Sistemas de Bases de Datos 2

2024

Ing. Luis Alberto Arias Solórzano

Unidad 4

Seguridad

La seguridad de los datos se asocia frecuentemente con la de la integridad de los mismos (al menos en contextos informales), pero ambos conceptos son bastante diferentes.

La *seguridad* se refiere a la protección de los datos contra su revelación, su alteración o su destrucción no autorizadas, mientras que la *integridad* se refiere a la precisión o validez de esos datos.

Para ponerlo un poco más claro:

- **Seguridad** significa proteger los datos ante usuarios no autorizados.
- **Integridad** significa protegerlos de usuarios autorizados

Seguridad

Consideraciones generales

- Aspectos legales, sociales y éticos (por ejemplo, la persona que hace la petición ¿tiene derecho legal para conocer la información solicitada como, digamos, el crédito de un cliente?)
- Controles físicos (por ejemplo, ¿el lugar en donde se encuentra la computadora o terminal está bajo llave o con alguna otra protección?)
- Cuestiones de política (por ejemplo, ¿cómo decide la empresa propietaria del sistema a quién y a qué se le permite tener acceso?)
- Problemas operacionales (por ejemplo, si se utiliza un esquema de contraseñas, ¿cómo seles mantiene en secreto? ¿con cuánta frecuencia son cambiadas?)
- Controles de hardware (por ejemplo, ¿la unidad de procesamiento proporciona alguna característica de seguridad, como claves de protección de almacenamiento o un modo de operación protegido?)
- Soporte del sistema operativo (por ejemplo, ¿el sistema operativo subyacente borra el contenido de la memoria principal y los archivos de disco cuando ha terminado de utilizarlos?)
- Los asuntos que conciernen únicamente al sistema de base de datos (por ejemplo, ¿tiene el sistema de base de datos un concepto de propiedad de los datos?).

Seguridad: Enfoques

Actualmente los DBMS modernos soportan generalmente uno o ambos enfoques con respecto a la seguridad de los datos.

Estos enfoques son conocidos como control *discrecional* y control *obligatorio,* respectivamente. En ambos casos, la unidad de datos u "objeto de datos" que necesite ser protegido puede comprender desde toda la base de datos (por un lado) hasta un componente específico dentro de una tupia específica (por otro).

La manera en que difieren los dos enfoques está indicada brevemente por el siguiente esquema:

■ En el caso del control **discrecional,** un usuario específico tendrá generalmente diferentes derechos de acceso (también conocidos como **privilegios)** sobre diferentes objetos; además, existen muy pocas limitaciones —es decir, limitaciones inherentes— sobre qué usuarios pueden tener qué derechos sobre qué objetos (por ejemplo, el usuario *Ul* podría estar autorizado para ver a *A* y no a *B;* y en cambio, el usuario *U2* podría estar autorizado para ver a *B* y no a A). Por lo tanto, los esquemas discrecionales son muy flexibles.

■ Por el contrario, en el caso del control **obligatorio,** cada objeto de datos está etiquetado con un nivel de clasificación determinado y a cada usuario se le da un nivel de **acreditación.** Un objeto de datos específico sólo puede ser accedido por los usuarios que tengan el nivel de acreditación adecuado. Por lo tanto, los esquemas obligatorios tienden a ser jerárquicos por naturaleza y (por lo tanto) comparativamente rígidos. (Si el usuario *Ul* puede ver a A pero no a *B,* entonces la clasificación de *B* debe ser más alta que la de A y por lo tanto, ningún usuario *U2* podrá ver a *B* y no a A.)

Autenticación

- La autenticación se refiere a la tarea de verificar la identidad de una persona o software que se conecte a una base de datos. La forma más simple consiste en una contraseña secreta que se debe presentar cuando se abra una conexión a la base de datos.

- La autenticación basada en palabras clave se usa ampliamente por los sistemas operativos y bases de datos. Sin embargo, el uso de contraseñas tiene algunos inconvenientes, especialmente en una red. Si un husmeador es capaz de «oler» los datos que circulan por la red, puede ser capaz de encontrar la contraseña que se está enviando por la red. Una vez que el husmeador tiene un usuario y contraseña, se puede conectar a la base de datos pretendiendo que es el usuario legítimo.

- Un esquema más seguro es el sistema de **desafío respuesta**. El sistema de bases de datos envía una cadena de desafío al usuario. El usuario cifra la cadena de desafío usando una contraseña secreta como clave de cifrado y devuelve el resultado. El sistema de bases de datos puede verificar la autenticidad del usuario descifrando la cadena con la mima contraseña secreta, y comparando el resultado con la cadena de desafío original. Este esquema asegura que las contraseñas no circulen por la red.

CIFRADO DE DATOS

- Hemos supuesto —hasta el momento— que cualquier posible infiltrador estará usando las facultades normales del sistema para acceder a la base de datos. Ahora pondremos nuestra atención en el caso de un "usuario" que trata de *dejar de lado* al sistema (por ejemplo, eliminando físicamente parte de la base de datos o interviniendo una línea de comunicación). La medida más efectiva en contra de tal amenaza es el **cifrado de datos** (o encriptación, como también se conoce); es decir, el guardado y la transmisión de datos sensibles en forma cifrada

- Para explicar algunos de los conceptos del cifrado de datos necesitamos presentar un poco más de terminología. A los datos originales (sin cifrado) se les llama **texto plano.** El texto plano es **cifrado** sometiéndolo a un **algoritmo de cifrado** —cuyas entradas son el texto plano y la **clave de cifrado**— y a la salida de este algoritmo —la forma cifrada del texto plano— se le llama texto **cifrado.** Los detalles del algoritmo de cifrado son públicos —o al menos no están ocultos especialmente— pero la clave de cifrado se mantiene en secreto. El texto cifrado, que debe ser ininteligible para cualquiera que no posea la clave de cifrado, es lo que se guarda en la base de datos o se transmite por la línea de comunicación.

Seguridad

Privilegios en SQL

- La norma SQL incluye los privilegios **delete, insert, select** y **update** El privilegio **select** se corresponde con el privilegio de **lectura**. SQL también incluye un privilegio
- **references** que permite a un usuario o papel declarar claves externas al crear relaciones. Si la relación quese va a crear incluye una clave externa que hace referencia
- a atributos de otra relación, el usuario o papel debe haber recibido el privilegio **references** sobre esos atributos. El motivo de que el privilegio **references** sea
- una característica útil es algo sutil y se explica más adelante en este mismo apartado.
- El lenguaje de definición de datos de SQL incluye órdenes para conceder y retirar privilegios.
- La instrucción **grant** se utiliza para conferir autorizaciones. La forma básica de esta instrucción es la siguiente:**grant** <lista de privilegios> **on** <nombre de relación de lista/objeto de BD> **to** <lista de usuarios/papeles>

Tarea

- Investigar los tipos de autenticación de los principales DBMS (elegir uno)
- Y que roles por default o administrativos existen en cada uno.

Gracias