第四章 数据库安全性

1. 答：

数据库的安全性是指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。

2. 答：

安全性问题不是数据库系统所独有的，所有计算机系统都有这个问题。只是在数据库系统中大量数据集中存放，而且为许多最终用户直接共享，从而使安全性问题更为突出。

系统安全保护措施是否有效是数据库系统的主要指标之一。

数据库的安全性和计算机系统的安全性，包括操作系统、网络系统的安全性是紧密联系、相互支持的，

3. 答：

信息安全标准的发展历史详细请参见《概论》图 4.1。

简单地讲，TCSEC 是 1985 年美国国防部颁布的《DoD 可信计算机系统评估准则》。CC 通用准则 V2.1 版于 1999 年被 ISO 采用为国际标准，2001 年被我国采用为国家标准。目前 CC 已经基本取代了 TCSEC，成为评估信息产品安全性的主要标准。

TDI/TCSEC 标准是将 TCSEC 扩展到数据库管理系统，即《可信计算机系统评估标准关于可信数据库系统的解释》在 TDI 中定义了数据库管理系统的设计与实现中需满足和用以进行安全性级别评估的标准。

TDI 与 TCSEC 一样，从安全策略、责任、保证和文档四个方面来描述安全性级别划分的指标。每个方面又细分为若干项。这些指标的具体内容，参见《概论》4.1.2。

CC 提出了目前国际上公认的表述信息技术安全性的结构，即把对信息产品的安全要求分为安全功能要求和安全保证要求。

安全功能要求用以规范产品和系统的安全行为，安全保证要求解决如何正确有效地实施这些功能。

4. 答：

评估保证级	定　　义
EAL1	功能测试（functionally tested）
EAL2	结构测试（structurally tested）
EAL3	系统的测试和检查（methodically tested and checked）
EAL4	系统的设计、测试和复查（methodically designed, tested and reviewed）
EAL5	半形式化设计和测试（semiformally design and tested ）
EAL6	半形式化验证的设计和测试（semiformally verified design and tested）
EAL7	形式化验证的设计和测试（formally verified design and tested）

详细参见《概论》表 4.2（P134）

5. 答：

实现数据库安全性控制的常用方法和技术有：

（1）用户标识和鉴别：由系统提供一定的方式让用户标识自己的名字或身份。每次用户要求进入系统时，由系统进行核对，通过鉴定后才提供系统的使用权。

（2）存取控制：通过用户权限定义和合法权检查确保只有合法权限的用户访问数据库，所有未被授权的人员无法存取数据。

（3）视图机制：为不同的用户定义视图，通过视图机制把要保密的数据对无权存取的用户隐藏起来，从而自动地对数据提供一定程度的安全保护。

（4）审计：建立审计日志，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，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。

（5）数据加密：对存储和传输的数据进行加密处理，从而使得不知道解密算法的人无法获知数据的内容。

6. 答：

自主存取控制方法：定义各个用户对不同数据对象的存取权限。当用户对数据库访问时首先检查用户的存取权限。防止不合法用户对数据库的存取。

强制存取控制方法：每一个数据对象被（强制地）标以一定的密级，每一个用户也被（强制地）授予某一个级别的许可证。系统规定只有具有某一许可证级别的用户才能存取某一个密级的数据对象。

自主存取控制中自主的含义是：用户可以将自己拥有的存取权限“自主”地授予别人。即用户具有一定的“自主”权。

7. 答：

SQL 中 的自主存取控制是通过 GRANT 语句和 REVOKE 语句来实现的。如：

```
GRANT SELECT, INSERT ON Student
TO 王平
WITH GRANT OPTION;
```

将 Student 表的 SELECT 和 INSERT 权限授予了用户王平，后面的 “WITH GRANT OPTION”子句表示用户王平同时也获得了“授权”的权限，即可以把得到的权限继续授予其他用户。

```
REVOKE  INSERT  ON Student  FROM 王平 CASCADE;
```

将 Student 表 的 INSERT 权限从用户王平处收回，选项 CASCADE 表示，如果用户王平将 Student 的 INSERT 权限又转授给了其他用户，那么这些权限也将从其他用户处收回。

8. 答:

