



**UNIVERSIDAD NACIONAL AUTÓNOMA
DE MÉXICO**

FACULTAD DE INGENIERÍA

**DISEÑO E IMPLANTACIÓN DE UN MODELO DE
SEGURIDAD PARA LOS SISTEMAS DE BASES
DE DATOS DE USECAD**

T E S I S

**QUE PARA OBTENER EL TÍTULO DE:
INGENIERO EN COMPUTACIÓN**

P R E S E N T A N:

**NÁJERA MONTIEL SUSANA
REYNOSO ANDRADE MAYELLY**

**DIRECTOR DE TESIS
ING. ARMANDO VEGA ALVARADO**

**CODIRECTORA DE TESIS
M. C. MARÍA JAQUELINA LÓPEZ BARRIENTOS**



MÉXICO, D. F.

2005



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

A la Universidad Nacional Autónoma de México:
por brindarnos la oportunidad de realizar una carrera profesional
y el orgullo de pertenecer a ella.

A la Facultad de Ingeniería por la satisfacción
que sentimos de ser una parte integral dentro
de sus generaciones, y por ser una institución
de gran reconocimiento y prestigio.

A nuestros profesores de toda la carrera
por compartirnos sus conocimientos
dentro de nuestra formación profesional.

A nuestro director de tesis:
Ing. Armando Vega Alvarado,
y a nuestra codirectora:
M.C. Ma. Jaquelina López Barrientos:
que gracias a su orientación, apoyo y experiencia
logramos realizar este trabajo.

Y un enorme agradecimiento
por su apoyo y colaboración
al Ing. Filiberto Manson.

Gracias.

Susana y Mayelly.

A Dios: por darme la fortaleza en los momentos más difíciles de mi vida como Universitaria.

A mis padres: Micaela Montiel y Pedro Nájera por su gran esfuerzo y apoyo durante toda mi vida, pero sobre todo por su amor y entrega incondicional. Por su ejemplo, que me ha llevado a ser una mujer con valores, comprometiéndome a alcanzar cada uno de mis sueños y metas.

A mis hermanos: Cristina, Verónica y Víctor por su amor, comprensión y ejemplo; por mantenernos unidos y apoyándonos en los momentos que más nos necesitamos. Pero sobre todo por alentarme a nunca darme por vencida.

A Jorge Silva: sobre todo por tu gran amor y por estar siempre a mi lado. Gracias amor por tu comprensión y por apoyarme en cada momento de mi carrera y de mi vida.

A Osvaldo Flores, José Luis García, Rodrigo Flores y Luis Fernando García: un agradecimiento especial por su apoyo, cariño.

A mis amigos: que en las buenas y en las malas siempre confiaron en mí y nunca dudaron de nuestra amistad. Gracias por comprenderme y sobre todo por que los buenos amigos nunca se olvidan. Ellos saben quienes son ¡salud!

Y finalmente gracias a Mayelly por ser una gran amiga y compañera de carrera y tesis, por su paciencia y apoyo para la realización de este trabajo y culminación de esta etapa de nuestra vida.

A todos ellos por que siempre estarán en mi corazón, con respeto y admiración.

Susana Nájera Montiel

A Dios: Por haberme dado fuerza y voluntad para terminar una etapa de mi vida tan importante y ante todo por dejar que en esta etapa de mi vida esté rodeada de las personas que quiero.

A mis padres, María Elena Andrade y Leonidas Reynoso:
Por el amor que siempre han manifestado; el apoyo, dedicación, enseñanza y sacrificio que me han brindado durante toda mi vida; por alentarme a lograr la primera y una de mis más grandes metas. Por todo su esfuerzo, sirva este trabajo de tesis como un homenaje y reconocimiento a ellos.

Para ellos todo mi amor y respeto.

A mi hermano Eduardo Reynoso:
Por su ayuda y apoyo incondicional que me ha brindado para crecer como ser humano y profesionalmente.

Con cariño mil gracias.

A José Antonio:
Por lo que representas para mí en todo momento y por siempre, agradezco infinitamente todo tu amor, cariño, comprensión, apoyo y esfuerzo. Te amo.

Con todo mi amor.

A mis amigos:
Por que forman una parte muy importante de mi persona.

Gracias a todos.

A mi compañera de tesis, Susana: Por su paciencia sin igual.

Mayelly Reynoso Andrade

ÍNDICE

Prologo

CAPÍTULO 1. INTRODUCCIÓN

1. Introducción

1.1 Antecedentes

1.1.1 Secretaría de Servicios Académicos

1.1.2 Unidad de Servicios de Cómputo Administrativos

1.2 Bases de Datos

1.2.1 Historia de las Bases de Datos

1.2.2 Definición de Base de Datos

1.2.3 Lenguajes de Bases de Datos

1.2.4 Sistema de Base de Datos

1.2.5 Sistema Manejador de Base de Datos (SMBD)

1.2.6 Administrador de Bases de Datos (DBA)

1.2.7 Métodos de Organización de las Bases de Datos

1.2.8 Instancias y Esquemas

1.2.9 Independencia de los Datos

1.2.10 Claves (Llaves)

1.2.11 Bases de Datos Relacionales

1.2.12 Modificación de la Base de Datos

1.2.13 Diseño de Base de Datos Relacionales

1.2.14 Peligros en el Diseño de Base de Datos Relacionales

1.2.15 Normalización

CAPÍTULO 2. ARQUITECTURA DE SEGURIDAD

2. Arquitectura de Seguridad

2.1 Definición

2.1.1 Seguridad en Cómputo

2.1.2 Seguridad en Bases de Datos

2.1.3 Seguridad en la Tecnología de la Información y las Comunicaciones

2.2 Servicios de Seguridad

2.2.1 Confidencialidad

2.2.2 Integridad

2.2.3 Disponibilidad

2.2.4 Autenticidad

2.3 Tipos de Ataques

- 2.3.1 Ataques de acceso
- 2.3.2 Ataque de modificación
- 2.3.3 Ataque denegación de servicio
- 2.3.4 Ataque de refutación
- 2.4 Administración de Riesgos
 - 2.4.1 Activos
 - 2.4.2 Vulnerabilidad
 - 2.4.3 Amenaza
 - 2.4.4 Análisis de Riesgo
- 2.5 Problemas de Seguridad en las Bases de Datos
 - 2.5.1 Requerimientos de Protección de una Base de Datos
 - 2.5.2 Seguridad Informática
 - 2.5.3 Aspectos Principales de las Bases de Datos enfocados a la Seguridad
 - 2.5.4 Tipos de políticas para la base de datos
- 2.6 Norma ISO/IEC 17799
 - 2.6.1 Historia
 - 2.6.2 Objetivo
 - 2.6.3 Áreas de control
 - 2.6.4 Organización de la Seguridad

CAPÍTULO 3. ANÁLISIS DE RIESGOS

- 3. Análisis de Riesgos
 - 3.1 Metodología para el Análisis de Riesgo
 - 3.1.1 Metodología FRAP
 - 3.1.2 Riesgos Transferibles o Tolerables
 - 3.2 Análisis y Gestión de Riesgos
 - 3.2.1 Pre-FRAP
 - 3.2.2 FRAP
 - 3.2.3 Post-FRAP

CAPÍTULO 4. IMPLANTACIÓN E INTEGRACIÓN DE LAS SOLUCIONES DE SEGURIDAD

- 4. Implantación e Integración de las soluciones de seguridad
 - 4.1 Seguridad en Capa 1
 - 4.1.1 Control de Acceso al Servidor de Base de Datos
 - 4.1.2 Limitación de Recursos de la Base de Datos
 - 4.1.3 Roles y tareas asignadas
 - 4.1.4 Roles Especiales
 - 4.2 Seguridad en Capa 2

- 4.2.1 Control de Acceso a la Base de Datos de Usuarios
- 4.2.2 Grupos de acceso
- 4.3 Seguridad en Capa 3
 - 4.3.1 Permisos Sobre Objetos
 - 4.3.2 Herramientas de Auditoría de la Base de Datos

CAPÍTULO 5. RESULTADOS Y CONCLUSIONES

- 5. Resultados y Conclusiones
 - 5.1 Resultados
 - 5.2 Conclusiones

ANEXOS

- 1. Lista de parámetros de configuración
- 2. Valores en evento y columna extrainfo

APÉNDICES

- A. Guía para la elaboración de políticas
- B. Glosario de términos

Bibliografía

PRÓLOGO

A pesar de que actualmente existen muchos autores interesados por generar una cultura de seguridad informática, son pocas las tesis que se han publicado con esta temática. El rápido avance de la tecnología requiere que los estudiantes interesados tengan más herramientas de apoyo. La presente tesis tiene como objetivo diseñar e implantar los procesos de seguridad para garantizar la integridad, confidencialidad y disponibilidad de los sistemas de bases de datos de la USECAD.

En el Capítulo 1, se da una breve introducción de la Unidad de Servicios de Cómputo Administrativos (USECAD) la cual forma parte de la Secretaria de Servicios Académicos (SSA) y las principales actividades que realiza, posteriormente continuamos con una reseña histórica de las bases de datos, así como, de los conceptos fundamentales de las bases de datos.

Para el Capítulo II, se tomaron conceptos básicos, prácticos y necesarios para su elaboración; ya que para los siguientes capítulos el lector deberá conocerlos para comprender el objetivo de esta tesis y empaparse poco a poco de los conceptos que un sistema necesita para considerarse seguro.

Una herramienta que los lectores podrán utilizar es la metodología que se describe paso a paso en el Capítulo III, dicha herramienta será útil para llevar a acabo un análisis de riesgos para cualquier sistema informático, pues es sencilla y eficaz.

En el Capítulo IV, el principal y último de esta tesis, presentamos el desarrollo del análisis de riesgos para el Servidor de Producción de Base de Datos de USECAD , así como, la integración de las herramientas de seguridad que a través el Sistema Gestor de Base de Datos (SQL server SYBASE), puede utilizar el Administrador de Base de Datos (DBA).

Finalmente, se presentan el glosario de términos como apoyo para el lector, además un apéndice que será de utilidad a las personas que requieran crear e implementar políticas de seguridad, el documento es una guía de creación de políticas.

CAPÍTULO I

INTRODUCCIÓN

1. Introducción

1.1 Antecedentes

1.1.1 Secretaría de Servicios Académicos

La Facultad de Ingeniería cuenta con la Secretaría de Servicios Académicos (SSA).

La cual tiene como objetivo:

- Proporcionar los servicios de cómputo para las acciones de administración escolar que requieren los órganos de la Facultad de Ingeniería, así mismo la conformación de una red de cómputo, que permita a los funcionarios de la facultad el acceso a la base de datos almacenada en la Unidad de Servicio de Cómputo Administrativo (USECAD).

Sus funciones son:

- Coordinar y supervisar las actividades con el fin de proporcionar adecuada y oportunamente los servicios de cómputo que requieren los usuarios de la USECAD.
- Proponer a las autoridades de la facultad proyectos externos que permitan obtener ingresos extraordinarios.
- Participar en la revisión del avance de las actividades de administración escolar y el establecimiento de los calendarios semestrales de actividades.
- Administrar los recursos de cómputo e instalaciones a su cargo, para que estén en condiciones óptimas de operación.
- Brindar los servicios de diseño, desarrollo, capacitación, consultoría y mantenimiento de los sistemas elaborados bajo la responsabilidad de la USECAD.
- Dar soporte técnico a las áreas usuarias que así lo requieran.
- Responsabilizarse en la formación de personal a través de la capacitación en cómputo y administración de redes a las áreas usuarias que así lo requieran.
- Participar en las actividades de los comités a los que sea integrado como son el de bibliotecas y el de cómputo.

Estrategias de Desarrollo Informático.

- Crear y mantener archivos históricos confiables que permitan generar reportes estadísticos veraces.
- Capacitar continuamente al personal de la USECAD en los lenguajes de programación y sistemas operativos de interés.
- Mantener actualizada la red de cómputo de las bibliotecas de la Facultad y el acervo de CD ROM.
- Establecer un plan de becarios para contar con personal capacitado que ayude a desarrollar las actividades de la USECAD.
- Dar consistencia a los datos almacenados en la base de datos.
- Completar el equipamiento básico, lo más pronto posible.
- Considerar actividades de superación académica (cursos avanzados) impartidos por los fabricantes o proveedores autorizados, para el personal de USECAD.
- Adquirir nuevos volúmenes de CD ROM en función de las solicitudes de los alumnos y profesores.
- Completar los sistemas que requieren el Departamento de Administración Escolar y la Oficina de Servicios Escolares para proporcionar el mejor servicio posible a los alumnos de la facultad.

A continuación se presenta en la figura 1.1 el organigrama de la SSA.



Figura 1.1 Organigrama de la SSA

Como el desarrollo informático se enfoca a USECAD, ahora veremos afondo lo que es USECAD, sus funciones y objetivos.

1.1.2 Unidad de Servicios de Cómputo Administrativos

La Facultad de Ingeniería cuenta con la Unidad de Servicios de Cómputo Administrativos (USECAD) la cual forma parte de la Secretaría de Servicios Académicos (SSA).

En USECAD se hace la actualización de hardware y software a servidores. USECAD se encarga de las inscripciones y reinscripciones de los alumnos, por medio de consultas a la base de datos vía Web. Así como controlar el acceso a la base de datos a funcionarios de la Facultad que así lo requieran.

Objetivos de la USECAD:

- Tendrá conexión en red de todos los funcionarios de la facultad y su acceso a la base de datos que administra la USECAD.
- Proporcionara servicios de consulta vía Internet sobre asignaturas, grupos, profesores, horarios, cupos, vacante y resultados de la inscripción.
- Se podrá hacer el llenado de la solicitud de reinscripción vía Internet.
- Inscribirá en línea únicamente.

Sus funciones son:

- La actualización de hardware y software a servidores de base de datos para un mejor desempeño para brindar un mejor servicio a dependencias, alumnos y académicos.
- Inscripción y reinscripción de los alumnos de consulta a la base de datos vía Web.
- Acceso a la base de datos a funcionarios de la Facultad que así lo requieran.

La USECAD está desarrollando un sistema de consulta general, es una aplicación exclusivamente para uso de los secretarios y jefes de división, con lo cual se está minimizando los tiempos de respuesta, las cargas de trabajo, y la elaboración de información estadística. Este sistema tiene las siguientes características:

- Genera varios tipos de estadísticas del semestre como de semestres anteriores, tales como avances por semestres, por sexo, por generación, etc.
- Consulta de historiales académicos.
- Consulta de información personal tanto de profesores como de alumnos.
- Consulta de grupos por materia, horarios, profesores, etc.
- Consulta de historiales por grupo.

Este sistema se encuentra por liberarse, con lo cual cada división o secretaría podrá generar la información estadística necesaria en línea.

Docencia.

- Se tiene considerado la inserción de tres becarios adicionales con el fin de capacitarlos para la gran variedad de necesidades dentro de la USECAD.

A continuación en la Figura 1.2 se presenta el organigrama de USECAD.

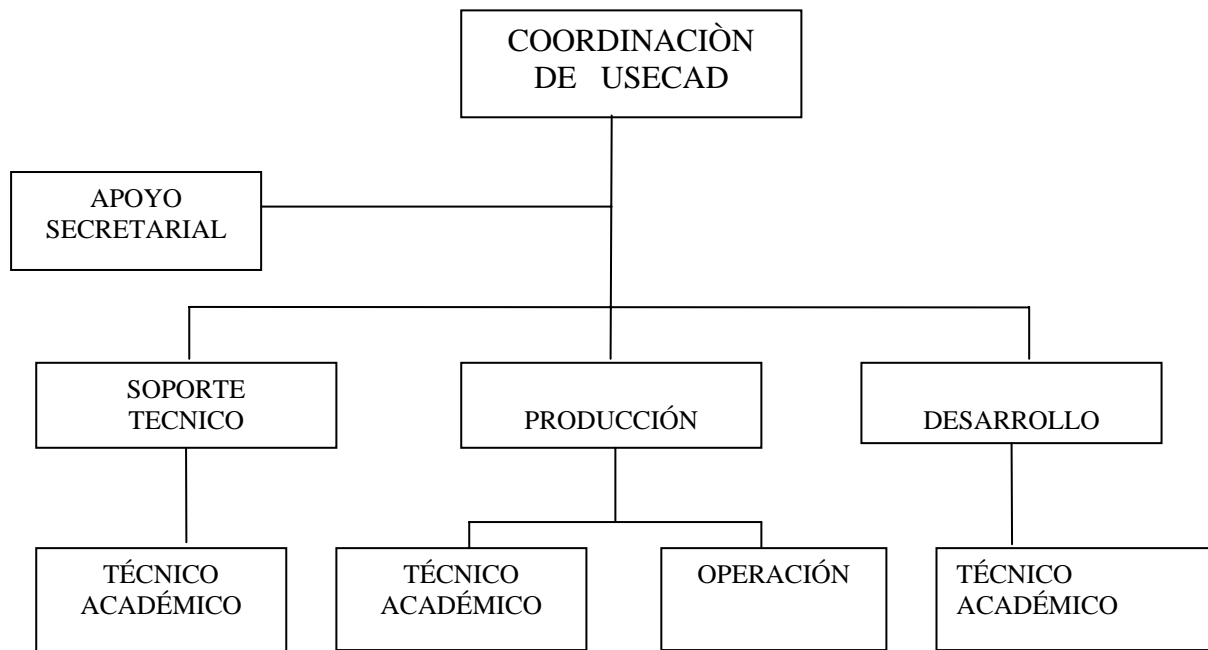


Figura 1.2 Organigrama de USECAD

1.2 Bases de Datos

1.2.1 Historia de las Bases de Datos

Tuvieron sus orígenes en 1960 - 1962, cuando se empezaron a usar las maquinas que codificaban la información en tarjetas perforadas por medio de agujeros. Las bases de datos se crean con el objetivo de almacenar grandes cantidades de datos que antes se almacenaba en libros, lo que era lento, costoso y complejo (cualquier actualización a realizar, había que hacerla en cada uno de los libros en los que apareciera dicha información a modificar). Las primeras bases de datos manejaban ficheros que eran almacenados en tarjetas o soportes magnéticos.

Cuando los ordenadores evolucionan, aparecen las cintas y los discos, a la vez que las maquinas son dotadas de mucha mas potencia y facilidad de manipulación, es por tanto en ese momento cuando las bases de datos comienzan a ser realmente útiles.

En 1963 se acuña el término de bases de datos en el simposio de Santa Mónica ('DATA BASE'). Y en 1967 Codasyl, cambia su nombre por el de 'Data Base Taskgroup'.

En 1970 se convoca una Conferencia de Lenguajes de Programación y se establece un modelo llamado CODASYL (Modelo para el tratamiento de bases de datos) que fue publicado por E. Cod en 1970. Cod, propuso una forma de organizar las bases de datos mediante un modelo matemático lógico. Una vez creado este modelo se crea un modelo estándar de actuación. Las bases de datos se crean por que son compactas, rápidas, cómodas y ofrecen la información actual sometida a la última actualización.

1.2.2 Definición de Base de Datos

¿Qué es una base de datos?

- “Es un conjunto, colección o depósito de datos almacenados en un soporte informático de acceso directo”.¹
- “Se designa una colección de datos que es administrada por un sistema de administración de base de datos”.²
- “Es una colección de archivos interrelacionados, creados por un DBMS”.³
- “Es un conjunto de datos relacionados entre sí”.⁴
- “Es un conjunto de datos persistentes que es utilizado por sistemas de aplicación de alguna empresa dada”.⁵

Definición

Base de datos es un conjunto exhaustivo no redundante de datos estructurados organizados independientemente de su utilización y su implementación en máquina accesibles en tiempo real y compatibles con usuarios concurrentes con necesidad de información diferente y no predicable en tiempo.

El contenido de una base de datos engloba a la información concerniente (almacenadas en archivos) de una organización, de tal manera que los datos estén disponibles para los usuarios, una finalidad de la base de datos es eliminar la redundancia o al menos minimizarla.

¹ De Miguel, Adoración y Piattini, Mario Gerardo, *Concepción y Diseño de Base de Datos*, Segunda Edición, Addison-Wesley Iberoamericana, México 1999, p. 43.

² Ullman, Jeffrey D. y Widon, Jennifer, *Introducción a los Sistemas de Base de Datos*, Prentice-Hall Hispanoamericana, S.A., México 1999, p. 1.

³ Tsai, Alice, *Sistema de Base de Datos: Administración y Uso*, Prentice-Hall Hispanoamericana, S.A., México 1990, p. 5.

⁴ Batini, Carlo et al., *Diseño Conceptual de Base de Datos*, Addison-Wesley Iberoamericana, México 1994, p. 4.

⁵ Elmasri, Ramez y Navathe, Shamkant, *Sistemas de Bases de Datos: Conceptos Fundamentales*, Segunda Edición, Addison-Wesley Iberoamericana, Wilmington, Delaware, EUA 1997, p. 2.

Los tres componentes principales de un sistema de base de datos son el hardware, el software DBMS y los datos a manejar, así como el personal encargado del manejo del sistema. En rigor, una base de datos es el conjunto de datos almacenados con una estructura lógica. Es decir, tan importante como los datos, es la estructura conceptual con la que se relacionan entre ellos. En la práctica, podemos pensar esto como el conjunto de datos más los programas (o *software*) que hacen de ellos un conjunto consistente.

Si no tenemos los dos factores unidos, no podemos hablar de una base de datos, ya que ambos combinados dan la coherencia necesaria para poder trabajar con los datos de una manera sistemática.

Conceptos básicos de archivos computacionales

Dato: “Conjunto de caracteres con algún significado, pueden ser numéricos, alfabéticos, o alfanuméricos.”⁶

Campo: “Es la unidad más pequeña a la cual uno puede referirse en un programa. Desde el punto de vista del programador representa una característica de un individuo u objeto.”⁷

Registro: “Colección de campos de iguales o de diferentes tipos.”⁸

Archivo: “Colección de registros almacenados siguiendo una estructura homogénea.”⁹

A continuación se presenta en la tabla 1.1 las características de una base de datos.

Características de una Base de Datos
Control centralizado de los datos.
Integridad de los datos.
Minimización de la redundancia.
Independencia de los datos y las aplicaciones.
Acceso concurrente a los datos.
Costo mínimo de almacenamiento y mantenimiento.
Versatilidad para la representación de relaciones.
Establecimiento de medidas de seguridad.
Facilidad para el cambio de hardware o software.

Tabla 1.1 Características de una base de datos

⁶ Tsai, Alice, op. cit., p. 3.

⁷ Idem.

⁸ Idem.

⁹ Idem.

Se considera que las bases de datos así como tienen una serie de ventajas para el manejo de la información también presentan algunas desventajas las cuales se muestran en la tabla 1.2.

Ventajas de las Bases de Datos	Desventajas de las Bases de Datos
Independencia de datos y tratamiento.	Situación Sistema tradicional Sistema de bases de datos.
Cambio en datos no implica cambio en programas y viceversa (Menor costo de mantenimiento).	Fuerte costo inicial.
Coherencia de resultados.	Programa.
Reduce redundancia.	Personal.
Acciones lógicamente únicas.	Equipos.
Se evita inconsistencia.	Rentable a medio o largo plazo.
Mejora en la disponibilidad de datos.	No hay Standard
No hay dueño de datos (No igual a ser públicos).	No solo se puede cambiar datos sino también el enfoque del sistema.
Ni aplicaciones ni usuarios.	Son costosas.
Guardamos descripción (Idea de catálogos).	Representan un consumo de recursos elevados.
Cumplimiento de ciertas normas.	Se necesita contratar personal capacitado.
Restricciones de seguridad.	La recuperación de una base de datos después de una falla puede requerir bastante tiempo.
Accesos (Usuarios a datos).	
Operaciones sobre datos.	
Efecto sinérgico.	

Tabla 1.2 Ventajas y Desventajas de las Bases de Datos

1.2.3 Lenguajes de Bases de Datos

Un sistema de base de datos proporciona dos tipos de lenguajes diferentes: uno para especificar el esquema de base de datos y el otro para expresar las consultas y actualizaciones de la base de datos.

Lenguaje de Definición de Datos.

Lenguaje de definición de datos (DDL: Data Definition Language): Un esquema de base de datos se especifica mediante un conjunto de definiciones expresadas mediante un lenguaje especial llamado lenguaje de definición de datos.

Este lenguaje permite definir la estructura lógica (o esquema) de la bases de datos. El esquema define las características de los registros dentro de un archivo: los campos de cada registro, sus nombres, el tipo de dato y la extensión.

Un subesquema es la manera en la cual a un programa de aplicación o a un usuario específico se les permite acceder los datos de un archivo. Esto puede limitar el acceso a los campos definir los derechos de acceso (sólo leer, leer y escribir).

El resultado de la compilación de las instrucciones en DDL es un conjunto de tablas que se almacenan en un archivo especial llamado Diccionario de Datos (DD) o Directorio de Datos. Un diccionario de datos es un archivo que contiene metadatos; es decir, datos acerca de los datos.

Este archivo se consulta antes de leer o modificar los datos reales del sistema de base de datos. La estructura de almacenamiento y los métodos de acceso usados por el sistema de base de datos se especifican mediante un conjunto de definiciones en un tipo especial del DDL llamado un lenguaje de almacenamiento y definición de datos.

El resultado de la compilación de estas definiciones es un conjunto de instrucciones para especificar los detalles de implementación de los esquemas de la base de datos los detalles normalmente se ocultan a los usuarios.

Lenguaje de Manipulación de Datos (Lenguaje de Consultas).

Lenguaje de Manipulación de Datos (DML: Data Manipulation Language): Por manipulación de datos se quiere decir:

- La recuperación de información almacenada en la base de datos.
- La inserción de información nueva en la base de datos.
- El borrado de información de la base de datos.
- La modificación de información almacenada en la base de datos.

Un lenguaje de manipulación de datos es un lenguaje que permite a los usuarios acceder o manipular los datos organizados mediante el modelo de datos apropiado. Incluye todos los comandos que permiten al usuario almacenar, recuperar, cambiar, borrar u ordenar los datos o registros dentro de la bases de datos.

Los dos tipos de lenguaje básicamente son:

- **DML procedimentales:** Requieren que el usuario especifique qué datos se necesitan y cómo obtener esos datos.
- **DML no procedimentales:** Requieren que el usuario especifique qué datos se necesitan, sin especificar cómo obtener esos datos.

Una consulta es una instrucción de solicitud para recuperar información. La parte de un DML que implica recuperación de información se llama lenguaje de consultas.

1.2.4 Sistema de Base de Datos

Sistema de información diseñado para manejar grandes cantidades de datos y producir información. Un sistema de bases de datos es básicamente un sistema para archivar en el computador; es decir, es un sistema computarizado cuyo propósito general es mantener información y hacer que éste disponible cuando se solicite. La información en cuestión puede ser cualquier cosa que se considere importante para el individuo o la organización a la cual debe servir el sistema; dicho de otro modo, cualquier cosa necesaria para apoyar el proceso general de atender los asuntos de un individuo u organización.

Equipo.

Los componentes de equipo del sistema son:

- Los volúmenes de almacenamiento secundario - por lo regular discos magnéticos de cabeza móvil - donde se conservan los datos almacenados, junto con los dispositivos de E/S asociados (unidades de disco, etc...), controladores de dispositivos, canales de E/S y demás.
- El procesador o procesadores y la memoria principal asociada que hacen posible la ejecución de los programas del sistema de bases de datos.

Programas.

Entre la base de datos física misma (los datos y como están almacenados) y los usuarios del sistema existe un nivel de programas, el manejador de base de datos o, en la mayoría de los casos, el sistema de administración de base de datos (SMBD, database management system).

Usuarios.

Podemos definir a los usuarios como toda persona que tenga todo tipo de contacto con el sistema de base de datos desde que este se diseña, elabora, termina y se usa.

Hay 4 tipos diferentes de usuarios, diferenciados por la forma en que esperan interactuar con el sistema.

- **Programadores de aplicaciones:** Los profesionales en computación interactúan con el sistema por medio de llamadas en DML, las cuales están incorporadas en un programa escrito en un lenguaje principal (Cobol, Pascal, etc...). Estos son programas de aplicación. Las llamadas en DML están precedidas de un carácter especial de forma que se pueda generar el código apropiado. Un preprocesador especial, llamado precompilador de DML, convierte las sentencias en DML a llamadas normales en el lenguaje de programación. El programa resultante se ejecuta entonces por el compilador del lenguaje de programación, el cual genera el código objeto apropiado.
- **Usuarios sofisticados:** Los usuarios sofisticados interactúan con el sistema sin escribir programas. En cambio escriben sus preguntas en un lenguaje de consultas de base de datos. Cada consulta se somete a un procesador de consultas cuya función es tomar una sentencia en y descomponerla en instrucciones que entienda el gestor de bases de datos.
- **Usuarios especializados:** Usuarios sofisticados que escriben aplicaciones de bases de datos que no encajan en el marco tradicional de procesamiento de datos; sistemas de diseño ayudados por computadoras, sistemas expertos y basados en conocimiento, etc.
- **Usuarios ingenuos:** Los usuarios no sofisticados interactúan con el sistema invocando a uno de los programas de aplicación permanentes que se han escrito anteriormente en el sistema de base de datos, podemos mencionar al usuario ingenuo como el usuario final que utiliza el sistema de base de datos sin saber nada del diseño interno del mismo por ejemplo: un cajero.

Información.

Es un conjunto ordenado de datos los cuales son manejados según la necesidad del usuario, para que un conjunto de datos pueda ser procesado eficientemente y pueda dar lugar a información, primero se debe guardar lógicamente en archivos. En general, la información en la base de datos estará integrada y además será compartida. Integrada significa que la base de datos puede considerarse como una unificación de varios archivos de datos, por lo demás distintos, y que elimina del todo o en parte cualquier redundancia entre ellos.

Compartida significa que los elementos individuales de información en la base de datos pueden compartirse entre varios usuarios distintos, en el sentido de que todos ellos pueden tener acceso al mismo elemento de información (y diferentes usuarios pueden utilizarlo para propósitos diferentes). Esta capacidad de compartir (en forma simultánea o no) se desprende en parte de la integración de la base de datos.

En la Figura 1.3 se muestra la forma como se integran los cuatro componentes principales de un sistema de base de datos: la información, el equipo, los programas y los usuarios.

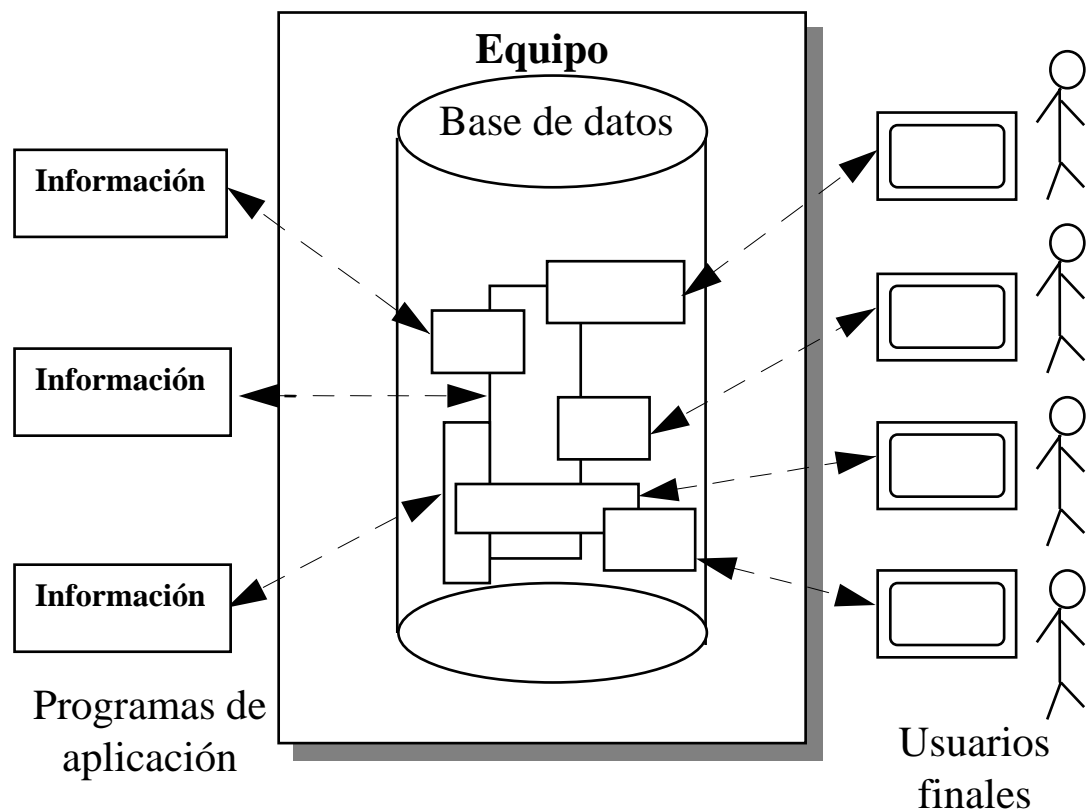


Figura 1.3. Esquema simplificado de un sistema de base de datos.

Objetivos de los sistemas de Bases de Datos.

Los objetivos principales de un sistema de base de datos es disminuir los siguientes aspectos:

- **Redundancia e inconsistencia de datos.**

Puesto que los archivos que mantienen almacenada la información son creados por diferentes tipos de programas de aplicación existe la posibilidad de que si no se controla detalladamente el almacenamiento, se pueda originar un duplicado de información, es decir que la misma información sea más de una vez en un dispositivo de almacenamiento. Esto aumenta los costos de almacenamiento y acceso a los datos, además de que puede originar la inconsistencia de los datos - es decir diversas copias de un mismo dato no concuerdan entre si -, por ejemplo: que se actualiza la dirección de un cliente en un archivo y que en otros archivos permanezca la anterior.

- **Dificultad para tener acceso a los datos.**

Un sistema de base de datos debe contemplar un entorno de datos que le facilite al usuario el manejo de los mismos. Supóngase un banco, y que uno de los gerentes necesita averiguar los nombres de todos los clientes que viven dentro del código postal 78733 de la ciudad. El gerente pide al departamento de procesamiento de datos que genere la lista correspondiente. Puesto que esta situación no fue prevista en el diseño del sistema, no existe ninguna aplicación de consulta que permita este tipo de solicitud, esto ocasiona una deficiencia del sistema.

- **Aislamiento de los datos.**

Puesto que los datos están repartidos en varios archivos, y estos no pueden tener diferentes formatos, es difícil escribir nuevos programas de aplicación para obtener los datos apropiados.

- **Anomalías del acceso concurrente.**

Para mejorar el funcionamiento global del sistema y obtener un tiempo de respuesta más rápido, muchos sistemas permiten que múltiples usuarios actualicen los datos simultáneamente. En un entorno así la interacción de actualizaciones concurrentes puede dar por resultado datos inconsistentes. Para prevenir esta posibilidad debe mantenerse alguna forma de supervisión en el sistema.

- **Problemas de seguridad.**

La información de toda empresa es importante, aunque unos datos lo son más que otros, por tal motivo se debe considerar el control de acceso a los mismos, no todos los usuarios pueden visualizar alguna información, por tal motivo para que un sistema de base de datos sea confiable debe mantener un grado de seguridad que garantice la autenticación y protección de los datos. En un banco por ejemplo, el personal de nóminas sólo necesita ver la parte de la base de datos que tiene información acerca de los distintos empleados del banco y no a otro tipo de información.

- **Problemas de integridad.**

Los valores de datos almacenados en la base de datos deben satisfacer cierto tipo de restricciones de consistencia. Estas restricciones se hacen cumplir en el sistema añadiendo códigos apropiados en los diversos programas de aplicación.

Abstracción de la Información

Un objetivo importante de un sistema de base de datos es proporcionar a los usuarios una visión abstracta de los datos, es decir, el sistema esconde ciertos detalles de cómo se almacenan y mantienen los datos. Sin embargo para que el sistema sea manejable, los datos se deben extraer eficientemente.

Existen diferentes niveles de abstracción para simplificar la interacción de los usuarios con el sistema; Interno, conceptual y externo, específicamente el de almacenamiento físico, el del usuario y el del programador.

- **Nivel de Vistas (Visión Externa):** Es el nivel más alto de abstracción, en él se escriben sólo vistas o subconjuntos de la bases de datos completa, esto con el fin de mostrar a los usuarios sólo las partes que necesitan. Es el nivel más cercano a los usuarios, es decir, es el que se ocupa de la forma como los usuarios individuales perciben los datos.
- **Nivel Conceptual (Visión Conceptual):** Es el siguiente nivel que se define, describe que datos son realmente almacenados en la base de datos, así como las relaciones que existen entre estos, describe la base de datos completa en términos de su estructura de diseño.

El nivel conceptual de abstracción lo usan los administradores de bases de datos, quienes deben decidir qué información se va a guardar en la base de datos. Se describe la base de datos a través de un número pequeño de estructuras. Es un “nivel de mediación” entre los otros dos. En el nivel conceptual la base de datos aparece como una colección de registros lógicos, sin descriptores de almacenamiento. En realidad los archivos conceptuales no existen físicamente. La transformación de registros conceptuales a registros físicos para el almacenamiento se lleva a cabo por el sistema y es transparente al usuario.

Consta de las siguientes definiciones:

Definición de los datos.

Se describen el tipo de datos y la longitud de campo todos los elementos direccionables en la base. Los elementos por definir incluyen artículos elementales (atributos), totales de datos y registros conceptuales (entidades).

Relaciones entre datos.

Se definen las relaciones entre datos para enlazar tipos de registros relacionados para el procesamiento de archivos múltiples.

- **Nivel Físico (Visión Interna):** Es el nivel más bajo, el más cercano al almacenamiento físico, es decir, es el que se ocupa de la forma como se almacenan físicamente los datos y describe cómo se almacenan realmente los datos. Se describe en detalle las estructuras complejas de datos utilizadas. Esta arquitectura permite implementar el concepto de Independencia de Datos, definida como la capacidad de modificar la definición del esquema en el primer nivel sin afectar el nivel intermedio.

Existen 2 niveles de independencia:

- **Independencia Física.**

Se define entre el nivel físico y el nivel conceptual. Capacidad de modificar el esquema físico sin alterar la estructura lógica de la bases de datos, es decir, el nivel conceptual.

- **Independencia Lógica.**

Se define entre el nivel externo y el nivel conceptual. Capacidad de modificar la estructura lógica de la bases de datos sin obligar a rescribir los programas de aplicación.

La interrelación entre estos tres niveles de abstracción se ilustra en la figura 1.4.

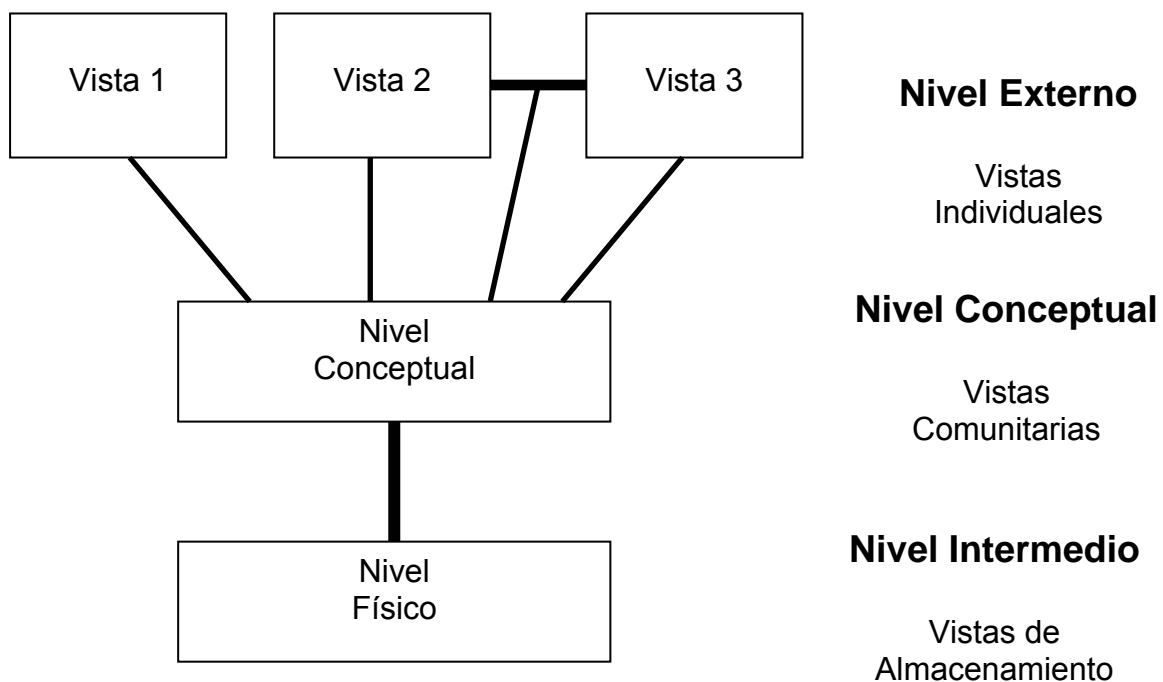


Figura 1.4 Arquitectura detallada del sistema.

1.2.5 Sistema Manejador de Base de Datos (SMBD)

Es el conjunto de programas que maneja todo acceso a la base de datos.

Un SMBD consiste de un conjunto de datos relacionados entre sí y un conjunto de herramientas de software (y/o hardware) para tener acceso a esos datos, el cual consiste de un conjunto de programas que son usados para definir, procesar y administrar la base de datos y sus aplicaciones.

Por lo que es un conjunto de equipos y programas que organiza los datos y proporciona los mecanismos utilizados para crear un archivo computarizado de bases de datos; añadir, borrar o cambiar datos dentro del archivo; cambiar el modo en que están almacenados los datos dentro de los archivos de una bases de datos, buscar en la bases de datos aquellos datos que cumplen ciertos criterios, etc.

El SMBD para organizaciones grandes requiere de un gran número de personas y altos gastos de equipos, programas y capacitación personal.

Conceptualmente, lo que sucede es lo siguiente:

1. Un usuario emite una solicitud de acceso utilizando algún lenguaje de manipulación de datos específico (DML).
2. El SMBD intercepta la solicitud y la interpreta.
3. El SMBD inspecciona en orden a) el esquema externo, b) la correspondencia externa / conceptual, c) el esquema conceptual, d) la correspondencia conceptual/interna y e) la definición de la estructura de almacenamiento.
4. El SMBD realiza las operaciones necesarias sobre la base de datos almacenada.

Se requieren programas de computadora para almacenar y consultar el contenido de una base de datos.

Un SMBD puede organizar, procesar y presentar los datos seleccionados de una base de datos. Esta capacidad permite a quienes toman decisiones rastrear, probar y consultar el contenido de la base de datos para extraer las respuestas a las preguntas no recurrentes y no previstas en informes regulares.

Los SMBD administrarán los datos almacenados y reunirán las partes necesarias de la bases de datos común para responder a las preguntas de quienes no sean programadores.

El Objetivo primordial de un SMBD es proporcionar un entorno para recuperar información y almacenar nueva información en la bases de datos, para lo cual debe proporcionar a los usuarios una visión abstracta de los datos. Es decir, los detalles de cómo se almacenan y se mantienen los datos, son transparentes para los usuarios.

Esto se debe a que muchos de ellos, no tienen experiencia en computadores, por ello se les esconde la complejidad a través de diversos niveles de abstracción, para simplificar la interacción con el sistema.

Las Funciones del SMDB son:

- **Definición de datos:** El SMDB debe ser capaz de aceptar definiciones de datos (esquema externo, el esquema conceptual, el esquema interno y todas las correspondencias asociadas) en versión fuente y convertirlas en la versión objeto apropiado. Dicho de otro modo, el SMDB debe incluir componentes procesadores de lenguajes para cada uno de los diversos lenguajes de definición de datos (DDL).
- **Manipulación de datos:** El SMDB debe ser capaz de atender las solicitudes del usuario para extraer, poner al día, datos que ya existen en la base de datos o para agregar en ella datos nuevos. Dicho de otro modo, el SMDB de datos debe incluir componentes procesadores de lenguajes para cada uno de los diversos lenguajes de manipulación de datos (DML).
- **Seguridad e integridad de los datos:** El SMDB debe supervisar las solicitudes de los usuarios y rechazar los intentos de violar las medidas de control y seguridad definidas por el DBA.
- **Recuperación y concurrencia de los datos:** El SMDB debe cuidar del cumplimiento de ciertos controles de recuperación y concurrencia.
- **Diccionario de Datos:** El SMDB debe incluir una función de diccionario de datos.
- **Desempeño:** El SMDB debe ejecutar todas las funciones recién identificadas en la forma más eficiente posible.

El poder de los SMDB, integra muchos conjuntos de datos anteriormente separados y proporcionan un conjunto completo de programas que sirven como interfaz entre uno o varios usuarios y sus diversas aplicaciones.

En un SMDB, los datos se pueden crear, borrar o cambiar en una base de datos integrada. El término integrada se refiere a la capacidad del SMDB de relacionar lógicamente un registro con otro. El usuario tiene acceso directo mediante instrucciones en el teclado. Un SMDB permite entonces:

1. **Independencia de los Datos:** La independencia de los datos es un objetivo primordial de los sistemas de bases de datos. Esta independencia puede definirse como la inmunidad de las aplicaciones ante los cambios en la estructura de almacenamiento y en la técnica de acceso, lo cual implica que las aplicaciones en cuestión no dependen de una estructura de almacenamiento o una técnica de acceso.

Todos los datos necesarios pueden ser almacenados en una base general. Si hay que hacer cualquier cambio a los datos pueden efectuarse sin necesidad de cambiar los programas que accedan datos. Esto es posible porque el SDBD proporciona dos aspectos de los datos. La visión física de una base de datos, se relaciona con la localización actual de los datos en el dispositivo de almacenamiento. La visión lógica representa los registros.

2. **Eliminación de la redundancia e incremento de la integridad de los datos:** Todos los datos relacionados se almacenan en un lugar, si un elemento de los datos debe ser cambiado sólo tiene que hacerse en un lugar.
3. **Datos integrados, a partir de otros archivos:** Un usuario puede recabar datos de cierto número de archivos de una base de datos y aplicar esos datos combinados, a reportes u otras aplicaciones, creando relaciones entre los registros. Realza la flexibilidad.
4. **Mayor seguridad, a través del manejo de acceso de datos:** La capacidad para negar el acceso a usuarios no autorizados, a datos restringidos, mejora enormemente la seguridad de los datos y pone a salvo la integridad.
5. **Normalización de reportes y consultas:** Un SDBD permite a un usuario que realizar reportes normalizados. Esto permite que el usuario formule preguntas breves.

