针对实时数据库中隐蔽通道问题采取的安全策略*)

罗 琼 张立臣
(广东工业大学计算机学院 广州510090)

摘 要 实时数据库的根本目标就是使满足截止时间的事务数为最大,因而,管理和支持优先考虑时间的事务处理是非常重要的。然而,随着实时计算的迅速发展,安全成了实时应用另一个必须考虑的问题。因此,在实时数据库系统中,提出一种结合实时性和安全性的方法是非常必要的。本文针对隐蔽通道讨论了安全问题。通过定义的CCF、DCCF变量对实时数据库中的并发控制算法进行扩展使其既满足时间限制条件又满足安全需求。陈述了消除隐蔽通道的非干预性原则,并对同时维持实时和安全需求的可行性进行了探讨。

关键词 隐蔽通道,Bell LaPadula,安全2PLHP,安全OPT,非干预性原则,部分安全

Security Policy for Covert Channel Problem of Real-Time Databases

LUO Qiong ZHANG Li-Chen
(Faculty of Computer, Guangdong University of Technology, Guangzhou510090)

Abstract The decisive objective of real-time database is to maximize the number of the transactions that complete before their deadlines, then management and transaction processing for real-time systems are essential in supporting time-critical applications. However, as real-time systems continue to evolve, security is another problem that must be considered in many real-time application. So security becomes another problem that must be considered. This paper discusses security problem according to covert channel, extends conurrency control algorithm in real-time database systems to satisfy both timing constraint and security constraint by defining CCF and DCCF variables. Demonstrating non-interference which can avoid covert channel and studying the feasibility of timeliness and security requirement maintained at the same time.

Keywords Convert channel,Bell_LaPadula,S2PLHP,SOPT,Non-Interference,Partial security

1 引言

随着计算领域的迅速扩张,航空业、国防、工业自动化、商业信息以及交通控制等系统所保存的大量信息需要被不同安全权限的多个用户共享。在这些应用中,事务和数据有不同安全级别,对直接/间接的非法信息访问提供防卫措施对实时数据库来说是非常关键的。

许多多级安全数据库系统运用基于Bell-LaPadula模式的强制访问控制机制。Bell-LaPadula是针对安全数据库系统中特定的访问规则而存在的其中一个重要概念模式。在此模式中,事务和数据存在几个安全级别,分配给事务的代表权限级别,给数据的代表安全级别,访问数据必须符合下列条件:

a) 简单安全属性:一个主体只有当它的权限大于等于它要求进行读访问客体的安全级别,才可以对客体进行读访问。

b) *-属性:一个主体只有当它的权限小于等于它要求进行写访问客体的安全级别后,才可以对客体进行写访问。

据此规则,一个事务不能直接访问高安全级别的任何数据,这2个属性确保了高、低访问级别主体之间无直接的信息流。但是,当数据库中的某个资源被两个不同访问级别的主体共享时就会产生隐蔽通道[2],其允许信息就间接地从高访问级别主体流向低访问级别主体。

2 安全控制因素

安全实时数据库系统的并发控制算法必须像使用事务截止时间解决数据冲突一样来使用安全级别,而且在解决安全冲突时也要考虑事务安全级别的不同。如果最高安全级别事务和另一最低安全级别事务间存在隐蔽通道,那么,此种情况下隐蔽通道造成的后果比2个邻近安全级别事务所造成的后果更大。为了更形象说明安全违背的后果,针对实时数据库系统引入"隐蔽通道属性"。

2.1 隐蔽通道属性

隐蔽通道属性:在同一数据上产生冲突的2个事务间安全级别相差越大,隐蔽通道造成的后果就越严重。也就是说,访问级别相差越大,维持安全性就越重要。如果2个冲突事务的安全级别处于2个极端,隐蔽通道造成的后果是最大的。如果这2个冲突事务为邻近2个安全级别,后果是最小的。

