido como: llave privada. Esta llave privada es la que se envía al servidor, el cual verifica su validez. Si la información es correcta (cuenta y contraseña existen), el sistema combina el valor de la llave privada con un valor determinado que él tiene y obtiene así una llave única. Al momento de darle el acceso al usuario a la red, le envía esa llave única, la cual deberá acompañar cualquier petición que haga el usuario durante la sesión de trabajo. En cada petición, el servidor verifica que la llave única siga siendo la misma que el usuario envió al inicio de la sesión.

Derechos

Una vez dentro de la red, el usuario esta restringido. Cada cuenta de usuario tiene determinados privilegios dentro de la red, algunos pueden ver y modificar toda la información de la red, otros solo la pueden ver, otros solo pueden ver sus archivos personales, etc. Estos privilegios o derechos se otorgan por cuenta y sobre archivos y directorios, de tal manera que cada usuario cree que lo que el ve y con lo que él trabaja es todo lo que hay en la red, el administrador debe, junto con el usuario, diseñar con detalle lo que se conoce como: Esquema de seguridad. En pocas palabras: quien puede hacer que dentro del sistema. Un buen diseño, ahorrara muchos dolores de cabeza durante la implantación del sistema.

Sistemas de respaldo.

Además de la seguridad de acceso, es necesario proteger los datos contra los usuarios que tienen derechos. Esto puede sonar extraño, sin embargo es muy común (sumamente frecuente) que el usuario que tiene derechos suficientes borre información por error. Se debe estar preparado para recuperar la mayor parte de esa información. Existen sistemas de respaldo que permiten desde un solo punto, respaldar todos los equipos de la empresa, de manera que no se tenga que mover el equipo de respaldo de un lugar a otro. Otros permiten respaldar por estadísticas, de tal manera que solo respalde la información que fue modificada desde el ultimo respaldo.

Los mejores sistemas son aquellos capaces de respaldar tanto los datos como los esquemas de seguridad; que respalden los derechos otorgados a los usuarios, lo que significa que si alguno de los servidores se estropea, se tenga un medio para reestablecerlo por completo.

Administración de la seguridad.

Dentro de la administración de redes, una parte muy importante es la administración de la seguridad. Finalmente lo que esta hará, es instalar o habilitar los sistemas de respaldo, los derechos, las contraseñas, etc. Pero de manera ordenada y previo análisis.

La administración de la seguridad empieza por analizar la red:

1. Identificar los datos confidenciales. Además de tener acceso a sus propios datos, habrá usuarios que tengan que compartir información. Este paso identifica cuales son los usuarios que compartirán datos y cuales son esos datos.

2. Identificar los puntos de acceso a la red. Dentro de una organización, hay un solo punto de acceso a la red, que es desde la estación de trabajo al servidor. Si pensamos ahora en una Internet, en donde hay gran diversidad de equipos y tecnologías, la seguridad se complica un poco ya que los puntos de acceso a la red incrementan, ya que habrá usuarios que, físicamente estén dentro de la empresa y accedan al servidor de archivos o de correo, o a ambos, desde su estación de trabajo y usuarios móviles, los cuales llegaran a la red a través de modem u otros medios de acceso remoto.

3. Determinar protección. Determinar que tipo de software y/o hardware se requiere colocar en cada uno de los puntos de acceso para asegurar que solo los usuarios autorizados podrán acceder a la red. Dentro de este paso se puede establecer también el criterio de respaldo: cada cuando se va a respaldar, que es lo que se va a respaldar, etc.

4. Monitoreo. La parte más importante en la administración de la seguridad es el monitoreo. Tener una manera de saber que la protección esta funcionando. Esto se puede hacer con bitácoras que registren cada vez que un usuario no autorizado trata de entrar a la red. Estas bitácoras sirven para sacar estadísticas y ver que tan segura es la red.

En la actualidad existen muchos sistemas y medidas de seguridad para proteger los datos de accesos no autorizados, cada uno con determinadas características. De acuerdo al tamaño de la red, es el tipo de análisis que se tiene que hacer antes de decidirse por algún sistema de seguridad. Si en la red se tiene equipo heterogéneo, será necesario evaluar que las herramientas de seguridad entre un equipo y otro sean compatibles o por lo menos que cumplan con el mismo nivel de confiabilidad. No todos los excesos son malos y menos aun cuando se trata de proteger los datos que manejan nuestra vida cotidiana.

Kerberos.

A principios de los ochenta, científicos del MIT (Massachussets Institute of Technology) trabajaban en un proyecto de redes llamado Athena, que consistía en integrar múltiples computadoras heterogéneas en una red de alto rendimiento y alta velocidad. La necesidad de integrar una red con cientos de computadoras de las cuales gran parte eran PCs, rápidamente despertó preocupación con respecto a como evitar que estudiantes (o usuarios externos) corrieran programas que espieran o dañaran la información de esta red. Muy pronto llegaron a la conclusión de que en un ambiente distribuido es casi imposible asegurar que todas las estaciones de trabajo en una red sean seguras. Basados en esto, y en la teoría de "piensa mal y acertaras" diseñaron un avanzado esquema de seguridad al cual llamaron Kerberos.

Al igual que el monstruo mitológico el mecanismo de seguridad Kerberos consiste en verificar la identidad de los usuarios, cliente o servidores -llamados unidades de confianza- y permitir revocar acceso a los servicios de la red. Kerberos requiere que las unidades de confianza intercambien contraseñas secretas que comprueben su identidad. El procedimiento que utiliza es muy parecido a lo que sucede en las películas de espías para identificarse entre sí. Un espía que personifica a un empleado de limpieza se acerca a un agente sentado en una estación de tren y le dice: "parece que lloverá en el Amazonas". El agente le contesta: "desafortunadamente, no compre zapatos color café". A través de este

intercambio de frases en clave, ambos espías verifican la identidad del otro y obtienen el suficiente nivel de confianza para continuar con sus asuntos.

En el caso de los sistemas computacionales de una red se deben establecer altos niveles de confianza entre los diferentes elementos antes de que haya intercambio de información ya que además de tener el peligro de usuarios no autorizados, existe el peligro de usurpación de servidores o de recursos. Por ejemplo, si se habla de un usuario que desea imprimir información confidencial en una impresora de red, donde los sistemas de seguridad no hacen una verificación de que el dispositivo al cual va la información sea efectivamente una impresora, esta puede ser usurpada por una máquina de fax que este imprimiendo la información confidencial en las oficinas de un competidor.

La OSF (Open Software Foundation) adoptó el sistema de autenticación de Kerberos del MIT, como parte de su arquitectura de ambiente de cómputo distribuido DEC (Distributed Computing Environment). Lo mejoró con algunas funciones de seguridad de HP y creó un estándar abierto de seguridad con el propósito de facilitar a los fabricantes la creación de aplicaciones distribuidas y para dar a los usuarios la capacidad de compartir recursos transparentemente a través de una gran variedad de redes, sistemas operativos y proveedores.

Kerberos en 8 pasos.

Paso 1. Los administradores deben crear la base de datos con seguridad y registrar a los usuarios o "unidades de confianza".

Paso 2. Cuando un usuario se da de alta al sistema el servicio de "login" manda una petición de autenticación. El nombre del cliente (no la contraseña) es enviado a través de la red.

Nota: en ningún momento el proceso de la contraseña viaja por la red.

Paso 3. El servidor de autenticación crea un "ticket" o boleto de identificación para el usuario. Los boletos son usados por el cliente para comprobar que efectivamente es quien dice ser. Estos normalmente tienen un tiempo de vida de ocho horas y son encriptados dos veces, la primera con un código que solo es conocido por los servidores de autenticación y privilegios y la segunda vez utilizando la contraseña como código.

Paso 4. El servidor de autenticación envía el boleto al cliente. Si este provee la contraseña correcta, puede decodificar el boleto y enviarlo ahora al servidor de privilegios.

Paso 5. Al recibir el boleto el servidor de privilegios verifica si el tiempo del boleto no ha expirado y si el cliente efectivamente es quien dice ser.

Paso 6. El servidor de privilegios regresa al cliente un certificado de atributos de privilegio (PAC). El PAC incluye información referente a los grupos de acceso a los cuales el usuario pertenece.

Paso 7. Con el PAC el usuario hace las llamadas al servidor de la aplicación que desea usar.

Paso 8. El servidor decodifica el PAC y decide basado en las listas de control de acceso (ACL) si el cliente puede acceder a los recursos de ese servidor o no.

Seguridad en ambientes de computo distribuido.

La seguridad en red es una prioridad en el cuidado de los dispositivos y de la integridad en la información, los niveles de seguridad aplicables van desde restricciones en hardware y software, hasta limitaciones a personas o regiones.

La evolución en las redes informáticas ha originado que más personas accedan a la información desde diversas computadoras, en ocasiones, ubicadas en diferentes localidades geográficas. Esto propicia que sea difícil detectar el lugar de acceso, lo cual es un reto complicado para la seguridad informática de una empresa, en la cual se está exponiendo la integridad de los datos, el cuidado del equipo y su funcionamiento.

¿Dónde están los riesgos?

Cuando se diseña y administra una red, es necesario tomar en cuenta la gran cantidad de factores que pueden ocasionar fallas en el sistema. De ahí que se tenga que desarrollar un análisis de riesgos enfocado a los que representan: los usuarios finales, las estaciones de trabajo, redes de área local, redes de área metropolitana, redes de área amplia, los sistemas operativos, dispositivos de interconectividad, manejadores de bases de datos, bases de datos y archivos, además de las aplicaciones. Estos aspectos se debe analizar y tomar en cuenta los riesgos que cada uno de ellos puede ocasionar, tanto de manera individual como en conjunto.

¿Cuales son los problemas que enfrenta la seguridad en redes?

Al no existir una definición respecto a la protección en el ambiente de redes, se debe tener especial cuidado en el nivel de conocimientos y confiabilidad de las personas que utilizan la red. Por ejemplo, en casos de violación que no son reportados inmediatamente o cuando las redes no se supervisan a tiempo. Es importante hacer hincapié al respecto, ya que el 75% de las violaciones en redes, son ocasionadas por el personal que trabaja diariamente con los dispositivos de la red debido a la poca aceptación de la empresa a las ideas de incorporar niveles y sistemas de defensas, como: Políticas y procedimientos de seguridad.

Entre los problemas que se pueden propiciar por esta falta de seguridad esta la proliferación de "intrusos de software" como: virus, intrusos que se hacen pasar como usuarios para modificar y alterar el funcionamiento y la información del sistema e infinidad de riesgos más.

Para evitar estos riesgos se deben planear las actividades a seguir en la administración de la red y el modelo de seguridad que se va a implantar.

Según datos de la OSF, mas de cien proveedores están desarrollando productos para el ambiente de computo distribuido-DCE.

La seguridad es un reto en sistemas distribuidos, sin embargo existe una solución abierta que ofrecerá comunicaciones seguras y acceso controlado a recursos en este tipo de ambientes. Se espera ver próximamente mas productos desarrollados bajo esta especificación que permitan la ejecución de operaciones de negocio en ambientes de computo heterogéneo y seguro.

Dimensiones de protección.

Se pueden distinguir dos dimensiones de seguridad para las redes informáticas. Una es la protección física de las redes, que incluye la seguridad del edificio o lugar de ubicación de la red; seguridad del equipo de computo y de telecomunicaciones; así como de los usuarios. A este nivel existen diversas recomendaciones que deben tomarse en cuenta como:

Controlar el acceso del usuario al equipo. Restringir el acceso a la estación o a la PC que opcionalmente debe de estar sin unidad de disco externo para evitar contagio de virus o robo de información que de lugar a la piratería, comprometer al usuario con la seguridad mediante un documento escrito y firmado.

Colocar los servidores en áreas protegidas.

Proteger el sistema de cableado y los sistemas de Inter.

1.2 Políticas de red

Fundamentalmente la seguridad informática es una colección de soluciones técnicas a problemas que no son técnicos. Se puede invertir mucho tiempo, dinero y esfuerzo en seguridad informática, pero nunca se resolverá realmente el problema de la pérdida accidental de datos o de la interrupción intencional de las actividades. Dadas las condiciones adecuadas, por ejemplo un error en algún programa, una equivocación, mala suerte mal tiempo o un agresor motivado y bien equipado, cualquier computadora puede ser comprometida, paralizada o algo peor.

El trabajo de los profesionales de la seguridad informática es ayudar a las organizaciones a decidir cuanto tiempo y dinero quieren gastar en la seguridad. Otra parte del trabajo es cerciorarse de que las empresas tengan políticas, recomendaciones y procedimientos vigentes para que el dinero que se gaste se gaste bien. Finalmente el profesional debe auditar el sistema para verificar que se implanten los controles necesarios para lograr los objetivos de las políticas. Esto indica que la seguridad practica es realmente una cuestión de administración y manejo más que una cuestión de destreza técnica. Por lo tanto, la seguridad tiene que ser una de las prioridades de la administración de la empresa.

En esta propuesta el proceso de la planeación de la seguridad se divide en seis etapas diferentes:

Planeación de las necesidades de seguridad
 Análisis de riesgos
 Análisis de costo-beneficio
 Creación de políticas que reflejen las necesidades
 Implementación
 Auditoría y respuesta ante incidentes

Hay dos principios de importancia fundamental, implícitos en política efectiva y planeación de seguridad:

La conciencia sobre seguridad y políticas debe ir de arriba hacia abajo en una organización. La preocupación y conciencia de los usuarios son importantes, pero no bastan para construir y mantener una cultura efectiva de seguridad. Los directivos de la organización deben considerar que la seguridad es importante y aceptar las mismas reglas y reglamentos que todos los demás.

La seguridad efectiva quiere decir proteger la información. Todos los planes, políticas y procedimientos deben reflejar la necesidad e proteger la información en cualquiera de sus manifestaciones. Los datos privados no pierden su valor si se imprimen o envían por fax en lugar de estar en lugar de estar archivados en un disco. La información confidencial de un cliente no pierde su valor si la mencionan por teléfono dos usuarios en lugar de estar contenida en mensajes de correo electrónico. La información debe protegerse en todas sus formas.

Cómo se planean los requerimientos de seguridad

Una computadora es segura si se comporta como se espera que se comporte.

Hay muchos tipos de seguridad informática y muchas definiciones distintas. En lugar de presentar una definición formal, se adopta el enfoque práctico y se discuten las categorías de protección que se deben tomar en cuenta. Se piensa que una computadora segura es una computadora que puede usarse y, asimismo, una computadora que no puede usarse, por cualquier motivo, no es muy segura.

