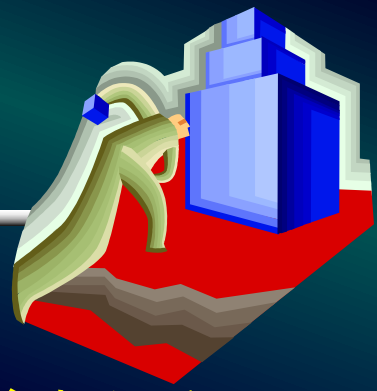
UNIT 9 数据库安全

本讲主要目标



学完本讲后，你应该能够了解：

1、数据库的安全性措施是层层设置的，包括用户标识和鉴定、DBMS存取控制、视图机制、数据加密和审计追踪等；

2、DBMS的安全子系统由用户权限定义和合法权检查机制组成，其中授权规则放在数据字典中，合法权的检查也只检查数据字典；

3、两种存取控制方法：自主存取控制方法的灵活性和强制存取控制方法的严格性；

4、审计日志的作用及与恢复机制中日志的区别；

5、统计数据库中可能存在着隐蔽的信息通道。

本讲主要内容

一. 数据库系统安全概述

二. 用户标识与认证

三. DBMS的存取控制子系统

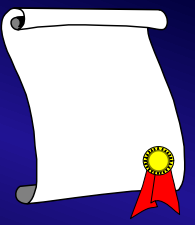
四. 授权与回收

五. 角色

六. 视图与权限

七. 审计

八. 数据加密与统计数据库的安全



一、数据库系统安全概述

（参见教材P140-141）

1、数据库安全性定义

—— 对数据库进行安全控制，保护数据库以防止不合法的使用所造成的数据泄露、更改或破坏。

不合法的使用的操作包括了对数据库的查询与修改。因此，对数据库的任何操作都要进行安全性检查。

数据库的安全性与计算机系统的安全性，包括计算机硬件、操作系统、网络系统等的安全性是紧密联系、相互支持的。

一、数据库系统安全概述

2、安全标准

◆TCSEC（trusted computer system evaluation criteria，可信计算机系统评估标准）

- ✓ TCSEC标准是计算机系统安全评估的第一个正式标准，具有划时代的意义。
- ✓ 该准则于1970年由美国国防科学委员会提出，并于1985年12月由美国国防部公布。
- ✓ TCSEC最初只是军用标准，后来延至民用领域。
- ✓ TCSEC将计算机系统的安全划分为4组、7个级别。
- ✓ 也被称为桔皮书。

一、数据库系统安全概述

- TDI（trusted database interpretation，可信数据库系统解释）
  - ✓ 1991年4月美国NCSC（国家计算机安全中心）颁布了《可信计算机系统评估标准关于可信数据库系统的解释》（TDI），将TCSEC扩展到数据库管理系统
  - ✓ 定义了DBMS的设计和实现需满足和用以进行安全性级别评估的标准
  - ✓ 此标准将数据安全划分为四组七级
  - ✓ 又称紫皮书

一、数据库系统安全概述

- TCSEC(TDI)标准：
  1. D级标准 —— 无安全保护的系统
  2. C1级标准 — 具有自主访问控制
  3. C2级标准 — 满足C1级标准，并有审计功能
  4. B1级标准 — 满足C2级标准，并有强制访问控制
  5. B2级标准 — 满足B1级标准，并解决隐蔽通道问题和具有数据库安全的形式化
  6. B3级标准 — 满足B2级标准，并具有访问监控器
  7. A级标准 — 满足B3级标准，并具有较高的形式化

一、数据库系统安全概述

- TCSEC(TDI)标准的国内和国际实现现状

1. D级标准
2. C1级标准 – 目前国内使用的大都符合该标准
3. C2级标准 – 目前国内使用的一部分符合该标准
4. B1级标准 – 目前国内使用的系统基本不符合该标准,在国际上有部分系统符合此标准
5. B2级标准 – 国内外经认证符合此类标准的系统很少,主要难点是数据库安全的形式化表示困难
6. B3级标准 – 国内外均尚无符合此类标准的系统
7. A级标准 – 目前尚无法实现,仅是一种理想化等级

一、数据库系统安全概述

◆我国1999年颁布了”计算机信息系统评估准则”，将数据安全划分为五个级别

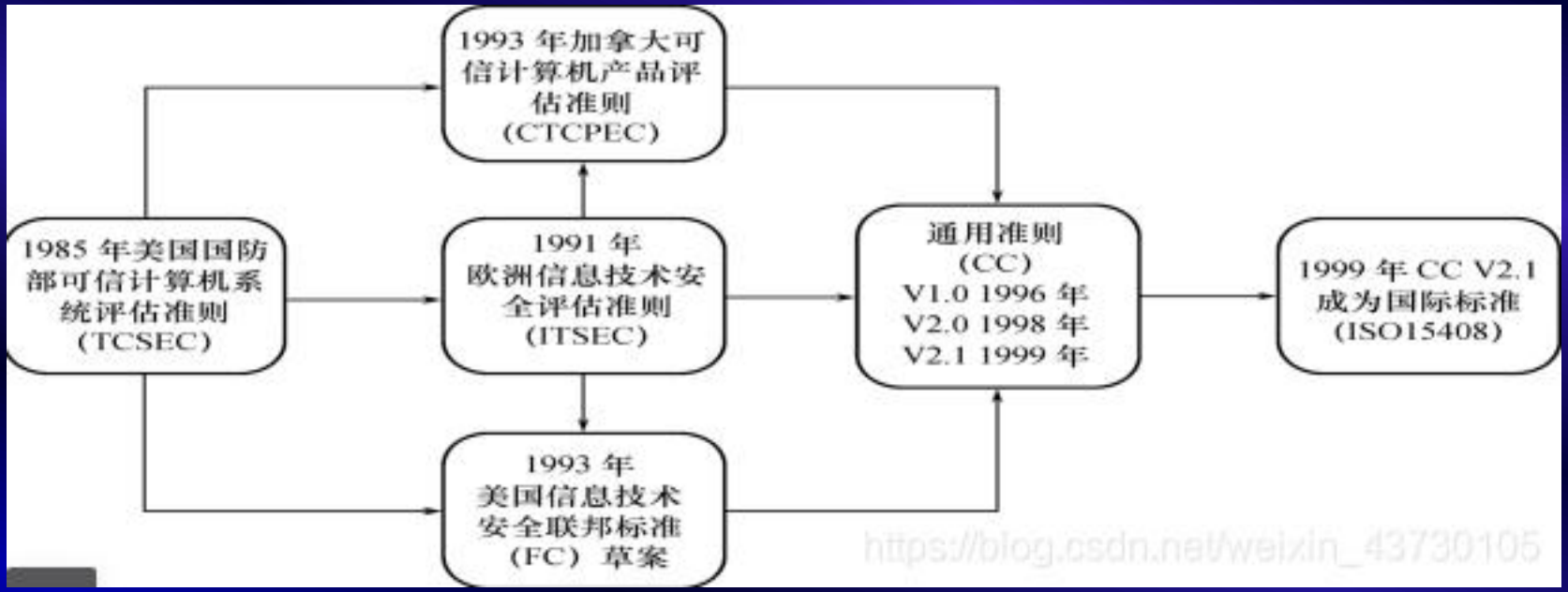
TCSEC标准	我国标准
D级标准	无
C1级标准	第1级：用户自主保护级
C2级标准	第2级：系统审计保护级
B1级标准	第3级：安全标记保护级
B2级标准	第4级：结构化保护级
B3级标准	第5级：访问验证保护级
A级标准	

