UNIVERSIDAD INTERNACIONAL DE LA RIOJA

Universidad Internacional de La Rioja
Máster universitario en Seguridad Informática

Bastionado de un servidor de BBDD y su validación mediante auditoría

Trabajo Fin de Máster

Presentado por: Díaz García, Pedro

Director/a: Castillo Sanz, Andrés

Ciudad: Alcalá de Henares
Fecha: 22 de julio de 2014

Esta obra está dedicada a mi esposa Amelia,

mi hija Chary, y mis padres Juan y Cristina

"El que se tenga por sabio y prudente, demuestre con su buena conducta que sus actos tienen la sencillez propia de la sabiduría. Pero si ustedes están dominados por la rivalidad y por el espíritu de discordia, no se vanagloríen ni falten a la verdad. Semejante sabiduría no desciende de lo alto sino que es terrena, sensual y demoníaca. Porque donde hay rivalidad y discordia, hay también desorden y toda clase de maldad. En cambio, la sabiduría que viene de lo alto es, ante todo, pura; y además, pacífica, benévola y conciliadora; está llena de misericordia y dispuesta a hacer el bien; es imparcial y sincera"

Santiago 3

Tabla de Contenidos

0x000 Índice de ilustraciones y tablas

1x000 Resumen

La rápida proliferación de los sistemas de información y la facilidad de acceso a las mismas, ha llevado aparejada también un enorme aumento de las amenazas contra esos sistemas.

El propósito del presente trabajo es, en primer lugar, definir una metodología para la instalación y configuración segura de un servidor de bases de datos y en segundo lugar, definir una auditoría de seguridad con la que se verifique si un servidor está configurado de acuerdo a lo establecido en dicha metodología de instalación y configuración.

Para definir dicha metodología, hay que analizar el contexto y los riesgos. En base a ese análisis hay que definir las medidas de seguridad concretas para poder mitigarlos, y por último realizar los controles y auditorías necesarios para comprobar la correcta implementación de dichos controles y para validar su efectividad a la hora de mantener seguro un sistema determinado.

Palabras Clave: Seguridad, Bases de Datos, Buenas Prácticas, Auditoría.

Abstract

The rapid proliferation of information systems and the ease of access to them, has led also rigged a huge increase in threats against such systems.

The purpose of this paper is, first, to define a methodology for the safe installation and configuration of a server database and secondly, to define a security audit to verify that if a server is configured according to the methodology set out in the installation and configuration.

To define this methodology, we have to analyze the context and risks. Based on that analysis should define the specific security measures to mitigate them, and finally the inspections and audits required to ensure the correct implementation of such controls and to validate its effectiveness in maintaining secure a system.

Keywords: Security, Data Bases, Best Practices, Audit.

2x000 Desarrollo en capítulos.

2x010 Capitulo 1. Introducción

En las últimas décadas ha crecido exponencialmente el uso de los sistemas de información y en especial el de las aplicaciones online, proceso acelerado con la proliferación de dispositivos móviles de todo tipo, con conexión de datos al alcance de cualquier usuario y con la implantación de multitud de servicios en Internet tanto de administración electrónica como comerciales.

En paralelo han aumentado también las amenazas a las que están expuestos dichos sistemas, pero no ha sucedido lo mismo con la seguridad de los mismos. En efecto, solo se ha tomado conciencia de manera generalizada de la necesidad de proteger los sistemas de información, después de haberse dado a conocer casos muy sonados y reiterados de fraude, denegación de servicio y robo de información en todo el mundo.

Con la ejecución de ataques contra los sistemas de información de un país como un elemento más de una estrategia militar, y la aparición desde hace ya algunos años de las APTs[1], la seguridad informática ha dejado de ser solo un mecanismo para garantizar la continuidad de las actividades de cualquier organización ciberdependiente[2], sino que ha además es, a día de hoy, una cuestión de seguridad nacional.

Los ataques informáticos pueden tener objetivos y motivaciones muy diversas, pero quizás la principal sea la obtención de información. La información es el activo más importante para muchas organizaciones y de ahí surge la necesidad de protegerla. Dicha información, por lo general, reside en bases de datos corporativas, a las cuales accederán múltiples aplicaciones desde la propia organización o a través de Internet mediante dispositivos móviles.

Las diversas técnicas de recogida de información y de hacking[3] como por ejemplo hacerse con unas credenciales de usuario, o conseguir una elevación de privilegios, tienen como objetivo último el poder acceder de manera ilegítima a cierta información para extraerla y/o modificarla provocando un grave perjuicio a la víctima. Las motivaciones pueden ir desde el espionaje industrial hasta el ciberterrorismo[4], pero en todo caso existe un problema real y crítico para cualquier organización con la seguridad de sus bases de datos.

El objetivo del presente trabajo es proporcionar una metodología práctica para cualquiera que tenga necesidad de asegurar un sistema corporativo de bases de datos, y también para llevar a cabo una auditoría sobre el mismo que permita verificar dicha seguridad.

Se pretende establecer un marco de buenas prácticas de seguridad que sean aplicables a cualquier SGBDR[5], sin embargo, con el fin de que el trabajo sea lo más ilustrativo posible, trataré de particularizarlas para uno de los sistemas de bases de datos relacionales más utilizados en el mundo: Oracle. De esta manera, se puede ir estableciendo una comparativa entre los mecanismos de seguridad propuestos y su implementación en un sistema concreto.

Como es sabido, Oracle contempla la licencia de desarrollador (**Developer License**) exclusivamente para uso académico, y esto me ha permitido descargar sus productos desde el sitio oficial para realizar el presente trabajo. Los términos y condiciones de la citada licencia se pueden comprobar en la dirección corporativa [a].

2x020 Capitulo 2. Contexto y estado del arte

En este capítulo, expongo la situación actual de la seguridad en BBDD (contexto), una taxonomía de las vulnerabilidades, y las tendencias y métodos actuales para enfrentar el tema.

Como se mencionó en la introducción, debido a que muchos sistemas de información deben ser accesibles desde Internet, y dichos sistemas necesitan acceder a sus bases de datos, estas cada vez están más cerca del perímetro de seguridad de las organizaciones y en consecuencia están también más cerca de los posibles atacantes. Además, con el paso de los años, han ido apareciendo regulaciones que obligan legalmente a contemplar ciertas medidas de seguridad como la Ley Orgánica de Protección de Datos (LOPD) en el caso de España. Todo ello, fuerza a las organizaciones tanto públicas como privadas a adoptar medidas de seguridad que hace pocos años ni siquiera se contemplaban, tanto para poder garantizar la continuidad de sus operaciones como para cumplir con las disposiciones legales vigentes.

En el caso que nos ocupa, solo un conocimiento exhaustivo de la arquitectura y características específicas de un SGBDR concreto, nos puede dar garantías de que su instalación y configuración son las correctas desde el punto de vista de la seguridad. Por ello, en la descripción de la metodología, se indicarán cada una de las prácticas recomendadas de manera genérica, pero a continuación se especificará la manera de llevarlas a la práctica en sistemas específicos.

No hay que perder de vista en ningún momento, que la seguridad de un servidor de bases de datos no solo depende de la adecuada instalación y configuración del SGBDR, sino que también descansa en la seguridad implementada a nivel de sistema operativo y a nivel físico. La seguridad física y la de la plataforma, no son objetivo principal de este trabajo, pero debido al gran peso que tienen en el resultado final, se harán también algunas recomendaciones.

Las investigaciones en el campo de la seguridad en bases de datos, tratan de buscar soluciones a los diferentes fallos de seguridad que de manera genérica se pueden encontrar en cualquier sistema, independientemente de la tecnología empleada. La investigación de incidentes de seguridad concretos, entrarían más bien en el campo de la investigación de incidentes y del análisis forense digital.

El primer paso para empezar a buscar soluciones, es tener bien identificados los problemas. A continuación, se proponen una serie de categorías en las que se pueden clasificar la gran mayoría de los problemas de seguridad que aquí interesan [1]:

- *Unauthenticated Flaws in Network Protocols*
  Permite a un atacante acceder a ciertas funciones sin necesidad de usuario y contraseña y provocar un buffer overflow[6]. La mejor forma de evitarlo es mantener actualizados los parches de seguridad, y permitir conexiones únicamente desde host de confianza autenticados mediante sistemas como SSH[7] o IPsec[8].

- *Authenticated Flaws in Network Protocols*
  El número de bugs encontrados en esta categoría es mucho menor que en la anterior, pero permiten a un atacante conectarse a algún servicio de la base de datos, por lo general con autenticación a nivel de sistema operativo, y ejecutar código malicioso con los privilegios del administrador de la base de datos.

- Flaws in Authentication Protocols
  Algunos sistemas de base de datos emplean protocolos de autenticación muy débiles, en los que las credenciales se envían en claro o con cifrados tan elementales que es como si no existieran (intercambio de posición entre los bits más y menos significativos de una palabra, o empleo de la función XOR). En versiones antiguas de algunos productos, existe un importante error de concepto, por cuanto solo se comprueban los hash[9] de las contraseñas y no las contraseñas en sí mismas.

- *Unauthenticated Access to Functionality*
  Algunos componentes de bases de datos, permiten el acceso sin autenticación, cuando en realidad debería ser un requisito ineludible. Esta vulnerabilidad permite a un atacante ejecutar comandos en el servidor de base de datos, con los privilegios con los que esté corriendo el gestor de base de datos en el sistema operativo. En este caso se ve claramente la importancia de la seguridad de la plataforma sobre la que se van a gestionar las bases de datos.

- *Arbitrary Code Execution in Intrinsic SQL Elements*
  Este tipo de vulnerabilidades se pueden explotar vía inyección SQL[10], por lo que generalmente el problema está en la falta de control de las cadenas de caracteres que se aceptan en los campos de entrada de datos de las aplicaciones web. Se han descrito varias vulnerabilidades que permiten provocar buffer overflow.

- *Arbitrary Code Execution in Securable SQL Elements*
  Aquí la cuestión fundamentalmente, es que algunos usuarios por defecto tienen permisos para crear objetos en la base de datos, o para ejecutar procedimientos almacenados con lo que se puede, una vez más, provocar un desbordamiento de buffer. Desde luego, la mejor práctica posible en este caso es ni más ni menos que erradicar la costumbre de dejar configuraciones por defecto.

- *Privilege Elevation via SQL Injection*
  La mayoría de las organizaciones están familiarizadas con el riesgo asociado a la inyección SQL en aplicaciones web, pero no son conscientes de las implicaciones de dicha inyección en los procedimientos almacenados. Hay que tener en cuenta que cualquier elemento que genera dinámicamente y ejecuta una sentencia SQL, es susceptible de ser vulnerable a este tipo de ataques. De nuevo son importantes los parches de seguridad, y eliminar todos los procedimientos almacenados presentes por defecto y que no sean necesarios para la aplicación.

- *Local Privilege Elevation Issues*
  En esta categoría, se incluyen todos los fallos de seguridad que permiten una elevación de privilegios desde el sistema operativo. En general, la mejor defensa en este caso, consiste en ejecutar el SGBDR con un usuario que tenga el menor nivel de privilegios posible.

Además hay que tener presente que tipo de vulnerabilidades son las más frecuentemente explotadas, para priorizar las medidas de seguridad a tomar. Son varios los estudios reconocidos internacionalmente que se publican periódicamente a este respecto.

Uno de los más recientes (publicado este mismo año) es WP-TOP10-DATABASE-THREATS-0314.1 de IMPERVA [c]. En él se establece un ranking con las 10 principales amenazas a la seguridad en bases de datos a lo largo de 2013, y se compara con ese mismo ranking elaborado en el año 2010.

Como se puede observar, en líneas generales las amenazas son las mismas y en un orden muy parecido pero llama especialmente la atención, que la mayor fuente de incidentes de seguridad es, invariablemente, el exceso de privilegios. Esta cuestión por tanto, deberá ser objeto de especial atención a la hora de configurar las bases de datos y emplear los mecanismos de autorización.

Ranking	2013 Top Threats	2010 Top Threats
1	Excessive and Unused Privileges	Excessive Privilege Abuse
2	Privilege Abuse	Legitimate Privilege Abuse
3	SQL Injection	Privilege Elevation
4	Malware	NEW Exploitation of Vulnerable, Miconfigured Databases
5	Weak Audit Trail	SQL Injection
6	Storage Media Exposure	Weak Audit Trail
7	Exploitation of Vulnerabilities and Misconfigured Databases	Denial of Service
8	Unmanaged Sensitive Data Database	Communication Protocol Vulnerabilities
9	Denial of Service	Unauthorized Copies of Sensitive Data
10	Limited Security Expertise and Education	NEW Backup Data Exposure

En 2012, APPLICATION SECURITY INC. Publicó otro interesante estudio a este respecto, obteniendo el siguiente ranking [d]:

Ranking	Vulnerabilities and Misconfigurations
DB1	Default and Weak Password
DB2	SQL Injection in the DBMS
DB3	Excessive User & Group Privileges
DB4	Unnecessary Enabled DBMS Features
DB5	Broken Configuration Management
DB6	Buffer Overflows
DB7	Privilege Escalation
DB8	Denial of Service
DB9	Unpatched Database
DB10	Unencrypted Data – At Rest and In Motion

La mayoría de las vulnerabilidades detectadas aparecen también en el otro estudio, pero en este caso, la primera posición la ocupa el hecho de que después de la instalación del SGBD se dejan contraseñas por defecto, o se establecen contraseñas débiles vulnerables a los ataques de fuerza bruta o de diccionario. El exceso de privilegios en este caso ocupa la tercera posición, y llama la atención también el disponer de funcionalidades innecesarias que pueden ser objeto de ataques (DB4).