Los componentes de procesamiento de consultas incluyen:

- Compilador DML.
- Precompilador DML.
- Intérprete DDL.
- Motor de Evaluación de Consultas.

Los componentes de gestión de almacenamiento proporcionan la interfaz entre los datos de bajo nivel almacenados en la base de datos y los programas de aplicación y envío de consultas al sistema.

El gestor de almacenamiento incluye:

- Gestor de Transacciones.
- Gestor de Archivos.
- Gestor de Memoria Intermedia.

Además, se necesitan varias estructuras de datos como parte de la implementación física del sistema:

- **Archivos de Datos:** Almacenan la base de datos.
- **Diccionario de Datos:** Almacena metadatos (datos acerca de los datos).
- **Índices:** Proporcionan acceso rápido a elementos de datos que tienen valores particulares.
- **Datos estadísticos:** Almacenan información estadística sobre los datos en la base de datos. El procesador de consultas utiliza esta información para seleccionar las formas eficientes para ejecutar una consulta.

A continuación se presentan las ventajas y limitaciones de los Sistemas Manejadores de Base de Datos, en la siguiente tabla 1.3.

VENTAJAS	LIMITACIONES
Mejor integración y menos duplicidad de los datos que se originan en los diferentes puntos.	Se necesitan hardware y software más complejos y caros.
Menos errores cuando varios registros pueden actualizarse en forma simultánea.	Fallas del hardware o del software pueden ocasionar la destrucción de información vital de la bases de datos.
Facilitan el almacenamiento de grandes cantidades de información.	Pueden requerirse un largo período de conversión, elevados gastos de capacitación y habilidades mayores en quienes son responsables del Sistema de bases de datos.
Facilitan la organización y reorganización de la información.	
Facilitan la recuperación rápida y flexible de la información.	
Ahorrar en el costo de desarrollo de nuevas aplicaciones, así como en los costos de entrada de los datos y su almacenamiento.	

Tabla 1.3 Ventajas y Limitaciones de los Sistemas Manejadores de Base de Datos

1.2.6 Administrador de Bases de Datos (DBA)

El Administrador de bases de datos (DBA: Databas Administrador) es la persona o equipo de personas profesionales responsables del control y manejo del sistema de base de datos, generalmente tiene (n) experiencia en DBMS, diseño de bases de datos, sistemas operativos, comunicación de datos, hardware y programación.

Los sistemas de base de datos se diseñan para manejar grandes cantidades de información, la manipulación de los datos involucra tanto la definición de estructuras para el almacenamiento de la información como la provisión de mecanismos para la manipulación de la información, además un sistema de base de datos debe de tener implementados mecanismos de seguridad que garanticen la integridad de la información, a pesar de caídas del sistema o intentos de accesos no autorizados.

Un objetivo principal de un sistema de base de datos es proporcionar a los usuarios finales una visión abstracta de los datos, esto se logra procurando no dejar a la vista ciertos detalles de como se respecto a la forma de almacenamiento y mantienen mantenimiento de los datos. Es designado usualmente por la dirección de una compañía y provisto de un personal que trabaje con los usuarios para crear, mantener y salvaguardar los datos en la bases de datos. Para un SMBD basado en micro, recae en el individuo, el DBA. Es conveniente definir el administrador de datos y el administrador de base de datos. Así, es menester resaltar que el Administrador de Datos, es la persona que toma las decisiones estratégicas y de políticas con respecto a la información de la empresa.

En tanto que el Administrador de bases de datos (DBA: database administrator), es la persona que proporciona el apoyo técnico necesario para poner en práctica las decisiones tomadas por el administrador de datos.

Funciones del DBA:

1. Definición del esquema de la bases de datos. Desarrollo y mantenimiento de un diccionario de datos (DD). EL DD define el significado de cada elemento de datos (cada campo) en la bases de datos; esto incluye los nombres de los datos (nombres de los campos), tipos de datos, tamaño del campo y cualquier interrelación entre elementos de datos.
2. Modificación del esquema y de la organización física de la bases de datos.
3. Definición de la estructura de almacenamiento y del método de acceso.
4. Mantenimiento de un control de transacciones: Un control de transacciones contiene una auditoria completa de toda la actividad de una base de datos en un tiempo. El control ayuda al respaldo, en el caso de que un dato quedará inutilizado o destruido, el control lleva un registro de todos los cambios, que sirve para restaurar la base de datos a su condición original.

5. Definición de las verificaciones de seguridad e integridad: esto puede considerarse parte del esquema conceptual. Concesión de autorización para el acceso a los datos: La seguridad incluir decidir que acceso se permite a un campo o archivo. Incluye proporcionar medios para la recuperación eficiente si ocurre un desastre y se pierde la información en la bases de datos.
6. Definición de los procedimientos de respaldo y recuperación. El DBA debe definir y poner en práctica un plan de recuperación adecuado que incluya, por ejemplo, el realizar backups periódicos de la base de datos como respaldos y procedimientos para cargar la base de datos a partir del respaldo más reciente que se tenga. El DBA asegura que se ejecute el respaldo apropiado de la bases de datos. El respaldo se refiere a las copias y al registro de todos los cambios que han sido hechos a la bases de datos. Si sucede algo que dañe o destruya la base de datos; esta puede ser reconstruida (recuperada) usando el respaldo.
7. Supervisión del desempeño y respuesta a los cambios en los requerimientos. Podría ser necesario reorganizar la base de datos en forma periódica con el fin de garantizar que los niveles de desempeño sigan siendo aceptables. Realizar cambios a nivel físico y actualizar la correspondencia interna - conceptual.
8. Especificación de las restricciones de integridad, cada vez que hay actualización en el sistema.

1.2.7 Métodos de Organización de las Bases de Datos

Una característica fundamental del enfoque de bases de datos es que proporciona cierto nivel de abstracción de los datos al ocultar detalles de almacenamiento que la mayoría de los usuarios no necesitan conocer.

Los modelos de datos son el principal instrumento para ofrecer dicha abstracción.

Un modelo de datos es un conjunto de conceptos que pueden servir para describir la estructura de la base de datos. Es decir, un modelo de datos no es más que una colección de herramientas conceptuales que se utilizan para describir los datos, las relaciones existentes entre ellos, la semántica asociada a los mismos y las restricciones de consistencia.

Los modelos de datos se dividen en 3 grupos:

- A. Modelos Lógicos basados en Registros
- B. Modelos Lógicos basados en Objetos
- C. Modelos Físicos de Datos

Definición de Modelo de Datos

“Es un conjunto de conceptos, reglas y convenciones que nos permiten describir los datos del universo del discurso, constituyendo una herramienta que facilita la interpretación de nuestro universo del discurso y su representación en forma de datos en nuestro sistema de información.”¹⁰

Modelo de datos y lenguaje de datos.– Modelos es en lo que se basan los lenguajes de datos.

A. Modelos Lógicos Basados en Registros

Los tres modelos de datos más ampliamente aceptados son:

- Modelo Relacional
- Modelo de Red
- Modelo Jerárquico

Modelo Relacional

En este modelo se representan los datos y las relaciones entre estos, a través de una colección de tablas, en las cuales los renglones (tuplas) equivalen a los cada uno de los registros que contendrá la base de datos y las columnas corresponden a las características (atributos) de cada registro localizado en la tupla.

Existen dos formas de representar las relaciones entre las entidades en este modelo; pero para ello necesitamos definir que es una llave primaria: Es un atributo el cual definimos como atributo principal, es una forma única de identificar a una entidad. Las formas de representar las relaciones en este modelo son:

- i. Haciendo una tabla que contenga cada una de las llaves primarias de las entidades involucradas en la relación.

Tomando en cuenta que la llave primaria del empleado es su RFC, y la llave primaria del artículo es la Clave.

- ii. Incluyendo en alguna de las tablas de las entidades involucradas, la llave de la otra tabla.

Este modelo se volverá a ver mas adelante.

¹⁰ Elmasri, Ramez y Navathe, Shamkant, op. cit., p. 22.

Modelo de Red

Este modelo representa los datos mediante colecciones de registros y sus relaciones se representan por medio de ligas o enlaces, los cuales pueden verse como punteros. Los registros se organizan en un conjunto de gráficas arbitrarias.

Modelo Jerárquico

Es similar al modelo de red en cuanto a las relaciones y datos, ya que estos se representan por medio de registros y sus ligas. La diferencia radica en que están organizados por conjuntos de árboles en lugar de gráficas arbitrarias.

B. Modelos Lógicos Basados en Objetos.

Se usan para describir datos en los niveles conceptual y de visión, es decir, con este modelo representamos los datos de tal forma como nosotros los captamos en el mundo real, tienen una capacidad de estructuración bastante flexible y permiten especificar restricciones de datos explícitamente. Existen diferentes modelos de este tipo, pero el más utilizado por su sencillez y eficiencia es el modelo Entidad-Relación.



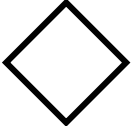

Modelo Entidad-Relación.

Denominado por sus siglas como: E-R; El modelo E-R se basa en una percepción del mundo real, la cual está formada por objetos básicos llamados entidades y las relaciones entre estos objetos así como las características de estos objetos llamados atributos.

- **Tangibles:** Son todos aquellos objetos físicos que podemos ver, tocar o sentir.
- **Intangibles:** Todos aquellos eventos u objetos conceptuales que no podemos ver, aun sabiendo que existen, por ejemplo: la entidad materia, sabemos que existe, sin embargo, no la podemos visualizar o tocar.

Para la representación de un modelo E-R gráficamente, se emplean símbolos, los cuales son: Diagramas Entidad-Relación (ER).

La estructura lógica general de una base de datos puede expresarse en forma gráfica por medio de un diagrama E-R, que se integra con las siguientes componentes:

Símbolo	Representa
	Rectángulos: Representan conjuntos de entidades.
	Elipses: Representan atributos.
	Rombos: Representan conjuntos de relaciones.
	Líneas: Enlazan atributos a entidades y entidades a relaciones.

Estructura

i. Entidades y conjuntos de entidades

“**Entidad** representa un objeto o concepto del mundo real”.¹¹

Las entidades pueden ser identificadas unívocamente en una situación a modelar. Así una entidad corresponde a una categorización de objetos de la situación a modelar. También un tipo entidad puede ser visto como una agregación de atributos. **Conjunto de entidades:** Es un grupo de entidades del mismo tipo.

ii. Relaciones y conjuntos de relaciones.

Una relación es una asociación entre varias entidades. En términos de abstracción un tipo de relación corresponde a una agregación de uno o más tipos de entidades. Las relaciones no tienen existencia propia, ya que dependen de entidades. Un **conjunto de relaciones** es un grupo de relaciones del mismo tipo.

Se puede decir entonces que es posible tener más de un conjunto relación entre dos conjuntos entidades.

Un conjunto asociación puede ser n-ario, es s decir, entre n conjuntos entidades.

¹¹ Ibidem, p. 23.

iii. Atributo, valor y conjunto de valores.

Un atributo puede ser definido formalmente como una función que transforma un conjunto de entidades o relaciones.

Se puede decir que los atributos son interpretaciones de conjunto de valores en un contexto de conjunto E-R.

Por lo que una entidad relación puede ser representada por un conjunto de atributos.

Limitantes de Mapeo.

Un esquema E-R empresarial puede definir ciertas limitantes con las que deben cumplir los datos contenidos en la bases de datos.

Una limitante importante es la cardinalidad de mapeo que expresa el número de entidades con las que puede asociarse otra entidad mediante una relación.

La cardinalidad de asignación expresan el número de entidades a las que puede asociarse otra entidad a través de un conjunto de relaciones.

Para un conjunto binario de relaciones R entre los conjuntos de entidades A y B, la cardinalidad de mapeo puede ser:

- Una a una (1:1)
- Una a muchas (1:N)
- Muchas a muchas (M:N).

Otra clase de limitantes lo constituye la dependencia de existencia.

Refiriéndonos a las mismas entidades A y B, decimos que si la entidad A depende de la existencia de la entidad B, entonces A es dependiente de existencia por B, si eliminamos a B tendríamos que eliminar por consecuente la entidad A, en este caso B es la entidad dominante y A es la entidad subordinada.

Tipos de relaciones:

- **Relación uno a uno.**

Se presenta cuando existe una relación como su nombre lo indica uno a uno, denominado también relación de matrimonio. Una entidad del tipo A solo se puede relacionar con una entidad del tipo B, y viceversa.

- **Relación uno a muchos.**

Significa que una entidad del tipo A puede relacionarse con cualquier cantidad de entidades del tipo B, y una entidad del tipo B solo puede estar relacionada con una entidad del tipo A.

- **Muchos a uno.**

Indica que una entidad del tipo B puede relacionarse con cualquier cantidad de entidades del tipo A, mientras que cada entidad del tipo A solo puede relacionarse con solo una entidad del tipo B.

- **Muchas a muchas.**

Establece que cualquier cantidad de entidades del tipo A pueden estar relacionados con cualquier cantidad de entidades del tipo B.

A los tipos de relaciones antes descritos, también se le conoce como cardinalidad.

Reducción de diagramas E-R a tablas

Un diagrama E-R, puede ser representado también a través de una colección de tablas. Para cada una de las entidades y relaciones existe una tabla única a la que se le asigna como nombre el del conjunto de entidades y de las relaciones respectivamente, cada tabla tiene un número de columnas que son definidas por la cantidad de atributos y las cuales tienen el nombre del atributo.

Generalización.

Es el resultado de la unión de 2 o más conjuntos de entidades (de bajo nivel) para producir un conjunto de entidades de más alto nivel. La generalización se usa para hacer resaltar los parecidos entre tipos de entidades de nivel más bajo y ocultar sus diferencias.

La generalización consiste en identificar todos aquellos atributos iguales de un conjunto de entidades para formar una entidad(es) global(es) con dichos atributos semejantes, dicha entidad(es) global(es) quedara a un nivel más alto al de las entidades origen.

La Generalización trata de eliminar la redundancia (repetición) de atributos, al englobar los atributos semejantes. La entidad(es) de bajo nivel cuentan (heredan) todos los atributos correspondientes.

Especialización:

Es el resultado de tomar un subconjunto de entidades de alto nivel para formar un conjunto de entidades de más bajo nivel.

- En la generalización cada entidad de alto nivel debe ser también una entidad de bajo nivel. La especialización no tiene este limitante.
- Se representa por medio de un triángulo denominado con la etiqueta "ISA", se distingue de la generalización por el grosor de las líneas que conectan al triángulo con las entidades.
- La especialización denota la diferencia entre los conjuntos de entidades de alto y bajo nivel.

Agregación.

La agregación surge de la limitación que existe en el modelado de E-R, al no permitir expresar las relaciones entre relaciones de un modelo E-R en el caso de que una relación X se quiera unir con una entidad cualquiera para formar otra relación.

La generalización consiste en agrupar por medio de un rectángulo a la relación (representada por un rombo) junto con las entidades y atributos involucrados en ella, para formar un grupo que es considerado una entidad y ahora sí podemos relacionarla con otra entidad.

C. Modelos Físicos de Datos

Se usan para describir a los datos en el nivel más bajo, aunque existen muy pocos modelos de este tipo, básicamente capturan aspectos de la implementación de los sistemas de base de datos. Existen dos clasificaciones de este tipo que son:

- Modelo unificador
- Memoria de elementos.

1.2.8 Instancias y Esquemas

Con el paso del tiempo la información que se va acumulando y desechando en la base de datos, ocasiona que está cambie.

Instancia: Al estado que presenta una base de datos en un tiempo dado.

Veámoslo como una fotografía que tomamos de la base de datos en un tiempo t , después de que transcurre el tiempo t la base de datos ya no es la misma.

Esquema: Es la descripción lógica de la base de datos, proporciona los nombres de las entidades y sus atributos especificando las relaciones que existen entre ellos. Es un banco en el que se inscriben los valores que irán formando cada uno de los atributos. El esquema no cambia los que varían son los datos y con esto tenemos una nueva instancia.

1.2.9 Independencia de los Datos

Se refiere a la protección contra los programas de aplicación que puedan originar modificaciones cuando se altera la organización física o lógica de la base de datos.

Existen 2 niveles de independencia de datos.

- **Independencia física de datos.**

Es la capacidad de modificar el esquema físico sin provocar que se vuelvan a escribir los programas de aplicación.

- **Independencia lógica de datos.**

Capacidad de modificar el esquema conceptual sin provocar que se vuelvan a escribir los programas de aplicación.

1.2.10 Claves (Llaves)

Son atributos que identifican una entidad dentro de un conjunto de entidades. La distinción de una entidad entre otra se debe a sus atributos, lo cual lo hacen único.

Tipos de Llaves

- **Superclave.**

Es un conjunto de uno o más atributos que, tomados colectivamente permiten identificar de forma única una entidad en el conjunto de entidades.

- **Clave candidata.**

Es una superclave para la cual ningún subconjunto es superclave, excepto el mismo. Conjunto no vacío de atributos que identifican unívocamente y mínima cada tupla. Atributo o atributos que pueden distinguir de forma unívoca una tupla dentro de una tabla.

Puede haber varias claves candidatas para distinguir una misma entidad. Se elegirá como clave candidata aquel atributo que posea un dominio en el que se tenga valores únicos. Si esto no es posible, entonces usaremos como clave candidata la combinación de varios atributos, de manera que esta combinación sí sea única.

- **Clave primaria (Llave Primaria).**

Es la clave candidata escogida por el diseñador. Atributo o conjunto de atributos que permiten identificar en forma única una tupla en la tabla (una entidad en un conjunto de entidades) y ningún subconjunto de ella posee esta propiedad.

Una llave primaria es aquel atributo el cual consideramos clave para la identificación de los demás atributos que describen a la entidad. Claro que puede haber más de un atributo que pueda identificarse como llave primaria en este caso se selecciona la que consideremos más importante, los demás atributos son denominados llaves secundarias.

Una clave o llave primaria es indicada gráficamente en el modelo E-R con una línea debajo del nombre del atributo.

Una clave (primaria, candidata y superclave) es una propiedad del conjunto de entidades más que de las entidades individuales. Cualesquiera dos entidades en el conjunto no pueden tener el mismo valor en sus atributos clave al mismo tiempo. La designación de una clave representa una ligadura en el desarrollo del mundo real que se modela.

- **Llave foránea.**

También llamada clave ajena. Es el conjunto de atributos de la tabla cuyos valores han de coincidir con los de la clave primaria de otra tabla. (Clave ajena y primaria debe estar definida sobre los mismos dominios). Se trata de un atributo que es clave principal en otra tabla.

- **Claves alternativas.**

Son las claves candidatas que no han sido escogidas.

1.2.11 Bases de Datos Relacionales

- Los sistemas relacionales operan conceptualmente sobre archivos o Tablas de datos y no sobre los datos individuales contenidos en el archivo.
- Las tablas permiten representar la información de forma mas compacta.
- Es posible acceder a la información contenida en dos o más tablas simultáneamente.

La ventaja del modelo relacional es que los datos se almacenan, al menos conceptualmente, de un modo en que los usuarios entienden con mayor facilidad. Los datos se almacenan como tablas y las relaciones entre las filas y las tablas son visibles en los datos.

Este enfoque permite a los usuarios obtener información de la base de datos sin asistencia de sistemas profesionales de administración de información.

Las características más importantes de los modelos relacionales son:

- Es importante saber que las entradas en la tabla tienen un solo valor (son atómicos); no se admiten valores múltiples, por lo tanto la intersección de un renglón con una columna tiene un solo valor, nunca un conjunto de valores.
- Todas las entradas de cualquier columna son de un solo tipo. Por ejemplo, una columna puede contener nombres de clientes, y en otra puede tener fechas de nacimiento. Cada columna posee un nombre único, el orden de las columnas no es de importancia para la tabla, las columnas de una tabla se conocen como atributos. Cada atributo tiene un dominio, que es una descripción física y lógica de valores permitidos.

No existen 2 filas en la tabla que sean idénticas.

La información en las bases de datos son representados como datos explícitos, no existen apuntadores o ligas entre las tablas.

En el enfoque relacional es sustancialmente distinto de otros enfoques en términos de sus estructuras lógicas y del modo de las operaciones de entrada/salida. En el enfoque relacional, los datos se organizan en tablas llamadas relaciones, cada una de las cuales se implanta como un archivo.

En terminología relacional una fila en una relación representa un registro o una entidad; Cada columna en una relación representa un campo o un atributo. Así, una relación se compone de una colección de entidades(o registros) cuyos propietarios están descritos por cierto número de atributos predeterminados implantados como campos.

Tabla relacional: Es una tabla que debe cumplir las siguientes características:

- Cada fila debe ser única
- Cada columna debe ser única
- Los valores de las columnas deben pertenecer al dominio de cada atributo
- Debe tener un solo tipo de fila, cuyo formato está definido por el esquema de la tabla o relación.
- El valor de la columna para cada fila debe ser único

Estructura de la Base de Datos Relacionales

La arquitectura relacional se puede expresar en términos de tres niveles de abstracción: nivel interno, conceptual y de visión.

La arquitectura relacional consta de los siguientes componentes:

1. Modelo relacional de datos
2. Submodelo de datos
3. Esquema de almacenamiento
4. Sublenguaje de datos.

1.2.12 Modificación de la Base de Datos

El lenguaje SQL, cuenta con módulos DDL, para la definición de datos que nos permite crear o modificar la estructura de las tablas.

Las instrucciones para realizar estas operaciones son:

- CREATE TABLE: Nos permite crear una tabla de datos vacía.
- INSERT: Permite almacenar registros en una tabla creada.
- UPDATE: Permite modificar datos de registros almacenados en la tabla.
- DELETE: Borra un registro entero o grupo de registros de una tabla.
- CREATE INDEX: Crea un índice que nos puede auxiliar para las consultas.
- DROP TABLE: Permite borrar una tabla.
- DROP INDEX: Borra el índice indicado.

1.2.13 Diseño de Base de Datos Relacionales

Una base de datos es un conjunto de datos estructurados, almacenados en algún soporte de almacenamiento de datos y se puede acceder a ella desde uno o varios programas.

Antes de diseñar una base de datos se debe establecer un proceso partiendo del mundo real, de manera que sea posible plasmar éste mediante una serie de datos.

La imagen que se obtiene del mundo real se denomina modelo conceptual y consiste en una serie de elementos que definen perfectamente lo que se quiere plasmar del mundo real en la base de datos.

- Planificación del tipo de información a almacenar:
 - Información disponible.
 - Información que necesitamos.
- Esquematizar sobre papel el problema.
- Considerar los datos a gestionar y estimar el espacio de memoria que necesitan.
- Los dos aspectos mas importantes a la hora del diseño de las Tablas son:
 - Nombre del campo
 - Tipo del campo
 - Anchura del campo
 - Campos
 - Datos

Caracteres (texto), valores numéricos, fechas, informaciones lógicas, imágenes, multimedia.

Fases del diseño de una Base de Datos

- Definición de los datos (análisis de los datos existentes).
- Refinamiento de los datos (depuración de los datos necesarios).
- Establecer relaciones entre los campos.

1.2.14 Peligros en el Diseño de Base de Datos Relacionales

Uno de los retos en el diseño de la base de datos es el de obtener una estructura estable y lógica tal que:

- El sistema de base de datos no sufra de anomalías de almacenamiento.
- El modelo lógico pueda modificarse fácilmente para admitir nuevos requerimientos.

Una base de datos implantada sobre un modelo bien diseñado tiene mayor esperanza de vida aun en un ambiente dinámico, que una base de datos con un diseño pobre. En promedio, una base de datos experimenta una reorganización general cada seis años, dependiendo de lo dinámico de los requerimientos de los usuarios. Una base de datos bien diseñada tendrá un buen desempeño aunque aumente su tamaño, y será lo suficientemente flexible para incorporar nuevos requerimientos o características adicionales.

Existen diversos riesgos en el diseño de las bases de datos relacionales que afecten la funcionalidad de la misma, los riesgos generalmente son la redundancia de información y la inconsistencia de datos.

La normalización es el proceso de simplificar la relación entre los campos de un registro. Por medio de la normalización un conjunto de datos en un registro se reemplaza por varios registros que son más simples y predecibles y, por lo tanto, más manejables.

La normalización se lleva a cabo por cuatro razones:

1. Estructurar los datos de forma que se puedan representar las relaciones pertinentes entre los datos.
2. Permitir la recuperación sencilla de los datos en respuesta a las solicitudes de consultas y reportes.
3. Simplificar el mantenimiento de los datos actualizándolos, insertándolos y borrándolos.
4. Reducir la necesidad de reestructurar o reorganizar los datos cuando surjan nuevas aplicaciones.

En términos más sencillos la normalización trata de simplificar el diseño de una base de datos, esto a través de la búsqueda de la mejor estructuración que pueda utilizarse con las entidades involucradas en ella.

Pasos de la Normalización:

- Descomponer todos los grupos de datos en registros bidimensionales.
- Eliminar todas las relaciones en la que los datos no dependan completamente de la llave primaria del registro.
- Eliminar todas las relaciones que contengan dependencias transitivas.
- La teoría de normalización tiene como fundamento el concepto de formas normales; se dice que una relación está en una determinada forma normal si satisface un conjunto de restricciones.

1.2.15 Normalización

Las reglas de normalización fueron definidas por Codd (1970). El punto fundamental en el proceso es que dada una relación que posee ciertas propiedades indeseables, las reglas de normalización permiten reconocer tales casos y muestran como esa relación puede ser descompuesta en una forma más deseable.

La Normalización se define como un método de diseño ascendente basado en el concepto de formas normales, así se dice que una relación está en una forma normal particular si cumple con un conjunto de restricciones.

A medida que se incrementan las formas normales se incrementa el número de restricciones que debe cumplir esa relación.¹² La normalización se encarga de obtener los datos agrupados en distintas tablas siguiendo una serie de pasos, de tal manera que los datos obtenidos tienen una estructura óptima para su implementación, gestión y explotación desde distintas aplicaciones futuras. Una de las ventajas principales que se obtiene al realizar la normalización es que la información no estará duplicada innecesariamente dentro de las estructuras: habrá mínima redundancia. Al modelar una base de datos, desearemos evitar puntos que crean confusión, duplicación de la información y por ende, un mal funcionamiento y exploración de la información. Entre las propiedades indeseables en un diseño de bases de datos tenemos:

- Redundancia en la información.
- Incapacidad de representar cierta información.
- Registrar información que no sea identificable.

Formas Normales

Son las técnicas para prevenir las anomalías en las tablas. Dependiendo de su estructura, una tabla puede estar en primera forma normal, segunda forma normal o en cualquier otra. El Objetivo es obtener la forma normal mayor posible. La teoría de normalización consiste en obtener esquemas relacionales que cumplan unas determinadas condiciones y se centra en las determinadas Formas normales. Se dice que un esquema de relación está en una determinada forma normal si satisface un conjunto determinado de restricciones.

En la figura 1.5 se muestra la relación entre las formas normales.

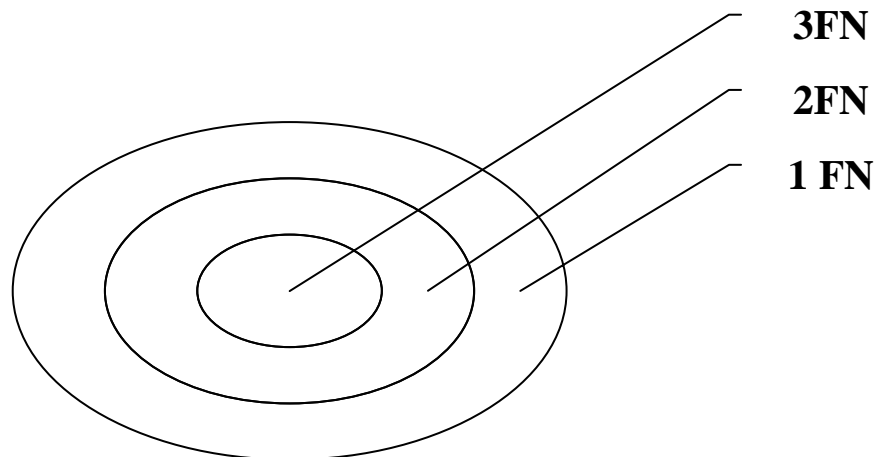


Figura 1.5 Relación entre las formas normales

¹² Hansen, Gary y Hansen, James, *Diseño y Administración de Base de Datos*, Tercera Edición, Prentice-Hall Hispanoamericana, México 2000, p. 12.

Primera forma normal

Abreviada como 1FN, una relación R se encuentra en 1FN si y solo si por cada renglón columna contiene valores atómicos, se considera que una relación se encuentra en la primera forma normal cuando cumple lo siguiente:

- Las celdas de las tablas poseen valores simples y no se permiten grupos ni arreglos repetidos como valores, es decir, contienen un solo valor por cada celda.
- Todos los ingresos en cualquier columna (atributo) deben ser del mismo tipo.
- Cada columna debe tener un nombre único, el orden de las columnas en la tabla no es importante.
- Dos filas o renglones de una misma tabla no deben ser idénticas, aunque el orden de las filas no es importante.

Por lo general la mayoría de las relaciones cumplen con estas características, así que podemos decir que la mayoría de las relaciones se encuentran en la primera forma normal.

Una relación está en primera forma normal (1FN) si y sólo si todos los dominios son atómicos. Un dominio es atómico si los elementos del dominio son indivisibles. Es decir, no tenemos grupos de repetición o un conjunto de valores asociados repetidos asociados a una misma tupla.

Segunda forma normal.

Abreviada como 2FN, Para definir la segunda forma normal requerimos saber que es una dependencia funcional: Consiste en edificar que atributos dependen de otro(s) atributo(s). Como se ve en la Figura 1.6

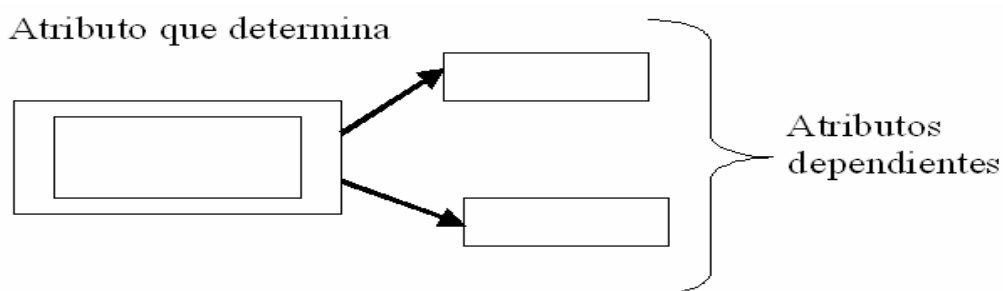


Figura 1.6 Segunda forma normal

Una relación R está en 2FN si y solo si está en 1FN y los atributos no primos dependen funcionalmente de la llave primaria.

Una relación se encuentra en segunda forma normal, cuando cumple con las reglas de la primera forma normal y todos sus atributos que no son claves (llaves) dependen por completo de la clave. De acuerdo con esta definición, cada tabla que tiene un atributo único como clave, esta en segunda forma normal. La segunda forma normal se representa por dependencias funcionales.

Tercera forma normal.

Abreviada como 3FN. Una relación R está en 3FN si y solo si esta en 2FN y todos sus atributos no primos dependen no transitivamente de la llave primaria. Consiste en eliminar la dependencia transitiva que queda en una segunda forma normal, en pocas palabras una relación esta en tercera forma normal si está en segunda forma normal y no existen dependencias transitivas entre los atributos, nos referimos a dependencias transitivas cuando existe más de una forma de llegar a referencias a un atributo de una relación.

En la tabla 1.4 se muestra la relación entre las formas normales, así como el enfoque de dependencias y el enfoque intuitivo de las formas normales.

Relación entre formas normales.	Enfoque de Dependencias.	Normalización. Enfoque intuitivo.
	Dependencias funcionales 2FN, 3Fn	2FN Una relación está en 2FN si además de estar en 1FN todos los atributos que no forman parte de ninguna clave candidata suministran información acerca de la clave completa.
Una relación que esta en 1FN no tiene porque estar en 2FN.		Toda relación cuya clave esta formada por un solo atributo está en 2FN.
		3FN Si además de estar en 2FN, los atributos que no forman parte de ninguna clave candidata facilitan información solo acerca de las claves y no acerca de otros atributos.

Tabla 1.4. Relación, Enfoque de Dependencias e Intuitivo de las Formas Normales

CAPÍTULO II

ARQUITECTURA DE SEGURIDAD.

2. Arquitectura de Seguridad

Introducción

Las bases de datos incluidas hoy en día en los sistemas de información de cualquier organización, nacen con el fin de resolver las limitaciones que en algunos casos presentan los ficheros para el almacenamiento de información.

En los entornos de bases de datos, las diferentes aplicaciones y usuarios utilizan un único conjunto de datos integrado a través de un Sistema de Gestión de Bases de Datos (SGBD).

De esta forma se pueden resolver problemas como duplicación de información, inconsistencia de los datos y dependencia entre programa y estructura de datos.

Por otra parte, la agrupación de datos pertenecientes a distintos usuarios y catalogados en niveles de seguridad diferentes aumenta los riesgos en cuanto a la seguridad de los datos.

El sistema de bases de datos debe controlar si está autorizada cada operación sobre la bases de datos.

Para conseguir un entorno de bases de datos seguro se han de identificar las amenazas reales, elegir las políticas de seguridad adecuadas que establezcan mecanismos de prevención así como métodos que comprueben que no se han producido accesos ilícitos.

Las bases de datos que contienen información sobre datos financieros son un blanco tentador para los ladrones. Así mismo, las bases de datos que contienen información referente a las operaciones de una compañía pueden ser de interés para sus competidores sin escrúpulos.

También existen bases de datos que por su contenido personal deben ser mantenidas en estricto secreto, por lo tanto, nuestro objetivo es resguardar la base de datos de USECAD ya que contiene información valiosa y confidencial de académicos y alumnos.

Por otra parte, la pérdida de operatividad del SGBD o de información de sus bases de datos es un problema que afecta muy seriamente al funcionamiento de cualquier organización y necesita de la implementación de la seguridad para garantizar su operatividad.

2.1. Definición

2.1.1. Seguridad en Cómputo

¿Qué es seguridad en Cómputo?

Términos como "seguridad", "protección", "privacidad", pueden tener más de un significado, dependiendo de quién lo aplica, y en que ámbitos. Incluso los profesionales que trabajan en el área de seguridad no siempre coinciden en lo que estos términos significan.

Una definición bastante práctica de seguridad es: Un sistema, es seguro si se puede confiar en él y su software se comporta como los usuarios esperan que lo haga.

Retomando la definición anterior sabemos que la seguridad tiene varios ámbitos, ya que nosotros nos enfocaremos a la información a continuación daremos su definición.

La información se define como el conocimiento obtenido a partir de la investigación, el estudio o instrucción, inteligencia, noticias, hechos, datos, una señal o carácter (como en un sistema de comunicación o computadora) representando datos, algo (como un mensaje datos experimentales o una imagen) que justifique el cambio en una construcción (como un plan o una teoría) que representa la experiencia física o mental u otra construcción.

2.1.2. Seguridad en Bases de Datos

La seguridad de la información en las bases de datos incluye cuatro principales servicios: integridad, autenticidad, confidencialidad y disponibilidad.

El objetivo es proteger la Base de Datos contra accesos no autorizados.

La seguridad de los datos se refiere a la protección de estos contra el acceso por parte de las personas no autorizadas y contra su indebida destrucción o alteración.

Seguridad significa el prevenir y/o detectar la publicación de datos de forma no autorizada de información. En general seguridad se refiere a la protección de los datos en diferentes ambientes, tanto en ambientes militares como en ambientes comerciales.

Otro termino importante en el estudio de la seguridad de la información es el de la privacidad de los datos que se refiere a la información individual de cada individuo, grupo o institución para determinar cuando o que información le concierne para su propio propósito, y está a su vez puede ser almacenada o liberada para otras personas o entidades.

Privacidad se refiere a los datos de las personas que están protegidos por las leyes o reglas dependiendo del país donde se encuentre.

En los ambientes comerciales se encuentra información secreta estrictamente, acompañada por sistemas de seguridad (llamados policías de seguridad) que aseguran que la información denominada como crítica por la organización tendrá una alta seguridad en donde los empleados no tendrán acceso a información que no les corresponde.

- La información en un sistema informático reside, fundamentalmente, en bases de datos cuya seguridad es, por consiguiente, de vital importancia.
- Uno de los filtros de seguridad de la base de datos lo constituye, evidentemente el sistema operativo.
- No obstante, la seguridad que debe proporcionar la base de datos tiene algunas características diferenciadas:
 - Hay muchos más objetos a proteger.
 - El promedio de tiempo de vida de los objetos es mayor.
 - La granularidad del control es mayor.
 - Los objetos son estructuras lógicas complejas.
 - La seguridad está relacionada con la semántica de los datos, no con sus características físicas,

Dada la complejidad de los problemas anteriores, es el propio Sistema de Gestión de Bases de Datos (SGBD) el que proporciona la seguridad de éstas.

- Un SGBD debe mantener los cuatro criterios básicos:
 - Confidencialidad.
 - Integridad.
 - Disponibilidad.
 - Autenticidad

La seguridad en bases de datos se implementa mediante mecanismos de:

- Identificación y autenticación.
- Control de acceso a los objetos (datos y recursos).
- Registro de Auditoría.
- Protección criptográfica de alguno de los datos.

- Las posibles vulnerabilidades en la bases de datos son:
 - Los Ataques.
- La prevención frente a los ataques pasa por mecanismos de identificación, autenticación y control de acceso.

2.1.3. Seguridad en la Tecnología de la Información y las Comunicaciones

Seguridad en las TIC

- En los últimos tiempos se han realizado grandes avances en las tecnologías de la información y las comunicaciones (TIC).
- Los sistemas informáticos se han introducido de forma generalizada en el comercio, banca, industria, administración, defensa, investigación.
- La aparición y expansión de Internet, juntamente con los sistemas informáticos, han hecho posible la realización de tareas impensables hace unos pocos años: transacciones bancarias, resolución de problemas complejos, control, docencia, comercio electrónico, base de datos.
- Pero, a medida que los sistemas de información son más complejos, han puesto de manifiesto una mayor cantidad de puntos vulnerables:
 - El número de posibles atacantes crece muy rápidamente.
 - Los medios disponibles para efectuar ataques siguen una evolución tan rápida como los propios sistemas.
 - La generalización de Internet hace que los ataques puedan provenir de cualquier lugar.

¿Qué hay que proteger?

- Es evidente la necesidad de proteger la información.
- Pero es muy difícil concretar qué es lo que hay que proteger, dado que el concepto de información es, en sí mismo, poco claro.
- Se choca también con el derecho a la intimidad:
 - Los gobiernos, y las empresas, necesitan multitud de datos de los ciudadanos, o clientes, para poder hacer planificaciones, estrategias de ventas, promover leyes, censos, tratamientos médicos, etc.

- Algunas de esas informaciones pueden ser manipuladas o utilizadas con fines distintos a los originales, o ser empleadas por organizaciones a las que no se les entregaron en su momento.

Existen situaciones muy comunes en las organizaciones de las cuales podemos tomar como clave para cualquier análisis, sin embargo no quiere decir que todas las organizaciones tengan las mismas situaciones, pues éstas pueden variar en cada organización dependiendo de sus actividades. A continuación mencionamos algunas de dichas situaciones comunes:

- Seguridad insuficiente en los centros de cómputo.
- Resguardo inseguro de archivos de respaldo.
- Falta de puntos de control en las interfaces de los sistemas.
- Segregación inadecuada de funciones entre personal que opera el sistema.
- Vulnerabilidad en los mecanismos de seguridad derivada de la falta de coordinación entre las diferentes áreas que permite el acceso a Internet.
- Instalación de software pirata.
- Instalación de juegos.
- Utilización de versiones obsoletas de software.
- Procedimientos de acceso a los sistemas mal definidos.
- Perfiles de usuarios que no corresponden 100% a sus actividades específicas.

2.2. Servicios de Seguridad

Los servicios de seguridad de la información son los servicios de nivel básico que son utilizados para combatir los ataques definidos en el siguiente subtema. Cada uno de los cuatro servicios de seguridad combate ataques específicos, es importante que no sean confundidos con mecanismos de seguridad, ya que los mecanismos es la implementación de los servicios de seguridad.

2.2.1. Confidencialidad

La información debe estar disponible solamente para aquellos usuarios autorizados a usarla. Es prevenir, detectar, impedir el descubrimiento de información. En general la Confidencialidad se refiere a la protección de datos implicados en entornos altamente protegidos, como entornos militares, comerciales, etc. Privacidad se refiere a información sobre individuos. En la mayoría de los países la Privacidad está protegida por las leyes.

2.2.2. Integridad

La información no se puede falsear. Los datos recibidos (o recuperados) son los mismos que fueron enviados (o almacenados), etc.

Es prevenir, detectar, impedir la modificación inadecuada de información. Por ejemplo en un entorno militar, el mando responsable de un misil no debe ser modificado inadecuadamente. En un entorno comercial, la integridad de los datos es especialmente relevante, puesto que el éxito de una organización depende de lo correctas que son las operaciones que se llevan a cabo y la coherencia en los datos.

Tenemos los siguientes tipos de integridad:

- a) **Integridad semántica:** Respeto en todo momento de las reglas de integridad definida en la base de datos.
- b) **Integridad Operacional:** Garantizar la consistencia de la base de datos con respecto al uso concurrente de la misma.

2.2.3. Disponibilidad

Es quién puede acceder a la información y cuando. La falta de accesibilidad produce una denegación de servicio, que es uno de los ataques más frecuentes en Internet.

La disponibilidad es prevenir, detectar, impedir la denegación inadecuada del acceso a servicios ofrecidos por el sistema.

El servicio de disponibilidad mantiene la utilidad de la información permite a los usuarios tener acceso a los sistemas de computo, a la información de los sistemas y a las aplicaciones que realizan operaciones sobre la información así como también permite que los sistemas de comunicaciones transmitan información entre ubicaciones o sistemas de computo.

La importancia relacionada con los mecanismos de recuperación de la base de datos ante caídas del sistema asegura la disponibilidad de la información.

2.2.4. Autenticidad

Se asegura el origen y el destino de la información. Autenticación: Verificación de la identidad de un componente que genera datos (principal) por parte de otro componente (verificador).

Ahora definiremos algunos más, que no son menos importantes ya que son manejados por algunos autores, los cuales son:

- **No repudio:** cualquier entidad que envía o recibe datos no puede alegar desconocer el hecho.
- **Consistencia:** asegurar que el sistema se comporta como se supone que debe hacerlo con los usuarios autorizados.
- **Aislamiento:** impedir que personas no autorizadas entren en el sistema.
- **Auditoria:** capacidad de determinar qué acciones o procesos se han llevado a cabo en el sistema y quién y cuándo las han llevado a cabo.
- **Prevención:** los usuarios deben saber que sus actividades quedan registradas.
- **Información:** posibilidad de detectar comportamientos sospechosos.

2.3. Tipos de Ataques

Existen varios aspectos a analizar para conocer con que recursos contamos. Para emprender el siguiente paso, a continuación se darán los tipos de ataques a los que está expuesto cualquier sistema informático. Sin importar como ocurren los eventos que afectan a la organización, los podemos clasificar de la siguiente forma:

1. Acceso
2. Modificación
3. Denegación de servicios
4. Refutación

Antes de definirlos es importante mencionar que estos tipos de ataques a su vez pueden ser clasificados de la siguiente forma:

- Ataques pasivos.
- Ataques activos.

Ataques Pasivos

En los ataques pasivos el atacante no altera la comunicación, sino que únicamente la escucha o monitoriza, para obtener información que está siendo transmitida. Sus objetivos son la interceptación de datos y el análisis de tráfico, una técnica más sutil para obtener información de la comunicación, que puede consistir en:

- Obtención del origen y destinatario de la comunicación, leyendo las cabeceras de los paquetes monitorizados.

- Control del volumen de tráfico intercambiado entre las entidades monitorizadas, obteniendo así información acerca de actividad o inactividad inusuales.
- Control de las horas habituales de intercambio de datos entre las entidades de la comunicación, para extraer información acerca de los períodos de actividad.

Los ataques pasivos son muy difíciles de detectar, ya que no provocan ninguna alteración de los datos. Sin embargo, es posible evitar su éxito mediante el cifrado de la información y otros mecanismos que se verán más adelante.

Ataques Activos

Estos ataques implican algún tipo de modificación del flujo de datos transmitido o la creación de un falso flujo de datos, pudiendo subdividirse en cuatro categorías:

- **Suplantación de identidad:** el intruso se hace pasar por una entidad diferente. Normalmente incluye alguna de las otras formas de ataque activo. Por ejemplo, secuencias de autenticación pueden ser capturadas y repetidas, permitiendo a una entidad no autorizada acceder a una serie de recursos privilegiados suplantando a la entidad que posee esos privilegios, como al robar la contraseña de acceso a una cuenta.
- **Reactuación:** uno o varios mensajes legítimos son capturados y repetidos para producir un efecto no deseado, como por ejemplo ingresar dinero repetidas veces en una cuenta dada.
- **Modificación de mensajes:** una porción del mensaje legítimo es alterada, o los mensajes son retardados o reordenados, para producir un efecto no autorizado. Por ejemplo, el mensaje “Ingresa un millón de pesetas en la cuenta A” podría ser modificado para decir “Ingresa un millón de pesetas en la cuenta B”.
- **Degradación fraudulenta del servicio:** impide o inhibe el uso normal o la gestión de recursos informáticos y de comunicaciones. Por ejemplo, el intruso podría suprimir todos los mensajes dirigidos a una determinada entidad o se podría interrumpir el servicio de una red inundándola con mensajes espurios. Entre estos ataques se encuentran los de denegación de servicio, consistentes en paralizar temporalmente el servicio de un servidor de correo, Web, FTP, etc.

2.3.1. Ataques de acceso

Un ataque de acceso es un intento de obtener información que el atacante no está autorizado a ver. Este tipo de ataque está dirigido contra la confidencialidad de la información.