一、数据库系统安全概述

- 不同国家建立在 TCSEC 概念上的评估准则：
  - ✓ 欧洲的信息技术安全评估准则（ITSEC）
  - ✓ 加拿大的可信计算机产品评估准则（CTCPEC）
  - ✓ 美国的信息技术安全联邦标准（FC）
- 1993 年，CTCPEC、FC、TCSEC 和 ITSEC 联合行动，解决原标准中概念和技术上的差异，将各自独立的准则集合成一组单一的、能被广泛使用的 IT 安全准则，这一行动被称为通用准则（Common Criteria，CC）项目。
  - ✓ 1999 年 CC V2.1版被 ISO 采用为国际标准，2001 年 CC V2.1 版被我国采用为国家标准。
  - ✓ 目前 CC 已基本取代了 TCSEC，成为评估信息产品安全性的主要标准。

一、数据库系统安全概述

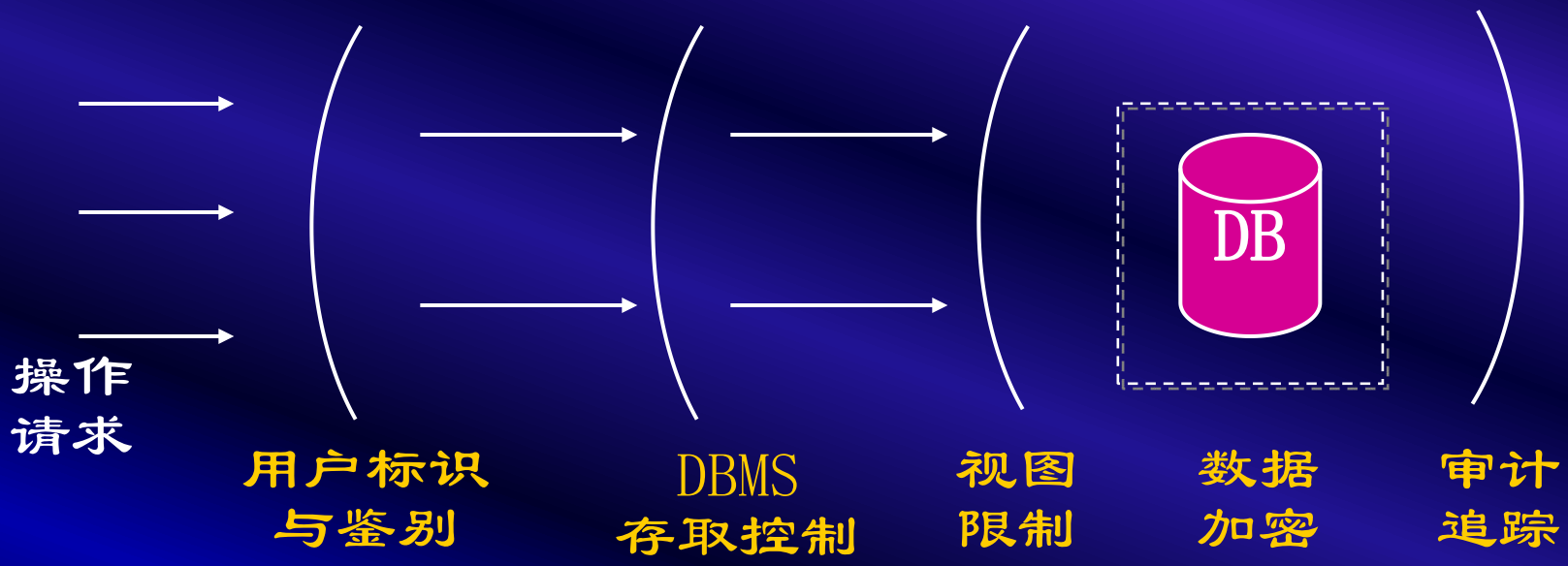
安全标准



一、数据库系统安全概述

3、数据库系统的安全模型

层层设防：



一、数据库系统安全概述

数据库安全性控制的常用方法：

- ✓ 用户标识和鉴定(用户身份认证)
- ✓ 存取控制
- ✓ 视图
- ✓ 审计
- ✓ 数据加密

本讲主要内容

一. 数据库系统安全概述

二. 用户标识与认证

三. DBMS的存取控制子系统

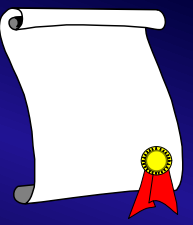
四. 授权与回收

五. 角色

六. 视图与权限

七. 审计

八. 数据加密与统计数据库的安全



二、用户标识与鉴别

（参见教材P140-141）

几个概念：

用户身份标识和鉴别（Identification & Authentication）：是一种用来判断用户身份是否属实的机制，是系统提供的最外层安全保护措施.

标识Identification：身份声明

用户标识：由用户名和用户标识号组成（用户标识号在系统整个生命周期内唯一）

鉴别Authentication：检验身份声明的有效性

二、用户标识与鉴别

用户身份鉴别的方法：

① 静态口令鉴别

- 静态口令一般由用户自己设定，这些口令是静态不变的

② 动态口令鉴别

- 口令是动态变化的，每次鉴别时均需使用动态产生的新口令登录数据库管理系统，即采用一次一密的方法

③ 生物特征鉴别

- 通过生物特征进行认证的技术，生物特征如指纹、虹膜和掌纹等

④ 智能卡鉴别

- 智能卡是一种不可复制的硬件，内置集成电路的芯片，具有硬件加密功能

二、用户标识与鉴别

1、数据库系统用户认证

- 系统管理员通过创建用户帐号来管理用户对计算机系统资源的访问. 给定每一位用户一个唯一的标识符和口令, 操作系统通过它来判断用户的身份.
- 上述过程实现了对计算机系统的授权使用, 但并不一定授权了对DBMS或相关应用程序的访问. 授予用户访问DBMS的权利还需要经过一个另外的类似过程. 这个过程通常是由DBMS的DBA来完成的, DBA负责为DBMS用户建立用户帐号和口令.

二、用户标识与鉴别

2、用户认证的位置和方法

- 一些DBMS维护一张有效用户标识符和相关口令清单，这张清单与操作系统的用户清单没有直接联系.
- 但是，其它一些DBMS的清单是基于当前用户的操作系统用户清单建立的，这样就能够避免用户使用一种身份登录操作系统，却使用另一种身份登录DBMS .