En cualquier caso, para el profesional de seguridad es imprescindible mantenerse al tanto de las nuevas vulnerabilidades de las que se va teniendo conocimiento, para mantener los sistemas bajo su responsabilidad seguros. Para ello existen varias bases de datos de vulnerabilidades que son libremente accesibles. Quizás la principal referencia sea la **National Vulnerability Database** (NVD), proporcionada por la División Nacional de Ciber Seguridad del **Department of Homeland Security** de los Estados Unidos [e].

Además, tiene definida una métrica que permite evaluar el impacto de todas las vulnerabilidades que se publican [f].

A efectos prácticos, para ayudar a implementar seguridad en sistemas de bases de datos, además de compendios de buenas prácticas como **ITIL**[11], existen plantillas específicas a aplicar sobre sistemas muy concretos. En este sentido, son especialmente populares en España, sobre todo en la administración pública las **plantillas STIC**[12] elaboradas por el **CCN**[13]. En concreto, para el bastionado de servidores de bases de datos, a día de hoy están disponibles las siguientes:

- 636 - Configuración Segura Oracle 10g R2 sobre Red Hat 3 y 4 (dic. 07)
- 639 - Configuración Segura Oracle 10g R2 Windows Server 2003 (ago.10)
- 540 - Configuración Segura MS SQL Server 2008 R2 (abr. 14)

Para más información se puede consultar el índice de guías STIC del CCN [g].

El principal problema a la hora de aplicar estas plantillas, radica en que implementan una seguridad tan restrictiva, que en muchos casos es incompatible con la funcionalidad que deben proporcionar las bases de datos.

Hay otras guías de seguridad bastante menos exigentes que las del Centro Criptológico Nacional, como las publicadas por la OTAN o el Departamento de Defensa de Estados Unidos. Para la realización del presente trabajo se ha consultado la siguiente:

- DATABASE SECURITY TECHNICAL IMPLEMENTATION GUIDE. Desarrollada por la Defense Information Systems Agency para el DoD[14]. Muchas de las guías publicadas por el organismo son de acceso libre [h].

Con todo lo anterior, se pueden dar recomendaciones genéricas, pero para entrar en detalles de configuración de sistemas específicos, hay que tener presente el/los probables

usos de las bases de datos a implementar sobre el servidor para compatibilizar el nivel de servicio/funcionalidad y la seguridad.

Evidentemente, el procedimiento de bastionado que se expone en el presente trabajo, trata de dar respuesta a las vulnerabilidades presentes. Para que dicho procedimiento siga siendo eficaz en el futuro no puede ser algo estático, sino que debe ser revisado y actualizado periódicamente a tenor de la evolución de las amenazas en un proceso de mejora continua para mantener la calidad del mismo.

Precisamente el futuro de la seguridad en las bases de datos está siendo objeto de debate en diferentes foros. Algunas de las conclusiones sobre las que parece que hay mayor consenso sobre las tendencias futuras son las siguientes:

La explotación de vulnerabilidades conocidas será más sencilla

Las herramientas disponibles en Internet permiten a los atacantes de todos los niveles técnicos penetrar en bases de datos utilizando sofisticados ataques. Cada anuncio de actualizaciones de Oracle, o de Microsoft, anuncian nuevas vulnerabilidades que son rápidamente desplegadas en estas herramientas.

Un cambio importante que se prevé, es que los ataques más que dirigirse a sistemas concretos, se dirijan a vulnerabilidades concretas.

La seguridad de los datos seguirá siendo un problema para el desarrollo de los sistemas en la nube.

Cloud Computing hace inherentemente más difícil de proteger los datos ya que los mismos se mueven con frecuencia, y las aplicaciones que acceden a estos datos también son ubicuas. Esto evitará que muchas organizaciones pasen a un modelo de Cloud para aplicaciones con requisitos de cumplimiento de auditoría. Precisamente estas cuestiones son mi propuesta para un trabajo futuro.

Presión para cumplir con las regulaciones ligadas a condiciones económicas.

La dicotomía en las empresas por un recorte del gasto y una mayor presión para el cumplimiento regulatorio impulsará a las organizaciones a invertir en soluciones que les ayuden a alcanzar el nivel mínimo de cumplimiento. Las empresas deberán disponer de

tecnología que proporcione una protección adecuada y un costo total de propiedad (TCO) bajo. La facilidad de aplicación, el costo de la entrada y el tiempo de cumplimiento van a ser los criterios para tomar decisiones claves. A medida que la economía se recupere, las principales organizaciones comenzarán a ver el aumento de la seguridad como un elemento diferenciador de su oferta, y la tendrán en cuenta más allá de los requisitos reglamentarios.

Un nuevo tipo de amenaza: ataques externos desde el interior.

En general, las empresas han visto los ataques, como procedentes de fuera del perímetro de la red o de usuarios internos abusando de sus privilegios. Sin embargo, la línea entre lo interno y externo, están desapareciendo como consecuencia de varios vectores de ataques de nuevos:

- La delincuencia organizada dirigida a determinadas empresas mediante la inserción de “dummies” para infiltrar en la organización como empleados o contratistas, con el único fin de obtener acceso a los datos sensibles.

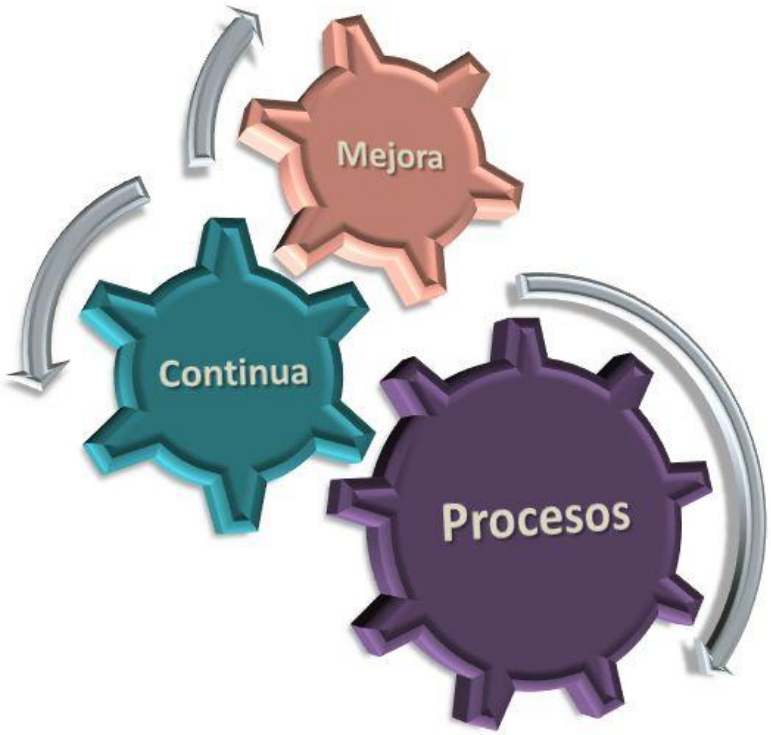
- Los nuevos tipos de malware, que tienen como propósito tomar el control de las máquinas internas y atacar de forma autónoma desde el interior.

- Como las ramificaciones de una economía deprimida sigue siendo causa de aumento y períodos más largos de desempleo, el uso de medios financieros para aprovechar información privilegiada para ayudar a los extranjeros a través de sobornos o extorsión se volverán más comunes.

Soluciones que protegen los datos, independientemente de la fuente del ataque, por lo tanto será un componente esencial de una estrategia de las organizaciones de seguridad.

La reducción al mínimo de la superficie de ataque será fundamental para reducir la exposición.

Cuando los datos sensibles son requeridos por varias aplicaciones, y estos datos se almacenan en aplicaciones a nivel local, las organizaciones utilizan técnicas como tokenización. En lugar de almacenar un número de tarjeta de crédito o número de seguro social, se guardará una muestra de que puede recuperar esta información de forma segura cuando sea necesario, asegurar de que sólo una única base de datos guarde el número real de la tarjeta de crédito. Mediante la protección adecuada de una sola fuente y el cifrado de los datos en tránsito, la exposición se reduce significativamente.



Proceso de Mejora Continua 1

2x030 Capitulo 3. Hipótesis de trabajo y objetivos

Como se ha mencionado en el capítulo anterior, es imprescindible conocer el uso que finalmente se le va a dar a un servidor de bases de datos para implementar así el mayor nivel de seguridad posible sin impedir el correcto funcionamiento de ninguna de las opciones necesarias.

En este caso, partiré del supuesto de que se pretende securizar un servidor de bases de datos al que tendrá que acceder una aplicación web de comercio electrónico. Es un caso mundo, y también por las innumerables amenazas para las que deberían estar preparadas. Innumerables amenazas en primer lugar por el hecho de ser accesibles desde Internet, y además porque cuentan con el aliciente extra de gestionar procesos de compra-venta en los que los clientes facilitan datos personales y números de tarjeta, o de cualquier otro medio de pago electrónico.

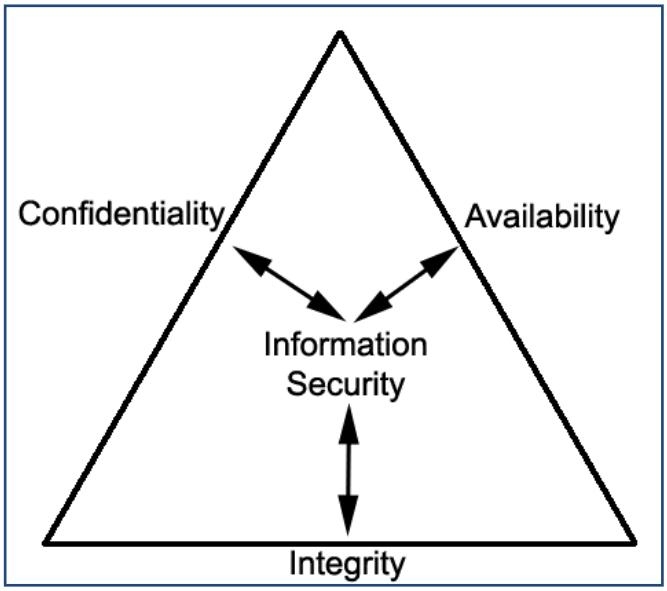
Una vez definida la hipótesis de trabajo, se pueden identificar los objetivos. Los objetivos específicos se han agrupado en dos grandes objetivos generales:

- Definir una metodología para la instalación y configuración segura de un servidor de bases de datos relacionales que formará parte de una solución de comercio electrónico on-line.

- Definir una metodología para llevar a cabo una auditoría de seguridad con la que se verifique si un servidor de bases de datos está configurado de acuerdo a lo establecido en el punto anterior.

La metodología para la instalación y configuración, proporcionará un nivel de seguridad adecuado para evitar posibles intrusiones, modificaciones o robo de la información contenida en la base de datos.

Se trata ni más ni menos de garantizar que se implementan los mecanismos de seguridad sobre los que descansan los 3 pilares de la seguridad que deben tenerse en cuenta en cualquier sistema de información: Confidencialidad, Integridad, y Disponibilidad.



Autenticación-Integridad-Disponibilidad 1

La necesidad de garantizar estas tres características se agudiza por cuanto en la base de datos se van a almacenar datos de carácter personal de los clientes de la tienda on-line. En este punto hay que tomar en consideración las normas vigentes e incluir un análisis de riesgos legales al plantear el diseño de la aplicación (y por tanto también del modelo de datos).

Hay que dejar claro no obstante, que en el presente trabajo queda excluido todo lo relacionado con el rendimiento y la optimización de los parámetros de la base de datos, siendo esto quizás objeto de otra monografía.

Además, una supuesta instalación y configuración segura de cualquier sistema no es suficiente por sí misma, y se debe acompañar siempre de la realización de auditorías periódicas. Precisamente el desarrollo de dichas auditorías es el segundo objetivo de este trabajo:

"La seguridad no es un producto, es un proceso"

***Bruce Schneier*[15]**

2x040 Capitulo 4. Metodología

2x041 Identificación de requisitos

Antes de entrar en cuestiones técnicas, creo importante hacer hincapié en el hecho de que la implantación de seguridad en los sistemas de información, al igual que cualquier actividad, debe estar alineada con los objetivos de la organización y contar además con el impulso e implicación de la alta dirección de la misma, pues de otra forma los esfuerzos estarían condenados al fracaso.

En el caso del servidor de bases de datos, los requisitos de seguridad son los mismos que para cualquier sistema de información: Confidencialidad, Integridad y Disponibilida. Describiré en este apartado cada uno de ellos, y en el siguiente los mecanismos y controles de seguridad concretos que constituyen la metodología de bastionado objeto de este trabajo.