Dada esta amplia definición hay muchas clases de seguridad que deben preocupar a los usuarios y a los administradores de sistemas de cómputo.

Confidencialidad

Proteger la información para que nadie pueda leerla o copiarla, sin autorización del dueño. Este tipo de seguridad no sólo protege toda la información su conjunto sino también protege cada pedazo de información, pedazos que en sí mismos pueden parecer inocuos pero que pueden usarse para inferir otra información confidencial.

Integridad de datos

Proteger la información (incluyendo los programas) para evitar que se borre o altere en cualquier manera, sin el permiso del dueño de la información. Los ítems de información

que deben protegerse incluyen registros contables, cintas de respaldo, hora de creación de archivos y la documentación.

Disponibilidad

Proteger o servicios para que no se degraden o dejen de estar disponibles sin autorización. Si el sistema no está disponible cuando un usuario con autorización lo necesita, la consecuencia puede ser tan dañina como perder información que esté guardada en el sistema.

Consistencia

Asegurar que el sistema se comporta como lo esperan los usuarios autorizados. Si los programas o el equipo de repente se comportan en forma radicalmente distinta a como lo hacían antes, en especial después de una actualización o de la eliminación de un error, puede suceder n desastre. Basta imaginar lo que pasaría si el comando *ls* borrara archivos de vez en cuando en lugar de listar sus nombres. Este tipo de seguridad también puede considerarse como asegurar que los datos y programas que se usan sean correctos.

Control

Reglamentar el acceso al sistema. Si individuos (o programas) desconocidos y no autorizados están en un sistema puede presentarse un enorme problema. Hay que preocuparse de cómo entraron, qué habrán podido hacer y quién mas habrá entrado al sistema. La recuperación de un episodio de esta naturaleza puede requerir mucho tiempo y dinero para reconstruir y reinstalar el sistema, y verificar que no se haya cambiado o divulgado algo importante, aunque en realidad no haya pasado nada.

Auditoría

Además de preocuparse acerca de usuarios no autorizados, los usuarios a veces se equivocan, o cometen actos maliciosos. Si esto sucede es necesario determinar qué se hizo, quién lo hizo y qué fue afectado. La única forma de lograr esto es tener un registro inexpugnable de la actividad que sucede en el sistema e identifica en forma no ambigua a todos los actores y acciones. En algunas aplicaciones críticas las trazas de auditoría pueden ser tan extensas que permiten deshacer las operaciones realizadas para ayudar a restablecer al sistema a su estado correcto.

Aunque estos aspectos de la seguridad son importantes cada organización les dará una importancia distinta. Esta variación se deba a que cada organización tiene sus propias preocupaciones de seguridad y eso es lo que determina sus prioridades y políticas. Por ejemplo:

En los *ambientes bancarios*, la integridad y auditabilidad son las preocupaciones más importantes, mientras que la confidencialidad y la disponibilidad vienen después.

En los *sistemas relacionados con la seguridad nacional* que procesan información clasificada, la confidencialidad puede venir primero y la disponibilidad al final.

En una *universidad*, la integridad y la disponibilidad pueden ser los requerimientos más importantes. La prioridad es que los estudiantes puedan hacer sus trabajos en lugar de saber con precisión cuando usaron sus cuentas.

Los administradores de seguridad deben entender a fondo las necesidades del entorno operativo y de los usuarios. Luego tienen que definir los procedimientos tomando eso en cuenta.

Confianza

Los profesionales de la seguridad no hablan de sistemas de cómputo “seguros” o “inseguros”. Más bien se usa el término “**confiable**” para describir el nivel de confianza que se tiene en que un sistema se comporte como se espera. Esto reconoce que no se puede lograr la seguridad absoluta. Sólo se puede tratar de acercarse a ella desarrollando una confianza suficiente en la configuración total para garantizar su uso en aplicaciones críticas.

Desarrollar una confianza adecuada en un sistema de cómputo requiere reflexión y planeación cuidadosas. Las decisiones deben basarse en decisiones sobre políticas sanas, y en un análisis de riesgos.

Toda esta propuesta se dedicará a estudiar los procedimientos generales que sirven para crear planes y políticas de seguridad que funcionen.

Las empresas, universidades y agencias gubernamentales deben hablar con sus departamentos de auditoría interna y/o administración de riesgos para obtener ayuda (a lo mejor ya tienen políticas y planes que es necesario conocer). Se pueden obtener más información consultando distintos textos o bibliografías. Además es posible contratar a una empresa consultora. Por ejemplo, muchas empresas de contadores o auditores cuentan con equipos de profesionales que saben evaluar la seguridad de instalaciones de cómputo.

Análisis de riesgos

El primer paso para mejorar la seguridad de un sistema es contestar estas preguntas básicas:

- ¿Qué se debe proteger?
- ¿Contra qué debe protegerse?
- ¿Cuánto tiempo, dinero y esfuerzo se está dispuesto a invertir para obtener una protección adecuada?

Estas preguntas forman el fundamento del proceso llamado *análisis de riesgos*.

El análisis de riesgos es una parte muy importante del proceso de seguridad informática. No se puede proteger algo si no se sabe contra qué hay que protegerlo. Después de conocer los riesgos se puede planear las políticas y técnicas que se necesitan para reducir esos riesgos.

Por ejemplo, si hay un riesgo de fallas en la alimentación eléctrica y la disponibilidad del equipo se considera es importante se puede reducir el riesgo adquiriendo una fuente de poder ininterrumpible.

Una estrategia de análisis simple

Para empezar se presentará una forma simplista de analizar los riesgos. Este ejemplo puede ser más complicado de lo que se necesita en el caso de una computadora casera o una empresa pequeña. Indudablemente es insuficiente para una empresa grande, una agencia gubernamental o una universidad grande. En esos casos se debe pensar en programas especializados en el análisis y en la posibilidad de contratar una empresa consultora externa que tenga experiencia en el análisis de riesgos.

Las tres etapas del análisis de riesgos son:

Identificación de los activos
Identificación de las amenazas
Cálculo de los riesgos

Existen muchas maneras de efectuar este proceso. Un método que ha resultado exitoso es llevar a cabo una serie de talleres. Se invita a una gama de usuarios, administradores y directivos de toda la organización. A lo largo de varias semanas se completan las listas de activos y amenazas. Este proceso no sólo permite obtener un conjunto de listas más completo sino que también promueve la concientización de todos los que asisten.

Identificación de activos

Se debe hacer una lista de todo lo que se quiere proteger, la cual debe basarse en el plan de negocios y el sentido común. El proceso puede requerir conocimientos de la legislación local vigente, un completo entendimiento del equipo y locales, y un buen conocimiento del plan de seguros que se ha contratado.

Los ítems que se tienen que proteger incluyen objetos tangibles (unidades de disco, monitores, cables de red, medios de respaldo, manuales) o intangibles (capacidad de seguir operando, imagen pública, reputación en el medio, acceso a la computadora, la clave *root* del sistema). Esta lista debe contener todo lo que se considera valioso. Para determinar si algo es valioso hay que pensar en lo que constaría en pérdida de ingresos, tiempo perdido o costo de reparación o reemplazo.

Algunos ítems que probablemente tienen que aparecer en lista son:

Tangibles:
Computadoras
Datos privados

Respalos y acervos
 Manuales, guías y libros
 Listados
 Medios de distribución de programas comerciales
 Equipo de distribución de programas comerciales
 Equipo y cableado de comunicaciones
 Registros de personal
 Registros de auditoría

Intangibles:
 Salud y seguridad del personal
 Privacidad de los usuarios
 Contraseñas personales
 Imagen pública y reputación
 Buena voluntad de los clientes o compradores
 Disponibilidad de proceso
 Información sobre la configuración

Hay que tomar una perspectiva amplia sobre éstos ítems en lugar de considerar sólo aspectos asociados al cómputo. Si existe la preocupación de que alguien lea los reportes financieros internos, no importa si lo hacen en un listado que se ha tirado o en un mensaje de correo electrónico.

Identificación de amenazas

El siguiente paso es hacer una lista de las amenazas a los activos. Algunas amenazas serán ambientales, como incendios, terremotos, explosiones o inundaciones. Pueden incluir eventos poco probables pero posibles como fallas estructurales o el descubrimiento de asbesto en la sala de cómputo que requeriría mantenerla vacía durante un largo tiempo. Otras amenazas provienen del personal y otras más de los extraños. He aquí algunos ejemplos:

Enfermedades de personas clave
 Enfermedades simultáneas del personal (epidemias)
 Pérdida (por renuncia, despido o muerte) de personal clave
 Interrupción de servicios de teléfono o red
 Interrupción breve de otros servicios (teléfono, agua, electricidad)
 Rayos
 Inundaciones
 Robo de discos o cintas
 Robo de la computadora portátil de una persona clave
 Robo de la computadora casera de una persona clave
 Aparición de un virus
 Quiebra de un proveedor
 Errores en programas
 Subversión de empleados
 Subversión de terceras partes (personal de mantenimiento de un proveedor)

Problemas laborales
Terrorismo político
Intrusos maliciosos
Colocación de información privada o inapropiada en Usenet

Cuantificación de los riesgos

Cuando se han identificado los riesgos debe estimarse la probabilidad de que ocurra cada uno de ellos. Esto puede ser más sencillo si se consideran ocurrencias anuales.

La cuantificación de riesgos es un trabajo pesado. Algunas estimaciones se pueden obtener de terceros, por ejemplo las compañías de seguros. Si algo sucede en forma regular, la estimación puede basarse en los registros históricos. Las organizaciones industriales pueden haber recopilado estadísticas o reportes que se han publicado. También es posible hacer estimaciones con base en cálculos serios extrapolados de la experiencia. Por ejemplo:

La compañía de luz puede proporcionar una estimación oficial de la probabilidad de que un edificio en particular sufra una interrupción del servicio durante el próximo año. También es posible que pueda cuantificar el riesgo de que la interrupción dure unos cuantos segundos o que dure minutos u horas.

La compañía de seguros puede proporcionar información actuarial sobre la probabilidad de muerte del personal clave según su edad y estado de salud.

Los registros personales pueden utilizarse para calcular la probabilidad de la salida de empleados clave en el área de cómputo.

La experiencia y la extrapolación inteligente pueden permitir apreciar la probabilidad de que se descubra que los programas de un cierto proveedor contengan errores serios durante el próximo año (tal vez 100%)

Si se espera que algo suceda más de una vez al año hay que anotar cuántas veces se espera que suceda. Si se espera que ocurra un terremoto serio una vez cada cien años se pone 1% en la lista, pero si se espera descubrir tres errores serios en el *sendmail* durante el próximo año se anota 300%

Se deben revisar los riesgos

El análisis de riesgos no debe hacerse sólo una vez y después olvidarse, sino que es necesario actualizarlo periódicamente. Además, la parte relativa a la evaluación de amenazas debe repetirse cada vez que la operación y la estructura cambian en forma significativa. Si ocurre una reorganización, se hace una mudanza a otro edificio, se cambian proveedores o suceden otros cambios importantes se tienen que volver a evaluar las amenazas y las pérdidas potenciales.

Análisis de costo-beneficio

Al terminar el análisis de riesgos es necesario asignar un costo a cada riesgo, y determinar el costo de defenderse. A esto se le llama *análisis de costo-beneficio*.

Costo de las pérdidas

Calcular las pérdidas puede ser muy difícil. En forma simple se toma en cuenta el costo de reparar o sustituir un ítem. Una evaluación mas sofisticada puede tomar en cuenta el costo de no disponer del equipo, de la capacitación adicional requerida, de los procedimientos adicionales que resulten de la pérdida, el costo de la reputación de la empresa e incluso el costo a los clientes. En general, la inclusión de más factores en el análisis de costos implicará más trabajo, pero mejorará su precisión.

Para la mayor parte de los propósitos no se requiere asignar un valor exacto a cada riesgo posible. Normalmente es suficiente un rango de costos para cada ítem. Por ejemplo la pérdida de una docena de disquetes en blanco se puede clasificar como “menor que 500 dólares”, mientras que un incendio destructivo en una sala de cómputo puede clasificarse como “mayor a 1 000 000 de dólares”. Algunos ítems se pueden clasificar como “irreparables / insustituibles”, por ejemplo la base de datos de contabilidad completa o la muerte de un empleado clave.

Puede ser conveniente asignar estos costos con una escala de pérdida más fina que simplemente “pérdida / no pérdida”. Por ejemplo, si se quiere asignar costos separados a cada una de las siguientes categorías (no se han ordenado por importancia):

- Interrupción de corto plazo (menos de 7 a 10 días)
- Interrupción de mediano plazo (una o dos semanas)
- Interrupción de largo plazo (más de 2 semanas)
- Destrucción o pérdida permanente
- Pérdida o daño parcial accidental
- Pérdida o daño parcial intencional
- Divulgación no autorizada interna
- Divulgación no autorizada a algunos extraños
- Divulgación completa a extraños, competidores y a la prensa
- Costo de reemplazo o recuperación

Costo de prevención

Para terminar hay que calcular el costo de prevenir cada tipo de pérdida.

Por ejemplo, el costo de recuperarse de una falla momentánea de potencia puede ser sólo el tiempo de inactividad del personal además del tiempo necesario para reiniciar el equipo. Pero el costo de la prevención puede ser la adquisición de un sistema de potencia ininterrumpida.

El costo debe amortizarse a lo largo de la vida esperada de las opciones, según sea apropiado. La obtención de estos costos puede revelar costos secundarios o ingresos que también deben tomarse en cuenta. Por ejemplo, la instalación de un sistema contra incendios mejor puede hacer que disminuyan las primas del seguro contra incendios y además representar un beneficio fiscal adicional por depreciación de capital. Pero gastar

dinero en un nuevo sistema contra incendios significa que el dinero no estará disponible para otras cosas, tales como capacitación de persona, o incluso para invertirlo.

Como sumar las cifras

Al término del proceso se debe tener una matriz multidimensional que consista de activos, riesgos y posibles pérdidas. Para cada pérdida hay que tener una probabilidad, la pérdida estimada y la cantidad de dinero que se requiere para defenderse de esa pérdida. Si se quiere ser preciso es necesario incluir la probabilidad de que la defensa falle.

El proceso de determinar si cada defensa debe emplearse o no es ahora directo. Se hace multiplicando cada pérdida esperada por la probabilidad de que ocurra como resultado de cada amenaza. Los resultados se ponen en orden descendente y se compara el costo del suceso con el costo de su defensa.