二、用户标识与鉴别

3、用户认证方式

在创建或更改用户时,DBA需要决定用户的身份认证方式:

- 通过操作系统认证
- 通过数据库认证
- 通过网络服务认证

本讲主要内容

一. 数据库系统安全概述

二. 用户标识与认证

三. DBMS的存取控制子系统

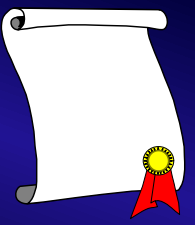
四. 授权与回收

五. 角色

六. 视图与权限

七. 审计

八. 数据加密与统计数据库的安全



三、DBMS的存取控制子系统

（参见教材P141-144）

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。这主要通过数据库的存取控制机制实现。

常见存取控制机制包括：自主存取控制机制和强制存取控制机制两类

三、DBMS的存取控制子系统

1、特权(Privilege)

◆一旦授予用户访问DBMS的权利，各种不同的特权就和用户自动联系在一起. 例如，特权可能是访问或创建数据库对象，也可能是运行各种DBMS工具软件，用户需要这些特权来完成工作所要求的任务.

◆所有权与特权

DBMS中的某些对象是由DBMS（通常是以超级用户的形式，例如DBA）本身拥有的. 对象的所有权给予了拥有者有关该对象的所有特权.

◆对象的创建者拥有对象，并能够分配对象的相应特权

三、DBMS的存取控制子系统

◆“最小特权策略”

只有一个人工作需要的才是他应该知道的

◆两种原则

- “封闭系统”原则：虽然给予用户访问DBMS的权利，但当访问特定数据库对象时还需要其他的授权，由DBA或对象的拥有者来执行这个授权过程
- “开放系统”原则：允许用户对数据库的所有对象都拥有访问权限，这时，访问控制可通过显式地去除用户的特权来实现

三、DBMS的存取控制子系统

2、授权(Authorization)

◆**授权**是授予一个主体权利或特权，使其能够实现对系统或者系统对象的合法访问.

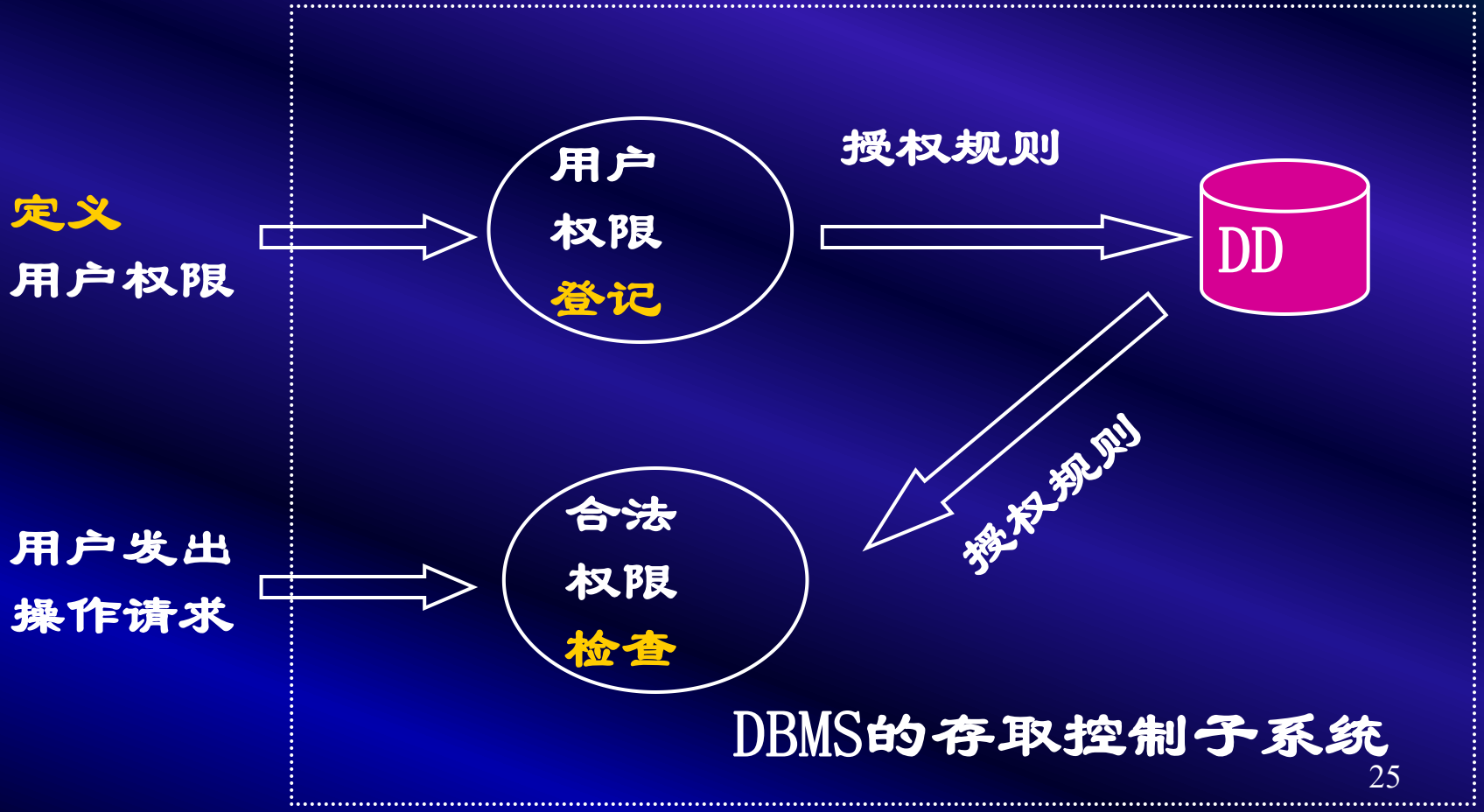
◆**授权控制**也称为访问控制，或存取控制

◆**存取控制机制包括两部分**：

- 定义用户权限
- 合法权限检查

三、DBMS的存取控制子系统

3、DBMS的存取控制子系统



三、DBMS的存取控制子系统

4、存取控制策略：

- 自主存取控制
- 强制存取控制
- 基于角色的存取控制

三、DBMS的存取控制子系统

5、自主存取控制

◆定义： 用户对于不同的数据对象有相应的存取权限，而且用户还可以将其拥有的存取权限转授给其他用户。

◆三个要素

-- （用户，数据对象，存取权限）

◆灵活性

-- 转授权限

三、DBMS的存取控制子系统

◆用户分类与权限

- 系统用户（或DBA）：拥有全部权限
- 数据对象的属主：是创建某个数据对象的用户表，拥有该对象的权限
- 一般用户：经过授权被允许进行特定操作的用户
- 公共用户：全体用户，便于共享操作而设置

＊ SQL中提供CREATE USER 等语句管理用户

三、DBMS的存取控制子系统

◆授权粒度（用户访问的数据对象的粒度 ）

授权粒度可能是：

- 数据库
- 表
- 字段
- 元组

授权粒度越细，授权子系统越灵活，提供的安全性就越完善，但系统定义与检查权限的开销越大

三、DBMS的存取控制子系统

◆存取权限

存取权限	数据对象	操作类型
模式（型）	模式（基本表）	建立、修改
	外模式（视图）	建立
	内模式（索引）	建立
数据（值）	表	查找、插入、修改、删除
	属性列	修改

