信息战条件下的数据库入侵限制和恢复技术*)

钟 勇 秦小麟
(佛山科学技术学院信息与教育技术中心 佛山528000)
(南京航空航天大学信息科学与技术学院 南京210016)

摘 要 如何在信息战条件下保持数据库的可生存性,传统的以预防为中心的数据库安全机制存在不足,预防技术、欺骗和隐藏技术、检测技术、限制和恢复技术是数据库在信息战条件下的重要生存技术。在此基础上总结了数据库入侵限制和恢复技术的发展现状,并对今后的研究重点进行了展望。

关键词 信息战,入侵限制和恢复,入侵容忍,数据库安全

Database Intrusion Containment and Recovery Techniques under Information Warfare

ZHONG Yong QIN Xiao-Lin
(Information & Educational Technology Center, Foshan University, Foshan 528000)
(School of Information Science and Engineering, Nanhang University, Nanjing 210016)

Abstract How can the databases survive information warfare attacks? The traditonal prevention-centric security is not enough. The techniques of prevention, deception and hiding, detection, containment and recovery are the important techniques for databases to survive information warfare attacks. Based on the situation the paper summaries the status of database containment and recovery techniques and the future research direction in this field is discussed.

Keywords Information warfare, Intrusion containment and recovery, Intrusion tolerance, Database security

1 信息战条件下传统数据库安全机制的不足

随着信息技术的发展,信息系统在社会生活中起着越来越重要的作用,对信息系统的攻击和破坏也成为日益严重的问题,近年来信息战(Information Warfare,IW)得到广泛的关注。Ivan Goldberg 将信息战定义为:"攻击或防卫性的使用信息或信息系统以便否认、利用、破坏或消灭敌对方的信息、基于信息的进程、信息系统和计算机网络从而保护自己的这些设施,这些行动是用来取得超出敌对方的军事或商业优势"[1]。在信息战的条件下,数据库的安全面临着前所未有的严峻形势,这主要体现在这几个方面。

首先,在信息战中,数据库往往成为受到攻击的目标。网络系统、操作系统和数据库管理系统(DBMS)是信息系统的主要支撑平台,这三者的安全性直接影响到整个信息系统的安全。在这三者之间,数据库往往成为最吸引攻击者的目标。这是因为作为信息系统关键部件的数据库在今天有着越来越重要的地位,例如,今天的数据库产品已经成为具有数十亿美元产值的工业,据统计,在1995年硬件服务器销售量的32%是由数据库产品的销售所带动的,而到达2000年这一比例达到39%[2]。数据库中往往保存着对组织或公司极为重要的数据,其重要性和价值对攻击者有很大的吸引力。同时,数据库系统本身的弱点也使其成为易受攻击的目标,数据库的数据经常需要更新以及其它众多的操作活动,再加上为了适应更新的需要许多数据库提供的优化接口,都可能受到攻击者的利用。另一方面,网络化也使数据库受到攻击的可能性、空间和时间都大大增加。

其次,传统的数据库安全机制重点在于预防,着眼于对外部用户的身份和权限约束的检查,来保证用户操作的合法性。然而,以身份认证和存取控制为主的数据库安全机制存在一定的限制。经验显示我们不可能百分之百地预防所有的安全问题,黑客们常常使人吃惊,因为他们总能发现新的方法闯入或干涉我们的系统[3]。对于合法用户特别是系统管理员的权限滥用,以预防为主的安全机制常常显示无能为力。而据统计计算机安全的主要威胁来自内部滥用而不是入侵,这些内部滥用者往往具有合法的身份认证和授权[4]。对于来自网络的攻击,攻击者往往能窃取到合法的身份或权限,如利用密码嗅探(password sniffing 或会话骑劫(session hijacking),攻击者可能获得合法的用户账号和密码或身份。

最后,数据库在受到攻击后的生存和恢复能力是传统的数据库安全机制较少涉及到的,数据的保密性,完整性和可用性是数据库安全研究领域主要关心的问题,而身份认证和存取控制主要着眼于数据库的保密性以及某些商业完整性,对数据本身的完整性和可利用较少涉及。而信息战的防卫要求采取任何手段防止攻击,但是在现有的条件下必须承认防止信息攻击的手段是不充分的,在某种程度上不可避免,总有攻击能够取得成功,因此对攻击的识别和受到攻击以后恢复手段是必要的[3]。

