# 基于扩展客体层次结构的安全数据库策略模型*

程万军+, 张 霞, 刘积仁

(东北大学 计算机软件国家工程研究中心,辽宁 沈阳 110004)

# A Secure Policy Model for Secure Database System Based on Extended Object Hierarchy

CHENG Wan-Jun+, ZHANG Xia, LIU Ji-Ren

(National Engineering Research Center for Computer Software, Northeastern University, Shenyang 110004, China)

+ Corresponding author: Phn: 86-24-83665580, Fax: 86-24-86662821, E-mail: chengwj@neusoft.com

http://www.neu.edu.cn

Received 2002-06-06; Accepted 2002-11-22

**Cheng WJ, Zhang X, Liu JR. A secure policy model for secure database system based on extended object hierarchy. *Journal of Software*, 2003,14(5):955~962.**

http://www.jos.org.cn/1000-9825/14/955.htm

**Abstract**: Security policy model is the groundwork for secure or trusted system. Bell-LaPadula model with its good adaptability has comprehensive applications to multilevel security system, but it is short of the rules about integrity and consistency. Based on that model, an extended policy model is proposed, which is founded on the extended object hierarchy. By this way, the integrity becomes one of the inherence properties of the model. The object domains, extended security axioms and operation rules are also introduced or redefined. The proposed model more suits the requirements of multilevel security databases, and guarantees the consistency among policy model, system specification and other high-level security model. The extensions and enhancements, especially other properties besides security, are the necessary steps for transforming a policy model into a practical system.

**Key words**: security policy; security model; database security; multilevel security; object integrity; object hierarchy

**摘 要**: 安全策略模型是安全可信系统的基础.Bell-LaPadula 模型是多级安全系统中广泛应用的安全策略模型,但它缺乏针对数据模型的完整性和一致性规则.以该模型为基础,针对数据库系统的数据模型,提出了一个以扩展客体层次结构为基础的安全策略模型.模型通过扩展客体层次结构使完整性成为模型的内在属性,并引入或重新定义了客体域、扩展安全公理和操作规则.模型更加适应多级安全数据库系统的要求,增强了策略模型与系统规格和高层模型的一致性.普遍性和通用性安全模型的扩展和增强,特别是安全性以外的特性的引入是安全策略模型向实际系统模型转化的必要步骤.

**关键词**: 安全策略;安全模型;数据库安全;多级安全;客体完整性;客体层次结构

**中图法分类号**: TP311 **文献标识码**: A

* Supported by the National High-Tech Research and Development Plan of China under Grant No.863-301-6-5-B (国家高技术研究发展计划(863))

**第一作者简介**: 程万军(1973－),男,黑龙江佳木斯人,博士生,主要研究领域为信息安全,数据库技术.

形式化安全模型是高可信级别系统的基础和前提.安全系统评估的核心之一就是验证系统安全功能与安全策略模型的一致性.一个好的安全模型应该提供语义丰富的表达能力,以描述系统在功能上和结构上的特性,同时具备向系统数据模型、事务模型和系统实现进行一致转化的特性.数据库系统自身的完整性特征及其在信息系统结构中的特点决定了为高可信级别数据库管理系统所提供的安全模型需要解决的问题有:安全性与完整性的协调、安全模型的表达能力、模型与系统规格的一致性、模型描述的安全策略对应用策略乃至多策略的一致性表达和支持等.

遵循 TCSEC[1,2]或 CC[3]标准,对高级别可信(多级安全)数据库系统模型的研究主要有 3 个方向或层次:抽象安全策略模型、多级数据模型和安全事务模型.它们是上述评估标准中保证要求的必要环节,分别对应着可信 DBMS 安全工程过程中从系统分析、高层设计、低层设计到系统实现等不同阶段的阶段成果或工作参照物.它们是递进的研究层次,前者是后者的基础和前提,后者对前者具有反馈和激励作用.

从系统体系结构角度来看,数据库系统的安全体系结构是以操作系统(甚至包括硬件)为基础的混合 TCB(可信计算基)[1,2]或混合 TSF(评估对象的安全功能)[3]结构.相应地,系统的验证和评估需要分层化的方法学进行指导[2].安全数据库系统的研发需要从安全策略模型、数据模型、事务模型乃至应用安全模型等一系列相关模型的支持,并且各级模型间需要一致、无歧义的支持和继承机制.安全策略模型是其中最为基础的环节.

