UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE CIENCIAS

INTRODUCCIÓN DE SISTEMAS DE SEGURIDAD INFORMÁTICA EN EMPRESAS MEXICANAS

REPORTE DE TRABAJO PROFESIONAL

QUE PARA OBTENER EL TÍTULO DE:

LICENCIADA EN CIENCIAS DE LA COMPUTACIÓN

P R E S E N T A:

VIOLETA ZAVALA AYALA

TUTOR:
LIC. FRANCISCO SOLSONA CRUZ

2010

Índice

Alcance

La mayoría de las empresas que basan sus servicios en tecnologías de información, como los bancos, compañías de telefonía fija o móvil, etcétera, no eran conscientes de que es vital proteger la información manejada por su infraestructura. Hasta hace poco se empezó a trabajar en este sentido con la introducción de mecanismos de seguridad. Este trabajo aborda el desarrollo de tales mecanismos, lo cual implica no sólo la implementación, sino también estructurar un trabajo de equipo entre las diversas áreas que interactúan con los sistemas informáticos.

Se requiere de un análisis detallado de qué es lo que se desea cubrir, considerando que no tenga impactos en el funcionamiento de los sistemas, para que las empresas puedan seguir prestando sus servicios de manera normal. Además de este análisis, es necesario capacitar tanto a los administradores como a todas aquellas personas que tengan injerencia directa o indirecta en las plataformas, como las áreas operativas, las de mantenimiento, las de consulta, etcétera.

Podemos agrupar los elementos a considerar para establecer un esquema de seguridad en los siguientes tres componentes:

- Procesos.
- Tecnología.
- Gente.

En este trabajo se cubrirán estos tres grupos.

Introducción

La seguridad de la información es un elemento clave para toda organización en la actualidad; los riesgos que se enfrentan son muy altos. La existencia de redes interconectadas ha motivado la aparición y continua expansión de nuevas amenazas que afectan o atañen, directa o indirectamente, a las operaciones de negocio.

Con la llegada de los fraudes informáticos, una de las condiciones impuestas a quienes pretenden cotizar en la bolsa de Nueva York, es contar con estándares de seguridad en tecnologías de información (TI). Por esto se crea la Ley Sarbanes-Oxley (SOX), encargada, entre otras cosas, de regular todas las medidas de seguridad de la información de las empresas.

Partiendo de esta situación, las empresas empiezan a dar cierta importancia a la introducción de seguridad informática y se empiezan a crear áreas especializadas en este tema. Donde con la frase "cierta importancia" me refiero a que todavía no se tiene una cultura suficientemente arraigada de lo que es y lo que conlleva tal seguridad informática. Este trabajo mostrará lo que enfrenta una empresa para introducir algunos sistemas de seguridad informática de la forma más transparente posible.

Capítulo I

Organización de la seguridad

Cumpliendo con la ley SOX

Como se mencionó en la introducción, todas las empresas que cotizan en la bolsa de Nueva York están obligadas a cumplir con la ley SOX.

En este trabajo sólo se va a hacer un breve análisis de lo que se requiere para cumplir con lo relativo a la sección 404 (Revelaciones Financieras) en cuanto a procesos, tecnología y gente. Para cubrir este punto es necesario revisar los procesos que se tienen en las áreas de tecnología que intervienen en:

- Control del desarrollo o cambios a las aplicaciones.
- Control de acceso lógico (sistemas operativos, bases de datos y aplicaciones).
- Control de la operación (respaldos y recuperación, calendarizar tareas, administración de problemas y vigilancia).

Asimismo se deben definir controles que nos aseguren el correcto funcionamiento de estos procesos. Para poder definir estos controles debemos estar conscientes de los diferentes tipos que podemos tener:

- De prevención (manuales, automatizados y dependientes de TI).
- De detección (manuales, automatizados y dependientes de TI).
- De proceso, actividades cotidianas (manuales, automatizados y dependientes de TI).

Para poder obtener estos controles se debe seguir una metodología ya definida para no perdernos en el camino de la búsqueda; en este trabajo hablaremos de COBIT (*Control Objectives for Information and related Technology*) como metodología a usar.

COBIT menciona que una parte primordial de un proceso de alto nivel es mantener los siguientes criterios de control:

- Eficiencia.
- Eficacia.
- Confidencialidad.
- Integridad.
- Disponibilidad.
- Cumplimiento.
- Confianza en la información.

La implementación de sistemas de seguridad informática se basa primordialmente en cubrir los puntos de confidencialidad, integridad y disponibilidad de los procesos de una empresa.

Necesidad de una cultura de seguridad

Hay que estar conscientes de que las empresas no sólo deben proteger sus activos intangibles contra los ataques externos, sino que también deben tomar medidas para que los propios empleados no divulguen esta información de forma intencional o accidental.

Nos tenemos que enfocar en la gente que conforma una empresa: administradores, proveedores, operadores y gente que explota los sistemas. Es muy importante que todos ellos tengan presente que es necesario cumplir con un grado de seguridad para no comprometer la información que se almacena en los sistemas. Es importante mencionar que la gente es el eslabón más débil en un esquema de seguridad informática[1]; y esto se debe principalmente a que estamos acostumbrados a utilizar contraseñas fáciles de recordar, como nombres de familiares, mascotas, fechas de cumpleaños, etcétera, al igual que a proporcionar esta información a los compañeros de trabajo. Otra mala práctica que también sucede muy a menudo es el anotar las contraseñas en un papel y dejarlo en nuestro lugar de trabajo al alcance de otros. También queremos hacer de la forma más fácil y rápida nuestro trabajo sin importarnos que sea incorrecta o insegura.

[1] "El eslabón más débil es la gente y son muy pocas las pruebas que se hacen de ingeniería social", Sergio Raúl Solís, gerente de Advisory de Ernet & Young [20].

Para combatir estas debilidades de seguridad no basta con implementar mecanismos de defensa contra ataques internos, sino que es necesario impartir capacitación constante y permanente a los empleados de la organización sobre las prácticas y políticas de seguridad.

Implementación de estándares de seguridad

Introducción

Para llevar un control de todas las medidas de seguridad informática en una empresa es necesario contar con estándares en los cuales se mencionen las modificaciones que se tienen que realizar en un sistema para poder decir que se encuentra seguro. De igual forma, se debe tener un procedimiento para realizar estas modificaciones. El que se utiliza en este caso es el siguiente: se inicia con un reporte que muestre las vulnerabilidades que se presentan en un sistema, posteriormente se analiza junto con el proveedor de la solución la manera de cerrar estas vulnerabilidades. Teniendo toda esta información, se procede a la implementación de los estándares de seguridad de la empresa (aplicando sólo aquellos que no causen impacto negativo al funcionamiento del sistema). Para mantener y administrar estas nuevas funcionalidades en el sistema es necesario capacitar a los administradores.

Es importante saber cómo funcionan las diferentes aplicaciones que se ejecutan en una plataforma para dar cierto servicio. Como sabemos, las aplicaciónes pueden ser software de terceros, en este caso proveedores de servicios; estas aplicaciones pueden utilizar una base de datos y deben correr sobre un sistema operativo. En el caso de software de terceros, nosotros sólo nos vamos a enfocar en proteger los sistemas operativos y las bases de datos (para el alcance de este trabajo quedarán fuera las aplicaciones propiamente dichas, ya que como son de terceros no se tiene mucha influencia sobre éstas).

En este apartado nos centraremos en los pasos que se siguen para proveer de mecanismos de seguridad a las plataformas que se administran en la empresa.

Los sistemas operativos con los que se trabaja son: Solaris, HP-UX, RedHat, Suse y Windows. Para el caso de bases de datos, Oracle es la base

de datos con la que se está trabajando hasta el momento, ya que es la que predomina en las plataformas de la empresa.

Cabe mencionar que el esquema de seguridad del que hablaremos sólo se centra en la seguridad interna (procesos, tecnología y gente), por lo que no se hablará de seguridad perimetral.

Estándares de seguridad

Como se mencionó antes, es necesario disponer de un punto de partida para la introducción de mecanismos de seguridad, esto con el fin de que todas las plataformas tengan el mismo grado de seguridad. Para esto es de vital importancia contar con estándares que nos indiquen cuáles son los puntos necesarios que se deben cubrir para garantizar que una plataforma es segura.

Hay que ser claros en que no siempre se van a poder implementar todos los puntos, pues hay gran variedad de aplicaciones en las empresas que no los soportan, pero es importante implementar la mayoría de ellos.

En una empresa es necesario (de ser posible) contar con un área que sea la que defina estos estándares y los valide; y otra para que los implemente, de lo contrario, estaríamos siendo juez y parte de esta implementación. En nuestro caso contamos con la primera área (para la creación de los estándares), la cual es responsable de emitir documentos oficiales para la empresa, indicando las medidas que debe cumplir una plataforma para garantizar que está segura. Los documentos que se emiten están basados en diferentes estándares y recomendaciones dependiendo de la plataforma.

Por mencionar algunos, tenemos:

- Recomendaciones del CERT (Computer Emergency Response Team) para la parte de sistemas operativos.
- Recomendaciones del CIS (Center for Internet Security) para sistemas operativos y bases de datos.
- Recomendaciones provistas por SUN para la parte de sistemas operativos Solaris.

Análisis de vulnerabilidades

El primer paso que se sigue para la protección informática es realizar un reporte de las vulnerabilidades con las que cuenta la plataforma, y que los mismos empleados pudiesen explotar provocando un mal uso. El análisis se realiza mediante una búsqueda interna de fallas en la plataforma las cuales están normadas bajo la metodología de la OSSTMM (*Open Source Security Testing Methodology Manual*), así como también con el uso de un conjunto de herramientas cuya finalidad es obtener la información de las vulnerabilidades y huecos de seguridad existentes en los sistemas.

En una empresa pueden existir diversos tipos de análisis en los que se tenga que generar alguna clase de reporte de vulnerabilidades, como pueden ser:

- Reportes hechos por un tercero. En este caso se contrata a una empresa especializada en encontrar deficiencias en los sistemas y reportarlas para su mitigación (*hackers*[2] éticos).
- Reportes realizados por auditores internos. Se llevan a cabo cuando está próxima una auditoría; su principal función es encontrar, antes que el auditor externo, las deficiencias que se tengan en los sistemas.
- Reportes realizados por auditores externos. Son los que se realizan por una empresa certificadora para avalar el grado de seguridad con el que cumple una empresa y poder determinar si cumple con los estándares necesarios para otorgarle una certificación (por ejemplo: SOX).
- Por último, y estos son en los que nos enfocaremos, reportes hechos por el área de la empresa encargada de proteger las diversas plataformas. Este tipo de reporte se realiza para tener un punto de partida al momento de implementar mecanismos de seguridad en una plataforma.

En este análisis se revisan primordialmente los siguientes puntos:

[2] El término "*hacker*" designa a aquella persona apasionada por el conocimiento de los sistemas de cómputo. También se asocia a las personas que poseen elevados conocimientos de seguridad informática: en estos casos se suele distinguir entre "*White Hats*" (sombreros blancos, los buenos) o "*Black Hats*" ("sombreros negros", los malos o *crackers*), según sus acciones sean sólo intrusivas o además destructivas [10].

Sistema operativo

- Servicios que se encuentran habilitados.
- Controles de acceso para los servicios TCP/IP.
- Control de cuentas (grupos y cuentas declaradas, permisos de archivos y directorios, administración de sesiones, etcétera).
- Generación de registros (*logs*).
- Utilización de contraseñas robustas.
- Configuraciones de red.
- Administración de servicios.
- Mecanismos de conexión.
- Protección de información sensible.

Base de datos

- Control de cuentas (cuentas declaradas, administración de sesiones, administración de privilegios y roles, etcétera).
- Utilización de contraseñas robustas.
- Asignación de permisos a los objetos y esquemas.
- Métodos de conexión.
- Protección de puertos y mecanismos de conexión.
- Protección de información sensible.

Este análisis nos da el punto de partida y nos permite tener un panorama más amplio de las condiciones actuales de la plataforma.

Análisis de posibles modificaciones a los sistemas

Una vez realizado el reporte de las posibles vulnerabilidades con las que cuenta una plataforma, se analizan, junto con el proveedor de la solución, los cambios que se pueden aplicar a ésta. Es necesario que el análisis se lleve a cabo en conjunto con el proveedor y el área administradora, pues son ellos los que más conocen el funcionamiento y la operación de la plataforma.

Como no se tiene una conciencia arraigada de las medidas de seguridad, la mayoría de las veces este análisis termina siendo muy escueto y con poca información, por lo que muchas de las implementaciones de estas medidas de seguridad se omiten o se introducen mediante prueba y error.

Implementación de estándares

Con el análisis del reporte se lleva a cabo un listado de las actividades que se van a realizar (instalación y configuración de herramientas de seguridad; y modificaciones en los archivos de configuración propios del sistema operativo y base de datos).

Los puntos a asegurar están basados en los estándares de seguridad definidos para cada plataforma y en los hallazgos obtenidos en el análisis de vulnerabilidades.

