

Modelos y bases de datos

Seguridad

CEIS

2024-1

Agenda

Seguridad

Mecanismos

Caso : Control de Acceso - Discrecional

- Privilegio sobre datos

- Privilegios sobre acciones

Agenda

Seguridad

Mecanismos

Caso : Control de Acceso - Discrecional

- Privilegio sobre datos

- Privilegios sobre acciones

Conceptos

Seguridad

Seguridad vs Integridad

Conceptos

Seguridad

La seguridad se refiere a la protección de los datos contra su revelación, su alteración o su destrucción no autorizadas.

Seguridad vs Integridad

- ▶ Seguridad significa proteger los datos ante usuarios no autorizados
- ▶ Integridad significa proteger los datos de usuarios autorizados

Garantizar que los usuarios tengan permiso de hacer las cosas que están tratando de hacer

Asegurar que las cosas que están tratando de hacer sean correctas

Conceptos

Seguridad

La seguridad se refiere a la protección de los datos contra su revelación, su alteración o su destrucción no autorizadas.

Mecanismos

Conceptos

Seguridad

La seguridad se refiere a la protección de los datos contra su revelación, su alteración o su destrucción no autorizadas.

Mecanismos

- ▶ **Control de acceso**
Definir explícitamente permisos de acciones sobre elementos determinados
- ▶ **Cifrado**
Guardar o transmitir la información sensible de manera cifrada
- ▶ **Registro de auditoría**
Guardar las acciones realizadas por los usuarios

Agenda

Seguridad

Mecanismos

Caso : Control de Acceso - Discrecional

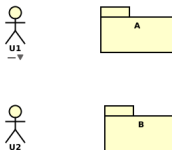
Privilegio sobre datos

Privilegios sobre acciones

Control de acceso

Control de acceso

Definir explícitamente permisos de acciones sobre objetos determinados a personas identificadas (ID.Clave)



Mecanismos

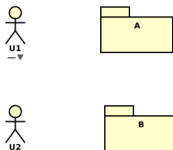
- ▶ Obligatorio

- ▶ Discrecional

Control de acceso

Control de acceso

Definir explícitamente permisos de acciones sobre objetos determinados a personas identificadas (ID.Clave)



Mecanismos

► Obligatorio

Cada objeto de datos está etiquetado con un nivel de clasificación determinado y a cada usuario se le da un nivel de acreditación

Si U1 está autorizado para ver A y no ver B entonces nadie podrá ver B y no A

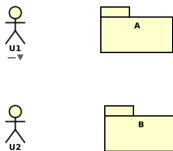
¿POR QUÉ?

► Discrecional

Control de acceso

Control de acceso

Definir explícitamente permisos de acciones sobre objetos determinados a personas identificadas (ID.Clave)



Mecanismos

► Obligatorio

Cada objeto de datos está etiquetado con un nivel de clasificación determinado y a cada usuario se le da un nivel de acreditación

Si U1 está autorizado para ver A y no ver B entonces nadie podrá ver B y no A

¿POR QUÉ?

► Discrecional

Un usuario específico tendrá diferentes niveles de acceso (privilegios) sobre diferentes elementos

U1 puede estar autorizado para ver A y no ver B y U2 puede estar autorizado para ver B y no A

Cifrado

Cifrado

Guardar o transmitir la información sensible de manera cifrada

Mecanismos

► Sustitución

Se usa una clave de cifrado para determinar el caracter que va a sustituir a cada caracter del texto original

► Permutación

Los caracteres del texto son organizados de una manera diferente

Registro de auditoría

Propósito

Si hay sospecha, el registro de auditoría permite examinar lo que ha estado sucediendo

- :) Verificar que todo está bajo control
- : (Para ayudar a señalar dónde hubo un error

Contenido

1. Petición (texto de origen)
2. Terminal desde la que se llamó a la operación
3. Usuario que llamó a la operación
4. Fecha y hora de la operación
5. Varrels, tuplas, atributos afectados
6. Valores antiguos Valores nuevos

Agenda

Seguridad

Mecanismos

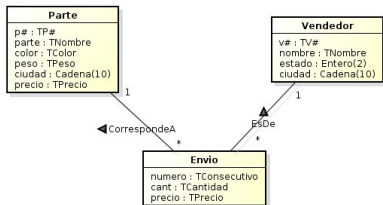
Caso : Control de Acceso - Discrecional

Privilegio sobre datos

Privilegios sobre acciones

Contexto

Envíos

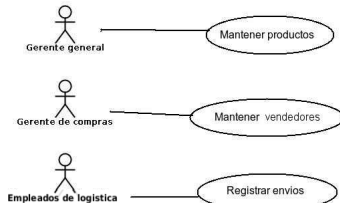


```
CREATE TABLE VENDEDOR(  
  v# CHAR(2),  
  nombre VARCHAR(20),  
  estatus NUMBER(2),  
  ciudad VARCHAR(10));
```