在以往的研究中,部分存在着过早地将安全策略模型与系统数据模型甚至事务模型相结合,将不同抽象层次的概念混合进行讨论的情况.这不仅不利于软件工程方法的实施和运用,也使得研究模型存在局限性,如模型中出现的不确定对象、不完备语义、安全策略折衷等.特别是由于所建立的模型中概念层次的不对等性和不完备性,使得它们在向实际系统转化时缺乏一致的方法以兼顾安全策略的完备性、数据的完整性和实际系统的功能性.这些问题的解决需要安全策略模型的良好和完备的定义[4~7].

针对上述问题和需求,本文提出了一个用于安全数据库系统的基于扩展客体层次结构的安全策略模型.下面首先分析可信 DBMS 安全策略的规格需求,然后介绍扩展客体层次结构模型,其后在扩展客体层次结构的基础上定义安全数据库系统的扩展安全策略模型,最后总结全文.

## 1 可信 DBMS 安全策略规格

数据库系统中完整性和安全性是两个不同的概念[5],但两者是密切相关的,特别是在系统实现方法方面,同一种机制常常既用于安全性保护亦用于完整性保护[5,8].这种相关性导致了系统安全性的强化可能造成与系统完整性的冲突或潜在违背.为保证逻辑的延续性和安全工程过程及方法的阶段性,安全数据库系统的安全模型应兼顾多级安全性和与之相应的完整性要求.只有这样,模型的可用性才能得到可靠保证.同时,为实现策略的一致支持,在通用安全策略模型的基础上进一步充实和扩展对主体和客体的约束以及访问控制规则是必要的.

概括而言,关系数据模型中完整性要求包括正确性(实体完整性)和一致性(参照完整性).实体和参照完整性是 DBMS 的特性.另一方面,因引入客体标记,客体标记本身提出了标记完整性的需求.

Bell-LaPadula(简称 BLP)模型[4]是在多级安全系统中广泛应用的通用安全策略模型.BLP 模型的客体结构是一种松散的结构,缺乏必要的约束.而应用于多级安全数据库的多级数据模型,如多级关系模型和多级对象模型显然是在这个松散结构上的逻辑增强,但它们与经典安全策略模型之间缺乏必要的结构和逻辑上的过渡.另一方面,BLP 模型没有也无法覆盖数据库系统及复杂并发系统中的事务性特征,而且在致力于敏感数据存取控制时遗留了一些涉及系统完整性的重要问题没有解决,特别是未对完整性设定规则.当其应用于控制粒度更细的数据库系统时,这些问题尤为突出.对于模型的不足,一些研究者认为这也许是一个优点,因为这允许人们在其宽泛的轮廓下进行系统建模,并专注于具体的实现细节[5].

本文提出基于 BLP 的扩展模型 BLP-i,“i”的含义即是模型的完整性特征.模型在策略抽象层次上扩展并完整定义了扩展客体层次结构及其内在相容性,同时对 BLP 模型的访问控制规则也进行了必要的扩展,使模型具备了在策略抽象层次上表达客体完整性的能力,使之增强了通过一致性方法向数据模型和事务模型[9]的转化,特别是向数据模型中的完整性属性一致转化的可能性.同时,扩展客体层次结构也可为更多安全模型提供基础支持.

## 2 扩展客体层次结构

安全策略模型一般由主体、客体和访问规则 3 个要素构成.在以往基于 BLP 的模型中,主体及访问规则都被详细地加以研究,而对客体的抽象一般仅限于树状层次结构.这也是其局限性的原因之一.正如在数据模型中那样,客体的完整性往往蕴涵在其结构中.而对安全策略模型的客体结构进行更详实的抽象和分析是建立 BLP-i 模型的基础之一.

### 2.1 扩展客体层次结构的定义

**定义 1**. 集合 $ID$ 是标识的集合,$SL$ 是系统安全级别的集合,客体标识符集合定义为 $OID:=(ID\times SL)$,元组 $oid=(id,u)\in OID$ 惟一确定安全级别为 $u$ 的客体.

[客体标识符完整性]客体 $O_i$ 满足标识符完整性,当且仅当其客体标识符 $oid_i=(id_i,u)$满足:

① 属性 $A_j\in id_i\Rightarrow A_j\neq$null;

② 属性 $A_j,A_k\in id_i\Rightarrow SL(A_j)=SL(A_k)=SL(id_i)$;

③ $A_j\in H(O_i)\wedge A_j\notin id_i\Rightarrow SL(A_j)\geq SL(id_i)$.

该完整性保证了客体标识符必须是有意义的;对某特定级别的主体,客体标识符对其或者完全可见,或者完全不可见;客体属性(外延)的安全级别支配其先驱(内涵)的安全级别.该属性与 SeaView 模型和多级关系的实体模型[10]的实体完整性属性在概念上是类似的.但该属性是安全策略模型层次上的客体完整性,而非数据模型意义上的数据实体完整性.其中,$H(O_i)$表示 $O_i$ 在客体层次结构中的相应结点,$SL(X)$表示客体 $X$ 的安全级别函数.