Confidencialidad
Capacidad del sistema para evitar que personas no autorizadas puedan acceder a la información almacenada en él.

Integridad
Permite asegurar que no se ha falseado la información, es decir, que los datos recibidos o recuperados son exactamente los que fueron enviados o almacenados, sin que se haya producido ninguna modificación no autorizada.

Disponibilidad
El sistema se mantiene funcionando eficientemente y es capaz de recuperarse rápidamente en caso de fallo.

Seguridad de la Información 1

Para poder preservar los requisitos de seguridad mencionados, es necesario observar una serie de principios fundamentales:

Principio de menor privilegio

Cualquier objeto (usuario, administrador, programa, sistema, etc.) debe tener tan solo los privilegios de uso necesarios para desarrollar su tarea y ninguno más.

Principio del eslabón más débil

En todo sistema de seguridad, el máximo grado de seguridad no es la suma de toda la cadena de medidas sino el grado de seguridad de su eslabón más débil.

Punto de control centralizado

Se trata de establecer un único punto de acceso a nuestro sistema, de modo que cualquier atacante que intente acceder al mismo tenga que pasar por él. No se trata de utilizar un sólo mecanismo de seguridad, sino de "alinearlos" todos de modo que el usuario tenga que pasar por ellos para acceder al sistema.

Seguridad en caso de fallo

Este principio afirma que en caso de que cualquier mecanismo de seguridad falle, nuestro sistema debe quedar en un estado seguro.

Participación universal

Cualquier mecanismo de seguridad que establezcamos puede ser vulnerable si existe la participación voluntaria de algún usuario autorizado para romperlo.

Simplicidad

Mantener las cosas lo más simples posibles, las hace más fáciles de comprender mientras que la complejidad permite esconder múltiples fallos.

Todos estos principios, están presentes en la metodología que a continuación se va a desarrollar.

2x042 Descripción de la metodología

En este capítulo se describirá de manera detallada el conjunto de buenas prácticas para la instalación y configuración segura de un servidor de base de datos al que va a acceder una aplicación web de ventas por internet. El conjunto de buenas prácticas de este proceso las agruparé de la siguiente forma:

1. Seguridad Física.
2. Pre-requisitos.
3. Seguridad en el Servidor de Bases de Datos

Seguridad Física

Las cuestiones relacionadas con la seguridad física no son objeto del presente trabajo. No obstante, dada la incidencia que tiene en el resultado final, merece la pena recordar que las principales amenazas que se pretenden afrontar con la seguridad física se resumen en lo siguiente:

1. Desastres naturales, incendios accidentales, tormentas e inundaciones.
2. Amenazas ocasionadas por el hombre.
3. Disturbios, sabotajes internos y externos deliberados.

Con el objeto de ayudar a los profesionales de la seguridad a implantar medidas de seguridad física eficaces, existen guías, metodologías y estándares internacionales al respecto. Cabe destacar los siguientes:

1. Estándar de seguridad ISO 17799-2000.
2. Metodologías de análisis de riesgos CRAMM, OCTAVE y MAGERIT.
3. Guía CNI - NS/03 Seguridad Física -.
4. Guía CNI - OR-ASIP-01-02.02 Orientaciones para la constitución de zonas de acceso restringido -.

La última guía está indicada para habilitar Zonas Restringidas (ZR) para albergar sistemas de información que necesiten de acreditación por manejar información clasificada.

Pre-requisitos

Los pre-requisitos para instalar el motor de base de datos se centran fundamentalmente en partir de un servidor bastionado o plataforma segura. Dicho proceso de bastionado depende del sistema operativo servidor específico sobre el que se instalará el SGBD.

Para la instalación y configuración seguras del sistema operativo existen recomendaciones y guías específicas basadas en diferentes versiones de Linux y de Windows.

Como este trabajo se centra en la instalación y configuración del gestor de base de datos, simplemente recordaremos que previamente hay que comprobar lo siguiente:

VERSIONES DE SOFTWARE

Antes de iniciar el proceso, hay que asegurarse de que las versiones del sistema operativo del host y del SGBD son compatibles. Además, el gestor de base de datos debe proporcionar toda la funcionalidad necesaria para la aplicación.

VERSIONES DE SOFTWARE

201. En primer lugar hay que asegurarse que las versiones del sistema operativo y del software gestor de base de datos Oracle a instalar son compatibles.

ACTUALIZACIONES DEL SISTEMA OPERATIVO

La plataforma sobre la que se montará el servidor de base de datos debe estar permanentemente actualizada.

ACTUALIZACIONES DEL SISTEMA OPERATIVO

202. El sistema operativo tiene que estar permanentemente actualizado, en especial en lo que se refiere a los parches de seguridad.

FUNCIONALIDAD MÍNIMA

El sistema operativo de host, solo debe tener instaladas y habilitadas las funcionalidades y servicios mínimos necesarios para cumplir con su función.

FUNCIONALIDAD

203. Es fundamental que únicamente estén disponibles los componentes necesarios del sistema operativo para reducir así la superficie de ataque. Las instalaciones por defecto suelen dejar habilitados muchos servicios que no son necesarios y pueden ser un riesgo para la seguridad.

ANTIVIRUS

El servidor tiene que estar protegido por un software antivirus que esté permanentemente actualizado.

ANTIVIRUS

204. El servidor tiene que estar protegido por un software antivirus que esté actualizado de manera permanente.

CUENTAS DE USUARIO DEDICADAS

El host debe tener cuentas de usuario específicas para la instalación y la ejecución del SGBD. Estas cuentas no se deben utilizar para ninguna otra cosa, y tendrán los privilegios mínimos imprescindibles para cumplir con su función.

CUENTAS DE USUARIO DEDICADAS

205. Se debe contar con cuentas de usuario específicas para la instalación y operación del software de bases de datos. Estas cuentas no se deben utilizar para ninguna otra tarea y deben tener los privilegios mínimos imprescindibles. Es importante que el servidor de base de datos se ejecute con una cuenta de

sistema operativo específica, cuyo nombre no coincida con ninguna de las cuentas por defecto.

SEGURIDAD DEL SISTEMA DE FICHEROS

Los directorios de instalación del SGBD y de las BBDD, y especialmente algunos ficheros de configuración, deben estar protegidos para que únicamente las cuentas mencionadas en el punto anterior puedan acceder a ellos.

SEGURIDAD DEL SISTEMA DE FICHEROS

206. Hay que prestar mucha atención a los permisos sobre ficheros y carpetas para no dejarlos expuestos. Generalmente todos los archivos que se encuentran en un directorio tienen los mismos permisos por lo que hay que tener perfectamente localizados y asegurados determinados archivos importantes del software de base de datos, como por ejemplo INIT.ORA, SQLNET.ORA y LISTENER.ORA en el caso de Oracle.

PERMISOS SOBRE EL REGISTRO DE WINDOWS

En los sistemas Windows, el SGBD debe correr con una cuenta de usuario que no permita introducir modificaciones en el registro de configuraciones.

PERMISOS SOBRE EL REGISTRO DE WINDOWS

207. Si se opta por una plataforma Windows Server, el usuario con el que opera el servidor de base de datos no debe tener permisos que posibiliten la modificación del registro de configuraciones, pues esto constituye un importante riesgo para la seguridad como por ejemplo la posibilidad de crear usuarios ilegítimos (escalada de privilegios).

PROTECCIONES DE RED

El acceso al servidor de bases de datos debe estar restringido para evitar ataques de DoS. Además, se recomienda el empleo de protocolos que garanticen la identidad de los dos servidores extremos de la conexión (de aplicación web y de bases de datos), y el cifrado de los datos que intercambian entre ellos para garantizar la integridad y la confidencialidad de los mismos.

El conjunto de protocolos IPsec proporciona todas las funcionalidades recomendadas aquí, por lo que puede ser una buena solución.

PROTECCIONES DE RED

208. Puesto que la aplicación web tiene que acceder directamente a la base de datos, la configuración de la red a través de la cual han de comunicarse adquiere una importancia crítica. La primera consideración que hay que hacer, es que equipos se van a poder conectar con el servidor de base de datos. En segundo lugar hay que decidir de qué manera se van a configurar los dispositivos y la arquitectura de red para proteger dicho acceso. Una correcta restricción de los accesos al servidor de base de datos es la mejor forma de prevenir un ataque de denegación de servicio (DoS) contra el mismo.

209. Por otro lado, hay que proteger la confidencialidad e integridad de los datos tanto en reposo como cuando viajan por la red, tal y como se verá en el siguiente apartado. Para ello la mejor opción es el cifrado de los mismos. En mi opinión, la opción más recomendable es el empleo de IPsec que permite crear un túnel punto a punto entre el servidor web y el servidor de base de datos, de manera que todos los datos que se intercambien entre ambas máquinas viajen cifrados y además con la garantía de que cada una se autentica frente a la otra.

SEGURIDAD PERIMETRAL Y DEFENSA EN PROFUNDIDAD

En la siguiente página, se propone un esquema de configuración perimetral, que posibilita la conexión de los servidores corporativos con internet en condiciones de seguridad, siguiendo el principio de defensa en profundidad, por lo que se propone una solución con dos firewall, y tráfico de red tunelizado entre el servidor de la aplicación web y el de base de datos.

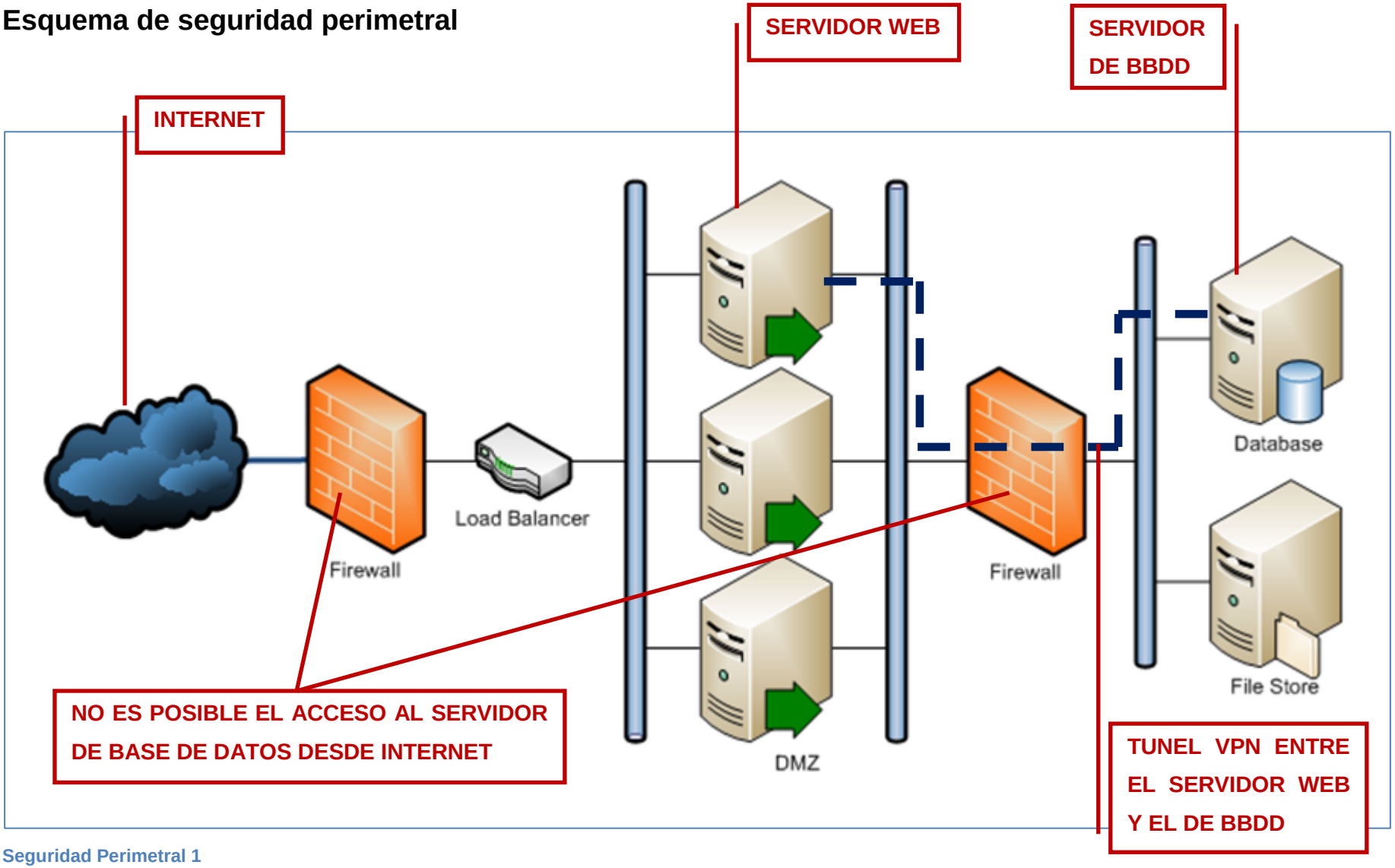
SEGURIDAD PERIMETRAL Y DEFENSA EN PROFUNDIDAD

210. Además de lo indicado en el punto anterior, es importante la arquitectura de una red que va a estar expuesta a internet. En la página siguiente se propone una solución adecuada de seguridad perimetral.