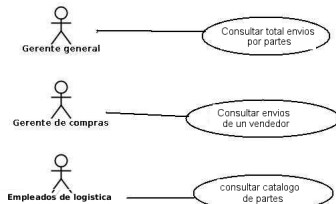
```
CREATE TABLE PARTES(  
  p# CHAR(2),  
  parte VARCHAR(20),  
  color CHAR(10),  
  peso NUMERIC(5,2),  
  ciudad VARCHAR(10));
```

```
CREATE TABLE ENVIOS(  
  v# CHAR(2),  
  p# CHAR(2),  
  cant NUMERIC(5));
```

Funciones



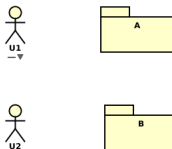
Consultas operativas



Control de acceso: Personas

Control de acceso

Definir explícitamente permisos de acciones sobre objetos determinados a personas identificadas (ID.Clave)



Personas

- ▶ Usuarios
- ▶ Roles

Usuarios y roles.

Usuarios

```
CREATE USER <nombreUsuario>  
IDENTIFIED BY <claveUsuario>;
```

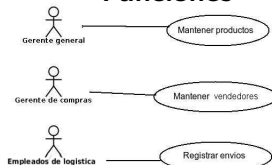
Roles

```
CREATE ROLE <nombreRol>;  
  
GRANT <nombreRol>  
TO <nombreRol>;
```

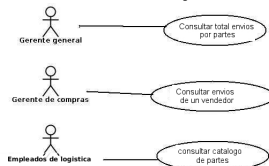
Usuarios - Roles

- ▶ Martha Perez (Gerente General)
- ▶ Jorge Amador (Gerente de Compras)
- ▶ Pedro Vargas, Luisa Medina (Empleados de Logística)

Funciones



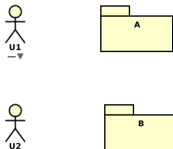
Consultas operativas



Control de acceso: Objetos

Control de acceso

Definir explícitamente permisos de acciones sobre objetos determinados a personas identificadas (ID.Clave)



Objetos

- ▶ Privilegio sobre datos
- ▶ Privilegio sobre acciones

Agenda

Seguridad

Mecanismos

Caso : Control de Acceso - Discrecional

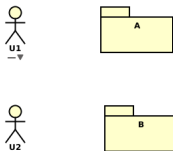
Privilegio sobre datos

Privilegios sobre acciones

Control de acceso: Privilegio sobre datos

Control de acceso

Definir explícitamente permisos de acciones sobre objetos determinados a personas identificadas (ID.Clave)



Objetos

- ▶ Privilegio sobre datos
 - ▶ Mínimos: partes de tablas
 - ▶ Tabla
 - ▶ Vistas

Discrecional. Privilegios sobre datos.

Dar

```
GRANT privilegios  
ON elemento  
TO [ usuario | rol | PUBLIC ]  
[WITH GRANT OPTION]
```

Quitar

```
REVOKE privilegios  
ON elemento  
FROM [ usuario | rol | PUBLIC ]  
[RESTRICT | CASCADE]
```

Discrecional - Datos

```
GRANT privilegio {, privilegio}  
ON [ tabla | vista ]  
TO [ usuario | rol | PUBLIC ]  
[WITH GRANT OPTION]
```

privilegios

```
INSERT [(columnas)]  
DELETE  
UPDATE [(columnas)]  
SELECT [(columnas)]  
ALL
```

Discrecional. Privilegios sobre datos.