2.2 隐蔽通道因子

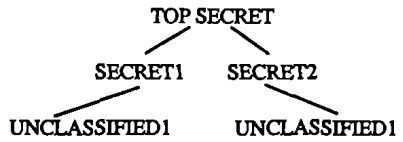
CCF(Covert Channel Factor)是衡量隐蔽通道的一个尺度。当2个事务的访问级别处于2个极端时,CCF的最大值为1。当2个事务的访问级别为相邻安全级别时,CCF最小。如为同一安全级别,则CCF为0,无隐蔽通道存在。

CCF = Difference in access levels/Maximum difference possible
= Difference in access levels/# of access levels-1

为了计算CCF,需要知道安全级别中每个可能对之间的次序。可以通过把部分次序转换为树获得事务间的CCF。对于诸如A>B的次序,A为父母节点,B为孩子节点。如其中一个节点已经存在,只产生新节点并和已经存在的节点构成

*)本文受国家自然科学基金(No.60174050)、广东省自然科学基金(No.010059)、教育部骨干教师基金、广东省"千百十"工程人才培养基金、广东省高校自然科学研究项目(No,Z03024)基金资助。罗琼 硕士研究生,主要研究方向为实时与分布式系统。张立臣 博士,教授,主要研究方向为实时与分布式系统。

层次结构。同一安全级别在处于树中同一层次。事务间安全级别差等于它们在树中深度差。设根节点深度最小，叶子节点深度最大。如果 depth(A)<depth(B)，则 A 的安全级别比 B 高(此工作假设在 MLS 模式下的部分次序可以转换为一棵单数)。例如，有5种安全级别 TOP SECRET、SECRET1、SECRET2、UNCLASSIFIED1、UNCLASSIFIED2。TOP SECRET>SECRET1、TOP SECRET>SECRET2，SECRET1>UNCLASSIFIED1、SECRET2>UNCLASSFIED2。根据上述规则，产生的树如下：



2.3 动态隐蔽通道因子

由于事务间可以间接地产生信息流，因此用 covert_channel_history(T)存储事务 T 以记录卷入隐蔽通道的事务 T 的历史记录来追踪产生隐蔽通道的源头。所有事务的初始 covert_channel_history(T)为0，TL、TH 分别为低、高安全级别事务。如 TL、TH 间存在信息流，则 Covert_channel_history(TL)= Covert_channel_history(TL)+ Covert_channel_history(TH)+access_level(TH)-access_level(TL)。

当 TL、TH 间产生隐蔽通道时，Covert_channel_history(TL)改变。如果在 T_1、T_2 间存在隐蔽通道且 level(T_1)>level(T_2)，那么，动态隐蔽通道因子 DCCF(Dynamic Covert Channel Factor)，DCCF={$AL(T_1)-AL(T_2)$+Covert_Channel_History(T_1)}/{# of access level-1}。

3 安全并发控制算法

传统乐观算法，事务的读、写操作无任何限制条件。在提交之前事务只要通过校验测试以检查当前是否存在执行冲突事务，如果校验失败，那么，校验事务被重启。否则，提交校验事务。

3.1 安全2PLHP(S2PLHP)

给定事务 T_1、T_2，T_1要求锁，T_2拥有锁。如果 $D(T_1)<D(T_2)$终止 T_2，否则终止 T_1。

对于安全实时数据库来说，可能发生5种冲突。

·如果 $D(T_1)>D(T_2)$且 $AL(T_1)>AL(T_2)$，则采取终止拥有低优先级和高安全级别的请求者 T_1，这样不会违背优先级或产生隐蔽通道的用得优先级和安全都得到维持。

·在 $D(T_1)<D(T_2)$且 $AL(T_1)<AL(T_2)$满足时，如果 $DCCF>CCF$，采用终止 T_2而维持安全，否则组塞 T_1以维持优先级。

·在 $D(T_1)<D(T_2)$ 且 $AL(T_1)>AL(T_2)$满足时，如果 $DCCF>CCF$，终止 T_1而维持安全，否则终止 T_2以维持优先级。