Esta implementación se puede llevar a cabo de tres formas: una es que se realice antes de que sean instaladas las aplicaciones, es decir, cuando se tiene el sistema operativo y la base de datos solamente. Esta es la forma más recomendable, ya que en esta situación prácticamente se pueden implementar todas las recomendaciones y, en caso de que las aplicaciones requieran que se habiliten algunas, solamente se estarían habilitando las necesarias. Otra forma de llevar a cabo la implementación es cuando la plataforma ya tiene todo instalado pero no ha entrado a producción. Para estos casos, de igual manera se pueden hacer todas las configuraciones necesarias para cubrir los huecos que tiene la plataforma, sólo que en caso de tener algún problema en el funcionamiento no se puede saber exactamente qué fue lo que lo causó, ya que varias actividades están fuertemente relacionadas; éste es el caso del cierre de servicios. También es el caso cuando la plataforma no queda completamente protegida. La última forma de implementación es cuando la plataforma ya es productiva. Este es el caso más sensible que se tiene, ya que para validar las implementaciones que se llevan a cabo en los equipos es necesario que se reinicie el sistema operativo, la base de datos y algunas aplicaciones; esto con el fin de estar completamente seguros de que no se causó ningún impacto. Ya que los equipos se encuentran funcionando, es necesario llevar a cabo estos reinicios en ventanas de mantenimiento para que todas las áreas relacionadas estén enteradas de estos trabajos y conscientes de que habrá interrupciones en el servicio que provee la plataforma que se está implementando.

Una vez que se ha terminado la implementación de los estándares de seguridad, se debe ejecutar un ATP (*Accepted Test Plan*) de funcionalidad de las medidas de seguridad para asegurarse de que se encuentren trabajando correctamente.

Capacitación

Para que una plataforma siga cumpliendo con las medidas de seguridad se requiere que los administradores conozcan los cambios realizados a sus equipos y la forma de administrar estas nuevas configuraciones; es de vital importancia el intercambio de información entre implementadores y administradores.

Una vez que se asegura la plataforma, la parte implementadora toma un papel de administrador por cierto tiempo, en el cual se les va transmitiendo el conocimiento a los administradores, así como delegándoles la responsabilidad de las nuevas medidas de seguridad. Al igual que a los administradores, es necesario capacitar a la gente que opera y vigila estas plataformas, ya que aunque se trata de que estas implementaciones sean lo más transparentes posible, es difícil que no cambien las formas de operar los equipos. Es decir, se debe capacitar sobre la forma de acceder a los equipos, una cultura de generación de contraseñas robustas, control de la privacidad, etcétera.

Mantenimiento

Aun cuando se le transmita la forma de administrar los mecanismos de seguridad al área encargada de controlar las plataformas, es necesario que periódicamente se lleven a cabo auditorías sobre las medidas que fueron instaladas. Esto para verificar que sigan activas y válidas. En caso de que estas medidas ya no se encuentren activas o válidas se necesitarán implementar y configurar nuevamente.

Administración de perfiles para sistema operativo

Introducción

Cuando en una empresa se tiene gran cantidad de plataformas que deben tener una disponibilidad del 99.999% (como es nuestro caso), es necesario contar con áreas que estén operando y vigilando los equipos para evitar que se pierda el grado de disponibilidad que requiere la empresa a fin de

proveer el servicio a sus clientes. Para estas actividades el personal debe contar con todas las herramientas necesarias para su ejecución; una de estas herramientas es el acceso a los sistemas con cuentas capaces de ejecutar actividades de vigilancia y solución de problemas. Para la mayoría de los casos estas cuentas suelen ser las de administración del sistema (`root` para los equipos UNIX y `administrador` para los equipos Windows) y las cuentas propias de las aplicaciones que se ejecutan en los equipos. En este caso sólo nos ocuparemos de las plataformas UNIX, que es donde se tiene una herramienta para cubrir el punto que se tratará en este apartado.

Debido a lo crítico que pueden ser estas cuentas, es necesario disponer de mecanismos que nos ayuden a la delegación y segregación de actividades; además de que deben existir registros de todos los movimientos que se realizan con estas cuentas.

Una de las herramientas que provee todas estas características es **Power Broker**[3], en este apartado se hablará de cómo se hacen las actividades de delegación y segregación de actividades de tal forma que no se impacte la operación de los sistemas.

Funcionamiento

Para crear perfiles adecuados para la operación y supervisión de equipos sin tener que hacer uso de cuentas con demasiados privilegios (administrativas y aplicativas), hay que tener la capacidad de crear estos perfiles con el detalle que se muestra a continuación.

- Comando a ejecutar.
- Cuenta con la que se debe de ejecutar el comando.
- Argumentos válidos para el comando.
- Rutas autorizadas para ejecutar el comando.
- Horarios válidos de ejecución.
- Equipo donde se debe ejecutar el comando.

Además es necesario contar con registros que nos indiquen qué fue lo que se ejecutó en cierto momento y quién lo ejecutó.

[3] **Power Broker** es una solución de seguridad y registro de cuentas diseñada para el uso de protocolos de control de acceso en las plataforma Unix/Linux. Permite a los administradores delegar privilegios sin revelar las contraseñas de las cuentas sensibles.

El objetivo de todas estas características es hacer responsable a cada usuario por las acciones realizadas en los sistemas. Porque, como sabemos, si una gran cantidad de personas tienen acceso a los sistemas con las cuentas sensibles, es difícil identificar, en caso de algún uso inadecuado de éstas, quién fue exactamente la persona que lo realizó. Con estas características, las cuentas sensibles de las plataformas quedan protegidas de posibles errores o abusos; esto sin tener que limitar a los operadores en sus tareas diarias.

Con lo mencionado anteriormente, podemos decir que la finalidad de la creación de perfiles es:

- Segregar la funcionalidad de las cuentas sensibles de los equipos, permitiendo que varias personas puedan ejecutar sus actividades sin la necesidad de tener el acceso completo de estas cuentas.
- Crear registros de todas las actividades sensibles realizadas en las plataformas.

Análisis de requerimientos

El primer paso para introducir el uso de perfiles en una empresa es saber qué es lo que realmente necesita cada persona para realizar sus actividades. Inicialmente es difícil proporcionar a cada quien todos los comandos que requiere para realizar su trabajo, pero es mejor ir agregando los comandos faltantes y no dar desde un principio más herramientas de las necesarias. Para que los comandos faltantes sean los mínimos posibles, se necesita que tanto las áreas operativas como las administrativas se reúnan para llegar a un acuerdo acerca de las actividades que se deben realizar por cada área, llámense estas últimas operativas, de supervisión, obtención de información, resolución de problemas, etcétera.

Así es como en conjunto se llega a la definición de los perfiles necesarios, indicándose en cada uno qué usuarios pueden ejecutar qué comandos, con qué argumentos, con qué privilegios de cuenta se va a ejecutar el comando y en qué horarios.

Creación de políticas

Una vez definidos los perfiles necesarios, se debe proceder a la creación de las políticas que nos ayudarán a delegar las diferentes tareas que debe realizar cada área. Para crear estas políticas se requiere de programación en un lenguaje híbrido de C, Pascal y propietario de la herramienta. Como se mencionó en la parte de análisis de requerimientos, con este lenguaje se pueden granular las políticas al punto de validar argumentos, horarios, rutas de ejecución, etcétera.

Cuando se requiere crear una política, se crea un archivo donde se le va a indicar a la herramienta **Power Broker** cuáles van a ser las reglas que debe validar para ver si la solicitud que esta realizando el usuario debe ser aceptada o rechazada. Por estándar, estos archivos tienen la terminación `.conf` y existe un archivo principal de políticas `/etc/pb.conf` en el cual se van agregando todas las políticas que se van generando.

La manera en la que se trabaja en la creación de políticas tiene un principio básico: una vez que un usuario quiere ejecutar un comando con privilegios especiales (con cuentas administrativas o aplicativas), la herramienta va a "leer" las políticas que tiene declaradas. La herramienta se detendrá cuando encuentre un *accept*, un *reject* o ya no tenga más código que leer. Es así como una petición sólo puede ser rechazada (cuando encuentra un *reject* o cuando termina de leer las políticas) o aceptada (cuando encuentra un *accept*).

Por eso, al momento de crear políticas es necesario saber en qué parte de todo el código se debe rechazar o aceptar una petición, ya que si por algún error se coloca en otra parte, se podrían estar rechazando peticiones válidas o, lo que es más riesgoso aún, se podrían estar aceptando peticiones que no debería ejecutar el usuario.

Revisión de registros

Power Broker posee tres tipos de registros: los registros de todas las llamadas que se hacen a la herramienta, sean exitosas o fallidas (registros de eventos); registros de las salidas de los comandos que se ejecutan con **Power Broker** y son exitosas (registros de entrada y salida); y por último registros del funcionamiento de la herramienta (registros de diagnóstico).

Los registros de las salidas sólo se generan si así se especifica en cada una de las políticas que se tienen declaradas.

Registros de eventos *(Event logs)*
Éstos se producen por omisión y recaban la siguiente información:

- Peticiones aceptadas y rechazadas.
- Ambiente en el que se ejecutó el comando (cuenta, equipo donde se ejecutó el comando, fecha, hora, etcétera).
- Eventos de teclado (*keystroke).*

Al revisar los registros se puede tener la siguiente información:

- Qué comando se mandó a ejecutar (comando que tecleó el usuario).
- Qué cuenta lo mandó ejecutar.
- Cuándo se ejecutó el comando.
- Qué comando se ejecutó (comando que realmente se ejecuta).
- Dónde se ejecutó el comando (en que equipo).
- Con qué cuenta se ejecutó el comando.
- Con qué estado finalizó la ejecución.

Registros de entrada y salida *(I/O logs)*
Power Broker tiene la funcionalidad de poder registrar cualquier entrada o salida de un comando, siempre y cuando se ejecute en un shell de UNIX o una terminal. La creación de estos registros se configura desde las políticas definidas en **Power Broker**.

Registros de diagnóstico *(Diagnostic logs)*
Los mensajes de diagnóstico nos sirven para ver el funcionamiento y posibles fallas que tiene la herramienta. Estos mensajes se pueden encontrar en los registros de eventos del sistema (*syslog*) y dentro de archivos propios de **Power Broker**.

Auditorías a bases de datos

Introducción

La información que posee cualquier empresa es de suma importancia; desafortunadamente la mayoría de las veces las empresas sólo se enfocan

en proteger el perímetro de sus equipos, y no consideran que también es necesario proteger los datos hacia el interior de sus organizaciones.

Pensando en seguridad interna es necesario estar consciente de que algo que se debe proteger es la información con la que cuenta una empresa. En este apartado se va a hablar sobre un mecanismo de auditoría de bases de datos en ambientes UNIX, **IPLOCKS *Database Security & Compliance***.

Este mecanismo nos sirve para controlar las actividades realizadas en cada una de las bases de datos y al mismo tiempo la generación de registros de los DML's (*Data Manipulation Language*) o DDL's (*Data Definition Language*) utilizados.

La herramienta de la que hablaremos posee varios módulos, cada uno dedicado a diferentes actividades. Por ejemplo, se tiene un módulo para la revisión de vulnerabilidades en las bases de datos; este reporte se basa en mejores prácticas. Un módulo más para la revisión de privilegios que se asignan a las cuentas declaradas en las bases de datos, que se enfoca en vigilar cambios en los privilegios de las bases de datos y genera alertas de estas amenazas potenciales; rastrea cambios en los privilegios a través de enunciados como `grant`, `revoke`, permisos de sistema u objetos y cambios en los roles o contraseñas. También cuenta con un módulo capaz de supervisar cambios en la estructura de las bases de datos.

Pero el módulo en el que nos enfocaremos será el relacionado con el comportamiento de los usuarios, es decir, nos centraremos en saber qué es lo que está haciendo cada usuario. Así, este módulo nos ayudará a conocer la información que obtiene y manipula el usuario.

Reglas a auditar

La manera en la que trabaja esta herramienta, y en específico este módulo, consiste en que se define una serie de reglas y cuando alguna de ellas se infringe se lanza una alerta.

Estas reglas pueden ser activadas por alguna de las siguientes características:

- Operaciones que se están realizando (`select`, `update`, `insert` o `delete`).

- Cuenta que se ha conectado.
- Cuenta que está accediendo a ciertos objetos.

Para la creación de estas reglas es necesario que el área responsable de la base de datos nos indique cuáles son sus objetos o cuentas sensibles y qué cuentas se requieren vigilar con un mayor control. Teniendo esta información se procede a la generación de las reglas que vigilen todos estos puntos.

La herramienta cuenta con algunas reglas predefinidas, como son: el cuidado de objetos (tablas, índices, vistas, etcétera) y la auditoría de cuentas, entre otras. Pero ya que las bases de datos pueden ser muy complejas, también se tiene la opción de generar reglas personalizadas para vigilar exactamente lo que la empresa requiera.

Revisión de registros

El sólo monitorear no nos da un panorama de todo lo que se realiza en nuestra base de datos; es importante tener la capacidad de contar con toda esa información de manera ordenada y completa.