2 信息战条件下的数据库可生存性技术

以预防为主的数据库安全机制在信息战的条件下是不够的,一些关键部门如交通、银行等的信息系统在社会中占有非常重要的地位,它们可能需要提供一周七天、一天二十四小时

*)航空科学基金(编号:02F52033)、江苏省高技术项目(编号:BG2004-005)资助。钟 勇 讲师,博士研究生,主要研究方向为数据库安全、网络安全。秦小麟 教授,博士生导师,主要研究方向安全数据库、空间数据库、时空数据库、GIS等。

的不间断服务，这些系统需要有较强的生存能力，在系统出错或受到攻击的情况下提供服务的能力。对这些关键系统来说，需要以错误容忍为中心的可生存机制来保障安全。信息系统的可生存性代表系统在受到攻击、系统故障、意外事故的情况下能及时完成任务的能力[5]。可生存性要求系统在发生诸如硬件失效、软件错误、操作失误或恶意攻击时仍旧能提供部分基本服务或替代服务。信息系统的可生存能力是一种以错误容忍为中心的保护机制，是系统对错误的可适应性。作为信息系统的重要组成部分，数据库的可生存能力也正在成为研究的热点之一。提高数据库的可生存性重点之一是提高数据库的入侵(主动错误)容忍能力，数据库的入侵容忍指的是数据库在受到攻击的情况下继续提供基本服务的能力[2]，现存的数据库安全机制在入侵容忍能力的作用是很有限的，如身份认证和存取控制机制无法完全防止所有的攻击；实体和域约束可以保证数据的存在和合法性，但不能保证特定数据的合理性和精确性；参考完整性，攻击者可以同时修改参考和被参考数据，如果使用级联删除等规则，这些规则甚至有可能帮助攻击者传播恶意事务；事务机制也无法区分恶意事务和正常事务。

信息战假设在一定的条件下总有攻击能取得成功，强调攻击成功后信息系统的生存和恢复能力，因此，信息战防卫考虑攻击和恢复的整个过程。Ammann 将信息战的过程分为九个阶段：预防、情报收集、攻击、检测、隔离、损坏评测、重配置、修复、错误处理[3]。预防阶段是传统的保护机制；情报收集是攻击者观察系统从而确定系统的弱点，并发现作为目标的系统关键功能或数据；攻击是攻击者实现攻击计划阶段；检测是防卫者检测发生或正在进行攻击的阶段；隔离是防卫者采取行动制止攻击者在系统中的存取，孤立或限制攻击者的行动防止造成进一步的破坏；损坏评测阶段防卫者确定问题的范围，检查失效的服务和损坏的数据。重配置阶段防卫者配置在恢复进行的情况下还可以继续提供的精简或替代服务；修复阶段防卫者通过修复损坏或丢失的数据、重置或重新安装失效的系统功能以便使系统恢复正常运作；错误处理阶段是尽最大的可能识别被攻击者所利用的系统弱点，然后采取有效的手段防止这些错误再被攻击者利用。保护、检测和反应、恢复是信息系统可生存战略的主要环节。预防技术、欺骗和隐藏技术、检测技术、限制和恢复技术[3]是数据库在信息战条件下的重要生存技术。

3 信息战条件下的数据库入侵限制和恢复技术

3.1 信息战条件下的数据库入侵限制和恢复技术的独特性

信息战防卫者的目标是保证数据库在受到进攻时的可生存性，因此，在数据库受到损坏后进行修复的同时数据库应该继续提供未受损的服务，如果将数据库停止下来进行修复的话将影响数据库的生存能力，同时这可能也正是攻击者要达到的目地。在信息战条件下的数据库入侵限制和恢复技术应达到这些目的，在入侵和损坏数据被查找到以后，应该阻止入侵和损坏行动的进一步传播并对损坏的数据进行恢复，而在恢复的同时，数据库应该继续使用未损坏的数据提供服务。

