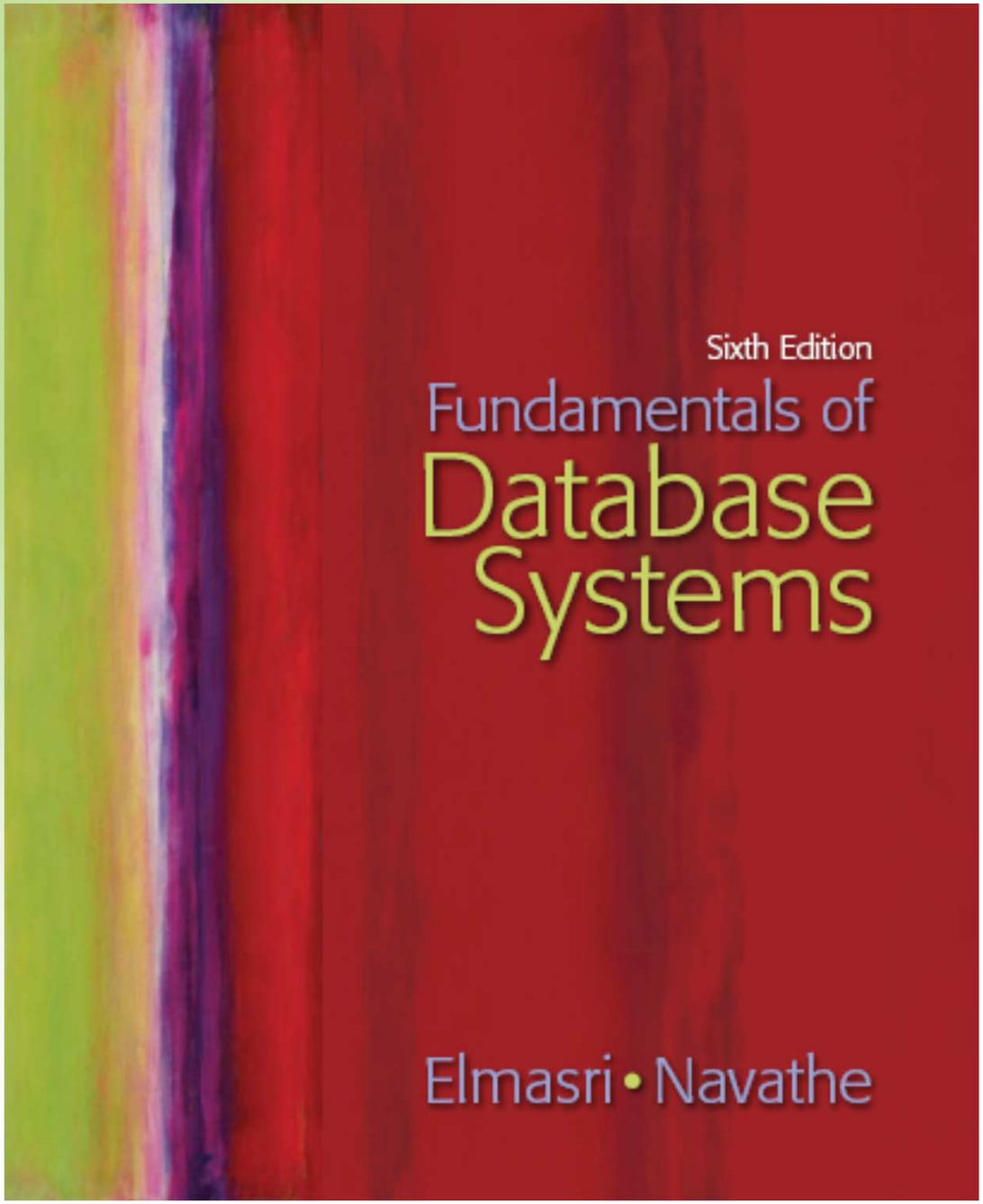
Chapter 24

Database Security



Addison-Wesley
is an imprint of
PEARSON

1.1 Introducción a temas de seguridad de bases de datos

- Tipos de seguridad
  - Se consideran varios aspectos, incluyendo:
    - Cuestiones éticas y legales que consideran el derecho a acceder a cierto tipo de información
    - Políticas gubernamentales, institucionales o corporativas (v. g., registros médicos personales)
    - Cuestiones relacionadas con el sistema: seguridad a nivel de hardware, sistema operativo o DBMS
    - Múltiples niveles de seguridad en las organizaciones

1.1 Introducción a temas de seguridad de bases de datos (2)

- Las amenazas a las bases de datos pueden causar
  - Pérdida de **integridad**
  - Pérdida de **disponibilidad**
  - Pérdida de **confidencialidad**

1.1 Introducción a temas de seguridad de bases de datos (3)

- Un DBMS típicamente incluye un subsistema de seguridad y autorización que es el responsable de garantizar la seguridad de porciones de una base de datos contra accesos no autorizados.