Dar

```
GRANT privilegios  
ON elemento  
TO [ usuario | rol | PUBLIC ]  
[WITH GRANT OPTION]
```

Quitar

```
REVOKE privilegios  
ON elemento  
FROM [ usuario | rol | PUBLIC ]  
[RESTRICT | CASCADE]
```

Discrecional - Datos

```
GRANT privilegio {, privilegio}  
ON [ tabla | vista ]  
TO [ usuario | rol | PUBLIC ]  
[WITH GRANT OPTION]
```

privilegios

```
INSERT [(columnas)]  
DELETE  
UPDATE [(columnas)]  
SELECT [(columnas)]  
ALL
```

```
CREATE TABLE VENDEDOR(  
  v#          CHAR(2),  
  nombre      VARCHAR(20),  
  estatus     NUMBER(2),  
  ciudad      VARCHAR(10));
```

```
CREATE TABLE PARTES(  
  p#          CHAR(2),  
  parte       VARCHAR(20),  
  color       CHAR(10),  
  peso        NUMERIC(5,2),  
  ciudad      VARCHAR(10));
```

```
CREATE TABLE ENVIOS(  
  v#          CHAR(2),  
  p#          CHAR(2),  
  cant        NUMERIC(5));
```

```
GRANT SELECT(p#,parte,peso)  
ON PARTES  
TO JUAN,ANA
```

Discrecional. Privilegios sobre datos.

Dar

```
GRANT privilegios  
ON elemento  
TO [ usuario | rol | PUBLIC ]  
[WITH GRANT OPTION]
```

Quitar

```
REVOKE privilegios  
ON elemento  
FROM [ usuario | rol | PUBLIC ]  
[RESTRICT | CASCADE]
```

Discrecional - Datos

```
GRANT privilegio {, privilegio}  
ON [ tabla | vista ]  
TO [ usuario | rol | PUBLIC ]  
[WITH GRANT OPTION]
```

privilegios

```
INSERT [(columnas)]  
DELETE  
UPDATE [(columnas)]  
SELECT [(columnas)]  
ALL
```

```
CREATE TABLE VENDEDOR(  
  v#          CHAR(2),  
  nombre      VARCHAR(20),  
  estatus     NUMBER(2),  
  ciudad      VARCHAR(10));
```

```
CREATE TABLE PARTES(  
  p#          CHAR(2),  
  parte       VARCHAR(20),  
  color       CHAR(10),  
  peso        NUMERIC(5,2),  
  ciudad      VARCHAR(10));
```

```
CREATE TABLE ENVIOS(  
  v#          CHAR(2),  
  p#          CHAR(2),  
  cant        NUMERIC(5));
```

GRANT

```
ON PARTES  
TO CARLOS;
```

Discrecional. Privilegios sobre datos.

Dar

```
GRANT privilegios  
ON elemento  
TO [ usuario | rol | PUBLIC ]  
[WITH GRANT OPTION]
```

Quitar

```
REVOKE privilegios  
ON elemento  
FROM [ usuario | rol | PUBLIC ]  
[RESTRICT | CASCADE]
```

```
CREATE TABLE VENDEDOR(  
  v#          CHAR(2),  
  nombre      VARCHAR(20),  
  estatus     NUMBER(2),  
  ciudad      VARCHAR(10));
```

```
CREATE TABLE PARTES(  
  p#          CHAR(2),  
  parte       VARCHAR(20),  
  color       CHAR(10),  
  peso        NUMERIC(5,2),  
  ciudad      VARCHAR(10));
```

```
CREATE TABLE ENVIOS(  
  v#          CHAR(2),  
  p#          CHAR(2),  
  cant        NUMERIC(5));
```

Discrecional - Datos

```
GRANT privilegio {, privilegio}  
ON [ tabla | vista ]  
TO [ usuario | rol | PUBLIC ]  
[WITH GRANT OPTION]
```

privilegios

```
INSERT [(columnas)]  
DELETE  
UPDATE [(columnas)]  
SELECT [(columnas)]  
ALL
```

```
CREATE VIEW RESUMEN_ENVIOS AS  
  (SELECT p#, SUM(cant) AS totales  
   FROM ENVIOS  
   GROUP BY p#);
```

```
GRANT SELECT  
ON RESUMEN_ENVIOS  
TO FIDEL;
```


Discrecional. Privilegios sobre datos.

Dar

```
GRANT privilegios  
ON elemento  
TO [ usuario | rol | PUBLIC ]  
[WITH GRANT OPTION]
```

Quitar

```
REVOKE privilegios  
ON elemento  
FROM [ usuario | rol | PUBLIC ]  
[RESTRICT | CASCADE]
```

```
CREATE TABLE VENDEDOR(  
  v#          CHAR(2),  
  nombre      VARCHAR(20),  
  estatus     NUMBER(2),  
  ciudad      VARCHAR(10));
```

```
CREATE TABLE PARTES(  
  p#          CHAR(2),  
  parte       VARCHAR(20),  
  color       CHAR(10),  
  peso        NUMERIC(5,2),  
  ciudad      VARCHAR(10));
```

```
CREATE TABLE ENVIOS(  
  v#          CHAR(2),  
  p#          CHAR(2),  
  cant        NUMERIC(5));
```