DBA授权

数据对象的创建者授权

没有元组级，不支持存取谓词

＊ SQL中提供GRANT、REVOKE语句进行权限的授权和回收

三、DBMS的存取控制子系统

◆与数据值有关的授权

-- 若授权依赖于数据对象的内容，则称为是与数据值有关的授权。

与数据值有关的授权的实现难点？

难以判断**存取谓词**之间的语义关系

三、DBMS的存取控制子系统

◆自主存取控制存在的问题

“无意泄露”问题

原因：数据本身并无安全性标志

三、DBMS的存取控制子系统

6、强制存取控制

◆强制存取控制的定义

强制存取控制就是指系统为保证更高程度的安全性所采取的强制存取检查手段。

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。对于任意一个对象，只有具有合法许可证的用户才可以存取。

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。

三、DBMS的存取控制子系统

◆主体、客体与敏感度标记

在强制存取控制中，DBMS管理的实体被分为主体和客体两大类，DBMS为它们每个实例（值）指派一个敏感性标记

- **主体** -- 系统中的活动实体，即包括DBMS所管理的实际用户，也包括代表用户的各进程。
- **客体** -- 系统中的被动实体，是受主体操纵的，包括文件、基表、索引、视图等。
- **敏感度标记** -- 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**。

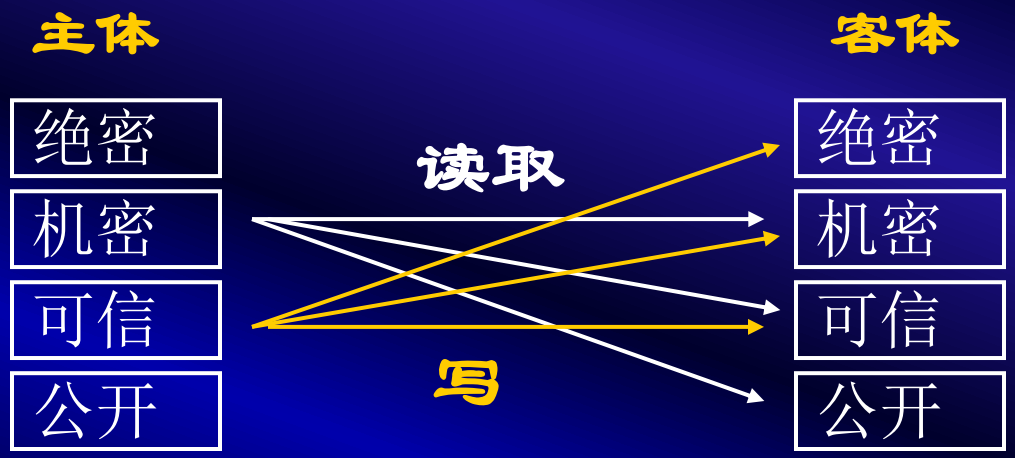
三、DBMS的存取控制子系统

◆强制存取控制机制

- **概念：** 通过对比主体的敏感性标记和客体敏感性标记，最终确定主体是否能够存取客体
- 强制存取控制规则举例

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（小于或）等于客体的密级时，该主体才能写相应的客体。



三、DBMS的存取控制子系统

7、基于角色的存取控制

- 基本思想：对系统操作的各种权限不是直接授予具体的用户，而是在用户集合与权限集合之间建立一个角色集合。每一种角色对应一组相应的权限。一旦用户被分配了适当的角色后，该用户就拥有此角色的所有操作权限。

- 即：用户通过角色与权限进行关联

- 优点：简化用户的权限管理，减少系统的开销。

＊ SQL中提供CREAT ROLE等语句进行角色管理

本讲主要内容

一. 数据库系统安全概述

二. 用户标识与认证

三. DBMS的存取控制子系统

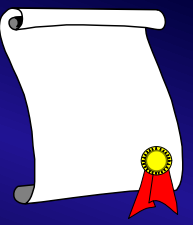
四. 授权与回收

五. 角色

六. 视图与权限

七. 审计

八. 数据加密与统计数据库的安全



四、授权与回收

(参见教材P145-147)

1、SQL中的GRANT语句

GRANT语句是表(基本表或视图)授予一个或一类用户访问表的各种权利的SQL命令.

◆基本SQL的GRANT语句形式为:

```
GRANT {ALL PRIVILEGES | privilege[, privilege] ...}
  ON [TABLE]{ 〈tablename〉 | 〈viewname〉 }
  TO {PUBLIC | 〈username〉 [, 〈username〉 ] ...}
  [WITH GRANT OPTION];
```

四、授权与回收

◆ GRANT语句可以授予下面列出的所有访问权限(ALL PRIVILEGE)，也可以给出逗号隔开的一个权限序列：

对象	对象类型	操作权限
属性列	TABLE	SELECT,INSERT,UPDATE,DELETE, ALL PRIVILEGES
视图	TABLE	SELECT,INSERT,UPDATE,DELETE, ALL PRIVILEGES
基本表	TABLE	SELECT,INSERT,UPDATE,DELETE,ALTER,INDEX, ALLPRIVILEGES
数据库	DATABASE	CREATETAB

四、授权与回收

GRANT语句说明

- 每条GRANT命令只能针对一个对象授权

例. 把查询S表的权限授予用户U1。

```
GRANT   SELECT
ON   TABLE   S
TO   U1;
```

- 可同时向多个用户授予对同一对象的多种权力

例. 把查询S表的权限授予用户U1和U2。

```
GRANT SELECT
ON TABLE S
TO U1, U2;
```

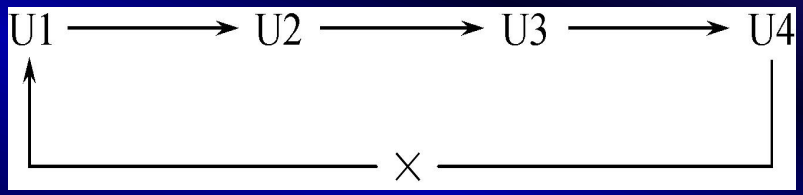
四、授权与回收

- WITH GRANT OPTION选项

使用WITH GRANT OPTION时，被授权的用户不仅具有对指定对象的指定的操作特权，而且，也可以将这个对象特权授予其他用户。

例．把查询S表的权限授予用户U1，并允许U1将此权限再授予其它用户。

```
GRANT SELECT
ON TABLE S
TO U1
WITH GRANT OPTION;
```



不允许循环授权

四、授权与回收

GRANT语句其它说明：

- 访问列的安全性可以通过视图实现；
- 访问行的安全性可以通过视图实现；
- 若要在一个视图上授予插入、删除或更新权限，视图必须是可更新的；
- 表的所有者自动拥有表的所有权限，而且不能被取消

四、授权与回收

2、SQL中的REVOKE语句

- REVOKE命令用于收回已授予给用户的特权。
- 用法与GRANT相同，是GRANT的反操作。
- SQL的REVOKE语句的语法

```
REVOKE [GRANT OPTION FOR]
    {ALL PRIVILEGES | privilege[, privilege] …}
 ON [TABLE]{〈tablename〉| 〈viewname〉}
 FROM {PUBLIC | 〈username〉[, 〈username〉] …}
[CASCADE | RESTRICT];
```