3.2 信息战条件下的数据库入侵限制技术

在数据库中，由于事务操作的结果能够影响其它事务，如事务 T_i 读取被事务 T_j 更新的数据 X，则 T_i 受到 T_j 的影响，由于传递关系，T_i 也可能间接受到 T_j 的影响，当 T_j 是恶意事务时，其损坏将可能传递到所有受 T_j 影响的事务所存取的数据中。入侵限制的目地是在数据库受到入侵的情况下阻止损坏传播以及入侵者进一步造成破坏。防止损坏传播主要有两种途径，一种是对损坏数据的限制，在损坏数据得到清除或恢复前阻止其它事务存取，一种是对怀疑用户的限制，怀疑用户是一些偏离正常轨道，但在不进一步观察的情况下无法确认是否为正常或恶意用户，限制技术既可以防止过早地停止这些用户对数据库的存取又可以防止这些用户对数据库的进一步破坏。入侵限制技术主要有静态分区、标记、多阶段隔离和多版本等方法。

3.2.1 *静态分区法* 静态分区的方法将数据库数据划分成不同的区间，事务活动的范围限制在单个区间，损坏传播也限制在单个区间，当单个区间受到损坏时还可以使用其它区间。分区的概念类似于在多级安全数据库中使用的范围(category)。由于在许多数据库中分区的方法并不适用，也可以在数据库中定义区间的边界，记录通过边界的触发或传递更新，限制数据通过边界的带宽和条件。静态分区的方法实现简单但易导致过多的数据受到隔离。

3.2.2 *数据标记法* 数据标记方法是通过损坏标记来区分损坏和正常的数据，再通过改进事务协议来防止正常事务读取标记为损坏的数据从而达到隔离的目地。Ammann[3]使用颜色标记的方法来确定损坏信息，将数据分成四个部分：红、浅红、浅绿和绿。红色是损坏的不安全使用的数据；浅红色是损坏的数据，但如果禁止存取这些数据的代价高于允许存取这些数据；浅绿是近似数据，常常是由于在线修复造成的；绿色是完全正确的数据。使用这些标记，Ammann 重新定义了数据库一致性的概念和保持这种一致性的事务语法协议。Ammann 将数据库中的事务分为正常事务、攻击事务、防卫(countermeasure)事务。防卫事务包括检测事务和修复事务，入侵检测和数据修复都是由防卫事务来完成。隔离策略是通过颜色标记和改进的事务处理协议，红色标记的数据必须完全隔离，事务不能存取，浅红色的数据不完全隔离但其传播必须要有详细的审计。Ammann 的方法假设数据具有初始标记，但并没有说明如何标记这些数据以及由于如何处理不精确的标记等问题。

3.2.3 *多阶段隔离法* 在数据库入侵检测过程中，为了检测的精确度需要有一个检测潜伏期(detection latency)，同时对事务检测过程的执行也需要一定的时间，在这个阶段，一个繁忙的数据库可能已快速将损坏传播开来而导致原先的隔离范围无效。例如在查找损坏数据 x 的潜伏期内，许多事务可能已经读过 x 并使用它修改其它数据，当确定隔离 x 时，损坏已经传播从而导致隔离无效。多阶段的隔离策略首先以尽快的速度将所有与受损数据有关的数据隔离，确保损坏不传播，然后再在后面的阶段将错误隔离的数据释放。多阶段的隔离策略保证了潜伏期内损坏数据不能传播但对数据的可用性略有影响。Peng Liu[6]提出了一个多阶段的隔离策略，将每个数据对象附上时间戳，当恶意事务 B 被识别时，首先将 B 的写集数据和时间戳晚于 B 提交时间的所有数据对象隔离，再以后的阶段再逐步将错误隔离的正常数据释放。由于需要使用时间戳以及对事务语句的改写，多阶段隔离法对数据库的性能稍有影响。

3.2.4 *多版本法* 多版本法来源于对怀疑用户的隔离，怀疑用户的行为略微偏离正常轨道，在不进一步观察的情况下无法确认是否为正常或恶意用户，如立即停止该用户的存

取，如果后来证明该用户只是有不通常但合法的行为，停止存取将给该用户带来不必要的损失以及减少系统的可利用性，也不利于收集证据。但如果让该用户继续存取系统直到确认该用户的恶意行为，则可能给系统造成进一步的损失。多版本的方法将正常用户版本和怀疑用户版本分开，当怀疑用户证明是正常用户，其版本与正常版本合并，否则抛弃。多版本的方法可以使怀疑事务在监视下继续执行，又不会对数据库造成进一步的伤害。