Adicionalmente, se puede aumentar la seguridad con el empleo de un firewall de base de datos[16], intercalado entre la aplicación web y el servidor de BBDD. En el caso de Oracle, tenemos disponible ***Oracle Database Firewall***, pero hay multitud de soluciones en el mercado. Uno de los principios de la seguridad es no dejarlo todo en manos de un solo producto o fabricante, por lo que propongo el empleo de una solución de terceros.

Una opción perfectamente válida es ***GreenSQL Database Security***, que protege las bases de datos frente a intentos de ataque de Inyección SQL, accesos no autorizados, robo de datos, y casos de abuso. Este producto permite crear reglas de firewall para tener un control total sobre quien accede a información sensible, y registra todos los intentos de ataque. La instalación y configuración es muy sencilla y su funcionamiento conocido puesto que en este máster precisamente el empleo de esta herramienta software ha sido una de las actividades desarrolladas. Se puede obtener información adicional y la descarga de las versiones de prueba de los productos de GreenSQL en su sitio oficial [i].

Esquema de seguridad perimetral



Seguridad Perimetral 1

Seguridad en el Servidor de Bases de Datos

Hasta aquí se ha hecho lo necesario para partir de una plataforma convenientemente bastionada sobre la que montar el software de base de datos. En este apartado me centraré en el objeto principal del trabajo que es la configuración segura de dicho software y de las bases de datos.

Voy a tratar de exponer de manera sencilla, las mejores prácticas de seguridad en bases de datos, primero de manera genérica, es decir, para cualquier SGBD (recuadros grises), y a continuación esas buenas prácticas particularizadas para los servidores Oracle.

PROCESO DE INSTALACIÓN DEL SOFTWARE Y DE CREACIÓN DE LA BASE DE DATOS

Se recomienda en cualquier caso, no instalar el software en la ruta por defecto y descartar todos los componentes que no sean estrictamente necesarios. Tampoco se deben dejar los valores por defecto en los nombres de las bases de datos ni en los números de puerto de escucha.

Si existe la posibilidad, hay que almacenar de forma redundante los registros de transacciones.

Los usuarios administradores de BBDD deben tener contraseñas fuertes.

Para terminar, se eliminarán los archivos de instalación innecesarios.

PROCESO DE INSTALACIÓN DEL GESTOR ORACLE Y DE CREACIÓN DE BASES DE DATOS.

El proceso de instalación está basado en una aplicación de instalación con interfaz gráfica. Siempre se ha de elegir la opción ***Instalación avanzada -> Personalizada*** para poder controlar exactamente qué productos se instalan.

301. Cuando el asistente solicite nombre y ruta de instalación, se deben introducir manualmente y no dejar nunca los valores por defecto.

302. En la lista de los ***Componentes Disponibles del Producto***, hay que seleccionar exclusivamente aquellos necesarios según los requisitos funcionales de la aplicación.

303. A continuación el asistente mostrará la opción de creación de una base de datos. Para ello, entre otras cosas, tiene que crear el servicio ***LISTENER***, al cual hay que asignarle un nombre diferente del nombre por defecto. Además añadiremos el protocolo TCPS lo que permitirá conectarse a la base de datos utilizando SSL.

304. También hay que cambiar el número de puerto del LISTENER. Una vez más hay que evitar los valores por defecto.

305. Una vez creado el LISTENER comienza la configuración de la base de datos. En primer lugar hay que darle un nombre, una vez más distinto del predeterminado.

306. Se debe activar la opción ***ARCHIVELOG*** (más adelante se explica con más detalle) durante la instalación, aunque también se puede modificar este parámetro una vez terminada la creación de la base de datos.

307. Finalmente hay que introducir las contraseñas para los usuarios ***SYS*** y ***SYSTEM***. Estas contraseñas deben ser diferentes para cada una de las cuentas y deben ser fuertes: ni contraseñas por defecto, ni nombres de usuario, ni palabras de diccionario, ni contraseñas cortas. Más bien deben ser largas (más de 14 caracteres) y combinar letras mayúsculas y minúsculas con números.

Cuando termine la instalación, el asistente mostrará un resumen. Si es necesario se pueden crear varias bases de datos siguiendo siempre las mismas directrices.

308. Para terminar hay que eliminar los archivos de instalación innecesarios. Si se hace la instalación de los productos básicos, simplemente hay que eliminar la aplicación iSQLPlus si estuviera presente en el sistema. No hay que confundirla con los comandos sqlplus.exe y sqlplusw.exe.

CUENTAS DE USUARIO

Es habitual que en el proceso de instalación de una bbdd se generen multitud de cuentas de usuario. Todas las que no sean necesarias deberán ser eliminadas o deshabilitadas.

Se crearán cuentas específicas de aplicación con funciones separadas (y en consecuencia con diferentes permisos), que no podrán ser utilizadas por usuarios interactivos.

Se seguirá siempre el principio de privilegios mínimos.

En una bbdd en producción no pueden existir cuentas empleadas para el desarrollo de la aplicación.

Cualquier cambio en cuanto a creación/eliminación de cuentas, o modificación de permisos de las mismas, ha de ser notificado y documentado convenientemente.

CUENTAS DE USUARIO.

Durante el proceso de creación de la base de datos, además de los usuarios SYS y SYSTEM se crean multitud de cuentas de usuario predeterminadas, cuyos nombres y contraseñas son conocidos.

309. Todas las cuentas que no sean estrictamente necesarias deben ser eliminadas o al menos estar deshabilitadas y/o expiradas. Para todas las cuentas que se dejen hábiles, hay que cambiar la contraseña que traen por defecto. Como el listado de cuentas predeterminadas varía de unas versiones a otras de Oracle, lo más práctico es obtenerla mediante la siguiente sentencia:

```
SELECT USERNAME, ACCOUNT_STATUS FROM DBA_USERS;
```

310. Adicionalmente, y dado que en el objeto de este trabajo se pretende dar servicio a una aplicación web, se crearán cuentas específicas (cuentas de aplicación) para ser empleadas única y exclusivamente por dicha aplicación. Esto posibilita el control total sobre los permisos con los que contarán cada una de estas cuentas, y mejorar significativamente la precisión y utilidad de los log de transacciones y posteriores procesos de auditoría.

Una cuenta de aplicación, es una cuenta de usuario especializada que en ningún caso estará disponible para los usuarios interactivos, y que será utilizada exclusivamente por una aplicación, un servicio, o un proceso batch local. Hay que tener especial cuidado en que estas cuentas no tengan privilegios elevados (administración) sino únicamente los necesarios para poder ejecutar las operaciones asociadas a la funcionalidad de la aplicación.

Creo importante insistir en la necesidad de crear varias cuentas de aplicación diferentes, una por cada función o tarea, en lugar de usar una sola por dos razones. En primer lugar para que cada una de las tareas de ejecute con los privilegios mínimos imprescindibles, y en segundo lugar para mejorar la trazabilidad de las operaciones y poder afinar al máximo los procesos de auditoría, investigación de incidentes y análisis forense.

En general hay que seguir el principio de privilegios mínimos, de manera que ningún usuario tenga ni un solo permiso que no necesite. Como ya se ha explicado, esta es una de las razones por la que se recomienda crear varias cuentas para la aplicación. Cada una de ella se utilizará específicamente para ejecutar una tarea concreta, y en consecuencia tendrá un conjunto de permisos limitado estrictamente a los requisitos funcionales de esa tarea.

311. Por supuesto, en una base de datos en producción no debe existir ninguna cuenta de usuario de desarrollo, puesto que dichas cuentas tienen un conjunto de privilegios muy superior a los que necesitan las cuentas de aplicación.

312. Las cuentas proxy proporcionan un mecanismo para que los usuarios se pueden conectar a la base de datos a través de otros usuarios. A no ser que sea absolutamente necesario, se deshabilitarán todas las cuentas proxy.

313. Finalmente, cualquier alteración en los privilegios de una cuenta o creación de cuentas nuevas, debe ser informada y quedar convenientemente documentada.

ACTUALIZACIONES DE ORACLE

El DBA deberá estar al corriente de las nuevas vulnerabilidades, y aplicar las actualizaciones y parches de seguridad que el fabricante del software vaya liberando, previa comprobación de que dichas actualizaciones no impiden la continuidad del negocio.

ACTUALIZACIONES DE ORACLE

314. Una de las responsabilidades del administrador de la base de datos será estar al tanto de las vulnerabilidades que se vayan publicando para actuar en consecuencia. Una de las acciones a tomar es instalar las actualizaciones y los parches de seguridad que vaya publicando el fabricante del software, después de comprobar en el entorno de preproducción que dichas actualizaciones no implican perdida de la funcionalidad necesaria para la aplicación web.

CONFIGURACIÓN DE PARÁMETROS

Los archivos de configuración y los recursos a los que apuntan, deben ser accesible solo con la cuenta del sistema operativo del host creada expresamente para ello.

Hay que cerrar todos los puertos que no den servicio a los requisitos funcionales de la aplicación.

Hay que activar la auditoría de los usuarios administradores o que tengan un nivel alto de privilegios.

El servicio de base de datos que recibe las conexiones desde los clientes debe estar protegido, modificando el número de puerto que emplea por defecto. Si es posible, se le indicará las direcciones IP de los equipos cuyas conexiones deba aceptar en lugar de emplear nombres de host. De esta forma se evitarán los ataque basados en DNS.

CONFIGURACIÓN DE PARÁMETROS [2]

315. Los archivos de configuración ***INIT.ORA***, ***SQLNET.ORA*** y ***LISTENER.ORA***, deben ser accesibles únicamente por el usuario de instalación de Oracle y por el

usuario con el que corre el servidor de base de datos tal y como se indicó en el apartado de pre-requisitos. Así mismo, los archivos, directorios y demás recursos a los que apuntan los diferentes parámetros contenidos en dichos ficheros, también deben ser accesibles solo para los mismos usuarios.

Para incrementar el nivel de seguridad, es posible establecer un valor específico para todos y cada uno de los parámetros de configuración definidos en los tres archivos anteriores (guías CCN-STIC), pero en la práctica, esto hace que la configuración sea tan sumamente restrictiva, que hace imposible que la base de datos cumpla con los requisitos funcionales de la aplicación que ha de atacarla. Por esta razón, creo que es preferible dejar el valor por defecto de dichos parámetros.

316. Hay algunos parámetros de configuración presentes en el archivo ***SPFILES.ORA***, que deben ser objeto de especial atención. Primero se eliminará la línea `DISPATCHERS = '(PROTOCOL=TCP) (SERVICE=[SID]XDB)'` para desactivar los puertos abiertos por defecto 2100 (FTP) y 8080 (HTTP) de los servicios de Oracle XML DB.

317. Además hay que auditar las operaciones de los usuarios ***SYSDBA*** y ***SYSOPER*** mediante la inicialización del parámetro `AUDIT_SYS_OPERATIONS = TRUE`.

SEGURIDAD DEL SERVICIO LISTENER

Este servicio es el encargado de recibir las peticiones de los clientes remotos a la base de datos. En el archivo ***LISTENER.ORA*** se almacena la configuración del mismo. En Windows la configuración puede modificarse utilizando la herramienta Oracle Net Manager (ONM).

La configuración correcta del LISTENER permitirá evitar un gran número de problemas asociados al mal uso de las conexiones remotas.

318. Es preferible utilizar direcciones IP en lugar de nombres de host para hacer más complicados los ataques basados en DNS. Tal y como se indicó en el punto 1, al crear el servicio LISTENER hay que asignarle un nombre distinto del nombre por defecto, y además se debe cambiar el puerto por defecto (1521), para dificultar los ataques automatizados.

319. Para incrementar la seguridad se puede crear una contraseña segura para la gestión de la configuración del LISTENER. Además, hay que configurar el parámetro `LOCAL_OS_AUTHENTICATION = ON`, para que el archivo LISTENER.ORA solo pueda ser modificado por el usuario de sistema operativo bajo el que se ejecuta el servicio.

COPIA DE SEGURIDAD Y RECUPERACIÓN ANTE DESASTRES

Para garantizar la disponibilidad, se necesitan copias no solo de los datos sino también de los registros de transacciones y de los archivos de control y configuración de una base de datos. La política de backup debe tener en cuenta la totalidad de la base de datos y no solo los datos de usuario.

Las copias de seguridad se deben almacenar observando las mismas precauciones que los datos en producción para evitar que se pueda acceder a ellas de forma no autorizada. Hay que preservar la confidencialidad y la integridad de la información, por lo que se debe tomar en consideración la opción de cifrar los backup.

COPIA DE SEGURIDAD Y RECUPERACIÓN ANTE DESASTRES

Los procedimientos de backup y recovery se salen del ámbito de esta relación de buenas prácticas de seguridad, pero sí que es necesario hacer algunas consideraciones al respecto.

Es imperativo desde el punto de vista de la seguridad que la base de datos de configure en modo ***ARCHIVELOG***. De esta manera, los registros de transacciones se almacenarán por duplicado aumentando significativamente la disponibilidad de la base de datos y su capacidad de recuperación ante desastres.