**定义 2**. $H\in P(\underline{O})^{\underline{O}}$表示客体的层次结构,它是一棵带根的有向树,$H$ 满足如下属性:

**性质 1**. 客体标识完整性.

**性质 2**. $!\exists\{O_1,O_2,\ldots,O_w\}\subseteq\underline{O}[\forall r(1\leq r\leq w\Rightarrow O_{r+1}\in H(O_r))\wedge(O_{w+1}\equiv O_1)]$.

[基本相容性]$\forall O_i\in\underline{O}(\exists O_j\in\underline{O}\wedge O_j$ 是 $O_i$ 的父结点$\Rightarrow SL(O_j)\geq SL(O_i))$.

性质 1 和性质 2 保证了客体在 $H$ 中的惟一性且 $H$ 是无环的.基本相容性属性即为经典 BLP 模型客体层次结构的相容性,它要求树中某结点的安全级别必须支配其父结点安全级别,保证客体按粒度方向的安全级别递增,使得持有某一许可的主体在沿 $H$ 中的有向路径访问时只能访问部分信息,执行“need-to-know”原理.

客体的安全级别是客体的属性,它可作为客体的直接后继结点来表达,也可通过客体结点自身的某个内在属性来表达,具体表达方式需由客体层次结构所支持的数据模型来决定.

**定义 3**. 对于结点 $P_i\in H$,子树 $ST(P_i)$为以 $P_i=H(O_i)$为根且包含其所有后继结点的有向树.

子树的最大下界定义为子树中所有结点(包含子树的根)安全级别的最大下界,记为 $GLB(ST(P_i))$.

**约定**. 客体层次结构 $H$ 中,根结点的安全级别由系统管理员(或系统安全管理员)设定,其安全级别的修改由系统 TCB 控制和维护[1,2].若某结点或其部分子结点未显式定义安全级别,则它们各自的隐含安全级别通过如下规则确定:

① 若结点为叶结点,则其继承父结点的安全级别;

② 若结点非叶结点,则其安全级别为以它为根的子树的最大下界;

③ 隐含安全级别随其依赖的其他客体的安全级别变化而更新.

**定理 1**. 依据上述规则确定的结点隐含安全级别满足基本相容性属性.

证明:设结点 $P$ 的父结点的安全级别为 $a$,以 $P$ 为根的子树的最大上界为 $b$.依据基本相容性属性,$P$ 的任一后继结点 $C$ 的安全级别 $SL_C\geq a$,所以 $a$ 是 $P$ 的所有后继结点安全级别的下界.根据最大下界函数定义,有 $b\geq a$.因此,$H$ 中的基本相容性得到满足. □

**定义 4**. 子树 $ST(P_i)$的安全级别记为 $SL(ST(P_i))$,它可被显式地定义为子树根结点的内在属性或子结点.

未显式定义安全级别的子树,隐含子树安全级别确定规则为:子树安全级别为子树中所有结点(包括根)安全级别的最小上界 $LUB(ST(P_i))$.下面定义针对子树安全级别的扩展相容性属性.

[扩展相容性]子树安全级别支配子树中所有结点的最小上界.

在客体层次结构中,子树的安全级别用来控制主体对子树的访问,该属性保证了只有持较高许可(支配子树

安全级别)的主体才能遍历整个子树.根据上述定义,显然有 $SL(ST(P_i))\geq SL(H(O_i))$.

下面以更抽象和一般性的方式定义和说明数据模型意义上的实体在扩展客体层次结构下的表达问题.

**约定**. 在客体层次结构 $H$ 中,实体通过 $H$ 中的某一结点或某一子树来表达.同样,实体在 $H$ 中表达的具体形式依赖于具体数据模型和数据模型所希望表达的客体特性.对上述两种实体表达方式,进一步要求:

① 若实体表达为 $H$ 中的一个结点,则该结点的 *oid* 被继承为该实体的实体标识符;

② 若实体表达为 $H$ 中的一棵子树,则该实体继承子树的安全级别和子树根结点的标识符.

基于上述约定,数据模型意义上的实体是将要定义的策略模型意义上客体的集合,实体表达为一个结点的情况被看作是一个特例;表达某数据模型意义上的实体的子树还可以包含表达另一层次实体的子树,即上述客体层次结构中的嵌套情形.