四、授权与回收

REVOKE语句说明

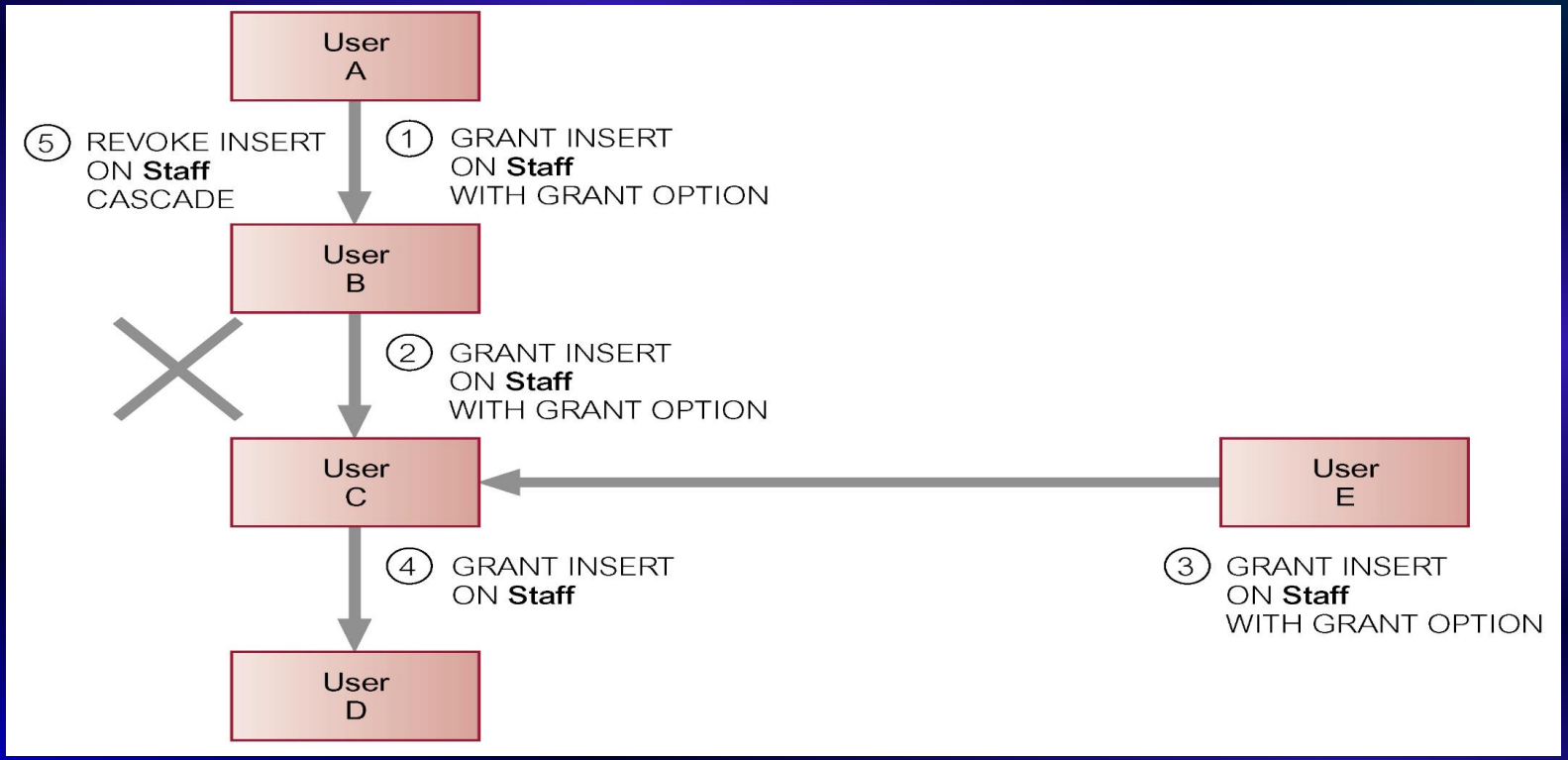
- GRANT OPTION FOR指撤销转授特权（在grant中用WITH GRANT OPTION 设置的）的权限
- ALL PRIVIEGES指该用户被授予的对指定对象所拥有的所有特权
- 基本SQL目前不支持CASCADE | RESTRICT 子句

四、授权与回收

- ORACLE与DB2 UDB都支持X/Open SQL标准的Grant和Revoke语句的语法，只是在CASCADE|RESTRICT子句等细节方面不同
- ORACLE和DB2 UDB都没有该必选的CASCADE|RESTRICT子句。ORACLE中语法的相应位置是一个可选的CASCADE CONSTRAINTS子句，这将使系统只删除与被撤消的REFERENCES权限有关的参照完整性约束。DB2 UDB中没有语法控制这种动作，当一个权限被撤消时，两个产品中与其有关的视图都缺省地变成无效的。
- ORACLE与DB2 UDB都有很多附加权限