IPLocks proporciona un reporte presentando la información de manera que su consulta pueda ser hacia adelante o hacia atrás, permitiendo regresar o adelantarse para mostrar en cada alerta un detalle de su origen.

El detalle de la alerta se muestra en el formato 4W contestando a las preguntas *Who* (quién), *What* (qué), *When* (cuándo) y *Where* (dónde). Indicando así, toda la información del evento que se considera una violación a la seguridad de la base de datos. Esta información de auditoría le da al administrador los elementos necesarios para implementar ajustes y políticas de seguridad manteniendo la integridad, disponibilidad y confidencialidad. La manera de implementar estos ajustes es revisando a qué accede cada uno de los usuarios e ir limitando sus accesos a sólo aquellos que requiera.

Análisis forense

Una de las actividades importantes de las áreas de seguridad, aunque no la más deseada, es el análisis forense. Cuando se realizan actividades de este tipo es porque la empresa ya sufrió un ataque que le provocó alguna pérdida o que puso en riesgo las plataformas, como puede ser algún daño

en el funcionamiento de las mismas; o por que se tiene alguna sospecha de uso malintencionado en las plataformas. El análisis forense se hace revisando los registros, rastreando accesos a los equipos y revisando modificaciones a los archivos de configuración. Por eso es importante que se generen registros de este tipo y de preferencia que sean almacenados en equipos diferentes y se hagan respaldos periódicos.

Capítulo II

Elementos técnicos

Configuración e instalaciones para el sistema operativo

Como ya se mencionó en la primera parte, es necesario asegurar las plataformas contra posibles amenazas. En este capítulo nos vamos a enfocar en lo correspondiente a los sistemas operativos y más adelante se hablará de las bases de datos. Existen dos formas de eliminar las vulnerabilidades en un equipo: configurando las características propias de los sistemas operativos o instalando software que nos ayude a mitigar o minimizar estas vulnerabilidades.

Configuración del sistema operativo

Aquí hablaremos de todas las modificaciones que se pueden realizar en las configuraciones del sistema operativo para proteger a un equipo de cómputo. Cabe mencionar que estas modificaciones no son las únicas que se pueden realizar, pero son en las que se va a enfocar este trabajo. En esta parte se mencionarán las características de los sistemas operativos Solaris, HP-UX, RedHat y Suse. Como estos sistemas son descendientes de System V[4], prácticamente trabajan igual en las cuestiones que se van a describir; en caso de que las configuraciones sean distintas se hará el comentario a fin de mencionar las diferencias que se tienen que considerar dependiendo del sistema operativo.

[4] System V, abreviado comúnmente SysV y raramente System 5, fue una de las versiones del sistema operativo Unix. Fue desarrollado originalmente por AT&T y lanzado por primera vez en 1983. El otro sistema operativo de las dos mayores ramas de los sistemas UNIX corresponde a la distribución de software de Berkeley BSD [13].

Administración de cuentas

Para una empresa siempre debe ser importante controlar todos los accesos que se hacen a las diversas plataformas disponibles. Para tener este control es necesario realizar tareas de administración sobre las cuentas de acceso; tales como asignar un responsable para cada cuenta, manejar periodos de vigencia de contraseñas y tipos de conexión válidos para cada cuenta.

•Asignación de responsable

Uno de los primeros aspectos que se tienen que cubrir, ya sea por seguridad o por obligaciones legales, es el de identificar a los responsables de cada una de las cuentas que estén declaradas en las diferentes plataformas que tenga una empresa.

La manera de cubrir este requerimiento es almacenando la información de los responsables de las cuentas en el archivo propio del sistema operativo `/etc/passwd`. En este archivo se guarda también información adicional sobre las cuentas dadas de alta. De acuerdo a su nombre, pensaríamos que en este archivo se guardan las contraseñas de las cuentas, pero esto es incorrecto. Este archivo se llama así porque anteriormente sí contenía las contraseñas, aunque actualmente se ha creado otro archivo con esta información. Hay que mencionar que en el sistema operativo HP-UX el archivo `/etc/passwd` sí contiene las contraseñas cifradas de las cuentas que se tienen declaradas.

En el archivo `/etc/passwd` existe una línea por cada cuenta que se tiene declarada y en cada una de estas líneas hay un campo en el que se puede poner algún comentario sobre cada una de las cuentas. En nuestro caso colocamos el nombre del responsable de la cuenta. La forma de editar este archivo es utilizando los comandos `useradd` o `usermod`.

• Vigencia de contraseña

Uno de los problemas más grandes relacionados con las contraseñas es convencer a los usuarios que utilicen contraseñas robustas; la mayoría acostumbra utilizar palabras comunes a fin de evitar que se les olviden. El mayor inconveniente de este tipo de contraseñas es que son demasiado fáciles de adivinar y un *hacker* puede hacer mal uso de ellas. Un método que nos

ayuda a proteger estos datos es utilizar tiempos de validez. El periodo estándar de validez es de 30 días; sin embargo, este periodo es muy corto cuando apenas se está introduciendo esta característica en las formas de trabajar de una empresa. Por eso es necesario empezar con un tiempo mayor y, una vez que los usuarios se han acostumbrado, se puede ir reduciendo poco a poco. Este tiempo podría ser de 60 días. Utilizando vigencia en las contraseñas se minimiza el tiempo en que son vulnerables para los *hackers*.

Para asignar vigencia a una contraseña se utiliza el comando `passwd`. También hay que tener en cuenta que cuando a alguien se le ponen reglas incómodas en su forma de trabajar, empezará a buscar maneras de burlar estas medidas. Cuando se utilizan vigencias para las contraseñas, los usuarios pueden cambiar éstas cuando se cumple el periodo de validez y regresar a las que tenían en un principio, lo que equivale a que no se realice ningún cambio. Para evitar este tipo de acciones, también se puede definir un tiempo en el que la contraseña no puede ser cambiada (este tiempo puede ser de unos cinco días) a fin de que el usuario se acostumbre a utilizar la recién introducida. Por otro lado, es importante notificar al usuario cuando su contraseña vaya a expirar a fin de que tome las medidas pertinentes y no tenga ningún problema. Estas dos últimas características también se pueden definir utilizando el comando `passwd`.

Toda la información sobre las contraseñas se guarda en el archivo `/etc/shadow`, que contiene las contraseñas cifradas de todas las cuentas que se encuentran declaradas en el equipo. Para asegurar que las cuentas del sistema no tengan acceso a esta información, `/etc/shadow` sólo puede ser visto por el administrador del sistema. La información contenida en este archivo se manipula mediante el comando `passwd`.

Todas las características que se han tocado en este apartado, por omisión no aplican en los sistemas operativos HP-UX; sin embargo, pueden ser implementadas haciendo algunas modificaciones.

Es importante saber que en los sistemas HP-UX no existe el archivo `/etc/shadow` que, como ya se mencionó, es donde se almacenan las contraseñas cifradas. Es entonces en el archivo `/etc/passwd` donde se guarda toda esta información. Como estos sistemas sólo trabajan con el archivo `/etc/passwd`, no pueden almacenar la información de las contraseñas en un archivo diferente, sea esta información la vigencia de contraseña, la fecha del último cambio o el periodo mínimo en el que no puede ser cambiada la contraseña.

Como muchos programas requieren tener acceso a la información que se encuentra en el archivo `/etc/passwd`, éste debe tener permisos de lectura para todas las cuentas que se tienen declaradas en el sistema, en HP-UX esto provoca una vulnerabilidad pues las contraseñas que se encuentran en él sólo deberían ser visibles para el administrador del sistema o para comandos con los privilegios adecuados.

Para asegurar las contraseñas y manejar la vigencia de éstas, se debe instalar un software propio de HP, **ShadowPassword**. Este software mejora la seguridad del sistema "ocultando" las contraseñas cifradas de las cuentas en un archivo propio (`/etc/shadow`) que sólo puede ser visto por `root`. Una vez creado el archivo es posible trabajar con las vigencias de contraseñas de la misma manera que con Solaris, Suse y RedHat.

Es importante mencionar que en la mayoría de los casos no hay ningún impacto con la instalación de este software; sin embargo, es necesario hacer un análisis antes de proceder pues en algunos casos las aplicaciones pueden verse afectadas si éstas acceden al campo de la contraseña en el archivo `/etc/passwd`. Esto se debe a que cuando se instala **ShadowPassword** el campo que contiene la contraseña en el archivo `/etc/passwd` es sustituido por una "x".

Otra forma de activar la propiedad de manejar vigencia de contraseñas es habilitando el sistema en un modo *trust* (confianza). Habilitar esta característica, además de servir para manejar la vigencia de las contraseñas, nos permite utilizar algunas otras características vinculadas a la seguridad, como: contraseñas de mayor longitud, generación aleatoria de contraseñas, bloqueo de cuentas, tiempos de acceso válidos para las cuentas, utilización del archivo `/etc/shadow`, etcétera. Este trabajo no contempla el uso de esta característica, ya que si es complicado instalar software en las plataformas de una empresa, lo es aún más cambiar las configuraciones de los sistemas operativos.

• Medios de conexión válidos

Un equipo puede tener habilitados diferentes medios de conexión para poder proveer el servicio para el que fue diseñado. Entre estos servicios de conexión podemos mencionar `ssh`, `telnet`, `ftp`, `rlogin`, `rsh` y `rexec`. Hay que mencionar que de estos servicios sólo `ssh` (*secure shell*) se considera un servicio de conexión seguro, ya que los restantes envían toda la información que pasa por la red en claro (como puede ser la cuenta y la

contraseña); por lo que se tienen que utilizar mecanismos adicionales para poder proteger la información que viaja; por ejemplo el uso de VPN[5] (*Virtual Private Networks*).

Es importante tener control sobre los servicios de conexión que pueden utilizar cada una de las cuentas, de tal forma que se puedan restringir los servicios vulnerables (`telnet` y `ftp` por ejemplo) así como el acceso con cuentas sensibles (`root` y cuentas de aplicaciones). Existen diferentes formas de restringir el uso de servicios, pero en esta parte nos vamos a enfocar en las que están disponibles con sólo modificar archivos de configuración propios de los sistemas operativos.

Archivo `/etc/ftpusers`. En este archivo se declaran todas aquellas cuentas que no deben conectarse al equipo mediante el servicio `ftp`. En un inicio, las cuentas que deben estar declaradas son las de sistema (`sys`, `adm`, `daemon`, etcétera), ya que éstas no se deben utilizar para acceder al equipo. Después es necesario hacer un análisis para definir qué cuentas deben estar contenidas en este archivo a fin de que no sean afectadas las aplicaciones que proveen el servicio para el que fue diseñada la plataforma. Los cambios que se realizan en este archivo los toma en automático el servicio `ftp`, por lo que no se requiere reiniciarlo.

Archivo `/etc/default/login`. En este archivo sólo se pueden bloquear los accesos de la cuenta `root` hacia el equipo a través de `telnet`, `ftp`, `rsh`, `rlogin` y `rexec`. Para esto es necesario agregar la variable `CONSOLE=/dev/console`. De esta forma la cuenta `root` sólo se podrá conectar al equipo por la consola de administración. Al igual que el archivo `/etc/ftpusers`, los cambios se toman de manera automática e inmediata.

Archivo `/etc/ssh/sshd_config`. Como su nombre lo dice, en este archivo podemos encontrar las configuraciones para el servicio de `ssh`. Aquí podemos indicarle al servicio que no se van a permitir accesos al equipo utilizando la cuenta `root`. Para establecer esta restricción es necesario incluir la siguiente variable en el archivo `PermitRootLogin no`. A diferencia de los dos archivos anteriores, en este caso sí es necesario indicarle al servicio de `ssh` que sus configuraciones han cambiado y que es necesario que las vuelva a leer para que se activen. Existen dos maneras de llevar a cabo esto, dependiendo de la forma en que este servicio es administrado. Si está

[5] VPN es una tecnología de red que permite una extensión de la red local sobre una red pública o no controlada [11].

siendo administrado desde el demonio `inetd`[6], es necesario indicarle a éste último que vuelva a leer sus configuraciones. En cambio, si está corriendo en modo autónomo (*stand-alone*), es decir que él sólo se administra, es necesario indicarle al propio servicio `ssh` que lea sus nuevas configuraciones.

Archivo `/etc/shells`. Este archivo es una pequeña base de datos en la que el sistema operativo almacena los shells válidos a utilizarse en el archivo `/etc/passwd`. Si una cuenta tiene asignado un shell que no esté declarado en este archivo, al momento de conectarse el usuario el sistema operativo no podrá ejecutar tal shell y por lo tanto no se podrá terminar con el proceso de conexión. Por omisión, este archivo no existe en los sistemas operativos Solaris y HP-UX. Para hacer uso de este archivo no es necesario reiniciar ningún proceso.

