

一个改进的主从结构表安全数据模型^{*}

王艳磊¹ 毛宇光^{1,2}

(南京航空航天大学信息科学与技术学院 南京 210016)¹

(计算机软件新技术国家重点实验室(南京大学) 南京 210093)²

摘要 到目前为止,大量的文献已经提出了许多用以实现多级安全数据库系统的安全模型,不同的模型有不同的优点。本文针对原有主从结构表安全模型容易产生语义模糊性和操作不完备性等问题,提出了一个能够消除语义模糊性和操作不完备性的新的主从结构表模型。该模型增加了基元组和数据继承的概念,重新定义了多实例完整性和参照完整性,将 PUPDATE 操作和数据继承完整性引入该模型,大大增强了系统的安全性和非二义性。

关键词 数据模型,多级安全,多实例,数据完整性,数据语义

An Extended Secure Data Model of Master-Slaver Structure Tables

WANG Yan-Lei¹ MAO Yu-Guang^{1,2}

(College of Information Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 210016)

(State Key Laboratory for Novel Software Technology at Nanjing University, Nanjing 210093)

Abstract By far, a secure model to realize multilevel secure database system has been mentioned in large quantities of papers and different models have different advantages. In consideration of semantic ambiguity and operational incompleteness caused by secure model of original master-slaver structure tables, a new secure model is presented to eliminate the problem mentioned above. In this model, the new conception of base tuple and data-inherit are added, polyinstantiation integrity and referential integrity are redefined, and PUPDATE operation and data-inheret integrity are introduced, which strengthen the security and unambiguous of system largely.

Keywords Data model, Multilevel security, Polyinstantiation, Data integrity, Data semantic

根据美国国防部(DoD)颁布的“可信计算机系统评估标准(TCSEC)^[1]”和 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(TDI)^[2]”要求,B1 级以上的安全系统必须实现强制存取控制(Mandatory Access Control),必须建立形式化的安全模型。实施了强制存取控制的数据库管理系统,称为多级安全数据库管理系统(MLS/DBMS)^[3]。当前流行的 MLS/DBMS 大多采用具有多级安全策略形式化的 Bell Lapadula(BLP)模型^[4,5]作为其安全模型。在 BLP 模型中,主体按照“向下读,向上写”的原则访问客体。“向下读”能有效地防止低安全级用户直接获取敏感信息,但“向上写”原则存在潜在的危险,可能形成隐秘通道(Covert Channel),这些存在的安全缺陷很可能被黑客利用。针对这些缺点,文[6]对安全级、范围和规则进行了改进,提出了一种新的多级安全关系数据模型,即主从结构表安全模型。改进后的模型虽然提高了多级安全数据库的安全性,但是改进后的读写安全策略通常会产生语义模糊性和操作不完备性。本文针对这些问题提出了一个新的主从结构表安全模型。

本文具体安排如下:第 1 节对原主从结构表安全模型产生的语义模型性和操作不完备性进了分析;第 2 节提出了一个改进的主从结构表安全模型;第 3 节讲述了改进后主从结构表安全模型的 5 个完整性约束规则;第 4 节分别对该模型中 3 个传统 SQL 语句和一个新增加的 PUPDATE 的数据处理操作进行了详细说明;最后总结全文。

1 主从结构表模型的缺陷

1.1 语义模糊性

多级安全数据库中语义模糊问题通常由下列情况产生^[7]:在多级安全数据库中允许具有不同安全级的主体(习惯地将安全级定义为:U 非密,C 秘密,S 机密,TS 绝密,U<C<S<TS)对同一关系进行操作。当安全级较低的主体向关系中插入数据的同时,在安全级较高的元组中已经存有该数据。由于避免推理安全级较高的数据,不能拒绝该插入操作。同时,为了保持数据的完整性,也不能删除安全级较高的数据,因此产生了造成语义模糊的多实例问题。同样,安全级较高的用户插入的数据与关系中安全级较低的数据相同时,也能造成该类问题,因为用其替代原来的数据可引起信息泄露问题。多实例问题虽然可用于防止信息推理问题,但同时也造成语义的模糊性。在文[6]中,NMD 的主表、从表分别如文中表 4、表 7 所示,那么安全级为 S 的用户看到的 NMD 如表 1 所示。如果对于一个更高密级为 TS 的用户来说,就很难确定 Mission 的值是“空间探索”还是“观光”,造成了安全级为 TS 的用户取值的模糊性。