四、授权与回收

- 其它用户授予该用户的特权不受影响



本讲主要内容

一. 数据库系统安全概述

二. 用户标识与认证

三. DBMS的存取控制子系统

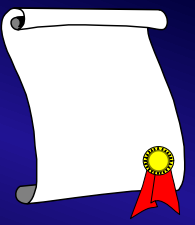
四. 授权与回收

五. 角色

六. 视图与权限

七. 审计

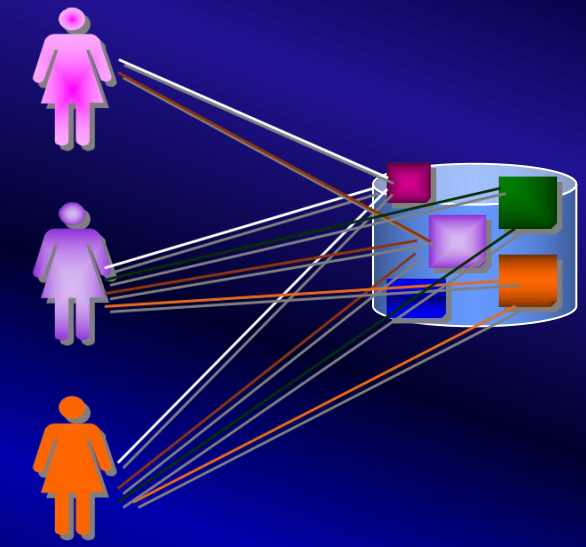
八. 数据加密与统计数据库的安全

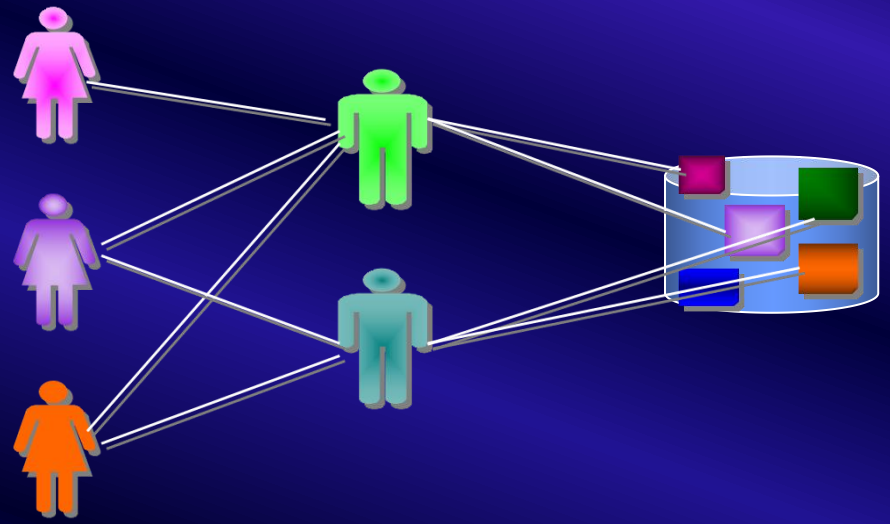


五、角色（参见教材P144）

1、角色——就是操作权限的集合

- 可以为一组具有相同权限的用户创建一个角色
- 使用角色来管理数据库权限可以简化授权的过程。





五、角色

2、数据库预定义角色

- 数据库建立时，系统自动创建的角色
- 如ORACLE中
  - ✓ CONNECT – 允许进入数据库
  - ✓ RESOURCE – 允许创建数据库对象
  - ✓ DBA – 除拥有CONNECT和RESOURCE权限外，还能对表的数据作操纵，并具有控制与数据库管理权限

五、角色

3. 部分与角色相关SQL语句

(1) 角色的创建

```
CREATE  ROLE  〈角色名〉
```

(2) 角色的删除

```
DROP  ROLE  〈角色名〉
```

(3) 给角色授权

```
GRANT  〈权限〉[,〈权限〉]…
ON 〈对象类型〉对象名
TO 〈角色〉[,〈角色〉]…
```

五、角色

(4) 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]
```

说明：

- 该语句把角色授予某用户，或授予另一个角色
- 授予者是角色的创建者或拥有在这个角色上的ADMIN OPTION
- 指定了WITH ADMIN OPTION则获得某种权限的角色或用户还可以把这种权限授予其他角色
- 一个角色的权限：直接授予这个角色的全部权限加上其他角色授予这个角色的全部权限

五、角色

(5) 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〉]…

- 用户可以回收角色的权限，从而修改角色拥有的权限
- REVOKE执行者是
  - 角色的创建者
  - 拥有在这个（些）角色上的ADMIN OPTION

五、角色

4. 角色操作举例

例： 通过角色来实现将一组权限授予一个用户。

（1）首先创建一个角色 R1

```
CREATE  ROLE  R1;
```

（2）然后使用GRANT语句，使角色R1拥有Student表的SELECT、UPDATE、INSERT权限

```
GRANT SELECT, UPDATE, INSERT
ON TABLE Student
TO R1;
```

五、角色

（3）将这个角色授予王平，张明，赵玲。使他们具有角色R1所包含的全部权限

```
GRANT   R1
TO 王平,张明,赵玲;
```

（4） 可以一次性通过R1来回收王平的这3个权限

```
REVOKE   R1
FROM 王平;
```

五、角色

例：　角色的权限修改

(1) 使角色R1在原来的基础上增加了Student表的DELETE 权限

```
GRANT DELETE
ON TABLE Student
TO R1;
```

(2) 使R1减少Student表的SELECT权限

```
REVOKE SELECT
ON TABLE Student
FROM  R1;
```

本讲主要内容

一. 数据库系统安全概述

二. 用户标识与认证

三. DBMS的存取控制子系统

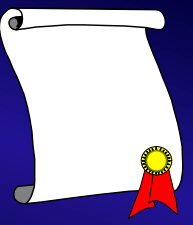
四. 授权与回收

五. 角色

六. 视图与权限

七. 审计

八. 数据加密与统计数据库的安全



六、视图与权限

- 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
- 进行存取权限控制时可以给不同用户定义不同的视图，把数据对象限制在一定范围内
- 视图机制和权限机制相配合，可间接地实现支持存取谓词的用户权限定义

六、视图与权限

例：建立计算机系学生的视图，把对该视图的SELECT权限授于王平，把该视图上的所有操作权限授于张明

（1）先建立计算机系学生的视图CS_Student

```
CREATE VIEW CS_Student
  AS
   SELECT   *
   FROM     Student
   WHERE    Sdept='CS';
```

六、视图与权限

(2) 在视图上进一步定义存取权限

```
GRANT   SELECT
ON   CS_Student
TO 王平;

