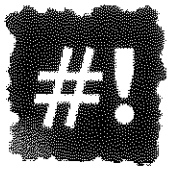

Preparador Informática

www.preparadorinformatica.com

TEMA 44. INFORMÁTICA

TÉCNICAS Y PROCEDIMIENTOS PARA LA SEGURIDAD DE LOS DATOS

TEMA 44 INF: TÉCNICAS Y PROCEDIMIENTOS PARA LA SEGURIDAD DE LOS DATOS

1. INTRODUCCIÓN

Hoy en día el mayor activo de las organizaciones son los datos y su gestión eficaz y segura. Por ello, si analizamos la mayoría de los ámbitos de actividad, nos encontramos que la utilización de las bases de datos está ampliamente extendida (al registrarse en una web, al acudir a la consulta médica, al consultar el catálogo de productos de una tienda online, etc.). Las bases de datos y los datos contenidos en ellas, son imprescindibles para llevar a cabo multitud de acciones. La necesidad de almacenar datos de forma masiva dio paso a la creación de los sistemas de bases de datos.

Todos los datos informatizados son vulnerables a muchos tipos de daños y abusos. Por eso cuando se planifica el diseño e implementación de una base de datos resulta esencial el que se prepare la forma de tratar con cualquier posible situación que pudiera dar como resultado una pérdida de datos. Entre estas situaciones podemos incluir fallos del hardware y software o errores humanos. Además, debemos tener muy en cuenta la confidencialidad de la información que contienen por lo que la información debe ser protegida contra accesos no autorizados. Por otro lado, necesitamos que nuestra base de datos esté libre de errores y sea consistente con los datos previamente almacenados.

La importancia y justificación de este tema radica en que cada vez es más importantes conocer y manejar adecuadamente las técnicas y procedimientos para la seguridad de los datos ya que en la actualidad es cada vez más habitual tener todo informatizado y nuestros datos están constantemente amenazados por personas externas para su uso fraudulento.

2. TÉCNICAS Y PROCEDIMIENTOS PARA LA SEGURIDAD DE LOS DATOS. CONCEPTOS BÁSICOS

Podemos definir una **base de datos** como un conjunto, colección o depósito de datos almacenados en un soporte informático no volátil. Los datos están interrelacionados y estructurados.

Un sistema gestor de base de datos o SGBD es una colección de programas de aplicación que permite a los usuarios la creación y el mantenimiento de una

base de datos, facilitando la definición, construcción y manipulación de la información contenida en ésta.

En la actualidad debido al gran volumen de datos que manejamos se requiere de sistemas gestores de bases de datos robustos, que nos permitan acceder y gestionar los datos de un modo eficaz y eficiente. Los sistemas gestores de bases de datos (SGBD) más extendidos son los relacionales.

El **problema de la seguridad** consiste en lograr que los recursos de un sistema sean, bajo toda circunstancia, utilizados para los fines previstos. Los datos de la base de datos deben estar protegidos contra los accesos no autorizados, de la destrucción o alteración malintencionada y de la introducción de inconsistencias. Las medidas de seguridad pueden ser físicas (p.ej. mediante tarjetas de acceso), personales (acceso sólo de personal autorizado con identificación directa de personal) y gracias al uso de las utilidades o herramientas que nos proporciona nuestro SGBD (perfiles de usuario, vistas, restricciones de uso de vistas...).

Las principales áreas de trabajo para garantizar la seguridad de los datos son la confidencialidad y la integridad:

- La **confidencialidad** trata de garantizar que los usuarios no accedan a información no autorizada. Esto se consigue por medio de:
    - Políticas activas: gestión de usuarios y permisos, vistas y cifrado de la información.
    - Políticas pasivas: auditoría.
- Garantizar la **integridad** consiste en asegurar que los datos reflejen la realidad, cumpliendo las reglas y restricciones definidas en el diseño de la BD. Para garantizar la integridad de los datos de la BD es fundamental partir de un diseño adecuado ya que eso evitará posibles anomalías en la inserción, modificación o borrado de información. Por su parte, el SGBD dispone de varias herramientas que permiten mantener la integridad de la información:
    - Políticas activas: restricciones, control de concurrencia, gestión de transacciones.
    - Políticas pasivas: respaldo y recuperación.

La forma en que se implementan estas medidas de seguridad es muy dependiente del modelo de BD utilizado. En la actualidad son los sistemas relacionales los más extendidos por lo que nos centraremos en la forma en que se implementan todas estas medidas de seguridad en los modelos relacionales.

