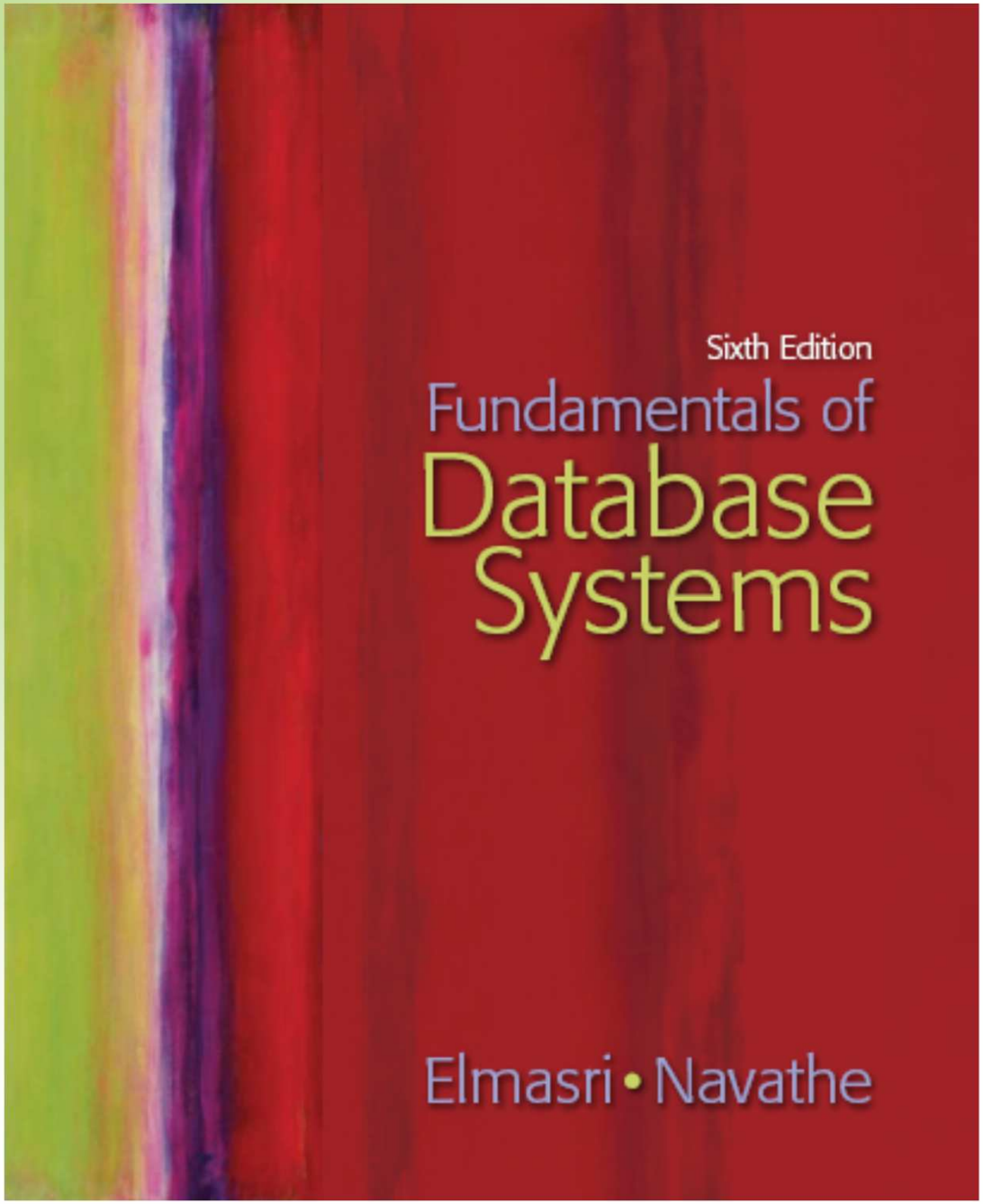
Chapter 24

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.

3 Control de acceso obligatorio para seguridad multinivel

- Las técnicas de control de acceso discrecional para otorgar y revocar privilegios sobre las relaciones tradicionalmente han sido el principal mecanismo de seguridad para las bases de datos relacionales.
- Éste es un método de todo o nada:
  - Un usuario tiene o no tiene un cierto privilegio.
- En muchas aplicaciones, una **política adicional de seguridad** es necesaria tal que clasifique a los datos y a los usuarios con un sistema basado en **clases de seguridad**.
  - Este enfoque de **control de acceso obligatorio** típicamente será **combinado** con los mecanismos de control de acceso discrecional.