Esta comparación tiene como resultado una lista en orden de prioridad de lo que se debe hacer. La lista puede ser sorprendente. El objetivo debe ser evitar pérdidas caras y probables antes de preocuparse de pérdidas menos probables y poco cuantiosas. En muchos ambientes, los incendios y la pérdida de personal clave son mucho más probables y dañinos que los virus o los intrusos que usen la red. Es sorprendente que los intrusos y los virus parecen acaparar la atención y los presupuestos de los administradores. Esto no es eficiente, ni mejora la confianza que se tenga en el sistema en su conjunto.

Comentario de que no es posible eliminar los riesgos

Para decidir qué hacer se deben estudiar las cifras que se han obtenido sobre prevención y recuperación, y analizar cómo se pueden atender los ítems de más alta prioridad. Para lograrlo hay que *sumar el costo de recuperación a las pérdidas promedio, y multiplicar ese resultado por la probabilidad de que suceda el evento*. Se debe comparar el resultado con el costo anual de prevención. Si el costo de prevención es menor que el de la defensa del riesgo es recomendable invertir en la estrategia de prevención si se cuenta con los recursos para hacerlo. Si el costo de prevención es mayor que el de la defensa no debe hacerse nada hasta que se hayan atendido otras amenazas.

Cómo convencer a los directivos

La seguridad no es gratuita. Las medidas más complicadas de seguridad son más caras. Los sistemas más seguros son más difíciles de usar, aunque esto no tiene porque ser así. La seguridad puede estorbar a los usuarios “avanzados” que muchas veces quieren llevar a cabo operaciones difíciles y a veces peligrosas sin autenticación o responsabilidad. Estos usuarios avanzados pueden ser muy poderosos dentro de una organización.

Al terminar el análisis de riesgos y el de costo-beneficio se debe convencer a los directivos de la organización que es necesario actuar. Normalmente se formulará una política que se adoptaría oficialmente. Con frecuencia, ésta es una marcha cuesta arriba. Afortunadamente no hay razón para que esto suceda.

El objetivo del análisis de riesgos y de costo-beneficio es asignar prioridades a las acciones y al gasto en seguridad. Si el plan de negocios sugiere que no se acepte ningún riesgo mayor a 10 000 al año, sin seguro, se puede emplear el análisis realizado para determinar cuánto hay que gastar para lograr este objetivo. El análisis es también una guía sobre qué debe hacerse primero, qué debe hacerse después y para identificar lo que puede dejarse para muchos años después.

Otro beneficio del análisis de riesgos es que ayuda a justificar ante los directivos las necesidades de recursos adicionales para la seguridad. Una gran cantidad de administradores y directores no sabe mucho sobre computadoras, pero entiende lo que es un riesgo y los análisis de costo-beneficio. Si se puede demostrar que una organización se enfrenta a un riesgo que podría ascender a 20 000 000 de dólares al año (incluyendo las pérdidas y los costos de recuperación de lo que se tiene), esta estimación ayudará a convencer a la administración de que asigne dinero para aumentar el personal y los recursos.

Por otra parte, si se llega a decir a la administración vagamente “Es muy probable que suframos varias penetraciones desde Internet después de la próxima recomendación del CERT”, es poco probable que se logre algo más que una leve preocupación (si acaso).

Políticas

Las políticas sirven para definir qué se considera valioso y especifican qué medidas hay que tomar para proteger esos activos.

Las políticas se pueden formular de varias maneras: Pueden escribirse políticas sencillas y generales que en unas cuantas páginas cubran muchas posibilidades. O se pueden establecer políticas para diversos activos: políticas para correo electrónico, para datos sobre el personal para información contable. Un tercer enfoque, el cual usan muchas empresas grandes, es tener políticas pequeñas y sencillas complementadas por estándares y recomendaciones sobre el comportamiento. Se estudiará brevemente este tercer enfoque pero el lector debe recordar que pueden establecerse políticas más sencillas.

El papel de las políticas

Las políticas juegan tres papeles principales. Primero, aclaran qué se está protegiendo y por qué. Segundo, establecen la responsabilidad de la protección. Tercero, ponen las bases para resolver e interpretar conflictos posteriores. Lo que las políticas *no* deben es listar riesgos específicos, computadoras o individuos por nombre. Las políticas deben ser generales y no variar mucho a lo largo del tiempo. Por ejemplo:

La información y el procesamiento de la información son un recurso crítico para la Empresa XX. Se debe proteger la información según su valor para XX, y de acuerdo con la legislación aplicable. Todos los empleados comparten la responsabilidad de proteger y supervisar la información que se produzca, manipule, reciba o transmita en sus

departamentos. Todos los empleados también comparten la responsabilidad del mantenimiento, operación adecuada y protección de los recursos de proceso de la información de XX.

La información que debe protegerse es aquella que se descubra, conozca, obtenga o maneje durante las actividades de negocios que no sea conocida en general fuera de XX. Esto incluye información privada de negocios (los nuestros y los de otras organizaciones y empresas), información sobre patentes, datos sobre el personal, información financiera, información sobre oportunidades de negocios y cualquier otra que le confiera una ventaja a XX si no se divulga. La información personal acerca de los empleados, clientes y proveedores también debe considerarse confidencial y debe protegerse.

Toda la información de XX que esté almacenada en cualquier forma –en medios de cómputo, listados en microfilm, medios fotográficos o cualquier otro medio tangible– es responsabilidad del director de Información. Esto implica que las facilidades de XX sólo se pueden usar para las funciones relacionadas con los negocios que indique el presidente. El director de Información es el responsable de proteger toda la información y los medios de procesar la información y los medios de procesar la información que pertenezcan a XX, estén o no ubicadas en instalaciones propiedad de la empresa. El director tendrá autoridad para actuar según esta responsabilidad con la aprobación del presidente de XX. El director de Información formulará los estándares y recomendaciones que sean necesarios, de acuerdo con la práctica usual de los negocios, para asegurar la protección y la continuidad del proceso de la información.

Es importante notar en este ejemplo de políticas que aparece la definición de qué se está protegiendo, quién es responsable de protegerlo y a quién le corresponde crear recomendaciones adicionales. Estas políticas se pueden mostrar a todos los empleados y a los extraños para explicárselas. Tiene vigencia sin hacer referencia a qué sistema operativo se emplee o a quién sea el director de Información.

Estándares

Los estándares sirven para codificar las prácticas exitosas de seguridad en una organización. Se redactan generalmente en términos de “es obligatorio”. Los estándares en general no dependen de las plataformas y por lo menos implican una métrica que permita determinar si se han cumplido. Los estándares se elaboran para apoyar las políticas y cambian lentamente en el tiempo. Pueden enfocarse a asuntos como la manera de investigar a los empleados nuevos, cuánto tiempo deben conservarse los respaldos y cómo deben probarse los sistemas de potencia ininterrumpida.

Recomendaciones

Las recomendaciones se redactan en términos de “debería”. La intención de las recomendaciones es interpretar los estándares en el contexto de un cierto entorno, ya sea un entorno de programas o un entorno físico. A diferencia de los estándares, se pueden violar las recomendaciones si resulta necesario. A diferencia de los estándares, se pueden violar

las recomendaciones si resulta necesario. Como lo indica su nombre, las recomendaciones no son estándares de comportamiento sino guías para el comportamiento.

Las recomendaciones suelen ser muy específicas y se basan en arquitecturas dadas, incluso relacionadas con una computadora determinada. Las recomendaciones cambian con mayor frecuencia que los estándares para tomar en cuenta los cambios de condiciones de trabajo.

Algunos consejos sobre cómo desarrollar políticas prácticas

El papel de las políticas (y de sus estándares y recomendaciones) es ayudar a proteger los ítems que (colectivamente) se consideran importantes. No necesitan ser demasiado específicas y complicadas en la mayoría de los casos. A veces es suficiente una afirmación de la política como se muestra en el siguiente ejemplo:

El uso y la protección de este sistema son responsabilidad de todos. Haga sólo lo que quisiera que hicieran todos. Respete la privacidad de los demás usuarios. Si aparece un problema arréglole usted mismo o repórtelo enseguida. Cumpla todas las leyes que se refieren al uso del sistema. Acepte responsabilidad por lo que haga e identifíquese siempre. ¡Diviértase!

A veces hay que emitir políticas más formales, que hayan sido revisadas por el departamento legal de la empresa y por varios consultores sobre seguridad, para proteger los activos. Cada organización es distinta. Se sabe en algunas organizaciones que tienen varios tomos de políticas, estándares y recomendaciones para los sistemas que usan UNIX.

Hay algunas ideas clave para formalizar las políticas que se deben mencionar más explícitamente.

Asignar un dueño

Cada pedazo de información y de equipo que daba protegerse debe tener asignado un “dueño”. Ésta es la persona responsable de la información, incluyendo sus copias, destrucción, respaldo y otros aspectos de la protección. Es también la persona que tiene cierta autoridad sobre otorgar acceso a la información.

El problema con la seguridad en muchos entornos es que existe información importante sin un dueño claramente establecido. El resultado es que los usuarios nunca saben quién toma las decisiones sobre el almacenamiento de la información, o quién regula el acceso a la información. A veces la información desaparece sin que nadie se dé cuenta durante un periodo largo porque no hay un “dueño” a quien avisar o que vigile la situación.

Ser positivo

La gente responde más favorablemente a las afirmaciones positivas que a las negativas. En lugar de escribir una lista larga de frases que contengan “no se debe”, piense en cómo decir lo mismo de una manera positiva. La breve política que se acaba de mencionar podría haber sido redactada en forma negativa como se muestra a continuación, pero observe cuánto mejor se lee en su forma original.

Es su responsabilidad no permitir que el sistema se use mal. No haga cosas que no quiere que los demás hagan. No viole la privacidad de los demás. Si encuentra un problema no lo mantenga en secreto si no puede arreglarlo usted mismo. No viole las leyes que rigen el uso del sistema. No trate de culpar a otros de lo que haga y no oculte su identidad. No lo pase mal.

Los empleados también son humanos

Al escribir las políticas hay que pensar en los usuarios. Cometerán errores y habrá malos entendidos. Las políticas no deben sugerir que los usuarios serán hostigados si ocurre un error.

Aún más, debe tomarse en cuenta que los sistemas de información pueden contener datos de los usuarios que les gustaría mantener privados. Esto puede ser parte del correo electrónico, registros de personal y evaluaciones de trabajo. Este material también debe protegerse, aunque no se puede garantizar una privacidad absoluta. Se debe ser considerado con los usuarios y sus sentimientos.

Concentrarse en la capacitación

Es prudente incluir estándares para la capacitación inicial y continua de todos los usuarios. Cada usuario debe tener una capacitación básica sobre la concientización de la seguridad, y tener alguna forma periódica de refrescar esa concientización. Los usuarios capacitados y educados son presas menos fáciles de timos y ataques de ingeniería social. También estarán más contentos sobre las medidas de seguridad si entienden por qué se toman.

Una parte importante de cualquier sistema de seguridad es darle al personal tiempo y apoyo para que se continúen capacitando y educando. Siempre hay nuevas herramientas y nuevas amenazas, nuevas técnicas y nueva información por aprender. Si los empleados pasan 60 horas cada semana rastreando un virus fantasma en computadoras personales y haciendo respaldos, no serán tan efectivos como quienes reciben unas cuantas semanas de capacitación cada año. Más aún, serán más felices en su trabajo si se les da la oportunidad de crecer y aprender durante el mismo, y se les permite pasar las tardes y fines de semana con sus familias en lugar de tener que usarlos poniéndose al corriente con los respaldos e instalando programas.

La autoridad debe estar conmensurada con la responsabilidad

El primer principio de la administración de la seguridad de Spaf dice así:

Si tiene responsabilidad sobre la seguridad, pero no tiene autoridad para fijar las reglas y castigar a quienes las violen, su papel en la organización es asumir la culpa si sucede algo grave.

Considérese el caso conocido de un administrador de un sistema que sorprendió a un programador tratando de irrumpir en la cuenta de *root* del sistema de nómina. Al investigar el incidente se descubrió que la cuenta de ese programador estaba llena de archivos de contraseñas de muchas máquinas de red, muchas de las cuales estaban descifradas. El administrador inmediatamente canceló la cuenta e hizo una cita con el supervisor del programador.

El supervisor no lo apoyó. Llamó al vicepresidente de la empresa y exigió que se le devolviera la cuenta al programador. Lo necesitaba para cumplir con un trabajo a tiempo. El administrador fue amonestado por cancelar la cuenta y se le dijo que no lo volviera a hacer.

Tres meses después despidieron al administrador cuando alguien penetró al sistema de nómina que debía haber protegido. Se dice que el programador fue promovido y que le aumentaron el sueldo, a pesar de que aparentaba tener mucho efectivo.

Quien se encuentre en una situación similar debe actualizar su currículo y empezar a buscar otro trabajo antes de que eso ocurra por circunstancias fuera de su control.

Elegir una filosofía básica

Debe decidirse si se adoptará el modelo: “Todo lo que no esté específicamente prohibido está permitido”, o bien, el que dice “Todo está prohibido excepto lo que esté específicamente permitido”. Luego hay que ser consistentes en lo que se defina como “todo lo demás”.

Adoptar defensas a fondo

Al planear defensas y las políticas no hay que detenerse en una sola capa. Deben instaurarse varios niveles independientes y redundantes de protección. Esto incluye la vigilancia y auditoría para asegurar que las protecciones funcionen. La oportunidad de que un agresor penetre una sola capa defensiva es mucho mayor de que penetre tres capas más un sistema de alarmas.

El problema de la seguridad por ocultación.

En esta parte se hablará de la formación de políticas con algunos comentarios acerca del conocimiento. En la seguridad tradicional, proveniente principalmente de las aplicaciones militares, existe el concepto de "necesidad de saber". La información se reparte, y a cada cual se le entrega sólo lo que necesita para hacer su trabajo. En ambientes en los que ciertos ítems de información son sensibles o donde se deduce un problema de seguridad, esta política tiene sentido. Si tres ítems de información juntos pueden permitir

una conclusión dañina y nadie conoce más de dos de ellos, se puede asegurar la confidencialidad.

En un ambiente de operación de computadoras el concepto de "necesidad de saber" por lo general no es apropiado. Esto es particularmente cierto si se basa en el hecho de que el enemigo desconoce algo técnico. Este concepto incluso puede ser dañino para la seguridad.

Cuatro pasos para lograr una computadora mas segura.