表 1. 安全级为 S 的用户看到的 NMD

Name	Mission	Destination	TC
小鹰 S	空间探索 S	火星 S	S
小鹰 C	观光 C	火星 C	C

^{*}973 计划:“海量信息系统规律、模型和维护机理研究”子课题,海量信息系统知识与管理研究,编号:G1999032701。王艳磊 硕士研究生,研究方向为数据库安全;毛宇光 副教授,博士后,主要研究方向为数据库理论、特种数据库及多值逻辑。

1.2 语义模糊性

在之前的多级安全数据库中,对关系中每个数据元素的密级的定义都是可比的,读写规则也是针对元素密级间具有可比性制定的。然而,这种可比的密级之间的规则对操作处理具有一定的局限性,造成了操作的不完备性。假设 M_1, M_2 是不可比的安全密级,它们的最小上界和最大下界分别是 S 和 U 。事实上,对于文[6]中的主从结构表安全模型来说,安全级为 S 的用户是无法将表 2 中这个元组添加到该 NMD 关系中的。

表 2 安全级为 S 的用户即将插入 NMD 的一个元组

Name	Mission	Destination	TC
长城 U	空间探索 M_1	土星 M_2	S

2 改进的主从结构表模型

针对上节提到的这些缺点,作者对主从结构表模型进行改进,建立一个新的主从结构表安全模型,并对该新模型进行数据语义解释。

2.1 模型定义

一个扩展的包括安全级的关系为多级关系,一个多级关系由如下两部分组成。

定义 1 设 A_i 是定义在域 D_i 上的数据属性,每个 C_i 是 A_i 的密级属性, C_i 在密级区间 $[L_i \dots H_i]$ (L_i 和 H_i 分别表示多级安全关系的最低和最高密级, $L_i \leq H_i$) 中取值, TC 为元组密级属性并且其安全密级大于等于该元组 C_i 中的最大值,那么 $R(A_1, C_1, A_2, C_2, \dots, A_n, C_n, TC)$ 被称为多级关系模式。

定义 2 设 $a_i \in D_i$ 且其密级 $c_i \in [L_i \dots H_i]$, 或者 $a_i = \text{null}$ 且其密级 $c_i \in [L_i \dots H_i] \cup \text{null}$; tc 是元组密级属性且其安全密级大于等于该元组 c_i 中的最大值,那么 $r(A_1, C_1, A_2, C_2, \dots, A_n, C_n, TC)$ 被称为关系实例,每个关系实例是形如 $(a_1, c_1, a_2, c_2, \dots, a_n, c_n, tc)$ 的元组集合。

在模型中,每个关系任何时候只具有唯一的关系实例。不同级别的主体可能会有实例的不同视图。

定义 3 当元组 t 满足 $t[TC] = t[C_1]$ 时,则称元组 t 为该实体的基元组。

定义 4 当元组 $t_1 \in r$ 且 $1 \leq i \leq n$, 如果 $t_1[A_i] \neq \text{null} \wedge t_1[C_i] < t_1[TC]$, 存在元组 $t_2 \in r$, 满足 $t_2[A_1, C_1] = t_1[A_1, C_1] \wedge t_2[TC] = t_2[C_1] = t_1[C_1] \wedge t_2[A_i] = t_1[A_i]$, 那么对于满足上面条件的 $t_1[A_i], t_2[A_i]$, 称 t_1 和 t_2 是存在继承关系的元组, $t_1[A_i]$ 继承了 $t_2[A_i]$ 的数据值。