[单结点多实例]对 $H$ 中的结点 $P_i$,若有 $P_i$ 的两个直接后继 $P_j,P_k\in ST(P_i)$满足 $id(P_j)=id(P_k)$,且 $SL(P_j)\neq SL(P_k)$,则产生单结点多实例,称 $P_j$ 和 $P_k$(兄弟关系)互为多实例结点,称它们所对应的客体 $O_j$ 和 $O_k$ 为单结点多实例客体.

[子树多实例]对 $H$ 中的结点 $P_i$,若以 $P_i$ 的两个直接后继 $P_j,P_k\in ST(P_i)$为根的两个兄弟子树 $ST(P_j)$和 $ST(P_k)$满足 $id(ST(P_j))=id(ST(P_k))$且 $SL(ST(P_j))\neq SL(ST(P_k))$,则发生子树多实例情况,称 $ST(P_j)$和 $ST(P_k)$互为 $P_i$ 下的多实例子树.

虽然单结点客体多实例可以看作子树多实例的特例,但在抽象数据层次结构所支持的数据模型上,它们存在着具体实体对象和逻辑含义的差别.如在多级关系模型上,它们可以为数据模型描述不同层次数据实体的不同粒度的多实例,而且数据模型中不同的多实例情况往往要求不同的完整性约束[8,11].

[单结点多实例完整性]对 $P_i\in H,P_i$ 满足单结点多实例完整性,当且仅当:对 $P_i$(所对应的客体为 $O_i$)的任一内在属性 $A_i\notin id(P_i)$,有 $id(P_i),SL(P_i)\rightarrow A_i$,即 $id(P_i),SL(P_i)\rightarrow P_i$.

[子树多实例完整性]对非叶结点 $P_i\in H,ST(P_i)$满足子树多实例完整性,当且仅当:

① 对 $P_i$ 下任一直接子树 $ST(P_j)$,有 $oid(P_j),SL(ST(P_j))\rightarrow ST(P_j)$或 $id(P_j),SL(P_j),SL(ST(P_j))\rightarrow ST(P_j)$;

② $ST(P_j)$满足子树多实例完整性;

③ $ST(P_i)$的叶结点满足单结点多实例完整性.

上述的两个多实例完整性定义是前面客体标识符和子树标识符以及子树安全级别定义的一致性形式转换,在此基础上可以定义抽象客体层次结构上的更一般的多实例完整性.

[客体层次结构的多实例完整性]客体层次结构 $H$ 满足多实例完整性,当且仅当:

① $\forall P_i\in H\Rightarrow P_i$ 的所有后继结点满足单结点客体多实例完整性;

② $\forall P_i\in H\Rightarrow P_i$ 下所有以 $P_i$ 的后继结点为根的子树满足子树多实例完整性.

[空结点完整性(空值完整性)]在客体层次结构中,空结点(null)只可以出现在叶结点上.

根据上述客体层次结构的定义,在模型中可能存在 4 种客体间的参照关系:结点对结点(记为 *nn*)、结点对子树(记为 *nt*)、子树对结点(记为 *tn*)和子树对子树(记为 *tt*).模型中,将它们统称为客体层次结构中的客体参照.参照关系是在作用于系统客体上的不同抽象层次的规则"引导"下发生的,这些规则可能是用户定义的,也可能是客体逻辑或结构关系隐含具有的.模型分别用 $RR_{nn},RR_{nt},RR_{tn}$ 和 $RR_{tt}$ 表示上述 4 种参照情形中各种参照规则的集合.设 $RR=RR_{nn}\cup RR_{nt}\cup RR_{tn}\cup RR_{tt}$.下面具体定义各种参照情况.

**定义 5**. 在客体层次结构 $H$ 中,$rr$ 是作用在 $P_i,P_j\in H$ 上或以它们为根的子树上的参照规则,

- 若 $rr\in RR_{nn}$,称 $P_i$ 参照 $P_j$,当且仅当$\exists$内在属性 $A_i\in P_i\Rightarrow A_i=id(P_j)\wedge SL(P_i)\geq SL(P_j)$,称 $A_i$ 为参照标识符;
- 若 $rr\in RR_{nt}$,称 $P_i$ 参照子树 $ST(P_j)$,当且仅当$\exists$内在属性 $A_i\in P_i\Rightarrow A_i=id(ST(P_j))\wedge SL(P_i)\geq SL(ST(P_j))$,称 $A_i$ 为参照标识符;
- 若 $rr\in RR_{tn}$,称子树 $ST(P_i)$参照 $P_j$,当且仅当$\exists P_i$ 的直接后继 $P_k\Rightarrow id(P_k)=id(P_j)\wedge SL(ST(P_i))\geq SL(P_j)$,称 $P_k$ 为参照标识符;
- 若 $rr\in RR_{tt}$,称子树 $ST(P_i)$参照 $ST(P_j)$,当且仅当$\exists P_i$ 和 $P_j$ 的直接后继 $P_k,P_m\Rightarrow id(P_k)=id(P_m)\wedge SL(ST(P_i))\geq SL(ST(P_j))$,称 $P_k$ 为参照标识符.