Peng Liu 等利用版本矢量的方法提出了一个在文件系统中使用的具体隔离协议[7]，当文件 F 被怀疑用户 S 修改时，S 对文件 F 正在进行及其随后的修改将被隔离，为了区分 S 对 F 的修改和可信用户对 F 的修改，F 的主版本和每个隔离版本都附有版本矢量号。Jajodia 等则将基于版本的隔离方法引入到数据库中[7]，将版本号安排到数据库中的数据项，对数据项 x 初始的版本号 $x[MAIN]$，当首位怀疑用户被识别时，增加特定版本号 $INIT$ 到数据项中如 $x[MAIN][INIT]$，当怀疑用户 S_i 识别时，再附加一独特版本号如时戳 t_i，则 x 的版本号为 $x[INIT][t_i]$，Jajodia 给出了隔离的具体协议和算法。利用版本矢量，Peng Liu 实现了一个基于 Oracle 的原型系统 DAIS，DAIS 使用触发器和事务轮廓追踪事务的数据读写，通过透明地修改和重定向用户的 SQL 语句到不同的数据版本。Peng Liu 的实现证明多版本法是很有效的，但由于需要改写和重定向 SQL 语句，对隔离用户事务的执行有一定延迟。

多版本法使系统能更细微地控制用户的活动，怀疑用户可以继续使用系统资源，减少了否认服务的发生。但当怀疑用户被证实是合法的，将其数据合并到主数据库是一个耗时的过程。而在很多场合，数据库中只有一部分数据由于其重要性和完整性需要防止被损坏，在这些场合，完全孤立怀疑用户消耗更多的系统资源且无必要，Fayad 等[10]将数据按照重要性和完整性分成三类：无限制和非关键数据项、限制和非关键数据项、关键数据项，并定义了在不同数据项之间所允许的信息流。Fayad 提出了通过检测数据不一致性实现的应用级孤立算法，Fayad 方法的缺点是解决数据的不一致性需要人工干涉。

多版本法由于需要多个数据版本，占用系统资源较多，且系统开销较大，同时在版本合并时可能会导致正常事务的退出或重做。

3.3 信息战条件下的数据库恢复技术

在 TCSEC 将可信恢复(trusted recovery)描述为在自动数据处理系统(ADP)故障或其它引起系统运行的不持续后能提供过程或机制确保系统恢复到保护未受损的状态。在 TCSEC 的可信恢复的解释中文[11]将可信恢复的目地解释为在系统发生故障或其它不持续操作时确保系统安全和审计性质的维护。Jajodia[12]将恢复分为当损坏恢复时系统停止运行的冷启动(ColdStart)、系统继续提供部分关键服务的暖启动(WarmStart)和恢复对用户透明的热启动(HostStart)，并提出了在信息战条件下的损坏恢复模型和可信恢复的准则。在数据库领域，传统的数据库恢复机制是为了保持事务的 ACID 性质，并不是用来处理恶意事务的。传统的恢复机制决不会撤消已提交的事务，而在信息战中，在被检测到之前，恶意事务可能已完成且已提交，可能也存在良性事务已读取被恶意事务所毁坏的数据并将受到这些数据污染的数据写入数据库中，因此在信息战条件下的数据库恢复技术分为两个阶段，第一阶段是确定应该撤消的事务，第二阶段是撤消已提交的事务。

3.3.1 *确定撤消事务的方法* 在恶意事务被发现之前，恶意事务可能已经提交并影响到其它事务，必须将所有这些事务确定下来。如何确定受影响的事务，可以通过事务之间通过对数据读-写形成的依赖关系或数据本身存在的依赖关系来确定。