由定义可知,任一安全级元组的数据都可以看作由两部分组成:属于其自身的数据和从较低安全级元组中所继承的数据。例如表 3 关系 NMD 中的两个元组存在继承关系, U -元组被称为基元组。在 S -元组中,属性 Mission 的值“空间探索”是属于该元组自身的数据,而属性 Destination 的值“火星”是从 U -元组中所继承的数据。从 U -元组中继承的数据 Destination 可以被安全级为 U -主体所改变,而属于 S -主体自身的数据 Mission 不能够被 U -主体所改变,只能被 S -主体所改变。

表 3 在 NMD 中存在继承关系的两个元组

Name	Mission	Destination	TC
小鹰 U	空间探索 S	火星 U	S
小鹰 U	观光 U	火星 U	U

在标准关系模式中,每个元组是由其码属性值唯一标识

的。而在多级关系模式中,为了维护数据的完整性,引入了多实例(Polyinstantiation)的概念。多实例^[7]是指同时存在多个同名但存取级不同的数据客体,这些实例是以它们的存取级区分的。多实例可以涉及元组和数据元素;元组多实例又称为实体多实例,是指具有相同外观主码,但其外观主码的存取级不同的多个元组;元素多实例又称为属性多实例,是指具有相同的外观主码和外观主码的存取级,但具有一个或更多不同属性值的多个元组。

在引入基元组、继承关系及实体多实例和元素多实例的概念后,原来主从结构表的概念已不能够满足多级关系模式和关系实例定义的要求,作者提出了一个新的、改进的主从结构表概念。主表还是用来模拟单级数据库中的表格,主要用来存储一个多级关系的基元组,所以主表又称为基表。从表是用来存储假写的结果,同时也是假读的数据源,在一开始的时候为空。这样,从表中出现的所有外观主码在主表中均出现过,从表的存在对用户来说也是透明的。在从表中,只要是存在一个属性的密级小于该元组的密级,那么该元组就和主表中一个具有相同外观主码和相同外观主码密级的基元组或从表中具有相同外观主码和外观主码密级的元组存在继承关系,该数据就继承了主表中基元组或从表中低密级元组的对应数据。

2.2 模型的数据语义

为了更好地解释新模型的数据语义,对于关系实例 $r(A_1, C_1, A_2, C_2, \dots, A_n, C_n, TC)$ 和所有元组 $t(t \in r)$, 下面给出该模型数据语义的形式化描述:

(1) 外观主码 A_1 和密级属性 $C_1, t[A_1, C_1]$ 确定实例 r 中的一个实体和该实体的密级。 $t[C_1] = c_1$ 表示该实体可以被 c_1 -主体所创建和仅能被 c_1 -主体所删除,该实体被称为 c_1 -实体。

(2) 若 $t[TC] = tc$ 和 $t[C_1] = c_1$,

① 元组 t 可以被 tc -主体所添加,并且元组 t 中的所有数据都能被 tc -主体所接受;

② 元组 t 只能被 c' -主体 ($c' \geq tc$) 看到,也就是说一个主体只能看到那些安全级等于或低于它的元组密级的数据;

③ 元组 t 既能够被 tc -主体删除,也能够被 c_1 -主体删除。当被 c_1 -主体删除时,该实体也将被删除。

(3) 若 $t[TC] = t[C_1]$, 由定义 3 可知,元组 t 是该实体的基元组,存储在主表 T^1 中。对于所有从表 T_2 中的 t' ($t' \in r$), 若满足 $t'[A_1, C_1] = t[A_1, C_1]$, 则元组 t 只能在整个实体被删除后才能够被删除。