用 *rid* 来表示上述参照客体的参照标识符.由子树安全级别的定义、上述定义隐含要求参照标识符的安全

级别支配被参照客体标识符.

[参照标识符完整性]在 $rr\in RR$ 作用下参照客体的参照标识符需满足:

① 或者为空(作为叶结点),或者在 $rr$ 的作用下存在满足定义的被参照客体结点(或子树);

② 参照标识符的所有内含属性具有相同安全级别.

模型只定义了结构可视化的显式参照(由用户指定或逻辑隐含的).对模型客体结构不可视(结构自身无法表达)的参照关系,需要由执行特定逻辑的高层系统主体维护,如文件内容间的参照等情况.即本模型所提供的控制是基于客体层次结构及其所定义的规则的.客体结点内在属性 $A_i$ 是否可见,或称模型是否可读取该属性是由模型的相应策略以及所支持的数据模型决定的.

[参照完整性]$H$ 满足参照完整性,当且仅当对任一 $rr\in RR$,在 $rr$ 作用下的参照客体对 $X$ 和 $Y$($X$ 参照 $Y$),$X$ 的参照标识符 $rid(X)$不为空,则有 $rid(X)=id(Y)\wedge SL(rid(X))\geq SL(id(Y))\wedge SL(rid(X))\geq SL(Y)$.

简单来说,参照关系需要满足:参照客体的安全级别支配被参照客体的安全级别,即参照关系也遵守“下读”规则.也就是说,在参照路径上保持客体安全级别的“递增”即相容性.这种参照关系的例子有 RDBMS 中关系间的参照关系、OODB 中对象间的参照等.

### 2.2 扩展客体层次结构的解释

上一节从独立于数据模型的角度分析和模型化了安全策略模型的扩展客体层次结构.相对于 BLP 模型,它弥补了其在客体表达能力,特别是在客体约束和完整性上的不足,缩减了安全策略模型和数据模型以及其他低层次安全模型间的概念差异.扩展客体层次结构在逻辑上更为贴近数据模型,这有利于采用一致的方法将安全策略模型转化为数据模型乃至事务模型,并且可以支持多种数据模型,如:关系、面向对象和半结构化等数据模型.该结构使客体在策略模型层面呈现更紧密的联系.

相对于安全性,完整性更是客体的内在本质属性.扩展客体层次结构增强了客体完整性和相容性(实质上也是完整性)要求.显然,这些新引入的要求可以根据需要进行增减,以适应数据模型和实际系统的需要.同时,它们丰富了模型自身的表达能力,为多策略的集成提供一致的基础结构.以关系数据模型为例,扩展客体层次结构中的完整性和相容性可以一致地转化为关系数据库中的实体完整性、多实例完整性、参照完整性等属性,并且有利于更好地分析和消除在多级数据模型或其他低层安全模型中出现的不确定性,帮助系统设计开发者更好地掌握和利用模型和系统中可能出现的复杂情况.限于篇幅,这些问题将另文详述.

在本文中,仍然有条件地采用乐观的观点,即认为高层抽象模型的非确定性往往是模型表达能力的一种体现,并且这些非确定性可以被其他低层模型或者实际系统所利用,为丰富多样的应用需求提供支持.

## 3 基于扩展客体层次结构的安全数据库系统策略模型 BLP-i

### 3.1 模型概述

#### 3.1.1 客体隔离域

客体层次结构被划分为 4 个基本隔离域:系统域 $D_s$、责任域 $D_a$、元数据域 $D_m$ 和用户域 $D_u$.系统域用于维护系统级数据客体或对象,它们是与系统 TCB 和核心相关的客体.责任域客体是有关系统责任保证(accountability)的客体.元数据域中的客体是关于用户域客体的定义、结构和控制数据的.用户域用于维护、容纳用户数据客体,它被进一步划分为常规用户域 $D_{gu}$ 和完整性维护域 $D_{iu}$,完整性维护域内的客体是用于维护客体层次结构相容性而需特殊处理的用户客体.

在多级环境下,被参照客体的删除往往导致隐蔽通道,已提出的解决方案为以标记表达式或复合标记为基础的只读升级或规范标记更新策略[5,6].它们的共同点是为避免隐蔽通道“升级”客体标记,使其成为只读客体(客体存在的原因是为了维护完整性约束).事实上,这些具体方案的概括即为将这类客体进行特殊处理,归入特殊的客体类型.完整性维护域 $D_{iu}$ 的目的即是如此.在上述的上下文环境中,也可称其为只读域.