• Depuración de cuentas

Cuando se trata de administración de cuentas, es necesario conservar sólo aquellas que sean necesarias para el funcionamiento correcto de la plataforma y las aplicaciones que se encuentran en ella. Cuando se hace una instalación por omisión, se crean muchas de éstas que en realidad no son necesarias y hacen que el equipo se vuelva más vulnerable. Por esta razón hay que hacer una revisión de todas las cuentas que se tienen declaradas y determinar cuáles de ellas no son necesarias. Una vez que se tienen identificadas, deben ser eliminadas del equipo. Para hacer esto sin que se presenten problemas o queden registros inválidos, es importante que se revisen los archivos que se encuentran en el equipo y que son propiedad de estas cuentas. Si las eliminamos sin borrar tales archivos, éstos permanecerán en el equipo sin una cuenta válida asignada. También es necesario revisar los directorios asignados a las cuentas que se van a eliminar para determinar si contienen directorios de sistema o compartidos con otras: en ambos casos deberá eliminarse la cuenta pero no el directorio. Para eliminar cuentas se utiliza el comando `userdel`.

[6] El demonio de Internet, `/usr/sbin/inetd`, es el servidor maestro de muchos servicios de Internet. Este demonio controla las solicitudes de conexión de los servicios enumerados en el archivo de configuración `/etc/inetd.conf` y crea el servidor apropiado al recibir una solicitud [3].

Administración de privilegios

Ya que se tienen bien definidas y declaradas todas aquellas cuentas necesarias para el funcionamiento, administración y operación de la plataforma, es necesario proveerlas de privilegios dependiendo de las actividades que realicen. Entre estos privilegios podemos mencionar: tiempos de conexión y de inactividad, uso de tareas programadas y ejecución de ciertos programas instalados en la plataforma.

• Tiempos de inactividad permitida

Existen dos tipos de inactividad, la de una sesión y la de una cuenta. Para el caso de inactividad de sesiones existen las variables `TMOUT` o `autologout` que sirven para establecer tiempos válidos de inactividad antes de que una sesión sea cerrada de manera automática por el sistema operativo. `TMOUT` la podemos utilizar para los shells Bash, Korn Shell y Z Shell; mientras que `autologout` sirve para TCShell. La forma de activar estas variables es editando el archivo propio de cada shell agregando lo siguiente `TMOUT=seg` o `autologout=seg`; donde `seg` es el número de segundos que puede permanecer inactiva una sesión antes de que ésta sea cerrada por el sistema.

Para no depender del shell que tiene asignado cada cuenta, se puede agregar otro tipo de mecanismo para cubrir el punto, el cuál se explicará en la parte de instalación de software.

El tipo de inactividad de una cuenta se utiliza cuando ésta cumple cierto tiempo sin haberse utilizado para establecer una conexión. Una vez transcurrido este tiempo la contraseña de la cuenta será bloqueada para que no pueda conectarse hasta que el administrador la desbloquee. Este mecanismo se utiliza para evitar que cuentas en desuso sean utilizadas para fines maliciosos. La forma de activar este tipo de validación es mediante el comando `useradd` o `usermod` y el argumento `-f`, donde el valor que recibe es el número de días consecutivos en los cuales la cuenta será bloqueada si no se tiene algún acceso al equipo. Hay que mencionar que esta característica sólo se encuentra en los sistemas operativos Solaris y HP-UX, ya que para el caso de RedHat esta opción (`-f`) se utiliza para especificar el número de días que deben transcurrir desde que una contraseña expira para que la cuenta sea bloqueada.

• Uso de tareas programadas

Cuando hablamos de tareas programadas, nos referimos a la ejecución de programas en un determinado tiempo sin necesidad de tener que ejecutarlos manualmente. Para los sistemas en los que se enfoca este trabajo tenemos las herramientas conocidas como *crontab* y *atjobs*. Es necesario restringir el uso de estas herramientas a cuentas de aplicaciones y de vigilancia para evitar que usuarios mal intencionados puedan dejar scripts maliciosos dentro de un equipo y sean ejecutados cuando el usuario no esté conectado al equipo con la cuenta en cuestión; esta clase de manejo hace más difícil identificar quién ejecutó este tipo de scripts.

En los sistemas operativos en los que nos estamos enfocando, existen dos archivos para cada una de las herramientas antes mencionadas los cuales nos permiten controlar su uso; estos archivos son: `cron.allow`, `cron.deny`, `at.allow` y `at.deny`. Por omisión estos archivos se encuentran en el directorio `/etc/cron.d` para el caso de Solaris y HP-UX, y en `/etc` para el caso de RedHat.

La mejor forma de controlar el uso de estas herramientas es negando todas las cuentas y permitiendo sólo aquellas que sean necesarias. La manera de hacer esto es dando de alta las cuentas válidas en los archivos `allow` y restringir todas las demás cuentas en los archivos `deny` con la cláusula `ALL`.

• Ejecución de programas válidos

Una solución para restringir la ejecución de comandos es crear grupos de trabajo dentro del mismo equipo y asignar diferentes permisos a los comandos que se quieran limitar, para que sólo aquellos usuarios válidos los puedan ejecutar.

Cada vez que se crea una cuenta se le debe asignar un grupo de trabajo. Para el caso de Solaris, por omisión, el sistema operativo asigna el grupo 1 (*others*), mientras que para otros sistemas operativos como LINUX, por omisión, se crea un grupo exclusivo para la cuenta. Para poder trabajar con diferentes grupos es necesario asignarle un grupo dependiendo de los trabajos que va a realizar. Para hacer esta asignación se utiliza el parámetro `-g` en los comandos `useradd` o `usermodd`. En caso de que se requiera que una cuenta pertenezca a más de un grupo de trabajo se puede hacer utilizando el parámetro `-G` en cualquiera de los comandos anteriores.

Ya que se tienen bien definidos los grupos de trabajo es necesario reestructurar los permisos de los comandos que se quieran limitar. Para reali-

zar este cambio se utiliza el comando `chown` (para cambiar el propietario y grupo) y `chmod` (para cambiar los permisos).

Control de accesos

Es muy importante tener en cuenta que existen varias áreas que acceden a la plataforma para llevar a cabo tareas de operación y administración. Debido a esto, se requiere tener identificadas a las personas que se pueden conectar a un equipo, así como las direcciones IP desde las cuales van a realizar estas conexiones. También es necesario tener identificada cada cuenta con el servicio que tiene autorizado utilizar para conectarse al equipo (`ssh`, `telnet`, `ftp`, etcétera).

En este apartado hablaremos de cómo manejar estos accesos hacia los equipos.

• **Restricciones por IP y servicio**

En RedHat y las nuevas versiones de Solaris ya se cuenta con herramientas que nos permiten trabajar con el control de accesos, tales como **TCP-wrappers** (para Solaris y RedHat) e **IPTables** (sólo para RedHat). En este trabajo se revisará **TCP-wrappers**. Para el caso de HP-UX y versiones de Solaris que no cuenten con esta herramienta se verá más adelante la forma de instalación.

Tanto la herramienta que ya viene instalada por omisión como la instalada manualmente, trabajan de la misma forma: su funcionamiento se basa en reglas de IP válidas y no válidas. La forma de configurar el uso de **TCP-wrappers** es, primero, indicar qué servicios van a trabajar con esta herramienta. En este trabajo sólo se va a hablar de los servicios que se administran con `inet` o `xinet`, ya que para los servicios *stand-alone* depende mucho de su configuración el poder filtrarlos con **TCP-wrappers** y no siempre se siguen los mismos pasos.

Para trabajar con **TCP-wrappers** se requiere modificar la forma de trabajar del servicio que se desea filtrar, indicando que el acceso a éste tiene que ser controlado por **TCP-wrappers**.

Existen dos formas de modificar los servicios administrados por `inet`, dependiendo del tipo de demonio que se tenga instalado, `inetd` para Solaris y HP-UX o `xinetd` para RedHat.

Para el caso de `inetd` es necesario editar el archivo `/etc/inetd.conf`.

Para `xinted` es necesario modificar el contenido de los archivos que se encuentran en el directorio `/etc/xinetd.d`.

Supongamos que se quiere filtrar el servicio `ssh` que es administrado por `inet`. En el archivo `/etc/inetd.conf` aparece la siguiente línea:

```
ssh stream tcp nowait root /usr/local/sbin/sshd   sshd –i
```

Para indicar que va a ser filtrado por **TCP-wrappers** hay que hacer el siguiente cambio:

```
ssh stream tcp nowait root /usr/sbin/tcpd \
/usr/local/sbin/sshd –i
```

donde el penúltimo y último argumento se refieren al servicio y los argumentos del mismo, respectivamente.

Para el caso de los servicios administrados por `xinetd` en la misma situación, hay que cambiar las siguientes líneas:

```
      server      = /usr/local/sbin/sshd
      server_args = –i
```

por

```
      server      = /usr/sbin/tcpd
      server_args = /usr/local/sbin/sshd –i
```

Donde `-i` significa que el servicio va a ser administrado por el demonio `inet`. Las rutas de los servicios pueden variar dependiendo del sistema operativo y de la versión de **TCP-wrappers** que se esté utilizando.

Para ambos casos es necesario indicarle al demonio `inet` o `xinet` que debe leer sus nuevas configuraciones.

```
# kill –HUP id_proceso
```

Ya que tenemos filtrados los servicios hay que indicar cuáles IP van a ser válidas para cada uno de ellos. Como en el caso del uso de las tareas programadas, es más seguro negar primero todos los accesos y después indicar puntualmente las direcciones IP válidas. Para hacer esto es necesario crear dos archivos con el siguiente contenido:

```
/etc/hosts.deny
ALL:ALL
```

```
/etc/hosts.allow
<servicio1>: lista de IP separadas por espacios
<servicio2>: lista de IP separadas por espacios
…
<servicioN>: lista de IP separadas por espacios
```

Para llevar un mayor control y administración de estas IP, es recomendable que en el mismo archivo `/etc/hosts.allow` se agreguen líneas (comentadas) indicando a quién pertenece cada IP y con qué servicio tienen permitido conectarse al equipo. Para agregar líneas comentadas a los archivos `hosts.allow` y `hosts.deny` basta con poner el símbolo `#` al inicio de cada línea.

Administración de servicios

Cuando hablamos de proteger una plataforma, nos referimos a tratar de cerrar todos los accesos que no requiera. Una parte muy importante consiste en bloquear o deshabilitar todos aquellos servicios que no son necesarios para la funcionalidad que provee la plataforma, esto con el fin de evitar que alguien pueda explotar sus posibles vulnerabilidades. Así, entre menos accesos tengan que ser administrados, éstos podrán ser protegidos de una mejor forma.

• Cierre de servicios innecesarios

Nos enfocaremos en los servicios que están siendo administrados por el demonio `inet`, aunque sabemos que existen otros que corren de manera autónoma. Sin embargo, por experiencia, gran cantidad de servicios administrados por el demonio `inet` no se requieren por las plataformas, a diferencia de los que se ejecutan solos. En este apartado sólo hablaremos de los primeros.

Existen dos formas de deshabilitar los servicios administrados por `inet`, dependiendo del tipo de demonio que se tenga instalado, `inetd` o `xinetd`.

Para el caso de `inetd` es necesario editar el archivo `/etc/inetd.conf` y comentar o eliminar las líneas de todos aquellos servicios que no sean requeridos por la plataforma.

Para `xinetd` es necesario modificar el contenido del directorio `/etc/xinetd.d`, ya sea eliminando los archivos relacionados con los ser-

vicios que queremos deshabilitar o editándolos para modificar la variable `disable` asignándole `yes`.

En ambos casos se requiere indicarle al demonio que vuelva a leer sus archivos de configuración.

• Deshabilitación de servicios automáticos (*runcontrols*)

Aunque este tipo de servicios no son administrados por `inet`, también los abordaremos pues existe gran cantidad de ellos que se instalan por omisión y no se requieren para las plataformas. El sistema operativo habilita y deshabilita automáticamente este tipo de servicios al encenderse o apagarse, respectivamente; así que no basta con deshabilitarlos, en un primer momento en la línea de comandos, ya que cuando el equipo sea reiniciado volverán a habilitarse.

Por lo general, estos servicios se administran desde los directorios `/etc/rc2.d`, `/etc/rc3.d` y `/etc/rc5.d`. La manera de deshabilitarlos es la siguiente:

Primero hay que apagarlos manualmente, lo que requiere ejecutar el siguiente comando dentro del directorio donde se encuentre el servicio que se quiera deshabilitar:

```
# ./servicio stop
```

Una vez que se ha detenido el servicio, es necesario borrar el archivo respectivo para que el sistema operativo ya no lo encuentre. También se puede simplemente renombrarlo, ya que todos los archivos que se encuentran en estos directorios y su nombre inicia con la letra `S` (*start*), el sistema operativo los reconoce como servicios que tienen que ser habilitados cuando el equipo sea encendido, por lo que es necesario renombrar los archivos ya sea quitándoles la `S` del inicio o anteponiéndoles alguna otra letra.

Para los archivos cuyo nombre inicia con `K` (*kill*) no es necesario hacer ninguna modificación, ya que estos archivos se utilizan para dar de baja el servicio cuando el equipo se apaga o para colocar el sistema operativo en un nivel de ejecución en particular.