- Hay dos mecanismos principales de seguridad en bases de datos:
  - **Discrecionales (privilegios a usuarios)**
  - **Obligatorios (seguridad por niveles)**

1.1 Introducción a temas de seguridad de bases de datos (4)

- Se pueden implementar cuatro tipos de medidas para proporcionar seguridad en las bases de datos:
  - **Control de acceso**
  - **Control de inferencia**
  - **Control de flujo**
  - **Cifrado**

1.1 Introducción a temas de seguridad de bases de datos: **control de acceso** (5)

- Los mecanismos de seguridad de un DBMS deben incluir previsiones para restringir el acceso a la base de datos como un todo.
  - Esta función es llamada **control de acceso** y es administrada creando cuentas de usuario y contraseñas para controlar el proceso de registro al DBMS.

1.1 Introducción a temas de seguridad de bases de datos: **control de inferencia** (6)

- Un problema de seguridad asociado con las bases de datos es el de controlar el acceso a **bases de datos estadísticas**, las cuales son usadas para proporcionar información estadística o resúmenes de valores basados en diversos criterios.
  - Las medidas para el control del problema de seguridad en **bases de datos estadísticas** se llaman **medidas de control de inferencia**.

1.1 Introducción a temas de seguridad de bases de datos: **control de flujo** (7)

- Otro tema de seguridad es el del **control de flujo**, el cual impide que la información fluya de tal manera que llegue a usuarios no autorizados.

1.1 Introducción a temas de seguridad de bases de datos: cifrado (8)

- Una cuestión final de seguridad es la del **cifrado de datos**, el cual es usado para proteger datos sensitivos (v. g., números de tarjetas de crédito) que son transmitidos vía redes de comunicación.
- Los datos son **codificados** usando algún **algoritmo de codificación**.
  - Un usuario no autorizado que accede a datos codificados tendrá dificultades para descifrarlos; sin embargo, usuarios autorizados tendrán algoritmos de decodificación (o claves) para descifrarlos.

1.2 Seguridad de la base de datos y el DBA

- El administrador de la base de datos (**DBA**) es la autoridad central encargada de la administración del sistema de base de datos.
  - Las responsabilidades del DBA incluyen:
    - Otorgar privilegios a los usuarios que necesitan usar el sistema
    - Clasificar a los usuarios y a los datos de acuerdo con las políticas de la organización
- El DBA es el responsable de la seguridad total del sistema de base de datos.

1.2 Seguridad de la base de datos y el DBA (2)

- El DBA tiene cuentas especiales en el DBMS
  - A veces éstas son llamadas cuentas del sistema o de súper usuario.
  - Estas cuentas proporcionan capacidades dentro del sistema tales como:
    - 1. Creación de cuentas
    - 2. Otorgamiento de privilegios
    - 3. Revocación de privilegios
    - 4. Asignación de niveles de seguridad

1.3 Control de acceso, cuentas de usuario y auditorías de la base de datos

- En términos generales, en sistemas multi-usuario, una persona o grupo de personas que requieren acceder al sistema de BD necesitan una **cuenta de usuario** y una **contraseña** para tal fin.

1.3 Control de acceso, cuentas de usuario y auditorías de la base de datos (2)

- El sistema de BD debe llevar un registro **de todas las operaciones** en la base que son efectuadas por cada usuario durante **cada sesión de trabajo**.
  - Para llevar un registro de todas las actualizaciones realizadas a la BD, junto con el usuario particular que las aplicó, se puede modificar la **bitácora del sistema** para tal fin (ya que ésta incluye una entrada para cada operación efectuada a la BD).

1.3 Control de acceso, cuentas de usuario y auditorias de la base de datos (3)

- Si se sospecha cualquier alteración/falsificación a la BD, una **auditoría de base de datos** debe ser realizada.
  - Una auditoría de base de datos consiste en revisar la bitácora para examinar todos los accesos y operaciones aplicadas a la BD durante un cierto periodo de tiempo.
- Una bitácora de base de datos puede ser usada, principalmente, para propósitos de seguridad (**audit trail)**.