·在 $D(T_1)<D(T_2)$ 和 $AL(T_1)<AL(T_2)$满足时，终止 T_2，同时安全和优先级都得到满足。

·上述4种情况都不满足且 $AL(T_1)=AL(T_2)$时，如果 $D(T_1)<D(T_2)$，则终止 T_2，否则组塞 T_1。

根据安全2PLHP，CCF 越小，安全性就越重要，通过为 CCF 选择一个恰当的值，系统完全可以避免隐蔽通道的产生。因此，CCF 为系统中安全和优先级的控制提供了一个方法。其中 D:Dealline，AL:access level. VT:valid transaction。

3.2 安全 OPT(SOPT)

实时乐观算法的主要目标是阻止和高优先级存在冲突的低优先级有效事务。针对实时事务，有2个不同的乐观策略：优先级牺牲；优先级等待。在优先级牺牲中，重启与更高优先级事务存在冲突的事务。在优先级等待中，与更高优先级事务存在冲突的事务等待更高优先级事务的完成。此算法并没有检查事务安全级别。因此，当有效事务提交和终止冲突集合中的事务时就会产生隐蔽通道。如，在优先级牺牲算法中，如果有效事务访问级别和优先级都高过在冲突集合中其他任何事务，那么，当有效事务提交终止所有的冲突事务时，就产生隐蔽通道。同样，在优先级等待算法中，如果有效事务被阻塞并存在一个高访问级别的冲突事务，也会产生隐蔽通道。下面是针对此问题的解决方法：

1)处于冲突集合中的任意事务 T，如果 $D(T)>D(VT)$且 $AL(T)>AT(VT)$，提交 VT 并终止冲突集合中的所有 T。

2)处于冲突集合中的任意事务 T，如果 $D(T)<D(VT)$且 $AL(T)<AT(VT)$，阻塞具有最低优先级和最高安全级别 VT。

3)当1)、2)都不满足时，被组塞还是被终止都会产生隐蔽通道。因而，如果一些校验阶段处于此情况且冲突集合中不止一个事务，系统不能100%保证不产生隐蔽通道。

a)最小化 CCF。当有效事务和冲突集合中事务被终止，计算 CCF。比较这2CCF 以便查看哪个选择会更导致隐蔽通道的产生。

b)考虑优先级。为了不丢失实时性能，需要考虑冲突集合中事务的优先级。为了达此目的，检查冲突集合中的所有事务优先级是否都比 VT 高。

上面2种情况，可以通过2种方法解决。a)如果终止冲突集合中事务而得出的 CCF<终止有效事务而得出的 CCF 或者在冲突集合中不存在高优先级事务，那么提交 VT 并终止冲突集合 S。否则终止 VT。b)如果终止冲突集合中事务而得出的 CCF<终止有效事务而得出的 CCF，那么提交 VT 并终止冲突集合 CS。否则终止 VT。然而，由于希望最小化 CCF 且识别优先级，因而如果上面的其中一个有利于 VT，就允许提交 VT。

4 非干预性原则

为了消除隐蔽通道的间接信息流而造成的安全违背，系统设计者需要设计一个系统使其满足针对系统安全而提出的非干预性原则，其属性表现为以下3个方面：

值安全：主体读取的客体值不会被高级别主体的行为所影响，即为了保证值安全需要输入调度集合的调度次序与调度后的输出次序相同。

延迟安全：一个行为的延迟不会被一个高级别主体的行为所影响(延迟定义为行为到达系统的时间到此行为完成的时间间隔段)。

恢复安全：由于冲突行为，实时数据库中的事务可能会卷入死锁状态。从此状态恢复采取的措施是：终止导致死锁的一个或多个行为。这种从死锁中恢复的行为确保了死锁发生在相同的低级别主体，而不会因其他高级别事务的出现受到影响。满足下面条件的调度就是恢复安全调度：