(1)基于事务读-写依赖的确定方法。事务之间通过数据读取存在一定的依赖关系，事务 T_j 在某历史中依赖 T_i 如果存在数据项 x 满足：1) T_i 更新 x 后 T_j 读取 x；2) 在 T_j 读取 x 前 T_i 没有被撤消；3)所有在 T_i 更新 x 和 T_j 读取 x 之间的更新过 x 的事务都被撤消。在一个历史中，如果事务 T_1 和 T_2 的有序对 (T_1, T_2) 出现在依赖关系的传递闭包中则称事务 T_1 影响 T_2。如果恶意事务 B_1 影响良性事务 G_1 则称 G_1 是怀疑的。如果良性事务不受恶意事务影响则不需要撤消和重新执行。Ammann 等[13]提出多个基于读-写依赖适用于不同条件的恢复算法，这些算法特别适用于不支持语义模型的主流商业系统，但这些算法撤消受怀疑的良性事务，而这些良性事务中有可能存在不需要撤消的情形，为了保存更多的良性事务不被撤消，Peng Liu 等[14]阐述了辨别读-写依赖和写-读依赖的重要性，提出了一个重写执行历史的算法，该算法通过将恶意事务尽可能地重写到执行历史的末尾，重写历史的前缀完全由良性事务组成，并证明该前缀相当于使用写-读依赖图撤消恶意事务和受恶意事务影响的良性事务，重写历史的算法比使用依赖图能保存更多的良性事务。

在实现方法上，原型系统 ODAM[15]使用[14]的方法建立在商业 DBMS 之上，ODAM 按照 SQL 基础的事务记录数据库更新日志，在恢复期，ODAM 通过分析 SQL 日志来找出事务的依赖关系，由于 ODAM 同时需要写日志和读日志来分析事务的依赖关系而读日志是普通商业 DBMS 所不提供的，ODAM 使用事务的预定义模板的方法来获得读信息，事务运行时分析事务的 SQL 语句将这些读模板实例化。原型系统 Fastrek[16]则通过在内部保持一张运行时的事务依赖图而不需要读日志，通过事务依赖图，Fastrek 能准确和自动确定潜在的能被恶意事务所损坏的事务。对每一数据项，Fasttrek 使用额外的字段记录最后更新该数据项的事务，对数据项 x 和 x 的最后更新字段 Prev(x)，当事务 T 读取 x，Fastrek 在 T 和 Prev(x)之间建立依赖关系。Fastrek 原型实现于数据库 PostgreSQL 之上，与 ODAM 相比 Fastrek 的方法不需要关于事务的预先知识且更为精确。

基于事务读-写依赖的方法需要将一些并未受恶意事务影响的事务退出或重做，且必须退出或重做事务的所有操作，对效率和恢复的速度有一定的影响，同时由于必须读取恶意事务及其影响事务的相关日志，而在某些情况下，入侵可能经过数天甚或数月才被发现，相关的日志必须保存，这种需要使日志变得庞大和搜索的时间变长，无法满足实时应用的需要。为了减少日志存取时间，Patnaik 和 Panda[17]提出使用事务依赖的方法将日志分成簇集(clusters)，将恢复操作限制在单个簇中，减少页面 I/O。

(2) 基于数据依赖的确定方法。这种方法需要撤消或重做受影响事务的所有操作，基于数据依赖的方法利用数据之间的依赖来关系进行恢复，只需要撤消或重做事务的受影响操作。对事务 T_i，写操作 $w_i[x]$ 如果是用从读操作 $r_i[x]$ 中的值计算出来的，则称写操作 $w_i[x]$ 依赖读操作 $r_i[x]$。对数据

值 v_1、v_2，如果写 v_1 的写操作依赖于 v_2 上的读操作则称 v_1 依赖于 v_2，v_1 和 v_2 可以是同一数据项的不同版本。在一个事务内部，一些操作可能独立于其它操作，因而，对一个受恶意事务影响的事务来说，不是所有操作都受到影响，在恢复阶段，不需要重新执行事务的所有操作而只需要撤消和重执行受影响的操作。Panda[18,19]首先提出考虑被各种事务存取的数据项之间的依赖性来识别受影响的数据项并将这些数据项恢复到一致性数值。

数据依赖方法也存在读取日志时间过长的问题，Tripathy 和 Panda[20]提出扩展的数据依赖模型，将日志分成多个簇集，相互依赖的操作才保存在同一簇中，损坏评定阶段，只扫描相关簇。性能分析显示，对入侵恢复来说，簇集方法的性能超过传统的日志方法，簇集方法的缺陷在于不能防止事务形成大的簇集从而减少其效率。为了判定受影响的事务，现存的大多数损坏评定和恢复算法都需要读取数据库日志的很大一部分，导致页面 I/O 的增加，影响恢复的速度。Lala 和 Panda[21]利用辅助数据结构在事务提交时捕获事务的依赖关系，相对于日志来说这些结构特别紧凑，为了加快这些结构的查找，使用事务提交序列来代替事务 ID，在损坏评定阶段通过查找这些结构而不是一大部分的日志有效地减少了页面 I/O。Lala 和 Panda 提供了四个执行快速的损坏评定算法，当检测到入侵时，这些算法使用上述结构迅速识别受影响的事务及其在日志中的起始地址而不用查询日志，因而加速了恢复过程。