2. Control de acceso discrecional basado en otorgamiento y revocación de privilegios

- El método típico de forzar el **control de acceso discrecional** en un sistema de base de datos está basado en el **otorgamiento y la revocación de privilegios**.

2.1Tipos de privilegios discrecionales

- A **nivel de cuentas**:
  - En este nivel, el DBA especifica los privilegios particulares que cada cuenta tiene independientemente de las tablas en la BD.

- A **nivel de tabla**:
  - En este nivel, el DBA puede controlar los privilegios de acceso a cada tabla individual, o vista, en la BD.

2.1Tipos de privilegios discrecionales (2)

- Los privilegios a **nivel de cuenta** se refieren a las capacidades proporcionadas a una cuenta y pueden ser:
  - El privilegio de **CREATE SCHEMA** o **CREATE TABLE**, para crear un esquema o una tabla base.
  - El privilegio de **CREATE VIEW**.
  - El privilegio de **ALTER**, para aplicar cambios al esquema tales como agregar o quitar atributos de tablas.
  - El privilegio de **DROP**, para borrar tablas o vistas.
  - El privilegio de **MODIFY**, para insertar, borrar o cambiar tuplas.
  - Y el privilegio de **SELECT**, para recuperar información desde la BD usando consultas con **SELECT**.

2.1Tipos de privilegios discrecionales (3)

- El segundo nivel de privilegios se aplica a **nivel tabla**
  - Esto incluye a las **tablas base** y a las virtuales (**vistas**).
- El otorgamiento y revocación de privilegios generalmente sigue un modelo de autorización conocido como el modelo de la matriz de acceso donde:
  - Las **filas** de una matriz M representan **sujetos** (usuarios, cuentas, programas)
  - Las **columnas** representan **objetos** (tablas, registros, columnas, vistas, operaciones).
  - Cada posición **M(i,j)** en la matriz representa los tipos de privilegios (read, write, update) que el **sujeto i** tiene sobre el **objeto j**.

2.1Tipos de privilegios discrecionales (4)

- Para controlar el otorgamiento y la revocación de privilegios sobre una relación, cada relación R en una BD tiene asignada una **cuenta propietaria**, que típicamente es la cuenta usada cuando la relación fue creada.
  - El propietario de una relación tiene todos los privilegios sobre ella.
  - El propietario de una cuenta puede **pasar privilegios** sobre cualquiera de sus relaciones a otro usuarios **otorgando** privilegios a las cuentas de ellos.

2.1Tipos de privilegios discrecionales (5)

- En SQL los siguientes tipos de privilegios pueden ser otorgados sobre cada relación individual R:
  - Privilegio **SELECT** (retrieval o read) sobre R:
    - Da a la cuenta el privilegio retrieval.
    - En SQL esto da a la cuenta el privilegio de usar la frase **SELECT** para recuperar tuplas de R.
  - Privilegio **MODIFY** sobre R:
    - Da a la cuenta la capacidad de modificar tuplas de R.
    - En SQL este privilegio está además dividido en privilegios **UPDATE**, **DELETE** e **INSERT** para aplicar las instrucciones SQL correspondientes a R.
    - Adicionalmente, los privilegios **INSERT** y **UPDATE** pueden especificar que sólo ciertos atributos pueden ser actualizados por la cuenta.

2.1Tipos de privilegios discrecionales (6)

- Privilegio **REFERENCES** sobre R:
  - Da a la cuenta la capacidad de **referir** a la relación R cuando se especifican restricciones de integridad.
  - El privilegio también puede ser **restringido** a atributos específicos de R.

2.2 Especificando privilegios usando vistas

- Las **vistas** representan un mecanismo importante de autorización discrecional. Por ejemplo,
  - Si el propietario A de una relación R quiere que otra cuenta B pueda recuperar sólo algunos campos de R, entonces A puede crear una vista V de R que incluya sólo esos atributos y entonces otorgar SELECT sobre V a B.
  - Lo mismo aplica para limitar B a recuperar sólo ciertas tuplas de R; una vista V' puede ser creada definiéndola por medio de una consulta que seleccione sólo las tuplas de R que A quiere permitir acceder a B.
  - Hay que notar que para crear una vista, la cuenta debe tener el privilegio **SELECT** sobre todas las relaciones involucradas en su definición.

2.3 Revocación de privilegios

- En algunos casos es deseable otorgar un privilegio a un usuario temporalmente. Por ejemplo,
  - El propietario de una relación puede querer otorgar el privilegio **SELECT** a un usuario para una tarea específica y luego revocárselo una vez que la tarea se completa.
  - Así, un mecanismo para **revocar** privilegios es necesario. En SQL, la instrucción **REVOKE** se incluye con el propósito de **cancelar privilegios**.