Conexiones confiables

En los sistemas operativos que estamos trabajando existen mecanismos que nos permiten llevar a cabo sesiones y ejecutar comandos de manera remota, sin la necesidad de que se nos solicite la contraseña de la cuenta con

la que nos queremos conectar o ejecutar un comando en otro equipo. Aunque estas características facilitan la manera de trabajar, son poco seguras ya que si alguien compromete la cuenta de un equipo que tenga habilitado este tipo de conexiones también podría comprometer la cuenta del otro servidor a donde se puede conectar sin ninguna autentificación. Existen cuatro servicios que trabajan de esta manera: `rexec`, `rsh`, `rlogin` y `ftp`.

• **Eliminación de archivos** `.rhosts`

Este tipo de archivos se utiliza para permitir la ejecución de los comandos remotos `rexec`, `rsh` y `rlogin`. Lo que se configura en este archivo son listas de IP y cuentas a los cuales se les va a permitir el acceso sin la necesidad de que se introduzca una contraseña.

Por ejemplo, podemos tener un archivo `.rhosts` en el equipo A con el siguiente contenido:

```
ip_equipoB cuenta1
```

Esto significa que si la cuenta1 ejecuta cualquier comando remoto desde el equipo B hacia el equipo A, no se le va a solicitar ninguna contraseña, además de que los comandos se van a ejecutar con los privilegios de la cuenta a la que pertenezca el archivo `.rhosts` en el equipo A.

Debido a esto es necesario eliminar esta clase de archivos de la plataforma que se quiere proteger.

• **Eliminación de archivos** `.netrc`

Estos archivos sirven para establecer conexiones `ftp` sin tener que introducir una contraseña. En este archivo se almacenan los nombres de los equipos, cuentas y contraseñas hacia donde se quieran realizar las conexiones de `ftp`, de tal forma que al realizar una transferencia de archivos ésta se hará en automático sin ninguna interacción con el usuario.

Una de las debilidades que tiene este tipo de archivos es que en ellos se almacenan contraseñas en claro. Otra más es que estos archivos se utilizan para establecer conexiones mediante `ftp` y, como ya se mencionó, estas conexiones no son seguras pues toda la información viaja en claro a través de la red.

Instalación de software

Después de tratar las características propias de los sistemas operativos que pueden modificarse para asegurar las plataformas, vamos a hablar de al-

gunas herramientas libres que nos ayudan a implementar más medidas para asegurar un equipo. Estas herramientas nos ayudarán a trabajar con la administración de sesiones concurrentes e inactivas, uso de contraseñas robustas, uso de conexiones seguras y, por último, control de accesos.

Administración de sesiones

Idled es una herramienta que se ejecuta en un segundo plano (*background*) y su función es vigilar las terminales que solicita cada usuario que se conecta a un equipo, a fin de cerrar todas aquellas conexiones que no cumplan con los parámetros que le sean definidos para el uso de sesiones concurrentes o inactivas. Hay que mencionar que **Idled** notifica con anticipación al usuario que la sesión se cerrará para que tenga oportunidad de cerrarla manualmente o siga trabajando en ella.

Este tipo de herramientas se utilizan para evitar que los usuarios dejen sesiones abiertas y que alguien más pueda hacer uso de ellas indebidamente, así como para evitar que varias personas utilicen la misma cuenta.

Por omisión, ninguno de los sistemas operativos de los que hemos hablado viene con **Idled** instalado, por lo que es necesario instalarlo en cada equipo en el que se pretenda utilizar. El software de **Idled** puede descargarse desde el mirror http://www.filewatcher.com/m/idled-1.16.tar.gz.70864.0.0.html, en forma de código fuente.

Una característica importante de este software es que está diseñado para trabajar con varias plataformas de sistema operativo, como Solaris, HP-UX, Linux, entre otros. En el archivo `Makefile` incluido con el código fuente, se debe de indicar la versión del sistema operativo que se está utilizando.Como este software está programado en C, para instalarlo es necesario contar con un compilador de C. Por seguridad, es conveniente que se desinstale el compilador una vez que se haya terminado de utilizar.

Las instrucciones para instalar este software vienen incluidas en el directorio que se descarga de esta herramienta.

Una vez que se ha instalado, se requiere configurar la herramienta modificando el archivo `/etc/idled.cf`. Éste cuenta con varios enunciados, de los cuales los más importantes, y de los que vamos a hablar en este trabajo, son:

`multiples` – Indica el número de sesiones concurrentes válidas para cada cuenta.

`timeout tty console` – Indica el tiempo de inactividad válido para cada cuenta que se utilice para una sesión en la consola.
`timeout default` – Indica el tiempo de inactividad válido para cada cuenta.

Es importante mencionar que **Idled** tiene la capacidad de excluir las reglas antes definidas para las cuentas, grupos o tipos de conexiones. Esto se hace mediante los siguientes enunciados en el archivo `/etc/idled.cf`:

`exempt tty console session` - Excluir las reglas para todas aquellas sesiones que se hagan desde la consola.
`exempt group staff all` - Excluir al grupo `staff`.
`exempt login root alls` - Excluir a la cuenta `root`.

Una vez que se ha configurado la herramienta es necesario ponerla a trabajar; la mejor manera es crear un script de arranque automático (*run-control)* para que se inicie cada vez que encienda el equipo. Para hacer esto hay que crear un script y guardarlo en el directorio que corresponde a los runcontrols de red, puede ser: `/etc/rc2.d`, `/etc/rc3.d` o `/etc/rc5.d` (es muy importante validar que este script tenga permisos de ejecución únicamente para la cuenta `root`).

Ya que tenemos trabajando la herramienta, toda la información de las sesiones que se han cerrado se guardará en el archivo `/var/log/idled.log`, indicando cuál fue la causa del cierre (exceso de sesiones o de tiempo de inactividad).

Uso de contraseñas robustas

Npasswd es una herramienta que reemplaza algunas características del comando `passwd` en los sistemas UNIX. Con esta herramienta, las nuevas contraseñas que se generen serán revisadas ampliamente para que sean lo suficientemente complejas a fin de disminuir, a usuarios malintencionados, la capacidad de obtenerlas.

Con **Npasswd** la creación de nuevas contraseñas se hace más robusta, ya que se pueden definir reglas, además de las que vienen por omisión en el sistema operativo, para la aceptación de contraseñas. Algunas de ellas son:

- Establecer cuántos tipos de caracteres debe contener la contraseña para que sea válida (minúsculas, mayúsculas, números y signos de puntuación).
- Uso de diccionarios predefinidos para evitar que se utilicen contraseñas fáciles de obtener.
- Mecanismos para la no reutilización de contraseñas.
- Longitud de contraseñas.

El software de **Npasswd** puede descargarse del sitio http://www.utexas.edu/cc/unix/software/npasswd/download2.html (es necesario descargar tanto el software para la instalación como los diccionarios).

Al igual que para **Idled**, se requiere contar con un compilador de C para su instalación. **Npasswd** trabaja para los sistemas operativos que estamos revisando, aunque para el caso de RedHat no es necesario instalarlo pues RedHat ya cuenta con mecanismos de validación de contraseñas robustas bastantes completos que incluyen diccionarios; para el caso de HP-UX hay que hacer una modificación al sistema operativo antes de su instalación, ya que por omisión no hace uso del archivo `/etc/shadow` para el manejo de las contraseñas. La tabla de contraseñas se encuentra en el mismo archivo que contiene la información de las cuentas (`/etc/passwd`). Como **Npasswd** internamente modifica el archivo de contraseñas (`/etc/shadow`), es necesario instalar un módulo de HP-UX (**ShadowPassword**) para separar la información de las cuentas de la de las contraseñas. Este módulo se descarga directamente de la página http://h20392.www2.hp.com (la forma de instalación también se puede encontrar en este sitio).

Una vez que contamos con los archivos `/etc/passwd` y `/etc/shadow` podemos iniciar la instalación de **Npasswd**. Al igual que en el caso de **Idled**, las instrucciones necesarias se encuentran en los archivos descargados.
Npasswd cuenta con un archivo de configuración (`/usr/lib/passwd/passwd.conf`) en el cual se pueden definir varias propiedades para su uso. A continuación mencionamos las más importantes:

`passwd.AlphaOnly`. Si se van a aceptar solamente letras en las contraseñas.

`passwd.CharClasses`. Los tipos de caracteres que se van a requerir en una contraseña (minúsculas, mayúsculas, números y signos de puntuación).

`passwd.History`. Periodo de tiempo en el cual una contraseña no puede ser reutilizada.

`passwd.History database`. Archivo donde se va a llevar un registro de las contraseñas utilizadas.

`passwd.MinPassword`. Longitud de la contraseña.

Si se va a utilizar la característica de reuso de contraseñas, es necesario darle mantenimiento al archivo donde se almacena el registro de las contraseñas utilizadas. La mejor manera es ejecutarlo vía `cron` para que se haga en automático. **Npasswd** cuenta con un comando que realiza esta actividad, `history_admin purge`.

Otra actividad que se tiene que considerar cuando se está trabajando con **Npasswd** en Solaris, es que cuando se llega a instalar un parche de sistema operativo, éste instala nuevamente la versión de `passwd` del sistema operativo, por lo que es necesario volver a indicarle al sistema que debe trabajar con **Npasswd**. Esto se hace copiando el binario que se crea al instalar **Npasswd** en la ruta donde se encuentra el binario del sistema operativo:

```
#cp /usr/lib/passwd/npasswd /usr/bin/passwd
```

Por último, al momento de la configuración, cuando se instala **Npasswd**, se puede crear una copia del archivo binario de `passwd` de Solaris. En caso de que se cree esta copia, será necesario cambiarle los permisos con los que se crea; de lo contrario cualquier usuario podría utilizarlo para modificar su contraseña. El directorio donde se localiza este respaldo es `/usr/lib/passwd/system` y se deberán eliminar todos los permisos para el grupo y otros (`others`).

Servicios de conexión segura

Secure Shell (`ssh`) es un servicio de comunicación entre equipos similar a `telnet`, sólo que `ssh` viaja por un canal cifrado y no es posible entender el tráfico que pasa entre los equipos que se encuentran en sesión. Este servicio también se utiliza para la transferencia de archivos como `ftp`, sólo que la información también va en un canal cifrado.

El software necesario para la instalación de `ssh` depende del sistema operativo con el que se esté trabajando; para el caso de Solaris se debe des-

cargar del sitio http://www.sunfreeware.com, para HP-UX hay que descargar directamente del sitio de Software Depot de HP http://h20293.www2.hp.com y para RedHat del sitio www.redhat.com/wapps/sso

Debe mencionarse que en la mayoría de los casos estos sistemas operativos ya cuentan con una instalación de `ssh`, aunque es importante estar actualizando este software.

Una vez que se ha instalado `ssh` (los manuales de instalación vienen en el directorio que se descarga del software) es necesario configurarlo para contar con algunos mecanismos adicionales de seguridad. Por ejemplo para el manejo de sesiones con la cuenta de `root` o el despliegue de mensajes de inicio (*banners*).

Para todos estos cambios se debe modificar el archivo de configuración de `ssh`, `sshd_config`, e indicarle al servicio que vuelva a leerlo para aplicar estos cambios.

Para conexiones directas como `root` es necesario modificar el parámetro `PermitRootLogin` asignándole el valor de `no`.

Para el uso de mensajes de inicio se debe asignar al parámetro `Banner` la ruta del archivo que contiene la información que uno quiere que se muestre cuando se inicie una petición de `ssh`.

Para mayor información de los parámetros que se pueden definir en este archivo, se puede consultar el manual del `sshd_config` (`$ man sshd_ config`).

Control de accesos

Como ya se mencionó anteriormente, **TCP-wrappers** es una herramienta que nos permite crear un control de accesos a servicios TCP/IP, como son `ftp`, `telnet` y `ssh`; es el encargado de registrar y validar todas las conexiones que se hagan hacia un equipo.

Los servicios se restringen mediante los archivos `/etc/hosts.allow` y `/etc/hosts.deny`; asimismo los accesos se graban en un archivo de registros.

Al igual que `ssh`, la descarga del software depende del sistema operativo con el que se esté trabajando y los sitios de descarga son los mismos que para `ssh`.

Dependiendo del sistema operativo que se esté trabajando, se puede tener ya una versión instalada de **TCP-wrappers**, como en el caso de RedHat y Solaris10; para los demás se tiene que hacer la instalación del software.

La forma de instalación depende del sistema operativo y los manuales de instalación vienen dentro de la documentación que se descarga con el software.

No basta con instalar **TCP-wrappers**, para que el demonio correspondiente filtre los accesos, es necesario configurarlo para indicarle las direcciones IP que van a tener acceso y las que estarán restrigidas; la generación de registros y los archivos a utilizar para almacenar éstos.

Todas estas configuraciones ya se mencionaron con anterioridad.

