数 据 库 系 统 课 程 教 案

第 11 单元 **学时：** 2

教材内容	第 4 章　数据库安全性
基本知识点	计算机系统安全性问题、数据库的安全性问题、统计数据库的安全性问题、TDI/TCSEC 标准的基本内容、实现数据库安全性控制的常用方法和技术、数据库中自主存取控制方法和强制存取方法、使用 SQL 语言中的 GRANT 和 REVOKE 语句来实现自主存取控制
教学重点	实现数据库安全性控制的常用方法和技术、数据库中自主存取控制方法和强制存取方法、使用 SQL 语言中的 GRANT 和 REVOKE 语句来实现自主存取控制
教学难点	实现数据库安全性控制的常用方法和技术、数据库中自主存取控制方法和强制存取方法、使用 SQL 语言中的 GRANT 和 REVOKE 语句来实现自主存取控制
要求掌握内容	实现数据库安全性控制的常用方法和技术、数据库中自主存取控制方法和强制存取方法、使用 SQL 语言中的 GRANT 和 REVOKE 语句来实现自主存取控制
教学思路，采用的教学方法和辅助手段，板书设计，重点如何突出，难点如何解决，师生互动等	**教学思路：** **一、复习旧课，巩固上次授课主要内容** 1、SQL 查询语句的格式是什么？ 2、视图的概念及其与基本表的区别。 **二、导入新课，明确本次授课的目的与要求** 1、了解 TDI/TCSEC 标准的基本内容。 2、理解数据库安全性与计算机安全性的概念。 3、掌握数据库的安全性控制。 4、掌握统计数据库的安全性规则。 **三、讲解本次授课的具体内容** **教学方法：**整合教学内容，强化基础训练；努力营造生动活泼的课堂气氛，搭建师生间良好的沟通渠道；采用多媒体教学与传统的板书设计相结合的方式，教学手段灵活多变。 **辅助手段：**通过 PPT 幻灯片演示结合板书设计加以阐述。
本章思考题和作业	P300 第 6、8、11 题
主要教材参考资料	1.《数据库系统概论》，萨师煊，王珊，高等教育出版社，2014.9 2.《数据库系统概论学习指导与习题解答》，王珊，张俊，高等教育出版社，2015.7
备 注	

本次授课具体内容

第 4 章　数据库安全性

4.1　计算机安全性概论

㈠数据库的安全性：指保护数据库，防止不合法的使用所造成的数据泄露、更改或破坏。它与计算机系统的安全性紧密联系。

㈡计算机系统的安全性：为计算机系统建立和采取的各种安全保护措施，以保护计算机系统中的硬件、软件及数据，防止因偶然或恶意的原因使系统遭到破坏，数据遭到更改或泄露等。

㈢可信计算机系统评测标准

1、1985 年美国国防部 DoD 颁布的《 DoD 可信计算机系统评估标准》（Trusted Computer System Evaluation Criteria， 简称 TCSEC，桔皮书）。

2、1991 年美国国家计算机安全中心 NCSC 颁布的《可信计算机系统评估标准关于可信数据库系统的解释》(Trusted Database Interpretation，简称 TDI，紫皮书)。

3、TDI/TCSEC 标准从安全策略、责任、保证和文档四个方面将系统划分四组七个等级的安全性级别：

⑴D 级：最小保护，最低级别，如 DOS。

⑵C1 级：自主安全保护，能实现对用户和数据的分离，进行自主存取控制(DAC)。

⑶C2 级：受控的存取保护，是安全产品的最低档次，将 C1 级的 DAC 进一步细化，以个人身份注册负责，实施审计和资源隔离。如 Oracle9i、SQL Server2000。