320. Los archivos de ***redo archivados***, deben ser almacenados en un dispositivo diferente a los ***redo online***, para evitar que un fallo del medio físico de almacenamiento suponga la pérdida de ambos juegos de ficheros.

Si no se indicó el modo ARCHIVELOG durante el proceso de instalación, hay que ejecutar el comando `ALTER DATABASE ARCHIVELOG`.

321. De la misma manera para garantizar la disponibilidad de la información es importante configurar la réplica y archivado de los ficheros de control. Los archivos de control y sus copias deben almacenarse también en diferentes ubicaciones físicas.

322. Las copias de seguridad de cualquier tipo han de ser preservadas convenientemente para poder garantizar la capacidad de recuperación ante desastres y por tanto la continuidad del negocio, y la integridad de la información. Hay que valorar si los backup han de guardarse cifrados.

ASIGNACIÓN DE RECURSOS Y POLÍTICA DE CONTRASEÑAS

Cada SGBD proporciona sus propios mecanismos para limitar de alguna manera el consumo de recursos por parte de las diferentes cuentas de usuario, como pueden ser el tamaño máximo del área de memoria asignada o el número de conexiones simultaneas que se pueden establecer con una determinada cuenta. Estas cuestiones deben ser analizadas, pues podrían posibilitar un ataque de denegación de servicio.

Por otro lado, hay que establecer una política de contraseñas que impida un número ilimitado de intentos de conexión (ataques de fuerza bruta), y que fuerce la complejidad de las mismas.

PERFILES DE ORACLE

Los perfiles de Oracle son un conjunto de parámetros relacionados con los recursos de máquina (tiempo máximo de CPU por conexión, sesiones por usuario, tiempo de conexión, etc.), que se configurarán en función de los requisitos funcionales de cada aplicación y con cuestiones de seguridad relativas a la política de contraseñas. Por defecto, en una base de datos Oracle existe el perfil ***DEFAULT***, aunque se pueden crear todos los perfiles que se consideren oportunos. Al definir una nueva cuenta de usuario siempre hay que indicarle su perfil, que por omisión será DEFAULT.

323. En relación a las cuentas de usuario y sus contraseñas hay que configurar los siguientes parámetros: `FAILED_LOGIN_ATTEMPTS` define el máximo número de intentos de inicio de sesión antes de que se bloquee la cuenta. `PASSWORD_LIFE_TIME` indica el número máximo de días de validez para una

contraseña. `PASSWORD_REUSE_MAX` especifica el número de cambios de contraseña necesarios antes de que se pueda reutilizar una contraseña, `PASSWORD_REUSE_TIME` hace lo propio pero en número de días. `PASSWORD_LOCK_TIME` especifica el número de días que permanecerá bloqueada una cuenta al superar el número máximo de intentos incorrectos de validación. El parámetro `PASSWORD_GRACE_TIME` indica el número de días tras haber caducado la contraseña durante los cuales todavía se permite el inicio de sesión para cambiarla por una nueva. Por defecto en Oracle no se comprueba la complejidad de las contraseñas cuando éstas se cambian. Para ello, se asignará el parámetro `PASSWORD_VERIFICATION_FUNCTION` a una función de verificación.

Oracle proporciona el script `[ORACLE_HOME]\RDBMS\ADMIN\UTLPWDMG.SQL`, aunque sus parámetros son modificables o incluso se puede crear un script nuevo.

LAS ESTRUCTURAS LÓGICAS DE LA BASE DE DATOS

Es fundamental mantener separados los espacios de trabajo para los datos del catálogo (diccionario de datos), y para los datos de usuario. Siempre que el SGBD lo permita, hay que configurar la bb.dd. para que los metadatos y los datos de usuario se almacenen por separado.

Además, hay que asegurarse (salvo que se especifique lo contrario en los requisitos funcionales), que las cuentas de aplicación no tengan cuota se uso en los espacios de almacenamiento, es decir, que no puedan crear objetos en la base de datos.

TABLESPACES [2]

Los espacios de tablas son estructuras lógicas que emplea Oracle para agrupar los diferentes objetos de usuario que se van definiendo en una base de datos (tablas, vistas, índices, procedimientos almacenados, triggers, etc.). Cada usuario de la base de datos tiene asignado un tablespace por defecto.

324. Es fundamental que salvo el usuario SYS, ningún otro tenga a ***SYSTEM*** como tablespace por defecto. Se deben crear espacios de tablas con sus ***datafiles***

correspondientes, para almacenar los objetos de usuario y dejar SYSTEM exclusivamente para el diccionario de datos.

325. También hay que asegurarse que las cuentas de usuario de aplicación no tengan cuota ***(QUOTA 0)*** en los tablespaces. De esta manera no podrán nunca crear objetos en la base de datos, aunque obtuvieran los permisos necesarios.

OBJETOS DE DICCIONARIO

Salvo los usuarios administradores de la base de datos, todos los demás deberían tener un acceso muy restringido a dichos objetos, especialmente a las tablas y vistas donde se puede obtener información de auditoría.

Hay que asegurarse de que al eliminar usuarios, se eliminen también sus roles o grupos de permisos asociados.

OBJETOS DE DICCIONARIO

Para aumentar el nivel de seguridad, hay que revisar el nivel de acceso de los diferentes usuarios a los objetos del diccionario (TABLAS, VISTAS, ROLES, SINÓNIMOS, PRIVILEGIOS, PAQUETES, CUENTAS PROXY). En principio, salvo los usuarios administradores de la base de datos, todos los demás deberían tener un acceso muy restringido a dichos objetos, especialmente a las tablas y vistas donde se puede obtener información de auditoría.

326. En particular se debe revocar el acceso a las tablas y vistas `SYS.USER_HISTORY$`, `SYS.LINK$` y `SYS.USER$` para todos los usuarios y roles excepto para las cuentas SYS y DBA para evitar la obtención de información de nombres de usuario y hashes de contraseñas. Lo mismo hay que hacer con `SYS.SOURCE$` para evitar la obtención de información sobre el código de los procedimientos y con `SYS.AUD$`, y las que comienzan por `X$` y por `V$` para evitar que se acceda a los registros de auditoría y a las estadísticas internas.

En cuanto a los roles, las cuentas de usuario y de aplicación no deben tener asignado ningún rol que les proporcione acceso a los recursos del catálogo, salvo

que expresamente lo necesiten. Hay que comprobar que no tengan asignado ningún rol de tipo `_CATALOG_`, ni los roles `RESOURCE` o `DBA`.

327. Por último, cuando se elimina un usuario, hay que verificar que se eliminan también los roles creados por él.

CÓDIGO EJECUTABLE

Para cualquier procedimiento o bloque de código que se almacene sin cifrar, hay que verificar que no contiene credenciales de usuario u otra información crítica que pueda ser aprovechada por un atacante.

Además, hay que calcular códigos de integridad (hashes) de dichos bloques y almacenarlos cuidadosamente, para comprobar periódicamente la integridad de los mismos. Un objetivo de un posible atacante bien podría ser alterar el código de procedimientos y funciones almacenadas.

PAQUETES Y BLOQUES DE PL/SQL [3]

328. En cuanto a los paquetes, se eliminarán todos los que no sean necesarios, pues en el proceso de instalación de la base de datos se crean varios. Los paquetes que se dejen disponibles por ser necesarios para realizar tareas administrativas, deben estar accesibles solo para los usuarios administradores.

329. En los bloques de PL/SQL (procedimientos almacenados, funciones almacenadas y triggers) que se guarden sin cifrar, no se deben incluir nombres de usuario ni contraseñas ni otras informaciones críticas.

330. Además, hay que calcular periódicamente códigos de control de integridad (hash) de los bloques PL/SQL y almacenar los resultados. De esta manera se puede comprobar que no se han producido alteraciones.

CIFRADO, INTEGRIDAD Y CREDENCIALES

Los datos restringidos deben ser cifrados en la base de datos para evitar su consulta y/o modificación (garantizar integridad y confidencialidad) por parte de usuarios no autorizados.

Si se utilizan tablas para almacenar claves de cifrado, hay que ser muy cuidadosos a la hora de otorgar permisos sobre dicha tabla. Además la estructura y especialmente los nombres de la tabla y sus campos no deben dar ninguna pista a cerca de su contenido.

Hay que valorar el probable empeoramiento en los tiempos de acceso a datos cifrados por si pudiera constituir un problema para la funcionalidad de la aplicación.

CIFRADO, INTEGRIDAD Y CREDENCIALES

331. Para complementar las políticas de seguridad, los datos restringidos deben ser cifrados en la base de datos para evitar su consulta y/o modificación (integridad y confidencialidad) por parte de usuarios no autorizados. Utilice campos de tipo RAW o BLOB como elementos de almacenamiento de información cifrada, se recomienda el uso del paquete `DBMS_CRYPTO` ya que proporciona multitud de funciones de cifrado, utilizar preferentemente los algoritmos 3DES y AES, con clave de tamaño mínimo de 128 bits.

332. Si se utilizan tablas como medio de almacenamiento de claves de cifrado, estas tablas deberán ser protegidas mediante el uso de varios niveles de seguridad:

    a) Debe restringirse el acceso a la tabla y activarse la auditoría de bajo nivel (FGA).
    b) Deben evitarse nombres de columna evidentes (user, password, contraseña, user_id y cosas por el estilo).
    c) Las claves cumplirán, como mínimo, con las políticas de complejidad de contraseña de los usuarios.
    d) Los valores de contraseñas deben ser cifrados a su vez mediante una clave maestra.
    e) Los procedimientos de uso y acceso de las claves deberán ser ofuscados.

Hay que tener en cuenta que el acceso a columnas cifradas puede penalizar el rendimiento de la aplicación por lo que hay que ponderar el uso que se va a hacer de esta posibilidad.

CONCIENCIACIÓN Y FORMACIÓN

Es imperativo mantener un proceso constante de concienciación de todos los implicados de alguna manera en la seguridad del sistema, y de formación y autoformación de los responsables de gestionar la misma.

CONCIENCIACIÓN Y FORMACIÓN

333. Es imperativo mantener un proceso constante de concienciación de todos los implicados de alguna manera en la seguridad del sistema, y de formación y autoformación de los responsables de gestionar la misma.

Los responsables de la seguridad del sistema deben conocer en detalle el funcionamiento del mismo y mantener una actitud proactiva. Así mismo, la organización debe proporcionar los instrumentos adecuados en forma de recursos de formación.

Hay consultoras especializadas en la formación en seguridad informática y en realizar auditorías externas, que no solo deben servir para comprobar el buen funcionamiento del Control Interno Informático, sino también para enriquecer la experiencia de los responsables del sistema.

Oracle en particular, tiene un buen catálogo de cursos y de herramientas que sirven para aumentar la seguridad en los servidores de bases de datos. Sirva por ejemplo Real Application Cluster (RAC), que es una potente utilidad pensada para sistemas que necesiten de una muy alta disponibilidad, y toda su formación oficial asociada.

Puesto que todas las medidas de seguridad aquí expuestas tienen como finalidad proporcionar seguridad ante las principales amenazas conocidas, en el siguiente cuadro resumen se muestra la correlación entre las mismas y dichas medidas:

GRUPO DE MEDIDAS	Amenazas según:	
	IMPERVA 2013	Application Security 2012
1. PROCESO DE INSTALACIÓN DEL GESTOR ORACLE Y DE CREACIÓN DE BASES DE DATOS	4,7, 9	DB1,DB4, DB5,DB9, DB8
2. CUENTAS DE USUARIO	1,2, 5,8	DB1,DB3, DB7
3. CONFIGURACIÓN DE PARÁMETROS	4,6, 7	DB6
4. SEGURIDAD DEL SERVICIO LISTENER	4,6, 7,9	DB6,DB8
5. COPIA DE SEGURIDAD Y RECUPERACIÓN ANTE DESASTRES	6	DB5
6. PERFILES DE ORACLE	2,7	DB1
7. TABLESPACES	2,8	DB!,DB2 DB3,DB7
8. OBJETOS DE DICCIONARIO	3,5	DB1,DB2, DB3
9. PAQUETES Y BLOQUES DE PL/SQL	3,8	DB1,DB2, DB3,DB4, DB7
10. CIFRADO, INTEGRIDAD Y CREDENCIALES	3,8	DB1,DB7, DB10
11. CONCIENCIACIÓN Y FORMACIÓN	10	DB1,DB5

Amenazas vs Medidas 1

2x043 Evaluación

Tal y como se estableció en el capítulo 3, uno de los objetivos de este trabajo es definir una metodología para llevar a cabo una auditoría de seguridad con la que se verifique si un servidor de bases de datos está configurado de acuerdo a lo establecido en el punto anterior.