3. SEGURIDAD DE LAS BASES DE DATOS. CONFIDENCIALIDAD

La información almacenada en ordenadores es un recurso valioso y este valor suele estar relacionado directamente con la capacidad para mantener la confidencialidad, y el nivel que debe alcanzar la protección depende hasta cierto punto de cómo sea de sensible. Existe la gran necesidad de proteger la información contra intentos malintencionados, por parte de los hackers o delincuentes informáticos. Para todo ello podemos, entre otras cosas, tomar medidas de seguridad como se explica en los siguientes apartados.

3.1. GESTIÓN DE USUARIOS Y PERMISOS

En los SGBD existen diferentes perfiles de usuario. Generalmente existen:

- **Administrador de la base de datos**: encargado de la función de administración de la BD. Va a tener las responsabilidades de definición, administración, seguridad, privacidad e integridad de la información utilizada.
- **Usuarios de la base de datos**: pueden existir diferentes usuarios con diferentes accesos y privilegios sobre los datos:
  - Usuarios **técnicos** (diseñadores, operadores y personal de mantenimiento, analistas y programadores de aplicación) que son profesionales informáticos cuyo objetivo es desarrollar los programas de aplicación que posteriormente utilizarán los usuarios finales de la BD.
  - Usuarios **finales** que son los que a través de programas de aplicación interactúan con la BD. Estos suelen ser usuarios no especializados.

Para controlar los usuarios que acceden a la base de datos y los tipos de operaciones que están autorizados a realizar los SGBD implementan varios mecanismos:

- **Autenticación de usuarios:** Los usuarios tendrán que identificarse, lo que garantiza que solamente tengan acceso a la base de datos los usuarios con permiso y además cada usuario tendrá acceso a su esquema externo de la

base de datos. El registro de lo usuarios que acceden a la base de datos permitirá también implementar funciones de monitorización y auditoría.

CREATE USER <nombre_usuario>;

- **Permisos:** Asignar permisos garantizará que los usuarios puedan acceder solamente a aquellos objetos o funciones para las que tengan permiso.

  GRANT:

  GRANT privilegios

  ON elemento

  TO nombreUsuario IDENTIFIED BY 'contraseña'

  [WITH GRANT OPTION];

  EJEMPLO:

  *GRANT ALL PRIVILEGES ON * to administrador@localhost IDENTIFIED BY "JOSE";*

  REVOKE:

  REVOKE privilegios

  ON elemento

  FROM nombre_de_usuario;

  EJEMPLO:

  *REVOKE ALL PRIVILEGES ON * FROM administrador@localhost;*

- **Roles:** Gestionar usuarios y permisos de manera individual en sistemas donde existen numerosos usuarios puede ser una labor muy tediosa. Para facilitar esto existe el concepto de rol. Los roles permiten definir tipos de usuarios en base a los permisos que éstos tendrán sobre la BD, de forma que para cada tipo de usuario se crearía un rol y en lugar de asignar los permisos a los usuarios se asignarían los permisos al rol y el rol a los usuarios.

  CREATE ROLE <rol>;

- **Perfiles:** Permiten delimitar qué recursos asignar a determinados usuarios.

  CREATE PROFIL <perfil> LIMIT <parámetros>;

3.2. VISTAS

Las vistas pueden ser utilizadas con propósitos de seguridad, delimitando el acceso de los usuarios a ciertas porciones de la información.

```
CREATE VIEW [user.]view [ ( alias [, alias] ... ) ]
AS query
[WITH CHECK OPTION [CONSTRAINT constraint] ]
```

Es importante destacar que el uso de "WITH CHECK OPTION" garantiza que se puedan insertar registros en la tabla, a través de la vista, sólo si se cumple la condición de restricción especificada.

Las vistas son, junto con los sinónimos, las herramientas principales para la definición de los esquemas externos de los usuarios de la BD.

3.3. ENCRIPTACIÓN DE DATOS

La protección mediante una palabra clave puede servir de protección, pero sólo mientras siga siendo secreta, y como sabemos cualquier método de protección mediante palabra clave puede rebasarse. Para información más delicada existe un segundo mecanismo de defensa, el encriptado de los datos. Se trata de una técnica en la cual la información de la base de datos se codifica de forma tal que resulte ininteligible sin la clave de desencriptado. Se ha dedicado una gran cantidad de investigación a esta área, y se han desarrollado muchos métodos diferentes, por ejemplo, el tipo más simple corresponde a una sustitución de un carácter por otro. Normalmente este método no se utiliza por su debilidad frente a ataques y se emplean métodos que den lugar a encriptados difícilmente desencriptables pero que a la vez no exijan una enorme cantidad de cálculo realizar esa encriptación.