Operar una computadora segura es un trabajo pesado. Si no hay tiempo para hacer un análisis de riesgos y de costo-beneficio completo, como se ha descrito, se recomienda que por lo menos se tomen los siguientes cuatro sencillos pasos:

Decidir cuán importante es la seguridad del sitio. Si es muy importante y se piensa que la organización sufriría un daño importante si se violara la seguridad, debe darse suficiente prioridad a la respuesta. Asignar la responsabilidad de la seguridad a un medio tiempo de un programador con exceso de trabajo y sin capacitación formal en seguridad es una invitación a que ocurran problemas.

Involucrar y educar a la comunicación de usuarios. ¿Entienden los usuarios del sitio los peligros y riesgos que implican las malas prácticas de seguridad (y saben cuáles son esas prácticas) ? Los usuarios deben saber qué hacer y a quien llamar si ven algo sospechoso o inapropiado. Educar a los usuarios ayuda a convertirlos en parte del sistema de seguridad. Ocultarles las limitaciones del sistema y de operación no mejora la seguridad del sistema. Los agresores siempre tienen otras fuentes de información.

Idear un plan para realizar y guardar respaldos de los datos del sistema. Debe haber un lugar fuera del sitio para guardarlos por si sucede un desastre. Así se podrá reconstruir el sistema. Esto se estudia con más detalle en el capítulo de seguridad física.

Permanecer curioso y sospechar. Si algo inusual sucede, hay que sospechar que existe un intruso, y debe investigarse. Normalmente se encontrará que es un error de programación o un error de uso de algún recurso del sistema. Pero de vez en cuando se encontrará algo más serio. Por ello, cada vez que algo suceda que no se pueda explicar completamente se debe sospechar que existe un problema de seguridad e investigarlo.

Considere un ambiente en el cual la administración decide esconder los manuales para que los usuarios no puedan reconocer los comandos y opciones que pueden usarse para penetrar al sistema.

Los administradores pueden pensar que han mejorado la seguridad, pero probablemente no lo han logrado. Un enemigo persistente encontrará la misma documentación en otro lugar, la obtendrá de otros usuarios o de otros sitios. Muchos proveedores venden copias de sus manuales sin pedir una licencia de uso. Con frecuencia lo único que hay que hacer es ir al colegio o universidad más cercana para

encontrar copias de los manuales. Muchísima información sobre la documentación de UNIX está disponible en la librería más cercana. Los administradores no pueden cerrar los caminos que permiten aprender cosas sobre el sistema.

Pero los usuarios locales pierden eficiencia puesto que no pueden consultar la documentación y aprender mejores opciones. También tendrán una actitud negativa por el mensaje implícito de la administración: “No confiamos completamente en que sean usuarios responsables”. Además, si alguien empieza a abusar de los comandos y características del sistema, la administración no tiene acceso al talento que se necesita para reconocer o resolver el mismo. Y si algo llega a suceder a los pocos usuarios que tienen permiso de ver la documentación, entonces no habrá nadie con la experiencia o conocimientos necesarios para tomar su lugar o ayudar a resolver un problema.

Ocultar errores de programación o las características excepcionales en secreto es también una mala práctica de seguridad. Los desarrolladores de sistemas con frecuencia colocan puertas traseras en sus programas para obtener privilegios sin tener que dar una contraseña. Otras veces se permite que errores de programación que tienen implicaciones profundas de seguridad persistan ya que la administración supone que nadie los conoce. El problema con estos enfoques es que las características y los errores de los programas tienden a ser descubiertos por accidente o por intrusos maliciosos persistentes. Si se mantienen en secreto no se pueden vigilar y no se pueden corregir. Cuando sean descubiertos, el problema volverá a todos los sistemas similares vulnerables a ataques por parte de quienes los hayan encontrado.

Mantener algoritmos secretos, tales como algoritmos de cifrado desarrollados localmente, también es una práctica dudosa. A menos que se sea experto en criptografía, es difícil juzgar la fuerza de un algoritmo. El resultado de esa política puede ser un mecanismo que tiene un enorme agujero. Un algoritmo secreto no puede ser estudiado por nadie y por lo tanto, si alguien descubre una falla, podrá tener acceso a los datos sin que nadie lo sepa.

Asimismo mantener en secreto el código fuente del sistema operativo o de una aplicación no garantiza la seguridad. Quienes verdaderamente quieran penetrar el sistema de vez en cuando encontrarán un agujero con o sin el código fuente. Pero sin el código fuente, los usuarios no pueden examinar un programa de manera sistemática para encontrar problemas.

La clave es la actitud. Cuando se toman medidas defensivas basadas principalmente en secretos se pierde toda la seguridad si la discreción se pierde. Se puede caer en la situación de no saber y no poder determinar si se ha perdido la discreción, porque para mantenerla se han tenido que restringir las auditorías y la vigilancia. Es mejor usar algoritmos y mecanismos inherentemente robustos aunque los conozcan los enemigos. El hecho de que se usen mecanismos robustos y conocidos puede desalentar a algunos agresores y puede lograr que los curiosos busquen su diversión en otros sitios. Poner dinero en una caja fuerte es mejor que esconderlo en un frasco de mayonesa en la cocina porque nadie sabe que está allí.

Cómo declarar los problemas

A pesar de que la “seguridad por ignorancia” no se recomienda, no se trata de proponer que se dé publicidad a los nuevos agujeros de seguridad en el momento en que se descubran. Hay una distinción entre mantener secretos y ser prudente. Si se descubre una falla de seguridad en programas distribuidos o ampliamente disponibles, es necesario avisárselo al proveedor *discretamente* en cuanto sea posible. También debe avisarse a uno de los equipos FIRST (Forum of Incident and Response Security Teams). Estas organizaciones pueden actuar ayudando a los proveedores a arreglar los errores y a distribuir las correcciones de manera apropiada.

Si se “anuncia” una falla de seguridad se pone en riesgo a todos los que usan ese programa pero no tienen la capacidad o el tiempo de corregirlo. En el ambiente de usuarios UNIX muchos están acostumbrados a tener a mano el código fuente para hacer localmente las modificaciones que se requieran. Lamentablemente no todos tienen esa suerte, y muchos esperan semanas o meses para que el proveedor les entregue versiones corregidas de los programas. Algunos sitios ni siquiera son capaces de actualizar sus programas porque usan aplicaciones de llave en mano o aplicaciones que han sido certificadas en la configuración actual. Otros sistemas los usan individuos que no tienen los conocimientos necesarios para arreglar los problemas. Aun otros ni siquiera están en producción, o cuando menos están fuera de mantenimiento. Es necesario actuar en forma responsable. Puede ser preferible distribuir los arreglos sin dar muchas explicaciones sobre la vulnerabilidad subyacente que dar a los agresores detalles de cómo penetrar los sistemas que no se hayan arreglado.

Se conocen instancias de que personas bien intencionadas han reportado problemas de seguridad en foros públicos. Aunque la intención era obtener arreglos rápidos de los proveedores afectados, el resultado ha sido una oleada de intrusiones en sistemas cuyos administradores no tenían acceso al foro, o que no pudieron hacer los arreglos apropiados a su ambiente.

Colocar los detalles de la vulnerabilidad más reciente de un sistema en un tablero de boletines electrónicos de Usenet no sólo pondrá en peligro a otros sistemas, sino que puede exponer al que lo haga a demandas civiles por daños si se usa esa falla para penetrar algún sitio. Si preocupa la seguridad se debe recordar que se es parte de una comunidad. Hay que tratar de reforzar la seguridad de todos los miembros de la comunidad y recordad que más adelante puede necesitar ayuda de otros.

Información confidencial

Alguna información relacionada con la seguridad es apropiadamente confidencial. Por ejemplo, evitar que las contraseñas se han diseñando para utilizarlas confidencialmente, que no es lo mismo que una falla o una puerta trasera que le otorga al intruso el poder de súper usuario. Además, las contraseñas deben cambiarse periódicamente para que sigan siendo confidenciales.

Palabras finales: la administración de riesgos es cuestión de sentido común

La clave para tener éxito en el análisis de riesgos es identificar todas las amenazas posibles al sistema, pero sólo defenderse de aquellas que parecen ser amenazas reales.

Sólo porque la gente constituye el eslabón más débil no hay que olvidar otras salvaguardas. Las personas no son predecibles, pero penetra a través de un módem que no tiene contraseña es más barato que un soborno. Así que deben emplearse defensas tecnológicas cuando se pueda y mejorar la seguridad a través del personal educando a los colaboradores y usuarios.

Las defensas deben emplearse a fondo, con varios niveles para respaldarse cuando uno falle. Por ejemplo, se puede adquirir una segunda unidad de potencia ininterrumpida, o poner una cerradura en la puerta de la sala de cómputo aunque exista una cerradura en la entrada del edificio. Se pueden derrotar estas combinaciones pero el esfuerzo. Por lo menos se tiene la esperanza de que actúen más lentamente y que la vigilancia y las alarmas convoquen ayuda antes de que algo importante se pierda o se dañe.

Pensando en estos límites es importante acercarse a la seguridad informática con un conjunto de prioridades bien pensado. No es posible protegerse de todas las amenazas. A veces dejar que algo suceda en lugar de prevenirlo y después limpiar lo que haya pasado. Por ejemplo, puede ser más barato y menos complicado dejar que se interrumpa el servicio por fallas de potencia y luego reiniciar los sistemas que comprar una unidad de potencia ininterrumpida. Y no vale la pena defenderse de algunas amenazas porque son demasiado poco probables (una invasión de extraterrestres) o las defensas son demasiado difíciles (una explosión nuclear a 500 metros del centro de datos) o sencillamente son demasiado catastróficas y horribles (la administración decide cambiar todas las computadoras que usan UNIX por computadoras con un conocido sistema operativo para computadoras personales). La clave de una buena administración es saber de qué preocuparse y hasta dónde preocuparse.

Hay que decidir qué se quiere proteger y cuánto va a costar la prevención de esas pérdidas en comparación con el costo de recuperarse de las mismas. Luego, se tiene que decidir qué acciones y qué medidas de seguridad deben tomarse con base en una lista ordenada por prioridades de las necesidades más críticas. Es importante asegurarse de incluir no sólo las computadoras en lista: no hay que olvidar que las cintas de respaldo, las conexiones de red, las terminales y la documentación también forman parte del sistema, y representan una pérdida en potencia. La seguridad del personal, del sitio corporativo y de la reputación son también importantes y deben incluirse en sus planes.

Capítulo 2- Vulnerabilidades

2.1 Herramientas contra la Disponibilidad:

- Negar acceso a los recursos.
- Ataques los cuales son dirigidos para que los usuarios autorizados al sistema no puedan tener acceso a estos.
- Ataques de Negación de Servicio(DoS).
- Múltiples y coordinados ataques de servicios (DoS).

Exploits:

Categorías de exploits:

- Exploits a través del Internet
- Exploits de redes de área local
- Localmente (Servidor)
- Robo
- Engaño

Exploits a través del Internet:

- Ataques coordinados
- Robo de Sesiones(Hijacking)
- Robo de Información , passwds, etc(spoof)
- Ataque de tipo Relaying (Spam)
- Caballos de Troya
- Virus

Exploits a través de redes de área local:

- Capturadores de Tráfico(sniffing)
- Broadcast
- Acceso a los Archivos
- Exploits de control remoto
- Aplicaciones de robo de señal(hijacking)

Exploits Locales:

- Desbloqueo de Terminales
- Captura de Passwords
- Passwords en papel
- Desconexión de equipos
- Intento de explotación local de alguna programa para provocar un buffer overflow

Exploits fuera de línea:

- Copiado de Archivos passwd

- Bajar textos Cifrados
- Copiar grandes Cantidades de datos confidenciales

ROBO:

Acceso Físico y robo de :

- | | | |
|----------------|-----------|----------|
| • Datos. | • Llaves. | • Medios |
| • Equipo. | • Basura. | ópticos. |
| • Contraseñas. | • Cintas. | |

ENGAÑO:

- | | |
|---|---------------------------|
| • Convencer a la gente de poder brindar información que normalmente no debe proporcionar. | • Vía telefónica En-sitio |
| • Ingeniería Social | • Basura |
| | • Cámaras |

¿Que puede ser explotado?

- | | | |
|-------------|--------------------|---------------|
| • Todo | • Contraseñas | • Caballos de |
| • Puertos | • Puertas traseras | Troya |
| • Servicios | | • Virus |
| • Software | | |

Puertos:

Los puertos sirven como ventanas para que alguien pueda usarlos si desea acceder a los sistemas, entre mas puertos activos , mayor es el riesgo potencial de ser vulnerable a algún ataque dirigido al puerto.

Puertos más Comunes:

ftp:20,21
http: 80
SMTP: 25
SNMP.
DNS: 59.

Servicios:

Aplicaciones que se ejecutan en una maquina, procesan datos y peticiones, pero en algunos casos corren dichos servicios como administrador. Explotar un servicio puede proporcionar el control del administrador.

Servicios mas Comunes:

DNS
Sendmail
Finger
Telnet

Contraseñas:

La mayoría de los sistemas y programas del mismo vienen configurados con contraseñas por omisión que muy pocas veces la gente cambia, la mayoría de éstas son fáciles y triviales de adivinar, así como también los sistemas contienen cuentas sin contraseña y las contraseñas son raramente cambiadas.

Puertas Traseras:

Estas pueden ser accidentales o intencionales, es una forma común usada por los intrusos para poder forzar una entrada sin rastro a nuestro sistema y en algunos casos son estas puertas traseras puestas por los administradores y la mayoría de las veces olvidadas, dejando el hueco y riesgo potencial.

Caballos de Troya:

Tienen una función doble y oculta

Exploits de Propósito general:

IP Spoofing:

Ataca a un tercero robando la identidad de la dirección IP para hacerse pasar por otra IP que no le corresponde, la explotación implica la confianza entre los equipos.

Session Hijacking :

Este método mejor conocido como “Robo de terminales 2 no es mas que el robo y apropiación de una terminal válida de un usuario válido, requiere una conexión existente, una vez que los dos sistemas establecen una comunicación se toma dicha sesión , actuando de forma normal como si se hubiera establecido la sesión inicialmente. Esta es de forma transparente y sin rastro alguno en la comunicación, involucra un ataque de negación de servicios en algún momento contra la máquina donde se realiza el robo de sesión.

Ataque tipo DoS:

Un ataque de Negación de servicio se refiere a la negación de los recursos y servicios del sistema hacia los usuarios válidos, puede ser causada de forma deliberada o accidental afecta una gran variedad de sistemas operativos, incluyendo ruteadores, sistemas operativos, programas, etc.

Buffer Overflow:

Es un programa que envía datos desconocidos o no válidos a los programas, provocando un sobre flujo en el almacenamiento de los mismo abortando su ejecución, estos son causados por una deficiente programación.