- **Fisgoneo:** Consiste en hurgar entre los archivos de información con la esperanza de hallar algo interesante.

- **Escuchar furtivamente:** Cuando alguien escucha una conversación de la que no forma parte, se dice que escucha furtivamente. Esto ya no solo puede ser físicamente sino también electrónicamente.
- **Intercepción:** Es un ataque activo contra la información. Cuando un atacante obtiene información se coloca el mismo en la ruta de la información y la captura antes de que alcance su destino. Después de examinar la información, el atacante puede permitir que la información llegue o no a su destino.

2.3.2. Ataque de modificación

Es un intento de modificar la información que una persona no está autorizada. Este tipo de ataque es en contra de la integridad de la información.

- **Cambios:** El cambio de la información existente es un tipo de ataque, ya que al hacer cualquier cambio no autorizada la información es incorrecta. Los ataques de cambios pueden ser:
 - **Inserción:** Es otro tipo de ataque. Cuando se lleva a cabo la inserción, se agrega información que no existía con anterioridad. Este ataque puede ser montado en contra de información histórica o de información sobre la que todavía se harán modificaciones futuras.
 - **Eliminación:** Un ataque de eliminación es la remoción de la información existente. Esto puede ser la eliminación de la información en un registro histórico o bien la eliminación de un registro sobre el que todavía se harán modificaciones.

2.3.3. Ataque denegación de servicio

Los ataques de denegación de servicios (DoS, Denial-of-Service) son ataques que niegan el uso de los recursos a los usuarios legítimos del sistema, de la información o de las capacidades. Por lo general los ataques DoS no permiten que el atacante tenga acceso o modifique la información en el sistema de cómputo o en el mundo físico.

- **Denegación de acceso a la información:** Un ataque DoS en contra de la información provoca que dicha información no esté disponible. Esto puede ser causado por la destrucción de la información o por el cambio de la misma hasta dejarla de una manera utilizable. Esta situación también puede ser causada si la información aun existe pero ha sido removida hacia una ubicación inaccesible.
- **Denegación de acceso a las aplicaciones:** Otro tipo de ataque de DoS está dirigido a la aplicación que manipula o exhibe la información. Éste es normalmente un ataque en contra de un sistema de cómputo que ejecuta la aplicación. Si la aplicación no está disponible, la organización no puede realizar las tareas que son desempeñadas por esa aplicación.

- **Denegación de acceso a sistema:** Éste es un tipo común de ataque de DoS está dirigido para derribar sistemas de cómputo. En este tipo de ataque el sistema, junto con todas las aplicaciones que corren en el mismo y toda la información que se encuentra almacenada en él dejan de estar disponibles.
- **Denegación de acceso a comunicaciones:** Los ataques DoS en contra de las comunicaciones de se han realizado por muchos años. Este tipo de ataque puede abarcar desde cortar un alambre para entorpecer las comunicaciones de radio hasta inundar redes con tráfico excesivo. Aquí el objetivo es el medio de comunicación por sí mismo. Normalmente, los sistemas y la información permanecen ilesos, pero la carencia de comunicaciones evita el acceso a los sistemas y la información

2.3.4. Ataque de refutación

Es un ataque que va en contra de la responsabilidad de la información. Es tratar de transmitir información falsa o de negar que una transacción o evento reales hubieran ocurrido.

- **Denegación de evento:** Es simplemente negar que la acción se haya realizado como fue registrada. Este tipo de ataque es también conocido como no repudio.

2.4. Administración de Riesgos

Como vimos anteriormente las definiciones de los servicios de seguridad de la información nos beneficiaran para conocer los detalles de cómo se deben de emplear dentro de una organización, esto dependerán de un análisis de riesgo. Por lo tanto es importante conocer todo lo que involucra la administración de riesgos.

2.4.1. Activos

Los activos son aquellos componentes de la organización (tangibles e intangibles) que son parte del patrimonio de la misma y necesitan ser resguardados.

Se pueden así estructurar en 5 categorías:

1. El entorno del Sistema de Información
2. El Sistema de Información
3. La propia Información.
4. Las Funcionalidades de la Organización
5. Otros Activos

La falla de un Activo de una categoría puede generar cadenas de fallas en otras categorías.

Así por ejemplo, fallas en Activos del Entorno (1) provocarían otras fallas en el Sistema de Información (2); éstos inciden en fallas de la Información (3), que soporta las Funcionalidades de la Organización (4) y éstas condicionan los otros Activos (5).

Una frase común a la cual se suele recurrir es que "Una cadena se rompe por el eslabón más débil" lo mismo ocurre en materia de seguridad no importa que la seguridad para un activo sea alta si para otro esta es débil.

Metodología

Las técnicas de valoración del riesgo se basan en la identificación del mismo y en la asignación de un valor cualitativo y cuantitativo, para su ponderación respectiva.

2.4.2. Vulnerabilidad

Una vulnerabilidad se define como la "ocurrencia real de materialización de una Amenaza sobre un Activo"¹

La Vulnerabilidad es una propiedad de la relación entre un activo y una amenaza. También es considerada una vía de ataque potencial. Esta caracterizada por la dificultad que el nivel de capacidad técnica que se requiera para explotarla. El resultado de la explotación también debería ser tomada en cuenta. Ya que la consecuencia de una vulnerabilidad es que hace sistemas más propensos (débiles) de ser atacado por una amenaza o que un ataque tenga una mayor probabilidad de tener éxito.

2.4.3. Amenaza

Una amenaza se puede definir como: una acción o evento que puede violar la seguridad de un entorno de sistemas de información.²

Una amenaza contra los datos es una persona hostil que de manera casual o usando alguna técnica especial (usualmente denominamos como fuerza bruta, hace uso de técnicas especiales para poder acceder a los datos), para difamar o modificar los datos manejados por un sistema.

Tipos de amenazas para Una Bases De Datos

¹ Maiwald, Eric, *Fundamentos de Seguridad de Redes*, Segunda Edición, McGraw-Hill, México 2005, p. 144.

² Ibidem, p.145.

Los datos se dividen en 2 clases:

- Datos
- Metadatos

Las amenazas afectan a ambas clases de datos pero con diferente impacto.

Amenazas:

- **Accidentales:** Fallos del software/hardware y errores humanos.
- **Intencionadas:** Intrusos y abusos de usuarios autorizados.

Existen tres componentes de amenazas:

- a. **Objetivo:** El aspecto de la seguridad que puede ser atacado.
- b. **Agentes:** Las personas u organizaciones que originan las amenazas
- c. **Eventos:** El tipo de acción que representa la amenaza

a. Objetivos

Cuando la motivación es revelar la información sin autorización a individuos u organizaciones, confidencialidad es el blanco. Cuando la amenaza implica modificar la información, el objetivo es la integridad.

El atacante busca modificar ya sea propia o de otras personas (por ejemplo: alterar la base de datos para poner en duda la exactitud de los datos).

La disponibilidad cuando se ejecuta un ataque de denegación de servicios. Tales ataques pueden enfocarse en la disponibilidad de la información, de las aplicaciones, de los sistemas o de la infraestructura.

b. Agentes

Son las personas que pretenden dañar a una organización. Para ser creíbles de una amenaza, debe tener tres características.

- **Acceso.** La capacidad que tiene un agente para llevar a cabo el objetivo (contraseñas, password de acceso, etc.). Un agente debe tener acceso al sistema, a la red, a las instalaciones o a la información que desea.
- **Conocimiento.** El nivel y tipo de información que tiene un agente acerca del objetivo. El conocimiento que puede ser útil para un agente incluye lo siguiente:

- Identificación de usuarios.
 - Contraseñas.
 - Ubicación de archivos.
 - Procedimiento de acceso físico.
 - Nombres de empleados.
 - Número telefónico de acceso.
 - Dirección de red.
 - Procedimientos de seguridad.
- **Motivación.** Las razones que puede tener un agente para presentar una amenaza hacia el objetivo. Un agente requiere de una motivación para actuar en contra del objetivo. Probablemente por las siguientes causas:
 - Reto
 - Codicia
 - Intento malintencionado

Agentes a considerar

- Los empleados
- Los hackers
- Rival comerciales
- Terroristas
- Delincuentes
- Público en general
- Las compañías que prestan servicio a la organización
- Los clientes
- Los invitados
- Los desastres

c. Eventos

Son la manera o forma en la que un agente de amenazas puede ocasionar el daño a una organización.³

- Abuso del acceso autorizado a la información, a los sistemas o a los sitios
- Alteración malintencionada de la información
- Alteración accidental de la información
- Acceso no autorizado a la información, a los sistemas o a los sitios
- Destrucción malintencionada de la información, de los sistemas o de los sitios
- Destrucción accidental de la información, de los sistemas o e los sitios
- Interferencia física malintencionada con los sistemas o las operaciones
- Interferencia física accidental con los sistemas o con las operaciones

³ Idem

- Eventos físicos naturales que pueden interferir con los sistemas o con las operaciones
- Introducción de software malintencionado (de manera intencional) a los sistemas
- Interrupción de las comunicaciones internas o externas
- Escucha furtiva pasiva de comunicaciones internas o externas
- Robo de hardware o software

2.4.4. Análisis de Riesgo

La seguridad informática tiene como objetivo el mantenimiento de la confidencialidad, integridad, autenticidad y disponibilidad de los sistemas informáticos, así como definir e implantar las defensas necesarias para eliminar o reducir sus posibles consecuencias.

Sus componentes:

- Sistema de información.
- Amenaza
- Vulnerabilidad
- Impacto
- Riesgo
- Control o defensa

Algunos de estos componentes ya han sido explicadas sin embargo a continuación daremos la siguientes definiciones que harán que el tema sea mejor comprendido.

La defensa o control es todo medio físico o lógico empleado para mitigar un riesgo. Un riesgo es el potencial de lo que puede ser perdido y necesita protección y la probabilidad de que éste produzca un impacto en la organización.

$$\text{Riesgo} = \text{Amenaza} + \text{Vulnerabilidad}$$

El riesgo es la combinación de las amenazas con la vulnerabilidad. Las amenazas sin vulnerabilidades no representan un riesgo. Del mismo modo, las vulnerabilidades sin amenazas no plantean riesgo alguno. En el mundo real no existe ninguna de estas dos condiciones.⁴

Por tanto, la evaluación del riesgo es un intento de identificar la probabilidad de que ocurra un evento perjudicial. El riesgo puede definirse cualitativamente en estos tres niveles:

- Bajo
- Medio
- Alto

⁴ Ibidem, p. 149.

La identificación del riesgo es sencilla. Todo lo que se tiene que hacer es identificar tanto las vulnerabilidades como las amenazas. La identificación de los riesgos que puede tener una organización debe ser adaptada a sus características.

2.5. Problemas de Seguridad en las Bases de Datos

Como lo hemos visto en los ambientes de bases de datos, las diferentes aplicaciones y usuarios de una organización acceden a los datos a través del DBMS. Una de los principales problemas a los que se enfrenta un DBMS y debe resolver a través de una buena administración es el controlar que no halla datos duplicados, inconsistencia de los datos, y el manejo de amenazas de posibles usuarios no autorizados al acceso de los datos.

Una amenaza contra los datos la podemos definir de la siguiente manera: como una persona hostil que de manera casual o usando alguna técnica especial (usualmente denominamos como fuerza bruta, hace uso de técnicas especiales para poder acceder a los datos), para difamar o modificar los datos manejados por un sistema,

Las violaciones a la seguridad de las bases de datos consisten en, la lectura de datos y difamación de esos datos, modificaciones de los datos, y borrado de los datos, por personas no autorizadas.

Las consecuencias de las violaciones de los datos están agrupadas dentro de 3 categorías:

- **Difamación de los datos:** Esto es causado por lectura de datos de manera intencionalmente así como casualmente por usuarios no autorizados, incluyendo en esta categoría las violaciones de claves secretas de datos autorizados únicamente a ciertas personas.
- **Impropia modificación de los datos:** Esto en vuelve todas las violaciones a la integridad de los datos a través del manejo impropio de los datos o modificaciones de los mismos.
- **Mal funcionamiento del servicio:** Envuelve todas aquellas acciones que niegan el servicio a los usuarios al acceso de los datos o uso de los recursos.

Las amenazas contra la seguridad de los datos están clasificadas dentro de los siguientes factores en los que puede ocurrir de manera accidental o intencional.

- **Desastres naturales o accidentales:** tales como temblores, inundaciones o fuego. Estos accidentes pueden dañar los sistemas de hardware como los de almacenamiento de los datos, estos accidentes siempre causan violaciones de integridad y negación del servicio.
- **Errores o problemas en hardware o software:** Esto puede permitir la incorrecta aplicación de políticas de seguridad y por consiguiente el acceso fácil para la lectura de datos no autorizados y modificación de los mismos o la negación del servicios a las personas autorizadas al uso y manejo de los datos.

- **Errores humanos:** violaciones al acceso del sistema no intencionadas como al dar una clave de acceso incorrecta o un mal uso de las aplicaciones las consecuencias son las mismas a las anteriores causando problemas en el sistema o falta de integridad de los datos.

Factores intencionales denotan explícita y determinadamente fraude que causa problemas en los accesos de los datos, las violaciones a los sistemas envuelven dos tipos de usuarios los cuales son:

- **Usuarios autorizados:** Aquellos quienes abusan de sus privilegios y autorizaciones para hacer de los datos lo que a ellos mejor le convenga, como es el vender información dar accesos a ciertos datos o eliminar información.
- **Agentes hostiles:** Este tipo de personas realizan acciones de vandalismo hacia el software y/o hardware, leyendo, modificando datos privados. En ambos casos "legal" o "ilegal", el uso que estas personas le dan a los datos puede ser para sus propósitos fraudulentos. Usualmente las personas hostiles (denominadas también (hackers) suelen usar cierto tipos de técnicas para hacerse de información tales como; virus, caballos de troya y trampas de puerta, éstas son algunas de las técnicas que las personas hostiles usan para hacerse de la información.

Estos son solo algunos de los problemas que se pueden presentar para la base de datos, sin embargo como lo mencionamos anteriormente al realizar el análisis de riesgo, determinaremos cuales son sus principales problemas.

2.5.1. Requerimientos de Protección de una Base de Datos

Proteger una base de datos de posibles amenazas significa proteger sus recursos particularmente los datos almacenados. La protección de la base de datos se refiere a accidentes así como de accesos no autorizados para la lectura y/o modificación de los datos.

Protección de accesos no autorizados. Esto consiste en garantizar el acceso a la base de datos sólo a usuarios autorizados. El acceso a la base de datos debe ser revisado por el DBMS dependiendo de los recursos y aplicaciones que el usuario va a ejecutar.

Los controles de acceso son un punto muy complejo para las bases de datos debido a los archivos que son manejados por el sistema operativo. El encargado de dar los privilegios necesarios para que los usuarios puedan tener acceso a los recursos de Base de Datos es el DBA de acuerdo a los siguientes aspectos.

- Lo que necesite el usuario para llevar acabo sus actividades y
- De acuerdo a las políticas que se manejen dentro de la organización.
- Integridad de la Base de Datos.

Este requerimiento compete a la protección de la base de datos de accesos no autorizados que pueden modificar el contenido de la base, tanto de virus, sabotajes, o fallas en el sistema que pueden dañar los datos almacenados.

Este tipo de protección es atendido por el DBMS a través de controles adecuados del sistema, y de tener respaldos, y procedimientos automáticos de recuperación de datos. Los respaldos y los procedimientos de recuperación de datos son procesos que constantemente se están estudiando para implementar un mejor funcionamiento en el desarrollo de los DBMS. Para perseverar la consistencia de los datos se requiere que cada transacción sea atómica. Las transacciones atómicas se refieren a:

- El termino de una transacción correcta, modificando el acceso de los datos.
- El termino de una transacción no exitosa, sin modificar los datos al hacer el acceso a estos.

Después que se ha realizado una transacción correctamente los datos modificados son de manera permanente, hasta que otro proceso los vuelve a modificar.

El sistema de recuperación de datos usa un log diario, normalmente es un archivo que contiene toda la secuencia de las operaciones realizadas sobre los registros dentro de una tabla (relación) de almacenamiento.

Por cada transacción realizada. el log diario graba las operaciones que se han ejecutado sobre los datos, ya sea para lectura, escritura, inserciones y borrado, como también las operaciones de control, todas aquellas operaciones que se ejecutaron sobre los datos y no tuvieron éxito.

Auditoria y Contabilidad del sistema

Estos puntos consisten en la posibilidad de recobrar y verificar todos los accesos hechos a los datos en operaciones de lectura y escritura. Tanto la auditoría como la contabilidad son herramientas útiles para llevar un control de la integridad física de los datos así como también el análisis de acceso a la base de datos.

En ocasiones puede ser que el elevado número de accesos a la base de datos haga que la revisión de los registros almacenados de todas las entradas sea impráctica desde el punto de vista de tiempo y dinero, pero es tarea del administrador de la base de datos tener el control pleno del sistema.

Uso de autenticación

Este requerimiento determina la necesidad de identificar únicamente la base de datos del usuario y al usuario con la base de datos.

La identificación del usuario debe ser pedida por mecanismos del propio sistema, cada vez que el usuario intente hacer uso de los datos, los usuarios tendrán permitido el acceso a los datos y con ciertos privilegios que el administrador les dé, siempre y cuando se identifiquen con el sistema.

Protección de Multinivel

La protección de multinivel significa diferentes niveles hacia los datos dependiendo de su importancia de los mismos, como se menciono anteriormente habrá datos de dominio público y otros más que de uso exclusivo, es por eso la necesidad de clasificar los datos dependiendo de su importancia.

2.5.2. Seguridad Informática

Sistemas informáticos

Un sistema informático es el conjunto de elementos hardware, software, datos y personas que permiten el almacenamiento, procesamiento y transmisión de información que permite el funcionamiento de una organización.

Todos los elementos de un sistema informático son vulnerables:

- **Hardware:** aislamiento de los CPD, sistemas contra incendios, etc.
- **Software:** bombas lógicas, caballos de Troya, gusanos, virus.
- **Personas:** personas que trabajan en la administración de los CPD, en la gestión de los sistemas de comunicaciones, etc.
- **Datos:** son los ataques más sencillos de practicar y, por lo tanto, donde más se necesita la aplicación de políticas de seguridad.

Principios fundamentales de la seguridad informática

1. Principio de menor privilegio:

Este es quizás el principio más fundamental de la seguridad, y no solamente de la informática. Básicamente, el principio de menor privilegio afirma que cualquier objeto (usuario, administrador, programa, sistema, etc.)

Debe tener tan solo los privilegios de uso necesarios para desarrollar su tarea y ninguno más.

2. Seguridad no equivale a oscuridad.

Un sistema no es más seguro porque escondamos sus posibles defectos o vulnerabilidades, sino porque los conozcamos y corriamos estableciendo las medidas de seguridad adecuadas.

El hecho de mantener posibles errores o vulnerabilidades en secreto no evita que existan, y de hecho evita que se corrijan.

No es una buena medida basar la seguridad en el hecho de que un posible atacante no conozca las vulnerabilidades de nuestro sistema. Los atacantes siempre disponen de los medios necesarios para descubrir las debilidades más insospechadas de nuestro sistema.

3. Principio del eslabón más débil.

En un sistema de seguridad el atacante siempre acaba encontrando y aprovechando los puntos débiles o vulnerabilidades.

Cuando diseñemos una política de seguridad o establezcamos los mecanismos necesarios para ponerla en práctica, debemos contemplar todas las vulnerabilidades y amenazas. No basta con establecer unos mecanismos muy fuertes y complejos en algún punto en concreto, sino que hay que proteger todos los posibles puntos de ataque.

4. Defensa en profundidad.

La seguridad de nuestro sistema no debe depender de un solo mecanismo por muy fuerte que éste sea, sino que es necesario establecer varios mecanismos sucesivos. De este modo cualquier atacante tendrá que superar varias barreras para acceder a nuestro sistema.

5. 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.

6. 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. Por ejemplo, si nuestros mecanismos de control de acceso al sistema fallan, es preferible que como resultado no dejen pasar a ningún usuario a que dejen pasar a cualquiera aunque no esté autorizado.

7. Participación universal.

La participación voluntaria de todos los usuarios en la seguridad de un sistema es el mecanismo más fuerte conocido para hacerlo seguro. Si todos los usuarios prestan su apoyo y colaboran en establecer las medidas de seguridad y en ponerlas en práctica el sistema siempre tenderá a mejorar.

8. Principio de simplicidad.

La simplicidad es un principio de seguridad por dos razones:

- En primer lugar, mantener las cosas simples, las hace más fáciles de comprender. Si no se entiende algo, difícilmente puede saberse si es seguro.
- En segundo lugar, la complejidad permite esconder múltiples fallos.

Los programas más largos y complejos son propensos a contener múltiples fallos y puntos débiles.

Seguridad en Sistemas Operativos.

Los sistemas Operativos (Windows, Unix, MacOS,...) son los encargados de la interacción entre los usuarios de las máquinas y los recursos informáticos.

- Por tanto, forman la primera línea de seguridad lógica.

Un sistema operativo “seguro” debería contemplar:

- Identificación y autenticación de los usuarios.
- Control de acceso a los recursos del sistema.
- Monitorizar las acciones realizadas por los usuarios, sobre todo las que sean sensibles desde el punto de vista de seguridad.
- Auditoría de los eventos de posible riesgo.
- Garantía de integridad de los datos almacenados.
- Garantía de la disponibilidad de los recursos.

2.5.3. Aspectos Principales de las Bases de Datos enfocados a la Seguridad

Aspecto Legal.

Existen muchas cuestiones legales respecto a la seguridad de la información. La cuestión más obvia es que irrumpir dentro de las computadoras va contra la ley; bien, la mayor parte del tiempo lo es. Esto dependiendo de la parte del mundo en que se encuentre, éste definirá un delito computacional, como también definirá la penalización por ocuparse en tal actividad.

No importarán como se defina las actividades, para que los autores del delito sean castigados, los profesionales de la seguridad en la información deben comprender cómo reunir la información necesaria para ayudar a los responsables de hacer cumplir la ley en la captura de los individuos responsables y en el proceso judicial al que serán sometidos.

Todas las cuestiones legales y de privacidad deben ser consideradas, ya que las organizaciones deben comprender qué riesgos asumen respecto a empleados y otras organizaciones en red si la seguridad interna es negligente, toda esta actividad se deberá llevar a cabo en conjunción con los consejeros legales de la organización.

Aspecto Ético y Social.

La ética es la ciencia del comportamiento humano. Es una ciencia práctica y normativa que estudia racionalmente la conducta humana. (La bondad y la maldad del ser humano). Y se dice que es ciencia porque “explica las cosas que lo causan”, se dice que es práctica porque esta hecha justamente para realizarse en la vida diaria, es normativa porque da normas para la vida, orienta la conducta práctica. Es rectora de la conducta humana.

Finalmente la ética es evitar cometer actos que puedan dañar a otros. En las recientes investigaciones se ha encontrado que el 80% de las amenazas de las empresas es de los propios empleados, las cuales pueden ser de manera intencional o no y el otro 20% es de cualquier otro tipo.

Es importante mencionarlo, ya que muchas de estas conductas es por la falta de ética, no solo de los ingenieros sino de muchas otras personas con malas intenciones o beneficios propios.

Responsabilidad Profesional, Ética y Legal de los Ingenieros en la Sociedad y Consigo mismo.

Siendo el propósito de los Códigos de Ética Profesional de los Códigos de Ingenieros que los ingenieros proporcionen un servicio profesional con lealtad, honestidad y competencia. El profesional de la ingeniería debe de ser responsable de los intereses de sus clientes y de sus empleados y salvaguardar la seguridad y el bienestar público. Recientemente se han recomendado, en términos de prioridades éticas que el ingeniero tratara con las necesidades de los otros de acuerdo con la siguiente escala:

- La sociedad y el público.
- La ley.
- La profesión de ingeniero.
- Otros ingenieros involucrados.
- Los clientes del ingeniero.
- Empresas u organismos públicos o privados.
- La persona del ingeniero.

Lo anterior nos permite señalar que la función de un código de ética profesional de la ingeniería es ayudar a guiar responsable, ética y legalmente, las decisiones que los ingenieros ejecutan diariamente. La ley de profesiones al delegar a los colegios de profesionistas al vigilancia del ejercicio profesional dio origen a la creación de colegios de las profesiones que necesitan título para su ejercicio, siendo la ingeniería en sus diversas ramas profesionales, una de las que las que lo requieren, se han establecido diversos colegios. Y complementariamente se han establecido organizaciones de ingenieros.

Finalmente cabe mencionar que en la mayoría de los países con organizaciones de ingenieros, cuentan con colegios profesionales y también con códigos de ética profesional, que se han ido modificando y adaptando al cambiante medio en que se desarrollan los ingenieros,

2.5.4. Tipos de políticas para la base de datos

Tipos de Políticas en relación a las bases de datos:

- a. Políticas Administrativas.
- b. Políticas De control de accesos.
- c. Políticas De control del flujo de información.

a. Políticas administrativas

- Control centralizado o descentralizado.
- Propietario (creador) y administrador (responsable).

b. Políticas de control de accesos

- De menor privilegio.
- Restringir la información de la base de datos necesaria para las tareas.
- Dependiente del atributo.
- Acceso en función del nombre del atributo
- Dependiente del contenido.
- Dependiente del contexto.
- Dependiente de la historia.
- Sistemas abiertos y cerrados.
- Prohibición/autorización explícita.
- Derechos de acceso. Tipos de derechos: Leer, Actualizar (Insertar, Borrar).
- Nivel de responsabilidad.

Dentro de las políticas de control de acceso también se encuentran las políticas de acceso funcional:

- **Política de acceso funcional:** En la base de datos a través de estadísticas deben poderse especificar atributos y funciones.

c. Políticas de control de flujo de información

En este tipo de políticas se tienen como objetivo prevenir el flujo de información desde un sujeto autorizado a otro no autorizado.

Para ello se especifican y hacen políticas de los canales autorizados a través de los cuales la información puede transmitirse.

Dentro de las políticas de control de flujo de información se tienen las siguientes políticas:

- Políticas de acceso discrecional:
 - Alguien autoriza el acceso a los datos.
 - No controlan el flujo de información.
- Política de acceso no discrecional u obligatorio, política de acceso no discrecional u obligatorio
 - Se compartimentan los usuarios y los datos.
 - El flujo de información se controla comprobando qué usuarios y datos pertenezcan al mismo compartimiento.
 - Debe ser posible la mezcla de ambas políticas.

2.6. Norma ISO/IEC 17799

Como se ha visto en el desarrollo de la tesis que la base de datos requiere de un modelo de seguridad para lo cual se hará un análisis de riesgo, sin embargo, sabemos que existen estándares de seguridad por lo que en este subtema nos enfocaremos a la siguiente norma que esta diseñada para la seguridad de la información.

2.6.1. Historia

Código de Prácticas para la Administración de la Seguridad de la Información

Los orígenes de ISO-17799 están en BS7799. En mayo de 1987 →un grupo dedicado a la seguridad de la información. “Centro Comercial de Seguridad en Cómputo del Departamento Británico de Comercio e Industria” (CCSC - DTI) tareas principales:

- Brindar ayuda a los vendedores de productos de seguridad TI mediante el establecimiento de un conjunto de criterios de evaluación de seguridad reconocidos internacionalmente.
- Brindar ayuda a los usuarios mediante un código de buenas prácticas de seguridad dando origen al “Código de prácticas para usuarios” publicado en 1989. El “Código de prácticas para usuarios” retomado y desarrollado por NCC (Centro Nacional de Cómputo) por un grupo de usuarios de la industria británica para asegurar que el código fuera significativo y práctico desde el punto de vista del usuario.

En febrero de 1993 se publican los resultados como un documento guía del estándar británico (BS). Recibe el nombre “PD 0003: código de prácticas para el manejo de seguridad de la información”. Trabajo que da lugar a la primera versión del estándar británico BS7799, publicándose y poniéndose en circulación en 1995. En 1998 el BSI (Instituto Británico de Estándares) programa para acreditar a las firmas auditoras: cuerpos de certificación y auditores individuales para auditar organizaciones con base en el estándar BS7799

Esquema de acreditación conocido como C:cure.

El objetivo es proveer un nivel de confianza alto a las organizaciones y a sus socios comerciales buscando la seguridad de sus activos y recursos TI.

El cual fue de gran aceptación de la norma británica por: Australia, Sudáfrica, Nueva Zelanda, Holanda y Noruega el gobierno del Reino Unido recomendó como parte de su “Ley de Protección a la Información de 1998” (que entró en vigor el 1° de marzo de 2000) que las compañías británicas utilizaran el estándar BS7799 como método de cumplimiento de esa ley

Los objetivos del estándar son:

- **Capacitar.** A las organizaciones para implementar apropiadamente seguridad TI.
- **Proveer.** Una guía común de mejores prácticas.
- **Facilitar.** El comercio entre compañías dando confianza en la seguridad TI compartida.
- **Brindar.** A los profesionales TI anteproyecto para desarrollar políticas y procesos de seguridad empresarial.

Mientras algunas organizaciones utilizaron el estándar BS7799, otras expresaron la necesidad de un estándar común.

Esta demanda condujo al rápido seguimiento del estándar BS7799 dando origen al lanzamiento realizado por la ISO en Diciembre de 2000 y teniendo como nombre ISO/IEC 17799:2000.

ISO 17799 es el único estándar de alto nivel y de naturaleza conceptual dedicado al manejo de la seguridad de la información en un campo manejado por “Principios” y “Buenas Prácticas”.

2.6.2. Objetivo

Este estándar define a la información como un activo o recurso que existe de muchas formas y que tiene amplio valor para una cierta organización.

La meta de la seguridad de la información es proteger este recurso de manera oportuna y conveniente de un rango muy amplio de vulnerabilidades, de tal manera que se asegure la continuidad del negocio, se minimicen los daños y se maximicen las ganancias de las inversiones y las oportunidades de negocio.

ISO 17799

Define la seguridad de la información como la conservación de:

- Confidencialidad
- Integridad
- Disponibilidad
- Autenticidad

La seguridad de la información se logra implementando un conjunto de controles, los cuales pueden ser políticas, prácticas, procedimientos, estructuras organizacionales y funciones de software. Estos controles necesitan ser establecidos para asegurar que se alcancen los objetivos específicos de seguridad de la organización.

Tipos de Controles

a. Basados sobre requerimientos legislativos: (aspecto legal)

- Protección de datos y privacidad de información personal.
- Guardado de registros organizacionales.
- Derechos de propiedad intelectual.

b. Los considerados a ser la mejor práctica para la seguridad de la información:

- Información sobre la documentación de las políticas de seguridad.
- Distribución de responsabilidades en la seguridad de la información.
- Educación y entrenamiento en seguridad de la información.
- Reporte de incidentes de seguridad.
- Dirección de continuidad de negocios.

2.6.3. Áreas de control

- 1. Política de seguridad:** Se necesita una política que refleje las expectativas de la organización en materia de seguridad a fin de suministrar administración con dirección y soporte. La política también se puede utilizar como base para el estudio y evaluación en curso. (Ver apéndice A sobre creación de políticas).
- 2. Organización de la seguridad:** Sugiere diseñar una estructura de administración dentro la organización que establezca la responsabilidad de los grupos en ciertas áreas de la seguridad y un proceso para el manejo de respuesta a incidentes.
- 3. Control y clasificación de los recursos de información:** Necesita un inventario de los recursos de información de la organización y con base en este conocimiento, debe asegurar que se brinde un nivel adecuado de protección.
- 4. Seguridad del personal:** Establece la necesidad de educar e informar a los empleados actuales y potenciales sobre lo que se espera de ellos en materia de seguridad y asuntos de confidencialidad. También determina cómo incide el papel que desempeñan los empleados en materia de seguridad en el funcionamiento general de la compañía. Se debe tener implementar un plan para reportar los incidentes.
- 5. Seguridad física y ambiental:** Responde a la necesidad de proteger las áreas, el equipo y los controles generales.
- 6. Manejo de las comunicaciones y las operaciones:** Conservar la integridad y disponibilidad del procesamiento y la comunicación de la información protegiéndola en las redes e infraestructura de soporte.
- 7. Control de acceso:** Establece la importancia de monitorizar y controlar el acceso a la red y los recursos de aplicación para proteger contra los abusos internos e intrusos externos.
- 8. Desarrollo y mantenimiento de los sistemas:** Recuerda que en toda labor de la tecnología de la información, se debe implementar y mantener la seguridad mediante el uso de controles de seguridad en todas las etapas del proceso.

9. Manejo de la continuidad de la empresa: Aconseja estar preparado para contrarrestar las interrupciones en las actividades de la empresa y para proteger los procesos importantes de la empresa en caso de una falla grave o desastre.

10. Cumplimiento: Imparte instrucciones a las organizaciones para que verifiquen si el cumplimiento con la norma técnica ISO 17799 concuerda con otros requisitos jurídicos, como la Directiva de la Unión Europea que concierne la Privacidad, la Ley de Responsabilidad y Transferibilidad del Seguro Médico (HIPAA por su sigla en Inglés) y la Ley Gramm-Leach-Bliley (GLBA por su sigla en inglés).

Esta sección también requiere una revisión a las políticas de seguridad, al cumplimiento y consideraciones técnicas que se deben hacer en relación con el proceso de auditoría del sistema a fin de garantizar que las empresas obtengan el máximo beneficio.

Políticas de Seguridad

Objetivo:

Proporcionar dirección de administración y soporte para la seguridad de la información.

Documento sobre Información de Políticas de Seguridad

Un documento de políticas debe ser aprobado por la dirección, publicado y comunicado, como sea apropiado, a todos los empleados.

Revisión y evaluación

La política debe tener un propietario que es responsable de su mantenimiento y revisión de acorde a un proceso definido o cuando hay cambios que afectan las bases del riesgo original.

2.6.4. Organización de la Seguridad

Infraestructura de seguridad de la información

Objetivo:

Administrar la seguridad de la información dentro de la organización.

Coordinación de seguridad de la información

En una organización grande debe estar conformado un foro por representantes de partes relevantes de la organización para implementar controles de seguridad. Este foro asigna responsabilidades, establece metodologías de seguridad, revisión de incidentes, etc.

Asignación de responsabilidades de seguridad de la información

Deben ser definidas responsabilidades para cada bien y para realizar procesos de seguridad específicos. Niveles de autorización deben ser definidos y documentados.

Consejo de especialistas en seguridad de la información

Debe ser proporcionado por un consejero o asesor experimentado.

Cooperación entre organizaciones

Contactos apropiados con autoridades legales, cuerpos regulatorios, proveedores de servicio y operadores de telecomunicaciones, para asegurar que una acción apropiada puede ser tomada rápidamente en un incidente de seguridad.

Revisión independiente de seguridad de la información

Es recomendable que sea realizada por un administrador independiente o una organización externa especialista en auditorías.

Seguridad de Acceso de Terceras Personas

Objetivo:

Mantener la seguridad de la información organizacional accedida por terceras personas.

Identificación de Riesgos del Acceso de Terceras Personas

- Tipos de acceso.
- Debe ser considerado el acceso físico y el lógico.
- Requerimientos de seguridad en contratos de terceras personas
- Términos que deben ser considerados en el contrato:
- Política general sobre seguridad de la información.
- Descripción de cada servicio disponible.
- Niveles inaceptables de servicio.
- Obligaciones respectivas de las partes del contrato.
- Responsabilidades legales.
- Derechos de propiedad intelectual.
- Acuerdos de control de acceso.

Control y Clasificación de los Bienes Informáticos

Contabilidad de Bienes

Objetivo:

Mantener protección apropiada de bienes organizacionales.

Inventario de bienes

- Una organización necesita identificar sus bienes y el valor relativo e importancia.
- Con esta información una organización puede proporcionar niveles de protección de acuerdo a los distintos valores de bienes.
- Cada bien debe ser identificado y documentado, junto con su localización actual.

Seguridad del Personal

Definición de Seguridad en el Trabajo y Recursos

Objetivo:

Reducir los riesgos de error humano, fraude o mal uso.

Responsabilidades de seguridad en el trabajo

Las responsabilidades deben estar documentadas apropiadamente.

Resguardo de personal y políticas

- Disponibilidad de referencias satisfactorias.
- Verificación de currículos de los aspirantes.
- Confirmación de calificaciones académicas y profesionales.
- Verificación independiente de identidad.

Acuerdos de confidencialidad.

Son usados para señalar a la información que es confidencial o secreta. Los empleados deben firmar un acuerdo como parte de su contrato de trabajo.

Términos y condiciones de empleo:

- Deben mostrar las responsabilidades de seguridad de información.
- Cuando sea apropiado estas responsabilidades deben continuar por un periodo después del término del empleo.
- Derechos y obligaciones deben estar definidos.

Entrenamiento del Usuario

Objetivo:

Asegurar que los usuarios están conscientes de las amenazas y están preparados para soportar las políticas.

Educación y entrenamiento en seguridad de la información. Todos los empleados y, cuando sea necesario, usuarios externos, deben recibir entrenamiento apropiado y actualizaciones regulares en políticas y procedimientos y en buen manejo de TI.

Respuesta a Incidentes y Mal Funcionamiento

Objetivo:

Minimizar el daño de los incidentes de seguridad y mal funcionamiento.

Reporte de incidentes de seguridad

Deben ser reportados por los canales apropiados tan rápido como sea posible. Incluye el procedimiento de reporte y el procedimiento a la respuesta al incidente.

Reporte de malfuncionamiento de SW

Se deben establecer procedimientos para mal función de SW.

Proceso disciplinario

Debe establecerse para los empleados que han violado políticas de seguridad.

Seguridad Física y del Entorno

Áreas de Seguridad

Objetivo:

Prevenir acceso no autorizado, daño e interferencia.

Perímetro de seguridad física

La protección física puede ser lograda creando algunas barreras físicas a los sistemas de información.

Controles de entrada física

Las áreas de seguridad deben estar protegidas por apropiados controles de entrada para asegurar que solo personal autorizado tiene acceso.

Seguridad en oficinas, salas y facilidades

La selección y diseño de un área segura debe tomar en cuenta las posibilidades de amenazas naturales o humanas.

Trabajo en áreas seguras

El personal debe estar consciente de la existencia de áreas de seguridad. Debe ser evitado el trabajo no supervisado en áreas de seguridad.

No debe permitirse fotos, video, audio; sólo con autorización.

Seguridad del Equipo

Objetivo:

Prevenir pérdida, daño o compromiso de bienes e interrupción de actividades.

Colocación del equipo y protección

Debe ser colocado o protegido para reducir riesgos de amenazas del ambiente, humanas y oportunidades de acceso no autorizado.

Suministro de energía

El equipo debe estar protegido de fallos de energía y anomalías.

Seguridad del cableado

Cableado de energía y de telecomunicaciones debe estar protegido de interceptación o daño. Se sugiere el uso de fibra óptica para sistemas críticos, rutas o medios de transmisión alternativos.

Mantenimiento del equipo

Mantenimiento basado en especificaciones y normas. Realizado por personal autorizado.

Administración de Comunicaciones y Operaciones

Procedimientos Operacionales y Responsabilidades

Objetivo:

Asegurar la operación segura y correcta de facilidades de procesamiento de información.

Documentación de procedimientos de operación

Los procedimientos de operación deben ser documentados y mantenidos, los cambios deben ser autorizados por la dirección.

Control de cambio organizacional

Los cambios a las facilidades de procesamiento de información deben ser controlados.

Procedimientos de dirección de incidentes

Cubrir todos los tipos potenciales de incidentes de seguridad. Acciones para recuperación y corrección de fallos deben ser controlados.

Planeación del Sistema y Aceptación

Objetivo:

Minimizar el riesgo de fallo de sistemas.

Capacidad de planeación

Las demandas de capacidad deben ser monitoreadas y proyecciones de requerimientos de capacidad futuros deben ser hechos para asegurar disponibilidad.

Aceptación del sistema

Criterios de aceptación para nuevos sistemas de información y nuevas versiones deben ser establecidos, acordados y documentados, así como las pruebas pertinentes.

Protección Contra Software Malicioso

Objetivo:

Proteger la integridad de software e información.

Controles contra software malicioso

- Políticas para licencias de software y prohibir el uso de software no autorizado.
- Control sobre instalaciones, actualizaciones, vacunas y reparación.
- Revisiones regulares de software y contenido de datos.
- Procedimientos contra protección de virus y ataques.

Respaldos

Objetivo:

Mantener la integridad y disponibilidad de la información.

Respaldo de información y Registros de operación

Administración de Red

Objetivo:

Asegurar el aseguramiento de info. en redes.

Controles de red

Administradores deben implementar controles para asegurar la seguridad de los datos en red y la protección de servicios.

Manejo de Medios y Seguridad

Objetivo:

Prevenir daños a bienes e interrupciones.

Administración de medios removibles

Procedimientos para la administración de medios removibles, su copia, lectura y almacenamiento.

Seguridad de documentación del sistema

Controles para proteger documentación del sistema de acceso no autorizado.

Intercambio de Información y SW

Objetivo:

Prevenir pérdida, modificación o mal uso en el intercambio de información entre organizaciones.

Acuerdos de intercambio de información y software

- Procedimientos para transmisión y recepción.
- Estándares de identificación y Responsabilidades.

Seguridad de medios en tránsito

- Transporte o correo confiable debe ser usado.
- Controles especiales para información sensible como firmas digitales o encriptación.
- Seguridad de comercio electrónico. Deben establecerse controles para amenazas del comercio electrónico como la autenticación, autorización.

Seguridad de correo electrónico

- Ataques.
- Protección.
- Consideraciones legales.
- Responsabilidades de proveedores y empleados.
- Uso de criptografía.

Acceso del Usuario

Objetivo:

Prevenir el acceso no autorizado a los sistemas de información.

Los procedimientos formales deben controlar la asignación de los derechos de acceso a los sistemas y a los servicios de información.

Los procedimientos deben cubrir todas las etapas que comprenden al acceso del usuario, del registro inicial de nuevos usuarios al registro final de los usuarios que solo requieren el acceso a los sistemas y a los servicios de información.

Responsabilidades del Usuario

Objetivo:

Prevenir el acceso de usuario no autorizado.

La cooperación de usuarios autorizados es esencial para la seguridad afectiva.

Los usuarios deben ser enterados de sus responsabilidades de mantener controles de acceso eficaces, particularmente con respecto al uso de contraseñas y a la seguridad del equipo del usuario.

Control de Acceso a los Sistemas Operativos

Objetivo:

Prevenir el acceso no autorizado a la computadora.

Las instalaciones de la seguridad en el sistema operativo ya no se deben utilizar para restringir el acceso a los recursos de la computadora. Estas instalaciones deben ser capaces de lo siguiente:

- Identificar verificar identidad del usuario.
- Abastecimiento de los medios apropiados para la autenticación.
- Cuando sea apropiado, restringir los tiempos de conexión de usuarios.

Control de Acceso al Uso de Sistemas

Objetivo:

Prevenir el acceso no autorizado a la información en los sistemas.

Las instalaciones de Seguridad se deben utilizar para restringir el acceso dentro del uso de sistemas.

El acceso al software y a la información se debe restringir a los usuarios no autorizados

Supervisión al Acceso y Uso del Sistema

Objetivo:

Detectar actividades no autorizadas.

Los sistemas se deben supervisar para detectar cualquier anomalía del control de acceso y para registrar acontecimientos.

La supervisión del sistema permite la eficacia de controles adoptados para ser comprobado.

Desarrollo y Mantenimiento

Objetivo:

Asegurarse de que la seguridad sea sólida en los sistemas de información.

Los requerimientos de seguridad se deben identificar antes del desarrollo de los sistemas de información.

Seguridad en el Uso de los Sistemas

Objetivo:

Prevenir pérdida, la modificación o el uso erróneo de los datos del usuario en el uso de los sistemas.

Los controles apropiados se deben diseñar en el uso de los sistemas. Éstos deben incluir la validación de los datos de entrada, del proceso interno y los datos de la salida.

Control de Acceso de Red

Objetivo:

Protección de servicios de red.

Todo acceso servicios de red deben ser controlados. Esto es necesario para asegurarse de que los usuarios que tienen acceso a las redes y a los servicios de red no comprometan la seguridad de estos servicios de red. Esto se logra a través de:

- Interfaces apropiadas entre la red de la organización y las redes poseídas por otras organizaciones, o redes públicas.
- Mecanismos apropiados de la autenticación para los usuarios y el equipo.
- Control del acceso del usuario a los servicios informativos.

Seguridad de los Ficheros del Sistema

Objetivo:

Asegurarse de que la ayuda sea de una manera segura.

Todo acceso a los ficheros del sistema debe ser controlado.

La integridad del sistema debe ser mantenida por el usuario.

Seguridad de los Procesos del Desarrollo y de la Ayuda

Objetivo:

Mantener la seguridad del software del sistema y del uso de la información.

Los procesos del proyecto y de la ayuda deben ser controlados.

Los encargados responsables del uso de los sistemas deben también ser responsables de la seguridad del proyecto. Deben asegurar la seguridad del sistema.

CAPÍTULO III

ANÁLISIS DE RIESGOS

3. Análisis de Riesgos

Introducción

Dentro de este capítulo veremos cuál será la metodología que utilizaremos para el análisis del riesgo que incluye técnicas para determinar la relación entre el valor de sus activos de la información y las medidas requeridas para protegerlas.

Para establecer un programa de control eficaz, nuestro objetivo para la seguridad en la base de datos de USECAD será de la seguridad de la información. El personal que intervendrá deberá trabajar con los dueños y los usuarios de la información para encontrar la mejor mezcla de la productividad y de los controles.

Este capítulo proveerá de las herramientas necesarias para poner un proceso eficiente del análisis en ejecución del riesgo que identifique que controles serán los apropiados. El proceso permite que las organizaciones conduzcan el análisis del riesgo del sistema en una cuestión de ahorro de tiempo; semanas más bien que de meses como algunas otras metodologías.

A través de esta metodología podremos hacer un análisis completo de cómo un riesgo se puede priorizar en alto, medio o bajo nivel de impacto para el negocio y al mismo tiempo para detectar las vulnerabilidades de manera eficiente, fácil y práctica.

3.1. Metodología para el Análisis de Riesgo

Fundamentos Del Análisis Del Riesgo

Cada metodología del análisis del riesgo utiliza el mismo proceso básico. Examinaremos brevemente la metodología estándar y asistiremos en la internación de la metodología para su propia organización.