(4) 数据属性 A_k 和密级属性 C_k ($2 \leq k \leq n$),

① 若 $t[A_k, C_k]$ 满足 $t[C_k] = c_k$ 和 $t[TC] = tc$ ($c_k \leq tc$), 数据 $t[A_k]$ 能够被 tc -主体接受且该数据属于 c_k -主体自身的数据, $t[A_k, C_k]$ 能够被 c_k -主体或 tc -主体所更新;

② 若 $t[C_k] < t[TC]$ 且 $t[A_k] \neq \text{null}$, 则 $t[A_k]$ 是从 $t'[A_k]$ (t' 满足 $t'[A_1, C_1] = t[A_1, C_1] \wedge t'[TC] = t'[C_k] = t[C_k]$) 继承来的数据。当 $t'[A_k, C_k]$ 发生变化或 t' 被删除时, $t[A_k]$ 也要发生相应的变化。

(5) 若 $t[A_k, C_k] = [\text{null}, c_k]$ ($c_k < tc$), 表示 tc -主体希望从 c_k -主体中继承数据属性 A_k 的值,但属性 A_k 不属于 c_k -主体的自身数据,因此 tc -主体所继承的数据值为空值,且该 c_k -元组的实体不能够被 c_k -主体所接受。

(6) 若 $t[A_k, C_k] = [\text{null}, \text{null}]$ 和 $t[A_k, C_k] = [\text{null}, tc]$, 表示在安全级 tc 上属性 A_k 没有数据是可用的。当 $tc \notin \{L_k \dots H_k\}$ 时, $t[A_k, C_k]$ 取值 $[\text{null}, \text{null}]$; 其它情况下取值 $[\text{null}, tc]$ 。

3 模型的规则

性质 1(实体完整性)

- (1)外观主码中的每一个数据属性值非空;
- (2)外观主码中的每一个数据属性对应的密级相同;
- (3)外观主码对应的密级要么最小,要么非主码属性的密级为空值。

性质 2(多实例完整性)

(1) $A_1, C_1, C_i \rightarrow A_i$,即外观主码、外观主码的存取级以及属性的存取级函数决定此属性的值,表明关系的真正主码是 $A_1, C_1, C_2, \dots, C_n$;

(2) $A_1, C_1, TC \rightarrow A_i, C_i$,即一个关系中的每个实体在每个存取级上至多有一个元组;

(3) $A_1, TC \rightarrow C_1$,即在一个关系中有很多的实体具有相同的外观主码值,但是对于每个存取级别上的主体,至多能够接受一个具有该外观主码值的实体。

性质 3(数据继承完整性)

如果从表中 $t[A_k] \neq \text{null} \wedge t[C_k] < t[TC]$,则在主表或从表中必存在元组 $t' \in r$,满足 $t'[A_1, C_1] = t[A_1, C_1] \wedge t'[TC] = t[C_k] = t[C_k] \wedge t'[A_k] = t[A_k] (1 \leq k \leq n)$,即若从表中安全级较高的元组中存在安全级较低的数据时,则在主表中的基元组或从表中安全级较低的元组必须存在。

性质 4(外码完整性)

外码中的每个数据属性值要么取空值,要么等于被参照关系的外观主码值且外码中的每个数据属性对应的密级相同。

性质 5(参照完整性)

- (1)每一个非空的外码必须在被参照关系中有一个相匹配的元组,且被参照关系的外观与外码相等;
- (2)外码的安全级必须大于等于被参照主码的安全级;
- (3)包含外码的元组的安全级必须等于被参照元组的安全级,即 c -元组只能参照 c -元组。

4 模型的存取操作

4.1 插入操作

设 t 是 c -主体要插入的元组, $1 \leq k \leq n$,若 A_k 包括在插入语句的属性列表中,则 $t[A_k, C_k] = (a_k, c)$;若 A_k 未包括在属性列表中,则①如果 $c \in [L_k \dots H_k]$,则 $t[A_k, C_k] = (\text{null}, c)$;②如果 $c \notin [L_k \dots H_k]$,则 $t[A_k, C_k] = (\text{null}, \text{null})$;且该元组 t 的元组密级为 c ,即 $t[TC] = c$ 。元组 t 通过下面步骤完成插入操作:

第一步,检查元组 t 是否满足实体完整性。若不满足,插入失败。

第二步,若元组 t 中含有外码,检查 t 是否满足外码完整性和参照完整性。若不满足,插入失败。

第三步,在从表中寻找元组密级 TC 等于 c 且与元组 t 具有相同外观主码的元组。若能找到这样的元组,插入失败。在主表中寻找元组密级 TC 等于 c 且与元组 t 具有相同外观主码的元组。若能找到这样的元组,插入失败。

第四步,在主表中寻找与元组 t 具有相同外观主码的元组。若能找到这样的元组 s 且 $s[C_1] < s[TC]$,则将元组 s 转存至从表中,然后将元组 t 插入主表中;若 $s[C_1] = s[TC]$,则将元组 t 插入到从表中。

如果 NMD 的主表如文[6]中表 4 所示且从表为空,那么安全级为 C 的用户插入元组(小鹰、观光、火星)后的主、从表分别如表 4、表 5 所示。

表 4 INSERT 操作后 NMD 的主表

Name	Mission	Destination	TC
长城 U	空间探索 U	月球 U	U
小鹰 C	观光 C	火星 C	C

表 5 INSERT 操作后 NMD 的从表

Name	Mission	Destination	TC
小鹰 S	间谍 TS	火星 S	TS

4.2 数据继承操作

c -主体的数据继承操作可以写成如下形式的语句($2 \leq i, j \leq n$):

```
PUPDATE R
GET Ai FROM ci[ , Aj FROM cj ]...
[WHERE p]
```

此处, p 是一个谓词表达式,用以确定要继承的元组,可以包括数据属性上的条件,也可以包括存取级属性和元组存取级属性上的条件。在此操作中,只有那些元组密级小于等于 c 的元组才能够参加满足谓词 p 条件的计算。对每个实体至少有一个元组 $t' \in r$ 满足条件 p ,那么由上面操作获得的 c -元组 t 可通过下面的步骤来完成整个操作。

第一步,首先计算继承元组 t 的各数据属性值、数据密级及元组密级:

1) $t[A_1, C_1] = t'[A_1, C_1]$;

2) $2 \leq k \leq n$,有:

(1)如果 A_k 在 GET 子句中,①若存在一个元组 t'' 满足 $t''[A_1, C_1] = t[A_1, C_1] \wedge t''[TC] = t''[C_k] = c_k$,令 $t[A_k, C_k] = t''[A_k, C_k]$;②若不存在一个元组 t'' 满足 $t''[A_1, C_1] = t[A_1, C_1] \wedge t''[TC] = t''[C_k] = c_k$,令 $t[A_k, C_k] = (\text{null}, c_k)$;

(2)如果 A_k 不在 GET 子句中,①若 $c \in [L_k \dots H_k]$,则 $t[A_k, C_k] = (\text{null}, c)$;②若 $c \notin [L_k \dots H_k]$,则 $t[A_k, C_k] = (\text{null}, \text{null})$ 。

第二步,检查元组 t 是否满足实体完整性。若不满足,继承失败。

第三步,若元组 t 中含有外码,检查 t 是否满足外码完整性和参照完整性。若不满足,继承失败。

第四步,检查元组 t 是否满足多实例完整性。若不满足,继承失败。

第五步,在从表中寻找是否存在满足 $t''[A_1, C_1] = t[A_1, C_1] \wedge t''[TC] = c$ 的元组 t'' ,

(1)若存在这样的元组 t'' ,那么首先用元组 t 替换 t'' ;然后,对于从表中任何满足 $t'''[A_1, C_1] = t[A_1, C_1] \wedge t'''[TC] > c \wedge t'''[C_k] = c$ 的元组 t''' ,如果 $t[A_k, C_k] \neq t'''[A_k, C_k]$,则令 $t[A_k] = \text{null} (2 \leq k \leq n)$;