Discrecional - Datos

```
GRANT privilegio {, privilegio}  
ON [ tabla | vista ]  
TO [ usuario | rol | PUBLIC ]  
[WITH GRANT OPTION]
```

privilegios

```
INSERT [(columnas)]  
DELETE  
UPDATE [(columnas)]  
SELECT [(columnas)]  
ALL
```

```
CREATE VIEW HORAS_OFICINA AS  
(SELECT *  
  FROM VENDEDORES  
  WHERE '08' <= TO_CHAR(SYSDATE, 'HH24')  
        AND TO_CHAR(SYSDATE, 'HH24') <= '16'  
        AND TO_CHAR(SYSDATE, 'DY') NOT IN ('SAT', 'SUN'));
```

```
GRANT SELECT  
ON HORAS_OFICINA  
TO CONTABILIDAD;
```

Discrecional. Privilegios sobre datos. Mínimos.

Grant

```
CREATE TABLE VENDEDOR(  
  v#          CHAR(2),  
  nombre      VARCHAR(20),  
  estatus     NUMBER(2),  
  ciudad      VARCHAR(10));
```

```
CREATE TABLE PARTES(  
  p#          CHAR(2),  
  parte       VARCHAR(20),  
  color       CHAR(10),  
  peso        NUMERIC(5,2),  
  ciudad      VARCHAR(10));
```

```
CREATE TABLE ENVIOS(  
  v#          CHAR(2),  
  p#          CHAR(2),  
  cant        NUMERIC(5));
```

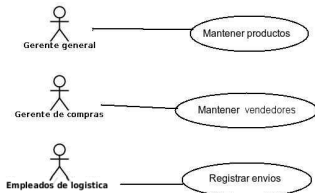
```
UPDATE VENDEDORES  
SET estatus = estatus + 1  
WHERE ((SELECT COUNT(p#) FROM PARTES) =  
       (SELECT COUNT(DISTINCT p#) FROM ENVIOS WHERE VENDEDORES.v#=ENVIOS.v#));
```

Privilegios mínimos

- ▶ ¿Qué se está haciendo?
- ▶ ¿Cuáles privilegios mínimos debe tener 'empleado' para realizar esta actualización?

Discrecional. Privilegios sobre datos. Tablas.

Envíos



```
CREATE TABLE VENDEDOR(  
  v#      CHAR(2),  
  nombre  VARCHAR(20),  
  estatus NUMBER(2),  
  ciudad  VARCHAR(10));
```

```
CREATE TABLE PARTES(  
  p#      CHAR(2),  
  parte   VARCHAR(20),  
  color   CHAR(10),  
  peso    NUMERIC(5,2),  
  ciudad  VARCHAR(10));
```

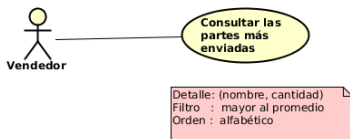
```
CREATE TABLE ENVIOS(  
  v#      CHAR(2),  
  p#      CHAR(2),  
  cant     NUMERIC(5));
```

Privilegios generales

- ¿Qué permisos generales sobre tablas daríamos?

Discrecional. Privilegios sobre datos. Vistas.

Envíos



```
CREATE TABLE VENDEDOR(  
  v#      CHAR(2),  
  nombre  VARCHAR(20),  
  estatus NUMBER(2),  
  ciudad  VARCHAR(10));
```

```
CREATE TABLE PARTES(  
  p#      CHAR(2),  
  parte   VARCHAR(20),  
  color   CHAR(10),  
  peso    NUMERIC(5,2),  
  ciudad  VARCHAR(10));
```

```
CREATE TABLE ENVIOS(  
  v#      CHAR(2),  
  p#      CHAR(2),  
  cant    NUMERIC(5));
```

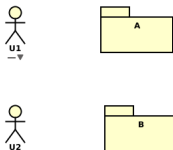
Privilegios generales

- ▶ ¿Qué vista definiríamos?
- ▶ ¿Qué permisos daríamos sobre esta vista?

Control de acceso

Control de acceso

Definir explícitamente permisos de acciones sobre objetos determinados a personas identificadas (ID.Clave)



Objetos

- ▶ Privilegio sobre datos
 - ▶ Mínimos: partes de tablas
 - ▶ Tabla
 - ▶ Vistas
- ▶ Privilegio sobre acciones
 - ▶ Subprogramas
 - ▶ Paquetes

Agenda

Seguridad

Mecanismos

Caso : Control de Acceso - Discrecional

Privilegio sobre datos

Privilegios sobre acciones

Discrecional. Privilegios sobre acciones.