Análisis Cualitativo Del Riesgo

El análisis cualitativo del riesgo es una técnica que se puede utilizar para determinar el nivel de la protección requerido para los usos, los sistemas, las instalaciones y otros activos de la empresa.

Es una examinación sistemática de activos, de amenazas, y de vulnerabilidades que establece las probabilidades de las amenazas que ocurren, del coste de pérdidas si ocurren, y del valor de las salvaguardias o de las contramedidas diseñadas para reducir las amenazas y las vulnerabilidades a un nivel aceptable. La metodología cualitativa procura dar la prioridad solamente a los varios elementos del riesgo en términos subjetivos.

Métodos Cualitativos

Para reforzar el proceso cualitativo, los asistentes examinarán y trabajarán un número de usos prácticos de esta metodología. Incluido en este proceso sea:

- Análisis de la vulnerabilidad
- Análisis Del Impacto Del Peligro
- Análisis De la Amenaza
- Cuestionarios

3.1.1. Metodología FRAP

¿Qué es FRAP?

Es una metodología para realizar un proceso de análisis de riesgos de forma fácil (FRAP: Facilitated Risk Analysis Process por sus siglas en inglés).

¿Cómo trabaja FRAP?

- Requiere un nivel de vulnerabilidad.
- Requiere del impacto desfavorable al negocio.

El proceso facilitado del análisis del riesgo (FRAP) es una metodología conducida por el dueño del activo y conducida por un facilitador.

El FRAP utiliza metodologías cualitativas formales del análisis del riesgo para determinar las soluciones rentables para los temas, los usos o los sistemas específicos.

Este proceso no requiere ningún gasto de capital y se puede conducir por cualquier persona con buenas habilidades de la facilitación.

Es un proceso subjetivo que obtiene resultados haciendo preguntas. Los resultados del FRAP son un documento comprensivo que ha identificado riesgos, controles identificados para atenuar esos riesgos, y un plan de acción -creado por el dueño- para poner esos controles en ejecución.

¿Para que sirve FRAP?

- Identificar y priorizar los riesgos
- Determinar las acciones que deberán ser seguidas en función de los niveles de riesgo.

Ventajas Dominantes

- Identifica y da prioridad a los riesgos de su empresa
- El FRAP aprovecha a la gente que mas sabe de su organización
- Proporciona una lista de controles para atenuar los riesgos

Ventajas de FRAP vs. otras metodologías

- **En tiempo.** En el escenario de una institución mediana, un Análisis de riesgos duraría lo siguiente:
 - TRA (Treath and risk Assessment) del NIST- 6 a 9 meses.
 - ISRM (Information Security Risk Management) del GAO – 8 a 12 meses.
 - FRAP (Facilitated Risk Analysis Process) – de 2 a 4 meses.
- **En Esfuerzo.** Nivel de participación requerido del negocio y TI:
 - TRA –Muy alto.
 - ISRM- Alto
 - FRAP – Bajo
- **Acertividad Cualitativa:**
 - TRA – Alta.
 - ISRM- Muy Alta, puede ser obsoleta antes de que el esfuerzo se concluya.
 - FRAP – Alta
- **Elementos de Riesgo**
 - TRA:
 - Amenazas
 - Impacto
 - Probabilidad de ocurrencia
 - ISRM:
 - Amenazas
 - Probabilidad de ocurrencia
 - Valor y Criticidad del activo
 - Impactos
 - FRAP:
 - Vulnerabilidad
 - Impacto

Es muy importante tener en cuenta que al utilizar esta metodología detectaremos las vulnerabilidades que estén presentes en la base de datos y por lo tanto al priorizar el riesgo sabremos que tan importante es para la organización y cual será el impacto.

Como sabemos la materialización de un riesgo es el impacto al negocio, una medida del grado de daño o cambio sobre la misión de la organización y sus objetivos.

Posibles Impactos a la Organización

- Revelación de información confidencial.
- Pérdidas materiales por eventos catastróficos.
- Interrupción de los procesos críticos y no críticos de la organización.
- Insatisfacción e incumplimiento de terceras personas.
- Incumplimiento de requerimientos legales.

La materialización de los riesgos, puede resultar en:

- Acceso no autorizado.
- Revelación no autorizada de información.
- Observar y monitorear transacciones.
- Copiar sin autorización.
- Acceder a información confidencial.

3.1.2. Riesgos Transferibles o Tolerables

En el análisis de riesgos obtendremos riesgos que posiblemente serán importantes, sin embargo no siempre es parte de la empresa colocar algún control para mitigar el riesgo.

Existen dos casos para controlar el riesgo los cuales son:

- **Transferir el riesgo:** En cuanto sean identificados los riesgos, algunos de ellos podrá ser transferibles, esto será, cuando las políticas de la empresa así lo requieran, o si la organización depende de otra de mayor jerarquía, es decir, si encontramos alguna vulnerabilidad en la configuración de algún hardware o software que afecte a nuestro activo, será necesario informarlo a los responsables inmediatos, para que ellos sean los encargados de mitigar el riesgo.
- **Tolerar (aceptar) el riesgo:** En este caso al identificar algún riesgo el o los responsables de seguridad continuaran con su análisis para colocar el control óptimo al activo en cuestión y reducir el riesgo.

Al identificar los riesgos encontraremos que los podemos clasificar en tres tipos los cuales son los siguientes:

- **Riesgos de Integridad**

Se refiere a la pérdida o deficiencia en la autorización, totalidad o exactitud de la información de la organización.

- **Riesgo de Confidencialidad**

Es el riesgo de que personas no autorizadas tengan acceso a información confidencial o que sea negado al personal que si cuenta con dicha autorización.

- **Riesgo de Disponibilidad**

El riesgo de que la información no este disponible en el momento que se requiere.

3.2. Análisis y Gestión de Riesgos

Metodología Formal Del Análisis Del Riesgo.

Usando el acercamiento cualitativo y los resultados de la pre-investigación, examinaremos el método lo más popular posible hoy usado de análisis del riesgo en uso, el proceso facilitado del análisis del riesgo (FRAP).

Análisis Del Impacto Del Negocio

Es utilizado por organizaciones para determinar recursos críticos. Usando todas las técnicas discutidas, estudiaremos un proceso facilitado para repasar el impacto en procesos del negocio. Una vez que se anoten los recursos críticos, la organización puede entonces identificar controles apropiados para asegurar el negocio continúa resolviendo sus objetivos o misión de negocio. Para el análisis de riesgo son tres las etapas, las cuales se muestran en la figura 3.1

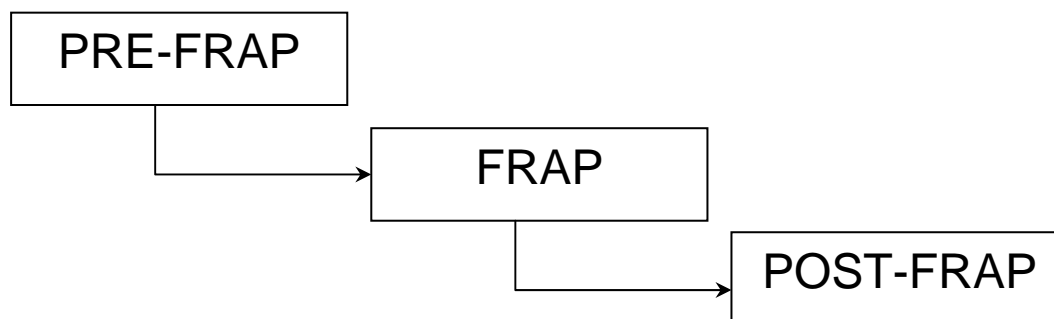


Figura 3.1 Etapas del FRAP

3.2.1. Pre-FRAP

Descripción de las etapas:

- Es una etapa clave para el éxito del proyecto.
- Son definidos, desarrollados y afinados todos los requerimientos de la sesión FRAP.
- Es necesaria e importante la participación de :
 - Administradores del negocio
 - Líder del proyecto
 - Facilitador
- Las etapas son:
 - Pre-FRAP Meeting (reunión)
 - FRAP Team (Equipo)
 - FRAP Facilitator (Facilitador)

Pre-Frap Meeting

Realizar una reunión de trabajo y se definirán los 5 componentes claves.

1.- Declaración del alcance

- Es desarrollado por los gerentes del negocio y los líderes de proyecto.
- Es la definición en palabras, que exactamente se va a revisar.

2.- Modelo visual

- Es un diagrama que ilustra el proceso a revisar.
- El modelo es utilizado durante la sesión FRAP para familiarizar al equipo FRAP con el proceso de principio a fin.

3.- Establecimiento del equipo FRAP

- Un equipo típico de FRAP tiene entre 7 y 15 miembros.
- Incluye representantes de las áreas del negocio.
- Incluye también representantes de las áreas de soporte al negocio.

4.- Mecánica de la reunión

- Es la definición de la logística de la sesión.
- Incluye, definir las salas, ubicación, material (gente, proyectores, acetatos, presentación, etc.)

5.- Acuerdo de definiciones

- La uniformidad de términos es un requisito.
- Con estos acuerdos se unifican los conceptos y hace que las divergencias sean menores durante el desarrollo del proyecto.

Durante la sesión pre-FRAP es importante discutir el proceso para priorizar los riesgos.

Existen dos maneras para hacer esto:

1. Hacer que el equipo FRAP revise todos los riesgos como sino existieran controles en sitio.
 - Esta alternativa permitirá pensar en el control “ideal”.
 - Esto permite al equipo FRAP hacer un análisis entre “cómo deben de ser” los controles y “cómo son” demostrando las vulnerabilidades.
2. Evaluar los riesgos considerando los controles en sitio.
 - La palabra clave es evaluar.
 - Quienes serán integrantes del equipo FRAP así como el numero de participantes.
 - Es recomendable incluir a las siguientes áreas en el proceso de FRAP
 - Dueños funcionales
 - Usuarios de sistema
 - Administradores de sistema
 - Analistas de sistema
 - Programadores de sistema
 - Programadores de aplicación
 - Administradores de bases de datos
 - Seguridad informática
 - Seguridad física
 - Telecomunicaciones
 - Administradores de red
 - Proveedores de servicio
 - Auditoria, Legal y Recursos Humanos (si es apropiado)

- El facilitador de la sesión debe cubrir con un número de habilidades especiales.
- Estas habilidades pueden ser mejoradas con entrenamiento especial.

Las habilidades requeridas son:

- Escuchar
 - Tener la habilidad de ser sensible a los comportamientos verbales y no verbales de los asistentes.
 - Ser capaz de aclarar respuestas.
- Liderar
 - Iniciar la sesión y motivar a los participantes a abrir discusiones enfocadas en un tópico a la vez.
- Reflejar
 - Repetir ideas con palabras frescas y con nuevo énfasis
- Resumir
 - Ser capaz de resumir temas e ideas.
- Confrontar
 - Ser capaz de retroalimentar opiniones.
 - Reaccionar honestamente a las entradas de información.
 - Ser capaz de tomar comentarios severos o ásperos y convertirlos en declaraciones positivas.
 - Ser capaz de tomar comentarios severos o ásperos y convertirlos en declaraciones positivas.
- Soportar
 - Crear un clima de confianza y aceptación
- Intervenir en Crisis
 - Ayuda a ampliar la visión de las personas con opciones y alternativas.
 - Reforzar puntos de acción que pueden resolver conflictos o crisis.

- Centrar
 - Ayuda al equipo a aceptar otros puntos de vista.
 - Generar confianza para una participación de todos.
- Resolver Problemas
 - Obtener información revelante en relación al manejo de los temas.
 - Ayudar al equipo a establecer objetivos de control específicos.
- Cambio de Comportamiento
 - Identificar aquellos participantes que en apariencia no son parte de procesos.
 - Lograr que tengan una participación activa.

Reglas básicas que el facilitador tendrá que observar:

- Observar cuidadosamente todo lo que los participantes hagan y digan.
- Reconocer todas las entradas y alertar la participación.
- Observar las respuestas no verbales
- Nunca leer, escuchar y lograr el involucramiento del grupo.
- Nunca perder el aspecto de la objetividad.
- Ser neutral (o siempre aparentar ser neutral).
- Aprender a esperar hostilidad, pero nunca ser hostil.
- Evitar ser “la autoridad experta”. El rol del facilitador es escuchar, preguntar, alentar el proceso y ofrecer alternativas.
- Adherirse a los tiempos y ser puntual.
- Utilice intermedios (descansos) para liberar una discusión.
- Esta allí para servir al equipo FRAP.
- Terminar el proceso si el grupo es indolente y difícil de controlar.

Reglas básicas que deberán seguirse en la sesión:

- Todos participan.
- Todos tienen roles identificados.
- Todos se apegarán a una agenda.
- Todas las ideas tienen el mismo valor.
- Escuchar otros puntos de vista.
- Todos los puntos son registrados.
- Registrar y postergar los puntos fuera de alcance.
- Fijar la idea antes de discutirla.

- Asegurar que todas las ideas fueron registradas por el apuntador.
- Una conversación a la vez.
- Aplicar la regla de los 3 minutos. Todas las discusiones deberán ser concluidas en un marco de tiempo acordado.
- Generalmente tiene una programación de 4 horas.
- Algunas organizaciones pueden extender este proceso hasta por tres días

Ser:

- Puntual
- Justo
- Agradable
- Creativo
- Generalmente se tiene una programación de 4 horas.
- Algunas organizaciones pueden extender este proceso hasta por tres días.

En cuanto se tenga recopilada la información detallada del pre-FRAP y se hallan llegado a los acuerdos y requerimientos necesarios se continuara con el FRAP una de las etapas más importantes.

3.2.2. FRAP

FRAP Sesión

La Sesión FRAP está dividida en cuatro secciones:

- Logística
- Lluvia de ideas
- Priorizando los riesgos
- Identificando controles

LOGÍSTICA

- El equipo FRAP se representará por sí mismo.
- Indicara Nombre, Título, Departamento y Teléfono (todo esto deberá ser registrado).
- Los roles del equipo FRAP deben ser identificados y discutidos.
- Típicamente existen 5 roles:
 - Dueño
 - Líder de proyecto
 - Facilitador
 - Apuntador

- Equipo FRAP
- El equipo FRAP deberá proporcionar una idea clara del proceso en el cual forma parte.
- Debe también exponer la declaración de alcance.
- Y algún miembro que sea del equipo de tecnología deberá describir el proceso bajo revisión del modelo visual
- Finalmente, deben revisarse las definiciones de acuerdos por todo el equipo FRAP.

LLUVIA DE IDEAS

- Una vez entendido el proceso de negocio, las definiciones clave y el alcance, el siguiente paso es la lluvia de ideas.
- Por cada elemento bajo revisión, se tendrá que identificar los riesgos que pueden impactar en la integridad, confidencialidad y disponibilidad.
- En este proceso el facilitador deberá proporcionar la definición y algunos ejemplos de riesgo.
- El equipo FRAP tendrá unos minutos para escribir los riesgos relacionados con ellos.
- El facilitador obtendrá del equipo FRAP un riesgo por cada miembro.

Aquí unos ejemplos de riesgos de a la confidencialidad.

- Acceso sin autorización.
- Revelación sin autorización.
- Observar o monitorear transacciones.
- Copiar sin autorización.
- Terceros con acceso a información confidencial.

El proceso de lluvia de ideas continuara hasta que cada elemento haya sido revisado.

Los miembros deberán revisar los riesgos que han definido con la finalidad de:

- Identificar riesgos duplicados y en algunos casos renombrar los ya definidos.
- Lo anterior dará como resultado una lista de riesgos y se tendrá lista la información para la siguiente etapa.

PRIORIZANDO LOS RIESGOS

- El equipo deberá ser instruido y concentrado en priorizar los riesgos.
- Esto será realizado mediante la determinación del nivel de vulnerabilidad y el impacto al negocio, si el riesgo ocurre.

- Las siguientes definiciones debieron ser definidas en la definición de acuerdos y presentadas al equipo en la introducción o logística.
- Vulnerabilidad Alta
 - Es una debilidad sustancial que existe en los sistemas o rutinas operacionales.
 - Son debilidades en la seguridad que representan un riesgo elevado, y que de ser explotadas pueden generar una interrupción de los servicios, o bien, proporcionar un acceso sin restricciones, o con muy pocas de ellas, a personal no autorizado.
- Vulnerabilidad Media
 - Existen algunas debilidades.
 - Son debilidades en la seguridad que por si misma no constituyen un nivel de riesgo significativo, sin embargo, al encontrarse de forma conjunta origina la posibilidad de acceso no autorizado, e inclusive interrupciones en el servicio.
- Vulnerabilidad Baja
 - Los sistemas han sido bien contruidos y operan correctamente.
 - Son debilidades en la seguridad que al ser explotadas proporcionan información confidencial, como datos de componentes tecnológicos y de usuarios, dicha información es utilizada para conocer las configuraciones y tecnología existente.
- Impacto Severo (alto)
 - Puede poner a la empresa fuera de su negocio.
- Impacto Significativo (medio)
 - Puede causar daño y costo significativo, pero la empresa puede sobrevivir.
- Impacto Menor (bajo)
 - Es un impacto operacional que tiene que ser administrado como parte de las operaciones cotidianas.

Para priorizar los riesgos existen diferentes técnicas, las tres más populares son:

1. El facilitador toma riesgo por riesgo y el equipo FRAP discute para lograr un consenso.
2. El facilitador revisa tres o cuatro riesgos para asegurar que el equipo tiene una idea correcta de cómo trabajará.
Cada miembro del equipo tiene un marcador de color y marcará sobre una tarjeta blanca la prioridad, si no hay opinión, mostrara su tarjeta en blanco.

Finalmente, el facilitador revisará la dispersión en las respuestas, y si considera pertinente discutirá el tema con el equipo FRAP para asegurarse de la respuesta más correcta.

3. La tercera técnica, puede ser utilizada si el facilitador da a cada miembro del equipo diez puntos.

Una vez finalizada la lista de riesgos y entendidas las definiciones, el facilitador podrá hacer uso del siguiente modelo para priorizar los riesgos, como se muestra en la Figura 3.2

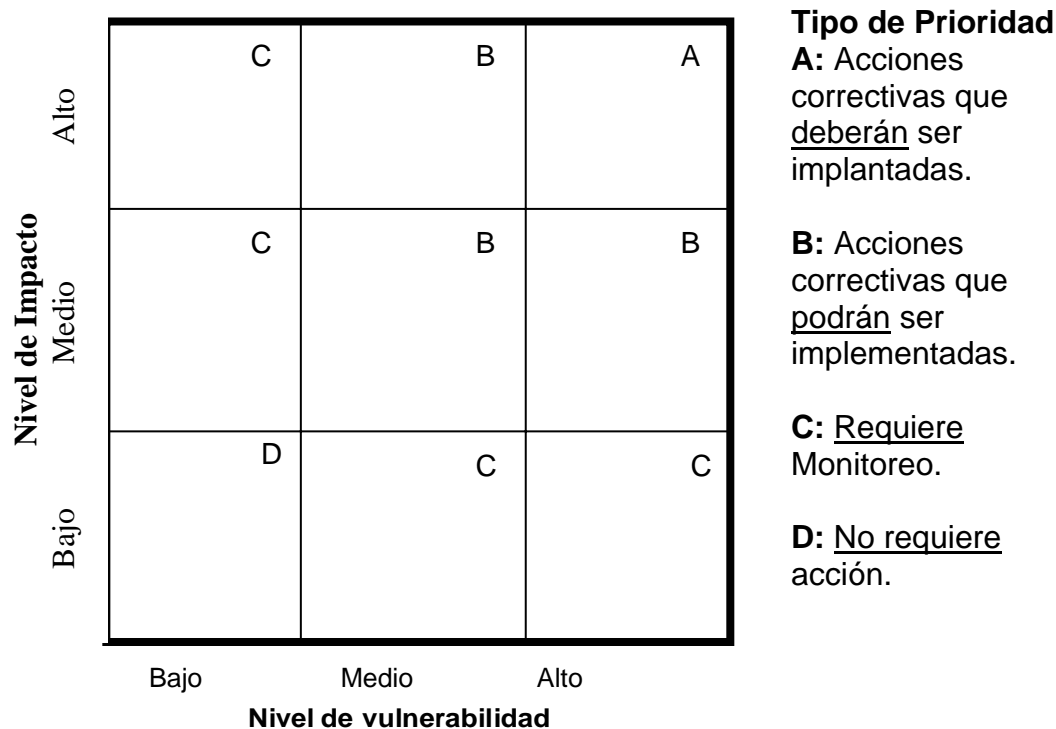


Figura 3.2 Mapa de Riesgos

Cada miembro tendrá permitido votar por diez mayores riesgos.

El resultado que veremos al finalizar la priorización de riesgos puede ser de la siguiente forma (tabla 3.1.):

#	Riesgo	Prioridad
1	La información puede ser acesada por persona que no debe tener acceso	B
2	Versiones no claras o existentes de información	B
3	Los datos pueden ser corruptos por una transacción no completada	C
4	Falla que reporte puntos de integridad	A
5	No notificación de problemas de integridad	A
6	Información autorizada en un contexto erróneo	B
7	Acceso de terceros a información confidencial	A

Tabla 3.1. Priorización de riesgos.

- La etapa final del proceso FRAP es identificar controles para los riesgos identificados.
- Para esto es posible utilizar una lista de controles previamente definidos, que deberán ser distribuidos en el equipo FRAP.
- Los controles son el punto de partida, sin embargo estos podrán ser modificados y adicionados según sea necesario.
- En la tabla 3.2. se muestra una lista de 26 controles de ejemplo desarrollados por el facilitador de FRAP.

Sin embargo los controles que nosotros utilicemos serán los que se adapten a nuestras necesidades (riesgos encontrados). Pues como ya se había mencionado anteriormente las necesidades de cada organización son diferentes.

#	CONROL	DESCRIPCIÓN DEL CONTROL
1	Back-up	Los requerimientos de respaldos deben ser determinados y comunicados al proveedor de servicios, incluyendo los requerimientos de notificación electrónica que la realización de respaldos ha sido completada y debe de ser enviada al administrador de las

		aplicaciones. El proveedor de servicios tiene que requerir las pruebas de los procedimientos de respaldos.
2	Plan de Recuperación	Desarrollar, documentar y probar los procedimientos de recuperación diseñados para asegurar que las aplicaciones e información pueden ser recuperadas, usando los respaldos creados, en caso de suceder una pérdida.
3	Control de Acceso	Implementar mecanismos de control de acceso que prevenga un acceso no autorizado a la información. Estos mecanismos deben de incluir la capacidad de detectar, registrar y reportar intentos de violación de la seguridad en esta información.
4	Control de Acceso	Origen de acceso: Implementar mecanismos para limitar el acceso a información confidencial especificada en rutas específicas de la red o en localidades físicas.
5	Control de Acceso	Implementar mecanismos de autenticación de usuarios (firewall, controles de dial-in, identificadores seguros) para limitar el acceso al personal autorizado.
6	Control de Acceso	Implementar mecanismos de encriptación (datos, end-to-end) para prevenir un acceso no autorizado y proteger la integridad y confidencialidad de la información.
7	Control de Aplicaciones	Diseño e implementación de control en aplicaciones (entrada de datos, verificación de edición, validación de requerimientos de campos, indicadores de alarma, capacidades de expiración de password, checksum) para asegurar la integridad, confidencialidad y disponibilidad de la información de las aplicaciones.
8	Pruebas de Aceptación	Desarrollar procedimientos de pruebas para ser aplicados durante el desarrollo y modificaciones de las aplicaciones existentes las cuales incluyen la participación y aceptación del usuario.
9	Administración de Cambios	Apegarse a los procesos de administración de cambios diseñados para facilitar el aprovechamiento de la estructura de las modificaciones, para asegura los pasos apropiados y cuidados a seguir. Las modificaciones de "emergencia" se deben de incluir en este proceso.
10	Antivirus	a) Asegurar que el administrador de red instale el software corporativo estándar de antivirus en todas las computadoras b) Capacitación y concienciación en las técnicas de prevención de los virus deben ser incorporadas en el programa IP de al organización.

#	CONROL	DESCRIPCIÓN DEL CONTROL
11	Políticas	Desarrollar políticas y procedimientos para limitar el acceso y privilegios de operación para que estos sean sólo necesarios para el negocio.
12	Entrenamiento	El entrenamiento del usuario deberá incluir instrucciones y documentación en el uso adecuado de la aplicación. La importancia del mantenimiento de la confidencialidad de las cuentas de usuario, passwords la confidencialidad y la naturaleza competitiva de la información debe ser acentuada.
13	Auditoria/Mantenimiento	Implementar mecanismos para el monitoreo, reporte e identificación de actividades de auditoria como revisiones de requerimientos independientes, incluyendo revisiones periódicas de identificador de usuario (user-id) para asegurar y verificar las necesidades del negocio.
14	Respaldos	Control en las operaciones: Capacitación para la realización de respaldos en los sistemas administrativos debe de ser provista conjuntamente con la rotación de funciones para asegurar la suficiencia del programa de entrenamiento
15	Entrenamiento	Control en las operaciones: Los desarrolladores de aplicaciones deben de proveer la documentación, guías y el soporte al staff de operación (proveedor de servicios) con la implementación de mecanismos para asegurar que la transferencia de información entre las aplicaciones sea de manera segura.
16	Control de Acceso	Control en las operaciones: Mecanismos para proteger la base de datos contra accesos no autorizados y modificaciones realzadas fuera de la aplicación, pueden ser determinadas e implementadas.
17	Dependencia de Interfases	Control en operaciones: Sistemas que puedan alimentar información podrían identificar y comunicarse con el proveedor de servicios para enfatizar el impacto de la funcionalidad si estas aplicaciones de alimentación no están disponibles.
18	Mantenimiento	Control en operaciones: Requerimientos en tiempos para el mantenimiento podrían seguirse y solicitar para el ajuste mientras sea comunicado al administrador si la garantía expira.

#	CONROL	DESCRIPCIÓN DEL CONTROL
19	Entrenamiento	Control de usuarios: Implementar un programa para los usuarios (evaluación del desempeño del usuario) diseñado para alentar el cumplimiento con las políticas y procedimientos en lugar de asegurar la apropiada utilización de la aplicación.
20	Acuerdos de nivel de servicio	Adquirir acuerdos de nivel de servicio para establecer niveles esperados con los clientes y de esta forma asegurarse del soporte en las operaciones.
21	Mantenimiento	Adquirir mantenimiento que sustituya los acuerdos para facilitar la continuidad en el estado de operación de la aplicación.
22	Seguridad Física	En comparación con las facilidades de administración, las facilidades de implementación de los controles de seguridad física designados para proteger la información, software y hardware son requeridos para el sistema.
23	Administración de Soporte	Requerimientos de administración del soporte para asegurar la cooperación y coordinación de las unidades de negocio para facilitar y preparar el terreno para la transición de la aplicación.
24	Propiedad	Controles sobre la propiedad
25	Estrategias Correctivas	El equipo de desarrollo deberá de desarrollar estrategias correctivas como trabajos de reprocesamiento, aplicaciones lógicas para la revisión, etc.
26	Administración de Cambios	Controles para la migración a producción, como búsquedas y remoción de procesos para asegurar que los datos almacenados estén limpios.
27	Revisión del diseño	Contar con los diagramas lógicos.

Tabla 3.2 Controles

FRAP no eliminará con la sesión todos los riesgos, la administración tiene la responsabilidad de determinar cuales riesgos deberán ser mitigados, aceptados, transferidos o evitados.

La sesión FRAP ha terminado con los siguientes entregables:

- Riesgos Identificados
- Riesgos Priorizados
- Controles identificados

3.2.3. Post-FRAP

Finalmente, el gerente del negocio, líder del proyecto y facilitador deberían trabajar en completar el plan de acción para la administración de riesgo.

La etapa Post-FRAP tiene cinco documentos entregables:

- Hoja de Referencias cruzadas
- Identificación de controles existentes
- Identificación y selección de nuevos controles para los riesgos nuevos
- Reporte final
- La hoja de referencia cruzada es una de las tareas que mas consume tiempo del facilitador y del apuntador.

Este documento toma cada control e identifica todos los riesgos que pueden ser impactados por este simple control.

Esta actividad llega a tardar hasta dos días por su nivel de análisis y complejidad.

La hoja de referencias cruzadas podrá ser como se muestra en la Figura 3.3

Control #	Control	Riesgo #		Tipo	Prioridad

Figura 3.3 Hoja de referencias cruzadas.

El informe final se muestra en la figura 3.4. El plan de acción muestra cuáles de los controles identificados durante el Análisis de Riesgos existen, y cuáles deben de ser implementados. Por lo que se deberá realizar el análisis y decisión de los controles correspondientes que se muestra en la figura 3.5.

Fecha:	04 de Mayo de 2005
Para:	Dueño de la empresa Seguridad de la información Participantes
De:
El grupo de Seguridad Informática ha facilitado una sesión de análisis de riesgo del sistema.... Los participantes del Análisis de Riesgos han identificado los riesgos y controles en el plan de acción anexo etc.....	
.....	
Fecha del FRAP:	10 de Abril de 2005
Sistema/Aplicación	Sistema Analizado (Nombre)
Facilitador	Nombre del facilitador
.....	

Figura 3.4 Hoja de Informe Final

Lea la Declaración del Entendimiento para su firma y presentación

DECLARACIÓN DEL ENTENDIMIENTO: Yo DUEÑO, entiendo que los riesgos identificados y anexados en el plan de acción de Análisis de Riesgo pueden causar que la Integridad, Confidencialidad y Disponibilidad se ve afectada de manera negativa. Yo, tendré que implementar los controles acorde con el calendario de Plan de Acción. Yo, entiendo que los riesgos no controlados pueden impactar de forma adversa la información de la compañía y del negocio.

Yo, estoy de acuerdo en enviar una copia de Plan de Acción de Análisis de Riesgo a la división de Auditoría.

_____	_____
Dueño del proceso del negocio	Fecha
_____	_____
IS Seguridad	Fecha

Figura 3.5 Análisis y decisión de los controles.

CAPÍTULO IV

IMPLANTACIÓN E INTEGRACIÓN DE LAS SOLUCIONES DE SEGURIDAD

4. Implantación e Integración de las soluciones de seguridad

Introducción

En la actualidad el avance de la tecnología es tan rápido que las empresas o instituciones tienen que estar actualizados al mantener protegidos sus activos con reglas para asegurar que su uso es ético, responsable y óptimo.

Uno de los principales activos en las instituciones es la Información y su protección es de suma importancia.

Una guía de apoyo para los responsables de este activo son los lineamientos de seguridad que protegen a la información de una serie de amenazas. Debido a que sin un análisis de riesgos es difícil mantener el control de la seguridad, en este capítulo se realizara dicho análisis con la finalidad de garantizar la continuidad de las operaciones, minimizar los daños al servidor de base de datos, maximizar los beneficios a la Comunidad, así como brindar servicio de calidad y colaborar en la difusión de una cultura de seguridad informática

En el departamento de informática son muchos los niveles de seguridad que deben de ser cuidados aún por mínimos que parezcan, sin embargo en nuestro caso de estudio solamente nos enfocaremos en la seguridad de la administración del servidor de base de datos, ya que consideramos que la información contenida en dicho servidor es fundamental para la USECAD.

Para llevar a cabo el análisis de riesgos se puede observar el capítulo 3 de esta tesis, en él se explica como se va desarrollando cada paso del análisis a través de la metodología FRAP, la cual es sencilla, eficaz y con la intención de encontrar en cada capa toda aquella vulnerabilidad que pudiese poner en riesgo la integridad, la confidencialidad y la autenticidad de la información.

Fundamentos Del Análisis Del Riesgo

PRE-FRAP Meeting.

1. Declaración del alcance.

Encontrar las posibles vulnerabilidades, así como los riesgos e impactos en la administración del servidor de base de datos de USECAD, con la finalidad de garantizar la integridad, confidencialidad y autenticidad de la información en el sistema, además de mitigar los posibles riesgos, a través de controles de aplicación que puedan guiar al administrador del SGBD con el objetivo de ampliar la seguridad en la administración de las Bases de Datos.

2. Modelo visual.

En la figura 4.1 se muestra el modelo visual, en el que se observan las etapas que se analizarán a través del análisis de riesgo.

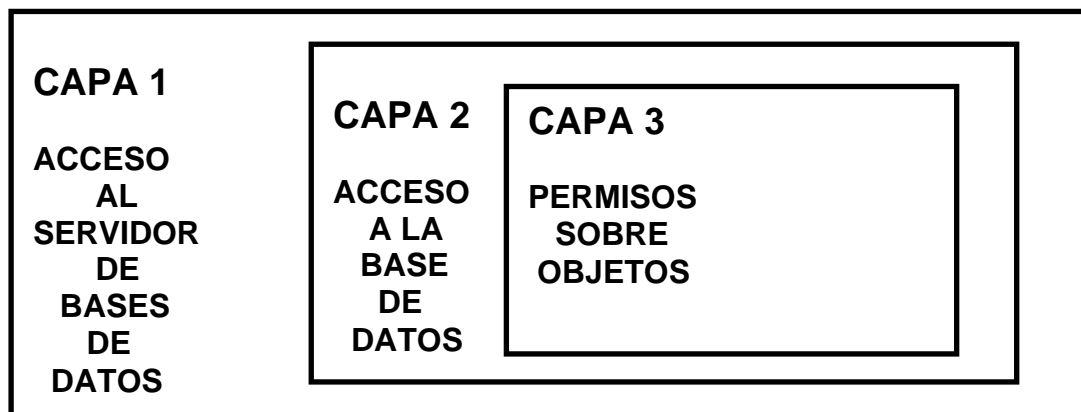


Figura 4.1 Modelo Visual del SGBD

3. Establecimiento del equipo FRAP.

Integrantes del equipo para el desarrollo del proyecto: Ing. Armado Vega, M.C. Ma. Jaquelina López, Susana Nájera, Mayelly Reynoso.

4. Mecánica de la reunión.

Reuniones: Área de informática en USECAD, Cubículo # 211 Edificio Valdez Vallejo, ubicados en Ciudad Universitaria.

Material: SGBD Sybase versión 11.5 y la base de datos primordial Base_de_Datos_1.

5. Acuerdo de Definiciones.

Los acuerdos de uniformidad de términos se basan en el capítulo 1; se basa en las definiciones de bases de datos, en el capítulo 2; se basa en los términos referentes a la seguridad informática y en el capítulo 3; se da la descripción de la metodología para un análisis de riesgo, estos serán los términos utilizados y acordados por el equipo.

FRAP

En la sesión FRAP son 4 las etapas:

1. Logística se muestra en la tabla 4.1

Nombre	Título	Departamento	Teléfono
Armando Vega Alvarado	Ingeniero	DGAE	95857365
Jaquelina López Barrientos	Maestra en Ciencias	DIE	98758425

Tabla 4.1 Logística.

Los roles serán los siguientes:

Dueño: USECAD

Líder del proyecto: Ing. Armando Vega y M. C. Jaquelina López Barrientos

Facilitador: Ing. Filiberto Manson

Apuntador y Equipo FRAP: Nájera Montiel Susana, Mayelly Reynoso Andrade.

2. Lluvia de Ideas.

En esta etapa es muy importante mencionar que al realizar la lluvia de ideas se tomaron en cuenta los riesgos identificados como duplicados o se renombraron algunos con el objetivo de tener la uniformidad de términos y finalmente se obtuvo la tabla 4.2.

ACTIVO	FACTOR DE RIESGO	POSIBLES RIESGOS
Servidor de Base de Datos	Falta de políticas	Violaciones a la seguridad
	Falla de Base de Datos	Inconsistencia en los datos
	Mala configuración de backups	Datos sin backups Inconsistencia y redundancia de datos
	Mala distribución de privilegios	Pérdida de tiempo y productividad
	Falta de limitación de recursos	Explotación de información

Tabla 4.2 Tabla de identificación de riesgos

Priorizando los Riesgos.

Basándonos en el capítulo 3 donde se describe cada una de las vulnerabilidades los riesgos se priorizaran en los siguientes niveles:

- Vulnerabilidad Alta
- Vulnerabilidad Media
- Vulnerabilidad Baja

Así como los impactos:

- Impacto Severo (alto)
- Impacto Medio
- Impacto Bajo

A continuación se muestra la figura 4.2 donde se hace una representación de las vulnerabilidades y amenazas al servidor de Base de Datos.

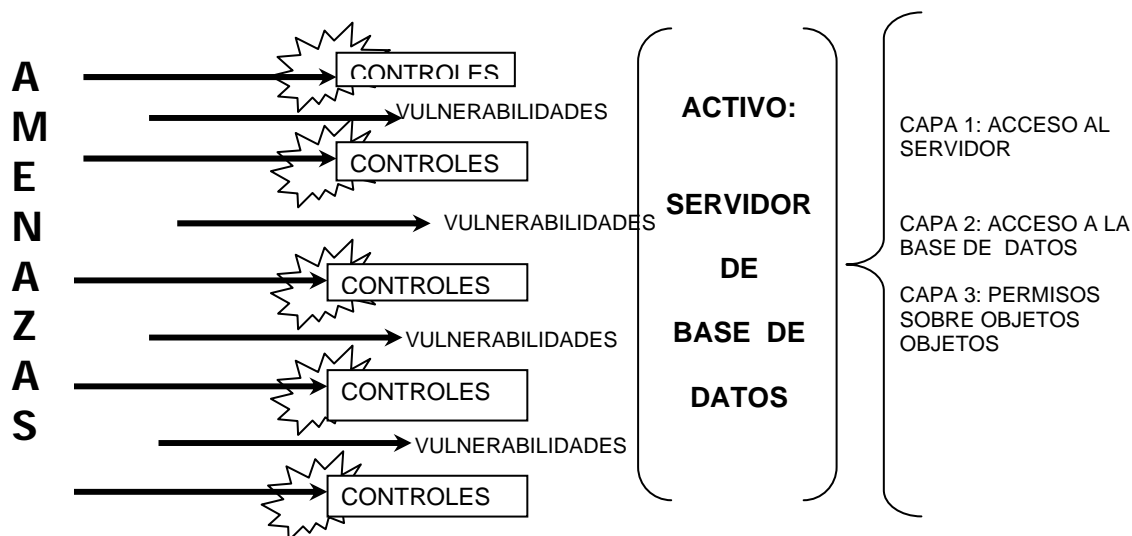


Figura 4.2 Representación de vulnerabilidades del servidor

Se muestra la priorización de riesgos y el nivel de impacto posible, que se obtuvo para el sistema y se muestran en la tabla 4.3.

RIESGOS	IMPACTO AL NEGOCIO	NIVEL DE VULNERABILIDAD	TIPO DE PRIORIDAD	CONCLUSIÓN DEL RIESGO
Falta de políticas	B	A	C	B
Falla de la Base de Datos.	A	A	A	A
Mala configuración de backups	A	M	B	B
Mala distribución de privilegios	A	A	A	A
Falta de limitación de recursos	A	B	C	B

Tabla 4.3 Se muestra la priorización de riesgos y el nivel de impacto posible

A continuación en las figuras 4.3 y 4.4 se encuentran los Mapas de Riesgos basados en la tabla anterior:

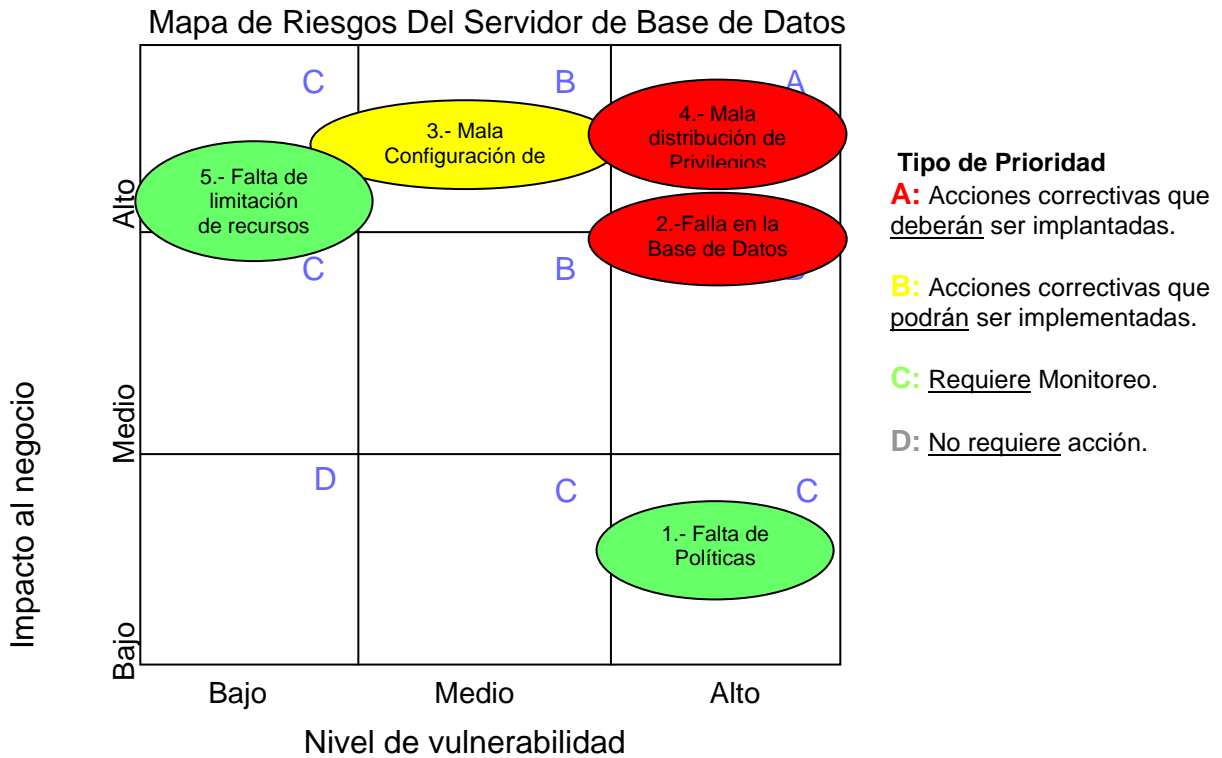


Fig. 4.3 Mapa de riesgos

Mapa de Riesgos Del Servidor de Base de Datos Para la Implementación.

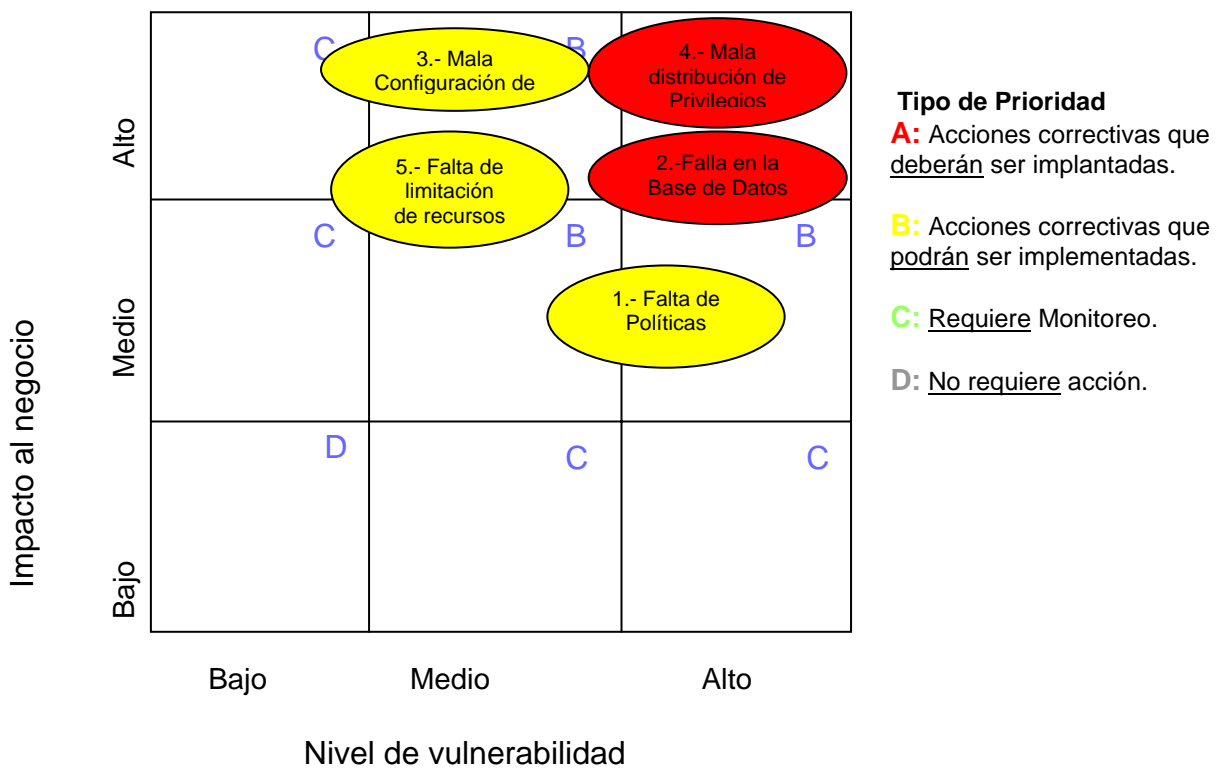


Fig. 4.4 Mapa de riesgos para la implementación

Estudio de Controles para cada Capa.

Después de elaborar los mapas de riesgos veremos cuales son los controles para cada una de las capas que propusimos en el modelo visual, con el objeto de garantizar que los controles sean óptimos.

4.1 Seguridad en Capa 1

4.1.1 Control de Acceso al Servidor de Base de Datos

El control de acceso es fundamental para mantener la seguridad del sistema, por tal motivo se realizó el análisis para el acceso al servidor de base de datos de USECAD.

Para el desarrollo del análisis del sistema se obtuvieron las tablas más críticas del sistema de la base de datos Base_de_Datos_1 la más importante del sistema, las cuales se muestran a continuación:

- TABLA_0
- TABLA_1
- TABLA_2
- TABLA_3
- TABLA_4
- TABLA_5
- TABLA_6
- TABLA_7
- TABLA_8
- TABLA_9
- TABLA_10
- TABLA_11
- TABLA_12
- TABLA_13
- TABLA_14
- TABLA_15
- TABLA_16
- TABLA_17
- TABLA_18
- TABLA_19
- TABLA_20

Introducción a los procedimientos del sistema

Los procedimientos del sistema, creados por **installmaster** en la instalación, se ubican en la base de datos *sybssystemprocs* y son propiedad del administrador del sistema, pero muchos de ellos pueden ejecutarse desde cualquier base de datos.