Exploits Basados en Contraseñas:

- Ataque basados en diccionarios
- Ataque de Fuerza Bruta
- L0pthcrack

Otros:

- NT Crack
- John the ripper
- Cracker Jack

Ataques basados en Diccionarios:

Involucra una lista de passwords determinados ya conocidos o triviales, debido a que es muy común el poner alguna palabra clave o algún nombre, esta técnica se hace muy útil y sus porcentajes de poder ser exitoso son altas.

De igual manera usa la combinación y permutaciones de caracteres.

Ataque de Fuerza Bruta:

Trata con todos los passwords posibles hasta que es exitoso, Existe una batalla por los recursos del sistema, tiempo de consumo de CPU, memoria, etc

¿Como son almacenados los passwords en el sistema ?

Los passwords de los usuarios deben ser protegidos contra:

Moficiacion no autorizada

Borrado accidental

Acceso no autorizado.

2.2 Herramientas contra la Integridad:

Virus

¿Por qué llamarlos virus? La gran similitud entre el funcionamiento de los virus computacionales y los virus biológicos, propicia que a estos pequeños programas se les denominara virus.

Los virus de las computadoras no son mas que programas; y estos virus casi siempre los acarrean las copias ilegales o piratas. Provocan desde la perdida de datos o archivos en los medios de almacenamiento de información, hasta daños al sistema y, algunas veces, incluyen instrucciones que pueden ocasionar daños al equipo. Estos especiales:

- Son muy pequeños.
- Casi nunca incluyen el nombre del autor, ni el registro, copyright, ni la fecha de creacion.
- Se reproducen a si mismos.
- Toman el control o modifican otros programas.

Los científicos del area de la computacion discutieron por primera vez la posibilidad de un programa capaz de duplicarse a si mismo y extenderse entre las computadoras desde los 50's. Pero no fue sino hasta 1983 que un software de virus real fue creado, cuando un estudiante en la Universidad de California, Fredh Cohen, escribio una tesis de doctorado sobre el tema.

Motivos para crear un virus.

A diferencia de los virus que causan resfriados y enfermedades en humanos, los virus de computadora no ocurren en forma natural, cada uno debe ser programado. No existen virus benéficos. Algunas veces son escritos como una broma, quizá para irritar a la gente desplegando un mensaje humorístico. En estos casos, el virus no es más que una molestia. Pero cuando un virus es malicioso y causa daño real, ¿quién sabe realmente la causa? ¿aburrimiento? ¿coraje? ¿reto intelectual?. Cualquiera que sea el motivo, los efectos pueden ser devastadores. Los fabricantes de virus por lo general no revelan su identidad, y algunos retan a su identificación.

Clasificación

Virus de Macros/código fuente. Se adjuntan a los programas fuente de los usuarios y, a las macros utilizadas por:

Procesadores de palabras (word, works, wordperfect), hojas de cálculo (excel, quattro, lotus).

Virus Mutantes. Son los que al infectar realizan modificaciones a su código, para evitar ser detectados o eliminados (NATAS o SATAN, Miguel Angel, por mencionar algunos)

Gusanos. Son programas que se reproducen a sí mismos y no requieren de un anfitrión, pues se “arrastran” por todo el sistema sin necesidad de un programa que los transporte. Los gusanos se cargan en la memoria y se posicionan en una determinada dirección, luego se copian en otro lugar y se borran del que ocupaban, y así sucesivamente. Esto hace que queden borrados los programas o la información que encuentran a su paso por la memoria, lo que causa problemas de operación o pérdida de datos.

Caballos de Troya. Son aquellos que se introducen al sistema bajo una apariencia totalmente diferente a la de su objetivo final; esto es, que se presentan como información perdida o “basura”, sin ningún sentido. Pero al cabo de algún tiempo, y esperando la indicación programada, “despiertan” y comienzan a ejecutarse y a mostrar sus verdaderas intenciones.

Bombas de tiempo. Son los virus que realizan las funciones más parecidas a los virus biológicos, ya que se autoreproducen e infectan los programas ejecutables que se encuentran en el disco. Se activan en una fecha u hora programadas o cada determinado tiempo, contando a partir de su última ejecución, o simplemente al “sentir” que se les trata de detectar. Un ejemplo de estos es el virus del Viernes 13, que se ejecuta en esa fecha y se borra (junto con los programas infectados), evitando así ser detectado.

Infectores del área de carga inicial. Infectan los discos duros o discos flexibles, alojándose inmediatamente en el área de carga. Toman el control cuando se enciende la computadora y lo conservan todo el tiempo.

Infectores del sistema. Se introducen en los programas del sistema, por ejemplo **COMMAND.COM** y otros que se alojan como residentes en memoria. Los comandos del sistema operativo, como **COPY**, **DIR**, o **DEL**, son programas que se introducen en memoria al cargar el sistema operativo y es así como el virus adquiere el control para infectar todo disco que sea introducido a la unidad con la finalidad de copiarlo o simplemente para ver sus carpetas.

Infectores de programas ejecutables. Estos son los virus más peligrosos, porque se diseminan fácilmente hacia cualquier programa (hojas de calculo, juegos, procesadores de palabras). La infección se realiza al ejecutar el programa que contiene al virus, que en ese momento se posiciona en la memoria de la computadora y a partir de entonces infectará todos los programas cuyo tipo sea EXE o COM, en el instante de ejecutarlo, para invadirlos autocopiándose en ellos.

Aunque la mayoría de estos virus ejecutables “marca” con un byte especial los programas infectados –para no volver a realizar el proceso en el mismo disco-, algunos de ellos (como el jerusalem) se duplican tantas veces en el mismo programa y en el mismo disco, que llegan a saturar su capacidad de almacenamiento.

Desde la aparición de los virus informáticos en 1984 y tal como se les concibe hoy en día, han surgido muchos mitos y leyendas acerca de ellos. Esta situación se agravó con el advenimiento y auge de Internet. A continuación, un resumen de la verdadera historia de los virus que infectan los archivos y sistemas de las computadoras.

1939-1949 Los Precursores

En 1939, el famoso científico matemático John Louis Von Neumann, de origen húngaro, escribió un artículo, publicado en una revista científica de Nueva York, exponiendo su “Teoría y organización de autómatas complejos”, donde demostraba la posibilidad de desarrollar pequeños programas que pudiesen tomar el control de otros, de similar estructura. Cabe mencionar que Von Neumann, en 1944 contribuyó en forma directa con John Mauchly y J. Presper Eckert, asesorándolos en la fabricación de la ENIAC, una de las computadoras de Primera Generación, quienes contruyeron además la famosa UNIVAC en 1950.

John Louis von Neumann (1903-1957)

En 1948, en el laboratorio de la Bell Computer, subsidiaria de la AT&T, 3 jóvenes programadores: Robert Thomas Morris, Douglas Mcllory y Victor Vysotsky, a manera de entretenimiento crearon un juego al que denominaron ***CoreWar***, inspirados en la teoría de John Von Neumann, escrita y publicada en 1939.

Robert Thomas Morris fue el padre de Robert Tappan Morris, quien en 1988 introdujo un virus en ArpaNet, la precursora de Internet.

Puesto en la práctica, los contendores del CoreWar ejecutaban programas que iban paulativamente disminuyendo la memoria del computador y el ganador era el que finalmente conseguía eliminarlos totalmente. Este juego fue motivo de concursos en importantes centros de investigación como el de la Xerox en California y el Massachusetts Technology Institute (MIT), entre otros.

Sin embargo durante muchos años el CoreWar fue mantenido en el anonimato, debido a que por aquellos años la computación era manejada por una pequeña elite de intelectuales.

A pesar de muchos años de clandestinidad, existen reportes acerca del virus **Creep**, creado en 1972 por Robert Thomas Morris, que atacaba a las famosas IBM360, emitiendo periódicamente en la pantalla el mensaje: *"I'm a creeper... catch me if you can!"*. Para eliminar este problema se creó el primer programa antivirus denominado **Reaper**, ya que por aquella época se desconocía el concepto de los antivirus.

En 1980 la red ArpaNet del ministerio de Defensa de los Estados Unidos de América, precursora de Internet, emitió extraños mensajes que aparecían y desaparecían en forma aleatoria, a sí mismo algunos códigos ejecutables de los programas usados sufrían una mutación. Los altamente calificados técnicos del Pentágono se demoraron 3 largos días en desarrollar el programa antivirus correspondiente. Hoy día los desarrolladores de antivirus resuelven un problema de virus en contados minutos.

1981 La IBM PC

En agosto de 1981 la International Business Machine lanza al mercado su primera computadora personal, simplemente llamada IBM PC. Un año antes, la IBM había buscado infructuosamente a Gary Kildall, de la Digital Research, para adquirirle los derechos de su sistema operativo **CP/M**, pero este se hizo de rogar, viajando a Miami donde ignoraba las continuas llamadas de los ejecutivos del "gigante azul".

Es cuando oportunamente surge Bill Gates, de la Microsoft Corporation y adquiere a la Seattle Computer Products, un sistema operativo desarrollado por Tim Paterson, que realmente era un "clon" del CP/M. Gates le hizo algunos ligeros cambios y con el nombre de PC-DOS se lo vendió a la IBM. Sin embargo, Microsoft retuvo el derecho de explotar dicho sistema, bajo el nombre de MS-DOS.

El nombre del sistema operativo de Paterson era **"Quick and Dirty DOS"** y tenía varios errores de programación.

La enorme prisa con la cual se lanzó la IBM PC impidió que se le dotase de un buen sistema operativo y como resultado de esa imprevisión todas las versiones del llamado PC-DOS y posteriormente del MS-DOS fueron totalmente vulnerables a los virus, ya que fundamentalmente heredaron muchos de los conceptos de programación del antiguo sistema operativo CP/M, como por ejemplo el **PSP** (Program Segment Prefix), una rutina de apenas 256 bytes, que es ejecutada previamente a la ejecución de cualquier programa con extensión EXE o COM.

1983 Keneth Thompson

Este joven ingeniero, quien en 1969 creó el sistema operativo UNIX, resucitó las teorías de Von Neumann y la de los tres programadores de la Bell y en 1983 siendo protagonista de una ceremonia pública presentó y demostró la forma de desarrollar un virus informático.

Al año siguiente, el Dr. Fred Cohen al ser galardonado en una graduación, en su discurso de agradecimiento incluyó las pautas para el desarrollo de un virus. Este y otros hechos posteriores lo convirtieron en el primer autor oficial de los virus, aunque hubieron varios autores más que actuaron en el anonimato.

El Dr. Cohen ese mismo año escribió su libro “*Virus informáticos: teoría y experimentos*”, donde además de definirlos los califica como un grave problema relacionado con la Seguridad Nacional. Posteriormente este investigador escribió “*El evangelio según Fred*”, desarrolló varias especies virales y experimentó con ellas en un computador VAX 11/750 de la Universidad de California del Sur.

La verdadera voz de alarma se dió en 1984 cuando los usuarios el BIX BBS de la revista BYTE reportaron la presencia y difusión de algunos programas que actuaban como “caballos de troya” logrando infectar a otros programas. Al año siguiente los mensajes y quejas se incrementaron y fue en 1986 que se reportaron los primeros virus conocidos que ocasionaron serios daños en las IBM PC y sus clones.

1986 El comienzo de la gran epidemia.

En ese año se difundieron los virus *Brain*, *Bouncing Ball* y *Marihuana* y que fueron las primeras especies representativas de difusión masiva. Estas 3 especies virales tan sólo infectaban al sector de arranque de los diskettes. Posteriormente aparecieron los virus que infectaban los archivos con extensión EXE y COM.

El 2 de Noviembre de 1988 Robert Tappan Morris, hijo de uno de los precursores de los virus y recién graduado en Computer Science en la Universidad de Cornell, difundió un virus a través de ArpaNet, (precursora de Internet) logrando infectar 6’000 servidores conectados a la red. La propagación la realizó desde uno de los terminales del MIT. Cabe mencionar que el ArpaNet empleaba el UNIX, como sistema operativo. Robert Tappan Morris al ser descubierto, fue enjuiciado y condenado en la corte de Syracuse, estado de Nueva York, a 4 años de prisión y el pago de US \$10’000 de multa, pena que fue conmutada a libertad bajo palabra y condenado a cumplir 400 horas de trabajo comunitario.

Robert Tappan Morris Jr.

1991 La fiebre de los virus.

En Junio de 1991, el Dr. Vesselin Bontchev, que por entonces se desempeñaba como director del Laboratorio de Virología de la Academia de Ciencias de Bulgaria, escribió un interesante y polémico artículo en el cual, además de reconocer a su país como el líder mundial en la producción de virus da a saber que la primera especie viral búlgara, creada en 1988, fue el resultado de una mutación del virus Vienna, originario de Austria, que fuera desensamblado y modificado por estudiantes de la Universidad de Sofía. Al año siguiente los autores búlgaros de virus, se aburrían de producir mutaciones y empezaron a desarrollar sus propias creaciones.

En 1988 su connacional, el virus Dark Avenger o el “vengador de la oscuridad”, se propago por toda Europa y los Estados Unidos haciendose terriblemente famoso por su ingeniosa programación, peligrosa y rápida técnica de infección, a tal punto que se han escrito muchos articulos y hasta más de un libro acerca de este virus, el mismo que posteriormente inspiró en su propio pais la producción masiva de sistema generadores automáticos de virus, que permiten crearlos sin necesidad de programarlos.

1991 Los virus peruanos

Al igual que la corriente búlgara, en 1991 apareció en el Perú el primer virus local, autodenominado “**Mensaje**” y que no era otra cosa que una simple mutación del virus Jerusalem-B y al que su autor le agregó una ventana con su nombre y número telefónico. Los virus con apellidos como Espejo, Martínez y Aguilar fueron variantes del Jerusalem-B y prácticamente se difundieron a nivel nacional.

Continuando con la lógica del tedio, en 1993 empezaron a crearse y diseminarse especies nacionales desarrolladas con creatividad propia, siendo alguno de ellos sumamente originales, como los virus Katia, Rogue o F03241 y los polimórficos Rogue II y Please Wait (que formateaba el disco duro). La creación de los virus locales ocurre en cualquier pais y el Perú no podía ser la excepción.