Habitualmente se usan dos formas para cifrar la información por parte de los SGBD:

- Por medio de funciones: Implementa funciones para encriptar y desencriptar la información, de forma que para insertar datos encriptados habrá que hacerlo usando la función de encriptado y para recuperarlos habrá que hacerlo usando la función de desencriptado.
- Cifrado transparente: Este cifrado se puede implementar a nivel de columna (permitirá cifrar columnas concretas de una tabla) o a nivel de almacenamiento de datos (se cifrará el fichero o estructura de datos donde se almacena la información con todas sus tablas). El usuario únicamente tendrá que establecer una contraseña de acceso (que no se guardará con los datos, sino en un módulo aparte del SGBD), y será el gestor el que se encargue de encriptar y desencriptar la información automáticamente.

3.4. PROGRAMAS DE APLICACIÓN

El uso de programas de aplicación es una técnica bastante común de proporcionar seguridad a una base de datos. La mayoría de los sistemas soportan el empleo de programas de aplicación. Si un sistema de bases de datos no soporta mecanismos de protección mediante contraseñas, o si la cantidad de protección incorporada es poco adecuada, las deficiencias pueden ser corregidas mediante programas de protección escritos por el usuario. Así, un programa puede actuar como interfaz entre una persona que está introduciendo datos y la base de datos, incorporando al mismo tiempo a ésta una seguridad adicional. Esencialmente, un programa de aplicación puede realizar virtualmente cualquier tipo de protección mediante contraseñas.

3.5. AUDITORÍA

La auditoría de la BD permitirá al administrador de ella conocer detalles acerca del uso de la BD por parte de los usuarios. Existen dos niveles de auditoría:

- A nivel de sesión: Nos permite conocer detalles propios de la sesión (usuarios conectados a la BD, tablas a las que accede, etc)
- A nivel de objetos: Nos permite conocer qué usuarios acceden en qué momentos a ciertos datos y que operaciones realizan sobre ellos.

Toda la información de auditoría se guarda en BDs internas del SGBD lo que implica mayor carga para el SGBD y consumo de espacio en disco para guardar todos estos accesos, por lo que se deben limitar los objetos auditados para no comprometer los recursos del sistema.

4. INTEGRIDAD DE LAS BASES DE DATOS

Además de proteger nuestra base de datos ante accesos no deseado también es necesario protegerlas contra la corrupción originada por la presencia de información de poca calidad como datos no válidos o inconsistentes. Estos errores de datos pueden aparecer en cualquier momento, pero afortunadamente es posible proteger la base de datos contra muchos de estos errores mediante técnicas adecuadas de validación.

4.1. RESTRICCIONES

Una condición impuesta sobre un conjunto determinado de datos se suele denominar una restricción o control de integridad. Las restricciones pueden aplicarse bien a columnas individuales, a la relación entre dos columnas diferentes (normalmente entre tablas distintas) o a las filas de una o más tablas. Cuando se intente introducir una nueva fila de datos que viole las condiciones especificadas por alguna restricción, se negará la entrada de la misma en la base de datos.

Los SGBD nos permiten tener verificación de restricciones de manera automática sólo requiriendo que el diseñador introduzca las líneas apropiadas dentro de la definición de la base de datos. Pero la mayoría de los SGBD tienen capacidades muy limitadas en esta área. Por eso existe otro mecanismo para la especificación de restricciones que es el uso de programas de aplicación para el control de la entrada de toda la información de una base de datos. Aunque este mecanismo es mucho más seguro tiene como desventaja el gasto considerable de trabajo por parte de los diseñadores e implementadores de la base de datos. Por todo esto la solución más frecuente suele ser la combinación de los dos tipos de mecanismos de comprobación.

4.2. TRANSACCIONES

Una transacción es una unidad lógica de trabajo. Es una secuencia de operaciones en una base de datos mediante la cual un estado consistente de la base de datos se transforma en otro estado consistente, sin conservar por fuerza la consistencia en todos los estados intermedios.

Para ello tenemos un componente del sistema encargado de este propósito denominado "gestor de transacciones", y las operaciones en SQL son COMMIT y ROLLBACK.

La mayoría de los SGBD incluyen las funciones rollback, commit y autocommit. Supongamos que queremos borrar una fila de una tabla, pero, al teclear la orden SQL, se nos olvida la cláusula WHERE y borramos todas las filas de la tabla. Esto no es problema pues podemos dar marcha atrás a un trabajo realizado

mediante la orden ROLLBACK, siempre y cuando no hayamos validado los cambios en la base de datos mediante la orden COMMIT.