Uso de procedimientos del sistema

Si un valor de parámetro para un procedimiento del sistema contiene signos de puntuación o espacios en blanco incrustados, o es una palabra reservada, debe incluirse entre comillas simples o dobles. Si el parámetro es un nombre de objeto calificado por un nombre de base de datos o un nombre de propietario, incluya el nombre completo entre comillas simples o dobles.

Nota: Se recomienda no utilizar identificadores delimitados como parámetros de procedimientos del sistema, ya que pueden producir resultados inesperados.

Todos los procedimientos del sistema se ejecutan en el nivel de aislamiento 1.

Todos los procedimientos del sistema producen un estado de retorno. Por ejemplo:

```
return status = 0
```

Significa que el procedimiento se ha ejecutado satisfactoriamente.

Es posible crear procedimientos del sistema propios para ejecutarse desde cualquier base de datos.

Valores para parámetros opcionales

Si un procedimiento tiene varios parámetros opcionales, es posible suministrar parámetros con el formato:

```
@ parametername = value
```

En lugar de suministrar todos los parámetros. Los nombres de parámetros en las instrucciones de sintaxis coinciden con los nombres de parámetros definidos por los procedimientos.

La sintaxis para `sp_addlogin` es:

```
sp_addlogin login_name , password [, defdb  
[, deflanguage [, fullname ]]]
```


Creación de Logines:

Para tener acceso al servidor es necesario dar el acceso a los usuarios, para lo cual hay que crear el login de cada usuario, un login es el que permite entrar en un sistema informático. Este nombre, clave o keyword con el que una persona está registrada en un determinado ordenador o aplicación, en nuestro caso es el acceso al servidor de base de datos. A continuación mostramos información importante para la creación de logines; así como opciones de seguridad para mantener la confidencialidad de las contraseñas tanto para el sistema como a los usuarios, así por lo tanto dependerá de la política de la institución el nivel de seguridad que se quiera brindar.

Para dar de alta a un login se utiliza el siguiente procedimiento del sistema:

sp_addlogin

- **Función**

Añade una nueva cuenta de usuario a SQL Server.

- **Sintaxis**

```
sp_addlogin loginame , passwd [, defdb [,deflanguage [, fullname ]]]
```

- **Parámetros**

loginame

Es el nombre de login del usuario. Los nombres de login deben cumplir con las reglas correspondientes a los identificadores. Se recomienda encarecidamente que los nombres de login a SQL Server de los usuarios sean los mismos que sus nombres de login del sistema operativo. Esto facilita la conexión a SQL Server, simplifica la administración de las cuentas de login del servidor y sistema operativo, y facilita la correlación de los datos de auditoría generados por SQL Server y el sistema operativo.

passwd

Es la contraseña del usuario. Las contraseñas deben tener al menos seis bytes. Si se especifica una contraseña más corta, **sp_addlogin** devuelve un mensaje de error y se cierra. Incluya las contraseñas que contengan caracteres distintos de A-Z, a-z o 0-9 entre comillas. Asimismo, incluya entre comillas las contraseñas que **comiencen** con 0-9.

defdb

Es el nombre de la base de datos predeterminada asignada cuando un usuario hace el login a SQL Server. Si no especifica *defdb*, el valor predeterminado será *master*.

deflanguage

Es el nombre oficial del idioma predeterminado asignado cuando un usuario hace el login a SQL Server. Si no se especifica *deflanguage*, se usará el idioma predeterminado del servidor, definido por el parámetro de configuración **default language id**.

fullname

Es el nombre completo del usuario que es propietario de la cuenta de login. Puede usarse con fines de documentación e identificación.

- **Comentarios**

- Para facilitar la administración, se recomienda encarecidamente que todos los nombres de login de los usuarios de SQL Server sean los mismos que sus nombres de login del sistema operativo.
- Tras asignar una base de datos predeterminada a un usuario con **sp_addlogin**, el propietario de la base de datos o el administrador del sistema deberá proporcionar acceso a la base de datos ejecutando **sp_adduser** o **sp_addalias**.
- Aunque un usuario puede utilizar **sp_modifylogin** para cambiar su propia base de datos predeterminada en cualquier momento, no es posible utilizar una base de datos sin permiso del propietario de la base de datos.
- En cualquier momento, un usuario puede utilizar **sp_password** para cambiar su propia contraseña. Un oficial de seguridad del sistema puede usar **sp_password** para cambiar la contraseña de cualquier usuario.
- Un usuario puede utilizar **sp_modifylogin** para cambiar su propio idioma predeterminado. El administrador del sistema puede usar **sp_modifylogin** para cambiar el idioma predeterminado de cualquier usuario.
- Un usuario puede utilizar **sp_modifylogin** para cambiar su propio *fullname*. El administrador del sistema puede utilizar **sp_modifylogin** para cambiar el *fullname* de cualquier usuario.

- **Mensajes**

- ' *loginame* ' no es un nombre válido. *loginame* debe cumplir con las reglas correspondientes a los identificadores.
- ' *deflanguage* ' no es un nombre de idioma oficial de syslanguages. Use **sp_helplanguage** para determinar los idiomas alternativos disponibles. Añada un idioma alternativo con **langinstall**, o especifique *us_english*.
- No se puede ejecutar **sp_addlogin** dentro de una transacción. **sp_addlogin** modifica las tablas del sistema, de modo que no es posible ejecutarlo dentro de una transacción.
- Ya existe un usuario con el nombre de login especificado. Seleccione otro *loginame* . Si sólo desea cambiar la contraseña, la base de datos predeterminada o el idioma predeterminado del usuario, utilice **sp_password** o **sp_modifylogin** .

- Nombre de base de datos inválido -- no se añadió el login. La base de datos predeterminada especificada no existe. Cree primero la base de datos o seleccione una base de datos que ya exista.
- Se creó un nuevo login.

- **Permisos**

Sólo un oficial de seguridad del sistema puede ejecutar **sp_addlogin** .

En seguida mostramos los lineamientos para creación de las contraseñas (Password).

Objetivo

Generar una cultura de seguridad para los usuarios y administradores a través de un estándar para la generación de contraseñas fuertes y confiables.

Responsable para el cumplimiento del lineamiento

DBA, SSO.

Alcance

Incluye a todo el personal que tiene cualquier forma de acceso y requiere de una contraseña para acceder al SGBD.

Lineamientos de construcción de contraseñas

1.- Características de contraseñas débiles, cuando:

- La contraseña contienen menos de 8 caracteres
- La contraseña es una palabra de diccionarios (ya sea español o extranjero)
- La contraseña es una palabra de uso común:
 - Nombre de familiares, animales domésticos, amigos, compañeros de trabajo, caracteres de fantasía, etc.
 - Términos de computadora y nombres, sitios, compañías, software hardware.
 - Cumpleaños y otra información personal como direcciones y números telefónicos.
 - Palabra o modelos de número como aaabbb, qwerty, zyxwvuts, 123321, etc.
 - Cualquiera de los susodichos deletreados hacia atrás.
 - Cualquiera de los susodichos precedidos o seguido de un dígito (p. ej, secret1, 1secret)

Características de contraseñas fuertes, cuando:

- Contiene ambos caracteres mayúscula y minúscula (por ejemplo, un-Z, UN-Z)
- Tiene los dígitos y signos especiales, 0-9! @#\$%^&*()_+|β-=\ç{}[]: ";' <> ?, . /)
- Al menos tienes ocho caracteres alfanuméricos largos.

- No es una palabra en ningún idioma, tecnicismo o dialecto.
- No están basados en la información personal, los nombres de la familia, etc.
- Las Contraseñas nunca se deben anotar ni almacenados en línea. La prueba para crear las contraseñas que se puede recordar fácilmente. De un solo sentido para hacer esto es crea una contraseña basada en un título de la canción, en la afirmación, o en otra frase. Por ejemplo, la frase quizás sea: "Este mayo Es de un solo sentido para Recordar" y la contraseña podría ser: "TmB1w2R!" o "Tmb1W>rß" o alguna otra variación.

Estandar

- Prohibido usar la misma contraseña para la de acceso al SGBD que en otras cuentas. Dentro de sus posibilidades, cambie sus contraseñas para varias necesidades de diferentes tipos de acceso.
- Principalmente y definitivamente prohibido compartir las contraseñas, ya sean colaboradores cercanos, secretarios y personal de extrema confianza. Todas las contraseñas serán tratadas como sensibles y de información confidencial del SGBD.
- Lo que ningún usuario podrá hacer:
 - a) Prohibido revelar una contraseña a través de la línea telefónica.
 - b) Prohibido revela una contraseña en un mensaje de correo electrónico.
 - c) Prohibido revela una contraseña al jefe.
 - d) Prohibido habla de una contraseña delante de otros.
 - e) Prohibido expresar el formato de una contraseña (p.ej, "mi apellido").
 - f) Prohibido revelar una contraseña en formas de seguridad o cuestionarios.
 - g) Prohibido compartir una contraseña con miembros de la familia.
 - h) Prohibido revelar una contraseña a compañeros de trabajo por inasistencia así se justificada o injustificada.
- En caso de que alguien exija una contraseña, mándelos a este documento o hacer que ellos llamen alguien al responsable del sistema DBA o SSO.
- No utilice el "Recordar la Contraseña" (p. ej, Eudora, Outlook, Netscape Messenger).
- Otra vez, no anote contraseñas y almacénelos en todas partes en su oficina. No almacene contraseñas en un archivo en NINGÚN sistema de computadora (incluso Pilotos de Palma o dispositivos similares) sin codificación.
- Cambie las contraseñas por lo menos una vez cada seis meses (excepto contraseñas de nivel de sistema que deben ser cambiadas cada tres meses). El intervalo recomendado del cambio es cada cuatro meses.

- Si se sospecha que una cuenta o la contraseña ha estado comprometida, informan el incidente al responsable del sistema para que sean cambiadas todas las contraseñas.
- Cuando se realice alguna revisión periódica a través de algún mecanismo y se encuentre que alguna contraseña es débil se le pedirá al usuario fortalecerla.

A. Estándares de Desarrollo de Aplicación

- Los desarrolladores de aplicaciones deben asegurar que sus programas contienen las precauciones de seguridad siguientes.
 - a) Deberá apoyar la autenticación de usuarios individuales, no grupos.
 - b) No debería almacenar contraseñas en el texto claro o en ninguna forma fácilmente reversible.
 - c) No habrá ningún motivo para revelar contraseñas aun se tengan que emplear funciones de otro usuario.

B. Uso de Contraseñas para Usuarios de Acceso Remotos

El acceso a la red de la UNAM vía el acceso remoto debe ser controlado usando una autenticación de contraseña antigua o un sistema clave público/privado fuerte.

Amonestación

Cualquier empleado que viole estos lineamientos puede ser sujeto a una acción disciplinaria, hasta incluso la terminación del empleo.

Es importante mencionar que posteriormente a la creación de los *logines* el administrador tendrá grandes beneficios, ya que podrá llevar a cabo una buena y segura administración al crear grupos de trabajo, utilizando la limitación de recursos que veremos en seguida.

4.1.2 Limitación de Recursos de la Base de Datos

Con la finalidad de no dañar a otros usuarios y mantener la integridad y confidencialidad de los datos se mostraran los comandos, parámetros y características para implementar la limitación de recursos, la cual es una propuesta para el administrador del servidor de base de datos.

Regulador de recursos

Se discutirá la implementación y las características del regulador de recursos “test”.

Objetivos:

- Explicar las características del regulador de recursos.
- Autorizar el regulador de recursos.
- Crear los límites que están activados por todo el tiempo.

Descripción:

- El regulador de recursos suministra un camino para el sistema administrador a poner límites en los recursos del ASE.
- Los tipos de límites abarcan:
 - Costos I/O (estimados y actuales).
 - El tiempo transcurrido (de un grupo o transacción).
 - El número de filas.
- Acciones a tomar en violaciones:
 - Advertencias.
 - Abortar los grupos de preguntas.
 - Abortar la transacción.
 - Matar la sesión del usuario.
- Los límites de recursos pueden ser activados a diferentes tiempos.

Costos I/O

- Ambos físico y lógico I/O son incluidos en el cálculo de optimización de costos de I/O.

Motivación.

- Prevenga las preguntas fugitivas en el nivel de la aplicación o la entrada

Acceso

- El SA puede limitar recursos al igual una entrada o aplicación.
- Los nombres de aplicación pueden ser suministrados en los paquetes de entrada.

Beneficios.

- Poner restricciones base en entradas individuales en aplicaciones.
- La habilidad de crear las gamas denominadas de tiempo durante la cuál las restricciones son vigentes.

Permitir los límites de recurso en ASE.

- El ASE debe ser configurado para permitir los límites de recursos.
- Usar los límites de recursos permitidos en la opción de configuración.

“allow resource limits”

```
1>sp_configure “allow resource limits”, 1
2>go
```

sp_configure

- **Función**

Muestra o cambia parámetros de configuración.

- **Sintaxis**

```
sp_configure [ configname [ configvalue ] | group_name |
non_unique_parameter_fragment ]
```

```
sp_configure "configuration file", 0, {"write" | "read" | "verify" | "restore"} " file_name "
```

- **Parámetros (mostrados en la tabla 4.4)**

Sintaxis	Efecto
sp_configure	Muestra todos los parámetros de configuración por grupo, sus valores actuales, sus valores predeterminados, el valor más reciente definido para ellos y la cantidad de memoria que este valor determinado utiliza.
sp_configure configname	Muestra el valor actual, el valor predeterminado, el valor cambiado más recientemente y la cantidad de memoria utilizada por el valor para todos los parámetros que coinciden con el parámetro.
sp_configure configname, configvalue	Reinicializa <i>configname</i> a <i>configvalue</i> .
sp_configure configname, 0, "default"	Reinicializa <i>configname</i> a su valor predeterminado.
sp_configure group_name	Muestra todos los parámetros de configuración en <i>group_name</i> , sus valores actuales, sus valores predeterminados, el valor (si corresponde) más reciente definido para ellos y la cantidad de memoria que este valor determinado utiliza.
sp_configure non_unique_parameter_fragment	Muestra todos los nombres de parámetro que coinciden con <i>non_unique_parameter_fragment</i> .

Sintaxis	Efecto
sp_configure "configuration file", 0, "write", " file_name "	Crea <i>file_name</i> a partir de la configuración actual. Si <i>file_name</i> ya existe, se escribe un mensaje en el diario de errores y el archivo existente cambia de nombre utilizando la convención <i>file_name.001</i> , <i>file_name.002</i> , etc. Observe que si ha cambiado un parámetro estático pero no ha reiniciado el servidor, "write" proporciona el valor que se ejecuta actualmente para ese parámetro.
sp_configure "configuration file", 0, "read", " file_name "	Realiza la verificación de validación para valores contenidos en <i>file_name</i> y lee los valores que pasan la validación en el servidor. Si falta algún parámetro de <i>file_name</i> , se utilizan los valores que se ejecutan actualmente para esos parámetros.
sp_configure "configuration file", 0, "verify", " file_name "	Realiza la verificación de validación para los valores de <i>file_name</i> .
sp_configure "configuration file", 0, "restore", " file_name "	Crea <i>file_name</i> con los valores de <i>sysconfigures</i> . Esto resulta útil si se han perdido todas las copias del archivo de configuración y necesita generar una copia nueva.

Tabla 4.4 Parametros de sp_configure

- **Comentarios**

- Cualquier usuario puede ejecutar **sp_configure** para mostrar información sobre parámetros y sus valores actuales, pero no puede modificar parámetros. Los administradores del sistema pueden ejecutar **sp_configure** para cambiar valores de parámetros de configuración específicos. Sólo los oficiales de seguridad del sistema pueden ejecutar **sp_configure** para modificar los parámetros **systemwide password expiration** , **audit queue size** , **allow updates to system tables** y **allow remote access**.
- Al ejecutar **sp_configure** para modificar un parámetro dinámico:
 1. Se actualizan los valores de configuración y de ejecución.
 2. Se actualiza el archivo de configuración.
 3. El cambio entra en efecto inmediatamente.
- Al ejecutar **sp_configure** para modificar un parámetro estático:
 4. Se actualiza el valor de configuración.
 5. Se actualiza el archivo de configuración.
 6. El cambio sólo entra en efecto al reiniciar SQL Server.

- Si se ejecuta sin parámetros, **sp_configure** muestra todos los parámetros de configuración por grupo, sus valores actuales, sus valores predeterminados, el valor (si corresponde) más recientemente definido para ellos y la cantidad de memoria que este valor determinado utiliza en un informe de cuatro columnas, como se indica a continuación:
 - o La columna *default* muestra el valor suministrado con SQL Server. Si no se reconfigura un parámetro explícitamente, éste conserva su valor predeterminado.
 - o La columna *memory used* muestra la cantidad de memoria utilizada por el parámetro y su valor actual. Algunos parámetros relacionados se extraen del mismo banco de memoria. Por ejemplo, la memoria utilizada para **stack size** y **stack guard size** ya está contabilizada en la memoria utilizada para **number of user connections** . Si ha añadido la memoria que utiliza cada uno de estos parámetros por separado, la cantidad total será mayor que la cantidad realmente utilizada. En la columna *memory used* , los parámetros que "comparten" memoria con otros parámetros se marcan con un símbolo de número ("#").
 - o La columna *config_value* muestra el valor más reciente definido por el parámetro de configuración con **sp_configure** .
 - o La columna *run_value* muestra el valor que SQL Server está utilizando. Este cambia después de modificar un valor del parámetro con **sp_configure** (y, para parámetros estáticos, reinicie SQL Server). Este es el valor en *syscurconfigs.value* .
- **En el anexo 1 se muestra la lista de parámetros de configuración.**
- **Mensajes**
 - No existe la opción de configuración. El nombre suministrado como parámetro *configname* es desconocido.
 - La opción de configuración no es única. El nombre suministrado como parámetro *configname* no es exclusivo. No se ha cambiado ningún parámetro de configuración. Por ejemplo, dos de los parámetros de configuración son **recovery interval in minutes** y **print recovery information** . El uso de **recovery** para el parámetro *configname* genera este mensaje porque coincide con ambos nombres. Los nombres completos que coinciden con la cadena de caracteres suministrada se imprimen para poder especificar *configname* con más detalle.
 - El valor de la opción de configuración no es válido. El *configvalue* suministrado no se encuentra en el rango de valores permitidos para el parámetro de configuración especificado. Para obtener una visualización de los valores permitidos, vuelva a ejecutar **sp_configure** con el nombre del parámetro de configuración como el único parámetro. Un *configvalue* 0 siempre es válido. Este indica a SQL Server que defina el valor de configuración como su valor predeterminado.
 - No se puede definir el número de dispositivos como menor que el número de dispositivos ya definidos en *sysdevices*. Utilice **sp_helpdevice** para ver una lista de los dispositivos definidos para este servidor.

- No se puede ejecutar `sp_configure` dentro de una transacción. **sp_configure** modifica tablas del sistema, por lo que no puede ejecutarse dentro de una transacción.
- No se puede definir el idioma predeterminado como una ID de idioma no definida en `syslanguages`. Utilice **sp_helplanguage** para ver la lista de nombres de idiomas oficiales disponibles en este SQL Server.
- Máximo de descriptores de archivos o cuota de proceso `FILLM` demasiado baja como para soportar el número de conexiones de usuario solicitadas. La variable 'user connections' no se modificará. Utilice este comando:

```
select @@max_connections
```

Para hallar el valor máximo para el que puede configurarse **user connections**.

- **Permisos**

Cualquier usuario puede visualizar valores de parámetros de configuración ejecutando **sp_configure** sin parámetros o sólo con el primer parámetro (*configname*). Un administrador del sistema puede modificar parámetros de configuración ejecutando **sp_configure** con ambos parámetros, excepto para los parámetros **systemwide password expiration**, **audit queue size**, **allow updates to system tables** y **allow remote access**. Sólo un oficial de seguridad del sistema puede definir estos parámetros.

Los administradores de sistema pueden modificar todos los otros parámetros.

“allow resource limits”

- Señales de el servidor a asignar memoria interna por:
 - rangos de tiempo
 - alarma interna del servidor
 - rangos de aplicación y límites para las sesiones de usuario.
- El parámetro es estático: reinicie el servidor después del escenario.
- Por defecto es 0 (off)
- Miembro de servicio administración y configuración de grupos.

Creando un límite de recursos.

Un límite de recursos es creado usando el procedimiento almacenado

sp_add_resource_limit

- **Función**

Creará un límite en el número de recursos de servidor que pueden ser usados por una entrada al sistema del Adaptive Server y/o una aplicación para ejecutar un *query*, *query batch*, o transacción.

- **Sintaxis**

```
sp_add_resource_limit name, appname, rangename, limittype, limitvalue  
[, enforced [, action [, scope ]]]
```

- **Parámetros**

name

Es la entrada al sistema de *Adaptive Server* a la cual se le aplica el límite. Se debe especificar *name* o un *appname* o ambos. Crear un límite esto se aplica a todos los usuarios de una aplicación particular, especificar un *name* de *NULL*.

appname

Es el nombre de la aplicación a la cual el límite se aplica. Se debe especificar *name* o un *appname* o ambos. Crear un límite esto se aplica a todas las aplicaciones usadas por una entrada al sistema de *Adaptive Server*, especificar un *appname* de *null*. Crear un límite que se aplica a una aplicación particular, especifique el nombre de aplicación que el programa de cliente pasa al *Adaptive Server* en el paquete de entrada al sistema.

rangename

Es la variedad de tiempo durante la cual el límite es hecho cumplir. La variedad de tiempo debe existir en el *sysrangeranges* tabla de sistema de *master* en la base de datos de tiempo usted crea el límite.

limittype

Es el tipo de recurso a limitar. Este debe ser uno de los que se muestran en la tabla 4.5:

Tipo de limite	Descripción
row_count	Limita el número de filas que una pregunta puede devolver
elapsed_time	Limita el número de segundos, en el tiempo <i>wall-clock</i> , que un query batch o transacción pueden dirigir
io_cost	Limita costo actual o la estimación de costos de optimizacion para procesar un query

Tabla 4.5 Tipo de recurso a limitar

limitvalue

Es la cantidad máxima del recurso de servidor (costo de entrada - salida, pasó el tiempo en segundos, o cuenta de fila) que puede ser usado por la entrada al sistema o aplicación antes de que el *Adaptive Server* haga cumplir el límite. Este debe ser un aspecto positivo, el número entero distinto a cero que es menos o igual a 231.

La tabla 4.6 indica qué valor se especifica para cada tipo de límite:

Tipo de Limite	Valor de limite
row_count	El número máximo de filas que pueden ser devueltas por un query antes que el límite es cumplido.
elapsed_time	Número de segundos, en el tiempo <i>wall-clock</i> que un query batch o transacción pueden dirigir antes que el límite sea cumplido.
io_cost	Es una medida de <i>unitless</i> se derivó de la fórmula de presupuesto de optimizar la formula de costos

Tabla 4.6 Valor que se especifica para cada tipo de límite

enforced

Determina si el límite es hecho cumplir antes de o durante un query de ejecución. La tabla 4.9 pone los valores válidos en una lista para cada tipo de límite:

Si se especifica un *enforced* de 3, el *Adaptive Server* realiza un lógico "o" de 1 y 2. Por ejemplo, asuma *enforced* es puesto a 3. Si usted dirige un query cuyo *io_cost* excede el costo estimado, el especificado la acción es ejecutada.

Si la pregunta es dentro de los límites especificados para el costo estimado pero excede el costo actual, la *action* especificada es también ejecutada.

Si no se especifica un *enforced*, el *Adaptive Server* hace cumplir el límite 2 para *row_count* y *elapsed_time* y límite 3 para *io_cost*. En otras palabras, si el tipo de límite es *io_cost*, la acción especificada es ejecutada si el query excede el costo estimado o actual.

action

Es la acción a tomar cuando el límite es excedido. En la tabla 4.10 se muestra los codigos de acción que son válidos para todos los tipos de límite:

Si usted no especifica un valor de *action*, el *Adaptive Server* utiliza un valor predefinido de 2 (Abortar el query batch).

scope

Si usted no especifica un valor de *scope*, el límite se aplica a todos los posibles *scopes* para el tipo de límite, esto se muestra en la tabla 4.7.

Codigo scope	Descrpcion	Tipo de limite
1	Query	io_cost row_count
2	Query batch (one or more SQL statements sent by the client to the server)	elapsed_time
4	Transaccion	elapsed_time
6	Query y Query batch	elapsed_time

Tabla 4.7 Codigos de scope que son válidos para todos los tipos de límite

- **Uso**

- Usted debe permitir *sp_configure "allow resource limits"* para el recurso límites para entrar en vigor.
- Límites de recurso Múltiples puede existir para un usuario dado, aplicación, el tipo de límite, el alcance, y el tiempo de imposición, mientras sus variedades de tiempo no hacen traslapo.
- Todos los límites para el tiempo llamado actualmente activo se extiende y el “*at all times*” variedad para una entrada al sistema y/o nombre de aplicación están ligados a la sesión del usuario en tiempo de entrada al sistema. Por lo tanto, si un usuario registra en el *Adaptive Server* independientemente de una aplicación dada, límites de recurso esto restrinja al usuario en la combinación con aquella aplicación no se aplican. A restricciones de garantía contra aquel usuario, cree un límite de recurso que es específico al usuario e independiente de cualquier aplicación.

- Desde que el nombre de la entrada de usuario o nombre de aplicación, o ambos, son utilizados para identificar un límite del recurso, el *Adaptive Server* observa un predefinido de la búsqueda al escudriñar la tabla de *sysresource_limits* para límites aplicables para una sesión de la entrada. La tabla 4.8 describe la precedencia de emparejar ordenó los pares del nombre de la entrada y el nombre de la aplicación:

Nivel	Nombre de entrada	Nombre de aplicación
1	"joe_user"	payroll
2	NULL	Payroll
3	"joe_user"	NULL

Tabla 4.8 Precedencia de emparejar para *sp_add_resource_limit*

- Si uno o varios partidos son encontrados para un nivel de precedencia dado, ningunos niveles adicionales se buscan. Esto previene los conflictos con respecto a límites semejantes para combinaciones diferentes de entrada/aplicación. Si ningún partido es encontrado en algún nivel, ningún límite es impuesto a la sesión.
- Cuando usted agrega, borra, o modifica los límites del recurso, *Adaptive Server* liga de nuevo los límites para cada sesión para aquella entrada al sistema y/o aplicación en el principio del siguiente grupo de preguntas (query) para esa sesión.
- Cuando usted cambia las variedades de tiempo actualmente activas, el *Adaptive Server* liga de nuevo límites para la sesión. Esta nueva encuadernación ocurre al principio de la siguiente del siguiente query batch.
- Usted no puede asociar los límites para una entrada al sistema particular, aplicación, o combinación de entrada al sistema/aplicación con variedades de tiempo llamadas aquel traslapo (excepto límites que comparten la misma variedad de tiempo).

- **Permisos**

*Sólo un administrador de sistema puede ejecutar *sp_add_resource_limit*.*

- Creando un límite de recursos de jerarquía.

Un limite de recurso de jerarquía puede ser creado asignando limites a la aplicación y entradas por separado.

Paso 1

- Asignar límites a las salidas de aplicaciones.

Paso 2

- Asignar limites a las entradas individuales

Paso 3

- Asignar límites de recursos específicos a los usuarios individuales usando aplicaciones específicas.

Comentarios

ASE chequea los límites de recursos contra:

- Combinación de aplicaciones y entradas, paso 1.
- Aplicación de nombre, paso 2.
- Nombre de la entrada, paso 3.

Esta jerarquía suministra tres niveles de límites de recursos protección por cada entrada en el sistema.

Rango de tiempo en el límite de recursos (*rangename*).

- El tiempo de rango durante el cual el límite es respetado.
- Es creado por el procedimiento almacenado *sp_add_time_range*
- Por defecto la duración del rango es nombrado *at all time* (por todo el tiempo).
 - Cubrir todo el tiempo, del primer día de la semana a través de la última, de 00:00 a través de 23:59.
 - No puede ser borrada o modificada.
- Creando, inclinando y modificando nombre, rangos de tiempo son discutidos después.

Los tipos de límites de recursos. (*limittype*)

- Costos I/O
 - *Showplan*
Provee una estimación de costos de I/O de información.
 - *Statistics io*
Provee un costo de información actual de i/o
- Tiempo de ejecución transcurrida (*elapsed_time*)
- Número de filas de vueltas (*row_count*)

I/O Costs

- Basado en el número de acceso lógico y físico usado durante el procesamiento de preguntas.
- Puede ser respetado antes o durante la ejecución.

- La estimación de optimización del costo de I/O antes de la ejecución de preguntas.
- Determinar durante se declara el cursor excepto por la ejecución del cursos.
- Para ejecutar cursores, los costos son determinado durante *open cursor*.

Showplan

- *Showplan* muestra una línea adicional cuando los reguladores de recursos son autorizados.
- Igual para *statistics io*.

Elapsed Execution Time

- Aplicar a todos los comandos de SQL.
- Grupo de procedimientos almacenados o transacciones que no son reconocidos.
- Medida en segundos.

Number of Rows

- Líneas contadas referidas al número de líneas regresadas a los usuarios.
- Generación interna de líneas *subquery* no es incluida.
- El contador de línea aplica a el numero acumulativo de retorno de líneas crear un cursor para el tiempo este es abierto a el tiempo este es cerrado. El contador de líneas límite es recalculado cada vez que un cursor es abierto.

Valor de límites de recursos (*limit_value*)

- El salto superior de el *login* o aplicación puede ser alcanzado antes de imponer el limite del ASE.
- El valor debe ser de carácter tipo entero.
- Valores específicos:
 - Costos de I/O en una cantidad pasada.
 - Transcurrir el tiempo en segundos.
 - Contador de líneas en números de registros.
- Conseguir una idea de posible costos de i/o valores de limite, típica corrida de preguntas con el *set statistics io* o *set showplan* poner opciones en *on*.

Respetando los límites de recursos (*enforced*)

Los límites de recursos pueden ser *enforced* (1) previo a/o (2) durante la ejecución del *query*. En la tabla 4.9 se enlistan los valores validos por cada tipo de límite:

Enforced	Descripción	Tipo de limite
1	La acción es tomada cuando el costo estimado de ejecución excede el límite especificado.	<i>io_cost</i>
2	La acción es tomada cuando la fila actual cuenta, el tiempo pasado, o el costo de ejecución excedido el límite especificado.	<i>row_count</i> <i>elapsed_time</i> <i>io_cost</i>
3	La acción es tomada cuando tanto el costo estimado o el costo actual excede el limite especificado	<i>o_cost</i>
NULL	Si tu especificas un valor respetado de NULL, el ASE respeta el limite a todos los tiempos validos por el tipo de limite	

Tabla 4.9 Valores validos por cada tipo de límite

Acciones a tomar en violaciones (*action*) se observan en la tabla 4.10

- Especificar las acciones a tomar cuando el límite es excedido.
- El aplicar el código acción es validar para todos los tipos de limite:

Acción	Descripción
1	Seguir una advertencia
2	Abortar el grupo <i>query</i>
3	Abortar la transacción
4	Matar la sesión
Null	Si se especifica una acción de null, los reguladores de recursos abortan los <i>query batch</i> .

Tabla 4.10 Acciones a tomar en violaciones

Ámbito de los límites de recursos (*scope*)

Acumulativamente el uso de recursos puede ser calculado sobre agrupaciones de declaraciones SQL.

Query

- Solo T-SQL declara que el acceso a un objeto de servicio.
- Cada declaración en el grupo o transacción es evaluada individualmente.

Query Bacth

- Un grupo *query* consiste de uno o más declaraciones T-SQL.

Transacción

- Los reguladores de recursos no son reconocidos en el grupo de las transacciones.

Compatibility

- Especifica uno de los códigos a seguir apropiada a los tipos de límites estos se muestran en la tabla 4.11.

Ámbito	Descripción	Tipos de límites
1	<i>Query</i>	<i>io_cost</i> <i>row_count</i>
2	<i>Query batch</i> (una o más declaraciones SQL enviadas por el cliente o el servidor)	<i>elapsed_time</i>
4	<i>Transaction</i>	<i>elapsed_time</i>
6	<i>Query batch (2) + transaction (4)</i>	<i>elapsed_time</i>

Tabla 4.11 Códigos apropiados a los tipos de límites

Modificando los límites de recursos.

- Modificando los valores de los límites de recursos usando *sp_modify_resource_limit*.

sp_modify_resource_limit

- **Función**

Cambia un límite de recurso especificando un nuevo valor de límite, o la acción a tomar cuando el límite es excedido, o ambos.

- **Sintaxis**

```
sp_modify_resource_limit {name, appname } rangename , limittype , limitvalue ,  
enforced , action , scope
```

- **Parámetros**

name

Es la entrada al sistema del *Adaptive Server* a la cual el límite se aplica. Se debe especificar *name* o un *appname* o ambos. Modificar un límite este se aplica a todos los usuarios de una aplicación particular, especificar un *name* de *null*.

appname

Es la entrada al sistema del *Adaptive Server* a la cual el límite se aplica. Se debe especificar *name* o un *appname* o ambos. Modificar un límite esto se aplica a todos los usuarios de una aplicación particular, especificar un *appname* de *null*.

Si el límite gobierna una aplicación particular, especificar el nombre de aplicación que el programa de cliente pasa al *Adaptive Server* en el paquete de entrada al sistema.

rangename

Es la variedad de tiempo durante la cual el límite es hecho cumplir. No se puede modificar este valor, pero se debe especificar un valor no nulo para identificarse únicamente el límite de recurso.

limittype

Es el tipo de recurso al cual el límite se aplica. No se puede modificar este valor, pero se debe especificar un valor no nulo para identificarse únicamente el límite de recurso.

El valor debe ser uno de los que se muestran en la tabla 4.12.

Tipo de limite	Descripcion
row_count	Limita el número de filas que un query puede devolver.
elapsed_time	Limita el número de segundos en el tiempo wall-clock que un query batch o transacción puede correr.
io_cost	Limita el costo actual, o la estimación de costos del <i>optimizer</i> , para tratar un query

Tabla 4.12 Tipo de limittype para sp_modify_resource_limit

limit_value

Es la cantidad máxima del recurso de servidor que la entrada al sistema o la aplicación puede usar antes de que el Adaptive Server haga cumplir el límite. Este debe ser un número entero positivo menos que o igual a 231 o nulo para retener el valor existente.

En la tabla 4.13 se indica que valor especificar para cada tipo de límite:

Tipo de limite	Descripcion
row_count	El maximo número de filas que un query puede ser regresado.
elapsed_time	El maximo número de segundos en el tiempo wall-clock que un query batch o transacción puedan ser regresados
io_cost	Una medida de <i>unitless</i> se derivó de la fórmula de presupuesto de optimizar

Tabla 4.13 Tipo de limit_value para sp_modify_resource_limit

enforced

Determina si el límite es hecho cumplir antes de o durante una ejecución de un query. No se puede modificar este valor. Use *null* como un placeholder

action

Es la acción para tomar cuando el límite es excedido. Los códigos que se aplican a todos los tipos de límite se muestran en la tabla 4.14:

Codigo de accion	Descripcion
1	Cuestione una advertencia
2	Abortar un query batch
3	Abortar las transacciones
4	Matar la sección
null	Retiene el valor existente

Tabla 4.14 Tipo de acción para `sp_modify_resource_limit`

scope

Es el alcance del límite. No se puede modificar este valor. Se puede usar *null* como un dueño del lugar.

- **Uso**

- No se puede cambiar la entrada al sistema o la aplicación a la cual un límite se aplica o especifique una nueva variedad de tiempo, el tipo de límite, el tiempo de imposición, o el scope.
- La modificación de un límite de recurso causa los límites para cada sección para la entrada al sistema y/o aplicación para ser rebote a principios y después pregunte al query batch para aquella sesión.

- **Permisos**

Sólo un administrador de sistema puede ejecutar `sp_modify_resource_limit`.

- Solo *limit value and action to take* pueden ser modificados
 - o borrar y recrear el límite si otros parámetros requieren modificación.
- Los procedimientos almacenados requieren parámetros idénticos al procedimiento almacenado `sp_add_resource_limit`.
- Los procedimientos almacenados pueden no ser excluidos por cada una de las transacciones.
- Solo el administrador del sistema puede ejecutar los procedimientos almacenados.

El limite de recursos drop.

- El limite de recursos drop uno o mas procedimientos almacenado *sp_drop_resource_limit*

sp_drop_resource_limit

- **Función**

Quita uno o varios límites de recurso del Adaptive Server.

- **Sintaxis**

```
sp_drop_resource_limit { name, appname } [, rangename, limittype, enforced,  
action, scope]
```

- **Parámetros**

name

Es la entrada al sistema del *Adaptive Server* a la cual el límite se aplica. El limite de recurso drop se aplica a todos los usuarios de una aplicación particular, especifica el *appname* y un *name* de *NULL*.

appname

Es la entrada al sistema del *Adaptive Server* a la cual el límite se aplica. El limite de recurso drop que se aplican a todos los usuarios de una aplicación particular, especifican el *appname* y un *appname* de *NULL*.

El limite de recurso drop que se aplica a una aplicación particular, especifique el nombre de aplicación que el cliente el programa pasa al *Adaptive Server* en el paquete de entrada al sistema.

rangename

Es la variedad de tiempo durante la cual el límite es cumplido. Este debe ser una variedad de tiempo existente almacenada en la tabla de sistema *systimeranges* o *NULL* suprimir todo el recurso limitado para el especificado, *name*, *appname*, *limittype*, *action*, y *scope*, sin hacer caso de *rangename*.

limittype

- Es el tipo de recurso limitado. Este debe ser uno de lo siguiente:
- *row_count* – El limite de recurso drop sólo restringe el número de filas que puede devolver un query.
- *elapsed_time* – El limite de recurso drop solo restringe el número de segundos que un query batch o la transacción puedan correr.
- *io_cost* – El limite de recurso drop que restringe un query actual o estimada procesando el costo.
- NULL – El limite de recurso drop especifica *name*, *appname*, *rangename*, tiempo de imposición, *action*, y *scope*, sin hacer caso de *limittype*.

enforced

Determina si el límite es hecho cumplir antes de o durante la ejecución de un query.

En la tabla 4.15 se pone los valores válidos en una lista para cada tipo de límite.

Codigo de enforced	Descripción	Tipo de limite
1	Drop sólo limita para que la acción sea tomada cuando el costo estimado de la ejecución excede el límite especificado.	<i>io_cost</i>
2	Drop sólo limitan para que la acción sea tomada cuando la cuenta de fila actual, pasó el tiempo, o costado de la ejecución excede el límite especificado.	<i>row_count</i> <i>elapsed_time</i> <i>io_cost</i>
3	Drop sólo limitan para que la acción sea tomada cuando costo estimado (1) o el el costo actual (2) excede el límite especificado.	<i>io_cost</i>
NULL	Drop todo el recurso limitan con el especifico <i>name</i> , <i>appname</i> , <i>rangename</i> , <i>limittype</i> , y el <i>scope</i> , sin hacer caso de <i>action</i> es hecha cumplir.	

Tabla 4.15 Valores válidos en una lista para cada tipo de límite

action

Es la acción tomada cuando el límite es excedido. Este debe ser uno de los que se muestran en la tabla 4.16.

Código de acción	Descripción
1	Drop sólo limita aquella cuestión de advertencia.
2	Drop sólo limita que se aborte el grupo query.
3	Drop sólo limita que se aborte la transaccion.
4	Drop sólo limita que se mate la sección.
NULL	Drop todo el recurso limita con el específico <i>name</i> , <i>appname</i> , <i>rangename</i> , <i>limittype</i> , imposición de tiempo, y <i>scope</i> , sin hacer caso de <i>action</i> que ellos toman.

Tabla 4.16 Código de acción tomada cuando el límite es excedido

scope

Es el alcance del límite. Este debe ser uno de los que se muestran en la tabla 4.17.

Código de scope	Descripción
1	Drop sólo limita lo que se aplica a queries.
2	Drop sólo limita lo que se aplica a un grupo query.
4	Drop sólo limita lo que se aplica a transacciones.
6	Drop sólo limita lo que se aplica a query batch como a transacciones.
NULL	Drop todo el recurso limitan con el específico <i>name</i> , <i>appname</i> , <i>rangename</i> , <i>limittype</i> , imposición de tiempo, y <i>scope</i> , sin hacer caso de <i>scope</i> que ellos toman.

Tabla 4.17 Código de scope tomada cuando el límite es excedido

- **Uso**

- Usan el procedimiento de sistema `sp_help_resource_limit` para determinar que los límites de recurso se aplican a un usuario dado, aplicación, o tiempo del día.
- Cuando usted usa `sp_droplogin` para dejar caer una entrada al sistema de Adaptive Server, todos los límites de recurso asociados con aquella entrada al sistema también son borrados

- La eliminación de un límite de recurso causa los límites para cada sesión para aquella entrada al sistema y/o aplicación para ser rebote a principios del siguiente query batch para aquella sesión.

- **Permisos**

Sólo un administrador de sistema puede ejecutar *sp_drop_resource_limit*.

Creando un nombre de rango de tiempo

- Un límite de recurso activo es determinado un nombre de tiempo de rango.
- El tiempo de rango es creado usando el procedimiento almacenado: *sp_add_time_range*

sp_add_time_range

- **Función**

Añade una variedad de tiempo llamada a un Adaptive Server.

- **Sintaxis**

sp_add_time_range name, startday, endday, starttime, endtime

- **Parámetros**

name

Es el nombre de la variedad de tiempo. Los nombres de variedad de tiempo deben ser 30 caracteres o menos. El nombre no puede existir ya en la tabla del sistema *sysrangeranges* de la base de datos *master*.

startday

Es el día de la semana durante la cual la variedad de tiempo comienza. Este debe ser el nombre completo del día laborable para el idioma predefinido del servidor, como almacenado en la tabla de sistema de *syslanguages* de la base de datos *master*.

endday

Es el día de la semana durante la cual la variedad de tiempo se termina. debe ser el nombre completo del día laborable para el idioma predefinido del servidor, como almacenado en la tabla de sistema de *syslanguages* de la base de datos *master*. El *endday* puede caerse antes o más tarde en la semana que el *startday* o puede ser el mismo día como el *startday*.

starttime

Es el tiempo de día cuando la variedad de tiempo comienza. Especifique el *starttime* en términos de un reloj de 24 horas, con un valor entre "00:00" (medianoche) y "23:59" (a las 23h59). Use la forma siguiente: "HH:MM"

endtime

Es el tiempo de día cuando la variedad de tiempo se termina. Especifique el *endtime* en términos de un reloj de 24 horas, con un valor entre "00:00" (medianoche) y "23:59" (a las 23h59). Use la forma siguiente: "HH:MM"

Nota Para crear una variedad de tiempo que atraviesa el día entero, especificar a ambos un *starttime* y un *endtime* de "00:00". El *endtime* no debe ocurrir más tarde que el *starttime*, a menos que el *endtime* es "00:00".

- **Uso**

- *Adaptive Server* incluye un nombre de rango de tiempo, el "at all times" rango de tiempo. Esta vez el rango cubre todos los tiempos, a partir del primer día durante la última de la semana, de 00:00 por 23:59. No puede ser modificado o suprimido.
- *Adaptive Server* genera un número de ID único para cada nombre de rango de tiempo e inserta este dentro de la tabla de sistema *systimeranges*, El *Adaptive Server* convierte su *startday* y valores de *endday* en números enteros. Para servidores con idioma predefinido de *us_english*, la semana comienza en El lunes (día 1) y finales el domingo (día 7).
- Es posible crear una variedad de tiempo que se superpone con uno o varios variedades de tiempo.
- El rango de días son contiguos, entonces los días de la semana pueden abrigarse alrededor del final al principio de la semana. En otras palabras, el domingo y el lunes son días contiguos, como son el martes y el miércoles.
- Los rangos de tiempo están ligados a una sesión a principios de cada un query batch. Un cambio en el tiempo activo del servidor se extiende debido a un cambio en el tiempo actual no tiene ningún efecto en una sesión durante el procesamiento de un query batch. En otras palabras, si un límite de recurso restringe una pregunta el query batch durante una variedad de tiempo dada pero un query batch comienza antes de esto la variedad de tiempo se hace activa, el query batch que corre ya no es afectado por el límite de recurso.
- La adición, modificación, y eliminación de rangos de tiempo usando los procedimientos de sistema no afectan los rangos de tiempo activos para sesiones actualmente en progreso.

- Si un límite de recurso tiene una transacción como su alcance, y un cambio ocurre en los rangos de tiempo activos del servidor mientras una transacción corre, el rango de tiempo recién activado no afecta la transacción actualmente en progreso.
- Se cambia a un límite de recurso que tiene una transacción cuando su alcance no se afecte cualquier transacción actualmente en el progreso.

- **Permisos**

Sólo un administrador de sistema puede ejecutar *sp_add_time_range*.

Meta del funcionamiento

- El regulador de recursos causa menos de el 1% de impacto de funcionamiento en el ASE medida en transacciones por segundo.

Memoria

- El regulador de recursos requiere algunas estructuras de memoria interna adicionales.
- Los cambios después de “*allow resource limits*” permitido por *default* en el ASE:
 - Procedimiento cache decrece de 1028 a 1026 (KB)
 - Numero de alarmas incrementadas de 2 a 4 (KB)

Resumen

- El regulador de recursos proporcionan una manera para el administrador del sistema para fijar limites en los recursos del ASE.
- Los tipos de limites incluyen:
 - Costos I/O (estimado y actual)
 - El tiempo transcurrido
 - El numero de filas
- Las acciones de violación incluyen:
 - Advertencias
 - Abortar los query batch
 - Abortar las transacciones
 - Matar las sesiones de usuario

Los límites de recursos pueden ser activados a diferentes tiempos. Otra herramienta que el administrador podrá usar para la seguridad de su sistema son los comandos para los roles definidos a usuarios.

4.1.3 Roles y tareas asignadas

Roles Definidos a Usuario.

Propósito:

Llegar a familiares con la implementación y característica de los roles definidos a usuarios.

Objetivos:

- Discutir la necesidad de los roles de usuario y como ellos comparan los grupos.
- Explicar como implementar los roles de usuarios.