1995 Los macro virus

A mediados de 1995 se reportaron en diversas ciudades del mundo la aparición de una nueva familia de virus que no solamente infectaban documentos, sino que a su vez, sin ser archivos ejecutables podían autoreplicarse infectando a otros documentos. Los llamados macro virus tan solo infectaban a los archivos de MS-Word, posteriormente apareció una especie que atacaba al AmiPro, ambos procesadores de textos. En 1997 se deseminó a través de Internet el primer macro virus que infecta hojas de cálculo de MS-Excel, denominado **Laroux**, y en 1998 surge otra especie de esta misma familia de virus que ataca a los archivos de bases de datos de MS-Access.

1999 Los virus anexados (attachments)

A principios de 1999 se empezaron a propagar los virus anexados a mensajes de correo, como el **Melisa** o el macro virus **Papa**. Ese mismo año fue difundido a través de Internet el peligroso **CIU** y el **ExploreZip**, entre otros muchos más.

A fines de Noviembre de este mismo año apareció el **BubbleBoy**, el primer virus que infecta los sistemas con tan sólo leer el mensaje de correo, el mismo que se muestra en formato HTML.

Resultará imposible impedir que se sigan desarrollando virus en todo el mundo, por ser esencialmente una expresión cultural de “*graffiti cibernetico*”.

¿Cómo evitarlos?

Sospecha de los programas activos todo el tiempo (residentes en memoria). Los virus tienen la mala costumbre de quedarse en memoria para realizar sus objetivos.

Sospecha de cualquier programa gratuito (shareware, freeware, fotos, videos, rutinas, parches) que bajes de Internet. Los fabricantes de virus colocan frecuentemente en estos sus nocivos productos.

Obten una lista de los virus más comunes y verifica contra esta lista cualquier programa nuevo que tengas.

Fijate en el tamaño de los archivos del sistema (COMMAND.COM principalmente). Sospecha si es diferente del original.

Haz una copia de la tabla de FAT y CMOS si te es posible. Te ahorrara mucho tiempo, dinero y esfuerzo el copiarla de nuevo si algun virus la daño.

Al estar buscando un virus, y si tienes disco duro, bloquea el acceso a este temporalmente.

Actualiza mensualmente tu antivirus. Si no tienes puedes conseguir una copia gratuita en varios sitios en internet.

Capítulo 3- Herramientas de seguridad

3.1 Algoritmo MD5

Uno de los comportamientos típicos de los intrusos al ingresar a un sistema después de inspeccionar quien se encuentra en el sistema en ese momento y que cosas interesantes existen en éste, es asegurar su regreso a través de abrir puertas traseras en el sistema.

Las formas típicas de hacer esto es creando cuentas o activando cuentas del sistema que no use el sistema actualmente, por ejemplo *uucp*, *bin*. Una forma más sofisticada es introducir programas troyanos al sistema, los cuales ocultan comportamientos del sistema.

Existen herramientas de intrusión, denominados *rootkit*, los cuales son arsenales completos de troyanos y programas espías.

El introducir un rootkit dentro del sistema genera *modificaciones* en el sistema y es esta característica la que usan las herramientas de verificación de integridad del sistema para avisar de la posible intrusión en la máquina.

¿Quién es el autor?

Ronald L. Rivest –Electrical Engineering and Computer Science- Massachusetts Institute of Technology.

¿Qué es el MD5?

MD2, MD4, y MD5 son algoritmos “message-digest” desarrollados por Rivest. Un message-digest son los algoritmos de **firmas digitales** que generan a partir de mensajes de longitud arbitraria una firma de longitud de 128-bits en los tres casos.

Aunque la estructura de su algoritmo son más o menos similares, una de sus diferencias principales está en el diseño sobre el cual fue pensado. Por ejemplo MD2 fue optimizado para máquinas de registro de 8 bits, mientras que MD4 y MD5 fueron pensados teniendo en mente sistemas de 32 bits.

MD5 fue desarrollado por Rivest en 1991. Básicamente incrementando mayor seguridad al algoritmo de MD4. Esencialmente MD5 lee datos y calcula un checksum de estos datos.

Van Oorschot y Wiener realizaron una investigación sobre las *colisiones* en una función hash, y estimaron el diseño de una máquina para búsqueda de colisiones en MD5 encontrando estas en un promedio de 24 días (además el costo estimado en \$10 millones de dólares en 1994)

Características:

- De los más rápidos actualmente conocidos
- Tiene una dispersión amplia
- Difícil de duplicar o generar colisiones
- No se necesita licencia para usar MD5 (no significa que no existan los derechos de autor)
- No existe patente sobre el MD5

Por estas características es una buena herramienta para verificar la integridad de los sistemas de archivos en un sistema.

3.2 Tripwire

Es un monitor de la integridad de los sistemas Unix, utiliza varias rutinas de **checksum/mensaje**, huellas seguras y hash/firmas para detectar cambios en archivos, además de rastrear determinados campos seleccionados de la administración del sistema.

Tripwire fue desarrollado por **Eugene Spafford** de : *Center for Education and Research in Information Assurance and Security*. Universidad de Purdue.

El sistema tripwire persigue los objetivos de **rastrear**:

- Cambios en los permisos de los archivos.
- Ligas.
- Tamaños en archivos y directorios.
- Groupid, userid's.

En 1992, Gene Kim y Eugene Spafford trabajando en el laboratorio **COAST** de la Universidad de Purdue desarrollaron el concepto de software Tripwire que podía ayudar a detectar cambios en la estructura de los archivos o directorios gracias al uso de algoritmos de firmas.

Dando como resultado la versión 1.2, que es la base de la versión **ASR** (Academic source release) que es depurada constantemente tanto por estudiantes como personas relacionadas con la industria y el gobierno.

En 1997 Gene Kim se asocia con W. Wyatt Starnes para formar lo que hoy se conoce como Tripwire Security Data, compañía comercial que proporciona una versión comercial de Tripwire con soporte técnico.

3.3 TCP Wrappers

El gran desarrollo que han tenido las redes han abierto a los usuarios posibilidades nunca antes imaginadas. Sin embargo también han abierto la posibilidad a muchos problemas.

A través de los servicios de red que nos permiten difundir y obtener información, en muchas ocasiones ha sido posible obtener acceso no autorizado a los sistemas, permitiendo a los intrusos utilizar los recursos e incluso realizar actos dañinos.

TCP Wrappers permite controlar y proteger los servicios de red, limitando el acceso como sea posible, y registrando todas las conexiones permitiendo de esta forma resolver problemas de forma más fácil. Fue desarrollado por **Wierze Venema**, de la *Mathematics and computing Science*.

¿Qué es un **Wrapper**?

Un Wrapper es un programa para controlar el acceso a un segundo programa. El wrapper literalmente cubre la identidad del segundo programa, obteniendo con esto un mayor nivel de seguridad.

Los wrappers son usados dentro de la seguridad en sistemas UNIX. Estos programas nacieron de la necesidad de modificar el comportamiento del sistema operativo sin tener que modificar su estructura.

Características:

- La seguridad lógica está concentrada en un solo programa, los wrappers son fáciles y simples de validar.
- El programa protegido se mantiene como una entidad separada, éste puede ser actualizado sin necesidad de cambiar el wrapper.
- Los wrappers llaman al programa protegido mediante la llamada al sistema estándar **exec()**, por lo que se puede usar un solo wrapper para controlar el acceso a diversos programas que se necesiten proteger.

¿Qué es **TCP Wrappers**?

TCP Wrappers es una herramienta libre basada en el concepto de Wrapper que nos sirve para monitorear y controlar el tráfico que llega por la red. Esta herramienta ha sido utilizada exitosamente en la protección de sistemas y la detección de actividades ilícitas.

Historia.

La historia del TCP Wrappers se remonta a 1990, cuando una de las máquinas de la Universidad de Eindhoven en Holanda era objeto de ataques por un hacker alemán, que seguido obtenía privilegios de **root**. Cuando obtenía dichos privilegios introducía el comando **"rm -rf /"**.

El comportamiento destructivo de este individuo hacía muy difícil averiguar que estaba pasando, dado que **"rm -rf"** remueve cualquier huella. Una noche **Venema** notó que

el intruso lo observaba a través de la red mediante el comando “finger” el cual casi nunca mantiene un registro de su uso, por lo que fue el comienzo del diseño de un sistema que pudiera detectar y registrar información de las conexiones realizadas.

Esquema de trabajo

Los servicios de red se basan en el modelo Cliente servidor. Por ejemplo, cuando alguien usa el comando “telnet” para conectarse a un servidor, el demonio del proceso “*telnet*” dentro del server es ejecutada actuando así como servidor, dando al usuario la capacidad para poder acceder al sistema.

Lo más común en sistemas de tipo servidor es correr un “*demonio*” que espera por cualquier clase de conexión a través de la red. Cuando una conexión tiene lugar, este *demonio* (usualmente llamado *inetd* en Unix) corre el servicio apropiado y regresa a su estado latente, en espera de otras conexiones.

El truco de TCP Wrappers consiste en hacer un intercambio, se mueve el programa servidor de red a otro lugar, y se instala un programa trivial en lugar del servidor original de red.

Siempre que una conexión tiene lugar, este programa registra el nombre del servidor remoto y entonces corre el servicio de red correspondiente.

Control de acceso.

TCP Wrappers hace uso de dos archivos de configuración para controlar los accesos a los servicios de (*/etc/hosts.allow*, */etc/hosts.deny*).

Hosts.allow- Si éste existe, permite definir las reglas de control para las máquinas y servicios autorizados.

Hosts.deny- Si éste existe, permite definir las reglas de control para las máquinas y servicios **NO** autorizados.

El demonio “*tcpd*” al recibir una petición de servicio verifica los archivos de control línea por línea empezando por el *hosts.allow* y terminando por el *hosts.deny*. La búsqueda se suspende cuando una regla de control de acceso es activada.

Cada uno de los archivos de control de acceso consiste de cero o más reglas de control con el siguiente formato:

List-daemons: list-hosts [:comando-shell]

Donde:

List-daemons. Es una lista de los nombres de los demonios de red (servicios).

List-hosts. Es una lista de nombres, direcciones o patrones de hosts que serán comparados con el nombre o dirección del host cliente que realiza la petición del servicio.

Comando-shell. Es un comando que podemos ejecutar, normalmente se utiliza para implementar anzuelos.

Comodines del lenguaje de control de acceso.

ALL. Si este comodín aparece en la lista de demonios de red y/o en la lista de host dentro de una regla de control de acceso, entonces indica la coincidencia con cualquier nombre de demonio de red y/o hosts.

LOCAL. Este comodín coincide con cualquier cadena que no contenga el carácter “.”, y su principal uso se encuentra en la lista de hosts.

Operador EXCEPT. Es un operador que tiene un uso de la siguiente forma: lista_1 **EXCEPT** lista_2; lo cual indica todo aquello de la lista_1 excepto lo de la lista_2.

Patrones del lenguaje de control de acceso.

- Una cadena que comience con el carácter de punto (.).
.dominio1.ulsal.mx
- Una cadena que termine con el carácter de punto.
132.111.122.
- Una expresión de la forma “n.n.n.n/m.m.m.m” es interpretada como un patrón “dir_red/mascara”, es decir, la dirección de una máquina cliente coincide si **dir_red** es igual al **AND** lógico de la dirección y la máscara.
132.111.222.0/255.255.255.248

3.4 Secure Shell

Secure Shell (ssh) es un programa que permite realizar conexiones con otras máquinas a través de alguna red, ejecutar programas en una máquina remota, y mover archivos de una máquina a otra de forma segura. Ssh provee **fuerte autenticación** y comunicación segura sobre un canal inseguro. Este es un reemplazo de los comandos **rlogin, rsh, rcp y telnet**.

Adicionalmente, Ssh provee seguridad para conexiones de servicios **X Window** y envío seguro de conexiones arbitrarias TCP.

Los servicios **r** (**rlogin, rsh, rcp**) son vulnerables a cierto tipo de ataques. Cualquiera que tenga acceso de **root** a una máquina en la red, o acceso físico a la red de cómputo, puede obtener acceso no autorizado por una gran variedad de formas, además de poder obtener información de ingresos a máquinas “escuchando” (sniffer) el tráfico sobre la red, esto incluye los **passwords**.

El sistema X window tiene varias vulnerabilidades. Con Ssh, se puede crear una sesion X remota segura que permanece transparente al usuario.

Ssh protege de los siguientes tipos de ataques:

- **IP spoofing**, donde un host remoto envia paquetes a otro host aparentando ser un tercer host.
- Ruteo de IP spoofing, donde un host puede pretender que un paquete de IP proviene de otro host valido.
- DNS spoofing, cuando un ataque provoca que los registros de servidor de nombre se pierda.
- Interseccion de passwords en claro, u otra informacion intercambiada entre dos hosts.
- Manipulacion de datos que se encuentran en intercambio entre hots.

Protocolo de Secure Shell

Caracteristicas del protocolo de **autenticacion**:

- Mecanismo de negociacion y autenticacion.
- Soporte de multiples mecanismos de autenticacion.
- Soporte de banner pre-autenticacion.
- Autenticacion basada en llave de cliente.

Caracteristicas del protocolo de **conexión**:

- Paso de variables de ambiente
- Agente de autenticacion
- Control de flujo y señales.

Caracteristicas del protocolo de **transporte**:

- Corre sobre TCP/IP, pero puede trabajar sobre UDP.
- Manejo de **regeneracion de llaves**.

- Verificación de identidad del cliente de servicio utilizando llave pública de cifrado.
- Negociación a través de mecanismo de **secreto compartido**.

¿Qué algoritmos de cifrado utiliza ssh?

Ssh proporciona las siguientes opciones de cifrado:

- DES
- IDEA
- TwoFish
- 3DES
- BlowFish

SSH utiliza

los siguientes cifrados para **autenticación**:

- RSA
- DSA

Ssh autentifica utilizando uno o más de los siguientes métodos:

- Llave pública (Rsa o DSA).
- Password
- Kerberos
- `.rhosts` o `/etc/hosts.equiv`

Cuando inicia una sesión, ssh establece comunicación con el servidor de ssh (**sshd**) y posteriormente intenta autenticar al usuario utilizando la siguiente secuencia:

Primero, si la máquina local del usuario se encuentra en los archivos */etc/hosts.equiv* o */etc/ssh/hosts.equiv* del servidor remoto y además el login del usuario es el mismo en ambas máquinas, la entrada del usuario es permitida inmediatamente.

Segunda, si en el **home** del usuario de la máquina remota existe alguno de los archivos *.rhosts* o *.shosts* y este archivo contiene una línea con el nombre de la máquina cliente y el login del usuario, cumpliéndose estos requisitos se permite la entrada del usuario.