Cuando hacemos transacciones sobre la base de datos, es decir, cuando insertamos, actualizamos y eliminamos datos en las tablas, los cambios no se aplicarán a la base de datos hasta que no hagamos un COMMIT. Esto significa que, si durante el tiempo que hemos estado realizando transacciones, no hemos hecho ningún COMMIT y de pronto se va la luz, todo el trabajo se habrá perdido, y nuestras tablas estarán en la situación de partida.

Para validar los cambios que se hagan en la base de datos tenemos que ejecutar la orden COMMIT:

```
COMMIT;
```

En algunos sistemas se nos permite validar automáticamente las transacciones sin tener que indicarlo de forma explícita. Para eso sirve el parámetro AUTOCOMMIT. El valor de este parámetro se puede mostrar con la orden SHOW, de la siguiente manera:

```
SHOW AUTOCOMMIT;
```

OFF es el valor por omisión, de manera que las transacciones (INSERT, UPDATE y DELETE) no son definitivas hasta que no hagamos COMMIT. Si queremos que INSERT, UPDATE Y DELETE tengan un carácter definitivo sin necesidad de realizar la validación COMMIT, hemos de activar el parámetro AUTOCOMMIT con la orden SET:

```
SET AUTOCOMMIT ON;
```

Ahora, cualquier INSERT, UPDATE y DELETE se validará automáticamente.

La orden ROLLBACK aborta la transacción volviendo a la situación de las tablas de la base de datos desde el último COMMIT siempre y cuando no esté activado el AUTOCOMMIT:

```
ROLLBACK;
```

Para asegurar la integridad de los datos se necesita que el sistema de base de datos mantenga las siguientes propiedades en las transacciones:

- Atomicidad.
- Consistencia.

- Aislamiento.
- Durabilidad.

4.3. RECUPERACIÓN DEL SISTEMA.

El sistema debe estar preparado para recuperarse no solo de fallos puramente locales, sino también de fallos globales. Un fallo local afecta sólo a la transacción en la que se presentó el fallo, mientras que un fallo global afecta a todas las transacciones que se estaban realizando en el momento del fallo.

Ante un fallo global, el sistema deberá recuperarse de dos tipos de fallos:

- Fallos del sistema, que afectan a todas las transacciones que se están realizando, pero no dañan físicamente a la base de datos. También se conocen como caídas suaves.
- Fallos de los medios de almacenamiento, que causan daños a la base de datos o a una porción de ella, y afectan al menos a las transacciones que están utilizando esa porción. También se conocen como caídas duras.

4.4. PROBLEMAS DE CONCURRENCIA

La mayoría de los SGBD son multiusuario. En estos sistemas se necesita un mecanismo de control de concurrencia para asegurar que ninguna transacción concurrente interfiera con las operaciones de las demás.

Son tres las situaciones en las que una transacción, aunque correcta en sí, puede producir de todos modos un resultado incorrecto debido a una interferencia por parte de alguna otra transacción:

- El problema de la modificación perdida.
- El problema de la dependencia no comprometida.
- El problema del análisis inconsistente.

El bloqueo es la técnica más usual que se utiliza para resolver problemas de concurrencia. La noción básica de bloqueo es simple: cuando una transacción requiere la seguridad de que algún objeto en la cual está interesada no cambiará de alguna manera no predecible sin que ella se dé cuenta, adquiere un bloqueo sobre ese objeto, con lo que ninguna transacción podrá leer ni modificar dicho objeto.

5. CONCLUSIÓN

En el presente tema se ha presentado una visión global de las técnicas y procedimientos para la seguridad de los datos centrándonos en las bases de datos y los sistemas gestores de bases de datos ya que actualmente son técnicas imprescindibles para protegernos porque prácticamente en la totalidad de las aplicaciones que utilizamos como usuarios o que utilizan las empresas para su gestión diaria se usan bases de datos que están expuestas a usuarios malintencionados que pueden utilizar nuestros datos para su beneficio y se encuentran también expuestas a problemas externos que pudieran ocurrir como inconsistencias, pérdidas de datos por errores humanos o incluso por catástrofes naturales. Por todo esto es necesario el conocimiento y buen uso de las técnicas y procedimientos vistos en el presente tema.

6. BIBLIOGRAFÍA

- Date D.J.: **Introducción a los sistemas de bases de datos**. Editorial Addison-Wesley
- Elmasri R. y Navathe S.: **Fundamentos de Sistemas de Bases de Datos**. Editorial Addison-Wesley
- Hansen, Gari W: **Diseño y Administración de Bases de Datos**. Editorial Prentice-Hall
- https://www.tecnologias-informacion.com/integridaddatos.html