MECANISMOS DE SEGURIDAD E INTEGRIDAD EN UN SISTEMA DE BASES DE DATOS

ALONSO TAMAYO ALZATE * NÉSTOR DARÍO DUQUE MÉNDEZ*

PC: criptografía, auditoría, perfil de usuario

RESUMEN

El masivo incremento en el uso de las computadoras y el desarrollo de aplicaciones cada vez más sofisticadas han creado la necesidad de aplicar técnicas de seguridad a las bases de datos para poder hacer frente a esos cambios y priorizar un enfoque preventivo e intentando actuar antes o durante el hecho, asegurando que los activos de la compañía permanezcan protegidos y que se han establecido los controles internos adecuados para salvaguardar los recursos informáticos apropiadamente.

ABSTRACT

The massive increment in the use of the computers and the development of more and more sophisticated applications has created the necessity to apply technical of security to the databases to be able to make in front of those changes and to prioritize a preventive focus and trying to act before or during the fact, assuring that the assets of the company remain protected and that the appropriate internal controls have settled down to safeguard the computer resources appropiately.

Introducción

El presente artículo tiene como uno de sus objetivos primordiales mostrar de forma sucinta el potencial que ofrecen los Sistemas de Gestión de Bases Datos (SGBD o DBMS; Data Base Manager System), en favor de la seguridad y consistencia de la información.

* Profesores Universidad Nacional de Colombia. Sede Manizales.

Se hará referencia a los Sistemas de Bases de Datos conforme al modelo relacional, el cual fue expuesto por el Dr. Codd en la primera mitad de los años 70. Los DBMS relacionales cuentan con el mayor número de instalaciones comerciales en el mundo de hoy.

Codd definió tres características del modelo relacional (traducción textual):

1. Son estructuras de datos simples. Consiste en tablas de dos dimensiones donde los elementos son ítems de datos. Esto permite un alto grado de independencia de la representación física de los datos.

2. El modelo relacional provee una sólida fundamentación para la consistencia de los datos. El diseño de las bases de datos es asistido por los procesos de normalización que elimina las anomalías en los datos. Adicionalmente, los estados de consistencia de las bases de datos pueden ser uniformemente definidos y mantenidos a través de reglas de integridad.

3. El modelo relacional permite la manipulación de las relaciones. Esta característica puede ser encargada a potentes lenguajes no procedimentales basados en la teoría (álgebra relacional) o en la lógica (cálculo relacional).

CUENTA

CUENTA	SALDO
C-104	345.876
D-055	567.000
L-188	98.078
P-205	76.453

CLIENTE

NOMBRE	CEDULA	CUENTA	CIUDAD
Cano Juan P	7.489.807	C-104	Cali
Pérez José	1.048.987	P-205	Manizales
López Sandra	10.128.999	L-188	Cartago
Duque Laura	42.879.003	D-055	Pereira

Figura 1. Modelo relacional

El estándar más sobresaliente, en cuanto a herramientas de organización y gestión de información en Bases de Datos es **SQL** (Structured Query Language), Lenguaje Estructurado de Consultas. A pesar que se mantiene este nombre SQL, es mucho más que un lenguaje de consultas, es realmente una potente herramienta para interactuar con el DBMS y permite controlar las funciones que suministra éste a los usuarios, incluyendo definición de la estructura y sus relaciones, recuperación y manipulación de datos, controles de acceso y manejo de restricciones de integridad. Tiene gran importancia en la arquitectura Cliente/Servidor de Bases de Datos, pues es el elemento que garantiza la conectividad en un mundo de diversos fabricantes y plataformas.

Las normas oficiales de SQL, son SQL-89 (SQL1), emitida inicialmente en 1986 y modificada en 1989. SQL-92 (SQL2), ratificada en 1992 y con una extensión 5 veces mayor a la original SQL1. En febrero de 1997 se disponía de un material preliminar que posteriormente se convertiría en SQL3, el cual fue ratificado en su totalidad en julio de 1998 y con una extensión de más de 1000 páginas. Dado lo nuevo de esta última norma, sólo algunas partes han sido incluidas en las implementaciones comerciales.