（1）

```
GRANT  SELECT ON  职工，部门
TO 王明;
```

（2）

```
GRANT  INSERT，DELETE ON职工，部门
TO  李勇;
```

（3）

```
GRANT  SELECT ON  职工
WHEN USER（）= NAME
TO  ALL;
```

这里假定系统的 GRANT 语句支持 WHEN 子句和 USER（）的使用。用户将自己的名字作为 ID。注意，不同的系统这些扩展语句可能是不同的。读者应该了解你使用的 DBMS 产品的扩展语句。

（4）

```
GRANT  SELECT，UPDATE（工资） ON 职工
TO  刘星;
```

（5）

```
GRANT  ALTER  TABLE  ON  职工，部门
TO  张新;
```

（6）

```
GRANT  ALL  PRIVILIGES ON  职工，部门
TO  周平
WITH GRANT     OPTION;
```

（7）

首先建立一个视图。然后对这个视图定义杨兰的存取权限。

```
CREATE  VIEW   部门工资 AS
 SELECT 部门.名称，MAX（工资），MIN（工资），AVG（工资）
 FROM   职工，部门
 WHERE 职工.部门号 = 部门. 部门号
 GROUP BY 职工.部门号;

GRANT  SELECT ON  部门工资
TO  杨兰;
```

9. 答:

（1）

```
REVOKE SELECT ON 职工，部门
FROM 王明;
```

（2）

```
REVOKE INSERT，DELETE ON职工，部门
FROM 李勇;
```

（3）

```
REOVKE SELECT ON 职工
WHEN USER（）= NAME
FROM ALL;
```

这里假定用户将自己的名字作为 ID，且系统的 REOVKE 语句支持 WHEN 子句，系统也支持 USER（）的使用。

（4）

```
REVOKE SELECT，UPDATE ON 职工
FROM 刘星;
```

（5）

```
REVOKE ALTER TABLE ON 职工，部门
FROM 张新;
```

（6）

```
REVOKE ALL PRIVILIGES ON 职工，部门
FROM 周平;
```

（7）

```
REVOKE SELECT ON 部门工资
FROM 杨兰;
DROP VIEW 部门工资;
```

10. 答：

强制存取控制（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。

11. 答：

主体是系统中的活动实体，既包括 DBMS 所管理的实际用户，也包括代表用户的各进程。

客体是系统中的被动实体，是受主体操纵的，包括文件、基表、索引、视图等。

对于主体和客体，DBMS 为它们每个实例（值）指派一个敏感度标记（Label）。敏感度

标记被分成若干级别，例如绝密（Top Secret）、机密（Secret）、可信（Confidential）、公开（Public）等。主体的敏感度标记称为许可证级别（Clearance Level），客体的敏感度标记称为密级（Classification Level）。

12. 答：

假设要对关系变量 S 进行 MAC 控制，为简化起见，假设要控制存取的数据单元是元组，则每个元组标以密级，如下表所示：（4=绝密，3=机密，2=秘密）

S#	SNAME	STATUS	CITY	CLASS
S1	Smith	20	London	2
S2	Jones	10	Paris	3
S3	Clark	20	London	4

假设用户 U1 和 U2 的许可证级别分别为 3 和 2，则根据规则 U1 能查得元组 S1 和 S2，可修改元组 S2；而 U2 只能查得元组 S1，只能修改元组 S1。

解析：

这里假设系统的存取规则是：（1）仅当主体的许可证级别大于或等于客体的密级时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才能写相应的客体。

13. 答：

审计功能是指 DBMS 的审计模块在用户对数据库执行操作的同时把所有操作自动记录到系统的审计日志中。

因为任何系统的安全保护措施都不是完美无缺的，蓄意盗窃破坏数据的人总可能存在。利用数据库的审计功能，DBA 可以根据审计跟踪的信息，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。

14. 答：

统计数据库允许用户查询聚集类型的信息，如合计、平均值、最大值、最小值等，不允许查询单个记录信息。但是，人们可以从合法的查询中推导出不合法的信息，即可能存在隐蔽的信息通道，这是统计数据库要研究和解决的特殊的安全性问题。

*15. 答：

不同的 DBMS 产品以及同一产品的不同版本的安全措施各不相同，仁者见仁，智者见智，请读者自己了解。