Configuración de bases de datos

Una de las partes más importantes en un sistema es la información que se tiene almacenada en las bases de datos, pues de esta información dependen todas las aplicaciones que se encuentran instaladas en los equipos de cómputo. En un inicio, las empresas se preocupaban solamente por proteger la información contenida en el sistema operativo, pero poco a poco esta perspectiva ha cambiado y hoy en día las empresas se ocupan además de la seguridad de sus bases de datos.

Como sabemos, existe gran cantidad de manejadores de bases de datos, los cuales llamaremos de aquí en adelante simplemente bases de datos. Entre algunas bases de datos podemos mencionar Oracle, Sybase, TimesTen y MySQL. El alcance de este capítulo será exclusivamente de las bases de datos Oracle; sin embargo, hay que estar conscientes de que es necesario proteger cualquier otra utilizada.

En esta sección hablaremos de algunas modificaciones que se pueden hacer a las bases de datos para proteger la información que se almacena en ellas. Cabe mencionar que los puntos que se tratarán no son los únicos para poder proteger las bases de datos, pero si son los más importantes y fáciles de implementar; donde "fáciles" quiere decir que se requiere un análisis mínimo para su implementación.

Administración de cuentas

Al igual que en los sistemas operativos, es necesario contar con una correcta administración de las cuentas que se tengan declaradas en las bases de datos, con el fin de tener un control de ellas y poder identificar el uso que se le da a cada una.

En Oracle existe un mecanismo que nos permite administrar las cuentas, por medio del uso de perfiles. Es importante que cada cuenta tenga asignado un perfil dependiendo de las actividades que se van a realizar con ésta. Existe un perfil por omisión, el cual también es necesario restringir para que, en caso de que a una cuenta no le sea asignado uno, ésta tome los valores que se tienen para el perfil por omisión, con las restricciones necesarias.

A continuación mencionaremos algunas características de los perfiles para la administración de cuentas.

• Creación y modificación de perfiles

El uso de perfiles nos permite limitar los recursos de la base de datos como son: uso de procesador, número de sesiones, tiempo de conexión, administración de contraseñas, etcétera. Como ya se mencionó, cuando se crea una base de datos, ésta se crea con un perfil por omisión que será asignado a todas las cuentas, por lo tanto es necesario crear perfiles dependiendo del uso que tenga cada una de las cuentas que se encuentren declaradas en las bases de datos, con las restricciones necesarias para cada caso.

A continuación se dan algunos casos típicos de uso:

Para crear un perfil la instrucción es la siguiente:

```
SQL> CREATE PROFILE NOMBRE_PERFIL LIMIT
     RECURSO_1 VALOR
     …
     RECURSO_n VALOR;
```

Para modificar un perfil ya existente, en vez de utilizar la palabra `CREATE` se utiliza `ALTER`.

Para que Oracle pueda trabajar con perfiles es necesario activar el parámetro `RESOURCE_LIMIT` de la base de datos, ya que si este parámetro tiene el valor `FALSE` las cuentas no tendrán restricción alguna.

La forma de modificar este parámetro es la siguiente:

```
SQL> ALTER SYSTEM SET RESOURCE_LIMIT=TRUE;
```

• Vigencia de contraseñas

Dentro de los perfiles de base de datos existe el parámetro

PASSWORD_LIFE_TIME,

el cual nos indica por cuántos días será válida una contraseña. Después de este tiempo ésta se tomará como inválida hasta que sea cambiada por el administrador de la base de datos y el contador de días se vuelva a reiniciar. A diferencia del sistema operativo, Oracle no solicita cambio de contraseña una vez que ésta ha caducado. Por eso es importante llevar un control de las fechas en las cuales ha sido cambiada una contraseña.

Cabe mencionar que existe otro parámetro, PASSWORD_GRACE_TIME, que nos permite asignar un periodo de gracia, en el cual la contraseña puede cambiarse aun cuando se haya vencido.

• Reutilización de contraseñas

También se cuenta con los parámetros

PASSWORD_REUSE_TIME y PASSWORD_REUSE_MAX

para establecer un periodo de tiempo en el cual las contraseñas anteriores no podrán ser reutilizadas. Es importante establecer un tiempo razonable para que los usuarios no puedan utilizar sus contraseñas más recientes. Por experiencia un tiempo de 10 o 15 días es suficiente. La diferencia entre estos dos parámetros, es que el primero se basa en los días que han transcurrido desde que se utilizó la contraseña y el segundo en el número de contraseñas que debe utilizar el usuario antes de volver a usar la primera.

• Tiempos de conexión

Otro parámetro que es de gran utilidad es el de CONECTION_TIME. Con él se define el intervalo de tiempo válido en el que una cuenta podrá tener una sesión abierta. Este parámetro debe definirse en minutos. Se recomienda hacer un análisis previo de los tiempos de conexión para las cuentas de aplicaciones, ya que establecer un tiempo muy corto podría provocar la pérdida del servicio.

• Tiempos de inactividad

Con el parámetro IDLED_TIME se define el intervalo de tiempo válido en el cual una cuenta puede estar inactiva. Una vez que este tiempo se haya cumplido, la misma base de datos cerrará la sesión. El valor de este parámetro es en minutos. De igual forma que con el parámetro anterior, es necesario realizar un análisis para las cuentas de aplicaciones.

• Sesiones concurrentes válidas

`SESSIONS_PER_USER` nos ayuda a definir las sesiones simultáneas que se podrán hacer con cada cuenta. Una vez que se haya alcanzado el número que se definió para este recurso ya no se permitirán más conexiones con la cuenta que se esté utilizando. Para este parámetro también es muy importante hacer un análisis previo para las cuentas de aplicaciones.

• Asignación de perfiles

Una vez que ya se ha creado un perfil de acuerdo a las necesidades de una cuenta específica, hay dos maneras de asignarlo: cuando la cuenta va a ser creada o cuando ésta ya existe. Las dos formas son las siguientes:

```
SQL> CREATE USER NOMBRE_CUENTA PROFILE NOMBRE_PERFIL;
```

o

```
SQL> ALTER USER NOMBRE_CUENTA PROFILE NOMBRE_PERFIL;
```

Es muy importante que siempre que se cree una cuenta se le asigne un perfil de acuerdo a las actividades que desempeñará. En la mayoría de las ocasiones no se hace este tipo de asignación, por lo que Oracle de forma automática le asigna el perfil por omisión. Por ello es recomendable que el perfil por omisión se restrinja lo más posible para forzar a crear perfiles acordes a cada cuenta.

• Depuración de cuentas

Al igual que en el sistema operativo, es necesario realizar un análisis de todas las cuentas que se tienen declaradas en la base de datos, para poder identificar cuáles deben ser eliminadas o bloqueadas. Existen algunas cuentas que el propio manejador utiliza, por lo que no es posible eliminarlas todas.

El resultado del análisis dependerá de las herramientas que se tengan instaladas en la base de datos, como pueden ser: *clusters*[7], ejecución de estadísticas, ejecución de respaldos, consolas de administración, entre otras.

[7] Conjuntos o conglomerados de computadoras construidos mediante la utilización de componentes de hardware comunes y que se comportan como si fuesen una única computadora; se emplean para mejorar el rendimiento y/o la disponibilidad por encima de la que provee una sola computadora [8].

Uso de contraseñas robustas

Oracle posee la capacidad de validar que las contraseñas que se están utilizando cumplan con un cierto grado de complejidad. Esto se realiza con una función de verificación de contraseñas que se asigna a los diferentes perfiles que se tienen declarados en la base de datos (incluyendo el perfil por omisión).

Oracle ya cuenta con una función de validación que se localiza en `$ORACLE_HOME/rdbms/admin/utlpwdmg.sql`; en este archivo se encuentra la programación de la función y puede ser modificada de acuerdo a los requerimientos que necesita cada empresa.

Por omisión, esta función valida, entre otro aspectos, que sean contraseñas alfanuméricas, que cuenten con cierta longitud (este valor puede cambiarse fácilmente) y maneja una lista de palabras prohibidas (un pequeño diccionario al que se le pueden agregar más palabras).

Esta función de validación puede ser asignada a los perfiles para que todas las cuentas utilicen contraseñas robustas. El parámetro que debe modificarse es `PASSWORD_VERIFY_FUNCTION` y se recomienda ponerlo en todos los perfiles que se creen.

Administración de privilegios

Ya que en las bases de datos es donde se almacena la mayoría de la información de una empresa, es necesario tener identificados los privilegios y accesos hacia los datos, que se asignan a cada cuenta. En este apartado se revisarán los privilegios que deben ser limitados a los usuarios. Al igual que en puntos anteriores, es necesario realizar un análisis previo para las cuentas de aplicaciones con el objeto de evitar que haya afectación al servicio que provee cada plataforma.

• Privilegios sobre objetos del sistema y privilegios de sistema

Con "objetos de sistema" nos referimos a aquellas tablas, vistas y detonantes (*triggers*), entre otros, que son propios de la base de datos. Estos objetos deben ser protegidos para que no cualquier usuario pueda acceder a ellos para consultarlos o manipularlos. La mayoría de estas asignaciones se realizan en cuentas de administración de la base de datos, como

son cuentas de *cluster*, de respaldos y de auditoría. Las cuentas de aplicaciones y de usuarios finales no deben contar con ningún tipo de privilegio para estos objetos.

Además de privilegios sobre objetos, también se pueden asignar privilegios de sistema; éstos también deben ser asignados de manera restringida a las cuentas. Entre otros podemos encontrar creación, borrado y modificación de objetos y asignación de espacios (*tablespaces*) ilimitados. En algunas ocasiones estos privilegios deben ser asignados a cuentas de sistema y de aplicaciones; sin embargo, para los usuarios finales no deberían existir este tipo de asignaciones.

• Privilegios sobre objetos de cuentas

Al igual que los privilegios a objetos de sistema, también se deben restringir los correspondientes a los objetos de las cuentas, haciendo énfasis en los objetos de las cuentas de aplicaciones. Estos privilegios únicamente deberían asignarse entre cuentas de aplicaciones y de auditoría; dejando fuera a todos los usuarios finales. En caso de que alguna cuenta requiera de privilegios hacia estos objetos, sólo se le deberían asignar de consulta.

• Asignación de roles

Para administrar de manera ordenada los privilegios que se trataron en los dos puntos anteriores, Oracle maneja el uso de roles a los cuales se les pueden asignar privilegios. Se recomienda que la asignación de privilegios a una cuenta sea mediante el uso de roles; de esta forma se tendrá un mejor control sobre ellos.

La manera de crear un rol, asignarle y revocarle privilegios es la siguiente:

```
SQL> CREATE ROLE NOMBRE_ROL;
SQL> GRANT PRIVILEGIO TO NOMBRE_ROL;
SQL> REVOKE PRIVILEGIO FROM NOMBRE_ROL;
```

Oracle también posee una serie de roles predefinidos con diversos privilegios asignados. Es necesario revisar que todas las cuentas tengan sólo aquellos privilegios que requieran para el desarrollo de sus actividades y no más.

La tabla donde se pueden consultar los roles que tienen asignados las cuentas es: `DBA_ROLE_PRIVS`

La tabla donde se puede ver qué privilegios tiene cada rol es: `ROLE_SYS_PRIVS` (para privilegios de sistema) y `ROLE_TAB_PRIVS` (para privilegios sobre objetos).

Para asignar o retirar un rol a una cuenta es necesario realizar lo siguiente:

```
SQL> GRANT NOMBRE_ROL TO NOMBRE_CUENTA;
SQL> REVOKE NOMBRE_ROL FROM NOMBRE_CUENTA;
```

Configuración del proceso de escucha

El escucha es un proceso que provee la conectividad de red con la base de datos Oracle. Está configurado para escuchar las peticiones que se hacen a la base de datos. Toda la información de este proceso está contenida en el archivo `$ORACLE_HOME/network/admin/listener.ora` y es aquí donde se pueden definir varios parámetros para asegurar el proceso.

• Uso de contraseña

El paso más importante para asegurar al escucha es asignarle una contraseña. Una vez que se ha definido una contraseña y se han guardado los cambios, el escucha nos solicitará la contraseña para cualquier cambio administrativo que se requiera hacer (por ejemplo, dar de baja o iniciar el proceso de escucha).

Para el caso de bases de datos que se encuentren trabajando en *cluster* esta opción es poco viable ya que se tendrán que reconfigurar las reglas del *cluster* para indicarle la contraseña del escucha.

• Puerto de escucha

Otra modificación que se puede realizar es cambiar el puerto por el cual el proceso trabaja. Por omisión se utiliza el puerto 1521. Para llevar a cabo este cambio es necesario validar todas las conexiones que realizan las aplicaciones y ver si es posible modificar el puerto al cual apunta el escucha, ya que una vez que sea cambiado, todas las aplicaciones deberán modificarse para que sus peticiones las hagan hacia el nuevo puerto.

• Extproc

Por omisión, Oracle configura el escucha con un servicio que se utiliza para

la ejecución de procedimientos externos (`extproc`); por seguridad, si este servicio no se requiere, debe eliminarse. Para ello, se debe editar el archivo `$ORACLE_HOME/network/admin/listener.ora` y quitar todas las llamadas que se hagan a este servicio.