Seguridad en bases de datos

Los Sistemas de Gestión de Bases de Datos proveen mecanismos que garantizan la seguridad, consistencia y reglas de integridad. Estos conceptos son implementados en la práctica usando varios elementos, algunos genéricos y estándares y otros más particulares del motor de Bases de Datos usado.

Este artículo abordará estos temas, comunes a la mayoría de los SGBD y mostrará un ejemplo práctico usando para ello el DBMS InterBase de la casa productora Inprise, un potente motor de Bases de Datos tanto a nivel de servidores como de clientes.

Características que apoyan la seguridad en los Sistemas de Bases de Datos

Los tópicos que se incluyen tienen que ver con la exactitud, consistencia y confiabilidad de la información y con la privacidad y confidencialidad de los datos. Las Bases de Datos tienen dentro de sus características elementos que pueden ser utilizados para garantizar la calidad de la información almacenada y procesada.

Claves Primarias. Es el mínimo subconjunto no vacío de atributos que permiten identificar en forma unívoca una tupla dentro de la relación. Si existen varios conjuntos que cumplan ésta condición se denominan llaves candidatas y debe ser seleccionada

una de éstas como llave primaria. Los atributos que conforman la clave primaria se denominan atributos primos. Esta definición determina que para un valor llave primaria solo existirá una tupla o registro en la tabla. Esta situación garantiza que no se tendrá información repetida o discordante para un valor de clave y puede ser usada como control, para evitar la inclusión de información inconsistente en las tablas.

Dominio de los atributos. El dominio de un atributo define los valores posibles que puede tomar este atributo. Además de los dominios "naturales", usados como tipos de datos, el administrador del sistema puede generar sus propios dominios definiendo el conjunto de valores permitidos. Esta característica, usada en forma correcta, se convierte en mecanismo de control, restricción y validación desde el DBMS, de los datos a ingresar.

Reglas de Integridad. Son restricciones que definen los estados de consistencia de la Base de Datos. El mantenimiento de las restricciones de integridad es generalmente costoso en términos de recursos del sistema, pero dadas las enormes capacidades disponibles hoy, éste aspecto no es relevante. Idealmente debería ser verificado en cada actualización de la Base de Datos, para evitar que se caiga en estados de inconsistencia.

E. F. "Ted" Codd, padre del modelo relacional, planteó dos restricciones mínimas que deben ser tomadas en cuenta, estas son:

a. Regla de la Entidad. Parte del hecho que toda tabla posee una llave primaria. Esta regla dicta que ningún atributo primo puede ser nulo.

b. Regla de Integridad Referencial. Involucra dos relaciones (tablas) e impone la restricción que un grupo de atributos que en una relación es clave primaria, en otra puede ser tabla. La definición de ésta característica en la construcción de la Base de Datos, impide ingresar valores en algunos atributos de tuplas que no tengan su correspondencia en la tabla relacionada. Como ejemplos podemos mostrar algunos casos:

- Impedir incluir novedades de nómina a una persona que no exista como trabajador en el archivo maestro de empleados.
- Impedir facturar a un cliente que no esté previamente creado en el archivo de clientes.
- Impedir borrar de la lista de clientes un registro cuyo código esté incluido en la relación de cuentas por cobrar.