Objetivos Detallados:

Aprender la sintaxis de sql para lo siguiente:

- Creación de roles
- Otorgar permisos de roles
- Otorgar roles a logines, roles.
- Remover roles y permisos.

Discutir la jerarquía de roles y exclusividad recíproca.

Descubrir de las características.

- Control de acceso basado en roles (RBAC) acceso restringido a objetos en los roles que un login esta asignado.
- Los logines no tienen acceso directo a objetos:
 - Los permisos de acceso están asociados a los roles
 - Los logines están hechos para los miembros de los roles adecuados.
- ASE está destinado al control de acceso basado en role con roles de usuario definido.

Beneficios.

- Los roles pueden ser definidos para una aplicación específica.
- Puede definir los roles de una relación o exclusividad mutua de un esfuerzo de seguridad.
- Puede definir jerarquías de roles como un principios de creación y administración de los roles eficientemente.

Roles Definidos de Usuarios.

- Son implementaciones para cumplir con el estándar ANSI SQL3 para RBAC y proveer de las extensiones para utilidad de ensamblado.
 - Define la jerarquía de roles para la creación de roles.
 - Define una exclusividad mutua de los roles para impedir fraudes.
- Existen similitudes entre los nuevos usuarios de los roles definidos y la funcionalidad provista por los grupos ASE, pero las diferencia son críticas.

Grupos.

Los grupos son a nivel de base de datos, son mecanismos del ASE que provee acceso controlados.

- Define un conjunto de usuarios como grupo.
- Maneja privilegios y los revoca para el grupo completo de usuarios con un comando a través de la especificación del grupo.
- Adiciona nuevos grupos con *sp_addgroup*.
- Adiciona miembros de usuarios para un grupo.

sp_addgroup

- **Función**

Añade un grupo a una base de datos. Los grupos se utilizan como nombres colectivos para conceder y revocar privilegios.

- **Sintaxis**

`sp_addgroup grpname`

- **Parámetros**

grpname

Es el nombre del grupo. Los nombres de grupo deben cumplir con las reglas correspondientes a los identificadores.

- **Comentarios**

- **sp_addgroup** añade el nuevo grupo a la tabla *sysusers* de una base de datos. La ID de usuario de cada grupo (*uid*) es igual o mayor que 16384 (excepto "public", que siempre es 0).
- Un grupo y un usuario no pueden tener el mismo nombre.
- Una vez creado un grupo, pueden añadirse nuevos usuarios con **sp_adduser** . Para añadir un usuario existente a un grupo, use **sp_changegroup** .
- Todas las bases de datos se crean con un grupo llamado "public". Cada usuario se convierte automáticamente en miembro de "public". Cada usuario también puede ser miembro de un grupo adicional.

- **Mensajes**

- Ya existe un grupo con el nombre especificado.El nombre de grupo indicado se está usando como nombre de grupo. Elija otro nombre.
- Ya existe un usuario con el nombre de grupo especificado. El nombre de grupo indicado se está usando como nombre de usuario. Elija otro nombre.
- *grpname* no es un nombre válido. Los nombres de grupo deben cumplir con las reglas correspondientes a los identificadores.
- Se añadió un nuevo grupo. El grupo se añadió a la tabla *sysusers* de la base de datos actual.

- **Permisos**

Sólo el propietario de la base de datos o un administrador del sistema puede ejecutar **sp_addgroup** .

Limitaciones de grupo:

- Los grupos son internos para una base de datos.
 - No puedes crear un grupo que consista de dos usuarios que sean de dos bases de datos diferentes.
 - Puede solo asignar permisos a objetos en la base de datos donde fue creado

- Todos los usuarios son miembros del grupo público.
 - Público es un grupo en cada base de datos.
 - Para excluir uno o más usuarios del público, necesita editar la declaración explícitamente para revocar.
- Los usuarios puede ser un miembro del público y solo uno de otro grupo.

GRUPOS CONTRA ROLES:

Prioridad de Accesos

- Cuando los privilegios de acceso son determinados los permisos son checados en este orden
 - Permisos de usuarios
 - Permisos de rol
 - Permisos de grupos

Los privilegios de acceso de grupo son considerados solo si el usuario no tiene acceso directamente o a través de un rol.

Prioridad de Acceso.

- Grupos

Si una revocación directa es ejecutada contra el usuario pero el grupo al que el usuario esta asignado tiene el privilegio, el usuario no obtiene el privilegio.

- Roles

Si una revocación directa es ejecutada contra el usuario, pero cuando este rol esta encendido, uno de los roles poseídos por el usuario tienen el privilegio, el usuario obtiene el privilegio.

Implementación de roles definidos a usuarios.

Son diferentes de los grupos ya que un grupo es específico de una base de datos.

Son cuatro los pasos para la implementación de roles de usuarios.

1.- Un login con privilegios SSO debe crear el rol de usuario

2.- Después de que el rol es creado cualquier login con el permiso *with grant* puede otorgar privilegios de acceso a un rol usuario.

3.- El login con privilegios SSO puede otorgar a otros logines membresía en un rol de usuario.

4.- A los usuarios a los que fueron otorgados la membresía a un rol debe activar explícitamente su rol para activarlo y obtener los privilegios asociados.

SINTAXIS:

- **Para la creación de Roles**

```
create role role_name [ with passwd password ]
```

- **Para cambiar el password o para un conjunto de password:**

```
alter role role_name add password password
```

- **Para borrar un password**

```
alter role role_name add drop password
```

- **Otorgar Privilegios de Acceso Para Roles.**

Después de crearlo:

```
grant update on student_info to teacher_role
```

- Borrar un rol.

```
drop role role_name [with override]
```

- **Otorgando Roles para Logines o Roles**

SNTAXIS:

```
grant role role_granted [, role_granted,.....] to grantee [,grantee, ...]
```

- **Revovando Roles Para Logines o Roles**

SINTAXIS:

```
revoke role_revoked [,role_revoked,.....]  
from grantee [, grantee, ...]
```

grantee: puede ser el nombre de un login o un rol, estas declaraciones las requiere el rol SSO.

- **Habilitando Un Rol**

Al asignar un rol a un usuarios significa que es usuario tiene la capacidad para habilitar el rol.

SINTAXIS:

```
set role role_name [with passwd password] on
```

JERARQUIA DE ROL

- La jerarquía de role define roles que tienen atributos únicos y que pueden contener otros roles.
- Es una ruta natural de organizar roles para reflejar la autoridad y la responsabilidad.

Ciclos de Roles Otorgados.

- Circular los privilegios del rol no esta permitido.

Revocarlos En Una Jerarquía

- Al remover un rol de otro rol remueve el contenido de la relación.
- El contenido de la relación debe ser directa.

Mostrar Información Del Rol

- Usar el proceso del sistemas *sp_displayroles* para encontrar lo siguiente:
 - Que roles están otorgados para el siguiente.
 - Que roles están contenidos por otros roles.
 - Cuales son los roles ascendientes para un rol particular.
- Usar el proceso del sistemas *sp_activeroles* para encontrar lo siguiente:
 - Que roles están activos/habilitados por el login actualmente.
 - Que roles están contenidos por los roles o rol activo.

sp_displaylogin

- **Función**

Muestra información sobre una cuenta de login.

- **Sintaxis**

`sp_displaylogin [loginame]`

- **Parámetros**

loginame

Es la cuenta de login de usuario sobre la que desea obtener información, si no es la suya. Para obtener información sobre la cuenta de login de otro usuario, es necesario ser oficial de seguridad del sistema o administrador del sistema.

- **Comentarios**

sp_displaylogin muestra los roles configurados, de modo que se mostrará la información incluso en el caso de que se haya desactivado un rol con el comando **set** .

Al utilizarse **sp_displaylogin** para obtener información sobre la propia cuenta de usuario, no es necesario usar el parámetro *loginame* . **sp_displaylogin** mostrará la ID de usuario del servidor, el nombre de login, el nombre completo, los roles concedidos al usuario, la fecha del último cambio de contraseña, y si la cuenta está bloqueada.

El oficial de seguridad del sistema o el administrador del sistema puede usar el parámetro *loginame* para obtener acceso a la información relativa a cualquier cuenta.

- **Mensajes**

No existe ningún login con el nombre especificado. Se ha especificado un *loginame* incorrecto.

- **Permisos**

Cualquier usuario puede ejecutar **sp_displaylogin** para obtener información sobre su propia cuenta de login. Los oficiales de seguridad del sistema y los administradores del sistema pueden usar **sp_displaylogin** con el parámetro *loginame* para obtener información sobre las cuentas de login de otros usuarios.

sp_activeroles

- **Función**

Despliega todos los roles activos.

- **Sintaxis**

`sp_activeroles [expand_down]`

- **Parámetros**

`expand_down`

Muestra el árbol de jerarquía de todos los roles activos contenidos por sus roles.

- **Uso**

- El `sp_activeroles` muestra todos sus roles activos y todos los roles contenidos por esos roles.

- **Permisos**

Cualquier usuario puede ejecutar `sp_activeroles`.

sp_helprotect

- **Función**

Informa sobre permisos para objetos, usuarios o grupos de base de datos.

- **Sintaxis**

`sp_helprotect [name [, username [, "grant"]]]`

- **Parámetros**

name

Es el nombre de la tabla, vista o procedimiento almacenado, o el nombre de un usuario o grupo de la base de datos actual. Si no se proporciona ningún nombre, **sp_helprotect** informa de todos los permisos de la base de datos.

username

Es el nombre de un usuario de la base de datos actual.

grant

Muestra los privilegios concedidos a *name with grant option* .

- **A continuación se ilustra la aplicación de `sp_helprotect`**

```
1. grant      select      on      titles      to      judy
   grant      update      on      titles      to      judy
   revoke     update      on      titles(price)  from    judy
   grant      select      on      publishers   to      judy
   with grant option
```

Después de esta serie de instrucciones **grant** y **revoke** , la ejecución de **sp_helprotect titles** genera esta pantalla:

grantor	grantee	type	action	object	column	grantable
dbo	judy	Grant	Select	titles	All	FALSE
dbo	judy	Grant	Update	titles	advance	FALSE
dbo	judy	Grant	Update	titles	notes	FALSE
dbo	judy	Grant	Update	titles	pub_id	FALSE
dbo	judy	Grant	Update	titles	pubdate	FALSE
dbo	judy	Grant	Update	titles	title	FALSE
dbo	judy	Grant	Update	titles	title_id	FALSE
dbo	judy	Grant	Update	titles	total_sales	FALSE
dbo	judy	Grant	Update	titles	type	FALSE
dbo	judy	Grant	Select	publishers	all	TRUE

```
2. grant select, update on titles(price, advance)
   to mary
   with grant option
   sp_helprotect titles
```

Después de esta instrucción **grant** , **sp_helprotect** muestra lo siguiente:

grantor	grantee	type	action	object	column	grantable
dbo	mary	Grant	Select	titles	advance	TRUE
dbo	mary	Grant	Select	titles	price	TRUE
dbo	mary	Grant	Update	titles	advance	TRUE
dbo	mary	Grant	Update	titles	price	TRUE

```
3. sp_helprotect judy
```

Muestra todos los permisos que "judy" tiene en la base de datos.

- **Comentarios**

- **sp_helprotect** informa sobre los permisos de un objeto de base de datos. Si se suministra el parámetro *username* , sólo se informará sobre los permisos de dicho usuario para el objeto de base de datos. Si *name* no es un objeto, **sp_helprotect** comprueba si es un usuario o un grupo. Si es un objeto, **sp_helprotect** enumera los permisos del usuario o grupo.
- **sp_helprotect** busca objetos y usuarios únicamente en la base de datos actual.

- **Mensajes**

- El objeto debe estar en la base de datos actual. El nombre suministrado para el parámetro *name* incluía una referencia a una base de datos. El nombre debe ser local para la base de datos.
- No existe ningún usuario con el nombre especificado en la base de datos actual. El nombre suministrado para *username* no es un usuario o grupo de la base de datos actual.
- No existe dicho objeto o usuario en la base de datos. El nombre suministrado para el parámetro *name* no es un objeto, usuario o grupo de la base de datos actual.

- **Permisos**

Cualquier usuario puede ejecutar **sp_helprotect** .

4.1.4 Roles Especiales

El rol SA: Es para los administradores del sistema; manejo de almacenamiento de disco, pueden borrar y modificar, bloquear y desbloquear logines, puede otorgar el rol a revocar el rol SA, crear Bases de Datos de usuario así como otorgar propiedad sobre ella, otorgar ciertos permisos a los usuarios de SQL, puede cerrar SQL y sus procesos, monitorea la recuperación de Bases de Datos en el arranque del SQL server y utiliza algunas herramientas para el diagnóstico de problemas en el sistema.

El rol SSO: Oficiales de seguridad del sistema puede crear logines en el SQL server asignando password iniciales, puede cambiar los password, podrá asignar un intervalo de expiración de dichos password, crea, otorga y revoca roles de usuarios, otorga roles de usuario SSO y oper maneja el sistema de auditoría, bloquea y desbloquea logines.

El rol OPER: Son operadores del SQL server pueden respaldar y cargar todas las Bases de Datos y logs de transacción en el servidor y realizan dump db, dump transaction, load db, load transaction. Este rol no necesariamente necesita ser propietario de una Base de Datos para darle mantenimiento.

En la figura 4.5 se muestran los roles especiales:

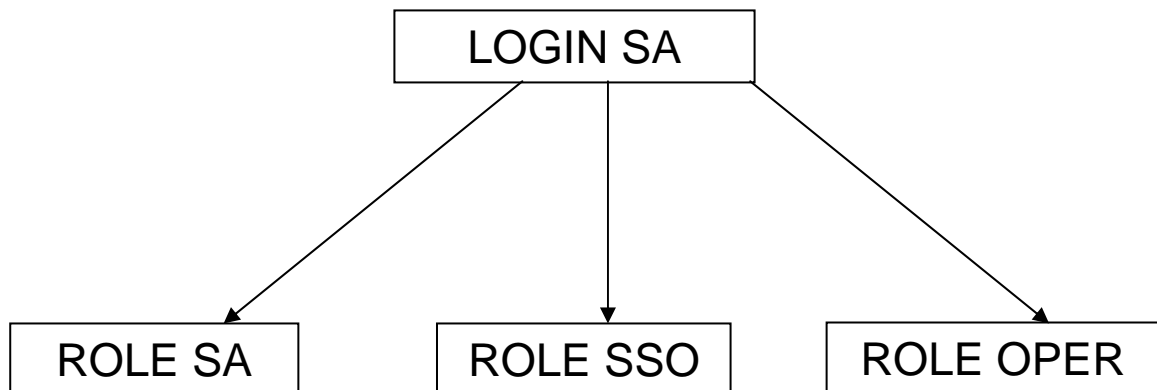


Figura 4.5 Roles especiales

4.2 Seguridad en Capa 2

4.2.1 Control de Acceso a la Base de Datos de Usuarios

sp_adduser

- **Función**

Añade un usuario nuevo a la base de datos actual.

- **Sintaxis**

`sp_adduser loginame [, name_in_db [, grpname]]`

- **Parámetros**

loginame

Es el nombre del usuario que aparece en *master.dbo.syslogins* .

name_in_db

Es el nuevo nombre del usuario en la base de datos actual.

grpname

Añade el usuario a un grupo existente en la base de datos.

- **Comentarios**

- El propietario de la base de datos ejecuta **sp_adduser** para añadir un nombre de usuario a la tabla *sysusers* de la base de datos actual, lo que permite al usuario tener acceso a la base de datos actual con su propio nombre.
- La especificación del parámetro *name_in_db* proporciona al nuevo usuario un nombre en la base de datos que es distinto de su nombre de login de SQL Server. La capacidad para asignar nombres de usuario distintos se proporciona para ofrecer mayor comodidad. No se trata de un alias proporcionado por **sp_addalias**, puesto que no está correlacionado con la identidad y los privilegios de otro usuario.
- Un usuario y un grupo no pueden tener el mismo nombre.
- Un usuario sólo puede pertenecer a un grupo además del grupo predeterminado, "public". Todos los usuarios son miembros del grupo predeterminado "public". Utilice **sp_changegroup** para cambiar el grupo de un usuario.
- Para tener acceso a una base de datos, un usuario tiene que aparecer en *sysusers* (con **sp_adduser**), debe estar correlacionado con otro usuario de *sysalternates* (con **sp_addalias**), o bien debe haber una entrada "guest" (invitado) en *sysusers*.

- **Mensajes**

- Ya existe un usuario con el mismo nombre en la base de datos. *name_in_db* ya es un usuario de la base de datos. Elija otro nombre.
- Todas las id de usuario fueron asignadas. La base de datos ha alcanzado el número máximo de IDs de usuario.
- '*name_in_db*' no es un nombre válido. El *name_in_db* especificado no sigue las reglas de los identificadores.
- Se añadió un nuevo usuario. El comando **sp_adduser** se ha ejecutado con éxito. El usuario ya es conocido en la base de datos actual.
- No existe ningún grupo con el nombre especificado. El nombre de grupo indicado no existe en esta base de datos. Excluya el parámetro *grpname* o cree el grupo con **sp_addgroup**.
- No existe ningún login con el nombre especificado. El *loginame* indicado es desconocido para SQL Server. Cada usuario debe tener un nombre de login a SQL Server antes de poder añadirse a una base de datos.
- El usuario ya tiene un login con un nombre diferente. El usuario con el *loginame* indicado se enumera en la tabla *sysusers* de la base de datos actual con un nombre distinto al indicado como parámetro *name_in_db*.
- El usuario ya tiene acceso de alias a la base de datos. La base de datos ya conoce el *loginame* mediante un alias. Para añadir al usuario, omita el alias con **sp_dropalias** y luego ejecute de nuevo **sp_adduser**.

- **Permisos**

Sólo el propietario de la base de datos o un administrador del sistema puede ejecutar **sp_adduser** .

sp_addalias

- **Función**

Permite que un usuario de SQL Server sea conocido en una base de datos como otro usuario.

- **Sintaxis**

`sp_addalias loginame , name_in_db`

- **Parámetros**

loginame

Es el nombre master.dbo.syslogins del usuario que desea una identidad alternativa en la base de datos actual.

name_in_db

Es el nombre de usuario de la base de datos que se utiliza como alias de loginame . El nombre debe existir en master.dbo.syslogins y en la tabla sysusers de la base de datos actual.

- **Comentarios**

- La ejecución de **sp_addalias** correlaciona a un usuario con otro en la base de datos actual. La correlación se muestra en *sysalternates* , donde las *suids* de los dos usuarios están conectadas.
- Un usuario sólo puede tener un alias de usuario de base de datos por vez.
- Se puede generar un informe sobre cualquier usuario correlacionado a un usuario específico con **sp_helpuser** , proporcionando el nombre del usuario especificado como argumento.
- Cuando un usuario intenta utilizar una base de datos, SQL Server comprueba *sysusers* para ver si el usuario está enumerado en la misma. Si no lo está, verifica *sysalternates* . Si la *suid* del usuario está en *sysalternates* , correlacionada a una *suid* de usuario de base de datos, el primer usuario será tratado como el segundo usuario mientras utilice la base de datos. Si el usuario con nombre en *loginame* está en la tabla *sysusers* de la base de datos, SQL Server no utilizará la identidad de alias del usuario, ya que verifica *sysusers* y encuentra *loginame* antes de verificar *sysalternates* , donde se enumera el alias.

- **Mensajes**

- Se añadió usuario de alias.El procedimiento se ha ejecutado satisfactoriamente. Ahora, *loginame* puede utilizar la base de datos actual. Mientras se haga así, el usuario será conocido como *name_in_db* .
- '*loginame*' ya es un usuario de la base de datos actual.Un usuario con un login en la base de datos actual no puede recibir un alias para otro login en esa base de datos.
- No existe ningún login con el nombre especificado. No existe ninguna entrada en *master.dbo.syslogins* para *loginame* . Todos los usuarios que utilizan SQL Server, independientemente de que tengan un alias o no, deben tener un login.
- No existe ningún usuario con el nombre especificado en la base de datos actual. Puesto que *name_in_db* no es un usuario de la base de datos, no es posible asignarle el alias *loginame* .
- El nombre de usuario especificado ya tiene alias. El *loginame* ya fue asignado como alias a un usuario de la base de datos actual. Cada *loginame* puede asignarse como alias a un sólo usuario de la base de datos al mismo tiempo. Para cambiar un alias, omite primero el alias actual usando **sp_dropalias** , y a continuación, añade el nuevo alias.

- **Permisos**

Sólo el propietario de la base de datos o un administrador del sistema puede ejecutar **sp_addalias**.

sp_changedbowner

- **Función**

Cambia el propietario de una base de datos. **No** cambie el propietario de la base de datos *sybssystemprocs* .

- **Sintaxis**

`sp_changedbowner loginame [, true]`

- **Parámetros**

loginame

Es el nombre de login del propietario nuevo de la base de datos actual. El propietario nuevo todavía no debe ser conocido ni como usuario ni como alias (es decir, el propietario nuevo no debe estar enumerado en *sysusers* o *sysalternates*).

La ejecución de **sp_changedbowner** con el parámetro único *loginame* cambia la propiedad de la base de datos para *loginame* y omite los alias de usuarios que podrían actuar como el "dbo" antiguo.

true

Transfiere alias y sus permisos al nuevo propietario de la base de datos. Los únicos valores aceptables son "true" y "TRUE".

- **Comentarios**

- Después de ejecutar **sp_changedbowner** , el propietario nuevo será conocido como propietario de la base de datos dentro de la misma.
- El propietario nuevo debe disponer ya de un nombre de login a SQL Server, pero **no** debe tener un nombre o alias de usuario de base de datos en la base de datos. Para asignar la propiedad de la base de datos a dicho usuario, omite la entrada del nombre o alias de usuario antes de ejecutar **sp_changedbowner**.
- Para conceder permisos al propietario nuevo, un administrador del sistema debe concederlos al propietario de la base de datos, puesto que el usuario ya no es conocido dentro de la base de datos bajo ningún otro nombre.

- **Mensajes**

- No se puede cambiar el propietario de la base de datos master. Nadie puede cambiar el propietario de la base de datos *master* .
- Se cambió el propietario de la base de datos. El comando **sp_changedbowner** se ha ejecutado satisfactoriamente y se ha cambiado el propietario de la base de datos.
- Sólo el administrador del sistema (SA) o el propietario de la base de datos (dbo) pueden cambiar el propietario de la base de datos. Es preciso ser un administrador del sistema o el propietario de la base de datos para ejecutar **sp_changedbowner** .
- Los alias dependientes se correlacionaron al nuevo dbo. Se ha definido el parámetro opcional "true". Los alias y sus permisos se han transferido al "dbo" nuevo.
- Se eliminaron los alias dependientes. No se ha definido el parámetro opcional "true". Se han omitido los alias y sus permisos.
- No existe ningún login con el nombre especificado. El nuevo propietario de la base de datos propuesto debe tener un login a SQL Server.
- El nuevo propietario db propuesto ya es un usuario de la base de datos. El *loginname* especificado ya es un usuario en la base de datos actual. Para convertir el usuario en el propietario de la base de datos, omite la entrada de usuario de la tabla *sysusers* de la base de datos actual.
- El nuevo propietario db ya tiene un alias en la base de datos. El *loginname* especificado ya tiene un alias en la base de datos actual. Para convertir el usuario en el propietario de la base de datos, omite la entrada de alias de usuario de la tabla *sysalternates* de la base de datos actual.

- **Permisos**

Sólo un administrador del sistema puede ejecutar **sp_changedbowner** .

4.2.2 Grupos de acceso

Los usuarios que se tienen y acceden a la base de datos Base_de_Datos_1 son los siguientes, los cuales están organizados por grupos como se muestra en la tabla 4.18 .

- ALUMNOS
- ACADEMICOS, PROFESORES
- ADMINISTRATIVOS
- JEFES DE DIVISIÓN
- JEFES DE DEPARTAMENTO
- DIRECTIVOS (DIRECTOR Y SECRETARIOS)
- USUARIOS DE ADMINISTRACION ESCOLAR

	NOMBRE DEL GRUPO	USUARIOS
1	GRUPO_1	DIV0
2	GRUPO_2	Vacia
4	GRUPO_3	karinags, marthaggi
5	GRUPO_4	carovgc, cesariob, cynthianp, gerardoaa, juancm, marirusa, marthafh, paolaet, soptec
6	GRUPO_5	Caalfi
7	GRUPO_6	COPADI, _INFOR, DAE, DCB, DCB0, DCB1, DCB2, DCB3, DCB4, DCSH, DCSH1, DCSH2, DCSH3, DCSH4, DCT, DCT1, DCT2, DCT3, DCT4, DCTG, DCTG1, DCTG3, DCTG4, DIE, DIE1, DIE2, DIE3, DIE31, DIE32, DIE33, DIE331, DIE332, DIE333, DIE334, DIE335, DIE39, DIE4, DIEJD, DIMEI, DIMEI1, DIMEI3, DIMEI4, GFB, UACT, miguelfb, mirna, sriagra, gferrando
8	GRUPO_7	DAE_SS, DAE1, emiliadae
9	GRUPO_8	dcbfcr, dcbiva, dcbjcr, dcbllhm, dcbpmm, marcosth
10	GRUPO_9	admin., SERVES1, SERVES2, SERVES3, SERVES4, SERVESC
11	GRUPO_10	Marcote
12	GRUPO_11	Patricia
13	GRUPO_12	Vacia

	NOMBRE DEL GRUPO	USUARIOS
14	GRUPO_13	Vacia
15	GRUPO_14	Usuario
16	GRUPO_15	encuestas
17	GRUPO_16	CDOCENCIA, D2_2001, D3_2001, D4_2001, D4_2002, D5_2001, D6_2001, D7_2001, ebarra, galarcon, jjamtz
18	GRUPO_17	Vacia
19	GRUPO_18	Usuario2, usuario3
20	GRUPO_19	Vacia
	GRUPO_20	usuario00, usuario10, usuario12, usuario13, usuario14, usuario15, usuario16, usuario17, usuario18, usuario9,
21	GRUPO_21	usuario4, usuario5, usuario6, usuario7, usuario8, usuario11, usuario19, usuario20
22	GRUPO_22	Vacia
23	GRUPO_23	Vacia
24	GRUPO_24	Vacia
25	GRUPO_25	Vacia
26	GRUPO_26	Usuario1
27	GRUPO_27	Vacia
28	GRUPO_28	Usuario21

Tabla 4.18 Grupos de trabajo que contiene la base de datos

4.3 Seguridad en Capa 3

4.3.1 Permisos Sobre Objetos

INTEGRIDAD DE LA INFORMACIÓN

Existen dos tipos de mecanismos de seguridad que bien podemos mencionar, ya que es de suma importancia para otorgar permisos sobre objetos, los cuales son:

- Discrecional, se usa para otorgar privilegios a los usuarios.
- Obligatorios, sirve para imponer seguridad de múltiple niveles clasificando los datos y los usuarios en varias clases de seguridad e implementando una política de seguridad.

Así entre las obligaciones del DBA son otorgar dichos privilegios a los usuarios y clasificar los usuarios y los datos de acuerdo con la política de la USECAD. Tomando en cuenta lo siguiente:

- a) Creación de Cuentas.
- b) Concesión de privilegios.
- c) Revocación de Privilegios.
- d) Asignación de niveles de seguridad.

La capa 3 es el acceso a los objetos, para lo cual existe el siguiente comando que es uno de los de mayor utilidad para los DBA`s.

Para garantizar la confidencialidad puede hacerse la siguiente clasificación:

- 1) **Autorización explícita.** Usada en los sistemas tradicionales. Almacena que sujeto puede acceder a ciertos objetos con determinados privilegios para lo que suelen utilizarse una matriz de control de acceso.
- 2) **Autorización implícita.** Esta autorización definida sobre un objeto puede deducirse a partir de otras.

El tipo de autorización que sea utilizada dependerá entre otras cosas de:

- La política de control elegida, pudiendo el SGBD operar con un sistema abierto (en el que un usuario puede acceder a todos los objetos excepto aquellos que se prohíbe explícitamente) o como sistema cerrado (el usuarios accede solo a aquellos objetos para los que tiene autorización previa.
- El modelo de datos, ya que usar autorización explícita en los SGBD consume mucho espacio de almacenamiento debido a la existencia de un gran número de de elementos a controlar (clases, subclases, servicios, objetos complejos, etc...)

El comando grant sirve para otorgar permisos a ciertos usuarios y/o puede otorgar permisos sobre ciertos objetos.

Sintaxis:

```
grant { all [privileges] | [permission_list] } on { table_name [(column_list)] | view_name [(column_list)] | store_procedure_name } to { public | name_list | role_name } [with grant option]
```

A continuación se muestra en las tablas 4.19 y 4.20 la información recabada de la base de datos Base_de_Datos_1.

GRUPOS	Objetos	Acción
GRUPO_3	TABLA_0	+S
GRUPO_4	TABLA_2	+S
	TABLA_12	+S
	TABLA_15	+S
GRUPO_5	TABLA_2	+R+S
	TABLA_3	+R+S
	TABLA_8	+R+S
	TABLA_12	+R+S
	TABLA_14	+R+S
	TABLA_15	+R+S
GRUPO_8	TABLA_2	+R+S
GRUPO_9	TABLA_8	+D+I+R+S+U
	TABLA_12	+D+I+S+U
	TABLA_15	+D+I+R+S+U
	TABLA_3	+R+S
	TABLA_2	+S
GRUPO_10	TABLA_16	+I+S
	TABLA_2	+R+S
	TABLA_8	+R+S
	TABLA_12	+R+S
	TABLA_14	+R+S
	TABLA_15	+R+S
	TABLA_1	+S
	TABLA_4	+S
	TABLA_5	+S
	TABLA_7	+S
	TABLA_10	+S
	TABLA_11	+S
	TABLA_13	+S
	TABLA_17	+S+U
	TABLA_18	+S
	TABLA_19	+S
GRUPO_17	TABLA_2	+S

Tabla 4.19 Permisos sobre objetos

	Sa	sso	OPER	dbo
Usuario_1	X	X	X	Master
Usuario_2	X	X	X	Base_de_Datos_1
Usuario_3	X	X	X	Base_de_Datos_1

Tabla 4.20 Permisos de Roles

El administrador debe de ser conciente y tener cuidado ya que a pesar de revocar los privilegios a un usuario, este puede mantenerlos a través de otros usuarios (que le hayan concedido los mismos privilegios).

Otro mecanismo muy importante en la confidencialidad del SQL lo constituyen las vistas, que permiten ocultar información a los usuarios; y, así conceder privilegios solo sobre subconjuntos de las tablas.

4.4 Herramientas de Auditoria de la Base de Datos

AUDITORIA EN INFORMATICA

Es la revisión evaluación de los controles, sistemas, procedimientos de informática, de los equipos de computo, su utilización, eficiencia y seguridad, de la organización que participan en el procesamiento de la información, a fin de que por medio del señalamiento de cursos alternativos se logre una utilización mas eficiente y segura de la información que servira para una adecuada toma de decisiones. Se deberán evaluar los sistemas de información en general desde sus entradas, procesamientos, controles, archivos, seguridad y obtención de información.

Su campo de acción será:

- La evaluación administrativa del departamento de procesos electrónicos.
- La evaluación de los sistemas y procedimientos, y de la eficiencia que se tiene en el uso de la información.
- La evaluación del proceso de datos y de los equipos de cómputo.
- La seguridad.

PAQUETES DE AUDITORIA

Definición:

Consiste en un programa o una serie de programas de computación, desarrollados para llevar a cabo ciertas funciones de procesamiento electrónico con finalidades de auditorias. Dichas funciones incluyen, normalmente, la lectura de la información contenida en medios magnéticos, selección de datos, realización de cálculos, e impresión de listados de acuerdo con las especificaciones del auditor.

sp_audit

- **Función**

Permite que un oficial de seguridad del sistema configure las opciones a auditar.

- **Syntax**

sp_audit *option*, *login_name*, *object_name* [,*setting*]

- **Parámetros**

option

Es el nombre del conjunto de opciones a revisar. En la tabla 4.21 enumera las opciones válidas a revisar.

Opción	Descripción
Adhoc	Permite a usuarios utilizar sp_addauditrecord para agregar sus propios registros user-defined por el usuario de la auditoría al registro de la auditoría
All	Auditar todas acciones realizadas por un usuario particular o por usuarios con un role particular. Se puede sólo utiliza esta opción para especificar los roles del sistema. <hr/> Nota: Auditar todas acciones no afectan si los usuarios pueden agregar el anuncio éste audita los registros. <hr/>
Alter	Audita la ejecución de la tabla alter o los comandos de alter database.
Bcp	Audita la ejecución del bcp.
bind	Audita la ejecución de sp_bindefault, de sp_bindmsg, y de los procedimientos de sistema sp_bindrule.
Cmdtext	Audita todas las acciones de un usuario particular.
Create	Audita la creación de objetos de base de datos.
dbaccess	Audita el acceso a la base de datos actual de otra base de datos.
dbcc	Audita la ejecución de cualquier comando de dbcc.
delete	Audita la supresión de filas de una tabla o la vista.
disk	Audita la ejecución de disk init, disk refit, disk reinit, disk mirror, disk unmirror, y de disk remirror.
drop	Audita el borrado de los objetos de base de datos.

Opción	Descripción
dump	Audita la ejecución de dump de la base de datos o la transacción de dump.
Errors	Audita los errores, si fatal o no.
exec_procedure	Audita la ejecución de un procedimiento almacenado.
exec_trigger	Audita la ejecución de un trigger.
func_dbaccess	Audita el acceso a una base de datos vía una función de Transact -SQL.
func_obj_access	Audita el acceso a un objeto de la base de datos vía una función de Transact -SQL.
grant	Audita la ejecución de el grant
Insert	Audita la insercion de filas en una tabla o vista.
load	Audita la ejecución de la carga de la base de datos o la carga de una transacción.
Login	Audita todas las tentativas de login en el Adaptive Server
logout	Audita todas tentativas de logout en el Adaptive Server.
reference	Auditora las referencias entre tablas.
revoke	Audita la ejecución del revoke.
rpc	Audita la ejecución de llamadas remotas de procedimientos.
security	Audita los eventos security-relevant siguientes: <ul style="list-style-type: none"> • Activando o desactivando un rol • Publicando cualquiera de los comandos siguientes: addcert, connect, dropcert, kill ,online database, set proxy, set, session authorization, sp_configure • Utilizando cualquiera de las funciones siguientes: valid_user proc_role (from within a system procedure) •Regenerando las contraseñas de SSO
select	Audita la ejecución del select..
setuser	Audita la ejecución del setuser.
table_access	Audita el acceso a cualquier tabla por un usuario específico.
truncate	Audita la ejecución de truncate table
unbind	Audita la ejecución de sp_unbindrule, sp_unbindmsg, and sp_unbinddefault.
update	Audita las actualizaciones de las filas en vista o tabla.
view_access	Audita el acceso a cualquier vista por un usuario específico.

Tabla 4.21: Revisión de opciones

login_name

Es el parámetro que permite que se especifique todo, un rol de sistema, o el nombre de una específico login a ser auditado. Sin embargo, los roles de sistema pueden sólo ser especificados si se utiliza la opción all. No se puede auditar las opciones individuales para un rol de sistema.

object_name

Es el nombre del objeto a ser auditado. Los valores válidos, dependiendo del valor que se especifico para la *option*, son:

- El nombre objetivo, inclusive el nombre de dueño si usted no posee el objeto.
- All para todos los objetos
- Tabla default, vista default, procedimiento default, o trigger. Auditar el acceso a cualquier nueva tabla, vista procedimiento o trigger. Tabla default y vista default son valores validos por *object_name* cuando se especifica delete, insert, select, o update para la opcion *parameter*. Procedimiento default es valido cuando Se especifica la opcion *exec_procedure*. Trigger default es valido cuando se especifica la opcion *exec_trigger*.

setting

Es el nivel de auditoría. Si no se especifica un valor para *setting*, el Adaptive Server muestra la actual situación para la opción de auditar.

Los valores válidos para el parámetro *setting* son descritos en la tabla 4.22.

Valor <i>setting</i>	Descripción
on	Activa la auditoria para la opción especificada. El Adaptive Server genera los registros de la auditoria para los eventos controlados por esta opción.
off	Desactiva la auditoria para la opción especificada
pass	Activa la auditoria para los eventos de pass en los controles de permisos.
fail	Desactiva la auditoría para los eventos que fallan en los permisos de controles

Tabla 4.22 Valores válidos para el parámetro *setting*

Si se especifica el paso para una opción y después falla la especificación para la misma opción, o viceversa, el resultado es equivalente a especificar on. El Adaptive Server genera los registros de la auditoría a pesar de si los eventos pasan o fallan en los permisos de controles. Los settings de on o off se aplica a todas las opciones que auditan. Los settings de paso y falla aplican a todas las opciones excepto errores y adhoc. Para estas opciones es off.

- **Uso**

- El sp_audit determina lo que será auditado cuando la auditoría es permitida.
- Ninguna auditoría actual ocurre hasta que se use sp_configure poner el parámetro a auditar en on. Entonces, todas las opciones de auditoría que han sido configuradas con sp_audit entran en vigor.
- Si no se es el dueño del objeto especificado, califique el valor del parámetro de object_name con el nombre del dueño, en el siguiente formato:

"ownername.objname"

- No se puede activar la auditoría default para las opciones siguientes en la base de datos de tempdb:
 - delete
 - insert
 - select
 - update
 - exec_procedure
 - exec_trigger
 - Tabla 4.23 listas los parámetros de configuración para la auditoría.

Parametros de configuracion	Efectos
auditing	Permite o incapacita la revisión para el servidor.
audit_queue_size	Establece el tamaño del buffer de auditoría.
current_audit_table	Define la tabla actual de auditoría. El Adaptive Server escribe todos los registros de auditoría a aquella tabla.
suspend_auditing_when_full	Controla el comportamiento del proceso de auditoría cuando un dispositivo de auditoría se llena

Tabla 4.23 Parámetros de configuración para la revisión de control

auditing, current_audit_table, y suspend_auditing_when_full los parámetros de configuración son dinámicos y entran en vigor inmediatamente. Como audit_queue_size afecta la asignación de memoria, el parámetro es estático y no entra en vigor hasta que el Adaptive Server sea reiniciado.

- Permisos

Sólo un encargado de seguridad de Sistema puede ejecutar sp_audit.

sysauditoptions

Base de datos de sybsecurity

La descripción sysauditoptions contiene una fila para cada opción de auditoría server-wide e indica el setting actual para esta opción. Otros tipos de opciones de auditoría y los settings son almacenados en otras tablas. Por ejemplo, la opción database-specific los settings son almacenados en sysdatabases, y la opción object-specific los setting son almacenados en sysobjects. El valor por default para cada opción es 0, o "off". a sysauditoptions pueden tenerle acceso sólo por los Guardias de Seguridad de Sistema. En la tabla 4.24 se muestran las columnas para sysauditoptions.

Columnas	Las columnas para sysauditoptions son:	
Nombre	Tipo de Dato	Descripción
num	smallint	Número de la opción server-wide.
val	smallint	Valor actual; uno de lo siguiente: 0 = off 1 = pass 2 = fail 3 = on
minval	smallint	Minimo valor válido para esta opción.
maxval	smallint	Máximo valor válido para esta opción.
name.	varchar(30)	Nombre de opción
sval	varchar(30)	Equivalente string del valor actual: por ejemplo, "on", "off", "nonfatal".
comment	varchar(255)	Descripción de opción

En la tabla 4.24 Muestra las columnas para sysauditoptions

sysaudits_01 – sysaudits_08

Base de datos de sybsecurity

Descripción

Estas tablas de sistema contienen el registro de auditoría. Sólo una tabla a la vez es activa.

La tabla activa es determinada por el valor de la tabla de auditoría actual el parámetro de configuración. Una instalación puede tener hasta ocho tablas de auditoría. Por el ejemplo, si su instalación tiene tres tablas de auditoría, las tablas son llamadas sysaudits_01, sysaudits_02, y sysaudits_03. Una tabla de auditoría contiene una fila para cada registro de auditoría. En la tabla 4.25 se muestran las columnas para sysaudits_01 – sysaudits_08:

Columnas:	Las columnas para sysaudits_01 – sysaudits_08 son:	
Nombre	Tipo de Datos	Descripción
event	smallint	Tipo de evento auditado.
eventmod	smallint	Información adicional sobre el evento. Los valores posibles son: 0 = ningún modificador para este evento 1 = el evento pasó la comprobación de permiso 2 = el evento falló la comprobación de permiso
spid	smallint	Server proces ID del proceso que hizo que el registro de auditoría fuera escrito.
eventtime	datetime	Fecha y tiempo del evento auditado.
sequence	smallint	Número de secuencia del registro dentro de un solo evento; algunos eventos requieren más de un registro de auditoría.
suid	smallint	Login ID del usuario que realizó el evento auditado.
dbid	int null	Database ID en el cual el evento auditado ocurrió o object/storedprocedure/trigger reside, según el tipo de evento.
objid.	int null	ID del acceso al objeto o stored procedure/trigger.

Nombre	Tipo de Datos	Descripción
xactid	binary(6) null	ID de la transacción que contiene el evento auditado. Para una transacción multi-database, este es la ID de la transacción de donde la transacción se originó.
loginname	varchar(30) null	Nombre del login correspondiente al suid.
dbname	varchar(30) null	Nombre de base de datos correspondiente al dbid.
objname	varchar(30) null	Nombre del objeto correspondiente al objid.
objowner	varchar(30) null	Nombre del dueño de objid.
extrainfo	varchar(255) null	Información adicional sobre el evento auditado. Este campo contiene una secuencia de artículos separados por puntos y coma.

En la tabla 4.25 se muestran las columnas para sysaudits_01 – sysaudits_08:

La columna extrainfo contiene una secuencia de campos separados por puntos y coma como se muestra en la tabla 4.26.

Artículo	Contenido
Roles	Pone los roles en una lista que son activos. Los roles son separados por espacios.
Subcommand	El nombre de la opción de orden o suborden que fue usada para el evento. Por ejemplo, para el comando alter table, las opciones añaden la columna o la coacción de drop podría ser usado. Subórdenes múltiples o las opciones son separados por comas. El nombre de la opción de orden o suborden que fue usada para el evento. Subórdenes múltiples o las opciones son separados por comas.
Previous value	El valor antes de la actualización si el valor causara la actualización de un valor.
Current value	El nuevo valor si el evento causara la actualización de un valor.
Other information	La información relevante de seguridad adicional que es registrada para el evento.
Proxy information	El nombre de entrada al sistema original, si el evento ocurriera mientras un set proxy era en efecto.
Principal information	El nombre principal del mecanismo de seguridad subyacente, si la entrada al sistema del usuario es secures default login, y el usuario registrado en el Adaptive Server vía la unificado login. El valor de este campo es NULL, si la entrada secures default login no esta siendo usada.

Tabla 4.26 Artículos en la columna extrainfo

Un ejemplo de una columna extrainfo para el evento security-relevant de cambio un parámetro de configuración de revisión podría ser:

```
sso_role;suspend auditing when full;1;0;;;;
```

Esta columna extrainfo indica que un Oficial de Seguridad de Sistema se cambió el parámetro de configuración suspende la revisión cuando llena de 1 (suspenda todos los procesos que implican un evento de revisión) a 0 (truncan la siguiente tabla de auditoría y haga la tabla de auditoría actual). Las otras columnas en el registro de auditoría dan otra información pertinente. Por ejemplo, el registro contiene al usuario de servidor id (suid) y el nombre de entrada al sistema (loginname).

Los valores de columna de evento que pertenecen a cada evento de auditoría se muestran en una lista en el anexo 2.

CAPÍTULO V

RESULTADOS Y CONCLUSIONES

5. RESULTADOS Y CONCLUSIONES

5.1 Resultados

Post-Frap

En la tabla 5.1 se muestra la hoja de referencias cruzadas:

Número de Control	Control	Número de Riesgo	Riesgo	Tipo De Prioridad
11	Políticas	3	Falta de Políticas	B
18 y 21	Mantenimiento	4	Fallas en la Base de Datos	A
14	Respaldos	6	Mala Configuración de Back-ups	B
11, 13 y 24	Políticas, Auditoría/Mantenimiento y Propiedad	8	Mala Distribución de Privilegios en el Servidor de Base de Datos	A
24 y 16	Propiedad y Control de Acceso	10	Falta de Limitación de Recursos en el Servidor de Base de Datos	B

Tabla 5.1 Hoja de referencias cruzadas.

En la tabla 5.1 se observan los tipos de controles e identificación de los riesgos que pueden impactar al sistema de información; a continuación se dará una breve explicación:

- a) La falta de políticas es un riesgo de prioridad Media (B), y el control sugerido es el número 11, descritos en la Tabla 3.2 del capítulo 3; evidentemente es la creación de políticas, ver apéndice A.
- b) Fallas en la base de datos y Falta de espacio en almacenamiento son riesgos de prioridad Alta (A) y Media (B) respectivamente, para el cual sugerimos los controles número 18 y 21 descritos en la Tabla 3.2 del capítulo 3, además ver el apartado 4.4 Herramientas de Auditoría de la Base de Datos.

- c) La Mala configuración de Back-ups es un riesgo prioridad Medio (B) y el control es el número 14, descritos en la Tabla 3.2 del capítulo 3, además ver apartado 4.4 Herramientas de Auditoria de la Base de Datos, ya que no solo es necesario encender la auditoria sino también configurarla adecuadamente.

- d) Mala Distribución de Privilegios en el Servidor de Base de Datos es un riesgo de prioridad Alta (A) y el tipo de controles sugeridos 11, 13 y 24 respectivamente. Ver apéndice A, además ver apartado 4.4 Herramientas de Auditoria de la Base de Datos y apartado 4.3.4 Roles y tareas asignadas.

- e) Falta de Limitación de Recursos en el Servidor de Base de Datos es un riesgo de prioridad Media (B) y los controles sugeridos corresponden a los números 24 y 16, descritos en la Tabla 3.2 del capítulo 3, además ver apartado 4.3.3 Limitación de recursos de la base de datos y apartado 4.3.4 Roles y tareas asignadas.