1)高级别事务的出现不会防碍低级别行为中死锁的产生。

2)如果在高级别行为出现时，低级别行为之间无死锁产生，那么高级别行为出现时也不会产生死锁。

然而，非干预性原则会造成高安全级别事务截止时间丢失率的增加。由于数据/资源冲突，出于安全目的，高安全级别事务将被抢断、终止和重启，这样可能导致安全实时数据库系

(下转第82页)

14 Liu P, Ammann P, Jajodia S. Rewriting Histories: Recovering From Malicious Transactions. Distributed and Parallel Databases, 2000, 8(1): 7～40
15 Luenam P, Liu P. ODAM: An On-the-fly Damage Assessment and Repair System for Commercial Database Applications. In: Proc. 15th IFIP WG 11.3 Working Conf. on Database and Application Security
16 Pilania D, Chiueh T. Design, Implementation, and Evaluation of an Intrusion Resilient Database System: [Technical Report TR-124]. Experimental Computer Systems Lab, Stae University of New York at Stony Brook. Dec. 2002
17 Patnaik S, Panda B. Dependency Based Logging for Database Survivability from Hostile Transactions. In: Proc. of the 12th Intl. Conf. Computer Applications in Industry and Engineering, Atlanta, GA, Nov. 1999
18 Panda B, Giordano J. Reconstructing the Database after Electronic Attacks. In: Database Security XII: Status and Prospect, S. Jajodia (editor), Kluwer Academic Publishers, 1999. 143～156
19 Panda B, Giordano J. An Overview of Post Information Warfare Data Recovery. In: Proc. of the 1998 ACM Symposium on Applied Computing, Atlanta, GA, Feb. 1998
20 Sani T, Brajendra P. Post-Intrusion Recovery Using Data Dependency Approach. In: Proc. of the 2001 IEEE Workshop on Information Assurance and Security, United States Military Academy, West Point, NY, June 2001
21 Lala C, Panda B. On Achieving Fast Damage Appraisal in case of Cyber Attacks. In: Proc. of the 2001 IEEE Workshop on Information Assurance and Security, United States Military Academy, West Point, NY, June 2000
22 Bertino E, Jajodia S, Mancini L V, et al. Advanced Transaction Processing in Multilevel Secure File Stores. IEEE Transactions on Knowledge and Data Engineering, 1998, 10(1): 120～135
23 Mohan C, Pirahesh H, Lorie R. Efficient and flexible methods for transient versioning of records to avoid locking by read-only transaction. In: Proc. of ACM SIGMOD Intl. Conf. on Management of Data, San Diego, CA, June 1992. 124～133
24 Ammann P, Jajodia S, Mavuluri P. On-the-fly reading of entire databases. IEEE Trans. on Knowledge and Data Engineering, 1995, 7(5): 834～838
25 Weikum G, Schek H J. Concepts and applications of multilevel transactions and open nested transactions. In: Almed K. Elmagarmid, ed, Database Transaction Models for Advanced Applications, chapter 13. Morgan Kaufmann Publishers, Inc., 1992
26 Lomet D B. MLR: A Recovery Method for Multi-level Systems. SIGMOD Conference, San Diego, California, 1992. 185～194
27 Korth H F, Levy E, Silberschatz A. A formal approach to recovery by compensating transactions. In: Proc. of the Intl. Conf. on Very Large Data Bases, Brisbane, Australia, Morgan Kaufmann, 1990. 95～106

(上接第77页)

统失败，因为事务的实时性在实时数据库中也是非常重要的。

5 安全实时数据库系统的实时性

5.1 实时性与安全性保障(STAR-Security and Timeliness Assurance in Real-Time databases)