3 Control de acceso obligatorio para seguridad multinivel (2)

- **Las clases de seguridad** típicas son: top secret (TS), secret (S), confidential (C) y unclassified (U), donde TS es el mayor nivel y U el menor: TS ≥ S ≥ C ≥ U
- El modelo comúnmente usado para la seguridad multinivel clasifica a cada **sujeto** (usuario, cuenta, programa) y **objeto** (relación, tupla, columna, vista, operación) en una de las clasificaciones de seguridad, TS, S, C o U:
  - A la **Autorización** (clasificación) de un sujeto S se le denomina como **clase(S**) y a la **clasificación** de un objeto O como **clase(O)**.

3 Control de acceso obligatorio para seguridad multinivel (3)

- Dos restricciones son forzadas en el acceso a los datos basadas en las clasificaciones sujeto/objeto:
  - **Propiedad de seguridad simple:** A un sujeto S no le es permitido acceso de lectura a un objeto O a menos que clase(S) ≥ clase(O).
  - **Propiedad estrella:** A un sujeto S no le es permitido escribir un objeto O a menos que clase(S) ≤ clase(O).

3 Control de acceso obligatorio para seguridad multinivel (4)

- Para incorporar las nociones de seguridad multinivel en el modelo relacional de bases de datos, es común considerar a los valores de los atributos y a las tuplas como objetos de datos.
- Así, cada atributo A está asociado con un **atributo de clasificación C** en el esquema y cada valor de atributo en una tupla está asociado con una clasificación de seguridad correspondiente.
- Además, en algunos modelos, un atributo de **clasificación de tupla** TC es agregado a los atributos de la relación para proporcionar una clasificación para cada tupla como un todo.
- Así, un esquema de **relación multinivel** R con n atributos sería representado como:
  - $R(A_1,C_1,A_2,C_2, \ldots, A_n,C_n,TC)$

  donde cada Ci representa el atributo de clasificación asociado con el atributo A_i.

3 Control de acceso obligatorio para seguridad multinivel (5)

- Ejemplo

Nombre	Salario	Evaluación	TC
Smith U	40000 C	Favorable S	S
Brown C	50000 S	Buena C	S

- El valor del atributo TC en cada tupla t – el cual es igual al mayor de todos los valores de los atributos de clasificación dentro de t – proporciona una clasificación general para la tupla misma, mientras que cada Ci proporciona una clasificación de seguridad más fina para cada valor de atributo dentro de la tupla.

3 Control de acceso obligatorio para seguridad multinivel (6)

- Una relación multinivel parecerá contener diferentes datos para sujetos (usuarios) con diferentes niveles de autorización.
  - En algunos sistemas, es posible almacenar una sola tupla en la relación a un nivel de clasificación mayor y producir las tuplas correspondientes a un nivel de clasificación menor o igual mediante un proceso de **filtrado**.
    - Ejemplo después de filtrar la tabla inicial para usuarios C:

Nombre	Salario	Evaluación	TC
Smith U	40000 C	Null C	C
Brown C	Null C	Buena C	C

    - Ejemplo después de filtrar la tabla inicial para usuarios U:

Nombre	Salario	Evaluación	TC
Smith U	Null U	Null U	U

3 Control de acceso obligatorio para seguridad multinivel (7)

- En otros sistemas, es necesario almacenar dos o más tuplas con diferentes niveles de clasificación teniendo el mismo valor para la clave aparente (que es el conjunto de atributos que habrían formado la clave primaria en una relación regular de un nivel).
- Esto lleva al concepto de **polinstanciación** donde varias tuplas pueden tener el mismo valor de clave aparente, pero tener diferentes valores de atributos para usuarios en diferentes niveles de clasificación.
  - Ejemplo (la clave aparente sería Nombre):

Nombre	Salario	Evaluación	TC
Smith U	40000 C	Favorable S	S
Smith U	40000 C	Excelente C	C
Brown C	50000 S	Buena C	S

3.1 Comparación entre el control de acceso discrecional y el obligatorio

- Las políticas de **Control de Acceso Discrecional (DAC)** se caracterizan por un alto grado de flexibilidad, el cual las hace adaptables para una gran variedad de tipos de aplicaciones.
  - El principal inconveniente de los modelos **DAC** es su vulnerabilidad a ataques maliciosos debido a que una vez que se accede a la información, no hay control alguno sobre cómo se propaga o se usa.