Estos dos métodos presentan *riesgos de seguridad*, estos riesgos pueden ser reducidos si se utiliza una combinación de estos junto con una autenticación basada en RSA. Esto significa que el usuario solo obtendrá el acceso si la máquina es reconocida por el */etc/hosts.equiv*, */etc/ssh/shosts.equiv*, *.rhosts*, *.shosts*, y además puede verificar la llave del hosts cliente.

Un tercer método de autenticación se basa en *Autenticación RSA*. El servidor conoce la llave pública y solo el usuario conoce la llave privada. El archivo *\$HOME/.ssh/authorized_key* contiene una lista de las llaves públicas de los hosts que son permitidos tener acceso a este servidor.

Ssh implementa el protocolo automático de autenticación RSA. Cuando un usuario intenta tener acceso a esta máquina el programa ssh transmite al servidor la llave pública esperando que sea utilizada para la autenticación, el servidor verifica si la llave es permitida, y si esto es así, envía un número generado de manera aleatoria utilizando esta llave pública, este número solo puede ser descifrado con la llave privada, si esto lo logra el hosts cliente la comunicación se establece.

El usuario crea un par de llaves ejecutando *ssh-keygen*.

Versiones de Secure shell

Hay dos protocolos desarrollados sobre ssh :

SSH1. Este protocolo se encuentra en Unix y actualmente puede ser utilizada para propósitos no comerciales.

SSH2. Provee licencias más estrictas que SSH1, la versión de Unix y algunas de windows pueden ser utilizadas libremente por instituciones educativas bajo una licencia expresa.

Cabe destacar que los algoritmos RSA e IDEA, que son utilizados en ssh, son reclamados como patentes en diferentes países, incluyendo US, así que su uso puede ser legal o no, todo dependiendo de la legislación.

3.5 PortSentry.

Lo primero que requiere un atacante para acceder a un sistema de forma ilícita es recopilar información acerca de este. Todo puede serle útil; sistema operativo, servicios que se ofrecen, versión de los programas que se tienen, etc. Cualquiera de estos datos puede ser suficiente para que su ataque sea exitoso.

La manera más común en que un atacante intenta obtener información acerca de nosotros es por medio del barrido de puertos, es decir, intenta conectarse a cada uno de los puertos que tiene abiertos nuestro servidor, anotando que es lo que tiene activo y

analizando dicha informacion. Una de las herramientas más comunes para realizar barrido de puertos es ***nmap***.

Cabe mencionar que ***nmap*** no es una herramienta utilizada exclusivamente por atacantes; nosotros como administradores de sistemas podemos usarla para barrer nuestras mpaquinas buscando servicios abiertos.

Los programas barredores de puertos se han vuelto muy sofisticados y cada vez es más difícil detectarlos por diferentes estrategias que emplean, sin embargo, existe un excelente programa dedicado precisamente a encontrar los patrones utilizados al rastrear puertos y permite tomar la accion que le indique el administrador del sistema: ***Portsentry***.

Portsentry es un programa libre utilizado para monitorear los puertos que le indiquemos que deben permanecer siempre inactivos. En caso de llegar una conexión a uno de ellos puede marcarlo en la bitacora del sistema, bloquear toda la comunicación con la maquina identificada como agresora y/o correr un comando externo.

Modo de operación.

Portsentry tiene varios modos de operación, el más comun es el ***modo clasico***, donde le decimos que escuche determinados puertos TCP y UDP que son poco solicitados, los cuales son especificados con las opciones ***UDP_PORTS*** y ***TCP_PORTS*** dentro del archivo de configuracion. Portsentry abre el socket de los puertos especificados sin proporcionar servicio alguno, registrando cualquier intento de conexión y tomando la accion que le definamos.

Existe otro modo llamado ***stealth***. En este modo Portsentry abre ***sockets crudos***, lo que le permite detectar una mayor cantidad de barrido tal como ataques de conexión normal, SYN/half-open, FIN, NULL y XMAS. Este odo es experimental, por lo cual no funcionara en todos los sistemas.

Por ultimo existe un ***modo avanzado***, tambien considerado hasta cierto punto dentro de la categoria ***stealth***. En este modo, portsentry no abre ningun puerto sino que le pide al kernel que le notifique si llega alguna peticion a algun puerto menor al especificado en las opciones ***ADVANCED_PORTS_TCP*** y ***ADVANCES_PORTS_UDP***.

La ventaja de este modo es que los puertos aparentemente no estaran escuchando dado que no estan realmente abiertos. Sí ejecutamos el comando “netstat -na” el sistema no reportara los puertos que está escuchando al igual que el ***modo stealth***. Esto puede simplificar un tanto nuestro trabajo como administradores.

En el modo avanzado las opciones ***ADVANCED_EXCLUDE_TCP*** y ***ADVANCED_EXCLUDE_UDP*** nos permitira excluir algunos puertos que sean particularmente ruidosos tal como el usado por el servicio de red SMB (red de microsoft).

El modo avanzado es mucho más sensible que el modo clásico dado que escucha a muchos más puertos por medio del kernel, por lo que es necesario configurarlo con cuidado ya que de otra forma podría causarnos una negación de servicio.

Registro en bitacoras.

Portsentry además de generar bitacoras en el sitio donde fue instalado, utiliza **syslog** para reportar toda la información generada por la ejecución del programa. El administrador puede acceder a esta información típicamente en el archivo “*messages*” o donde se lo hayamos especificado en el ***syslog.conf***.

Portsentry es excelente para evitar ciertos ataques al sistema, sin embargo, como administradores de sistemas, debemos de analizar las bitacoras generadas y llevar registro de quien y cuando intento barrer nuestros puertos. Además, solo leyendo las bitacoras sabremos si portsentry funciona como deseamos.

3.6 PGP. (Pretty Good Privacy)

Hace 25 años el intercambio de mensajes cifrados era un privilegio reservado solo para militares, diplomáticos y servicios secretos.

Al paso de los años, el cambio de las leyes y políticas gubernamentales y el nacimiento de modernos sistemas de cifrado de llaves públicas establecieron las bases para acercar esta potente tecnología a una gran cantidad de gente; pero no fue hasta 1991, con el lanzamiento del programa PGP, que hizo por fin realidad el acceso del público a las ventajas que proporcionan estos mecanismos al implementarse en el correo electrónico.

Problemas con el correo electrónico.

Los mensajes no pueden ser entregados y acaban en el buzón del administrador del sistema de correo, que puede o no leerlo.

En otras ocasiones, alguien puede interferir el mensaje en su largo trayecto por la red. Además de leerlo, puede incluso modificarlo y reenviarlo a su destino. Las consecuencias pueden ser variadas.

También existen determinadas personas u organizaciones que escanen de forma simple y automatizada grandes volúmenes de mensajes, quizás buscando información como números de tarjetas de crédito, etc.

El uso de PGP en el correo electrónico aporta significativas ventajas que no conviene pasar por desapercibido:

Evita todos los inconvenientes antes señalados.

Hace valer el derecho constitucional a la intimidad de nuestras comunicaciones.

Contribuye a que el cifrado de los mensajes se convierta en una practica comun en internet.

Pgp garantiza el secreto de las comunicaciones electronicas para todos los ciudadanos, y no solo para los servicios secretos de las grandes potencias o de los grandes delinquentes. Por ello, la legalidad de su empleo varia en los diferentes paises, en funcion de cómo se respeten las libertades en ellos. Asi, en Estados Unidos, el pais de origen del creador de PGP, su uso es legal pero se prohíbe la exportacion del programa a otros paises al considerarse la criptografía al mismo nivel que el armamento. Es por esto que nunca se debe obtener PGP en ningun sitio de este pais y tenemos que utilizar la version **Internacional** que fue desarrollada precisamente para disfrutar de PGP en el resto del mundo.

En México, la consitucion reconoce el derecho al secreto en las comunicaciones privadas, y el nuevo codigo penal extiende ese derecho al correo electronico. Por tanto, el uso de PGP es legal en este Pais. Esta es, por cierto, la situacion habitual en el mundo libre.

Historia.

La llegada de las computadoras comienza a complicar las cosas en el mundo; asimismo, la segunda guerra mundial hace que los paises en coentienda se interesen en la criptografía, desarrollando maquinas de cifrado entre las que podrian destacarse las maquinas **Enigma** del ejercito aleman. Los algoritmos empiezan a utilizar llaves aleatorias, y gracias a diversos avances en la investigacion y el incremento tan rapido de las potencias del cálculo se ha llegado a algoritmos por ahora indestructibles como IDEA (International Data Encryption Algorithm), diseñado en el Instituto ETH de Zurich en 1990 por James L. Massey y Xuejia Lai.

Paralelamente surge otro tipo de criptografía, la criptografía de llave publica. Su historia comienza en 1976, cuando **Whitfield Diffie y Martin Hellman** desarrollan el algoritmo DH (Diffie-Hellman) y DSS (Digital Signature Estándar) para firmas.

El año siguiente, en 1977, Ron Rivest, Adi Shamir y Leonar Adleman, del **MIT** diseñan un nuevo algoritmo muy potente, el **RSA**. Y tan potente era que la Agencia de Seguridad Nacional (NSA) del gobierno americano les sugiere no publicarlo. Haciendo caso omiso y sobre todo por temor a que mas adelante no se les sugiriese, sino que les prohibiese, publican el algoritmo en la revista Scientific American.

El algoritmo es patentado a nombre de la compañía RSA Data Security Inc. Siendo valida esta patente en Estados Unidos y Canada.

Unos años despues Ron Rivest diseña un algoritmo para producir resúmenes cifrados de mensajes, el MD5, siendo utilizados para comprobar que los mensajes no han sido alterados.

En 1991 empieza a extenderse el rumor en Estados Unidos que el gobierno quiere prohibir el empleo de la criptografía en líneas de comunicación por ello, el programador Phil R. Zimmermann, combinando el mejor algoritmo existente de llave única **IDEA**, con el mejor de llave pública, el **RSA**, y añadiendo el MD5 para las firmas digitales, crea el programa PGP y lo distribuye como freeware por decenas de BBS's.

Un tiempo más tarde surgiría el conflicto. Alguien envió en junio de 1991 a varios grupos de USENET una copia de PGP, y así empezó a distribuirse y utilizarse fuera de Estados Unidos, incumpliendo la legislación ITAR del Departamento de Estado Americano sobre exportación de armas.

Se llegó a una solución doble. Se crearon dos versiones paralelas de PGP, las de uso exclusivo para Estados Unidos y Canadá, que emplean una biblioteca de funciones, la RSAREF, de RSA Data y que no son exportables a otros países, encargándose el MIT de la distribución gratuita del PGP, y las de uso internacional (identificadas añadiendo al número de versión una "i"), que emplean la biblioteca MPILIB de Zimmermann y que desde la primera versión salida de Estados Unidos han sido desarrolladas en Europa para ahorrarle más quebraderos de cabeza a Zimmermann.

¿Qué es PGP?

PGP es el programa más extendido y acreditado para el cifrado del correo electrónico. En la práctica, utilizarlo equivale a dotar al correo de valores añadidos del máximo interés:

- Confidencialidad
- Integridad
- Autenticación

En pocas palabras: con PGP se dejan de sufrir las múltiples carencias del correo electrónico ordinario y se pasa a disfrutar de todas las ventajas del correo electrónico seguro.

PGP versión Internacional es un programa Libre para usos no comerciales. La instalación de PGP en la computadora aporta nuevas funciones con las cuales nadie podrá leer su correo electrónico (aparte del destinatario que indique) e impedirá que alguien pueda suplantar su identidad en Internet, pues sus mensajes llevarán su propia y exclusiva firma digital, la cual es un bloque de caracteres que acompaña a un documento o archivo, acreditando quien es su autor (autenticación) y que no ha existido ninguna manipulación posterior de los datos (integridad).

Actualmente PGP es soportado en las siguientes plataformas:

- MS-DOS
- MacOS
- Windows
- Unix
- Amiga

- Atari
- OS2

Modo de operación de PGP

Supongamos que queremos enviar un mensaje que nadie excpeto esa persona pueda leerlo. Entonces para hacer esto debemos de cifrar el mensaje, lo que significa revolverlo de una forma muy complicada, con el fin de que resulte ilegible para cualquiera que no sea esa persona. Colocamos en el mensaje una **llave criptografica** para cifrarlo y la persona que lo reciba debe de utilizar la misma llave para descifrarlo

A este tipo de mecanismo convencionales se les llama ***criptosistemas de llave unica*** y hacen uso necesariamente de un canal seguro para poder transferir la informacion.

Existe otro mecanismo llamado ***criptosistema de llave publica***, donde todo el mundo tiene dos llaves complementarias, una revelada publicamente y otra secreta (llamada tambien ***llave privada***). Cada llave abre el codigo que produce la otra. Saber cual es la llave publica no sirve para deducir la llave secreta correspondiente. La llave publica puede publicarse y distribuirse ampliamente por una red de comunicaciones. Este protocolo proporciona intimidad sin necesidad de ese canal seguro que requieren los criptosistemas de llave unica.

Cualquiera puede utilizar la llave publica de un destinatario para cifrar un mensaje y el empleara su llave secreta correspondiente para descifrarlo. Solo el podra hacerlo, porque nadie más tiene acceso a esa llave secreta. Ni siquiera la persona que lo cifro podria descifralo.

Este criptosistema tambien proporciona autenticacion para mensajes. La llave secreta del remitente puede emplearse para mensajes. La llave secreta del remitente puede emplearse para firmar un mensaje de forma cifrada. Se genera una firma digital, que el destinatario puede comprobar al descrifrarla con la llave pública del remitente. De esta forma se prueba el verdadero origen del mensaje y que no ha sido alterado por nadie, ya que sólo el remitente posee la llave secreta que ha producido esa firma.

Estos dos procesos pueden combinarse para obtener ***intimidad y autenticacion*** al mismo tiempo si se firma primero el mensaje con la llave secreta y se cifra despues el mensaje firmado con la lalve publica del destinatario. El destinatario sigue estos pasos en sentido contrario al descrifrar primero el mensaje con su propia llave secreta y comprobar despues la firma con la llave pública del remitente. El programa lo hace automaticamente.

3.7 Análisis Forense

Se refiere a la aplicación de métodos protocolos y técnicas suficientemente legales para reunir, analizar y preservar la información computacional.

Algunos de los tips para la investigación y persecución son:

- Relaciones Personales
- No Complicarse las cosas
- Entender que sucedió antes de responder
- No aislarse.

Un equipo de cómputo puede morir físicamente o lógicamente. Esto cambia la forma en que debe hacerse el análisis forense pues si el problema es físico, quizá tenga que usarse un sistema alternativo para seguir operando, mientras se determina cuál fue el problema.