GRANT ALL PRIVILIGES
ON   CS_Student
TO   张明;
```

本讲主要内容

一. 数据库系统安全概述

二. 用户标识与认证

三. DBMS的存取控制子系统

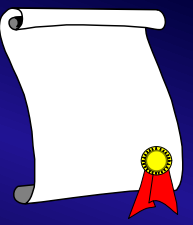
四. 授权与回收

五. 角色

六. 视图与权限

七. 审计

八. 数据加密与统计数据库的安全



七、审计（参见教材P144）

- 审计功能是把用户对数据库的所有操作自动记录下来放入审计日志中，DBA可以利用审计跟踪的信息，找出非法存取数据的人、时间和内容等。
- C2以上安全级别的DBMS必须具有审计功能

审计日志与恢复机制中的日志有什么区别？

记录的内容不同：恢复日志只记录更新操作

记录的组织方式不同：恢复日志严格按操作的时间顺序，审计日志按操作的对象

七、审计

1、审计的作用

（1）调查可疑的活动

例如，如果发现曾经有人恶意地删除了某个表中的记录，DBA可以对所有的数据库连接以及成功的或未成功的记录删除操作进行审计，以帮助确定肇事者.

（2）监视并收集某类数据库活动的信息

例如，利用审计功能，DBA可以统计哪些表经常被修改，哪些表的I/O操作比较频繁，或者在高峰时刻最多有多少并发用户连接等

七、审计

2、审计功能的可选性

- 审计很费时间和空间
- DBA可以根据应用对安全性的要求，灵活地打开或关闭审计功能
- 审计功能主要用于安全性要求较高的部门

七、审计

3、审计事件

- **服务器事件**
  - 审计数据库服务器发生的事件，如服务器启动、停止
- **系统权限**
  - 对系统拥有的结构或模式对象进行操作的审计
  - 要求该操作的权限是通过系统权限获得的
- **语句事件**
  - 对SQL语句，如DDL、DML、DQL及DCL语句的审计
- **模式对象事件**
  - 对特定模式对象（如表、视图、存储过程等）上进行的SELECT或DML操作的审计

七、审计

4、审计功能

- 基本功能
  - 提供多种审计查阅方式：基本的、可选的等等
- 多套审计规则：一般在初始化设定
- 提供审计分析和报表功能
- 审计日志管理功能
  - 防止审计员误删审计记录，审计日志必须先转储后删除
  - 对转储的审计记录文件提供完整性和保密性保护
  - 只允许审计员查阅和转储审计记录，不允许任何用户新增和修改审计记录等
- 提供查询审计设置及审计记录信息的专门视图

七、审计

5、ORACLE 的审计

- 审计记录可以存储在数据字典中，也可以存储在操作系统文件中
- AUDIT语句和NOAUDIT语句
  - AUDIT语句：设置审计功能
  - NOAUDIT语句：取消审计功能

七、审计

- SQL语句审计

  只审计语句本身，而不针对语句操作的对象

  例如：AUDIT SELECT BY U1;

- 系统权限审计

  对系统权限的使用情况进行审计

  例如：AUDIT CREATE ANY INDEX;

- 对象权限审计

  对某个指定对象的某一类操作进行审计

  例如：AUDIT ALTER, UPDATE ON SC;
  NOAUDIT UPDATE ON SC;

本讲主要内容

一. 数据库系统安全概述

二. 用户标识与认证

三. DBMS的存取控制子系统

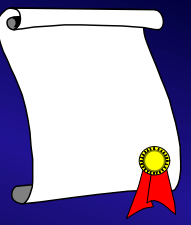
四. 授权与回收

五. 角色

六. 视图与权限

七. 审计

八. 数据加密与统计数据库的安全



八、数据加密与统计数据库安全

（参见教材P145）

1、数据加密

数据加密是防止数据库中数据在存储和传输中失密的有效手段，基本思想是将明文变换为密文。

主要的两种加密方法：

替换方法

—— 将明文中的每一个字符转换为密文中的一个字符

置换方法

—— 将明文的字符按不同的顺序重新排列

八、数据加密与统计数据库安全

2、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（例如合计、平均值等），但是不允许查询单个记录信息。

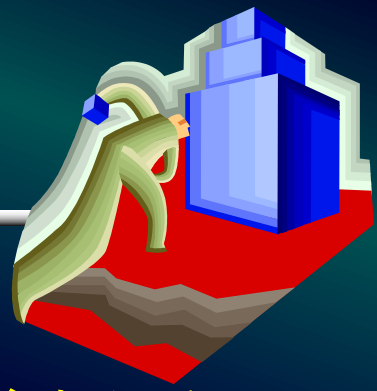
统计数据库中的安全性问题：可能存在着隐蔽的信息通道，使得可以从合法的查询中推导出不合法的信息。

解决办法：

- 规定任何查询至少涉及N个以上的记录
- 规定两个查询的相交数据项不能超过M个

Questions?

本讲主要目标



学完本讲后，你应该能够了解：

1、数据库的安全性措施是层层设置的，包括用户标识和鉴定、DBMS存取控制、视图机制、数据加密和审计追踪等；

2、DBMS的安全子系统由用户权限定义和合法权检查机制组成，其中授权规则放在数据字典中，合法权的检查也只检查数据字典；

3、两种存取控制方法：自主存取控制方法的灵活性和强制存取控制方法的严格性；

4、审计日志的作用及与恢复机制中日志的区别；

5、统计数据库中可能存在着隐蔽的信息通道。

问题讨论

1、为什么在数据库应用系统中安全性问题非常的突出?

2、数据库应用系统中存在哪些安全隐患?

3、安全性与完整性有什么本质区别?

4、SQL安全控制的粒度有哪些?

练习

教材：《数据库系统原理教程》（第2版）

P178

1）1

2）2

3）3

4）7

参考资料

- MySQL用户认证及权限控制
  - https://www.cnblogs.com/geaozhang/p/6710454.html
- MySQL用户与权限
  - https://zhuanlan.zhihu.com/p/55798418
- MySQL角色(role)功能介绍
  - https://zhuanlan.zhihu.com/p/367873840

附录

- SQL SERVER
- ORACLE
- DB2

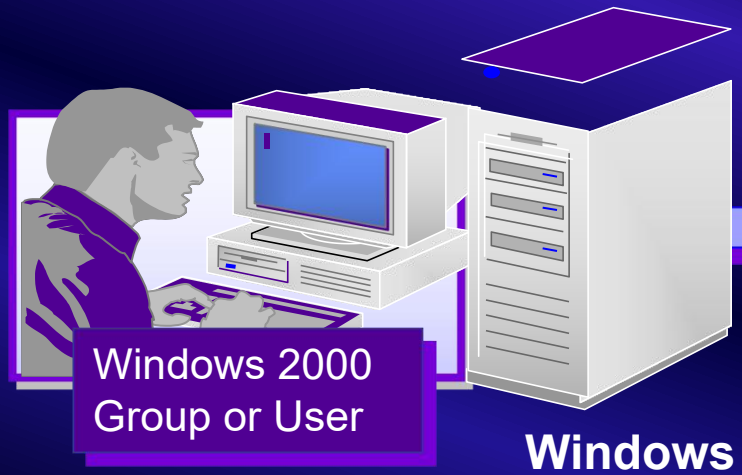
SQL SERVER

1. SQL SERVER 用户认证

- 两种身份认证模式
  - Windows模式:允许用户通过Windows用户帐户连接
  - SQL Server模式:Windows身份验证与SQL Server身份验证的混合使用
- 设置身份认证模式
  - 在安装SQL Server时设置
  - 在SQL Server Management Studio的对象资源管理器(OEM)中修改

SQL SERVER

1. SQL SERVER 用户认证

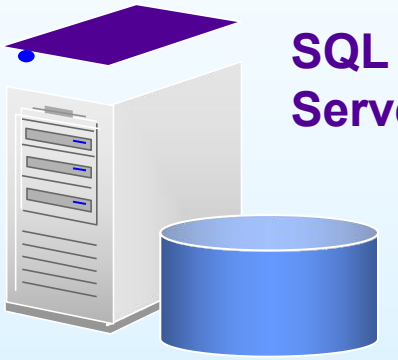


Windows 2000

Or

AUTHENTICATION

SQL Server
Verifies
Trusted Connection



SQL Server

SQL Server
Verifies Name
and Password

SQL SERVER

1. SQL SERVER 用户认证

- 登录帐号管理
  - 一个合法的登录帐号只表明该帐号通过了Windows或SQL Server认证
  - 一个登录帐号总是与一个或多个数据库用户帐号相对应
- 用户帐号管理
  - 用户帐号用来指出哪一个人可以访问哪一个数据库
  - 在一个数据库中用户帐号唯一标识一个用户，两个不同数据库中可以有相同的用户帐号
  - 用户对数据的访问权限以及对数据库对象的所有关系都是通过用户帐号来控制的

SQL SERVER

2. SQL Server的安全特征

- SQL Server中的角色：
  - 固定的服务器角色
  - 系统预先定义了一些服务器固有角色，这些角色各自具有某种或某些操作SQL Server服务器的权限。用户不能删除也不能创建新的服务器固有角色。