3.1 Comparación entre el control de acceso discrecional y el obligatorio (2)

- En contraste, las políticas **obligatorias** aseguran **un alto grado de protección** de manera que previenen cualquier flujo ilegal de información.
- Las políticas obligatorias tienen el **inconveniente de ser demasiado rígidas** y pueden ser aplicables sólo en ambientes limitados.
- En muchas **situaciones prácticas**, las **políticas discrecionales son preferidas** debido a que ofrecen un mejor balance entre seguridad y aplicabilidad.

3.2 Control de acceso basado en roles

- **El control de acceso basado en roles (RBAC)** surgió en los 90s como una tecnología para administrar y forzar la seguridad en sistemas empresariales de gran escala.
- Su noción básica es que los permisos están asociados con roles y los usuarios son asignados a los roles apropiados.
- Los roles pueden ser creados usando las instrucciones **CREATE ROLE** y **DESTROY ROLE**.
  - Las instrucciones **GRANT** y **REVOKE** comentadas bajo DAC pueden entonces ser usadas para asignar y revocar privilegios de los roles.

3.2 Control de acceso basado en roles (2)

- **RBAC** puede ser usado con los controles de acceso discrecional y obligatorio; asegura que sólo a usuarios autorizados les sea dado el acceso a ciertos datos o recursos.
- Muchos DBMSs tienen el concepto de roles y les pueden ser asignados privilegios.
- La jerarquía de roles en **RBAC** es un medio natural de organizar roles para reflejar las líneas de autoridad y responsabilidad de las organizaciones.

3.2 Control de acceso basado en roles (3)

- Otra consideración importante en los sistemas **RBAC** es la posibilidad de aplicar restricciones temporales a los roles, tales como el tiempo y la duración de actividades de un rol, y el disparo de un rol por una actividad de otro rol.
- Lo anterior permite satisfacer mejor requerimientos de seguridad en aplicaciones Web.

4 Introducción a la seguridad en bases de datos estadísticas

- **Las bases de datos estadísticas** son usadas principalmente para producir estadísticas sobre varios tipos de poblaciones.
- La base de datos puede contener **datos confidenciales** sobre individuos, los cuales deberán ser protegidos de accesos indebidos.
- A los usuarios se les permite recuperar **información estadística** sobre las poblaciones, tales como **promedios, sumas, cuentas, máximos, mínimos** y **desviaciones estándar**.

4 Introducción a la seguridad en bases de datos estadísticas (2)

- Una **población** es un conjunto de tuplas de una relación que satisface alguna condición de selección.
- Las consultas estadísticas involucran aplicar **funciones estadísticas** a la **población** de tuplas.

4 Introducción a la seguridad en bases de datos estadísticas (3)

- Por ejemplo, se puede querer recuperar la *cantidad* de individuos en una **población** o su *ingreso promedio*.
  - Sin embargo, a los usuarios estadísticos no se les permite recuperar datos individuales, tal como el ingreso de una persona específica.
- Las técnicas de seguridad en bases de datos estadísticas deben prohibir la recuperación de datos individuales.
- Esto puede lograrse prohibiendo consultas que recuperen valores de atributos y permitiendo sólo consultas que involucren funciones de agregados tales como COUNT, SUM, MIN, MAX, AVERAGE y STANDARD DEVIATION.
  - A tales consultas algunas veces se les llama **consultas estadísticas**.

4 Introducción a la seguridad en bases de datos estadísticas (4)

- Es responsabilidad del DBMS asegurar la confidencialidad de la información de los individuos, y aún así proporcionar resúmenes estadísticos útiles acerca de esos individuos. La **protección de la privacidad** de los individuos en una base de datos estadística es de primordial importancia.
- En algunos casos es posible **inferir** los valores de tuplas individuales a partir de una secuencia de consultas estadísticas.
  - Esto es particularmente cierto cuando las condiciones producen una población pequeña en cantidad de tuplas.