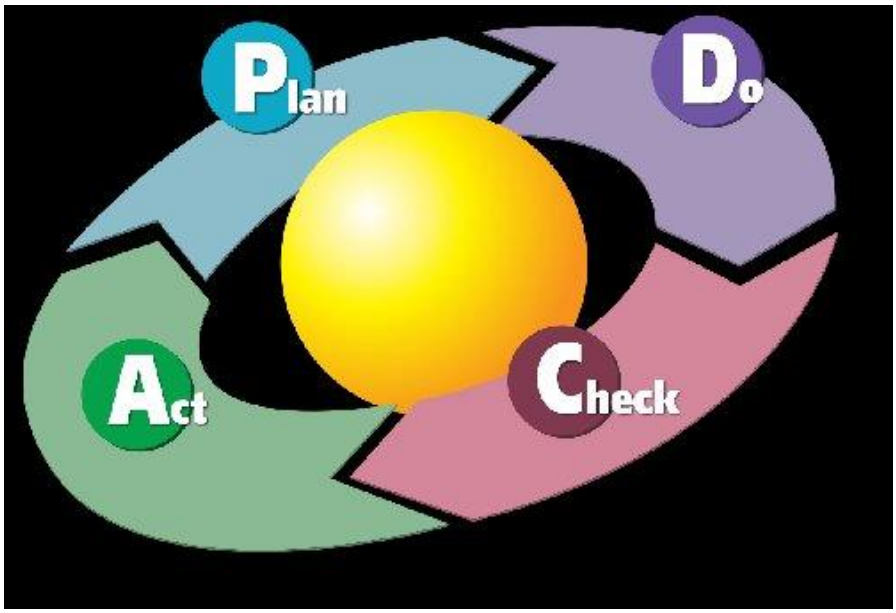
Para poder hacer efectiva esta verificación, la herramienta elegida es el empleo de una **lista de comprobación**. Esta lista de comprobación está recogida en el Anexo B, y es un elemento fundamental del proceso pues permite detectar las no conformidades, es decir, discrepancias entre los controles de seguridad definidos en la descripción de la metodología para la configuración segura del servidor, y la configuración real del sistema.

El servidor bastionado, debe ser sometido a la lista de comprobación de manera periódica para poder afirmar que su configuración es segura. Dicha periodicidad se definirá en el marco de la política de seguridad TIC de la organización.

No obstante, partimos de la base de que la **seguridad es un proceso**. Así pues, la herramienta de verificación elegida (lista de comprobación en este caso) debe ser sometida también a revisiones periódicas, así como el propio proceso de bastionado. Ciertos controles de seguridad, pueden ser perfectamente válidos en el momento de establecerlos, pero debido a la constante evolución de las tecnologías y de las amenazas, existe el riesgo cierto de quedar superados por la realidad en poco tiempo.

Por tanto, si se busca que la seguridad en cualquier sistema de información sea efectiva, no podemos pretender que los requisitos y controles seguridad permanezcan inmutables en el tiempo. En el proceso de la seguridad, como en cualquier otro, para alcanzar un cierto nivel de calidad, es imprescindible la mejora continua. Esta idea queda perfectamente descrita en el **Ciclo de Deming**, y debido a su amplísima aceptación, la tomare aquí como base.

El Ciclo de Deming, también conocido como **PDCA** (de Edwards Deming), es una estrategia de mejora continua de la calidad en cuatro pasos. Es muy utilizado por los Sistemas de Gestión de Calidad (SGC). Las siglas, **PDCA** son el acrónimo de **P**lan, **D**o, **C**heck, **A**ct (Planificar, Hacer, Verificar, Actuar).



Ciclo de Deming 1

Para reforzar el proceso de auditoría, además del empleo periódico de las listas de comprobación, es muy recomendable plantearse la seguridad desde el punto de vista del atacante. Una herramienta valiosísima es contar con un experto es seguridad o hacker, que de manera extraordinaria simule un ataque contra un determinado sistema de información con la intención de detectar vulnerabilidades y plantear propuestas para mitigarlas. El empleo en este sentido de las habilidades de los hackers se denomina **hacking ético**[17], y a este tipo de auditorías de seguridad **pentesting**[18].

Existen herramientas software que permiten automatizar en cierta medida el proceso de auditoría de las bases de datos. Sirvan los siguientes ejemplos extraídos de una interesante publicación de Gartner Inc [4].

- *La suite **DbProtect** de **Application Security** combina el producto de monitoreo de la compañía con su herramienta de gestión de derechos de usuario y su producto de escaneo de vulnerabilidad líder en el Mercado. Application Security tapó un hueco de funcionalidad en 2011 añadiendo prevención a la suite. La compañía tiene una base fuertemente instalada por el lado de la vulnerabilidad y la ha usado en ventas de monitoreo.*

- ***BeyondTrust** adquirió las acciones de Lumigent, uno de los primeros proveedores de DAM y de los primeros líderes en el mercado. La oferta de Lumigent encaja bien con el portafolio de BeyondTrust, pero el desarrollo limitado del viejo producto de*

Lumigent ha resultado en algunos huecos de funcionalidad en su oferta. BeyondTrust ha tapado algunos de estos huecos con su último producto y se ha comprometido a abordar otros en futuros productos.

- ***IBM InfoSphere Guardium*** *es el líder en el mercado en términos de ganancias y número de clientes. Su oferta tiene la cobertura de plataforma más amplia y el conjunto más robusto de características, y la compañía ha demostrado la capacidad para usar el modelo de ventas de IBM con su oferta de DAP.*

- ***Imperva*** *es segundo en el mercado de DAP, con alcances y ofertas similares a las de IBM. La compañía vende DAP como parte de una suite de seguridad de datos que incluye un firewall de red y un producto para seguridad de datos no estructurados.*

- ***McAfee*** *adquirió el producto DAM de Sentrigo y está empujando de manera importante en el mercado de seguridad de las bases de datos. Si McAfee puede utilizar su gran base de clientes y fuerza de ventas, entonces puede ser un fuerte competidor en este mercado.*

- ***Oracle*** *tiene una fuerte presencia en el mercado, especialmente entre su base de clientes. Oracle provee mucha funcionalidad DAP con dos productos separados:* ***Audit Vault y Database Firewall****, que proveen soporte para sistemas de gestión de base de datos Oracle y no Oracle. Oracle ofrece varias herramientas de distintas marcas ques e corresponden con capacidades DAP extendidas, algunas de las cuales son multiplataforma y algunas de las cuales sólo tienen compatibilidad para RDBMSs de Oracle.*

- ***WareValley****, un proveedor surcoreano con fuerte enfoque y presencia en el mercado de Asia/Pacífico, tiene una oferta que consiste en encriptación, monitoreo, gestión de vulnerabilidad y gestión de RDBMS. La compañía ha mostrado recientemente un deseo creciente por moverse a los mercados de Norteamérica y la Unión Europea con precios agresivos.*

En cualquier caso, el resultado de una auditoría, debe ser la elaboración de un informe que permita a los responsables de la seguridad TIC de una organización tomar las acciones correctivas necesarias, contando siempre con el compromiso de la alta dirección.

Para dicho informe, se propone el siguiente formato:

- **Antecedentes**: reflejan brevemente lo que se ha auditado. Sirve para situar a la persona que lo recibe como están las cosas.
- **Alcance**: aquí se redacta el alcance real, es decir, lo que realmente hemos hecho. Hay que identificar a todos los implicados de alguna, manera dado su diferente nivel de interés en el proceso (Stakeholders[19]). También hay que planificar las diferentes reuniones y acciones concretas a llevar a cabo.
- **Resumen de la Auditoria**: se resumen las etapas de la auditoría realizada y de los resultados obtenidos
- **Conclusiones**: son un resumen de comentarios (de los que pone el auditor). En esta fase se discute el informe con las personas auditadas. Es por ello que siempre habrá un informe preliminar donde el/los auditores expresen sus conclusiones, y un informe definitivo que verá la luz después de ser debatido con los implicados. En el informe definitivo hay que incorporar los desacuerdos o justificaciones que surjan en dicho debate.
- **Anexo y Comentarios**: se ponen los comentarios habitualmente en un anexo. Es decir, las conclusiones presentan de forma ejecutiva el resultado de la auditoría y en el anexo se detallan los comentarios que sustentan dichas conclusiones. Además, se debe incluir también como anexo la auditoría técnica informática, que puede ser perfectamente el resultado de aplicar un lista de comprobación (checklist), tal y como se propone en el presente trabajo.

Informe 1

LISTA DE COMPROBACIÓN (CHECKLIST).

CONTROL DE SEGURIDAD	CONFORMIDAD	OBSERVACIONES
PREREQUISITOS	☐ CONFORME ☐ NO CONFORME	
201. Compatibilidad del software	☐ CONFORME ☐ NO CONFORME	
202. Actualizaciones del sistema operativo	☐ CONFORME ☐ NO CONFORME	
203. Componentes necesarios del sistema operativo	☐ CONFORME ☐ NO CONFORME	
204. Antivirus	☐ CONFORME ☐ NO CONFORME	
205. Cuentas de usuario de host	☐ CONFORME ☐ NO CONFORME	
206. Permisos sobre ficheros	☐ CONFORME ☐ NO CONFORME	
207. Permisos sobre el registro	☐ CONFORME ☐ NO CONFORME	
208. Restricciones de acceso al servidor	☐ CONFORME ☐ NO CONFORME	
209. Cifrado de datos en movimiento	☐ CONFORME ☐ NO CONFORME	
210. Seguridad perimetral de la red	☐ CONFORME ☐ NO CONFORME	
SEGURIDAD EN EL SERVIDOR DE BASES DE DATOS	☐ CONFORME ☐ NO CONFORME	
301. Nombre y ruta de instalación	☐ CONFORME ☐ NO CONFORME	
302. Componentes a instalar	☐ CONFORME ☐ NO CONFORME	
303. Creación de LISTENER	☐ CONFORME ☐ NO CONFORME	
304. Número de puerto de LISTENER	☐ CONFORME ☐ NO CONFORME	
305. Nombre de la base de datos	☐ CONFORME ☐ NO CONFORME	
306. Modo ARCHIVELOG	☐ CONFORME ☐ NO CONFORME	
307. Contraseñas de SYS y SYSTEM	☐ CONFORME ☐ NO CONFORME	
308. Eliminación de archivos de instalación	☐ CONFORME ☐ NO CONFORME	
309. Cuentas innecesarias	☐ CONFORME ☐ NO CONFORME	
310. Cuentas de aplicación	☐ CONFORME ☐ NO CONFORME	
311. Cuentas de desarrollo	☐ CONFORME ☐ NO CONFORME	
312. Cuentas proxy	☐ CONFORME ☐ NO CONFORME	
313. Control de cambios	☐ CONFORME ☐ NO CONFORME	
314. Actualizaciones	☐ CONFORME ☐ NO CONFORME	
315. Permisos sobre archivos de configuración	☐ CONFORME ☐ NO CONFORME	

CONTROL DE SEGURIDAD	CONFORMIDAD	OBSERVACIONES
316. Desactivación de puertos por defecto	☐ CONFORME ☐ NO CONFORME	
317. Auditoría de SYSDBA y SYSOPER	☐ CONFORME ☐ NO CONFORME	
318. IPs vs nombres de host	☐ CONFORME ☐ NO CONFORME	
319. Seguridad en la configuración del LISTENER	☐ CONFORME ☐ NO CONFORME	
320. Almacenamiento de los registros de transacciones	☐ CONFORME ☐ NO CONFORME	
321. Almacenamiento de los archivos de control	☐ CONFORME ☐ NO CONFORME	
322. Seguridad de las copias de respaldo	☐ CONFORME ☐ NO CONFORME	
323. Configuración de perfiles. Política de contraseñas	☐ CONFORME ☐ NO CONFORME	
324. Preservación del tablespace SYSTEM	☐ CONFORME ☐ NO CONFORME	
325. Cuotas en tablespaces	☐ CONFORME ☐ NO CONFORME	
326. Restricción de acceso al diccionario de datos	☐ CONFORME ☐ NO CONFORME	
327. Verificación de eliminación de roles	☐ CONFORME ☐ NO CONFORME	
328. Eliminación de paquetes innecesarios	☐ CONFORME ☐ NO CONFORME	
329. No inclusión de información crítica en procedimientos sin cifrar	☐ CONFORME ☐ NO CONFORME	
330. Control de integridad de código	☐ CONFORME ☐ NO CONFORME	
331. Cifrado de datos restringidos	☐ CONFORME ☐ NO CONFORME	
332. Almacenamiento de claves de cifrado en tablas	☐ CONFORME ☐ NO CONFORME	
333. Concienciación y formación	☐ CONFORME ☐ NO CONFORME	

Checklist 1

TEST DE PENETRACIÓN (PENTEST).

Tal y como se dijo anteriormente, otra forma de auditar la seguridad de un sistema consiste en realizar un test de intrusión [5] en el que el auditor asume el rol de atacante para tratar de aprovechar alguna posible vulnerabilidad que le permita acceder a información de manera ilegítima, modificar datos, o dejar al sistema fuera de servicio. Este tipo de auditoría técnica informática reviste una mayor complejidad que el checklist y necesita de personal con un nivel superior de conocimientos.

La realización de este tipo de test se puede llevar a cabo cada cierto tiempo con el fin de evaluar la validad de las medidas de seguridad implementadas en el sistema. De su resultado se pueden extraer importantes conclusiones que aconsejen modificar o aumentar la configuración de seguridad en determinados aspectos.

De cualquier manera, este tipo de pruebas siempre se deben planificar de manera que perjudique lo menos posible a la continuidad del servicio y por tanto a los usuarios legítimos. Tienen que contar además con el visto bueno de la alta dirección.