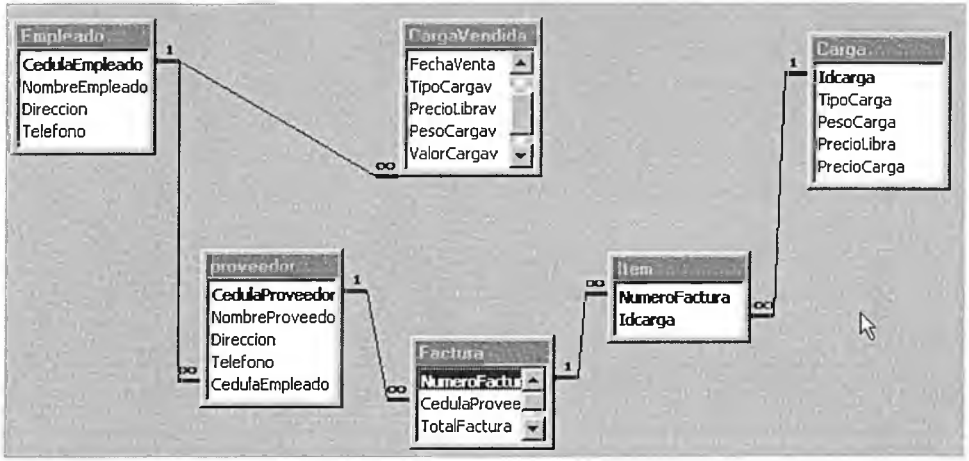

Figura 2. Definición de relaciones

Debe ser claro que éstas restricciones no son generales y deben corresponder a una situación en particular. Es cierto que si nuestra aplicación respeta los principios del negocio que se modela, y lo debe hacer, tal es el caso de segregación de funciones, la Integridad Referencial se convierte en un mecanismo automático de control para el ingreso y manipulación de datos.

Reglas de integridad del negocio. Cada negocio funciona en forma diferente y tiene reglas asociadas a su actividad que pueden ser definidas como restricciones en la Base de Datos. Esto implicaría que cualquier operación que se realice debe respetar éstas limitantes. Por decir algo, el valor mínimo para facturar a crédito es $300.000 o el descuento solo se otorga en compras superiores a $100.000. Estas son condiciones que la administración coloca a la operación y como principio en el desarrollo de una aplicación, deben ser respetadas por ésta. En el diseño de la Base de Datos no solo se respeta, sino que además se puede convertir en una regla que no puede ser violada.

Vistas. Son consultas SQL, que están almacenadas permanentemente en la base de datos y tienen asignado un nombre. Es una tabla virtual, cuyo contenido está determinado por su definición. Para el usuario del sistema la vista se presenta como una tabla real y puede ser objeto de las operaciones permitidas sobre éstas.

Sirve como mecanismo de compartimentación de la información almacenada, permitiendo presentar a diferentes usuarios parte del universo, según se considere necesario. Para el efecto las vistas pueden ser verticales u horizontales. Las verticales,

se definen generando salidas donde solo algunos de los atributos o campos del esquema de la tabla son presentados. Las vistas horizontales se logran separando las tuplas o registros seleccionados, partiendo de una condición o predicado. Según las políticas de seguridad, es usual que gran parte de los usuarios nunca tengan acceso directamente a las tablas completas, sino que lo hagan a través de las vistas, las cuales, por ser un objeto, son sujetas de otras medidas de seguridad.

Perfiles de usuario y acceso a objetos de la Base de Datos. Se refiere a sistemas donde muchos usuarios pueden tener acceso, esto obliga a que cada uno debe ser identificado en forma independiente (y esto es muy importante, para el control de acciones de los usuarios). Un usuario es creado por el administrador de la Base de Datos (DBA) y se le asigna una clave de acceso (password). También pueden ser creados roles, los cuales podrán ser concedidos a los usuarios.

Considerando los diferentes elementos como objetos de la Base de Datos (tablas, campos, procedimientos almacenados, triggers, vistas, etc.), se les puede conceder privilegios, según el caso, que permitan consultar, insertar, actualizar, borrar, ejecutar, definir integridad referencial. Un buen manejo de éstas posibilidades permite el control de las actividades de los usuarios sobre el sistema.