Protección de archivos de configuración y administración

Ya que se han hecho algunas recomendaciones para proteger la base de datos, también es necesario proteger los archivos del sistema operativo propios de la base de datos, como son: archivos de configuración, scripts de administración y mantenimiento de la base de datos.

• Archivos de configuración
Es importante que este tipo de archivos sólo tengan permisos para el propietario y posiblemente para el grupo. En algunas ocasiones se deberán asignar también permisos de lectura a todas las cuentas del equipo para los archivos `$ORACLE_HOME/network/admin/listener.ora` y `$ORACLE_HOME/network/admin/tnsnames.ora`.

• Scripts de creación
Cuando se instala Oracle y se crea una base de datos, se utilizan varios scripts. Es importante que estos archivos se eliminen ya que en ellos se encuentra la definición de todas las estructuras que conforman la base de datos. En caso de que no quieran eliminarse del equipo será necesario asignar permisos únicamente al propietario de los archivos.

• Scripts de mantenimiento
Para la administración y mantenimiento de bases de datos se crean varios scripts de auditoría, depuración, administración, etcétera. Es necesario que estos archivos sólo tengan permisos para el propietario y en algunos casos para el grupo también.

Conclusiones

Las empresas se ven obligadas a implementar medidas de seguridad en los sistemas de cómputo debido a las regulaciones que se requieren para pertenecer a ciertos grupos; sin embargo, asegurar las plataformas no debe hacerse por obligación sino por iniciativa propia para la protección de sus recursos y su información.

Implementar esquemas de seguridad en una empresa no sólo implica proteger los equipos de cómputo, sino también crear una conciencia de seguridad en todo el personal que labora en la empresa, pues el elemento humano es el eslabón más débil en la cadena que conforma la seguridad.

Implementar mecanismos de seguridad conlleva una serie de procesos, estándares y trabajo compartido entre diversas áreas; si alguno de estos componentes no se cumple correctamente la introducción de los mecanismos será complicada y tardada. Por ello es muy importante tener bien definidos estos procesos y estándares antes de empezar a implementar esquemas de seguridad en una empresa; de igual modo, todas las áreas involucradas deben tener bien especificadas las actividades y responsabilidades que les correspondan dentro del proceso de seguridad.

Hay que tener en mente la difusión y capacitación hacia el personal que opera, administra y explota las plataformas, ya que tendrá que adaptarse a las nuevas formas de trabajo para cumplir con la seguridad; también de estas personas dependerá que los mecanismos de seguridad se mantengan trabajando correctamente.

Para el mantenimiento de las plataformas, es necesario vigilarlas constantemente a fin de revisar que las configuraciones iniciales de seguridad no hayan cambiado debido a actualizaciones e instalaciones, entre otras actividades. Un elemento importante para este punto es que el área responsable de la seguridad esté trabajando en conjunto durante los procesos de las actualizaciones o instalaciones, para detectar todo cambio y validar y corregir lo que sea necesario para el nuevo funcionamiento de la plataforma. Otro tipo de mantenimiento consiste en estar actualizando

los niveles de seguridad con los que cuenta una plataforma, para ello es importante buscar nuevas configuraciones para proteger cada vez más los equipos de una empresa.

Como se puede observar en este trabajo, en lo referente a cuestiones técnicas, hay configuraciones muy sencillas que se pueden implementar sin ningún problema; sólo es necesario hacer un análisis, una correcta implementación y la documentación de los cambios. Lo más importante es una buena difusión de los cambios que se realizaron para propiciar su buen uso, administración y mantenimiento por parte de las áreas involucradas.

Además de las configuraciones de seguridad, es muy importante la reestructuración de las actividades que realice cada área, identificando y limitando estas tareas; esto se debe hacer para cada una de las áreas con el fin de evitar que, por duplicidad de actividades, se lleguen a presentar problemas en las plataformas debido a malas configuraciones o mala administración, entre otros aspectos. Hay que estar conscientes de que un equipo de cómputo nunca estará completamente protegido; sin embargo, nuestro trabajo es minimizar el riesgo de que sea vulnerado.

Este trabajo no contempla todas las configuraciones que se pueden aplicar a una plataforma en relación al sistema operativo y bases de datos, pero es un punto de partida para generar conciencia en los administradores de que por algo se debe empezar. Tampoco cubre todos los sistemas operativos y bases de datos disponibles en el mercado, pero ayudará a conocer un poco más las diferentes vulnerabilidades con las que cuenta un equipo de cómputo y así, si no es el mismo sistema o base de datos, se tendrá una idea de lo que puede ser modificado en éstos otros.

Podemos decir que, en general, este trabajo es una introducción al área de seguridad que, si bien es muy grande y compleja, debe atacarse poco a poco para proteger la información que alberga una empresa o institución, sea pública o privada.

Dentro de mi experiencia en una empresa privada, puedo decir que ha sido complicado el introducir todos estos mecanismos para proteger la información. En un inicio toda la gente que labora en la empresa se mostró renuente a las nuevas formas de trabajo, haciendo nuestra función más difícil. Una vez que la gente empezó a comprender las ventajas que tiene esta nueva forma de trabajo, empezó a presentarse una mejor disposición de las áreas para realizar todas los actividades que conllevan la introducción de la seguridad, hasta el punto en que los mismos administradores

son los que llaman al departamento de seguridad para solicitar que sean protegidas sus plataformas.

Aun con esto, nos falta mucho camino para poder decir que trabajamos dentro de un marco de seguridad; porque, si bien no todos, hay personas a las que todavía se les dificulta trabajar dentro de un esquema de seguridad, ya que piensan que el introducir ciertos mecanismos les va a quitar control o poder en sus áreas; es muy importante estar conscientes que la seguridad no implica quitar el control, sino simplemente saber qué se está haciendo con ese control.

Como ya lo mencioné, no es una tarea fácil pero tampoco imposible. En el tiempo que he trabajado en el campo de seguridad, me he percatado de que se puede realizar todo aquello que requiera un usuario final, de aplicaciones o de sistema; sólo es cuestión de buscar el mejor mecanismo de seguridad para no afectar la operación de estas cuentas.

También hay que mencionar que es mejor llevar a cabo acciones preventivas y no correctivas. En muchas ocasiones se introduce una herramienta de seguridad o se hace alguna configuración en una plataforma porque ya sucedió algo que comprometió o que puso en riesgo a ésta; hay que lograr que estas situaciones se vayan reduciendo y, en su lugar, se lleven a cabo instalaciones o configuraciones de una forma preventiva.

Otro punto importante que he encontrado, es que hay ocasiones en que no se pueden implementar todos los mecanismos de seguridad que uno quisiera, porque pueden provocar problemas en el funcionamiento de las plataformas y, como sabemos, lo principal en una empresa es que se dé el servicio que se está vendiendo.

Como punto final, hay que estar conscientes de que la seguridad depende de todos aquellos que tienen acceso a la información a través de los equipos, llámense administradores, usuarios finales o soporte, entre otros.

Información complementaria

Ley Sarbanes-Oxley (SOX)

Esta ley se creó en el año 2002 en Estados Unidos, después de los diversos escándalos financieros que se suscitaron en varias empresas como Enron[8], Tyco Internacional y WorldCom. Estos escándalos derivaron en la falta de confianza en los sistemas de contabilidad y auditoría y, por lo tanto, en la pérdida en el valor de las acciones así como en la confianza de los inversionistas. La legislación abarca y establece nuevos estándares para los consejos de administración, dirección y mecanismos contables de todas las empresas que cotizan en la bolsa de los Estados Unidos; evitando que las acciones de las mismas sean alteradas de manera dudosa.

Su objetivo primordial es restablecer la confianza de la gente en los mercados de capitales; entre otros objetivos se encuentran:

- Fortalecer el gobierno corporativo.
- Lograr una mayor transparencia.
- Reforzar la responsabilidad de los directores de las compañías y de los profesionales que trabajan en ellas.
- Reforzar la independencia del auditor.
- Aumentar la supervisión y ampliar las sanciones por acciones indebidas.

Esta ley se conforma de 11 secciones:

[8] Enron, empresa energética de los EEUU. En 2001, se presentó en quiebra con deudas por más de us$30.000 millones poco después de declarar ganancias por us$1.000 millones y hacer estallar los paneles bursátiles con alzas espectaculares.

El escándalo contable que consistió en maniobras fraudulentas para disimular pasivos abismales, salpicó rápidamente a la consultora Andersen, una de las firmas más importantes de auditoria contable. Las repercusiones llegaron incluso al Congreso de los EEUU, que decidió votar en el 2002 las leyes Sarbanes Oxley destinadas a proteger las empresas de defraudación y fortalecer los controles internos y externos y establecer reglas de gobierno corporativo [16].

- Buró para la vigilancia de la contabilidad de compañías públicas (*Public Company Accounting Oversight Board PCAOB*).
- Independencia del auditor externo.
- Responsabilidad corporativa.
- Revelaciones financieras .
- Conflicto de intereses.
- Comisión de recursos y autoridad.
- Reportes y estudios.
- Responsabilidad corporativa y sobre el fraude criminal.
- Penas para crímenes de cuello blanco.
- Impuestos corporativos.
- Fraude corporativo y responsabilidades.

Control Objectives Information Technology (COBIT)

COBIT (Objetivos de Control para la información y Tecnologías relacionadas) es una metodología que brinda buenas prácticas a través de un marco de trabajo para el manejo de información creado por ISACA *(Information Systems Audit and Control Association)* e ITGI *(IT Governance Institute)* en 1992.

En su cuarta edición, COBIT tiene 34 objetivos de alto nivel que cubren 318 objetivos de control (específicos o detallados) clasificados en cuatro dominios: Planificación y organización, Adquisición e implementación, Entrega y soporte; y Supervisión y evaluación.

La misión de COBIT es "investigar, desarrollar, publicar y promocionar un conjunto de objetivos de control generalmente aceptados para las tecnologías de la información que sean autorizados (dados por alguien con autoridad), actualizados, e internacionales para el uso del día a día de los gestores de negocios (también directivos) y auditores." Gestores, auditores y usuarios se benefician del desarrollo de COBIT porque les ayuda a entender sus sistemas de información (o tecnologías de la información) y decidir el nivel de seguridad y control que es necesario para proteger los activos de sus compañías mediante el desarrollo de un modelo de administración de las tecnologías de la información (TI).

Para que las TI tengan éxito en satisfacer los requerimientos del negocio, la dirección debe implantar un marco de trabajo. El marco de trabajo de COBIT contribuye a estas necesidades de la siguiente forma:

- Estableciendo un vínculo con los requerimientos del negocio.
- Organizando las actividades de TI en un modelo de procesos.
- Identificando los principales recursos de TI a ser utilizados.
- Definiendo los objetivos de control gerenciales.

Podemos decir que COBIT se utiliza mundialmente por aquellos que tienen la principal responsabilidad en procesos de negocio enfocados en la tecnología, aquellos que dependen de la tecnología para la obtención de información fidedigna y aquellos que ofrecen la calidad, confiabilidad y control en los mecanismos de TI.

Computer Emergency Response Team (CERT)

El CERT es un equipo de respuesta de emergencias a incidentes de seguridad. Fue creado por DARPA (*Defense Advanced Research Projects Agency*) en 1988 para centralizar y difundir toda la información sobre los ataques y las fallas de seguridad en la red y los sistemas informáticos.

El CERT investiga, analiza, y desarrolla tecnología para la protección y remedio de vulnerabilidades, además de enfocarse en el entrenamiento en esta tecnología para ayudar a los administradores a asegurar sus sistemas y redes. Basándose en su larga experiencia en el campo de la investigación de vulnerabilidades, el CERT está involucrado con el remedio de vulnerabilidades publicando resultados en el CERT Knowledgebase y en las notas sobre vulnerabilidades en el US-CERT. Su investigación sobre la supervivencia de los sistemas ha contribuido a ampliar el campo de investigación en áreas tales como el aseguramiento de la información de los sistemas, ingeniería computacional para el aseguramiento de software y sistemas, y requerimientos de seguridad.

El CERT también realiza cursos de formación y certificación para administradores de redes y sistemas en áreas como la seguridad de la información, el manejo de incidentes y el análisis forense.

Center for Internet Security (CIS)

El CIS (*Center for Internet Security*) es una empresa no lucrativa cuya misión es ayudar a las organizaciones a reducir los riesgos de negocio derivados de la falta de controles de seguridad informática, lo que se ve traducido en fallas técnicas o ataques deliberados.
Los miembros del CIS desarrollan y fomentan el uso estandarizado de los parámetros de configuración de seguridad a través de un proceso de consenso entre participantes de los sectores públicos y privados.
Los puntos de referencia del CIS sirven de apoyo de alto nivel para los estándares que se basan en las 4W "*Why*, *Who*, *When*, y *Where*"; aspectos básicos de la seguridad de TI.