⑷B1 级：标记安全保护，对数据加以标记并对标记的主体和客体实施强制存取控制(MAC)及审计等。

⑸B2 级(目前尚无此类数据库产品)：结构化保护，建立形式化的安全策略模型并对系统内所有主体和客体实施 DAC 和 MAC。

⑹B3 级：安全域，提供访问监控、审计跟踪和数据恢复。

⑺A1 级：验证设计，在 B3 基础上提供系统的形式化设计和验证

B2 以上系统还处于理论研究阶段，应用多限于某些特殊部门。

4.2　数据库安全性控制

㈠用户标识和鉴别：最外层安全保护措施。系统提供一定的方式让用户标识自己的名字或身份，系统内部记录着所有合法用户的标识，每次用户要求进入系统时，由系统核对用户提供的身份标识，通过鉴定后才提供机器使用权。用户标识和鉴定可以重复多次。常用方法是标识(用户名)+鉴别(口令)。

㈡存取控制：数据库安全最重要的一点就是确保只授权给有资格的用户访问数据库的权限，同时令所有未被授权的人员无法接近数据，这主要通过数据库系统的存取控制机制实现。

本次授课具体内容(续)

1、定义用户权限，并将用户权限登记数据字典中，形成安全规则

2、合法权限检查，当用户发出存取数据库的操作请求后，DBMS 查找数据字典，根据安全规则进行合法权限检查，若用户的操作请求超出了定义的权限，系统将拒绝执行此操作。

用户权限定义和合法权限检查机制一起组成了 DBMS 的安全子系统。

3、自主存取控制(DAC)：大型 DBMS 一般都支持 C2 级的 DAC。在 DAC 方法中，用户对于不同的数据对象有不同的存取权限，不同的用户对同一对象也有不同的权限，而且用户还可将其拥有的存取权限转授给其他用户，因此 DAC 非常灵活。SQL 标准支持 DAC，主要通过 Grant(授权)和 Revoke(收权)语句来实现。⑴衡量 DAC 授权机制精巧程度的两个尺度

①授权粒度：指可以定义的数据对象的范围，是衡量授权机制是否灵活的一个重要指标。授权定义中数据对象(数据库、表、属性列、行)的粒度越细，授权子系统就越灵活。

②是否提供与数据值有关的授权，即授权是否依赖于数据对象的值。如定义用户王平只能检索计算机系学生信息，这就要求系统能支持存取谓词。存取谓词可以引用系统变量，如终端设备号、系统时钟等，实现与时间地点有关的存取权限，这样规定用户只能在某段时间、某台终端上存取有关数据，如规定“某教师只能在每年 1 月份和 7 月份星期一至星期五上午 8 点到下午 5 点处理学生成绩数据”。

⑵DAC 的优点：由用户定义存取权限，DBMS 检查存取权限，能够通过授权机制有效地控制其他用户对敏感数据的存取。

⑶DAC 的缺点：可能存在数据的“无意泄露”。该机制仅通过对数据的存取权限来进行安全控制，而数据本身无安全性标记。

4、强制存取控制(MAC)：有些 DBMS 支持 B1 级的 MAC。在 MAC 中，每一个数据对象被标以一定的密级，每一个用户也被授予某一个级别的许可证。 MAC 中 DBMS 所管理的全部实体被分为主体和客体两大类。主体是系统中的活动实体，包括用户及其进程；客体是系统中受主体控制的被动实体，包括文件、基本表、索引、视图等。 DBMS 对主体和客体的实例分别指派一个敏感度标记 label(分为绝密、机密、可信、公开等级别)，主体 label 称为许可证级别，实体 label 称为密级。

⑴当某实体以标记注册后，系统要求他对任何客体的存取遵守：

①仅当主体许可证级别³客体密级时，主体才能读取相应客体。

②仅当主体许可证级别=客体密级时，主体才能写相应客体。

这两个规则的共同点是均禁止拥有高许可证级别的主体更新低密级的数据对象，从而防止了敏感数据的泄漏。因为用户可为写入的数据对象赋予高于自己许可证级别的密级，一旦数据被写入，该用户自己也不能再读该数据对象了。⑵MAC 的特点：MAC 是对数据本身进行密级标记，无论数据如何复制，标记与数据是一个不可分的整体，只有符合密级标记要求的用户才可以操纵数据，从而提供了更高级别的安全性。