5.2 Conclusiones

El objetivo de la tesis se cumplió, ya que gracias a la metodología FRAP con la cual se desarrollo el análisis de riesgos se encontraron los puntos más vulnerables del sistemas de información, puesto que son amenazas para el sistema de información y gracias a la implantación de los controles y una serie de sugerencias se podrán mitigar los riesgos para que el impacto al negocio sea lo mas bajo posible y en caso de sufrir algún ataque la institución este prevenida con un plan de recuperación que garantice la integridad, confidencialidad y la autenticidad de la información.

Para lograr al 100 % el aprovechamiento de nuestro análisis es necesario que los responsables administrativos así como el administrador de la base de datos (DBA), estén de acuerdo en llevar a cabo y apoyar este proyecto por el bienestar de la propia institución; haciendo cumplir cada una de las sugerencias que se presentan en este documentó.

Sabemos y estamos concientes de que algunas de las propuestas no se podrán llevar a cabo por cuestiones de política de la misma institución, pero sin embargo se deberán adaptar a las necesidades de la misma con la intención de no dañar al sistema y mantenerlo seguro.

Ya hemos mencionado que toda persona involucrada principalmente usuarios del sistema de información deberán ser controlados y supervisados, por lo tanto una de las herramientas básicas y más importantes para mantener un buena administración y el control del sistema es la auditoría, bien administrada; esta herramienta propuesta para este sistema de información ayudara a que el administrador mantenga controladas, administradas y supervisadas las actividades de los usuarios.

El administrador podrá auditar las entradas, salidas y los errores que le sean de mayor interés u otras actividades, en este documento se le presentan las opciones para llevar a cabo dicha auditoria. Así a través de esta herramienta podrá garantizar los servicios de integridad, autenticidad y confidencialidad. Puesto que a través de diversos análisis que se han llevado a cabo en otras instituciones esta comprobado que el 80 % de las amenazas son causadas por los mismos empleados, por lo tanto esta herramienta puede dar grandes ventajas y beneficios para una óptima administración del sistema.

Cabe resaltar que los riesgos y las vulnerabilidades no se pueden eliminar, sin embargo si se pueden controlar y mitigar; esto no implica que un análisis de riesgo no caduque, puesto que la revisión constante es importante ya que sabemos que la tecnología avanza día con día y trae consigo grandes beneficios, pero sabemos que a su vez trae también nuevos ataques y amenazas que tendrán que analizarse, por lo tanto se convierte en un ciclo de seguridad.

ANEXO 1

LISTA DE PARÁMETROS DE CONFIGURACIÓN

Anexo 1

Lista de parámetros de configuración

- En los siguientes párrafos se describen brevemente los parámetros de configuración.
- o **additional network memory** asigna memoria adicional a clientes que solicitan tamaños de paquetes mayores que el tamaño de paquete predeterminado para el servidor.
- o **allow nested triggers** determina si los disparadores pueden llamar a otros disparadores (es decir, "anidarse") o no. El valor predeterminado es 1 (las modificaciones de datos realizadas por disparadores pueden activar otros disparadores).
- o **address lock spinlock ratio** especifica el número de filas de la tabla de desmenuzamiento de bloqueos de direcciones protegidas por un bloqueo de giro (filas por bloqueo de giro).
- o **allow remote access** determina si los usuarios de servidores remotos pueden tener acceso a este SQL Server. El valor predeterminado es 1, para que SQL Server pueda comunicarse Backup Server.
- o **allow sql server async i/o** es un conmutador que habilita SQL Server para ejecutarse con una E/S de disco asíncrona .
- o **allow updates to system tables** permite la actualización directa de las tablas del sistema. El valor predeterminado es 0 (desactivado).
- o **audit queue size** determina el número de registros de auditoría que la cola de auditoría puede contener. El valor predeterminado es 100.
- o **configuration file** especifica la ubicación del archivo de configuración que desea utilizar.
- o **cpu accounting flush interval** especifica el número de impulsos del reloj que deberán acumularse antes de añadir datos de uso de la cpu a *syslogins* para su uso en las estadísticas de contabilidad de cargos.
- o **cpu grace time** especifica la cantidad máxima de tiempo (en milisegundos) que un proceso de usuario puede ejecutarse sin saturar la CPU antes de que SQL Server la infecte.
- o **deadlock checking period** especifica la cantidad mínima de tiempo (en milisegundos) que un proceso debe esperar por un bloqueo antes de que SQL Server inicie una verificación de bloqueo insoluble.
- o **deadlock retries** especifica el número de veces que una transacción intentará repetidamente adquirir un bloqueo después de convertirse en la víctima de un bloqueo insoluble.
- o **default character set id** es el número del juego de caracteres predeterminado utilizado por el servidor.
- o **default database size** define el número predeterminado de megabytes asignado a cada base de datos de usuario nueva. El valor de ejecución predeterminado es 2 (megabytes).

- **default fill factor percent** determina el nivel de llenado que SQL Server aplica a cada página cuando está creando un índice en datos existentes (a menos que el usuario especifique algún otro valor en la instrucción **create index**). El valor de ejecución predeterminado es 0.
- **default language id** es el número del idioma que se utiliza para visualizar mensajes del sistema, a menos que un usuario haya elegido otro idioma de los que están disponibles en el servidor.
- **default network packet size** define el tamaño predeterminado de paquetes de red para todos los usuarios de SQL Server.
- **default sortorder id** es el número del criterio de ordenación que es el valor predeterminado actual de este SQL Server. **No cambie este parámetro ..**
- **disk i/o structures** especifica el número inicial de bloques de control de E/S de disco que SQL Server asigna durante el arranque.
- **engine adjust interval no se utiliza actualmente.**
- **event buffers per engine** especifica el número de eventos por máquina SQL Server que puede controlarse simultáneamente. Los eventos se utilizan junto con Monitor Server y una herramienta cliente para observar el rendimiento de SQL Server.
- **executable code size** indica el tamaño del SQL Server ejecutable.
- **freelock transfer block size** especifica el número de bloqueos desplazados entre el caché de bloqueos libres de la máquina y la lista de bloqueos libres global.
- **housekeeper free write percent** determina el porcentaje máximo en que pueden aumentar las escrituras de base de datos como resultado de escrituras libres iniciadas por procesos de limpieza durante el ciclo inactivo del servidor. Los valores pueden oscilar entre 0 y 100. Al definir este parámetro en 0, se inhabilita el proceso de limpieza. Si se define en 100, permite al proceso de limpieza funcionar de forma continuada durante los ciclos inactivos del servidor. El valor predeterminado 10, permite al proceso de limpieza seguir desplazando memorias intermedias a la región de lavado de memorias intermedias durante los ciclos inactivos del servidor, siempre que las escrituras de base de datos no aumenten más de un 10%.
- **i/o accounting flush interval** especifica cuántas E/Ss de disco deberán acumularse antes de transferir los datos a *syslogins* para su uso en la contabilidad de cargos.
- **i/o polling process count** especifica el número de tareas que el planificador ejecutará antes de buscar terminaciones de E/S de disco y red.
- **identity burning set factor** determina el porcentaje de valores de columna IDENTITY potenciales disponible en cada bloque. El valor predeterminado 5000, libera el 05 por ciento de los valores de columna IDENTITY potenciales para su uso simultáneo.
- **identity grab size** permite a cada proceso de SQL Server reservar un bloque de valores de columna IDENTITY para inserciones en tablas que tengan una columna IDENTITY.
- **lock shared memory** deniega el intercambio de páginas de SQL Server al disco y permite al kernel del sistema operativo que evite el código de bloqueo de página interno del servidor.
- **lock promotion HWM** define el número máximo de bloqueos de página permitidos antes de que SQL Server escale a un bloqueo de tabla. El valor predeterminado es 200.

- **lock promotion LWM** define el número mínimo de bloqueos de página permitidos antes de que SQL Server escale a un bloqueo de tabla. El valor predeterminado es 200.
- **lock promotion PCT** define el porcentaje de bloqueos de página permitido antes de que SQL Server escale a un bloqueo de tabla. El valor predeterminado es 100.
- **max async i/os per engine** especifica el número máximo de solicitudes de E/S de disco asíncronas que puede ser significativo para una sola máquina al mismo tiempo.
- **max async i/os per server** especifica el número máximo de solicitudes de E/S de disco asíncronas que puede ser significativo para SQL Server al mismo tiempo.
- **max engine freelocks** especifica el número máximo de bloqueos disponibles en un caché de bloqueos libres de la máquina.
- **max online engines** controla el número de máquinas en un entorno multiprocesador simétrico.
- **max network packet size** define el tamaño máximo de paquetes de red que un programa cliente puede solicitar.
- **max number of network listeners** especifica el número máximo de oyentes de red que pueden abrirse de una vez.
- **memory alignment boundary** determina en qué límite se alinean los cachés de memorias intermedias.
- **min online engines** no se utiliza actualmente.
- **number of alarms** especifica el número de alarmas asignadas por SQL Server. Las alarmas se utilizan con el comando Transact-SQL **waitfor** .
- **number of devices** controla el número de dispositivos de base de datos que SQL Server puede utilizar. No incluye dispositivos utilizados para volcados de base de datos.
- **number of extent i/o buffers** asigna un número especificado de sectores (8 páginas de datos) para utilizarse mediante **create index** . No asigne a este parámetro un valor superior a 100.
- **number of index trips** especifica el número de veces que una página de índice envejecida se recicla en la cadena MRU.
- **number of languages in cache** es el número máximo de idiomas que el caché de idiomas puede contener simultáneamente. El valor predeterminado es 3.
- **number of locks** define el número de bloqueos disponibles. El valor de ejecución predeterminado es 5000.
- **number of mailboxes** especifica el número de estructuras de buzón que SQL Server asigna durante el arranque. Los buzones se utilizan para la comunicación y sincronización entre procesos.
- **number of messages** especifica el número de estructuras de mensaje asignadas por SQL Server durante el arranque. Los mensajes se utilizan junto con los buzones para la comunicación y sincronización entre procesos.
- **number of oam trips** especifica el número de veces que una página OAM envejecida se recicla en la cadena MRU .
- **number of open databases** define el número máximo de bases de datos que pueden abrirse al mismo tiempo en SQL Server. El valor de ejecución predeterminado es 12.

- **number of open objects** define el número máximo de objetos de base de datos que pueden abrirse al mismo tiempo en SQL Server. El valor de ejecución predeterminado es 500.
- **number of pre-allocated extents** especifica el número de estructuras de sector asignadas en una sola conexión al administrador de páginas.
- **number of remote connections** controla el límite de conexiones activas iniciadas hacia y desde este SQL Server. El valor predeterminado es 20.
- **number of remote logins** controla el número de conexiones de usuario activas procedentes de este SQL Server a servidores remotos. El valor predeterminado es 20.
- **number of remote sites** controla el número de sitios remotos simultáneos que pueden tener acceso a este SQL Server. El valor predeterminado es 10.
- **number of sort buffers** especifica el número de memorias intermedias utilizadas para contener páginas leídas desde tablas de entrada.
- **number of user connections** define el número máximo de conexiones de usuario que pueden conectarse a SQL Server al mismo tiempo. El valor máximo para su sistema se almacena en la variable global `@@max_connections` , y varía en función de la plataforma y el sistema operativo.
- **page lock spinlock ratio** especifica la relación de **bloqueos de giro** que protegen la tabla de desmenuzamiento de bloqueos de página internos.
- **page utilization percent** controla cuándo SQL Server realiza un barrido OAM (Object Allocation Map) para encontrar páginas no utilizadas. El valor de ejecución predeterminado es 95.
- **partition groups** especifica cuántos grupos de particiones es preciso asignar para el servidor. Los grupos de particiones son estructuras internas que SQL Server utiliza para controlar el acceso a particiones individuales de una tabla. SQL Server asigna grupos de particiones a una tabla cuando se divide la tabla en particiones o cuando se accede a ésta por primera vez después de reiniciar el servidor. Un grupo de particiones se compone de 16 cachés de particiones, cada uno de los cuales almacena información sobre una sola partición. Todos los cachés de un grupo de particiones se utilizan para almacenar información acerca de la misma tabla con particiones. El valor predeterminado 64 permite un máximo de 64 tablas con particiones abiertas y de 1024 (64 veces 16) particiones abiertas.
- **partition spinlock ratio** especifica el número de cachés de particiones que cada bloqueo de giro protege. Un bloqueo de giro de particiones impide que un proceso tenga acceso a un caché de particiones actualmente en uso por otro proceso. El valor predeterminado 32 (1 bloqueo de giro por cada 32 cachés de particiones) es correcto en la mayoría de los casos. Su aumento o reducción puede tener poco impacto en el rendimiento. El número sugerido de bloqueos de giro disponibles es un 10 por ciento del número total de particiones en uso en un momento determinado.
- **perform disk i/o on engine 0** se utiliza en máquinas multiprocesador para asociar una E/S de disco a la máquina 0 de SQL Server.
- **permission cache entries** determina el número de protectores de caché por tarea.
- **print deadlock information** habilita la impresión de información de bloqueos insolubles en el diario de errores.
- **print recovery information** define un conmutador que determina la información que SQL Server muestra en la consola durante la recuperación. El valor de ejecución

- predeterminado es 0, lo que significa que SQL Server muestra sólo el nombre de base de datos y un mensaje que indica que la recuperación está en curso.
- **procedure cache percent** especifica la cantidad de memoria asignada al caché de procedimiento después de satisfacer las necesidades de memoria de SQL Server. El valor de ejecución predeterminado es 20.
 - **recovery interval in minutes** define el número máximo de minutos por base de datos que SQL Server debería utilizar para terminar sus procedimientos de recuperación en caso de un fallo del sistema. El valor predeterminado es 5 (minutos por base de datos).
 - **remote server pre-read packets** controla el número de paquetes que un manipulador de sitio leerá previamente en conexiones con servidores remotos. El valor predeterminado es 3.
 - **runnable process search count** especifica el número de veces que una máquina se bloqueará buscando una tarea ejecutable antes de abandonar la CPU.
 - **shared memory starting address** determina la dirección virtual en la que SQL Server inicia su región de memoria compartida.
 - **size of auto identity column** define la precisión de columnas IDENTITY automáticamente creadas con la opción **sp_dboption "auto identity"** .
 - **sort page count** especifica la cantidad máxima de memoria que una operación de ordenación puede utilizar.
 - **sql server clock tick length** especifica la duración del impulso de reloj del servidor en microsegundos.
 - **stack guard size** especifica el tamaño del área de protección de pila.
 - **stack size** define el tamaño de la pila de ejecución de SQL Server.
 - **systemwide password expiration** es el número de días que las contraseñas permanecen activas después de cambiarse. El valor predeterminado es 0 (las contraseñas no vencen).
 - **table lock spinlock ratio** especifica el número de bloqueos de giro que protegen la tabla de desmenuzamiento de bloqueos de tabla.
 - **tape retention in days** define el número de días que se espera conservar cada cinta después de que se haya usado para un volcado de base de datos o de diario de transacciones. El valor de ejecución predeterminado es 0.
 - **tcp no delay** inhabilita la agrupación de paquetes TCP.
 - **time slice** define el número de milisegundos que un planificador de SQL Server permite la ejecución de un proceso de usuario. El valor de ejecución predeterminado es de 100 milisegundos.
 - **total data cache size** representa la cantidad de memoria actualmente disponible para utilizarla como un caché de datos. Se trata de un valor calculado que el usuario no puede configurar directamente.
 - **total memory** define la cantidad de memoria, en unidades de 2K, que SQL Server asigna desde el sistema operativo.
 - **upgrade version** se modifica mediante el programa de versión mejorada suministrado con versiones nuevas.
 - **user log cache size** especifica el tamaño (en bytes) de cada caché de diario del usuario.
 - **user log cache spinlock ratio** especifica el número de cachés de diario de usuario por bloqueo de giro de cachés de diario de usuario. .

ANEXO 2

VALORES EN EVENTO Y COLUMNA EXTRAINFO

Anexo 2

Valores en evento y columna extrainfo.

Los valores de columna de evento que pertenecen a cada evento de auditoría son puestos en una lista en el anexo 2.

Evento	Opción de auditoría	Comando o acceso auditado	extrainfo
1	adhoc	User-defined audit record	extrainfo está lleno por el parámetro de texto de sp_addauditrecord
2	alter	alter database	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i>– ALTER SIZE • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if a set proxy is in effect
3	alter	alter table	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – ADD COLUMN, REPLACE COLUMN, ADD CONSTRAINT, or DROP CONSTRAINT • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if a set proxy is in effect
4	bcp	bcp in	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – NULL • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy in effect
6	bind	sp_bindefault	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – NULL • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – Name of default • <i>Proxy information</i> – Original login name, if set proxy in effect

Evento	Opción de auditoría	Comando o acceso uditado	extrainfo
7	bind	sp_bindmsg	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – NULL • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – Message ID • <i>Proxy information</i> – Original login name, if set proxy in effect
8	bind	sp_bindrule	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – NULL • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – Name of the rule • <i>Proxy information</i> – Original login name, if set proxy in effect
9	create	create database	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – NULL • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy in effect
10	create	create table	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – NULL • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy in effect
11	create	create procedure	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – NULL • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy in effect
12	create	create trigger	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – NULL • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy in effect

Evento	Opción de auditoría	Comando auditado	extrainfo
13	create	create rule	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – NULL • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy in effect
14	create	create default	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – NULL • <i>Previous value</i> – NULL • <i>Other information</i> – NULL • <i>Current value</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy in effect
15	create	sp_addmessage	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – NULL • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – Message Number • <i>Proxy information</i> – Original login name, if set proxy in effect
16	create	create view	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – NULL • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy in effect
17	dbaccess	Any access to the database by any user	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – USE CMD or OUTSIDE REFERENCE • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy in effect
18	delete	delete from a table	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – DELETE • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy in effect

Evento	Opción de auditoría	Comando o acceso auditado	extrainfo
19	delete	delete from a view	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – DELETE • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy in effect
20	disk	disk init	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – disk init • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i>– Name of the disk • <i>Proxy information</i> – Original login name, if set proxy in effect
21	disk	disk refit	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – disk refit • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i>– Name of the disk • <i>Proxy information</i> – Original login name, if set proxy in effect
22	disk	disk reinit	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – disk reinit • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i>– Name of the disk • <i>Proxy information</i> – Original login name, if set proxy in effect
23	disk	disk mirror	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – disk mirror • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i>– Name of the disk • <i>Proxy information</i> – Original login name, if set proxy in effect
24	disk	disk unmirror	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – disk unmirror • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i>– Name of the disk • <i>Proxy information</i> – Original login name, if set proxy in effect

Evento	Opción de auditoría	Comando o acceso auditado	extrainfo
25	disk	disk remirror	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – disk remirror • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i>– Name of the disk • <i>Proxy information</i> – Original login name, if set proxy in effect
26	drop	drop database	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – NULL • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy in effect
27	drop	drop table	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – NULL • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy in effect
28	drop	drop procedure	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – NULL • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy in effect
29	drop	drop trigger	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – NULL • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy in effect
30	drop	drop rule	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – NULL • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy in effect

Evento	Opción de auditoría	Comando O acceso auditado	extrainfo
31	drop	drop default	<ul style="list-style-type: none"> • Roles – Current active roles • Subcommand – NULL • Previous value – NULL • Current value – NULL • Other information – NULL • Proxy information – Original login name, if set proxy in effect
32	drop	sp_dropmessage	<ul style="list-style-type: none"> • Roles – Current active roles • Subcommand – NULL • Previous value – NULL • Current value – NULL • Other information – Message number • Proxy information – Original login name, if set proxy in effect
33	drop	drop view	<ul style="list-style-type: none"> • Roles – Current active roles • Subcommand – NULL • Previous value – NULL • Current value – NULL • Other information – NULL
34	dump	dump database	<ul style="list-style-type: none"> • Roles – Current active roles • Subcommand – NULL • Previous value – NULL • Current value – NULL • Other information – NULL • Proxy information – Original login name, if set proxy in effect
35	dump	dump transaction	<ul style="list-style-type: none"> • Roles – Current active roles • Subcommand – NULL • Previous value – NULL • Current value – NULL • Other information – NULL • Proxy information – Original login name, if set proxy in effect
36	errors	Fatal error	<ul style="list-style-type: none"> • Roles – Current active roles • Subcommand – NULL • Previous value – NULL • Current value – NULL • Other information – Error number.Severity.State • Proxy information – Original login name, if set proxy in effect

Evento	Opción de auditoría	Comando o acceso auditado	extrainfo
37	errors	Non-fatal error	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – NULL • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – Error number.Severity.State • <i>Proxy information</i> – Original login name, if set proxy in effect
38	exec_procedure	Execution of a procedure	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – NULL • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – All input parameters • <i>Proxy information</i> – Original login name, if set proxy in effect
39	exec_trigger	Execution of a trigger	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – NULL • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy in effect
40	grant	grant	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – NULL • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy in effect
41	insert	insert into a table	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> • If insert – INSERT • If select into – INSERT INTO followed by the fully qualified object name • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy in effect

Evento	Opción de auditoría	Comando O acceso auditado	extrainfo
42	insert	insert into a view	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – INSERT • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if a set proxy is in effect
43	load	load database	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – NULL • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy in effect
44	load	load transaction	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – NULL • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy in effect
45	login	Any login to Adaptive Server	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – NULL • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – Host name of the machine from which login was done • <i>Proxy information</i> – Original login name, if set proxy in effect
46	logout	Any logouts from Adaptive Server	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – NULL • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – Host name of the machine from which login was done • <i>Proxy information</i> – Original login name, if set proxy in effect

Evento	Opción de auditoría	Comando o acceso auditado	extrainfo
47	revoke	revoke	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – NULL • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy in effect
48	rpc	Remote procedure call from another server	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – Name of client program • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – Server name, host name of the machine from which the RPC was done. • <i>Proxy information</i> – Original login name, if set proxy in effect
49	rpc	Remote procedure call to another server	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – Procedure name • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy in effect
50	security	Server start	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – NULL • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> <i>-dmasterdevicename</i> <i>-iinterfaces file path</i> <i>-Sservername</i> <i>-errorfilename</i> • <i>Proxy information</i> – Original login name, if set proxy in effect
51	security	Server shutdown	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – shutdown • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy in effect

Evento	Opción de auditoría	Comando o acceso auditado	extrainfo
55	security	Role toggling	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – NULL • <i>Previous value</i> – “on” or “off” • <i>Current value</i> – “on” or “off” • <i>Other information</i> – Name of the role being set • <i>Proxy information</i> – Original login name, if set proxy in effect
61	table_access	Table access	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – SELECT, SELECT INTO, INSERT, UPDATE, DELETE, REFERENCE, READTEXT, or WRITETEXT • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy in effect
62	select	select from a table	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – SELECT INTO, SELECT, or READTEXT • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy in effect
63	select	select from a view	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – SELECT, SELECT INTO, or READTEXT • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy in effect
64	truncate	truncate table	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – NULL • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy in effect

Evento	Opción de auditoría	Comando auditado	extrainfo
67	unbind	sp_unbindefault	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – NULL • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy in effect
68	unbind	sp_unbindrule	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – NULL • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy in effect
69	unbind	sp_unbindmsg	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – NULL • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy in effect
70	update	update to a table	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – UPDATE or WRITETEXT • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy in effect
71	update	update to a view	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – UPDATE or WRITETEXT • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy in effect
73	Nota: Este evento es auditado automáticamente. Esto no es controlado por una opción de auditoría.	Turning the auditing parameter on with sp_configure	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – NULL • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy in effect

Evento	Opción de auditoría	Comando o acceso auditado	extrainfo
74	<p>Nota: Este evento es auditado automáticamente. Esto no es controlado por una opción de auditoría.</p>	Turning the auditing parameter off with sp_configure	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – NULL • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy in effect
76	security	Regeneration of a password by a System Security Officer (SSO)	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – Setting SSO password • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – Login name • <i>Proxy information</i> – Original login name, if set proxy in effect
80	security	proc_role within a system procedure	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – NULL • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – Required roles • <i>Proxy information</i> – Original login name, if set proxy in effect
81	dbcc	dbcc	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – The dbcc subcommand name • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy in effect

Evento	Opción de auditoría	Comando o acceso auditado	extrainfo
82	security	sp_configure	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – Name of the configuration parameter • <i>Previous value</i> – Old parameter value if command is setting a new value • <i>Current value</i> – New parameter value if command is setting a new value • <i>Other information</i> – Number of configuration parameter, if a parameter is being set; name of configuration file, if a configuration file is being used to set parameters • <i>Proxy information</i> – Original login name, if set proxy in effect
83	security	online database	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – NULL • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy in effect
84	setuser	setuser	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – NULL • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – Name of the user being set • <i>Proxy information</i> – Original login name, if a set proxy is in effect
85	func_obj_access, func_dbaccess	Accesses to objects and databases via Transact-SQL functions	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – NULL • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy in effect

Evento	Opción de auditoría	Comando o acceso auditado	extrainfo
85	security	valid_user	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – valid_user • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy in effect
88	security	set proxy or set	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – NULL • <i>Previous value</i> – Previous suid • <i>Current value</i> – New suid • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy or set session authorization had no parameters; otherwise, NULL.
92	cmdtxt	All actions of a particular user, or by users with a particular role	<ul style="list-style-type: none"> • <i>Roles</i> – Current active roles • <i>Subcommand</i> – NULL • <i>Previous value</i> – NULL • <i>Current value</i> – NULL • <i>Other information</i> – NULL • <i>Proxy information</i> – Original login name, if set proxy in effect

APENDICE A

GUÍA PARA LA

ELABORACIÓN DE

POLÍTICAS

APENDICE A.

GUÍA PARA LA ELABORACIÓN DE POLÍTICAS DE SEGURIDAD

Esta metodología es potencialmente útil para el desarrollo, implementación, mantenimiento y eliminación de un conjunto completo de políticas – tanto de seguridad como en otras áreas

Es frecuente que las personas involucradas con seguridad informática tengan una visión estrecha de lo que significa desarrollar las políticas de seguridad, pues no basta con escribirlas y pretender ponerlas en práctica. En ocasiones se incluye la asignación de responsables, se realizan actividades para dar a conocerlas y, quizá, se supervise su cumplimiento; pero esto tampoco basta. Muchas políticas de seguridad informática fallan ya que se desconoce lo que implica realmente desarrollarlas.

Es importante resaltar que una política de seguridad tiene un ciclo de vida completo mientras esta vigente. Este ciclo de vida incluye un esfuerzo de investigación, la labor de escribirla, lograr que las directivas de la organización la acepten, conseguir que sea aprobada, lograr que sea diseminada a través de la empresa, concienciar a los usuarios de la importancia de la política, conseguir que la acaten, hacerle seguimiento, garantizar que esté actualizada y, finalmente, suprimirla cuando haya perdido vigencia. Si no se tiene en cuenta este ciclo de vida se corre el riesgo de desarrollar políticas que sean poco tenidas en cuenta, incompletas, redundantes, sin apoyo pleno por parte de los usuarios y las directivas, superfluas o irrelevantes.

Este documento presenta algunos puntos que deben tenerse en cuenta al desarrollar algún tipo de política de seguridad informática.

¿Por qué tener políticas escritas?

Existen varias razones por las cuales es recomendable tener políticas escritas en una organización. La siguiente es una lista de algunas de estas razones.

- Para cumplir con regulaciones legales o técnicas
- Como guía para el comportamiento profesional y personal
- Permite unificar la forma de trabajo de personas en diferentes lugares o momentos que tengan responsabilidades y tareas similares
- Permiten recoger comentarios y observaciones que buscan atender situaciones anormales en el trabajo
- Permite encontrar las mejores prácticas en el trabajo
- Permiten asociar la filosofía de una organización (lo abstracto) al trabajo (lo concreto)

Definición de política

Es importante aclarar el término política desde el comienzo. ¿Qué queremos dar a entender cuando decimos POLITICA o ESTÁNDAR o MEJOR PRÁCTICA o GUÍA o PROCEDIMIENTO? Estos son términos utilizados en seguridad informática todos los días, pero algunas veces son utilizados correctamente, otras veces no.

- **Política.** Declaración general de principios que presenta la posición de la administración para un área de control definida. Las políticas se elaboran con el fin de que tengan aplicación a largo plazo y guíen el desarrollo de reglas y criterios más específicos que aborden situaciones concretas. Las políticas son desplegadas y soportadas por estándares, mejores prácticas, procedimientos y guías.

Las políticas deben ser pocas (es decir, un número pequeño), deben ser apoyadas y aprobadas por las directivas de la empresa, y deben ofrecer direccionamientos a toda la organización o a un conjunto importante de dependencias. Por definición, las políticas son obligatorias y la incapacidad o imposibilidad para cumplir una política exige que se apruebe una excepción.

- **Estándar.** Regla que especifica una acción o respuesta que se debe seguir a una situación dada. Los estándares son orientaciones obligatorias que buscan hacer cumplir las políticas.

Los estándares sirven como especificaciones para la implementación de las políticas: son diseñados para promover la implementación de las políticas de alto nivel de la organización antes que crear nuevas políticas.

- **Mejor práctica.** Es una regla de seguridad específica a una plataforma que es aceptada a través de la industria al proporcionar el enfoque más efectivo a una implementación de seguridad concreta.

Las mejores prácticas son establecidas para asegurar que las características de seguridad de sistemas utilizados con regularidad estén configurados y administrados de manera uniforme, garantizando un nivel consistente de seguridad a través de la organización.

- **Guía.** Una guía es una declaración general utilizada para recomendar o sugerir un enfoque para implementar políticas, estándares y buenas prácticas.

Las guías son, esencialmente, recomendaciones que deben considerarse al implementar la seguridad. Aunque no son obligatorias, serán seguidas a menos que existan argumentos documentados y aprobados para no hacerlo.

- **Procedimiento.** Los procedimientos definen específicamente cómo las políticas, estándares, mejores prácticas y guías serán implementados en una situación

dada. Los procedimientos son dependientes de la tecnología o de los procesos y se refieren a plataformas, aplicaciones o procesos específicos.

Son utilizados para delinear los pasos que deben ser seguidos por una dependencia para implementar la seguridad relacionada a dicho proceso o sistema específico.

Generalmente los procedimientos son desarrollados, implementados y supervisados o del sistema.

Los procedimientos por el dueño del proceso seguirán las políticas de la organización, los estándares, las mejores prácticas y las guías tan cerca como les sea posible, y a la vez se ajustarán a los requerimientos procedimentales o técnicos establecidos dentro de la dependencia donde ellos se aplican.

Los términos antes mencionados, además de presentar una definición de los términos utilizados en la enunciación e implementación de políticas, muestra una jerarquía entre las definiciones.

Un ejemplo de los requerimientos de seguridad interrelacionados podría ser:

1. En el nivel más alto, se puede elaborar una POLÍTICA, para toda la organización, que obligue a “garantizar seguridad en el correo electrónico cuyo contenido sea información confidencial”.

2. Esta POLÍTICA podría ser soportada por varios ESTÁNDARES, incluyendo por ejemplo, que los mensajes de este tipo sean enviados utilizando algún sistema de aprobado por la empresa y que sean borrados de manera segura después de su envío.

3. Una MEJOR PRÁCTICA, en este ejemplo, podría estar relacionada sobre la manera de configurar el correo sobre un tipo específico de sistema (Windows o Linux) con el fin de garantizar el cumplimiento de la POLÍTICA y del ESTÁNDAR.

4. Los PROCEDIMIENTOS podrían especificar requerimientos para que la POLÍTICA y los ESTÁNDARES que la soportan, sean aplicados en una dependencia específica, por ejemplo la Oficina de Control Interno.

5. Finalmente, las GUÍAS podrían incluir información sobre técnicas, configuraciones y secuencias de comandos recomendadas que deben seguir los usuarios para asegurar la información confidencial enviada y recibida a través del servicio de correo electrónico.

Nótese que, en muchas ocasiones, el termino “política” es utilizado en un sentido genérico para aplicarlo a cualquiera de los tipos de requerimientos de seguridad expuestos.

En este documento se llamará política, de manera genérica, a todos los requerimientos de seguridad mencionados antes y POLÍTICA (en mayúsculas) a las políticas propiamente dichas.

ETAPAS EN EL DESARROLLO DE UNA POLITICA

En la figura A.1 se muestran las etapas en el desarrollo de una política:

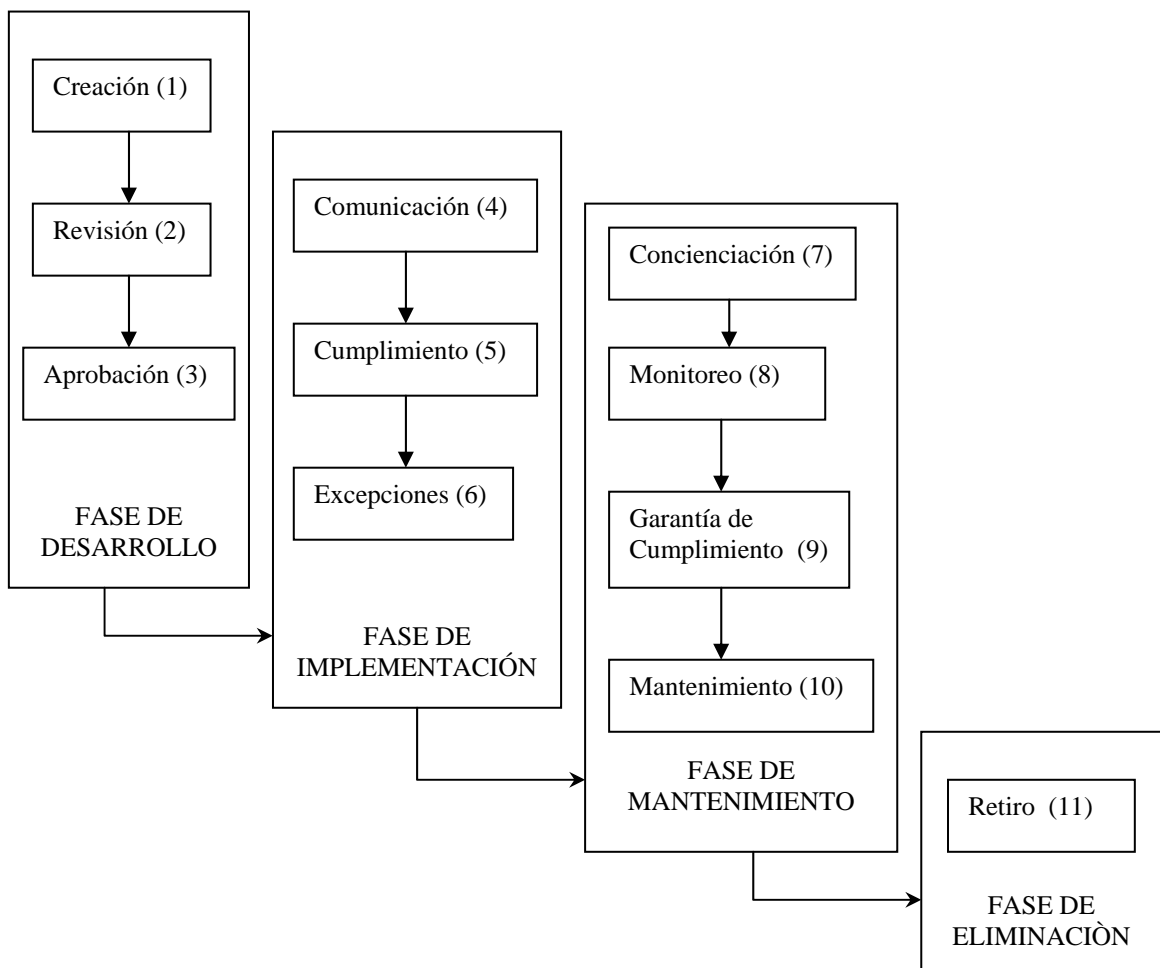


Figura A.1 Etapas en el desarrollo de una política

Hay varias etapas que deben realizarse a través de “la vida” de una política. Estas etapas pueden ser agrupadas en 4 fases.

- 1. Fase de desarrollo:** durante esta fase la política es creada, revisada y aprobada.

2. **Fase de implementación:** en esta fase la política es comunicada y acatada (o no cumplida por alguna excepción).
3. **Fase de mantenimiento:** los usuarios deben ser conscientes de la importancia de la política, su cumplimiento debe ser monitoreado, se debe garantizar su cumplimiento y se le debe dar mantenimiento (actualizarla)
4. **Fase de eliminación:** La política se retira cuando no se requiera más.

Creación: Planificación, investigación, documentación, y coordinación de la política

El primer paso en la fase de desarrollo de una política es la planificación, la investigación y la redacción de la política o, tomado todo junto, la creación. La creación de una política implica identificar por qué se necesita la política (por ejemplo, requerimientos legales, regulaciones técnicas, contractuales u operacionales); determinar el alcance y la aplicabilidad de la política, los roles y las responsabilidades inherentes a la aplicación de la política y garantizar la factibilidad de su implementación. La creación de una política también incluye la investigación para determinar los requerimientos organizacionales para desarrollar las políticas (es decir, que autoridades deben aprobarla, con quién se debe coordinar el desarrollo y estándares del formato de redacción), y la investigación de las mejores prácticas en la industria para su aplicabilidad a las necesidades organizacionales actuales. De esta etapa se tendrá como resultado la documentación de la política de acuerdo con los procedimientos y estándares de la empresa, al igual que la coordinación con entidades internas y externas que la política afectará, para obtener información y su aceptación. En general, la creación de una política es la función más fácil de entender en el ciclo de vida de desarrollo de una política.

Revisión: Evaluación independiente de la política

La revisión de la política es la segunda etapa en la fase de desarrollo del ciclo de vida. Una vez la documentación de la política ha sido creada y la coordinación inicial ha sido iniciada, esta debe ser remitida a un grupo (o un individuo) independiente para su evaluación antes de su aprobación final. Hay varios beneficios de la revisión independiente: una política más viable a través del escrutinio de individuos que tienen una perspectiva diferente o más vasta que la persona que redactó la política; apoyo más amplio para la política a través de un incremento de involucrados; aumento en el número de credibilidad en la política gracias a la información recibida de diferentes especialistas del grupo de revisión. Propio de esta etapa es la presentación de la política a los revisores, ya sea de manera formal o informal, exponiendo cualquier punto que puede ser importante para la revisión, explicando su objetivo, el contexto y los beneficios potenciales de la política y justificando por qué es necesaria. Como parte de esta función, se espera que el creador de la política recopile los comentarios y las recomendaciones para realizar cambios en la política y efectuar todos los ajustes y las revisiones necesarias para obtener una versión final de la política lista para la aprobación por las directivas.

Aprobación: Obtener la aprobación de la política por parte de las directivas

El paso final en la fase de desarrollo de la política es la aprobación. El objetivo de esta etapa es obtener el apoyo de la administración de la empresa, a través de la firma de una persona ubicada en una posición de autoridad.

La aprobación permite iniciar la implementación de la política. Requiere que el proponente de la política haga una selección adecuada de la autoridad de aprobación, que coordine con dicho funcionario, presente las recomendaciones emitidas durante la etapa de revisión y haga el esfuerzo para que sea aceptada por la administración. Puede ocurrir que por incertidumbre de la autoridad de aprobación sea necesaria una aprobación temporal.

Comunicación: Difundir la política

Una vez la política ha sido aprobada formalmente, se pasa a la fase de implementación. La comunicación de la política es la primera etapa que se realiza en esta fase. La política debe ser inicialmente difundida a los miembros de la comunidad universitaria o a quienes sean afectados directamente por la política (contratistas, proveedores, usuarios de cierto servicio, etc.). Esta etapa implica determinar el alcance y el método inicial de distribución de la política (es posible que deban tenerse en cuenta factores como la ubicación geográfica, el idioma, la cultura y línea de mando que será utilizada para comunicar la política). Debe planificarse esta etapa con el fin de determinar los recursos necesarios y el enfoque que debe ser seguido para mejorar la visibilidad de la política.

Cumplimiento: Implementar la política

La etapa de cumplimiento incluye actividades relacionadas con la ejecución de la política. Implica trabajar con otras personas de la empresa, vicerrectores, decanos, directores de departamento y los jefes de dependencias (de división o de sección) para interpretar cuál es la mejor manera de implementar la política en diversas situaciones y oficinas; asegurando que la política es entendida por aquellos que requieren implementarla, monitorearla, hacerle seguimiento, reportar regularmente su cumplimiento y medir el impacto inmediato de la política en las actividades operativas. Dentro de estas actividades está la elaboración de informes a la administración del estado de la implementación de la política.

Excepciones: Gestionar las situaciones donde la implementación no es posible

Debido a problemas de coordinación, falta de personal y otros requerimientos operacionales, no todas las políticas pueden ser cumplidas de la manera que se pensó al comienzo. Por esto, es probable que se requieran excepciones a la política para permitir a ciertas oficinas o personas el no cumplimiento de la política. Debe establecerse un proceso para garantizar que las solicitudes de excepciones son registradas, seguidas, evaluadas, enviadas para aprobación o desaprobación,

documentadas y vigiladas a través del periodo de tiempo establecido para la excepción. El proceso también debe permitir excepciones permanentes a la política al igual que la no aplicación temporal por circunstancias de corta duración.

Concienciación: Garantiza la concienciación continuada de la política

La etapa de concienciación de la fase de mantenimiento comprende los esfuerzos continuos realizados para garantizar que las personas están concientes de la política y buscan facilitar su cumplimiento.

Esto es hecho al definir las necesidades de concienciación de los diversos grupos de audiencia dentro de la organización (directivos, jefes de dependencias, usuarios, etc.).

En relación con la adherencia a la política, determinar los métodos de concienciación más efectivos para cada grupo de audiencia (es decir, reuniones informativas, cursos de entrenamiento, mensajes de correo, etcétera); y desarrollo y difusión de material de concienciación (presentaciones, afiches, circulares, etc.).

La etapa de concienciación también incluye esfuerzos para integrar el cumplimiento de la política y retroalimentación sobre el control realizado para su cumplimiento.

Monitoreo: Seguimiento y reporte del cumplimiento de la política

Durante la fase de mantenimiento, la etapa de monitoreo es realizada para seguir y reportar la efectividad de los esfuerzos en el cumplimiento de la política. Esta información se obtiene de la observación de los docentes, estudiantes, empleados y los cargos de supervisión, mediante auditorías formales, evaluaciones, inspecciones, revisiones y análisis de los reportes de contravenciones y de las actividades realizadas en respuesta a los incidentes.

Esta etapa incluye actividades continuas para monitorear el cumplimiento o no de la política a través de métodos formales e informales y el reporte de las deficiencias encontradas a las autoridades apropiadas.

Garantía de cumplimiento: Afrontar las contravenciones de la política

La etapa de garantía de cumplimiento de las políticas incluye las respuestas de la administración a actos u omisiones que tengan como resultado contravenciones de la política con el fin de prevenir que sigan ocurriendo.

Esto significa que una vez una contravención sea identificada, la acción correctiva debe ser determinada y aplicada a los procesos (revisión del proceso y mejoramiento), a la tecnología (actualización) y a las personas (acción disciplinaria) involucrados en la contravención con el fin de reducir la probabilidad de que vuelva a ocurrir. Se recomienda incluir información sobre las acciones correctivas adelantadas para garantizar el cumplimiento en la etapa de concienciación.

Mantenimiento: Asegurar que la política esté actualizada

La etapa de mantenimiento esta relacionada con el proceso de garantizar la vigencia y la integridad de la política. Esto incluye hacer seguimiento a las tendencias de cambios en la tecnología, en los procesos, en las personas, en la organización, en el enfoque del negocio, etcétera, que puede afectar la política; recomendando y coordinando modificaciones resultado de estos cambios, documentándolos en la política y registrando las actividades de cambio.

Esta etapa también garantiza la disponibilidad continuada de la política para todas las partes afectadas por ella, al igual que el mantenimiento de la integridad de la política a través de un control de versiones efectivo. Cuando se requieran cambios a la política, las etapas realizadas antes deben ser revisadas, en particular las etapas de revisión, aprobación, comunicación y garantía de cumplimiento.

Retiro: Prescindir de la política cuando no se necesite más

Después que la política ha cumplido con su finalidad y no es necesaria (por ejemplo, la empresa cambió la tecnología a la cual aplicaba o se creó una nueva política que la reemplazó) entonces debe ser retirada. La etapa de retiro corresponde a la fase de eliminación del ciclo de vida de la política, y es la etapa final del ciclo.

Esta función implica retirar una política superflua del inventario de políticas activas para evitar confusión, archivarla para futuras referencias y documentar la información sobre la decisión de retirar la política (es decir, la justificación, quién autorizó, la fecha, etcétera).

Estas cuatro fases del ciclo de vida reúnen 11 etapas diferentes que deben seguirse durante el ciclo de vida de una política específica. No importa como se agrupen, tampoco importa si estas etapas son abreviadas por necesidades de inmediatez, pero cada etapa debe ser realizada.

Si en la fase de desarrollo la empresa intenta crear una política sin una revisión independiente, se tendrán políticas que no estarán bien concebidas ni serán bien recibidas por la comunidad universitaria.

En otras circunstancias, y por falta de visión, puede desearse omitir la etapa de excepciones de la fase de implementación, pensando equivocadamente que no existirán circunstancias para su no cumplimiento. También se podría descuidar la etapa de mantenimiento, olvidando la importancia de mantener la integridad y la vigencia de las políticas. Muchas veces se encuentran políticas inoficiosas en los documentos de importantes organizaciones, indicando que la etapa de retiro no está siendo realizada.

No sólo se requiere que las once etapas sean realizadas, algunas de ellas deben ser ejecutadas de manera cíclica, en particular mantenimiento, concienciación, monitoreo, y garantía de cumplimiento.