Dar

```
GRANT privilegios  
ON elemento  
TO [ usuario | rol | PUBLIC ]  
[WITH GRANT OPTION]
```

Quitar

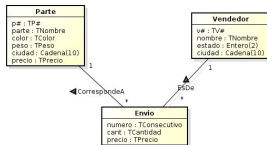
```
REVOKE privilegios  
ON elemento  
FROM [ usuario | rol | PUBLIC ]  
[RESTRICT | CASCADE]
```

Sobre acciones

```
GRANT EXECUTE  
ON [ subprograma | paquete ]  
TO [ usuario | rol | PUBLIC ]  
[WITH GRANT OPTION]
```

Privilegios por paquetes

Envíos

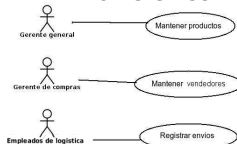


```
CREATE TABLE VENDEDOR(
v#          CHAR(2),
nombre     VARCHAR(20),
estatus    NUMBER(2),
ciudad     VARCHAR(10));

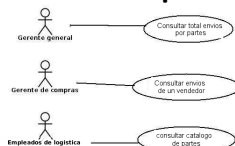
CREATE TABLE PARTES(
p#          CHAR(2),
parte      VARCHAR(20),
color      CHAR(10),
peso       NUMERIC(5,2),
ciudad     VARCHAR(10));

CREATE TABLE ENVIOS(
v#          CHAR(2),
p#          CHAR(2),
cant       NUMERIC(5));
```

Funciones



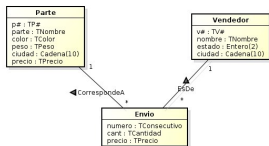
Consultas operativas



- ▶ ¿Cuáles serían los paquetes de componentes (CRUD)?
- ▶ ¿Cuáles serían los paquetes de seguridad (actores)?

Privilegios por paquetes

Envíos

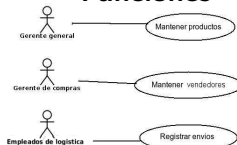


```
CREATE TABLE VENDEDOR(  
  v# CHAR(2),  
  nombre VARCHAR(20),  
  estatus NUMBER(2),  
  ciudad VARCHAR(10));
```

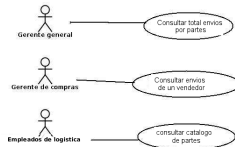
```
CREATE TABLE PARTES(  
  p# CHAR(2),  
  parte VARCHAR(20),  
  color CHAR(10),  
  peso NUMERIC(5,2),  
  ciudad VARCHAR(10));
```

```
CREATE TABLE ENVIOS(  
  v# CHAR(2),  
  p# CHAR(2),  
  cant NUMERIC(5));
```

Funciones



Consultas operativas



► ¿Cuáles serían los paquetes de componentes (CRUD)?

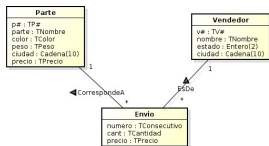
► ¿Cuáles serían los subprogramas de cada paquete?

Los únicos datos a modificar son el estado en vendedor y el precio en parte. Las partes no se pueden eliminar.

► ¿Cuáles serían los paquetes de seguridad (actores)?

Privilegios por paquetes

Envíos

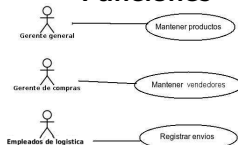


```
CREATE TABLE VENDEDOR(  
  v#          CHAR(2),  
  nombre     VARCHAR(20),  
  estatus    NUMBER(2),  
  ciudad     VARCHAR(10));
```

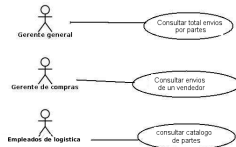
```
CREATE TABLE PARTES(
  p#          CHAR(2),
  parte       VARCHAR(20),
  color       CHAR(10),
  peso        NUMERIC(5,2),
  ciudad      VARCHAR(10));
```

```
CREATE TABLE ENVIOS(
  v#      CHAR(2),
  p#      CHAR(2),
  cant    NUMERIC(5));
```

Funciones



Consultas operativas



- ▶ ¿Cuáles serían los paquetes de componentes (CRUD)?
- ▶ ¿Cuáles serían los paquetes de seguridad (actores)?
 - ▶ ¿Cuáles serían los subprogramas de cada paquete?
 - ▶ ¿Cómo quedarían los permisos?