Todas estas pruebas se realizan de forma metódica, siguiendo detalladamente criterios implementados en las siguientes fases:

- Planeamiento y recolección de información del objetivo a evaluar.
- Realización de las prueba técnicas.
- Identificación y clasificación de las vulnerabilidades encontradas.
- Presentación de evidencias.
- Recomendaciones de tratamiento de vulnerabilidades y aseguramiento (Hardening) del sistema evaluado.

Existen muchos recursos y enlaces interesantes a este respecto [j], [k], [l], [m], [n].

2x050 Capitulo 5. Contribución

La principal contribución del presente trabajo es proporcionar una guía práctica y sencilla de aplicar para los administradores de bases de datos y supervisores de seguridad.

Basándome en buenas prácticas ya publicadas, multitud de libros sobre la cuestión de la seguridad en bases de datos, guías tan restrictivas como las del Centro Criptológico Nacional, y mi propia experiencia profesional, he pretendido sintetizar de manera sistemática, todos los elementos clave para instalar y configurar un servidor de bases de datos con ciertas garantías desde el punto de vista de la seguridad, sin impedir por ello que dicho servidor proporcione las funcionalidades necesarias para el buen funcionamiento de las aplicaciones que a él se conectarán. Uno de estos elementos es sin duda la realización periódica de auditorías, que permitan verificar el estado del servidor en cuanto a medidas de seguridad.

Esta guía es, además, aplicable a casi cualquier supuesto de empleo solo con hacer pequeñas modificaciones en los diferentes controles, y permite encontrar el equilibrio necesario entre servicio y seguridad, tanto en aplicaciones de uso general, como en aquellos sistemas que requieran acreditación por almacenar y gestionar información clasificada.

En definitiva, creo que la principal contribución ha sido el aglutinar y resumir diferentes aproximaciones al problema de la seguridad en BBDD, obteniendo como resultado un conjunto de recomendaciones simplificado y de fácil aplicación y comprobación. Todo ello, sin imponer una medidas excesivas que tal y como ya se ha mencionado, en la mayoría de las ocasiones provocan que finalmente el servidor de la base de datos no pueda proporcionar las funcionalidades que los requisitos de negocio habían previsto, y por tanto la solución resulte inviable. Se ha logrado, en mi opinión, el equilibrio necesario entre seguridad y funcionalidad.

2x060 Capitulo 6. Conclusiones

Después de un exhaustivo trabajo de recopilación de información, lectura, estudio y análisis de los resultados de diversos trabajos, además de mi propia experiencia con el uso y administración de servidores de bases de datos, puedo constatar fehacientemente que las amenazas sobre la seguridad de los sistemas de de bases de datos no solo no se han estabilizado sino que siguen su escalada.

Las causas son múltiples pero cabe destacar dos de ellas:

- Cada vez existen herramientas en mayor cantidad y calidad, que pueden llevar a cabo de manera automatizada todo tipo de ataques. Esto implica, que los atacantes no tienen que ser necesariamente gente con una gran experiencia y conocimientos. Digamos que la posibilidad de efectuar ataques está al alcance de más gente.

- La proliferación de servicios de almacenamiento en la nube, aunque se ha avanzado mucho en su seguridad, siguen siendo un reto en este ámbito. De hecho, hasta que no se esté en disposición de ofrecer mayores garantías, no llegará el boom definitivo de este tipo de tecnologías. Esta cuesto es precisamente, el objeto de futuros trabajos, tal y como propongo en el siguiente capítulo.

Por otro lado, también es cierto que la gran experiencia que se ha alcanzado en estas cuestiones de seguridad, y las excelentes soluciones que proponen muchos fabricantes de software, han alcanzado un nivel de madurez extraordinario, que permite contrarrestar las amenazas.

No obstante, la única manera de estar en disposición de dar ciertas garantías de seguridad a cualquier organización, consiste en seguir una metodología y un proceso de mejora continua.

Tal y como se ha hecho en el presente trabajo, en primer lugar hay que analizar el contexto, y llevar a cabo un análisis de riesgos. En base a ese análisis de riesgos hay que definir las medidas de seguridad concretas para poder mitigarlos, y por último realizar los controles y auditorías necesarios para comprobar la correcta implementación de dichos controles y para validar su efectividad a la hora de mantener seguro un sistema determinado.

2x070 Capitulo 7. Trabajo futuro

Una tendencia cada vez más en boga en el mundo TIC, consiste en externalizar los servicios de almacenamiento de información y su disponibilidad a cualquier hora y desde cualquier sitio. Es lo que se conoce como almacenamiento en la nube o ***Cloud Storage***.

Este tipo de soluciones proporcionan grandes ventajas a organizaciones y particulares, pero implican también grandes riesgos en cuanto a pérdida del control sobre las infraestructuras sobre las que se almacenan los datos, y sobre la seguridad de los mismos. El crecimiento desmesurado de este tipo de almacenamientos ha puesto en el límite de la seguridad a numerosos servicios de gran importancia.

La naturaleza "virtual" de la nube elimina muchos de los flujos físicos de trabajo y puntos de control basados en perímetros para contener la información confidencial. Por lo tanto, es esencial que las plataformas de seguridad en la nube tomen un enfoque centrado en los datos.

Hay dos aspectos cruciales en cuanto a la seguridad cuando hablamos de almacenamientos en la nube que son la autenticación y el cifrado de la información.

Una autenticación robusta es incluso más determinante que en el caso de que los sistemas de información estén bajo nuestro control. Por ello, cada vez se tiende más hacia la autenticación multifactor en estos entornos.

Se denomina autenticación multifactor a todo procedimiento de autenticación en el que se utilizan dos o más factores de autenticación para probar la identidad de la entidad (usuario, aplicación, etc.) involucrada. En general, es posible obtener las máximas garantías sobre la identidad de una entidad verificando satisfactoriamente dos factores de autenticación de diferente tipo. Este caso particular se conoce como *Two Factor Authentication* o 2FA y puede conseguirse, por ejemplo, con una clave privada dentro de un *token* hardware (algo que se tiene) cuya activación precisa de la entrada de un PIN (algo que se sabe).

La seguridad de la autenticación multifactor se basa en su enfoque por capas. Tener que superar varios factores de autenticación supone una dificultad considerable para los atacantes. Aunque un atacante logre descubrir la contraseña del usuario, no le servirá de nada sin el dispositivo de confianza. Por otra parte, si el usuario pierde el dispositivo, quien

encuentre dicho dispositivo no lo podrá utilizar a menos que conozca la contraseña del usuario.

Los sistemas de autenticación basada en teléfono aprovechan el teléfono del usuario como dispositivo de confianza para el segundo factor de autenticación. Es muy difícil duplicar teléfonos y también resulta muy difícil interceptar números de teléfono. Además, son dispositivos personales de uso muy extendido que normalmente llevan consigo todo el mundo, por lo que no es necesario invertir en la compra masiva de tarjetas inteligentes u otro tipo de hardware. La combinación del teléfono y un nombre de usuario con contraseña proporcionan una autenticación multifactor muy segura (este sistema está muy extendido entre los servicios de banca on-line).

Por otro lado, en lo que se refiere a la virtualización[20] de servicios en la nube, la necesidad de cifrado es, sin lugar a dudas, muy importante. Es imprescindible para que las empresas mantengan el control sobre sus datos.

Claves de Cifrado 1

Un correcto sistema de cifrado garantiza la privacidad de los datos, y al mismo tiempo hace que sea posible aprovechar los beneficios de infraestructuras como servicios (IaaS) y así reducir costes de complejos y caros centros de datos propios. Manteniendo el control de las claves de encriptación, mantendremos el control sobre los datos, aunque estos estén viajando por la red o en estado de reposo en la nube.

Las llaves, o 'keys', nos garantizan la posibilidad del desmantelamiento o des-aprovisionamiento de nuestros datos en cualquier momento. Es indispensable tener el control sobre estos keys ya que nadie alquilaría una caja fuerte y le dejaría la llave a un desconocido para que éste se la guarde. Lo mismo pasa con la encriptación. Que el proveedor de servicios custodie sus keys no es lo más recomendable.

Cuando hablamos de cifrado de datos, no solo nos referimos al cifrado de la información que manejamos a diario en nuestros servidores o equipos o en la nube, sino también de nuestras copias de seguridad. ¿Qué pasaría si perdemos una cinta de copias o que nos la roban?. Si no está cifrada, pues sería no solo como perder una caja fuerte, sino como perder una caja fuerte con la puerta abierta. ¿Si estas cintas estuviesen cifradas, que pasaría?. La respuesta es muy simple – ¡nada! Sin acceso a las claves de cifrado, es casi imposible recuperar los datos. Es por ello que deberían guardar las claves de acceso de la forma más segura posible. En teoría si las claves de cifrado no tienen puertas traseras o inversa si se pierde esta clave no se podrá acceder a los datos de la copia de seguridad. Hay proveedores que dejan puertas traseras de acceso a los datos y se deberían evitar, nuestros datos son demasiado importantes para que cualquier tenga libre acceso a ellos.

Por otro lado, los servicios en la nube no solo proporcionan unas ventajas y oportunidades evidentes a organizaciones y particulares. Por el contrario, lo delincuentes también encuentran en ello potentes herramientas para poder llevar a cabo sus fines.

En este punto es obligado hacer referencia a la arquitectura EC2 de Amazon:

Amazon Elastic Compute Cloud (Amazon EC2) es un servicio web que proporciona capacidad informática con tamaño modificable en la nube. Está diseñado para facilitar a los desarrolladores recursos informáticos escalables basados en web [ñ]. Para ayudar a que los nuevos clientes de AWS empiecen a utilizar los servicios en la nube, AWS ofrece una capa de uso gratuito. La capa de uso gratuito puede emplearse para todo aquello que desee realizar en la nube: ejecutar nuevas aplicaciones, probar aplicaciones existentes en la nube o simplemente obtener experiencia práctica con AWS. incluye 750 horas de microinstancias de Windows y Linux al mes durante un año.

Ya han aparecido noticias del uso de esta arquitectura para realizar ataques de ruptura de contraseñas: Thomas Roth, analista de seguridad alemán, desarrolló un software para la plataforma EC2. Este software le permitía probar por fuerza bruta unas 400000 claves por segundo gracias al uso de los equipos de Amazon.

Este rendimiento lo conseguía alquilando a Amazon 8 clusters, que constaban cada uno de ellos del siguiente hardware: 22GB RAM, 2 Intel Xeon X5570, 2 NVIDIA Tesla M2050. Cada uno de los cluster le costaba 2.10$ por hora por lo que en su conjunto costaba 16.80$ por hora. Finalmente consiguió obtener la clave en un tiempo de 20 minutos. Más tarde, depuró el software y obtuvo la misma clave en 6 minutos. Este software consta por una parte un

programa en el servidor, y por otra parte en el cliente, por lo que se podría decir que el funcionamiento de esta herramienta es muy parecida al software “Pirit“. Thomas Roth, tiene su cuenta en Twitter [o].

Existen otras formas de obtener una clave WPA/WPA2 usando Cloud Computing. Hay webs como **CloudCracker** que no sólo permiten descifrar claves WPA o WPA2, también permiten obtener otros cifrados como NTLM, SHA-512, MD5 o MS-CHAPv2. El precio de este servicio varía según el tamaño del diccionario a utilizar. Por el simple hecho de utilizar diccionarios, es mucho menos efectivo, ya que si nuestra clave WPA[21] es alfanumérica con caracteres especiales, probablemente no se encuentre en ese diccionario [p].

Por ultimo hay que decir que si utilizamos uno de estos servicios, debemos tener en cuenta que debemos usar para comprobar la fortaleza de nuestras redes, en caso de utilizarlo con redes ajenas estaríamos cometiendo un delito.

Quiero comentar brevemente las diferentes opciones de ataques o intrusiones, que parecen más fáciles de llevar a cabo en entornos cloud que en entornos tradicionales:

- VM-level Attacks: Se aprovechan de lasposibles vulnerabilidades en el hypervisor o en la tecnología de virtualización utilizada por los proveedores de servicios en la nube.

- Phishing Cloud Provider: Los ataques de phishing y de ingeniería social en general, tienen un nuevo vector.

- Cloud Provider Vulnerability: Se refiere a las vulnerabilidades de tipo SQL Injection o Cross Site Scripting.

- Expanded Network Attack Surface: Es muy complicado proteger toda la infraestructura utilizada para conectarse e interactuar con los servicios en la nube.

- Authentication and Authorization: Los procedimientos e infraestructuras de autenticación y autorización de las organizaciones, habitualmente no se extienden a los servicios en la nube.

- Forensics in the Cloud: Los métodos de análisis forense tradicionales, también se pueden aplicar en este entorno.