DAC 与 MAC 共同构成 DBMS 的安全机制：原因是较高安全性级别提供的安全保护要包含较低级别的所有保护。实现方法是先进行 DAC 检查，通过 DAC 检查的数据对象再由系统进行 MAC 检查，只有通过 MAC 检查的数据对象方可存取。

4.3 视图机制：把要保密数据对无权存取的用户隐藏起来，提供一定程度的数据保护，间接实现了支持存取谓词的用户权限定义

4.4 审计：是 DBMS 达到 C2 级以上安全级别必不可少的一项指标。它将用户对数据库的所有操作自动记录下来放入审计日志。DBA 利用审计日志，可以重现导致数据库现有状况的一系列事件，找出非法存取数据的人、时间和内容等。

4.5 数据加密：将原始数据(明文)通过一定算法变为密文的过程，较著名的有 DES 加密算法。

4.6 其他安全性保护：统计数据库的安全性

统计数据库允许用户查询聚集类型的信息(如合计、平均值等)，不允许查询单个记录的信息。但统计数据库可能存在隐蔽的信息通道，从中可以导出不合法的信息。

例 1：本公司共有多少名女程序员？女程序员的工资总额是多少?

若第 1 查询的结果是 1，则第 2 查询的结果就是该女程序员的工资，安全机制失效。为此规定规则 1：任何查询至少涉及 N 个以上的记录(N 足够大)。

例 2:用户 A 和其他 N 个程序员的工资总额是多少？用户 B 和其他 N 个程序员的工资总额是多少？

设第 1 查询结果是 X，第 2 查询结果是 Y，由于用户 A 知道自己的工资为 Z，则他可以计算出用户 B 的工资为 Y-(X-Z)，原因是两个查询之间有很多重复的数据项，安全机制失效。为此规定规则 2：任意两个查询的相交数据项 £ M 个，则用户 A 要计算用户 B 的工资至少要进行 1+(N-2)/M 次查询。为此规定规则 3：任一用户的查询次数不能超过 1+(N-2)/M。

附加：

SQL Server 的安全性技术

㈠SQL Server 的身份验证模式：SQL Server 提供了三种身份验证模式或安全管理模式，即标准模式、集成模式和混合模式。在 Windows NT 或 Windows 2000 上使用集成模式或混合模式,在 Windows98 上使用标准模式。

1、集成模式(也称 Windows 模式)：用户通过 Windows 身份验证后则自动进行 SQL Server 身份验证。即当用户通过 Windows 用户帐户进行连接时，SQL Server 通过回叫 Windows 以获得信息，重新验证帐户名和密码。 SQL Server 的连接分为信任连接和非信任连接，集成模式使用信任连接，即用户只要登录到 Windows，就可以通过信任连接直接连接到 SQL Server。

2、混合模式：适用于非信任连接，用户登录 Windows 后，要以同一帐户连接 SQL Server 时， SQL Server 通过检查该帐户是否已设置为 SQL Server 登录帐户，如果 SQL Server 未设置为登录帐户，则身份验证将失败，并返回错误信息。

㈡SQL Server 的用户管理和角色管理

1、用户的分类：

⑴系统管理员用户：在一个 DBMS 的运行实例上拥有一切权限的用户，负责整个系统的管理。在一个运行实例上可以建立多个数据库，系统管理员用户则在所有数据库上拥有所有权限。

⑵数据库管理员用户:在某一数据库上拥有一切权限的用户,负责一个具体数据库的建立和管理,也称为数据库属主。

⑶数据库对象用户：可以建立数据库对象(如表、视图等)的用户，在自己建立的数据库对象上拥有全部操作权限，也称为数据库对象属主。

⑷数据库访问用户：可对被授权数据库对象进行查询、更新操作一个 DBMS 在安装时至少有一个系统管理员用户，SQL Server 安装时默认的系统管理员用户是 sa。