Algunas prácticas recomendadas para escribir una política

Sin importar que una política se enuncie formal o informalmente, esta debe incluir 12 tópicos:

1. La declaración de la política (cuál es la posición de la administración o qué es lo que se desea regular)
2. Nombre y cargo de quien autoriza o aprueba la política
3. Nombre de la dependencia, del grupo o de la persona que es el autor o el proponente de la política
4. Debe especificarse quién debe acatar la política (es decir, a quién está dirigida) y quién es el responsable de garantizar su cumplimiento
5. Indicadores para saber si se cumple o no la política
6. Referencias a otras políticas y regulaciones en las cuales se soporta o con las cuales tiene relación
7. Enunciar el proceso para solicitar excepciones
8. Describir los pasos para solicitar cambios o actualizaciones a la política
9. Explicar qué acciones se seguirán en caso de contravenir la política
10. Fecha a partir de la cual tiene vigencia la política
11. Fecha cuando se revisará la conveniencia y la obsolescencia de la política
12. Incluir la dirección de correo electrónico, la página web y el teléfono de la persona o personas que se pueden contactar en caso de preguntas o sugerencias

Otras prácticas que se recomiendan seguir son:

1. Uso de lenguaje sencillo (evitar lenguaje técnico hasta donde sea posible)
2. Escribir la política como si fuese a utilizarse siempre
3. Debe escribirse de tal forma que pueda ser entendida por cualquier miembro de la empresa
4. Se debe evitar describir técnicas o métodos particulares que definan una sola forma de hacer las cosas
5. Cuando se requiera, hacer referencia explícita y clara a otras dependencias de la organización
6. Utilizar la guía para la presentación de documentos escritos de la empresa

ASPECTOS IMPORTANTES PARA DEFINIR RESPONSABILIDADES EN EL DESARROLLO DE POLÍTICAS

En muchas ocasiones se asume que la función seguridad informática –ya sea un grupo o un individuo- sea la encargada de adelantar la gran mayoría de las etapas en el ciclo de vida de una política y que también actué como el proponente para la mayoría de las políticas relacionadas con la protección de los activos informáticas. Por diseño, la función seguridad informática tiene la responsabilidad a largo plazo y debe ejecutar las tareas diarias para asegurar los activos de información y por tanto, debe ser el dueño y debe ejercer control centralizado sobre las POLÍTICAS, ESTÁNDARES, MEJORES PRÁCTICAS, PROCEDIMIENTOS Y GUÍAS relacionados con seguridad informática.

Pero en ningún caso la función seguridad informática debe ser el proponente *de todas las políticas relacionadas con seguridad*, ni tampoco debe realizar todas las etapas de desarrollo en el ciclo de vida de la política.

Por ejemplo, los dueños de los sistemas de información deben tener la responsabilidad para establecer los requerimientos necesarios para implementar las políticas de la universidad para sus propios sistemas. Cuando existan requerimientos de seguridad en cierta dependencia que deben cumplir con políticas de nivel superior, su proponente debe ser la dependencia que tiene interés en garantizar la efectividad de dicha política.

Aunque el proponente o dueño de una política tiene una responsabilidad continúa sobre el ciclo de vida completo de política, hay varios factores que influyen sobre la determinación y la decisión de quien o que dependencia tienen responsabilidad directa para realizar etapas específicas del ciclo de vida de la política en una organización.

Entre estos factores incluyen:

- 1. Separación de tareas.** El principio de separación de tareas debe ser aplicado para determinar la responsabilidad de una etapa en particular para garantizar que los chequeos y ajustes necesarios sean aplicados. Para proveer una perspectiva más amplia y diferente, un directivo, o un grupo que sea independiente del proponente, debe revisar la política y una directiva, superior al proponente, debe encargarse de aprobar la política. O, para disminuir los posibles conflictos de intereses. La función auditoria (o control interno), como oficina independiente dentro de la organización, debe ser encargada del monitoreo del cumplimiento de la política, en tanto que grupos u organizaciones de auditoria externos deben ser invitados a realizar una evaluación independiente del cumplimiento de las políticas para ser consistentes con el principio de separación de tareas.
- 2. Eficiencia.** Adicionalmente, por razones de eficiencia, dependencias diferentes a la proponente deben tener alguna responsabilidad para la realización de ciertas etapas del ciclo de vida del desarrollo de una política. Por ejemplo, la difusión y la comunicación de la política sería mejor realizada si se encomendara a la dependencia encargada de estas funciones dentro de la organización. Por otra parte, basados en la eficiencia, los esfuerzos de concienciación serían asignados a la función capacitación de la universidad, aun cuando puede ocurrir que el

personal de capacitación no este entrenado específicamente en la labor de la concienciación de la política de seguridad. En este último caso, sería mejor que la desarrollara la función de seguridad informática.

3. **Alcance del control.** Límites en el alcance del control que la dependencia proponente puede ejercer tiene impacto sobre quien debe ser el ponente de una política específica. Normalmente, el proponente solo puede jugar un papel limitado en el monitoreo y en la garantía del cumplimiento de la política debido a que el no puede estar en todos los sitios, en todo momento, donde esta debe ser implementada. Los vicerrectores, decanos, directores de departamento, jefes de oficinas, de dependencias, de divisiones o de secciones, por su ubicación jerárquica, están cerca de las personas (docentes, estudiantes o empleados) a quienes afecta la política de seguridad y por tanto están en una mejor posición para monitorear de manera efectiva y garantizar el cumplimiento de la política. Por tanto deben asumir la responsabilidad de estas etapas. Estos funcionarios pueden garantizar que la política esta siendo seguida y que las contravenciones se manejan de manera adecuada.
4. **Autoridad.** Límites en la autoridad que un individuo o una dependencia ejerce, puede determinar la habilidad para desarrollar exitosamente una etapa del ciclo de vida de una política. La efectividad de una política, a menudo, puede ser juzgada por su visibilidad y el énfasis que la administración de la universidad coloquen.

La efectividad de una política, en muchos casos, depende de la autoridad en la cual la política se soporta. Para que una política tenga un soporte en toda la organización, el directivo que la aprueba debe tener un reconocido grado de autoridad sobre una gran parte de la universidad.

Normalmente, la función de seguridad informática de la organización no goza del nivel de reconocimiento ideal a través de toda la organización y requiere el soporte de directivas de nivel superior para cumplir con su misión. En consecuencia, la aceptación y el cumplimiento de las políticas de seguridad informática tienen mayor probabilidad de darse cuando la autoridad que la aprueba es de nivel superior.

5. **Conocimiento.** La ubicación del proponente en la universidad puede inducir a deficiencias en el conocimiento del entorno en el cual la política será implementada, entorpeciendo su efectividad. El empleo de un comité que realice la evaluación de políticas puede ofrecer un entendimiento más amplio de las operaciones que afectará la política.

Un organismo de este tipo puede ayudar a garantizar que la política sea escrita con el fin de promover su aceptación y su implementación exitosa y puede ser útil para prever problemas de implementación y para evaluar efectivamente situaciones donde las excepciones a la política pueden ser justificadas. De acuerdo con el alcance de la política, la labor de evaluación puede ser realizada por el comité nacional de informática o los comités de informática de las sedes.

6. **Aplicabilidad.** Finalmente, la aplicabilidad de la política también afecta la responsabilidad en las etapas de desarrollo del ciclo de vida de la política. ¿Qué áreas de la universidad son afectadas por la política? ¿La política aplica a una

sola dependencia, sólo a los usuarios de una tecnología en particular o a toda la universidad? Si la aplicabilidad de una política esta limitada a una sola dependencia, entonces la jefatura de la dependencia debe tener su propia política. Sin embargo, si la política es aplicable a toda la universidad, entonces una dependencia de alto nivel debe asumir la responsabilidad en relación con la política.

RESPONSABILIDADES EN EL MODELO DE CICLO DE VIDA DE LA POLITICA

Para garantizar que todas las etapas del ciclo de vida sean realizadas de manera apropiada y las responsabilidades para su ejecución sean asignadas adecuadamente, la universidad debe establecer un marco de referencia para facilitar el entendimiento, promover la aplicación consistente, establecer una estructura jerárquica para soportar mutuamente los distintos niveles de políticas, y acomodar efectivamente los frecuentes cambios tecnológicos y organizacionales. Modelo de responsabilidad por etapa para cada tipo de política, se muestra en la tabla A.1.

Responsabilidad				
Etapa	políticas	Estándares y buenas prácticas	Guías	Procedimientos
Creación	Función seguridad informática	Función seguridad informática e ingenieros con conocimiento en el área	Función seguridad informática e ingenieros con conocimientos en el área	Dependencia que los propone
Revisión	Comité de evaluación de políticas	Comité de evaluación de políticas	Comité de evaluación de políticas	función seguridad informática y director de dependencia
Aprobación	Rector general o vicerrector general	Rector general o vicerrector general	Rector general o vicerrector general	Directivo del área
Comunicación	Docentes, estudiantes, empleados y funcionarios con responsabilidades de supervisión en toda la universidad	Docentes, estudiantes, empleados y funcionarios con responsabilidades de supervisión en toda la universidad	Docentes, estudiantes, empleados y funcionarios con responsabilidades de supervisión en toda la universidad	Empleados y funcionarios con responsabilidades de supervisión de la dependencia

Excepciones	Comité de evaluación de políticas	Comité de evaluación de políticas	No aplica	Directivo del área
concienciación	función seguridad informática y función capacitación	función seguridad informática y función capacitación	función seguridad informática y función capacitación	Jefe de dependencia
Monitoreo	Funcionarios con responsabilidades de supervisión, función seguridad informática y función auditoria	Funcionarios con responsabilidades de supervisión, función seguridad informática y función auditoria	Funcionarios con responsabilidades de supervisión, función seguridad informática y función auditoria	Funcionarios con responsabilidades de supervisión y personas asignadas dentro de la dependencia, función seguridad informática y función auditoria
Garantizar cumplimiento	Funcionarios con responsabilidades de supervisión	Funcionarios con responsabilidades de supervisión	No aplica	Funcionarios con responsabilidades de supervisión y personas asignados en la dependencia
Mantenimiento	función seguridad informática	función seguridad informática e ingenieros con conocimientos en el área	función seguridad informática e ingenieros con conocimientos en el área	Dependencia que los propone
Retiro	función seguridad informática	función seguridad informática e ingenieros con conocimientos en el área	función seguridad informática e ingenieros con conocimientos en el área	Dependencia que los propone

Tabla A.1 Modelo de responsabilidad por etapa para cada tipo de política

En la tabla A.1 se proporciona una orientación para asignar responsabilidades a cada etapa de desarrollo de una política de acuerdo al nivel del requerimiento. En general, este modelo propone que la responsabilidad para las etapas relacionadas con las

POLÍTICAS, ESTÁNDARES, MEJORES PRÁCTICAS Y GUÍAS sean similares en muchos aspectos. Al existir una dependencia encargada de la gestión del programa de seguridad informática debe servir como proponente para la mayoría de POLÍTICAS, ESTÁNDARES, MEJORES PRÁCTICAS Y GUÍAS relacionadas con la seguridad de los recursos de información de la universidad –en colaboración con los profesionales que tengan conocimientos en el área técnica específica-. Dentro de sus posibilidades, la función de seguridad informática debe realizar las etapas de creación, concienciación, mantenimiento y retiro para las políticas de seguridad de cada nivel.

Sin embargo, hay excepciones a este principio general. Por ejemplo, aun cuando tiene un impacto importante sobre la seguridad informática, es más eficiente que la dirección de personal sea quien proponga las políticas y los estándares, relacionadas con seguridad informática, para encontrar nuevos empleados. Las responsabilidades para las etapas relacionadas con el desarrollo de PROCEDIMIENTOS de seguridad son diferentes de las propuestas para las POLÍTICAS, ESTÁNDARES, MEJORES PRÁCTICAS Y GUÍAS. El cuadro anterior muestra que los proponentes para los PROCEDIMIENTOS están por fuera de la función seguridad informática (un enfoque descentralizado), basados en la aplicabilidad limitada de dichos procedimientos a cierta dependencia. Aunque los PROCEDIMIENTOS se crean e implementan de manera descentralizada (en varias etapas), estos deben ser consistentes con las políticas de seguridad de mayor nivel; por tanto deben ser revisados por la función seguridad informática de la organización al igual que por el funcionario superior de la dependencia. Adicionalmente, las funciones de seguridad y auditoría deben ofrecer retroalimentación al proponente sobre el cumplimiento de los PROCEDIMIENTOS cuando se estén conduciendo revisiones y auditorías.

La asignación de responsabilidades mostrada en el cuadro anterior se entiende mejor si se explora el modelo propuesto de acuerdo con las etapas del ciclo de vida:

- **Creación.** En la mayoría de las organizaciones la función seguridad informática debe servir como proponente de todas las políticas relacionadas con seguridad que engloban toda la organización y debe ser la responsable para crear estas POLÍTICAS, ESTÁNDARES, MEJORES PRÁCTICAS Y GUÍAS. Con el fin de garantizar la pertenencia de las políticas, es recomendable que en la universidad estas políticas sean elaboradas en conjunto con los profesionales conocedores del área técnica específica. Sin embargo, las actividades necesarias, para implementar GUÍAS y requerimientos de alto nivel deben ser realizadas por cada dependencia proponente para la cual los PROCEDIMIENTOS aplicaran ya que son específicos a la estructura y a la operación de la dependencia específica.
- **Revisión.** El establecimiento de un comité de evaluación de políticas proporciona un foro de amplio espectro para revisar y evaluar la viabilidad de POLÍTICAS, ESTÁNDARES, MEJORES PRÁCTICAS Y GUÍAS que afectan a toda la organización. Aquí se propone que esta labor sea realizada por los comités de informática, que en principio están conformados por personas de diversas áreas organizacionales, interesadas en la seguridad informática. La responsabilidad del comité de evaluación es garantizar que las POLÍTICAS, ESTÁNDARES,

MEJORES PRÁCTICAS Y GUÍAS estén bien redactadas, sean comprensibles, estén coordinadas y sean viables en términos de las personas, procesos y tecnologías que afecta. Debido al volumen y el número de dependencias involucradas, es muy probable que un comité central de evaluación de políticas no pueda revisar todos los PROCEDIMIENTOS desarrollados por todas las dependencias proponentes. Sin embargo, los PROCEDIMIENTOS requieren una revisión similar y el proponente debe buscar un igual que los revise o diseñar un proceso de revisión por otras dependencias o, en último caso, solicitar una revisión por la función seguridad informática.

- **Aprobación.** La diferencia más importante entre las responsabilidades con las POLÍTICAS, con los ESTÁNDARES, con las MEJORES PRÁCTICAS o con las GUÍAS, es el nivel de aprobación requerido para cada uno y el alcance de su implementación. Las POLÍTICAS de seguridad que afectan toda la organización deben ser firmadas por el rector general (o el vicerrector general) para garantizar el nivel necesario de énfasis y visibilidad a estas (quizá el tipo más importante de políticas). Ya que los ESTÁNDARES, las MEJORES PRÁCTICAS y las GUÍAS son diseñadas para cumplir una política específica, estos deben ser aprobados con la firma de un directivo subordinado del rector general (o el vicerrector general), quien tendrá la responsabilidad de implementar la política. El director de informática, normalmente, será el responsable de aprobar este tipo de políticas. Igualmente, los PROCEDIMIENTOS de seguridad deben ser aprobados por la directiva que tiene la responsabilidad administrativa directa de la dependencia para la cual aplican dichos procedimientos.
- **Comunicación.** Ya que la secretaria (general o de sede) o UNIMEDIOS cuentan con la infraestructura y la experiencia, deberían asumir la responsabilidad de la etapa de comunicación de las políticas que aplican a toda la universidad. Cuando sea una política que no cubra toda la universidad, el proponente debe asumir la responsabilidad de comunicar los procedimientos de seguridad, pero hasta donde sea posible debe buscar el apoyo de la secretaria o de UNIMEDIOS.
- **Cumplimiento.** Los mandos medios y empleados para quienes la política de seguridad son aplicables son los principales jugadores en la implementación y garantía inicial del cumplimiento de políticas que hayan sido publicadas recientemente. En caso de las POLÍTICAS, ESTÁNDARES, MEJORES PRÁCTICAS Y GUÍAS que afectan a toda la universidad, esta responsabilidad se extiende a todos los funcionarios con responsabilidades de supervisión, empleados, docentes y estudiantes a quien aplique. En relación con los PROCEDIMIENTOS de seguridad, esta responsabilidad estará limitada a los jefes y a los empleados de la dependencia donde apliquen los procedimientos.
- **Excepciones.** En todos los niveles de una organización, habrá situaciones potenciales que impedirán el cumplimiento total de una política. Es importante que el proponente de la política o individuo o grupo con una autoridad igual o superior revise las excepciones. El comité de evaluación de políticas puede ser efectivo en investigar las solicitudes de excepciones recibidas de las dependencias que no puedan cumplir con POLÍTICAS, ESTÁNDARES Y MEJORES PRÁCTICAS. Ya que las GUÍAS son, por definición, recomendaciones

o sugerencias y no son obligatorias, solicitudes formales de excepción en su aplicación no son necesarias (aunque es recomendable que existan argumentos documentados y aprobados para no seguirlas). En el caso de los PROCEDIMIENTOS de seguridad, el directivo que aprobó el procedimiento debe también servir como autoridad para aprobar las excepciones relacionadas.

- **Concienciación.** Para la mayoría de las organizaciones, la función de seguridad informática esta idealmente ubicada para administrar la etapa de concienciación en seguridad y debe por tanto tener la responsabilidad de esta etapa en el caso de las POLÍTICAS, ESTÁNDARES, MEJORES PRÁCTICAS Y GUÍAS que afectan a toda la universidad.

Sin embargo, el equipo de seguridad informática debe realizar esta etapa en coordinación con el departamento de capacitación de la organización para garantizar unidad en el esfuerzo y en óptimo uso de los recursos. El directivo o jefe de dependencia proponente de los PROCEDIMIENTOS debe responsabilizarse para concienciar los empleados de los procedimientos de seguridad que están a su cargo. Dentro de lo posible. Esto debe ser realizado con el consejo y la asistencia de la función de seguridad informática.

- **Monitoreo.** La responsabilidad para monitorear el cumplimiento de las POLÍTICAS, ESTÁNDARES, MEJORES PRÁCTICAS Y GUÍAS que son aplicables a toda la organización es compartida entre los docentes, estudiantes, empleados, directivos, decanos, jefes de dependencia (oficina, división o sección), la función de auditoría (control interno) y la función de seguridad informática.

Cada empleado que esta sujeto a los requerimientos de seguridad debe ayudar en el monitoreo del cumplimiento reportando las desviaciones que observe. Aunque no deberían estar involucrados en la garantía de cumplimiento de las políticas, la función seguridad informática y la función auditoría (control interno) pueden jugar un importante papel en el cumplimiento del monitoreo. Incluye el cumplimiento del monitoreo de PROCEDIMIENTOS propios de dependencias mediante reportes de las contravenciones dirigidos al proponente con el fin de que se adelante la acción apropiada.

- **Garantía de cumplimiento.** La responsabilidad de garantizar el cumplimiento de los requerimientos de seguridad esta en los funcionarios con responsabilidades de supervisión sobre los docentes, estudiantes y empleados afectados por la política. Por supuesto, esto no aplica para las GUÍAS, que por diseño no son obligatorias. Los jefes responsables de las dependencias en las cuales aplican los PROCEDIMIENTOS de seguridad son los garantes de su cumplimiento. La regla general es que cada persona que tenga autoridad para supervisar otras personas debe ser el funcionario que garantice el cumplimiento de la política de seguridad. Por tanto, en ningún caso la función de seguridad informática ni la función de auditoría debe asignársele autoridad “en lugar de” o “en adición a” el jefe.

Aunque la función de seguridad informática no debe estar involucrada directamente en las acciones de garantía de cumplimiento, es importante que

este enterada, para reportar las acciones correctivas de tal forma que esta información pueda ser integrada en los esfuerzos de la etapa de concienciación.

- **Mantenimiento.** Debido a su responsabilidad en el programa de seguridad informática de la organización, la función seguridad informática es la que mejor esta posicionada para dar mantenimiento a las POLÍTICAS, ESTÁNDARES, MEJORES PRÁCTICAS Y GUÍAS que tengan aplicabilidad en toda la organización para garantizar que estén actualizadas y disponibles a todos los afectados. En los niveles inferiores de la organización, la dependencia proponente, como dueña de los PROCEDIMIENTOS de seguridad, debe realizar el mantenimiento de los procedimientos que ellos desarrollaron.
- **Retiro.** Cuando una POLÍTICA, ESTÁNDAR, MEJOR PRÁCTICA o GUÍA no se necesita más, debe ser retirada. El proponente del requerimiento debe tener la responsabilidad de retirarlo. Normalmente el equipo de seguridad informática realizara esta función con las políticas de seguridad que afectan toda la organización, en tanto la dependencia que es la dueña de los PROCEDIMIENTOS de seguridad debe tener la responsabilidad de retirar el procedimiento.

APÉNDICE B

GLOSARIO DE TÉRMINOS

GLOSARIO DE TERMINOS

Adaptive Server	<i>El servidor en la arquitectura cliente/servidor de Sybase (llamado SQL Server en versiones anteriores al 11.5) Adaptive Server maneja múltiples bases de datos y múltiples usuarios, guarda la ruta de la ubicación actual de los datos en el disco, conserva mapas de la descripción lógica de los datos para almacenamiento físico, y mantiene los datos y procedimientos en la memoria cache.</i>
Adaptive Server login	<i>Es el nombre que un usuario utiliza para entrar al Adaptive Server. Un login es valido si Adaptive Server tienen una entrada para el usuario en la tabla del sistema syslogins.</i>
Administrador	<i>La persona que supervisa y controla el correcto funcionamiento de un sistema informático.</i>
Alfanumérico	<i>Conjunto de letras, números y otros símbolos, como signos de puntuación o símbolos matemáticos. Hace referencia a los caracteres del teclado y al conjunto de caracteres disponibles para las diferentes operaciones de transferencia de datos del ordenador.</i>
Algoritmo	<i>Descripción exacta de la secuencia en que se ha de realizar un conjunto de actividades tendientes a resolver un determinado tipo de problema o procedimiento.</i>
Alias	<i>Un pseudónimo que permite a un usuario del Adaptive Server y poder ser conocido en una base de datos como otro usuario.</i>
Antivirus	<i>Aplicación informática encargada de detectar y eliminar virus.</i>
Archivo	<i>Colección de registros almacenados siguiendo una estructura homogénea.</i>
Atributo	<i>Puede ser definido como una función que transforma un conjunto de entidades o relaciones.</i>
BackUp	<i>Copia de seguridad. Una copia de una base de datos o log de transacciones, usado para recuperarse de una falla</i>
Base de datos	<i>Es un conjunto, colección o depósito de datos almacenados en un soporte informático de acceso directo.</i>
BD	<i>Por sus siglas en español Base de Datos o en inglés (DB) Data Base.</i>

Binario	<i>Sistema de numeración en base 2, de modo que sólo hay dos dígitos posibles: el 0 y el 1. Para formar números "grandes", se usan varios dígitos binarios, que representan cada una de las potencias de 2. Por ejemplo, el número decimal 13 se representa 1101 (1x8 + 1x4 + 0x2 + 1x1).</i>
BIOS	<i>Sistema de entrada/salida básica (Basic Input Output System). Suele tratarse de uno o varios chips de memoria ROM (habitualmente EPROMs) que contienen las rutinas básicas de entrada y salida, los primeros pasos que debe dar un ordenador al encenderse, la configuración básica del sistema, etc.</i>
Bit	<i>Unidad mínima de información que puede ser transmitida o tratada. Procede del inglés, Binary Digit o Dígito Binario, y puede tener un valor de 0 (cero) ó 1 (uno).</i>
Bps	<i>Bits por segundo: es la unidad en que se mide la velocidad de transferencia efectiva de un módem o de una conexión serie.</i>
BSI	<i>Instituto Británico de Estándares, British Standar Institute por sus siglas en inglés.</i>
Bug	<i>Error en un programa, que hace que en ciertas circunstancias pueda no comportarse correctamente.</i>
Byte	<i>Es la unidad básica de información. En la práctica, se puede considerar que un byte es la cantidad de espacio necesaria para almacenar una letra. Tiene múltiplos como el Kilobyte, Megabyte, Gigabyte y Terabyte. Internamente, corresponde a 8 bits. En español, a veces se le llama octeto.</i>
Cache	<i>Es un tipo de memoria especial, más rápida que la RAM normal (y más cara), que se pone en el camino de los datos que van del procesador a la memoria RAM. Así, toda información que va de la RAM al procesador se deja almacenada temporalmente en la memoria caché. A la hora de volver a leer información, se comprueba primero si está en la memoria caché; si se encuentra allí, no hace falta ir a la RAM a buscar. En la práctica, es muy frecuente realizar operaciones repetitivas o trabajar con datos repetitivos, lo que hace que poseer memoria caché ayude a acelerar el funcionamiento normal del ordenador. Hoy en día es frecuente distinguir también la memoria caché de "primer nivel" y de "segundo nivel". La de primer nivel se encuentra dentro del propio procesador. La de segundo nivel se</i>

	<p><i>encuentra en la placa base, es de mayor tamaño y algo más lenta.</i></p>
Campo	<p><i>Es la unidad más pequeña a la cual uno puede referirse en un programa. Desde el punto de vista del programador representa una característica de un individuo u objeto. Una o más categorías de datos de una base de datos; una columna en una tabla.</i></p>
CCSC - DTI.	<p><i>Centro Comercial de Seguridad en Cómputo del Departamento Británico de Comercio e Industria.</i></p>
CD	<p><i>Normalmente se refiere a Compact Disc. Otras abreviaturas relacionadas son: CD-ROM, CD-R y CD-RW. Un Compact Disc para ordenador será capaz de almacenar cerca de 650 Mb de información.</i></p>
CD-ROM	<p><i>Acrónimo de Compact Disk-Read Only Memory, Disco compacto, memoria de solo lectura. El dispositivo más común de almacenamiento óptico, donde un láser lee superficies y hoyos de la superficie de un disco; puede almacenar hasta 640 MB pero no se puede escribir en él; Actualmente han salido nuevos que ya permiten la lectura y escritura pero requieren de una unidad especial de CD ROM llamada CD Writer, CD-WR</i></p>
Clase	<p><i>En el código fuente para un programa orientado a objetos, serie de declaraciones que definen un objeto.</i></p>
Clave	<p><i>En el mundo de las bases de datos, se conoce como clave (en inglés Key) al valor de es capaz de distinguir un registro de otro de forma fiable, como podría ser el DNI o el Pasaporte para el caso de una persona (el nombre no sería una clave correcta, ya que sí puede repetirse).</i></p>
Cliente de una BD	<p><i>Programa ejecutándose en una computadora en red que recibe datos mediante el acceso a un servidor de datos.</i></p>
Cluster	<p><i>Un disco duro está dividido en "páginas" en las que podemos guardar información. Cada fichero puede ocupar una o más páginas, pero cada página sólo puede estar ocupada por un único fichero. El tamaño de esas páginas (que son los clusters) influirá en la cantidad de espacio desperdiciado en el disco: si hay páginas de tamaño grande, y tenemos almacenados muchos ficheros de pequeño tamaño, se desperdiciará mucho espacio, con clusters pequeños, se desperdicia menos espacio al almacenar ficheros, pero el índice de contenido del disco ocupará más espacio, ya que hay mayor cantidad de clusters que controlar. En algunos</i></p>

Comando	<i>sistemas Unix, el tamaño de cluster puede ser fijado al instalar el sistema (por ejemplo, eligiendo entre 1, 2 o 4K). Una instrucción que especifica una operación para ser transformada por la computadora. Cada comando o declaración de SQL comienza con una clave, como un insert, es decir el nombre de operación básica a ejecutar. Muchos comandos tienen una o mas frases de palabras claves, o cláusulas, eso hace el comando para encontrar una necesidad particular.</i>
Copia de Respaldo	<i>Copia de un programa o archivo de datos, generalmente en disquete o cinta magnética, que se mantiene guardado en caso que el original se dañe.</i>
Cracker	<i>Persona que intenta romper las protecciones de un cierto sistema informático, normalmente con fines maliciosos (distinto de un "hacker", que procura profundizar en un cierto sistema para aprender de él).</i>
Dato	<i>Conjunto de caracteres con algún significado, pueden ser numéricos, alfabéticos, o alfanuméricos.</i>
DBA	<i>Administrador de bases de datos, DataBase Administrador por sus siglas en inglés. Es la persona o equipo de personas profesionales responsables del control y manejo del sistema de base de datos, generalmente tiene (n) experiencia en DBMS, diseño de bases de datos, sistemas operativos, comunicación de datos, hardware y programación.</i>
Dbase	<i>Gestor de bases de datos, realizado por Ashton Tate y posteriormente adquirido y mejorado por Borland.</i>
DBMS	<i>Database management system por sus siglas en inglés o Sistema Manejador de Base de datos SMBD.</i>
dbo	<i>Dueño de la base de datos. Database Owner. El creador de una base de datos. El dueño de la base de datos tiene control sobre todos los objetos de la base de datos en esa base de datos. El nombre del login para el dueño de la base de datos es "dbo".</i>
DD	<i>Diccionario de Datos. Un diccionario de datos es un archivo que contiene metadatos; es decir, datos acerca de los datos</i>
DDL	<i>Data Definition Language por sus siglas en inglés o Lenguaje de Definición de Datos LDD, permite definir la estructura lógica (o esquema) de la bases de datos.</i>

Default	<i>La opción de seleccionar por el sistema cuando otra opción no es especificada.</i>
Device	<i>Cualquier pieza de disco (como una partición) o un archivo en el archivo de sistema usado para almacenar bases de datos y sus objetos.</i>
DML	<i>Lenguaje de manipulación de datos, por sus siglas en inglés Data Manipulation Language.</i>
DoS	<i>Denegación de Servicios, por sus siglas en inglés Denial-of-Service</i>
DOS	<i>Sistema operativo de disco (Disk Operating System). Se trata de un sistema operativo monousuario y monotarea. Hay diversas versiones, con distintos nombres según la casa que lo desarrolle: MsDos (Microsoft), DrDos (Digital Research), PcDos (IBM), Novell Dos (Novell), etc.</i>
Dump	<i>La acción de hacer un backup de una entidad de base de datos, incluyendo los datos y el log de transacciones, el cual va con el comando dump de la base de datos. También, los datos que resultan de esa acción.</i>
E/S	<i>Entrada/salida (se suelen usar las siglas en inglés I/O).</i>
Esquema	<i>Es la descripción lógica de la base de datos, proporciona los nombres de las entidades y sus atributos especificando las relaciones que existen entre ellos.</i>
Firewire	<i>Nombre comercial de IEEE 1394, un nuevo tipo de conexión digital, que permite velocidades a partir de 100 Mbps (millones de bits por segundo), y se pretende que llegue hasta los 400 MBps (millones de bytes por segundo).</i>
FRAP	<i>Análisis de Riesgo de Forma Facilitada, por sus siglas en inglés Facilitated Risk Analysis Process</i>
Group	<i>Grupo es un únicamente un conjunto de nombres de usuarios asignado a un conjunto de permisos para los objetos y operaciones dentro de una base de datos.</i>
Guest	<i>Invitado es el nombre de un usuario en la tabla sysusers de la base de datos model, el cual habilita un usuario con un login valido del Adaptive Server para usar bases de datos creadas por model, con privilegios limitados.</i>
Hacker	<i>Entusiasta de la informática. La palabra se suele usar para indicar también un cierto intrusismo: un hacker es una</i>

	<p><i>persona que siempre está deseando aprender y superar nuevos retos, entre los que se pueden encontrar el acceder a un cierto sistema teóricamente cerrado. Pero esto no quiere decir que se haga con malicia, sino por el propio reto en sí. Cuando se trata de alguien con intenciones maliciosas se suele emplear la palabra "cracker".</i></p>
Index	<p><i>Índice es un objeto de una base de datos que consiste de valores claves de las tablas de datos y apuntan a las páginas que contienen esos valores. Los índices aceleran el acceso a renglones de datos por apuntadores del Adaptive Server para la ubicación de una columna de la tabla de datos en un disco.</i></p>
Hardware	<p><i>Todos los componentes electrónicos, eléctricos y mecánicos que componen una computadora, en oposición a los programas que se escriben para ella y la controlan (software).</i></p>
Instancia	<p><i>Al estado que presenta una base de datos en un tiempo dado.</i></p>
ISO/IEC 17799	<p><i>Es el único estándar de alto nivel y de naturaleza conceptual dedicado al manejo de la seguridad de la información en un campo manejado por Principios y Buenas Prácticas.</i></p>
ISRM	<p><i>Administrador de Riesgo de Seguridad de la Información por sus siglas en inglés Information Security Risk Management.</i></p>
IT	<p><i>Information Technologies por sus siglas en inglés o en español (TI) Tecnología de la Información.</i></p>
Lock	<p><i>Un mecanismo de control concurrente que protege la integridad de los datos y los resultados de transacción en un ambiente de multiusuarios. Adaptive Server asigna bloqueos a página o tabla para prevenir que dos usuarios intenten cambiar el mismo dato al el mismo tiempo, y prevenir procesos que están seleccionando datos desde la lectura, que están en proceso de ser cambiados.</i></p>
Login	<p><i>El nombre de un usuario para entrar a al Adaptive Server. Un login es valido si el Adaptive Server tiene estas entrada para estos usuarios en la tabla del sistema syslogins.</i></p>
Master database	<p><i>Es la base de datos del sistema que controla los usuarios de la base de datos y las operaciones del Adaptive Server en conjunto.</i></p>

Metadata	<i>Es un dato acerca de otro dato. Metadata esta almacenada en la tabla de proxy local por el Component Integration Services. Este almacena los matadatos representado por esquemas con información acerca de las tablas remotas.</i>
Mirror device	<i>Un duplicado del dispositivo de base de datos del Adaptive Server. Todo lo escrito en el dispositivo primario es copiado (espejado) en el segundo dispositivo físico. Lo escrito puede ser en todo caso serial (consecutivo) o paralelo (simultáneamente). Si un dispositivo falla, el otro contiene una copia actual de todas la transacciones.</i>
Modelo E-R	<i>Modelo entidad relación, esta formado por objetos básicos llamados entidades y las relaciones entre estos objetos así como las características de estos objetos llamados atributos.</i>
NCC	<i>Centro Nacional de Cómputo</i>
ODBC	<i>Open database connectivity. Abrir la conectividad de la base de datos. La interfase ODBC, definida por la corporación Microsoft, es una interfase estándar para el sistema administrador de la base de datos en el ambiente Windows y Windows NT.</i>
Parameter	<i>Parámetro es un argumento para un procedimiento almacenado un procedimiento del sistema.</i>
Password	<i>Clave de acceso o contraseña necesario para acceder a un determinado sistema.</i>
PD 0003	<i>Código de prácticas para el manejo de seguridad de la información.</i>
Permission	<i>Permiso es la autorización para realizar ciertas acciones sobre ciertos objetos de la base de datos o para correr ciertos comandos.</i>
Process	<i>Un programa en ambiente de ejecución hacia el CPU físico a través del Sistema Operativo.</i>
Procedure	<i>Una colección de declaraciones de SQL y declaraciones de control de flujo opcionales almacenadas bajo un nombre. Adaptive Server suministra procedimientos que son llamados procedimientos del sistema.</i>
Protocolo	<i>Normas a seguir en una cierta comunicación: formato de los datos que debe enviar el emisor, cómo debe ser cada una de las respuestas del receptor, etc.</i>

Public	<i>Son todos los usuarios registrados en una base de datos que son miembros de un grupo "public". Los usuarios en este nivel de autoridad pueden crear una tabla temporal y tener acceso a objetos cuyos dueños tienen otorgados permisos para "publico".</i>
Query	<i>Consulta es una declaración de SQL o un grupo de declaraciones de SQL que acceda y/o manipule datos en una base de datos.</i>
RAM	<i>Memoria de acceso directo (Random Access Memory). Normalmente se usa este nombre para referirse a memorias en las que se puede leer y también escribir (RWM). En los últimos PC es habitual que se use Fast Page Ram (386 y anteriores), EDO Ram (486 y Pentium) y SDRAM (últimos Pentium, Pentium MMX y superiores).</i>
Red	<i>En informática, interconexión de computadoras mediante cables u ondas radiales o telefónicas.</i>
Registro	<i>En el mundo de las bases de datos, cada una de las fichas que componen una tabla. Colección de campos de iguales o de diferentes tipos.</i>
Roles	<i>Los títulos reconocidos por Adaptive Server que provee responsabilidades individuales para usuarios capacitados en sistemas de administración y seguridad relacionados a las tareas en el Adaptive Swrver. Los roles del Administrador del sistema, Oficial de seguridad del sistema, y el Operador pueden ser otorgados para cuentas de entrada individuales al servidor. Además de estos roles del sistema, un Oficial de Seguridad del Sistema puede crear roles de usuario definidos, tales como "analista financiero" y "administrador de salarios".</i>
ROM	<i>Memoria sólo de lectura (Read Only Memory).</i>
Row	<i>Un conjunto de columnas relacionadas que describen una entidad específica. También llamada registro.</i>
Seguridad	<i>Mecanismos de control que evitan el uso no autorizado de recursos.</i>
Segment	<i>Un llamado subconjunto del dispositivo de la base de datos disponible para un base de datos particular. Este es un nivel de punto para uno o más dispositivos de bases de datos. Los segmentos pueden ser usados para controlar la colocación</i>

	<i>de las tablas en índices sobre un dispositivo específico de la base de datos.</i>
Señal	<i>Cambio de estado orientado a eventos (p. ej. un tono, cambio de frecuencia, valor binario, alarma, mensaje, etc.).</i>
Server user ID	<i>El número de identificación por el cual un usuario es conocido por el Adaptive Server.</i>
SQL	<i>Structured Query Language. Es el lenguaje usado para comunicarse con una base de datos relacional y que esta sujeto a un conjunto de estándares por varios cuerpos estándares.</i>
SSA	<i>Secretaria de Servicios Académicos, perteneciente a la UNAM.</i>
Tabla	<i>En el mundo de las bases de datos, un conjunto de registros (fichas) que tienen una cierta homogeneidad (por ejemplo, los datos de nuestros proveedores podrían estar almacenados en una misma tabla).</i>
TCP/IP	<i>Protocolo de control de transmisiones/Protocolo Internet. Es el protocolo estándar de comunicaciones en red utilizado para conectar sistemas informáticos a través de Internet.</i>
TIC	<i>Tecnología informática de comunicación.</i>
Tiempo real	<i>Rápida transmisión y proceso de datos orientados a eventos y transacciones a medida que se producen, en contraposición a almacenarse y retransmitirse o procesarse por lotes.</i>
TRA	<i>Treath and Risk Assessment por sus siglas en ingles</i>
Transaction	<i>Un grupo de declaraciones de Transact-SQL que es tratada como una simple unidad de trabajo. En todo caso estas declaraciones en el grupo son ejecutadas o no son declaraciones ejecutadas.</i>
Transact-SQL	<i>Es el lenguaje usado en el Sybase Adaptive Server.</i>
Trigger	<i>Una forma especial de procesos almacenados (stored procedure) que va dentro del efecto cuando un usuario da un comando de cambio tales como insert, delete, o update para una columna o tabla especificada. Los triggers son frecuentemente usados para reforzar la integridad referencial.</i>
UNAM	<i>Universidad Nacional Autónoma de México.</i>

UNIX	<i>Un sistema operativo diseñado para ser usado por un grupo de varias personal al mismo tiempo (multi-usuario) que maneja TPC/IP. Es el sistema operativo más común en los servidores Internet.</i>
URL	<i>Dirección de una cierta página de información dentro de Internet (Universal Resources Locator).</i>
USECAD	<i>Unidad de Servicio de Cómputo Académico, de la UNAM.</i>
View	<i>Una vista es una selección llamada declaración que es almacenada en la base de datos como un objeto. Este permite a los usuarios ver un subconjunto de registros o columnas de una o más tablas.</i>
Virus	<i>Un programa con intenciones malignas, que es capaz de propagarse de un fichero a otro del ordenador.</i>
Web site	<i>Dirección de un servidor de Web.</i>
Web	<i>ver WWW (literalmente: telaraña).</i>
WWW	<i>World Wide Web. Sistema de Internet para vincular mediante hipertexto en todo el mundo documentos multimedia, permitiendo un fácil acceso, totalmente independiente de la ubicación física, a la información común entre documentos.</i>

BIBLIOGRAFÍA

LIBROS

- De Miguel, Adoración y Piattini, Mario Gerardo, **Concepción y Diseño de Base de Datos**, Segunda Edición, Addison-Wesley Iberoamericana, México 1999.
- Ullman, Jeffrey D. y Widon, Jennifer, **Introducción a los Sistemas de Base de Datos**, Prentice-Hall Hispanoamericana, S.A., México 1999.
- Tsai, Alice, **Sistema de Base de Datos: Administración y Uso**, Prentice-Hall Hispanoamericana, S.A., México 1990.
- Batini, Carlo et al., **Diseño Conceptual de Base de Datos**, Addison-Wesley Iberoamericana, México 1994.
- Elmasri, Ramez y Navathe, Shamkant, **Sistemas de Bases de Datos: Conceptos Fundamentales**, Segunda Edición, Addison-Wesley Iberoamericana, Wilmington, Delaware, EUA 1997.
- Hansen, Gary y Hansen, James, **Diseño y Administración de Base de Datos**, Tercera Edición, Prentice-Hall Hispanoamericana, México 2000.
- Martín, James, **Organización de las Bases de Datos**, Prentice-Hall Hispanoamericana, México 1977.
- Silberschatz, Abraham et al., **Fundamentos de Bases de Datos**, Tercera Edición, McGraw-Hill, España 1998.
- Date, C. J., **Introducción a los Sistemas de Bases de Datos**, Séptima Edición, Pearson Educación de México, S.A. de C.V., México 2001.
- Maiwald, Eric, **Fundamentos de Seguridad de Redes**, Segunda Edición, McGraw-Hill, México 2005.
- Map, Magerit, **Metodología de análisis y Gestión de Riesgos de los Sistemas de Información**. BOE, 1997.
- Molina, J. M. **Seguridad, información y poder**. Incipit, 1994
- Morant, J. L. y otros. **Seguridad y Protección de la Información**. Cera, 2001.
- Ribagorda, A. **Glosario de términos de Seguridad de las T.I.** Coda, 1997.

TESIS

- Lazarini Castañeda, Ulises, **Seguridad y reglas de integridad en bases de datos relacionales**. Mexico 1997

INTERNET

- Trejo Martinez Janhil Aurora, Universidad Autónoma de Nuevo León; Facultad de Contaduría Pública y Administración. Bases de datos, <http://www.monografias.com/trabajos11/basda/basda.shtml>
- Michael cabello Alvino, Base de Datos, fecha de ultima actualizacion: 2001, <http://espanol.geocities.com/michelsftpe/bdatos.htm>
- Lola Cárdenas Luque, Introducción a las bases de datos, Ultima modificación: 22 de abril de 2003, <http://rinconprog.metropoliglobal.com/CursosProg/BDatos/IntroBD/index.php?cap=2>
- Facultad de Ingenieria, Plan Institucional de Desarrollo Informatico, Pagina de la Facultad de Ingenieria, <http://www.ingenieria.unam.mx/~pidi/>
- Campoy Medrano Lourdes Arlín, Tutorial de Base de datos 1, Instituto Tecnológico de La Paz, fecha de ultima actualizacion: 30 de junio de 1999, <http://www.itlp.edu.mx/publica/tutoriales/basedat1/>
- Castro Jesús Antonio, Tutorial de Base de datos , Instituto Tecnológico de La Paz, fecha de ultima actualizacion: 30 de junio de 1999, <http://www.itlp.edu.mx/publica/tutoriales/basedat2/>
- Cisterna Neira Mario, Metodos de Optimizacion de consultas para el lenguaje SQL, Universidad de Santiago de Chile, fecha de ultima actualizacion 2002, <http://macine.epublish.cl/tesis/index.html>
- Duque Méndez Néstor Darío, Curso de Bases de Datos, Universidad Nacional de Colombia-Sede Manizales, fecha de ultima actualizacion: 22-Agosto-2004 03:58P.M., <http://www.virtual.unal.edu.co/cursos/sedes/manizales/4060029/index.html>
- Marqués Andrés María Mercedes, Apuntes de ficheros y bases de datos, Universitat Jaume I, <http://www3.uji.es/~mmarques/f47/apun/>
- Sigüenza Juan Alberto, Introducción a las bases de Datos relacionales , Universidad Autonoma de Madrid, www.ii.uam.es/~siguenza/Bases%20de%20datos%20I.ppt
- Viaplus Tech, Curso de Introducción al Access, informatica x mail, fecha de ultima actualizacion: 20/10/03, <http://www.mailxmail.com/curso/informatica/access>
- Jesús M. Milán Franco, Bases de datos y sistemas de información, Universidad Computense de Madrid, fecha de ultima modificacion: 17 de septiembre del 2002, http://www.fdi.ucm.es/profesor/milanjm/bd/Practica_10.pdf
- Ro y Norick, Aprenda Bases de Datos con MS SQL SERVER 2000, fecha de : 2003, <http://usuarios.lycos.es/cursosgbd/UD4.htm>
- Ing Rubiel Navarro CH., Conceptos de Auditoria a Bases de Datos, Universidad Popular Católica Popular del Risaralda, http://www.ucpr.edu.co/auditores/Auditoria_BD.zip
- Fernandez Medina Eduardo, Seguridad en el Diseño de Bases de Datos y Sistemas de información, Universidad de Castilla-La Mancha, fecha de ultima actualizacion: 010205, <http://alarcos.inf-cr.uclm.es/doc/calidadSI/Tema4.pdf>

- Federico García Crespí., Marco A. Marhuenda García, seguridad en sistemas de información, Universidad Miguel Hernández de Elche, fecha de última actualización: 30 Mayo 2004, <http://ulises.umh.es/cc/personal/marco/ssi/default.html>
- Esther de Ves Cuenca, Bases de Datos II, Universidad de Valencia, fecha de última actualización: , <http://informatica.uv.es/iiguia/2000/BD2/BD2Tema6.pdf>
- Ingeniera Pilar Prados, Seguridad e integridad de bases de datos, Universidad Popular Católica Popular del Risaralda, Fecha de actualización:, <http://www.ucpr.edu.co/auditores/basesdatos/integridadYSeguridadBD.ppt>
- Raga Charlis, Base de Datos, Monografías, <http://www.monografias.com/trabajos7/bada/bada.shtml>
- Mtro. Tomás Rodríguez Gómez, Bases de Datos 1, Universidad de Guadalajara, Fecha de última actualización: 06 de enero de 2003, <http://academicos.cualtos.udg.mx/Informatica/Ceneval2003/Bases%20de%20Datos1.htm>
- Román Medina-Heigl Hernández, Análisis de seguridad, optimización y mejora de un portal web basado en PHP y MySQL, Universidad de Sevilla, fecha de última actualización: Diciembre de 2002, <http://www.rs-labs.com/papers/PFC.pdf>