4 Introducción a la seguridad en bases de datos estadísticas: ejemplo (5)

- Suponer que se tienen las consultas:
  - Q1: select count(*) from Persona where <condición>;
  - Q2: select avg(salario) from Persona where <condición>;
- Suponer que se quiere conocer el salario de Jane Smith y se sabe que tiene un PhD y que vive en Bellaire, Texas. Se puede dar la siguiente condición en Q1:
  - (grado='PhD' and sexo='F' and cd='Bellaire' and edo='Texas')
- Si la cuenta es 1, se puede dar Q2 con la misma condición.
- Prevención:
  - No permitir obtener resultados con poca cantidad de tuplas.
  - Prohibir secuencias de consultas que involucren a la misma población de tuplas.

5 Control de flujo

- El **control de flujo** regula la distribución o flujo de información entre objetos accesibles.
- Un **flujo** entre el objeto X y el objeto Y ocurre cuando un programa lee valores de X y escribe valores en Y.
  - Lo controles de flujo checan que la información contenida en algunos objetos no fluya explícita o implícitamente en objetos menos protegidos.
- Una **política de flujo** especifica los canales a lo largo de los cuales se permite que la información se mueva.
  - La política de flujo más simple especifica dos clases de información:
    - confidencial (C) y no confidencial (N)
  - y permite todos los flujos excepto aquellos de C a N.

5.1 Canales encubiertos

- Un **canal encubierto** permite una transferencia de información que viola la seguridad o la política.
- Un **canal encubierto permite** que la información pase de un nivel de clasificación más alto a un nivel más bajo a través de **medios impropios**.
- Pueden ser clasificados en dos categorías:
  - **Canales de almacenamiento**, en los que la información es transmitida con el acceso a los sistemas de información.
  - **Canales de tiempo**, que permiten que la información sea transmitida por el "timing" (disparo) de eventos o procesos.

6 Infraestructuras de cifrado y clave pública

- **El cifrado** es un manera de mantener a los datos seguros en un ambiente inseguro.
- **El cifrado** consiste en aplicar un **algoritmo de cifrado** a los datos usando alguna **clave de cifrado** pre-especificada.
- Los datos resultantes tiene que ser **descifrados** usando una **clave de descifrado** para recuperar los datos originales.

6.1 Los datos y los estándares avanzados de cifrado

- El **Estándar de Cifrado de Datos (DES)** es un sistema desarrollado por el gobierno de EE. UU. para uso público general.
  - Ha sido ampliamente aceptado como un estándar criptográfico tanto en los EE. UU., como fuera de ahí.
- **DES** puede proveer un cifrado fin-a-fin en el canal entre el emisor A y el receptor B.

6.1 Los datos y los estándares avanzados de cifrado (2)

- El algoritmo **DES** es una combinación cuidadosa y compleja de dos de los bloques de construcción fundamentales del cifrado:
  - **sustitución** y **permutación**.
- El algoritmo **DES** deriva su fortaleza a partir de una aplicación repetida de estas dos técnicas en un total de 16 ciclos.
  - **El texto plano** (la forma original del mensaje) es **cifrado** en bloques de **64 bits (o mayores)**.

6.1 Los datos y los estándares avanzados de cifrado (3)

- Después de cuestionarse la conveniencia de **DES**, el Instituto Nacional de Estándares (**NIST**) introdujo el Estándar de Cifrado Avanzado (**AES**).
  - Este algoritmo tiene un tamaño de bloque de **128 bits** y así toma más tiempo para romperlo.

6.2 Cifrado de clave pública

- En 1976 Diffie y Hellman propusieron una nueva clase de criptosistema el cual llamaron **cifrado de clave pública**.
- **Los algoritmos de clave pública** están basados en **funciones matemáticas** más que en operaciones sobre patrones de bits.
  - También involucran el uso de **dos claves separadas**
    - en contraste al cifrado convencional el cual usa sólo una clave.
  - El uso de dos claves tiene profundas consecuencias en las áreas de confidencialidad, distribución de claves y autenticación.

6.2 Cifrado de clave pública (2)

- Las dos claves usadas son referidas como la **clave pública** y la **clave privada**.
  - La **clave pública**, como su nombre lo indica, es conocida públicamente y puede ser transmitida en un medio no seguro.
  - La **clave privada** se mantiene secreta, sólo es conocida por el propietario y no se transmite en absoluto. Es referida como clave privada más que como clave secreta (para evitar confusión con el cifrado convencional).