Este mecanismo es conocido como Control Discriminatorio, el cual posibilita que diferentes usuarios tengan privilegios de acceso a diferentes objetos de bases de datos. Todos éstos privilegios son inicialmente concedidos a los usuarios por el administrador del sistema. Quien crea el objeto es el propietario del mismo y puede ceder o compartir sus derechos sobre el mismo. Para autorizar a los usuarios la manipulación de los datos, el comando general en lenguaje SQL, es el siguiente:

GRANT operaciones ON tablas TO usuarios

Las operaciones o privilegios, según los estándares, pueden ser insertar (insert), borrar (delete), actualizar (update), consultar (select), uso de integridad referencial (references). También puede ser All Privileges (todos los privilegios). Dependiendo de la implementación aparecen algunos otros.

Las tablas podrían generalizarse a objetos de la base de datos, como vistas, procedimientos almacenados, desencadenantes, etc. Los usuarios son reconocidos por el ID-usuario, único para cada usuario en una instalación. En ocasiones es posible manejar grupos de usuarios identificados por ID-grupo. Si se desea que el comando cubra a todos los usuarios se debe utilizar Public.

Si además de lo anterior se desea otorgar el privilegio a un usuario para que a la vez ceda parte de sus privilegios a otros usuarios, se debe agregar al final de la sentencia la cláusula

With grant option

Lo anterior puede facilitar la operación y administración, pero hace difícil el control de permisos que han sido dados a los usuarios y por ende al manejo de la seguridad.

El comando ***REVOKE*** *operaciones* ***ON*** *tablas* ***FROM*** *usuarios,* permite revocar la autorizaciones concedidas anteriormente.

Otra forma, conocida como Control de Acceso General, tiene que ver con el nivel de prioridad y clasificación otorgado a cada objeto de la Base de Datos, como por ejemplo, "super secreto", "secreto", "confidencial", "público", etc. A su vez a cada usuario es atribuido un nivel de visualización, que posee un valor igual a uno de los niveles de clasificación atribuidos a los objetos. El control del acceso a los objetos de la base de datos se determina con base en los niveles de prioridad de los objetos y de visualización de los usuarios.

Auditoría. En aquellas situaciones en que los datos se vuelven críticos, se debe contar con el riesgo de violación de la seguridad por una persona no autorizada, además de errores involuntarios que igual pueden causar inconsistencias o falta de veracidad de la información. Para estos casos es interesante mantener un archivo de auditoría (logs), donde son registradas todas las operaciones realizadas por los usuarios de las bases de datos. En caso de sospecha de falla en la seguridad, éste archivo puede ser consultado para conocer los daños causados y/o identificar a los responsables de las operaciones irregulares.

Criptografía de Datos. Como recurso de seguridad, se puede mezclar o codificar los datos de modo que, al momento de ser almacenados en disco duro o trasmitidos por alguna línea de comunicación, no sean más que bits ininteligibles para aquellos que los accedan por un medio no oficial. La críptografía es de gran importancia en las bases de datos pues la información está almacenada por largos períodos de tiempo en medios de fácil acceso, como discos duros.

Existen varias técnicas de criptografía de datos, cuya explicación no está dentro del alcance planteado en este artículo.

Disparadores o Triggers. Un Triggers o disparador es una rutina autónoma asociada con una tabla o vista que automáticamente realiza una acción cuando una fila en la tabla o la vista se inserta (INSERT), se actualiza (UPDATE), o borra (DELETE). Un Triggers nunca se llama directamente. En cambio, cuando una aplicación o usuario intenta insertar, actualizar, o anular una fila en una tabla, la acción definida en el disparador se ejecuta automáticamente (se dispara).