(2)若不存在这样的元组 t'' ,那么就将元组 t 添加到从表中。

如果 NMD 的主、从表如表 4、表 6 所示,一个 S-主体试图从 Name 为“长城”的 M1-元组继承 Mission 的值和从 Name 为“长城”的 M2-元组继承 Destination 的值,PUPDATE 操作后的主、从表分别如表 4、表 7 所示。

表 6 NMD 的从表

Name	Mission	Destination	TC
长城 U	观光 M ₁	月球 U	M ₁
长城 U	空间探索 U	火星 M ₂	M ₂

表7 PUPDATE操作后 NMD的从表

Name	Mission	Destination	TC
长城 U	观光 M ₁	月球 U	M ₁
长城 U	空间探索 U	火星 M ₂	M ₂
长城 U	观光 M ₁	火星 M ₂	S

4.3 更新操作

在c-主体的更新操作中,只有那些元组密级等于c的元组才能够参加满足谓词p条件的计算。对于满足谓词条件p的那些元组t,将通过如下步骤完成更新操作:

第一步,如果外观主码A₁中没有属性在SET子句中,而A_k(2≤k≤n)在SET子句中:

(1)若A_k是外码,即是另一关系s的外观主码,检查关系s中是否存在元组密级不大于主体安全级并且外观主码值等于属性A_k更新后的值。如果不存在,更新失败。

(2)在主表中寻找符合条件p的元组。如果能找到符合条件的元组t,首先令t[A_k, C_k]=(s_k, c),然后在从表中寻找满足t'[A₁, C₁]=t[A₁, C₁]∧t'[TC]>c∧t'[C_k]=c的元组t',令t'[A_k]=s_k。

(3)在从表中寻找符合条件p的元组t,t[A_k, C_k]=(s_k, c),再从表中寻找满足t'[A₁, C₁]=t[A₁, C₁]∧t'[TC]>c∧t'[C_k]=c的元组t',令t'[A_k]=s_k。

第二步,如果外观主码A₁的一些属性在SET子句中,那么

(1)若A₁被另一关系引用,则更新失败。

(2)在主表中寻找符合条件p的元组。如果能找到这样的元组t,那么首先令在SET子句中的A_k的值为t[A_k, C_k]=(s_k, c)(1≤k≤n),然后将从表中所有满足t'[A₁, C₁]=t'[A₁, C₁]∧t'[TC]>c的元组t'删除掉。

(3)在从表中寻找符合条件p的元组。如果能找到这样的元组t(2≤k≤n):

①如果A_k不在SET子句中且t[C_k]<c,令t[A_k, C_k]=(null, c);②对于满足t'[A₁, C₁]=t[A₁, C₁]∧t'[TC]>c∧t'[C_k]=c的元组t',令t'[A_k]=null;

(4)除此之外,在从表中,如果存在符合条件p的元组t的A_k在SET子句中,那么t[A_k, C_k]=(s_k, c)(1≤k≤n)。

如果NMD的主、从表分别如表4、表7所示,S-主体试图更新“长城”号宇宙飞船的Destination为“木星”,那么更新后的主从表分别如表4、表8所示。在前面的基础上,若一个M₁-主体再试图更新“长城”号宇宙飞船的Mission为“间谍”,那么更新后的主从表分别如表4、表9所示。

4.4 删除操作

在c-主体的删除操作中,只有那些元组密级等于c的元组才能够参加满足谓词p条件的计算,也就是说条件谓词p可以用p∧t[TC]=c来替代。对于满足谓词条件p的那些元组t,将通过下面的步骤完成删除操作:

