第4章 数据库的安全性

沈明玉

合肥工业大学 Hefei University of Technology 计算机与信息学院

第4章 数据库的安全性

■ **主要内容：**

4.1 数据库安全性概述

4.2 数据库安全性控制

4.3 视图机制

4.4 安全审计

第4章 数据库的安全性

4.1 数据库安全性概述

数据库安全性：是指保护数据库以防止非法用户的越权使用、窃取、更改或破坏数据。

■ **数据库的安全性可以划分为三个层次：**

✓ 网络系统的安全

✓ 操作系统的安全

✓ 数据库管理系统的安全

4.1 数据库安全性概述

◆ **安全标准**

✓ TCSEC：可信计算机系统评估准则，美国国防部

✓ ITSEC：信息技术安全评估准则，欧洲

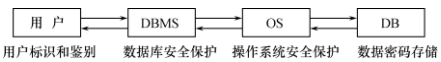
✓ CTCPSEC：加拿大可信计算机产品评估准则，加拿大

✓ CC：通用准则，ISO（我国采用）

第4章 数据库的安全性

4.2 数据库安全性控制

◆ 数据库系统安全模型



4.2 数据库安全性控制

◆ **数据库安全性控制的常用方法**

✓**用户标识和鉴定**

✓**访问控制**

✓**视图**

✓**审计**

✓**数据加密**

4.2 数据库安全性控制

4.2.1 用户标识与鉴别

用户标识：每个合法用户均被赋予一个身份标识。
身份鉴别：鉴别用户的合法身份 。

口身份鉴别方法

- ✓静态口令鉴别
- ✓动态口令鉴别
- ✓生物特征鉴别
- ✓智能卡鉴别

4.2 数据库安全性控制

4.2.2 存取控制

◆访问控制机制的组成

- 定义用户权限
- 合法权限检查

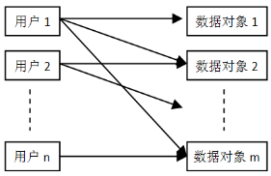
◆常用访问控制方法

- 自主访问控制 DAC：C2级、灵活
- 强制访问控制 MAC：B1级、严格

4.2 数据库安全性控制

4.2.3 自主访问控制方法 DAC

- ✓ 主体与客体直接关联；
- ✓ 主体的权限需要授权；
- ✓ 具有授权资格的用户均可实现授权。



4.2 数据库安全性控制

■ 关系数据库系统中的存取权限

对象类型	对象	操作类型
数据库模式	模式	CREATE SCHEMA
	基本表	CREATE TABLE，ALTER TABLE
	视图	CREATE VIEW
	索引	CREATE INDEX
数据	基本表和视图	SELECT，INSERT，UPDATE，DELETE，REFERENCES，ALL PRIVILEGES
	属性列	SELECT，INSERT，UPDATE，REFERENCES
		ALL PRIVILEGES

4.2 数据库安全性控制

4.2.4 授权：授予与收回

- 通过 SQL 的 GRANT 语句和 REVOKE 语句实现；
- 用户权限组成：数据对象、操作类型；
- 定义用户访问权限：定义用户可以在哪些数据库对象上进行哪些类型的操作。

■ 授权GRANT语句

GRANT < 权限 > [,< 权限>] ...
[ON < 对象类型 > < 对象名 >]
TO < 用户 > [, < 用户 >] …
[WITH GRANT OPTION];

4.2 数据库安全性控制

■ 谁可以发出GRANT

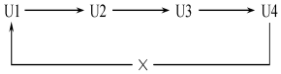
DBA、对象创建者（Owner）、拥有该权限的用户。

■ 可接受权限的用户

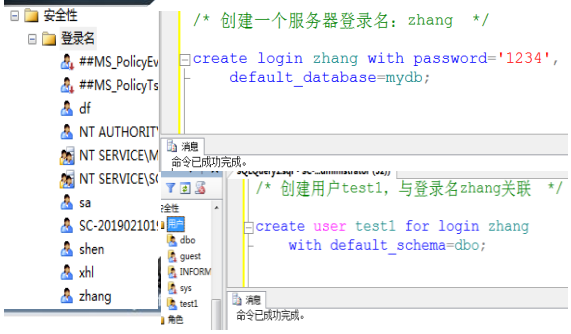
一个或多个具体用户、PUBLIC（全体用户）。

■ WITH GRANT OPTION子句

■ 不允许循环授权：



4.2 数据库安全性控制



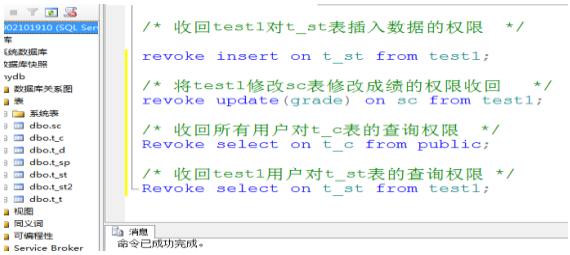
4.2 数据库安全性控制

```
/* 将t_st表的查询权限授予用户test1 */
Grant select on t_st to test1;

/* 将t_c的查询权限授予public  */
Grant select on t_c to public;

/* 将sc表修改成绩的权限授予test1   */
grant update(grade) on sc to test1;

/* 将t_st插入数据的权限授予test1  */
grant insert on t_st to test1;
```