El problema.

El mayor problema de las intrusiones en sistemas es la pérdida económica que provoca, además de prestigio. A cualquier organización, al menos le implica destinar tiempo en recuperar el estado normal del sistema y la investigación posterior.

Lo primero que hay que hacer cuando ocurre un problema es tratar de tener toda la evidencia. Esto en gran medida depende de la configuración actual del sistema.

Tipos de delitos computacionales

- La computadora es el objetivo
- La computadora es el medio para comisión del crimen
- La computadora es usada como almacenamiento para el crimen.
- La computadora es usada como almacenamiento para el crimen.

Cuando la computadora es el objetivo.

- Interno o externo
- Una auditoría puede ayudar a saber
- Respalda todos los archivos.

Cuando la computadora es usada en la comisión de un delito

En este tipo de crímenes, la Pc contiene, generalmente, la evidencia más importante, pero no tiene información externa que ayude como proxies, por omisión, guarda sólo lo mínimo en las bitácoras.

Ejemplo:

Windows NT/2000

- Archivos .evt (event viewer)
- No todas las aplicaciones usan Event Viewer
- IIS guarda sus bitácoras en WINNT\system32\LogFiles

- Proxy guarda sus bitácoras en WINNT\system32\msplogs
- NT, por omisión, guarda sólo lo mínimo en las bitácoras.

Cuando se usa como almacenamiento:

Este caso es, ciertamente menos comunes. Pero, en una computadora usada sólo como almacenamiento, podemos encontrar:

- Detalles de actos criminales
- Registro Financieros.
- Correspondencia, etc.

Desconectar o No desconectar.

No siempre es necesario ni adecuado desconectar la máquina de inmediato de la red. Es necesario evaluar, según la situación, que resulta más útil. Probablemente haya nuevos intentos de intrusiones que permitan esclarecer el caso.

La Investigación.

Recomendaciones:

- En la medida de lo posible, no borrar programas que no sepa exactamente que hacen.
- En la medida de lo posible, no modificar la configuración.
- Guardar las sesiones de auditorías
 - Unix: comando script

Bitácoras:

Unix:

El directorio de las bitácoras depende del sabor de Unix de que se trate. Puede estar bajo */var/log*, */var/adm*.

- | | |
|--------------------------------------|-----------------------------|
| • Syslog | • Sulog |
| • Historicos del shell
(.history) | • Bitácoras de los demonios |
| • Lastlog | |

<i>/var/run/utmp</i>	Sesiones actuales
<i>/var/log/wtmp</i>	Históricos de sesiones
<i>/var/log/btmp</i>	Historial de conexiones falliadas
<i>/var/log/messages</i>	Mensajes del syslog

/var/log/secure Acceso y autenticación

Utmp:

- Tiene información de las sesiones exitosas que se hicieron (Usuario, terminales, hora y host remoto)
- La Información está en formato binario
- Puede usarse who, finger o users para observarla

Wtmp:

- Similares a Utmp
- Este archivo es utilizado por who y last

Btmp:

- Registro de los intentos de acceso fallido
- Archivos binarios, similares a wtmp el comando lastb utiliza el contenido de este archivo.

Windows:

En Windows, se puede buscar información sobre las conexiones de red en el Registry.

Localización de Evidencia:

Algunos lugares donde se puede hallar evidencia son:

- Discos Duros
- Cinta
- RAM
- Bitácoras
- Debajo del teclado, etc.
- En particular , si se usan cintas, son muy útiles para encontrar evidencia que haya podido ser eliminadas del disco duro.

El problema de esto es que cuando la intrusión al sistema fue bien realizada, es muy difícil hallar la evidencia en la bitácoras o en los directorios de archivos del sistema, bibliotecas, etc.

Evidencia Oculta:

- Binarios Alterados para no mostrar la información verdadera



- Binarios alterados para destruir información.
- Cifrado de Datos.
- Directorios Ocultos
- Archivos ocultos
- Archivos renombrados para ocultar su verdadero propósito
- Bases de datos con nombres de programas
- Programas como archivos comunes
- Documentos como archivos de gráficos

En muchas ocasiones, el éxito o fracaso de una investigación depende de la prontitud con que se haga .

Las bitácoras, regularmente no se mantiene por mucho tiempo. Hay que determinar y entender qué pasó antes de iniciar la investigación.

Apendice A - Historias de hackers y crackers

Desde que Bell inventó el teléfono en 1876, inicio el mundo de las telecomunicaciones. En 1960 con el surgimiento de Arpanet, el mundo de las redes de computadoras empezó a formar parte de nuestra vida cotidiana, entonces surge el concepto del “Cyberespacio” termino acuñado en 1982 por William Gibson.

Originalmente las computadoras estaban aisladas entre si, y la única forma de acceder era mediante terminales que estaban directamente conectadas a ellas. La única preocupación de seguridad que existía era protegerse contra ataques internos, perpetrados por los mismos usuarios del sistema.

Sin embargo, hoy en día las redes de computadoras conectan entre si miles de computadoras en todo el mundo, haciendo posible el acceso a un sistema desde cualquier lugar del planeta.

Esto es particularmente cierto en redes de alcance mundial como lo es Internet, el gran auge de está en los últimos años a traído grandes beneficios, gran capacidad de comunicación, pero con ello trae consigo igual numero de problemas, lo que convierte a la seguridad en algo que debe preocupar a todos los usuarios, y ya no solo a los administradores de sistemas.

Ningún sistema es 100% seguro!

Hackers: Normalmente son personas con grandes deseos de obtener conocimiento sobre las computadoras, se dan a la tarea de irrumpir los sistemas para demostrar su capacidad.

Crackers: Personas que irrumpen en los sistemas para causar algún daño, o para obtener algún beneficio (económico o información confidencial)

Los inicios

Criptografía (Maquina enigma)

Enigma fue un sistema de cifrado desarrollado a principios de siglo en Alemania por Arthur Scherbius y usado en la Segunda Guerra Mundial para cifrar la información de los nazis, es aquí cuando las fuerzas aliadas se empeñaron en romper los mensajes cifrados con esta máquina, desde ese entonces empezaron a surgir las primeras computadoras, las cuales no tenían otro fin que no fuera el bélico; las computadoras al no tener red se consideraban físicamente seguras, pero con la llegada de las redes y la comunicación global se empieza a pensar en una “guerra cibernética”.

El mundo underground

En 1984 surgió la primer revista de hackers llamada 2600, editada por Emmanuel Goldstein, está es una revista técnica, cuyos principios son no robo, no fraude, no derechos

de autor, libertad. En 1085 esta revista tuvo problemas con el FBI por publicar información para crackear. Otro par de revistas de underground son Phrack y Rampart la cuál público como funcionan las “cajas azules” (intersección de llamadas).

En ocasiones se piensa que los hackers están mejor organizados que las personas que se dedican a hacer seguridad en los sistemas, he aquí algunos ejemplos de organizaciones de hackers.

Asociación de crackers Españoles

Legión of Doom

Atlanta three

The Hacker's Defense Fund.

Demented Digital Thugs

El clan de la desesperación.

El caso Clifford Stoll

Intrusión: Agosto 1986 (Lawrence Berkeley Laboratory)

Detección: Un error de 75c en la contabilidad del sistema, existencia de una nueva cuenta “hunter”, actividad en una cuenta dormida “sventek”.

Huecos explotados: en el editor de texto emacs

Objetivo del ataque: Obtener información militar (NSA, CIA, .mil), fuentes de sistema operativo y circuitos integrados.

Arresto: 29/06/1989

El gusano de internet

Aparición: 2/11/1988, se reproduce de máquina en máquina.

Daños: Carga las máquinas, se caen, y niega el acceso.

Víctimas: Sun3 y VAX (6'000 máquinas en total)

Huecos explotados: rsh, finger y sendmail

Responsable: Robert Tappan Morris.

Sentencia: 04/05/1990, \$10'000 dls de multa y 3 años de libertad provisional, 400 hrs de servicio comunitario.

El caso Fry Guy

Características: Ligas con LoD, Roba de códigos telefónicos. Cambia sueldos de amigos en el mainframe de Mc Donalds. Rutea llamadas telefónicas a “Tina”. Fraude por \$6'000 con tarjetas de crédito (Western Union). Amenaza con tirar AT&T el 4 de julio.

Arresto: Capturado el 1989, 16 años de edad, 44 meses a prueba 400 horas de servicio comunitario.

El caso Shimomura

Detección: 24/12/1994, desaparición de bitácoras.

Tipo de ataque: IP-Spoofing

Características: Amenazas telefónicas, Ingeniería Social

Daños: Robo de software, códigos telefónicos y números de tarjetas de crédito, Ataques a otras máquinas (TOAD, Well)

Arresto: Marzo 1996 (Kevin Mitnick) 2 años de prision

El caso Randal Schwartz (¿ataque?)

Detección: 21/10/1993, Mark Morrissey detecta que la cuenta “merlyn” corre crack atacando O’Reilly, Intel, SSD, Telpost.

Características: Instalación de una puerta trasera “gate”. Randal confiesa ser el responsable de ambos. No tiene autorización, va en contra de las políticas de Intel.

Daños: Uso de recursos de Intel, falta a las políticas de no acceso exterior, Exposición de máquinas de Intel. No se sabe si utilizó los passwords adivinados.

Proceso: 5 años de libertad condicional 480 horas de servicio comunitario. \$68’000 de multa, 90 días de cárcel, más de \$170’000 en gastos y abogados.

Apendice B. Datos Interesantes

Matriz de relación entre riesgos, controles y procedimientos de revisión

SISTEMA DE EVALUACIÓN DE RIESGOS Y CONTROLES

VALIDACIÓN DE PARÁMETROS

Control	Riesgos	Prueba
10000 Sistema de evaluación de riesgos y controles.		
11000 Controles generales.		
11100 Organización.		
11101 ¿Existe una adecuada segregación de funciones entre los usuarios y el departamento de PED donde se contemple la separación de funciones incompatibles (autorización de transacciones, ejecución de transacciones), corrección y reentrada de errores?	1. Acceso no autorizado a información bajo procesamiento. 2. Error en los datos difícil de detectar. 3. Cambios no autorizados.	1. Jerarquización externa. (Organización.) a) Revisar y analizar la estructura organizacional y la ubicación del departamento de PED, así como las funciones básicas del área. b) Observar operaciones actuales.

- Martell Aguirre, Felipe Antonio / “Evaluación de la seguridad del sistema operativo HP-UX mediante un estudio de penetración” / (Tesis ULSA)
- Alicia Benet Velez / “Análisis comparativo de los principales estándares de seguridad informática” / (Tesis ULSA)
- E. Santoyo / “Como prevenir la seguridad y protección del equipo de computo y su información” / (Tesis ULSA)
- Leonor del Socorro / “Elementos para la seguridad de datos” / (Tesis ULSA)
- Álvarez, Ángeles, Gutiérrez, Maldonado, Marín / “Implantación de un laboratorio de seguridad basado en software libre sobre plataforma Linux” / (Tesis ULSA)
- Castro, Díaz, Fremont, Meza, Zamores / “Métricas para evaluar la seguridad en sistemas Linux” / (Tesis ULSA)
- David J. Stang / “Network Security Secrets” / IDG Books
- William Stallings / “Network and Internetwork security” / Prentice Hall
- Chris Hare / “Internet y Seguridad en redes” / Prentice Hall
- Shimomura - Markoff / “Takedown” / El Pais – Aguilar
- Anonymous / “Maximum Security” / SAMS
- Anonymous / “Maximum Linux Security” / SAMS
- Manuel Mediavilla / “Seguridad en Unix” / AlfaOmega - Ra-ma



- Fuster, Dolores de la Guia, Hernández, Montoya, Muñoz / **“Técnicas Criptográficas de protección de datos”** / AlfaOmega - Ra-ma
- McClure, Sambray, Kurtz / **“Hacking Exposed”** / Mc Graw Hill
- Téllez / **“Derecho Informatico”** / Mc Graw Hill
- Barrios, Muñoz de Alba, Pérez / **“Internet y derecho en México”** / Mc Graw Hill
- Rosales / **“Determinación de riesgos en los centros de computo”** / Trillas
- Skoudis / **“Counter Hack”** / Prentice Hall
- Piatinni, Del Peso / **“Auditoria Informática”** / AlfaOmega - Ra-ma



Universidad La Salle

Directorio:

H. Raúl Valadez García, FSC
Rector

Ing. Edmundo Barrera Monsivais
Vicerrector Académico

Hno. Martín Rocha Pedrajo
Vicerrector de Formación

Ing. José Antonio Torres Hernández
Director de la Escuela de Ingeniería

Ing. Raúl Morales Farfán
Secretario de Talleres y Laboratorios

Jefatura del LCI
M. en C. Luis Manuel Aguillón Banda

Jefatura del Área Académica
Ing. Liliana Vicenteño Loya

Edición:

Isaías Calderón (HuiLo) calderon@lci.ulsal.mx

A todo el Laboratorio de Cómputo de Ingeniería
En La Universidad La Salle.

Gracias Mil

Laboratorio de Cómputo de Ingeniería
MMII



Contenido

Capítulo 1- Seguridad, ¿Para qué?	1
1.1 Introducción	1
<i>Nuevas tecnologías, nuevos métodos.</i>	2
<i>Software Antivirus.</i>	2
<i>Llave publica / Llave privada</i>	2
<i>Administración de la seguridad.</i>	3
<i>Kerberos</i>	4
<i>Seguridad en ambientes de computo distribuido.</i>	6
<i>¿Dónde están los riesgos?</i>	6
<i>¿Cuales son los problemas que enfrenta la seguridad en redes?</i>	6
<i>Dimensiones de protección.</i>	7
1.2 Políticas de red	7
<i>Confidencialidad</i>	8
<i>Integridad de datos</i>	8
<i>Disponibilidad</i>	9
<i>Consistencia</i>	9
<i>Control</i>	9
<i>Auditoría</i>	9
Capítulo 2- Vulnerabilidades	25
2.1 Herramientas contra la Disponibilidad:	25
2.2 Herramientas contra la Integridad:	28
<i>Virus</i>	28
Capítulo 3- Herramientas de seguridad	35
3.1 Algoritmo MD5	35
3.2 Tripwire	36
3.3 TCP Wrappers	36
3.4 Secure Shell	39
3.5 PortSentry.	42
3.6 PGP. (Pretty Good Privacy)	44
Apendice A - Historias de hackers y crackers	52
Apendice B. Datos Interesantes	55