登录用户和数据库用户：一个用户需要首先是一个数据库系统的登录用户，然后才可以访问某一个具体的数据库，所以有登录用户(login user)和数据库用户(database user)两个概念。一个登录用户可以是多个数据库的用户,登录用户由系统管理员管理,而数据库用户可以由数据库管理员管理。

2、用户管理

⑴登录用户的管理：主要包括建立新的登录用户、删除登录用户、修改登录密码等。其中前二项工作必须由系统管理员完成。

①建立：执行系统存储过程 sp_addlogin：sp_addlogin login_id [,passwd] [,defdb] [……]

其中， login_id 为登录名；passwd 为登录密码，默认为 NULL；defdb 为登录数据库，默认为 master。②删除：执行系统存储过程 sp_droplogin，sp_droplogin login_id

③修改登录密码：执行系统存储过程 sp_password，sp_password old_password new_password

登录密码在数据库中是以加密形式存储的，任何人都不可以查询登录密码(包括系统管理员)。用户的登录密码一旦遗忘，只能由系统管理员为该用户重设登录密码，这时系统管理员可将旧密码 old_password 指定为 NULL。作为系统管理员的一旦忘记或泄露了自身密码，就只能重装 SQL Server。

⑵数据库用户的管理：由数据库管理员或系统管理员完成。

①授权某个登录用户为数据库用户：执行 sp_grantdbaccess 系统存储过程，格式为：

sp_grantdbaccess login [,name_in_db]

其中，login 为登录名；name_in_db 为数据库中的用户名，默认为 NULL

②从数据库中删除某个用户：执行 sp_revokedbaccess 系统存储过程， sp_revokedbaccess name

其中，name 为数据库用户名。

3、角色管理：单独管理数据库中每一用户非常繁杂，若将具有相同操作权限的用户组织成组则可以简化对用户的管理工作。具有相同权限的用户也可以说是担当相同的角色，组和角色表面上是两个概念，实际上是一回事。从用户组解释，是先定义用户组，然后按用户组管理权限；而从角色解释，是先定义权限，然后为用户指定角色。

⑴定义角色：DBA 使用 sp_addrole 系统存储过程，格式是：sp_addrole role [,owner]

其中， role 是新角色的名称；owner 为新角色的拥有者，他必须是当前数据库中的某个用户或角色，默认值为 dbo。

⑵为用户指定角色：使用 sp_addrolemember 系统存储过程：sp_addrolemember role user_account

⑶取消用户的角色：使用 sp_droprolemember 系统存储过程：sp_droprolemember role user_account

⑷删除角色：使用 sp_droprole 系统存储过程：sp_droprole role

SQL Server 的预定义角色

⑴public 角色：是一个特殊的数据库角色，每个数据库用户都是该角色的成员。public 角色的特点是：public 角色自动获得数据库中所有用户的所有默认权限；不需要、也无法将用户指派给 public 角色，因为默认情况下所有用户都属于该角色；每个数据库都有 public 角色；不可以删除 public 角色。

⑵系统预定义角色：sysadmin、serveradmin、setupadmin、securityadmin、processadmin、dbcreator、diskadmin、bulkadmin，可以使用系统存储过程 sp_helpsrvrole 获得一个 SQL Server 实例上各种系统管理员角色的列表和描述；使用系统存储过程 sp_srvrolepermission 得到各系统管理员角色的特定权限。

⑶数据库预定义角色：db_owner、db_accessadmin、db_ securityadmin 、db_ddladmin、db_backupopeator、db_ datareader、db_ datawriter、db_ denydatareader、db_ denydatawriter，可以使用系统存储过程 sp_helpdbfixedrole 获得数据库上各种预定义角色的列表和描述；使用 sp_dbfixedrolepermission 系统存储过程得到每种数据库预定义角色的特定权限。