第一步,在主表中寻找所有符合删除条件p的元组。若能找到这样的元组t,那么首先删除从表中所有满足t'[A₁, C₁]=t[A₁, C₁]∧t'[TC]>c的元组t'后,然后再删除t。

第二步,在从表中寻找所有符合删除条件p的元组。若能找到这样的元组t,那么首先在从表中寻找所有满足t'[A₁, C₁]=t[A₁, C₁]∧t'[TC]>c∧t'[C_k]=c(2≤k≤n)的元组t',令t'[A_k]=null;然后再删除t。

第三步,如果t的外观主码是另外一个关系s的外码,删除t的同时删除关系s中以元组t的外观主码为外码且元组

密级等于c的所有元组。

如果NMD的主、从表分别如表4、表9所示,M₁-主体试图删除Name为“长城”的宇宙飞船的记录,那么删除后的主从表分别为表4、表10。

表8 UPDATE操作后 NMD的从表

Name	Mission	Destination	TC
长城 U	观光 M ₁	月球 U	M ₁
长城 U	空间探索 U	火星 M ₂	M ₂
长城 U	观光 M ₁	木星 S	S

表9 UPDATE操作后 NMD的从表

Name	Mission	Destination	TC
长城 U	间谍 M ₁	月球 U	M ₁
长城 U	空间探索 U	火星 M ₂	M ₂
长城 U	间谍 M ₁	木星 S	S

表10 DELETE操作后 NMD的从表

Name	Mission	Destination	TC
长城 U	空间探索 U	火星 M ₂	M ₂
长城 U	Null M ₁	木星 S	S

总结 本文针对原主从结构表安全模型产生语义模糊性和操作不完备性等问题,增加了基元组和数据继承的概念,提出了一个新的、改进的主从结构表模型。重新定义了多实例完整性和参照完整性,将PUPDATE操作和数据继承完整性引入该模型,增强了模型的安全性和非二义性。在新的模型中,任何一个主体不仅能够“向下读”(即一个主体能够接受安全级等于和小于该主体密级的元组的数据),而且能够“向上写”(即发生在低密级数据上的改动能够直接地传播到高密级数据上去)。如果一个主体想在关系中添加一个元组,不仅INSERT操作能达到目的,PUPDATE操作也能完成此项工作。

对一个多级安全数据库系统来说,仅仅有强制存取控制是不够的,自主存取和审计也是必不可少的。另外,在以后的工作中,还可以将安全级定义在关系模式级上,对模式安全级、模式多实例及关系-关系之间的操作处理等进行深入研究。

参考文献

- 1 National Computer Security Center. Department of Defense Trusted Computer Security Evaluation Criteria. DOD 5200. 28-STD, December 1985
- 2 National Computer Security Center. Trusted Database Management System Interpretation of the Trusted Computer System Evaluation Criteria. NCSC-TG021, April 1991
- 3 Polyinstantiation Issues in MLS Database Management Systems. [NCSC Technical Report-005][S]. Volume 3/5, May 1998
- 4 Bell D E, Lapadula L J. Secure Computer Systems; Mathematical Foundations and Model. Bedford, Mass; The MITRE Corp, 1973
- 5 Polyinstantiation Issues in MLS Database Management Systems; [NCSC Technical Report-005][S]. Volume 3/5, May 1998
- 6 武立福,毛宇光.一种改进的多级安全关系数据模型.计算机应用,2003,23(7):102~105
- 7 Jajodia S, Sandhu R S, Blaustein B T. Solutions to the Polyinstantiation Problem. Abrams M, Jajodia S, Podell H, eds. Essay 21 of Information Security, An Integrated Collection of Essays. Los Alamitos, California, IEEE Computer Security Press, 1995
- 8 Chen F, Sandhu R S. The Multilevel Relational (MLR) Data Model; [Technical Report]. ISSE-TR-95-101. George Mason Univ, 1995