2.4 Propagación de privilegios usando GRANT OPTION

- Cuando el propietario A de una relación R otorga un privilegio sobre R a otra cuenta B, el privilegio puede ser dado a B con o sin **GRANT OPTION**.
- Si se da **GRANT OPTION**, esto significa que B también puede otorgar ese privilegio sobre R a otras cuentas.
  - Suponga que A le da a B **GRANT OPTION** sobre R y que B entonces otorga ese privilegio a una tercera cuenta C, también con **GRANT OPTION**. De este modo, los privilegios sobre R pueden propagarse sin el conocimiento del propietario de R.
  - Si la cuenta propietaria A ahora revoca el privilegio otorgado a B, todos los privilegios que B propagó, basados en ese privilegio, deberían ser revocados automáticamente por el sistema.

2.5 Un ejemplo

EMPLOYEE

Name	Ssn	Bdate	Address	Sex	Salary	Dno

DEPARTMENT

Dnumber	Dname	Mgr_ssn

Figure 24.1
Schemas for the two relations EMPLOYEE and DEPARTMENT.

2.5 Un ejemplo (2)

- Suponga que el DBA crea cuatro cuentas
  - A1, A2, A3, A4
- Y quiere que sólo A1 pueda crear tablas base. Entonces el DBA debe dar la siguiente instrucción GRANT en SQL

```
GRANT CREATETAB TO A1;
```

2.5 Un ejemplo (3)

- Suponga que A1 **crea** las dos tablas base **EMPLOYEE** y **DEPARTMENT**
  - A1 es entonces **propietario** de estas dos tablas y de aquí, de todos los privilegios sobre las mismas.
- Suponga que A1 quiere otorgar a A2 el privilegio de insertar y borrar tuplas en ambas tablas , pero A1 no quiere que A2 pueda propagar estos privilegios a cuentas adicionales:

```
GRANT INSERT, DELETE ON
    EMPLOYEE, DEPARTMENT TO A2;
```

2.5 Un ejemplo (4)

- Suponga que A1 quiere permitir recuperar información a A3 de cualquiera de las dos tablas y que también pueda propagar el privilegio SELECT a otras cuentas.
- A1 puede dar la instrucción:

```
GRANT SELECT ON EMPLOYEE, DEPARTMENT
    TO A3 WITH GRANT OPTION;
```

- A3 puede otorgar el privilegio **SELECT** sobre **EMPLOYEE** a A4 con:

```
GRANT SELECT ON EMPLOYEE TO A4;
```

  - Note que A4 no puede propagar el privilegio SELECT debido a que no se le dio GRANT OPTION.

2.5 Un ejemplo (5)

- Suponga que A1 decide revocar a A3 el privilegio SELECT sobre EMPLOYEE; A1 puede dar:

```
REVOKE SELECT ON EMPLOYEE FROM A3;
```

- El DBMS también debería revocar automáticamente el privilegio SELECT sobre EMPLOYEE a A4, debido a que A3 otorgó ese privilegio a A4 y A3 no tiene más ese privilegio.

2.5 Un ejemplo (6)

- Suponga que A1 quiere devolver a A3 una capacidad limitada de SELECT sobre EMPLOYEE y quiere permitir a A3 que pueda propagar ese privilegio.
  - La limitación consiste en que sólo pueda recuperar NAME, BDATE y ADDRESS y sólo para las tuplas con DNO=5.
- Entonces A1 puede crear la vista:

```
CREATE VIEW A3EMPLOYEE AS
    SELECT NAME, BDATE, ADDRESS
    FROM EMPLOYEE
    WHERE DNO = 5;
```

- Después de que la vista es creada, A1 puede otorgar **SELECT** sobre esa vista a A3 con:

```
GRANT SELECT ON A3EMPLOYEE TO A3
    WITH GRANT OPTION;
```

2.5 Un ejemplo (7)

- Finalmente, suponga que A1 quiere permitir a A4 actualizar sólo el atributo SALARY de EMPLOYEE;
- A1 puede dar:

```
GRANT UPDATE ON EMPLOYEE (SALARY) TO
A4;
```

  - Los privilegios **UPDATE** o **INSERT** pueden especificar atributos particulares que pueden ser actualizados o insertados en una tabla.
  - Los privilegios **SELECT**, **DELETE** no permiten indicar atributos específicos.