De todos los puntos citados anteriormente, me parece central el problema de la autenticación y autorización. La computación empleando servicios en la nube, abre unas enormes posibilidades a los ciberdelincuentes tal y como comenté anteriormente. Esto implica la necesidad de imponer unos esquemas de cifrado y de firma que estén a la altura si se quiere conseguir la seguridad en las comunicaciones. A continuación se citan algunas técnicas de cifrado de tratan de dar respuesta a esta problemática:

Identity Based Encryption (IBE): Es una forma de criptografía de clave pública, en la que un servidor de una tercera parte utiliza un identificador simple como una dirección de e-mail, para generar una clave pública que pueda ser utilizada para cifrar y descifrar mensajes:

- Linear Search Algorithm: Emplea un algoritmo de cifrado simétrico para cifrar texto plano.

- Identity Based Signature: es un esquema de identidad basada en firma que es determinista, si la firma en un mensaje por parte de un mismo usuario es siempre el mismo.

- Homomorphic Encryption: Se refiere al cifrado en el que el texto plano y el texto cifrado, se tratan con funciones algebráicas equivalentes.

- Public Key Encryption With Keyword Search: Un esquema PEKS consiste en el empleo de 4 algoritmos polinomiales: KeyGen, Trapdoor, PEKS, y Test.

- Attribute Based Encryption: en ABE los atributos y políticas asociadas a un mensaje y a un usuario, deciden que usuario puede descifrar un texto cifrado. Una autoridad central creará las claves secretas para los usuarios, basándose en los atributos y directivas de cada usuario.

Fuente bibliográfica: Cryptography and Encription In Cloud Computing. Simarjeet Kaur. VSRD International Journal of Computer Science & Information Technology, Vol. 2 (3), 2012, 242-249 [q].

Creo que sería enormemente interesante poder en el futuro investigar todas estas cuestiones, para tratar de definir también una metodología de instalación y configuración segura de sistemas de almacenamiento en la nube y su correspondiente proceso de auditoría.

Seguridad en la Nube 1

3x000 Referencias

3x010 Bibliografía

[1] Litchfield, David (2005). The Database Hacker's Handbook: Defending Database Servers. John Wiley & Sons.

[2] Marlene Theriault & William Heney (1998).Oracle Security. O'Reilly.

[3] Andrew Cumming & Gordon Russell (2006). SQL Hacks: Tips & Tools for Digging into Your Data. O'Reilly.

[4] Jeffrey Wheatman (2012). El monitoreo de la actividad en bases de datos está evolucionando hacia la auditoría y la protección de bases de datos. Gartner Inc.

[5] Erikson, Jon (1977). HACKING The Art of Exploitation. No starch press.

3x020 Webgrafía

[a] http://www.oracle.com/technetwork/licenses/standard-license-152015.html

[b] http://www.inteco.es/Formacion/Conceptos_de_seguridad/

[c] http://www.imperva.com/Resources/WhitePapersAndEbooks

[d] http://www.sifma.org/uploadedfiles/societies/sifma_internal_auditors_society/top10-database-vulnerabilities-and-misconfigurations.pdf

[e] http://nvd.nist.gov/

[f] http://nvd.nist.gov/cvss.cfm

[g] https://www.ccn-cert.cni.es/publico/seriesCCN-STIC/series/Indice_series_CCN-STIC.pdf

[h] http://iase.disa.mil/stigs/

[i] http://www.greensql.com/

[j] http://www.pentest.es/

[k] http://www.blackploit.com/

[l] http://www.dragonjar.org/como-realizar-un-pentest.xhtml

[m] https://pentestmag.com/

[n] http://www.dominandolasredes.net/2012/01/curso-backtrack-5-en-espanol.html

[ñ] http://aws.amazon.com/es/ec2/

[o] https://twitter.com/StackSmashing

[p] https://www.cloudcracker.com/

[q] www.vsrdjournals.com

4x000 Anexos.

4x010 Anexo A. Glosario.

1 APTs (Advanced Persistent Threats)

Es tipo de riesgo de ciberseguridad de mayor gravedad a los habituales dada su sofisticación y su diseño específico para un objetivo concreto. Suelen ser diseñados para mantenerse oculto en el sistema atacado incluso durante años, y poder extraer información hacia el servidor de control.

2 Ciberdependiente

Es un término que expresa la dependencia de las organizaciones respecto de las tecnologías de la información y comunicaciones para poder desarrollar su actividad.

3 Hacking

La traducción literal sería algo así como la acción de coger un atajo o amañar algo. En términos informáticos, se refieren al conjunto de técnicas y acciones conducentes a vulnerar la seguridad de un sistema de información. Algunos lo definen incluso como un arte.

4 Ciberterrorismo

El ciberterrorismo o terrorismo electrónico es el uso de medios de tecnologías de información, comunicación, informática, electrónica o similar con el propósito de generar terror o miedo generalizado en una población, clase dirigente o gobierno, causando con ello una violación a la libre voluntad de las personas. Los fines pueden ser económicos, políticos o religiosos principalmente.

5 SGBDR

Un Sistema Gestor de Bases de Datos Relacionales es un conjunto de programas que permiten el almacenamiento, modificación y extracción de la información en una base de datos relacional, además de proporcionar herramientas para añadir, borrar, modificar y analizar los datos. Los usuarios pueden acceder a la información usando herramientas específicas de interrogación y de generación de informes, o bien mediante aplicaciones al efecto Los SGBDR también proporcionan métodos para mantener la integridad de los datos, para administrar el acceso de usuarios a los datos y para recuperar la información si el sistema se corrompe. Permite presentar la información de la base de datos en variados formatos. La mayoría de los SGBDR incluyen un generador de informes.

6 Buffer overflow

En seguridad informática y programación, un **desbordamiento de buffer** (del inglés *buffer overflow* o *buffer overrun*) es un error de software que se produce cuando un programa no controla adecuadamente la cantidad de datos que se copian sobre un área de memoria reservada a tal efecto (buffer): Si dicha cantidad es superior a la capacidad preasignada, los bytes sobrantes se almacenan en zonas de memoria adyacentes, sobrescribiendo su contenido original, que probablemente pertenecían a datos o código almacenados en memoria. Esto constituye un fallo de programación.

7 SSH

(**S**ecure **SH**ell, en español: intérprete de órdenes segura) es el nombre de un protocolo y del programa que lo implementa, y sirve para acceder a máquinas remotas a través de una red. Permite manejar por completo la computadora mediante un intérprete de comandos, y también puede redirigir el tráfico de X para poder ejecutar programas gráficos si tenemos un Servidor X (en sistemas Unix y Windows) corriendo.

Además de la conexión a otros dispositivos, SSH nos permite copiar datos de forma segura (tanto archivos sueltos como simular sesiones FTP cifradas), gestionar claves RSA para no escribir claves al conectar a los dispositivos y pasar los datos de cualquier otra aplicación por un canal seguro tunelizado mediante SSH.

8 IPsec

Internet Protocol security es un conjunto de protocolos cuya función es asegurar las comunicaciones sobre el Protocolo de Internet (IP) autenticando y/o cifrando cada paquete IP en un flujo de datos. IPsec también incluye protocolos para el establecimiento de claves de cifrado.

Los protocolos de **IPsec** actúan en la capa de red, la capa 3 del modelo OSI. Otros protocolos de seguridad para Internet de uso extendido, como SSL, TLS y SSH operan de la capa de transporte (capa 4 del modelo OSI) hacia arriba. Esto hace que IPsec sea más flexible, ya que puede ser utilizado para proteger protocolos de la capa 4, incluyendo TCP y UDP, los protocolos de capa de transporte más usados. Una ventaja importante de IPsec frente a SSL y otros métodos que operan en capas superiores, es que para que una aplicación pueda usar IPsec no hay que hacer ningún cambio, mientras que para usar SSL y otros protocolos de niveles superiores, las aplicaciones tienen que modificar su código.

9 Hash

A las **funciones hash** (adopción más o menos directa del término inglés *hash function*) también se les llama **funciones picadillo**, **funciones resumen** o **funciones de digest** (adopción más o menos directa del término inglés equivalente *digest function*). Una función hash H es una función computable mediante un algoritmo,

$$H: U \rightarrow M$$
$$X \rightarrow h(x),$$

que tiene como entrada un conjunto de elementos, que suelen ser cadenas, y los convierte (mapea) en un rango de salida finito, normalmente cadenas de longitud fija. Es decir, la función actúa como una proyección del conjunto U sobre el conjunto M.

10 Inyección SQL

Es un método de infiltración de código intruso que se vale de una vulnerabilidad informática presente en una aplicación en el nivel de validación de las entradas para realizar consultas a una base de datos.

El origen de la vulnerabilidad radica en el incorrecto chequeo y/o filtrado de las variables utilizadas en un programa que contiene, o bien genera, código SQL. Es, de hecho, un error de una clase más general de vulnerabilidades que puede ocurrir en cualquier lenguaje de programación o script que esté embebido dentro de otro.

Se conoce como Inyección SQL, indistintamente, al tipo de vulnerabilidad, al método de infiltración, al hecho de incrustar código SQL intruso y a la porción de código incrustado.

11 ITIL

La **Biblioteca de Infraestructura de Tecnologías de Información**, frecuentemente abreviada **ITIL** (del inglés *Information Technology Infrastructure Library*), es un conjunto de conceptos y prácticas para la gestión de servicios de tecnologías de la información, el desarrollo de tecnologías de la información y las operaciones relacionadas con la misma en general. ITIL da descripciones detalladas de un extenso conjunto de procedimientos de gestión ideados para ayudar a las organizaciones a lograr calidad y eficiencia en las operaciones de TI. Estos procedimientos son independientes del proveedor y han sido desarrollados para servir como guía que abarque toda infraestructura, desarrollo y operaciones de TI.

12 Plantillas STIC

Son las guías publicadas por el Centro Criptológico Nacional, conducentes a instalar y configurar sistemas concretos, observando estrictas medidas de seguridad que permitan certificarlos para manejar información de carácter clasificado.

13 CCN

El Centro Criptológico Nacional, es el Organismo responsable de coordinar la acción de los diferentes organismos de la Administración que utilicen medios o procedimientos de cifra, garantizar la seguridad de las Tecnologías de la Información en ese ámbito, informar sobre la adquisición coordinada del material criptológico y formar al personal de la Administración especialista en este campo.

14 DoD (Deny of Service)

Los ataques de denegación de servicio, son ataques a sistemas de computadoras o red que causa que un servicio o recurso sea inaccesible a los usuarios legítimos. Normalmente provoca la pérdida de la conectividad de la red por el consumo del ancho de banda de la red de la víctima o sobrecarga de los recursos computacionales del sistema de la víctima.

15 Bruce Schneiner

Nacido el 15 de enero de 1963, es un criptógrafo experto en seguridad informática y escritor. Es el autor de diversos libros de seguridad informática y criptografía, y es el fundador y oficial jefe tecnológico de Counterpane Internet Security.

16 Firewall de base de datos

Es una aplicación de software que permite filtrar, mediante un conjunto de reglas preestablecidas, las peticiones que llegan al manejador de bases de datos.

17 Hacking ético

Se entiende con este término, como aquellas actividades de los hackers conducentes a detectar vulnerabilidades y mejorar la seguridad de los sistemas de información, tanto desde el punto de vista profesional como amateur.

18 Pentesting

La **Prueba de Penetración** (Penetration Test o PenTest en Inglés) consiste en una evaluación activa de las medidas de seguridad de la información. El propósito es detectar los puntos débiles que puedan ser capitalizados para violar cualquiera de las tres condiciones necesarias de la información: confidencialidad, integridad y disponibilidad.

19 Stakeholder

Stakeholder es un término inglés utilizado por primera vez por R. E. Freeman en su obra: "*Strategic Management: A Stakeholder Approach*" (Pitman, 1984), para referirse a *«quienes pueden afectar o son afectados por las actividades de una empresa»*.

Estos grupos son los públicos interesados o el entorno interesado ("stakeholders"), que según Freeman deben ser considerados como un elemento esencial en la planificación estratégica de los negocios.

20 Virtualización

En Informática, **virtualización** es la creación -a través de software- de una versión virtual de algún recurso tecnológico, como puede ser una plataforma de hardware, un sistema operativo, un dispositivo de almacenamiento u otros recursos de red. En los ámbitos de habla inglesa, este término se suele conocer por el numerónimo "**v12n**".

Dicho de otra manera, se refiere a la abstracción de los recursos de una computadora, llamada Hypervisor o VMM (Virtual Machine Monitor) que crea una capa de abstracción entre el hardware de la máquina física (host) y el sistema operativo de la máquina virtual (virtual machine, guest), dividiéndose el recurso en uno o más entornos de ejecución.

21 WPA

Wi-Fi Protected Access, llamado también ***WPA*** (en español «Acceso Wi-Fi protegido») es un sistema para proteger las redes inalámbricas (Wi-Fi). Creado para corregir las deficiencias del sistema previo, Wired Equivalent Privacy (WEP). Los investigadores han encontrado varias debilidades en el algoritmo WEP (tales como la reutilización del vector de inicialización (IV), del cual se derivan ataques estadísticos que permiten recuperar la clave WEP, entre otros). WPA implementa la mayoría del estándar IEEE 802.11i, y fue creado como una medida intermedia para ocupar el lugar de WEP mientras 802.11i era finalizado. WPA fue creado por la Wi-Fi Alliance («Alianza Wi-Fi»).

HACKING ES EL ARTE DE HACER LO POSIBLE DE LA MANERA MENOS PROBABLE