  - 固定的数据库角色
  - 系统为每一个数据库预先定义数据库固有的角色，用户不能删除数据库固有角色，但用户可以在具体的数据库中再创建新的数据库安全角色
  - 用户定义的数据库角色

SQL SERVER

2. SQL Server的安全特征

- 权限管理
  - 两种类型的权限：
    - 对象权限

      总是针对表、视图、存储过程而言，它决定了能对表、视图、存储过程执行哪些操作
    - 语句权限

      主要指用户是否具有权限来执行某一语句。这些语句通常具有管理性的操作，如创建数据库、表和存储过程等。这些语句虽然仍包含有操作的对象，但这些对象在执行该语句前并不存在于数据库中

SQL SERVER

2. SQL Server的安全特征

- 权限管理
  - 三种管理权限的命令
    - GRANT
    - REVOKE
    - DENY：用来禁止用户对某一对象或语句的权限，明确禁止其对某一对象执行某些操作或运行某些语句。语法参数与GRANT语句中的类似

ORACLE

1. ORACLE 用户认证

在数据库创建之后，需要为DBA选择一种身份认证方式ORACLE数据库本身具有一套完善的安全管理机制，每个连接到ORACLE的用户都需要有一个数据库帐户．但这些帐户信息都保存在数据库内部，在打开数据库之前，不能使用这些信息对用户身份进行认证．因此，DBA在执行诸如启动实例、加载和打开数据库等操作时，需要使用独立于数据库的方法进行认证。

ORACLE

1. ORACLE 用户认证

- 特殊系统权限：SYSDBA 和 SYSOPER
  - 具有SYSDBA 和 SYSOPER系统权限的用户，甚至能够在不打开数据库的情况下直接访问实例，可见他们的身份认证是与数据库完全无关的，而是依赖于外部服务进行的。
  - 用户在使用SYSDBA 或 SYSOPER权限连接实例时，实际上并不是进入到用户所属的模式中，而是进入默认的模式。
    - SYSDBA 进入 SYS 模式
    - SYSOPER 进入 PUBLIC 模式

ORACLE

1. ORACLE 用户认证

- 设置DBA认证方式
  - 操作系统认证

    ORACLE将在在OS中创建一个特殊的用户组，所有属于这个组的用户都被授予SYSDBA或SYSOPER权限。如果登录到OS中的用户属于这个组，他将自动具有SYSDBA或SYSOPER权限。用户认证工作由OS完成。
  - 口令文件认证

    具有SYSDBA或SYSOPER权限的用户和口令被记录在一个经过加密处理的口令文件中。口令文件是一个OS文件，被存储在服务器中（不是数据库中）。当用户连接数据库时，ORACLE使用口令文件对用户的身份进行认证。

ORACLE

1. ORACLE 用户认证

- 具体选择哪种DBA认证方式主要取决于两个因素：
  - 用户的操作是在本地还是通过网络进行
  - 如果通过网络进行，是否具有一个可靠的安全网络连接
- 默认情况下，安装ORACLE服务器后，只有Administrator用户属于ORA_DBA组。
- 若DBA只需要在本地执行管理，建议使用OS认证方式
- 在不具备安全网络连接的情况下，如果要进行远程管理，就必须选择口令文件认证方式

ORACLE

1. ORACLE 用户认证

- 设置普通用户的认证方式
  - 通过操作系统认证
  - 通过数据库认证
  - 通过网络服务认证
- ORACLE支持在同一个数据库中使用多种身份认证方式

ORACLE

2. ORACLE的安全性特征

- ORACLE支持角色的概念.
  - CONNECT – 允许进入数据库
  - RESOURCE – 允许创建数据库对象
  - DBA – 除拥有CONNECT和RESOURCE权限外,还能对表的数据作操纵,并具有控制与数据库管理权限
- 可以为角色授予系统权限和对象权限
- 一个角色可以被授予另一个角色
- 可以将角色授予PUBLIC用户组

ORACLE

2. ORACLE的安全性特征

- 权限管理
  - 两类权限：系统权限和对象权限
  - 系统权限
    - 是在数据库中执行某种操作、或者针对某一类对象执行某种操作的权利。并不针对某一个特定对象，而是针对整个数据库范围
    - 只应当向DBA或应用程序开发者授予系统权限
  - 对象权限
    - 是针对某个特定的模式对象执行操作的权利。
    - 可设置权限的对象：表、视图、序列、存储过程、存储函数和包
  - 用户自动拥有他的模式中的所有对象的全部对象权限。

DB2

1. DB2 用户认证

- 在每次试图对本地或者远程数据库进行连接时，用户都被系统要求验证。DB2将会把用户名和口令传递给操作系统或者外部安全设施来进行检验。
- 在配置DB2客户端和服务器之间的连接时，用户可以指定验证类型。验证类型定义了如何进行验证以及在哪里进行验证。
- 验证类型在参与连接的每一台机器上进行设置，可以选择的验证类型要依赖所处的环境。
- 验证类型分两类：
  - 服务器端的验证类型
  - 客户端的验证类型

DB2

1. DB2 用户认证

- 服务器端的验证类型
  - SERVER
  - SERVER_ENCRYPT
  - CLIENT
  - KERBEROS
  - KRB_SERVER_ENCRYPT
- 客户端的验证类型
  - SERVER
  - SERVER_ENCRYPT
  - CLIENT
  - KERBEROS

DB2

2. DB2 UDB的安全性特征

- 组特权
  - 组特权相当于角色的概念
  - 可以对用户组授权，用户组的定义不是在DB2中进行，而是在操作系统或其它外部安全设施中进行
  - DB2有一个特殊的用户组：PUBLIC用户组，该组包含所有的DB2用户，并且，CONNECT特权自动授予PUBLIC组

DB2

2. DB2 UDB的安全性特征

- DB2为用户赋予不同级别的权限和特权
- 权限(authorities) ：一组高层次的用户权利，通常授予那些需要对数据库和实例进行管理和维护的用户
- 数据库的5种类型的权限
  - SYSADM – 系统管理
  - SYSCTRL – 系统控制
  - SYSMAINT – 系统维护
  - LOAD – 对表进行LOAD操作的权限
  - DBADM – 数据库管理
- SYSADM、SYSCTRL和SYSMAINT三种权限不通过GRANT命令进行授权，它们与特定的用户组相关，需要在实例级别定义 —— 设置数据库管理器配置参数SYSADM_GROUP、SYSCTRL_GROUP和SYSMAINT_GROUP 。

DB2

2. DB2 UDB的安全性特征

- 特权(Privileges)：是针对某个数据库对象的用户权利，通常授予那些需要对数据库对象进行存取的用户
- 三种类型的特权：
  - 拥有者特权

    创建对象的用户通常对于该对象具有全面的控制权限。
  - 个体特权

    个体特权允许用户对数据库对象执行特定操作
  - 隐式特权

    是当用户被显式赋予某些高层次特权时，自动赋予用户的特权

DB2

2. DB2 UDB的安全性特征

- 两级特权
  - 数据库级特权
    - 涉及某些将数据库作为一个整体来进行的操作
    - 只有具有SYSADM或DBADM权限的用户才可以将数据库特权对其他用户进行授予或收回
  - 数据库对象级特权
    - 数据库特权
    - 模式特权
    - 视图特权
    - 索引特权
    - 过程、函数和方法特权
    - 序列特权
    - 别名特权
    - 程序包特权