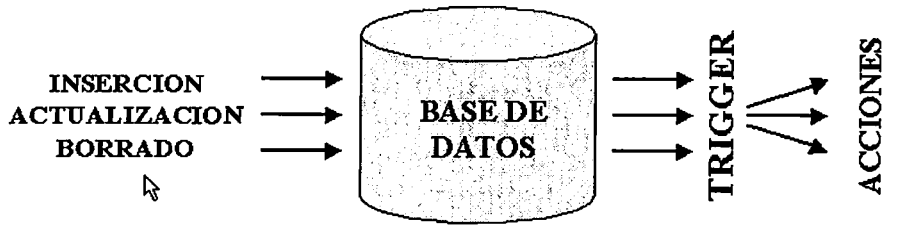

Figura 3. Desencadenante o trigger.

Las ventajas de usar los Triggers son:

- La entrada automática de restricciones de los datos, hace que los usuarios ingresen sólo valores válidos
- El mantenimiento de la aplicación se reduce, los cambios a un triggers se refleja automáticamente en todas las aplicaciones que tienen que ver con la tabla, sin necesidad de recompilar o relinquear.
- Logs automáticos de cambios a las tablas. Una aplicación puede guardar un registro corriente de cambios, creando un trigger que se dispare siempre que una tabla se modifique.
- La notificación automática de cambios a la Base de Datos con alertas del evento en los triggers.

Procedimientos Almacenados. Un Procedimiento Almacenado es un programa autocontrolado escrito en lenguaje del DBMS; el cual es almacenado como parte de la Base de Datos y sus metadatos. Una vez creado un procedimiento almacenado, se puede invocar directamente desde una aplicación, o sustituir el nombre de una tabla o

vista, por el nombre de procedimiento en cláusulas SELECT. Los procedimientos almacenados pueden recibir parámetros de entrada y retornar valores a la aplicación.

Las ventajas de usar los procedimientos almacenados incluyen:

- Diseño modular.
- Aplicaciones que acceden la misma Base de Datos pueden compartir los procedimientos almacenados, eliminando el código doble y reduciendo el tamaño de las aplicaciones.
- El fácil mantenimiento.
- Cuando un procedimiento se actualiza, los cambios se reflejan automáticamente en todas las aplicaciones, sin la necesidad de recompilar y relinkear. Las aplicaciones son compiladas sólo una vez para cada cliente.
- Los procedimientos almacenados son ejecutados por el servidor, no por el cliente, lo que reduce el tráfico en la red y mejora la performance o desempeño especialmente para el acceso del cliente remoto.
- Están almacenados en los servidores y asegurados por las medidas tomadas en la instalación, lo que impide que los usuarios normales puedan modificarlos e incluso desconocen su existencia. Este es un elemento de gran valor en lo que a seguridad se refiere.

Como se puede apreciar los Sistemas de Bases de Datos ofrecen a desarrolladores, administradores y usuarios, una gama muy completa de herramientas que permiten garantizar la integridad, consistencia, confidencialidad y en general seguridad de la información almacenada y con un elemento muy importante a favor: Las líneas de código que se requieren por parte del implementador son muy pocas, en ocasiones solo basta con una sencilla sentencia para obligar al DBMS a controlar y mantener las restricciones necesarias.

En la parte II de éste artículo se mostrará como éstas características se llevan a la práctica, usando un producto en particular, de los muchos que se encuentran disponibles en el mercado.

BIBLIOGRAFÍA

CODD, E. F. A. Relational Model for Large Shared Data Banks. ACM. 1970.

Día Internacional de Seguridad en Cómputo. Pereira. 1999.

GROFF, WEINBERG. Guía Lan Times de SQL. McGraw Hill. 1996.

Inprise. Manual Interbase Server. 1999.

KORTH. Fundamentos de Bases de Datos. McGraw-Hill.2000.

MARCON, GOMEZ; VAZ, M. SALETE; VALTER SCHASTAI; DUQUE, M. NESTOR D. Ponencia. Universidad Tecnológica de Pereira.1999.

OZSU, VALDURIEZ. Principles of Distributed Database System. Prentice Hall. 1991.

ORFALI; HARKEY; EDWARDS. Cliente/Servidor. Guía de Supervivencia. McGraw Hill. 1997.