Power Broker

Symark Software es una compañía de software de seguridad que se especializa en la identidad y gestión de accesos. Se centra en la solución de las deficiencias de seguridad inherentes en los sistemas operativos UNIX, Linux y Windows.

Symark se caracteriza por el rápido desarrollo, administración central y registros detallados de auditoría en todos sus productos. Éstos ayudan a reducir el creciente riesgo en la seguridad de las operaciones con información privilegiada; facilitan la rendición de cuentas y el cumplimiento de las leyes que se tienen hoy en día.

Symark Power Broker permite la delegación selectiva de privilegios administrativos de sistemas UNIX/Linux a cuentas de confianza, sin permitir el acceso como el administrador del sistema o a las cuentas de aplicaciones, reduciendo así el riesgo de daños accidentales o intencionales. También administra privilegios y derechos de acceso a aplicaciones y cuentas de terceros, incluyendo cuentas genéricas.

Las principales características de este software son:

- **Restricción del acceso como** root **de UNIX/Linux**
  Elimina la necesidad de revelar la contraseña de `root`.

• **Delegación de privilegios granulares**
Administra la delegación de funciones administrativas; restringe el acceso a comandos específicos, así como a archivos, estructuras de archivos y aplicaciones de terceros.

• **Formulación de políticas sólidas**
Permite a los administradores crear políticas integrales personalizadas, de acuerdo con las especificaciones de su entorno, utilizando lenguajes conocidos tipo C.

• **Administración central**
Aplica las políticas entre redes UNIX/Linux heterogéneas desde una sola máquina.

• **Registro de auditoría indeleble**
Centraliza el registro de eventos, solicitudes y sesiones de usuario completas, por pulsación de teclado de cada sistema.

• **Arquitectura cliente/servidor**
La tolerancia a fallas garantiza una disponibilidad continua ya que pueden ser configurados varios servidores para atender las peticiones de un cliente.

IPLocks

Fundada en 2002 y con sede en Silicon Valley, **IPLocks** es una corporación privada. Sus productos y servicios ayudan a mejorar la seguridad y la protección de la información, con el fin de resguardar a las empresas y los clientes de peligros de fraude y datos expuestos.

Es una empresa de soluciones de software que ayudan a mantener las bases de datos protegidas. **IPLocks** protege contra los ataques más comunes hacia las bases de datos, supervisa el uso de las mismas, alerta sobre actividades sospechosas e identifica eventos maliciosos o fraudulentos.

Mediante la supervisión y auditoría, **IPLocks** ayuda a las empresas en el cumplimiento de los requerimientos regulatorios y objetivos de seguridad

interna. **IPLocks** se usa comúnmente para apoyar el cumplimiento de regulaciones como SOX.

IPLocks *Database Security & Compliance* es una solución que ayuda a la supervisión y auditoría de bases de datos. Esta solución se usa para cubrir puntos como:

- Control de cambios.
- Supervisión de privilegios de cuentas.
- Protección a la privacidad.
- Prevención del robo de identidad.
- Controles internos.
- Seguridad.

Archivo /etc/passwd

El archivo `/etc/passwd` contiene los atributos de las cuentas declaradas en el sistema operativo. Es un archivo en formato ASCII que contiene una entrada por cada cuenta. Cada línea define los atributos básicos aplicados a la cuenta.

Cada entrada en el archivo tiene la siguiente forma:

```
User:Password:UID:GID:Comment:HomeDirectory:Shell
```

donde:

`User` es el nombre de la cuenta.

`Password` es la contraseña de acceso (en todos los casos que estamos revisando contiene una x, excepto para HP-UX).

`UID` es el número con el que el sistema operativo identifica la cuenta.

`GID` es el número con el que el sistema operativo identifica el grupo asignado al usuario.

`Comment` es un comentario sobre la cuenta.

`HomeDirectory` es el directorio de trabajo de la cuenta.

`Shell` es el intérprete de comandos (*shell*) de la cuenta.

Archivo /etc/shadow

El archivo `/etc/shadow` contiene toda la información referente a las contraseñas de las cuentas. Por seguridad sólo el usuario `root` tiene acceso a este archivo. Cada línea define los atributos de la contraseña de una cuenta. Cada registro en el archivo tiene la siguiente forma:

```
User:Password:Date:min:max:warn:inactivity:exp:flag
```

donde:

`User` es el nombre de la cuenta.
`Password` es la contraseña de acceso cifrada.
`Date` es la fecha del último cambio de la contraseña.
`Min` número de días en el que no puede ser cambiada la contraseña.
`Max` días en que es válida la contraseña.
`Warn` Número de días antes de la expiración de la contraseña en que se le avisará al usuario del vencimiento.
`Inactivity` Días después de la expiración en que la contraseña se inhabilitará, si es que no se cambió.
`Exp` fecha en que la contraseña expirará y por lo tanto se bloqueará la cuenta.
`Flag` Reservado para uso futuro.

Archivo /etc/profile

El archivo `/etc/profile` contiene variables de ambiente del sistema operativo entre otras cosas. Se utiliza para crear un entorno general de trabajo para todas las cuentas.

Archivo /etc/inetd.conf

`/etc/inetd.conf` es el archivo de configuración para el demonio inet. En este archivo se especifican todos aquellos servicios que van a ser administrados por el demonio.
Para mayor información sobre la sintaxis de este archivo se puede consultar el manual disponible en los sistemas operativos (`$ man inetd.conf`).

Directorio /etc/xinetd.d

En este directorio se encuentra toda la configuración de cada uno de los servicios que administra el demonio `xinet`. Es muy importante tener en cuenta que cada archivo que se localiza en este directorio debe tener el nombre del servicio que se está configurando.

Recursos que pueden limitarse mediante un perfil de base de datos

- **Recursos de kernel**
  - `CONNECT_TIME`. Tiempo de conexión válido por sesión.
  - `CPU_PER_CALL`. Uso máximo de CPU por llamada.
  - `CPU_PER_SESSION`. Uso máximo de CPU por sesión.
  - `IDLE_TIME`. Tiempo de inactividad permitido antes de cerrar una sesión.
  - `LOGICAL_READS_PER_CALL`. Número máximo de bloques de datos que se pueden leer en una operación.
  - `LOGICAL_READS_PER_SESSION`. Número máximo de bloques de datos que se pueden leer en una sesión.
  - `PRIVATE_SGA`. Uso de espacio privado en el SGA.
  - `SESSIONS_PER_USER`. Número de sesiones concurrentes permitidas por cuenta.
  - `COMPOSITE_LIMIT`. Suma del uso de `CPU_PER_SESSION`, `CONNECT_TIME`, `LOGICAL_READS_PER_SESSION` y `PRIVATE_SGA`.

- **Recursos de contraseñas**
  - `FAILED_LOGIN_ATTEMPTS`. Número de intentos fallidos de contraseña antes de que la cuenta sea bloqueada.
  - `PASSWORD_GRACE_TIME`. Número de días después de que expira la contraseña en la que ésta puede ser cambiada.
  - `PASSWORD_LIFE_TIME`. Vigencia de contraseña. Número de días en que la misma contraseña es válida.
  - `PASSWORD_LOCK_TIME`. Tiempo en el que una cuenta permanecerá bloqueada después de intentar acceder a la base de datos, de ma-

nera fallida, el número de veces especificado en FAILED_LOGIN_ATTEMPTS.
- PASSWORD_REUSE_MAX. Número de veces que una contraseña puede reutilizarse.
- PASSWORD_REUSE_TIME. Número de días en los que se puede reutilizar una contraseña.

- **Verificación de contraseñas**
  - PASSWORD_VERIFY_FUNCTION. Función para verificar longitud, contenido y complejidad de las contraseñas.

Parámetros de control del escucha

A continuación se listan los parámetros de control que pueden definirse para modificar el comportamiento del proceso de escucha.

- ADMIN_RESTRICTIONS_*listener_name*. Para restringir la administración del escucha en tiempo real.
- INBOUND_CONNECT_TIMEOUT_*listener_name*. Tiempo en el que el cliente debe completar su petición de conexión.
- LOG_DIRECTORY_*listener_name*. Dónde se guardará el archivo de los registros que genera el escucha.
- LOG_FILE_*listener_name*. Archivo donde serán guardados los registros del escucha.
- LOGGING_*listener_name*. Activa la generación de registros.
- PASSWORDS_*listener_name*. Uso de contraseña para la administración del escucha.
- SAVE_CONFIG_ON_STOP_*listener_name*. Sirve para permitir o impedir cambios en tiempo real del escucha.
- SSL_CLIENT_AUTHENTICATION. Uso de SSL (*Secure Sockets Layer*) para la autenticación.
- STARTUP_WAIT_TIME_*listener_name*. Tiempo que tarda el escucha en responder una petición del comando START de la utilería Listener Control.
- TRACE_DIRECTORY_*listener_name*. Directorio donde se guardarán los archivos *trace* (archivos de rastreo) del escucha.

- TRACE_FILE_*listener_name*. Nombre de los archivos *trace* del escucha.
- TRACE_FILELEN_*listener_name*. Tamaño de los archivos *trace* del escucha.
- TRACE_FILENO_*listener_name*. Número de archivos *trace* del escucha que se van a estar utilizando.
- TRACE_LEVEL_*listener_name*. Tipo de información que será registrada en los archivos *trace* del escucha.
- TRACE_TIMESTAMP_*listener_name*. Agrega la fecha y tiempo de creación a los registros de los archivos *trace* del escucha.
- WALLET_LOCATION. Ubicación de los certificados, llaves y puntos de confianza utilizados por SSL.

Referencias

[1] http://csrc.nist.gov
[2] http://docs.hp.com/en/B2355-90121/ch01.html
[3] http://docs.hp.com/es/5992-3422/ch05ss02.html
[4] http://download.oracle.com/docs/cd/B10500_01/network.920/a96581/listener.htm
[5] http://download.oracle.com/docs/cd/B10501_01/network.920/a96581/listener.htm#460305
[6] http://en.wikipedia.org/wiki/COBIT
[7] http://en.wikipedia.org/wiki/Sarbanes-Oxley_Act
[8] http://es.wikipedia.org/wiki/Cluster_(informática)
[9] http://es.wikipedia.org/wiki/COBIT
[10] http://es.wikipedia.org/wiki/Hacker
[11] http://es.wikipedia.org/wiki/Red_privada_virtual
[12] http://es.wikipedia.org/wiki/Sarbanes-Oxley_Act
[13] http://es.wikipedia.org/wiki/System_V
[14] http://es.wikipedia.org/wiki/TCP_Wrapper
[15] http://labmice.techtarget.com/articles/securingwin2000.htm
[16] http://management.infobaeprofesional.com/notas/33598-Caso-Enron-ley-penal-argentina-en-linea-con-normas-de-EEUU.html?cookie
[17] http://software.hp.com/portal/swdepot/displayInstallInfo.do?productNumber=ShadowPassword
[18] http://software.hp.com/portal/swdepot/displayProductInfo.do?productNumber=ShadowPassword
[19] http://www.booktou.com/node/119/0596003919_linuxsckbk-chp-3-sect-10.html
[20] http://www.bsecure.com.mx/ultimos-articulos/ernst-young-encuesta-la-seguridad-en-mexico/
[21] http://www.cert.org
[22] http://www.cert.org/security-improvement/#unix
[23] http://www.cisecurity.org/index.html

[24] http://www.cyberciti.biz/faq/howto-configure-shell-logout-user-automatically/
[25] http://www.faqs.org/docs/securing/chap6sec64.html
[26] http://www.faqs.org/faqs/hp/hpux-faq/section-68.html
[27] http://www.freebsd.org/doc/es/books/handbook/tcpwrappers.html
[28] http://www.infor.uva.es/~jvegas/cursos/bd/oraseg/oraseg.html
[29] http://www.infor.uva.es/~jvegas/cursos/bd/oraseg/oraseg.html
[30] http://www.iplocks.com/
[31] http://www.isaca.org
[32] http://www.orasite.com/tutoriales/seguridad-listener-oracle10g.html
[33] http://www.owasp.org
[34] http://www.psoug.org/reference/profiles.html
[35] http://www.redhat.com/docs/manuals/enterprise/RHEL-4-Manual/en-US/Reference_Guide/s2-tcpwrappers-xinetd-config-files.html
[36] http://www.symark.com/products/pboverview.html
[37] http://www.symark.com/spanish/pdf/pb_datasheet_spanish.pdf
[38] http://www.sun.com/bigadmin/collections/security.html
[39] http://www.windowsecurity.com
[40] http:// www.xinetd.org

Bibliografía

Cobit 4.0 en español.

THERIAULT, Marlene. NEWMAN, Aaron. Oracle Manual de seguridad. Oracle Press. 2002.

ROSEN, Kenneth. HOST, Douglas.FARBER James. ROSINSKI, Richard. UNIX: The Complete Reference. Osborne/McGraw-Hill. 1999.

Windows Server 2003 Checklist 5.1.6, Defense Information Systems Agency.

WATTERS, Paul A.Solaris 10: The Complete Reference. Osborne/McGraw-Hill. 2005.