6.2 Cifrado de clave pública (3)

- El esquema, o infraestructura, tiene seis ingredientes:
  - **Texto plano**: son los datos, o mensaje legible, que son dados como entrada al algoritmo.
  - **Algoritmo de cifrado**: el algoritmo ejecuta varias transformaciones sobre el **texto plano**.
  - **Claves pública y privada**: son el par de claves que han sido seleccionadas, tal que una es usada para el cifrado y la otra para el descifrado.
    - Las transformaciones exactas ejecutadas por el algoritmo dependen de la claves pública o privada proporcionadas como entrada.

6.2 Cifrado de clave pública (4)

- Esquema (cont.):
  - **Texto cifrado**:
    - Es el mensaje codificado producido como salida. Depende del **texto plano** y la clave.
    - Para un mismo mensaje, dos claves diferentes producirán dos **textos cifrados** diferentes.
  - **Algoritmo de descifrado**:
    - Este algoritmo acepta el **texto cifrado** y la clave privada y produce el **texto plano** original.

6.2 Cifrado de clave pública (5)

- El algoritmo criptográfico de clave pública de propósito general se basa en
  - **Una clave para el cifrado** y
  - **Una clave** diferente, pero relacionada, **para el descifrado**.

6.2 Cifrado de clave pública (6)

- Los pasos esenciales son como sigue:
  - Cada usuario genera un **par de claves** a ser usadas para el cifrado y descifrado de mensajes.
  - Cada usuario coloca una de los dos claves en un registro público u otro archivo accesible. Ésta es la **clave pública**. La clave acompañante es mantenida **privada**.
  - Si un emisor desea enviar un mensaje privado a un receptor, el emisor **cifra** el mensaje usando la clave pública del receptor.
  - Cuando el receptor recibe el mensaje, lo **descifra** usando su clave privada.
    - Ningún otro receptor puede descifrar el mensaje debido a que sólo él conoce su clave privada.

6.2 Cifrado de clave pública (7)

- El algoritmo **RSA**, uno de los primeros esquemas de clave pública, fue introducido en 1978 por Ron Rivest (R), Adi Shamir (S) y Len Adleman (A), en MIT, y fue nombrado así por ellos.
  - El algoritmo de cifrado RSA incorpora resultados de la **teoría de números**, combinados con la dificultad de determinar los factores primos de un número grande.

6.2 Cifrado de clave pública (8)

- Dos claves, **d** y **e**, son usadas para descifrar y cifrar.
  - Una propiedad importante es que **d** y **e** pueden ser intercambiadas.
  - Se escoge **n** como un entero grande que es el producto de **dos números primos distintos grandes**, **a** y **b,** n= a x b.
  - La clave de cifrado **e** es escogida aleatoriamente como un número entre 1 y **n,** tal que sea un primo relativo a (a-1) x (b-1).
  - El bloque de **texto plano P** es cifrado como $\mathbf{P^e \bmod n}$.
  - Debido a que la exponenciación es hecha **mod n**, factorizar $\mathbf{P^e}$ para descubrir el texto plano cifrado es difícil.
  - Sin embargo, la clave de descifrado **d** debe ser escogida cuidadosamente tal que $\mathbf{(P^e)^d \bmod n = P}$.
  - la clave de descifrado **d** puede ser calculada a partir de la condición $\mathbf{d \times e = 1 \bmod ((a-1)x(b-1))}$.
  - Así, el legítimo receptor que conoce **d** simplemente calcula $\mathbf{(P^e)^d \bmod n = P}$ y recupera **P** sin tener que factorizar a $\mathbf{P^e}$.

6.3 Firmas digitales

- Una **firma digital** es un ejemplo del empleo de técnicas de cifrado para proporcionar servicios de autenticación en comercio electrónico.
- Una firma digital es una manera de asociar una marca única a un individuo con un cuerpo de texto.
  - La marca deberá ser inolvidable, significando que otros deberán poder checar que la firma proviene de la fuente original.
- Una firma digital consiste de cadenas de símbolos.
  - La firma deberá ser diferente para cada uso.
    - Esto puede ser logrado haciendo cada firma digital una función del mensaje que es firmado, junto a una marca de tiempo.
  - También se requiere de una clave secreta del dueño de la firma, la cual no debe conocer el receptor del mensaje.
  - Las técnicas de clave pública son el medio de crear firmas digitales.