4.2 数据库安全性控制

■ **撤销（收回）权限** ***REVOKE***

REVOKE <权限>[,<权限>] … [ON <对象类型><对象名>]
FROM <用户>[,<用户>] … ;



4.2 数据库安全性控制

■ **创建用户时初始角色授权**

CREATE USER <username>

[WITH] [DBA | RESOURCE | CONNECT]

拥有的权限	可否执行的操作			
	CREATE USER	CREATE SCHEMA	CREATE TABLE	登录数据库 执行数据查询和操纵
DBA	可以	可以	可以	可以
RESOURCE	不可以	不可以	可以	可以
CONNECT	不可以	不可以	不可以	可以，但必须拥有相应权限

4.2 数据库安全性控制

4.2.5 数据库角色

- ✓ **数据库角色：被命名的一组与数据库操作相关的权限。**
- ✓ **角色是权限的集合。**
- ✓ **可以为一组具有相同权限的用户创建一个角色。**
- ✓ **简化授权的过程。**
- ✓ **角色的创建： CREATE ROLE <角色名>;**
- ✓ **给角色授权：**

***GRANT** <权限>[，<权限>]… **ON** <对象类型> 对象名*
***TO** <角色>[，<角色>]…;*

4.2 数据库安全性控制

● 将一个角色授予其他的角色或用户：

GRANT <角色1>[，<角色2>]… TO <角色3>[，<用户1>]…
[WITH ADMIN OPTION]；

● 角色权限的收回：

REVOKE <权限>[，<权限>]… ON <对象类型><对象名>
FROM <角色>[，<角色>]…;

4.2 数据库安全性控制

```
/* 创建角色   */
Create role rtest;

/* 为角色rtest授权   */
Grant select,insert,delete,update on t_st to rtest;
Grant create table to rtest;

/* 将角色rtest分配（授权）给用户test1  */
exec sp_addrolemember 'rtest', 'test1';

/* 删除用户test1  */
Drop user test1;

/* 删除角色 Rtest  */
Drop role rtest;

/* 删除登录名 zhang  */
Drop login zhang;
```

命令已成功完成。

4.2 数据库安全性控制

4.2.6 强制存取控制MAC

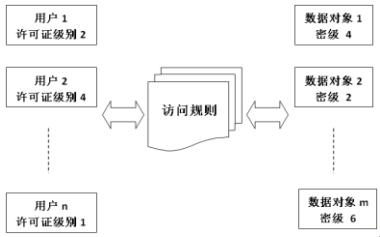
- ✓ 保证更高程度的安全性。
- ✓ 用户不能直接感知或进行控制。
- ✓ 适用于对数据有严格而固定密级分类的部门。
- ✓ 主体 是系统中的活动实体。
  - ➢ DBMS所管理的实际用户
  - ➢ 代表用户的各个进程
- ✓ 客体 是系统中的被动实体，是受主体操纵的。
  - ➢ 文件、基本表、索引、视图。

4.2 数据库安全性控制

- ✓ **敏感度标记（Label）**
  - ● 主体的敏感度标记称为**许可证级别**；
  - ● 客体的敏感度标记称为**密级**；
- ✓ **强制存取控制规则**
  - (1) 仅当主体的许可证级别大于或等于客体的密级时，该主体才能读取相应的客体；
  - (2) 仅当主体的许可证级别等于客体的密级时，该主体才能写相应的客体；

修正规则： 当主体的许可证级别 <= 客体的密级时，主体能写客体。

4.2 数据库安全性控制



第4章 数据库的安全性

4.3 视图机制

◆ 作用：把要保密的数据对无权存取这些数据的用户隐藏起来，对数据提供一定程度的安全保护。

```
CREATE VIEW CS_Student AS
        SELECT * FROM Student WHERE Sdept='CS';
GRANT SELECT ON CS_Student TO 王平；
GRANT ALL PRIVILIGES ON CS_Student TO 张明；
```


4.4 审计

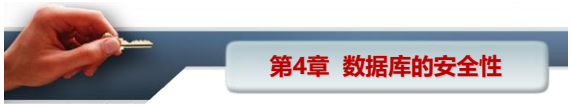
■ **AUDIT和NOAUDIT语句**

AUDIT 语句：设置审计功能
NOAUDIT 语句：取消审计功能

- ● 用户级审计：一般用户使用，对自己的数据对象进行审计。
- ● 系统级审计：DBA设置，监测成功或失败的登录、授权或收回操作，以及其他系统级权限的操作。

- ● **审计设置示例**

```
AUDIT  ALTER，UPDATE ON SC;
NOAUDIT  ALTER，UPDATE ON SC;
```



第4章 数据库的安全性

- **本章作业：**

  P155 习题6