客体层次结构上的域是一种逻辑划分,$H$ 中的 4 个基本域保持相对的隔离.系统对域的基本访问规则为:对不同域内客体对应结点的访问,需要请求主体持有与具体访问方式相关的、适当的安全级别和授权.

3.1.2 系统状态

系统状态是集合 $V=(B\times M\times F\times H)$中的元素,其中 $B\subseteq(\underline{S}\times\underline{O}\times A)$为当前存取集,$\underline{S}$为主体的集合,$\underline{O}$为客体的集合,$A=\{\underline{r},\underline{w},\underline{e},\underline{a},\underline{c},\varnothing\}$为访问权限集合,其中各元素表示读、写、执行、添加、控制和空.控制操作意味着主体可以将其所具有的上述 4 种访问属性中的一个或者多个授予给其他主体.$M$ 为存取控制矩阵,$M_{ij}$ 表示主体 $S_i$ 对客体 $O_j$ 访问权限.$F=\{(f_S,f_O,f_C)|f_S\in\underline{L}^S\wedge fo\in\underline{L}^O\wedge fc\in\underline{L}^S\wedge(\forall S\in\underline{S}(f_S(S)\geq f_C(S)))\}$为敏感标记函数集合,$f_S$ 为主体敏感标记函数,$f_O$ 为客体敏感标记函数,$f_C$ 为主体当前敏感标记函数.$H$ 即为前面定义的客体层次结构.在模型中,用 $\underline{S}_T$ 表示可信主体的集合,非可信主体集合 $\underline{S}_U=\underline{S}-\underline{S}_T$.

3.1.3 状态转换规则

系统状态间的转换由系统规则定义,各规则是对任意状态下输入(请求)的输出(判断)和下一状态的函数.规则定义为 $\rho:R\times V\to D\times V$,$R$ 为请求集合,$D=\{$yes,no,error,?$\}$为判定集合,$V$ 为状态集合.

规则 $\rho$ 保持安全状态,当且仅当 $\rho(R_k,v)=(D_m,v^*)$,则当 $v$ 是安全状态时,$v^*$也是安全状态.

### 3.2 模型公理

模型的安全公理在模型中用来裁决主体对客体的访问请求,因此也称为安全规则.

**公理 1(简单安全特性(*ss*-特性))**. 状态 $v=\{b,M,f,H\}$满足 *ss*-特性,iff $S\in\underline{S}\Rightarrow[(O\in b(S:\underline{r},\underline{w}))\Rightarrow(f_S(S)\geq f_O(O))]$.

**公理 2(*-安全特性(*-特性))**. 状态 $v=\{b,M,f,H\}$满足*-特性,iff $S\in\underline{S}_U\Rightarrow$

(1) $O\in b(S:\underline{a})\Rightarrow f_O(O)\geq f_C(S)$;

(2) $O\in b(S:\underline{w})\Rightarrow f_O(O)=f_C(S)$;

(3) $O\in b(S:\underline{r})\Rightarrow f_C(S)\geq f_O(O)$.

**公理 3(自主安全特性(*ds*-特性))**. 状态 $v=\{b,M,f,H\}$满足 *ds*-特性,iff $(S_i,O_j,\underline{x})\in b\Rightarrow\underline{x}\in M_{ij}$.

**客体域隔离公理**.

对属于不同客体域的客体只能由相应的持有适当许可和特权的主体访问.同时,在安全级别配置上,$D_s$ 域安全级别与 $D_a$ 域安全级别不可比较;$D_m$ 域比 $D_u$ 域有较高的安全级别;相对于 $D_s$ 域、$D_a$ 域和 $D_m$ 域,$D_u$ 域具有较低安全级别;$D_{iu}$ 和 $D_m$ 中客体的写操作需由系统可信主体完成.

客体域的引入,限制了 BLP 模型的可信操作(由可信主体执行),使系统更为全面地贯彻最小特权规则.这实际上约束了可信操作的绝对性,即系统中没有完全可信的主体.这是最小特权原则所要求的,是对权限分离、责任隔离原则的贯彻.

**激活性公理**.

激活性公理用于约束 $H$ 中客体的创建、激活和删除.

(1) 新建客体安全标记继承规则:对主体 $S$ 所创建的客体 $O$,有 $f_O(O)=f_C(S)$.

(2) 客体稳定规则:客体安全标记只可在客体被完全控制的情况下,由授权主体遵循访问规则来进行修改.

(3) 新建客体重写规则:每个新建客体被赋予一个与以前任何状态无关的初始状态.