为了对事务截止时间丢失率提供一定的保证，可以采用STAR方法，其完全支持Bell-LaPadula模式对客体进行读/写操作的规则以及阻止不同安全级别间的直接/间接信息流的非干预性原则，并对事务允许的瞬间/平均丢失率提供一定的保证。此方法把事务分为代表低、高安全级别的Class 0和Class 1。对CPU调度和并发控制，使用Class 0事务阻止由于争夺资源和数据所产生的隐蔽通道，对Class 0、Classs 1应用QoS管理、允许控制和反馈控制方案支持设定的平均/瞬间截止时间丢失率，QoS管理和允许控制方案利用调整CPU以支持特定的平均/瞬间截止时间丢失率。此方法既阻止了不同安全级别间信息流的直接/间访问，又为实时数据库提供实时性保障。具体细节请参阅文[4]。

5.2 部分安全

为达到期望的实时性及安全性，可以在需要时允许冲突违背而采用部分安全策略。因此，对于严格数据库应用，给部分安全一个确切的定义是非常重要的。较好的方法是采用Bell-LaPadula安全模式来定义在两个确定安全级别间安全需求允许的部分安全，以及为了允许部分安全而要模糊安全级别之间的界限。随着系统性能下降，需要模糊更多界限而允许更多安全违背以减少安全冲突的数量而使系统性能得到提高。

结束语 本文陈述了基于Bell-LaPadula模式的强制访问控制机制以及阻止不同安全级别间直接/间接的非法信息流的非干预性原则，讨论了安全实时数据库的隐蔽通道属性，并在此基础上探讨了安全2PLHP及安全OPT算法以及为了提高实时性能而允许安全违背的融合安全和实时需求的部分安全策略，并对一种平衡实时数据库中的实时性和安全性需求的STAR方法进行探讨。需要指出的是，通过上述方法，尽管可以100%地避免隐蔽通道，但是仍然有少量的事务错过其截止时间。并且，本文只对基于Bell-LaPadula模式进行探讨，也没有结合数据的时态性对安全问题进行考虑，STAR方法中也只设定2个安全级别。因此，需要把数据时态一致性与安全结合起来对多个安全级别事务和基于其他模式的实时数据库安全问题做进一步的研究。

参 考 文 献

1 Son Sang H. Supporting Timeliness and Security in Real-Time Database Systems. In: Proc. of the 9th Euromicro Workshop on Real Time System(EUROMICRO-RTS'97)1068-3070/97
2 Lampson B W. A Note on the Confinement Problem. Communications of the ACM, 1973, 16(10): 613～615
3 Son S H, Chaney C, Thomlinson N P. Partial Security Policies to Support Timeliness in Secure Real-Time Databases. Dept. of Computer Science, University of Virginia, Charlottesville, VA 22903
4 Kang K-D, Son S H, Stankovic J A. STAR: Secure Real-Time Transaction Processing with Timeliness Guarantees. In: Proc. of the 23rd IEEE Real-Time Systems Symposium1052～8725/02, 2002
5 Son S H, Mukkamala R, David R. Integrating Security and Real-Time Requirements Using Covert Channel Capacity. IEEE Transaction on Knowledge and Data ngineering, 2000, 12(6): 11～12
6 Keefe T F, Tsal W T. Multiversion Concurrency Control for Multilevel Secure Database Systems. CH2884-5/90/0000/0369, Computer Science Dept. Univ. of Minneapolis MN 55455
7 Ahmed Q N, Vrbsky S V. Maintaining Security in Firm Real-Time Database Systems. Department of Computer Science the University of Alabama, Tuscaloosa, AL, 35487-0290, U. S. A
8 Ahmed Q N, Vrbsky S V. Maintaining Security and Timeliness in Real-Time Database Systems. Lucent Technologies 2139 Highway 35 Holmdel, NJ 07733, qahmed@lucent.com
9 Park C, Park S, Son S H. Multiversion Locking Protocol with Freezing for Secure Real-Time Database System. IEEE Transaction on Knowledge and Data Engineering, 2002, 14(5): 9～10
10 Amirijoo M, Hansson J, Son S H. Specification and Management of QoS in Imprecise Real-Time Databases. In: Proc. of the 7th Intl. Database Engineering Applications Sysposium 1098-8068/03