4、权限管理：权限就是用户对数据库及其对象的使用权利。当用户连接到数据库后，他们可以执行的操作由其所拥有的权限确定。若用户没有默认权限，应从其他用户中得到相应的授权。 ⑴授予语句权限：要创建数据库的对象，必须有执行相应语句的权限。权限有：BACKUP DATABASE、BACKUP LOG、CREATE DATABASE、CREATE DEFAULT、CREATE FUNCTION、CREATE PROCEDURE、CREATE RULE、CREATE TABLE、CREATE VIEW。授予语句权限的命令格式是：GRANT {ALL|statement_list} TO name_list

其中， statement_list 是给出授权的语句列表，只有系统管理员才能使用 ALL 及 CREATE DATABASE 选项；name_list 可以是数据库用户名、用户组或角色，但他们必须已经存在。

⑵授予对象权限：SELECT、INSERT、UPDATE 和 DELETE 权限可以作用于整个表或视图上；SELECT 和 UPDATE 权限可以作用于表或视图的某个列上;INSERT 和 DELETE 权限可以作用于整个行上;EXECUTE 权限作用于存储过程和函数上。

授予对象权限的命令格式是：

GRANT {ALL|permission_list} [ON {table|view [(column_list)]

|ON stored_procedure|ON user_defined_function]

TO name_list [WITH GRANT OPTION] [AS {group|role}]

其中，ALL 说明将指定对象的所有操作权限都授予指定的用户，只有 SA、DBA 和数据库对象所有

者才可以使用此选项；permission_list 是权限列表；view 是当前数据库中被授予权限的视图名；column_list 是当前数据库中被授予权限的列名列表;stored_procedure 是当前数据库中被授予权限的存储过程名;user_defined_function 是当前数据库中被授予权限的用户自定义函数名;WITH GRANT OPTION 说明被授权用户可以将指定的对象权限授予其他用户，该语句仅对对象权限有效，对语句权限无效。在 SQL Server 中，可以使用系统存储过程 sp_helprotect 查询授权情况。

⑶收回语句权限：REVOKE {ALL|statement_list} FROM name_list

⑷收回对象权限：

REVOKE [GRANT OPTION FOR] {ALL|permission_list} [ON {table|view [(column_list)]
|ON stored_procedure|ON user_defined_function]FROM name_list [CASCADE] [AS {group|role}]

其中，GRANT OPTION FOR 表示只收回 WITH GRANT OPTION 权限；CASCADE 表示级联收回由于 WITH GRANT OPTION 授予的所有权限。⑸禁止语句权限：DENY {ALL|statement_list} FROM name_list

⑹禁止对象权限：DENY {ALL|permission_list} [ON {table|view [(column_list)]
|ON stored_procedure |ON user_defined_function] TO name_list [CASCADE]

5、用户定义的安全性措施：使用触发器可以定义特殊的用户级安全性措施。如规定只能在周一至周五可以对仓库表进行更新(包括 INSERT、DELETE 和 UPDATE)操作，则触发器定义如下：

```
CREATE TRIGGER  secure_wh ON 仓库 FOR INSERT,DELETE,UPDATE
AS IF DATENAME(weekday,getdate( ))='星期六'OR DATENAME(weekday,getdate( ))='星期日'
          OR (convert(INT,DATENAME(hour,getdate( ))) NOT BETWEEN 8 AND 18)
   BEGIN
      RAISERROR('只允许在工作时间内操作!',16,1)
      ROLLBACK TRANSACTION
   END;
```

当建立触发器后，如果对仓库表的操作发生在星期六或星期日，或上午 9 时之前，或下午 18 时之后，则会给出错误提示信息“只允许在工作时间内操作!”，并撤消操作。

本次授课小结

本次授课讲述了 TDI/TCSEC 标准的基本内容、数据库安全性与计算机安全性的概念、数据库的安全性控制、统计数据库的安全性规则、SQL Server 的安全性技术学生课后复习时应着重于其中的第 3、4、5 点内容，为进一步学习后续章节打好基础。

实验	综合实验一