(4) 未激活客体的不可存取性:不能通过授予在 $H$ 中没有出现的未激活客体相应操作权限而使其可存取.

(5) 若 $D_{iu}$ 内某客体置为非激活状态不破坏客体层次结构的完备相容性,则该客体可被系统主体删除.

(6) 已删除客体的不可存取性:对所有已从 $H$ 中删除的客体 $O\in\underline{O}$,有$(S,O,\underline{x})\notin B$.

### 3.3 安全系统定理

系统是安全系统,当且仅当:初始状态是安全状态,并且每次状态转换都满足 *ss*-特性和*-特性,并且客体层次结构始终保持完备相容性.

上述安全系统定理是在 BLP 模型安全系统定理上附加客体结构完备相容性的结果.该定理进一步扩展了安全系统的定义,使系统的安全性和完整性同时得到保证,同时限制了(存取)隐蔽通道的产生.

### 3.4 操作规则

模型利用前面定义的扩展客体层次结构重新定义了 BLP 模型中的操作规则 8 和规则 9.对其他规则,需要

将扩展客体层次结构的完备相容性引入到每个操作规则的执行条件中,而使这些规则继续沿用.

为描述方便,定义 $A(M_i)=\{j:1\le j\le m|S_i\in\underline{S},O_j\in\underline{O}\rightarrow M_{ij}\neq\varnothing\}$.对 $j\in A(M_i)$,它表示主体 $S_i$ 对 $OID$ 为 $oid_{oj}$ 的客体 $O_j$ 是可存取的.$Ref(RR)=\{j|rr\in RR,O_j\neq O_k,O_k$ 依据 $rr$ 参照 $O_j\}$.用 $H\oplus H(O_j)$表示在 $H$ 中添加 $O_j$ 的对应结点;$H\Theta ST(O_j)$表示在 $H$ 中删除以 $O_j$ 为根的子树.

令 $R8=\underline{S}\times RA\times\underline{O}\times\underline{O}\times X$,其中 $X=\{\varnothing,\underline{e}\}$,$R_k=(S_i,rq,O_j,O_p,x)$是 $R8$ 的元素,$H(O_p)$是 $H(O_j)$的父结点.

**规则8**. create-object:$\rho 8(R_k,v)\equiv$

if $R_k\notin R8$ or $S_i=\varnothing$ or $rq\neq\underline{c}$ or ($x\neq\underline{e}$ or $\varnothing$) then

$\rho 8(R_k,v)=(?,v)$;

if $j\in A(M_i)$ then

$\rho 8(R_k,v)=(\text{no},v)$;

if $\underline{a}\in M_{ip}$ and $H\oplus H(O_j)$满足完备相容性 then

if $f_C(S_i)\ge f_O(O_p)$ and $f_O(O_j)\ge f_O(O_p)$ and $f_C(S_i)=f_O(O_j)$ then

if $x=\varnothing$ then

$\rho 8(R_k,v)=(\text{yes},(b,M|M_{ij}\cup[\underline{r},\underline{w},\underline{a},\underline{c}],f\cup(O_j,f_O(O_j)),H\oplus H(O_j)))$;

else $\rho 8(R_k,v)=(\text{yes},(b,M|M_{ij}\cup[\underline{r},\underline{w},\underline{a},\underline{c},\underline{e}]_{ij},f\cup(O_j,f_O(O_j)),H\oplus H(O_j)))$;

else $\rho 8(R_k,v)=(\text{no},v)$;

end.

令 $R9=\underline{S}\times RA\times\underline{O}\times\underline{O}\times X$,$R_k=(S_i,rq,O_j,O_p,x)$为$R9$的元素,$H(O_p)$是$H(O_j)$的直接前驱.

**规则9**. delete-object:$\rho 9(R_k,v)\equiv$

if $R_k\notin R9$ or $S_i=\varnothing$ or $rq\neq\underline{d}$ or $x\neq\varnothing$ then

$\rho 9(R_k,v)=(?,v)$;

if $\underline{c}\notin M_{ij}$ or $\underline{w}\notin M_{ip}$ then

$\rho 9(R_k,v)=(\text{no},v)$;

if $f_C(S_i)=f_O(O_j)$ then

if $j\notin Ref(RR)$ then

$\rho 9(R_k,v)=(\text{yes},(b-\text{ACCESS}(O_j),(M|M_{tw}\leftarrow\varnothing,1\le t\le n,O_w\in INFERIOR(O_j)),f,H\Theta ST(O_j)))$;

else

$ST(O_j)$移入$D_{iu}$;

$\rho 9(R_k,v)=(\text{yes},(b-\{S_i,O_j,\underline{x}\},(M|M_{iw}\leftarrow\varnothing,O_w\in INFERIOR(O_j)),f,H))$;

else $\rho 9(R_k,v)=(\text{no},v)$;

end .

其中,$INFERIOR(O_j)=\{O_k|O_k\in ST(O_j)\}$,$ACCESS(O_j)=(S\times INFERIOR(O_j)\times A)\cap b.M|M_{tw}\leftarrow\varnothing$表示对 $M$ 中的$[t,w]$项置空并替代原 $M$,$M|M_{ij}\cup[\underline{x}]$则表示用指导元素填充 $M$ 的$[i,j]$项并替代原 $M$.

规则 8 要求在新建客体时,主体需对新建客体的父结点具有 $\underline{a}$ 权限,并且新建客体的引入不会破坏客体层次结构的完备相容性.规则 9 要求在删除客体时,主体需对客体有 $\underline{c}$ 或 $\underline{w}$ 权限,同时客体的去除不会破坏客体层次结构已有的完备相容性;若客体删除将导致参照完整性冲突,则该客体(及下属)将移入 $D_{iu}$ 而成为只读客体.扩展客体层次结构的引入,使得系统规则在处理机密性的同时维护了完整性.

**定理 2**. 规则8是*ss*-特性保持和*-特性保持的.

**定理 3**. 规则9也是*ss*-特性保持和*-特性保持的.

限于篇幅,定理的证明不再给出,参照 BLP 模型的证明和本文的约定,上述定理不难得证.

## 4 结束语

普遍和通用性安全模型的扩展和增强,特别是安全性之外的特性的引入是安全策略模型向实际系统模型

转化的必要步骤.BLP-i 模型是基于 BLP 模型建立的以数据库管理系统为应用对象,以系统标记保护级和结构化保护级为目标,适应混合TCB结构和客体完整性要求的扩展模型.模型对客体层次结构进行了更详实的定义,将抽象客体层次的参照关系和相应的完整性属性引入客体层次结构,使完整性属性成为安全策略模型自身的内在特征和属性.另外,对模型的公理和操作规则进行了重新定义,并验证了所提出的模型是多级安全的.

从安全策略模型到数据模型再到事务模型,这是可信数据库管理系统生命周期的必然过程,它们在所属的研究和设计阶段各自需要不同的分析方法和研究体系,具有不同的侧重.以抽象的安全策略模型结合数据库系统的特点及其数据模型来构造支持可信 DBMS 研究、设计和开发的安全数据模型,是安全数据库领域内的普遍研究方法.在本文提出的模型的基础上定义完备的 Relabel 策略和操作以及实用化的安全标记表达等都是安全策略模型进一步研究的重点.对多级数据模型和多级事务模型的研究将是我们未来的挑战性工作.

**References**:

[1] Department of Defense Standard. Department of defense trusted computer system evaluation criteria. DOD 5200.28-STD, 1985.

[2] National Computer Security Center. Trusted database interpretation of the trusted computer systems evaluation criteria. NCSC-TG-021, National Computer Security Center, 1991.

[3] The International Organization for Standardization. Common criteria for information technology security evaluation. ISO/IEC 15408:1999(E), 1999.

[4] Bell DE, LaPadula LJ. Secure computer systems: Unified exposition and multics interpretation. Technical Report, MTR-2997, Bedford, MITRE Corporation, 1976.

[5] Gong L, Qian XL. Enriching the expressive power of security labels. IEEE Transactions on Knowledge and Data Engineering, 1995,7(5):839~841.

[6] Foley SN, Gong L, Qian XL. A security model of dynamic labeling providing a tiered approach to verification. In: Proceedings of the IEEE Symposium on Security and Privacy. Oakland, 1996. 142~153.

[7] Sandhu R, Chen F. The multilevel relational (MLR) data model. ISSE-TR-95-101, George Mason University, 1995.

[8] Pernul G, Tjoa AM, Winiwarter W. Modeling data secrecy and integrity. Data & Knowledge Engineering, 1998,26:291~308.

[9] Atluri V, Jajodia S, Keefe TF, McCollum C, Mukkamala R. Multilevel secure transaction processing: Status and prospects. Database Security X: Status and Prospects. London: Chapman & Hall, 1997. 79~98.

[10] Smith K, Winslett M. Entity modeling in the MLS relational model. In: Proceedings of the 18th VLDB Conference. Vancouver, 1992. 199~210.

[11] Sandhu RS, Jajodia S. Referential intergrity in multilevel secure databases. In: Proceedings of the 16th NIST-NCSC National Computer Security Conference. Baltimore, 1993. 39~52.