3.3.2 *撤消已提交事务的方法* 撤消已提交的事务是信息战中数据库恢复区别于传统恢复技术的重要内容，撤消已提交事务的方法有两种，回滚和补偿。

(1)回滚。是一种后向恢复的方法(backward recovery)，将数据库所有的活动回滚到不存在损坏的检测点。回滚机制是数据库事务恢复机制使用较多的方法，事务机制一般使用 undo/redo 日志回滚整个事务，也存在回滚部分事务的方法[22]。回滚方法较为有效，但由于经常需要保持备份且在前一检测点到回滚阶段的操作都要撤消导致系统开销较大。在信息战的条件下，需要数据备份实时进行，在传统恢复机制研究中，有多种实时、增量读取整个数据库的方法如 Mohan 等提出的基于临时版本的方法[23]，Ammann 等的颜色标记法[24]可以应用。

(2)补偿。该方法通过执行补偿事务来撤消提交的事务或操作步骤，补偿法并不一定将数据状态恢复到恶意事务或步骤未发生前的状态。补偿方法可以是面向动作的(Action-oriented)或面向效果的(Effect-oriented)，面向动作的补偿方法只补偿事务或步骤 T 的操作，面向效果的补偿方法不仅补偿事务或步骤 T 的操作同时也补偿被 T 的操作所影响的其它操作。

在多级事务和嵌套事务中常使用补偿方来代替传统的 undo，Weikum[25]在多级事务模型中放松传统事务的 ACID 约束，利用操作的语义性质来放松并发事务的孤立性，同时使用补偿方法代替建立在事务状态基础上的 undo 来保持原子性。David[26]的多级恢复模型 MLR 为了取得更高的并发度，通过高级的补偿操作来撤消低级子事务，从而允许在子事务提交时用更少限制的高级锁来代替限制性的低级锁。为了方便捕获应用语义，Korth[27]提出了一个可有效应用于恢复的补偿事务的正式模型，在该模型中可以定义多个正确的补偿类型，一个补偿事务的类型既可以是从传统的 undo 类型到应用特定的特殊类型，补偿事务必须遵守一些限制。不同的补偿类型通过补偿的完全性来识别，历史 X 由被补偿事务 T、补偿事务 CT 和依赖 T 的事务集 $dep(T)$ 组成，X 是完全的如果 X 相等于 $dep(T)$ 中的事务的某些历史。补偿方法往往要求数据库能够辨识应用的语义信息，而主流的商业数据库系统往往不支持语义模型，其应用受到一定的限制。

结束语 当前，信息已成为社会发展的重要战略资源，国际上围绕信息的获取、使用和控制的斗争愈演愈烈，信息安全成为保障经济健康发展、维护国家安全和社会稳定的一个焦点。在信息战条件下的信息系统的可生存能力正在成为信息安全研究的热点之一。在数据库领域，提高数据库的可生存性重点在于提高数据库的入侵(主动错误)容忍能力，入侵容忍数据库扩展了传统数据库的安全能力，使之能够在受到攻击情况下有更强的生存和服务能力，成为未来数据库安全研究的重点。预防技术、欺骗和隐藏技术、检测技术、限制和恢复技术将是数据库在信息战条件下的重要生存技术。

信息战的严峻形势、数据库结构的复杂性，使数据库的可生存性技术面临着许多的研究难点，迄今为止，数据库可生存性技术还处在研究阶段，完善的实用系统尚未见到。现今对数据库可生存性技术的研究远远不够，需要有更多更深入的研究。

参 考 文 献

1 Ivan G. Institute for the Advanced Study of Information Warfare (IASIW). The IASIW Project, January 1996. Web page on-line. Available at: http://www.psycom.net/iwar.1.html Accessed February 4, 2000

2 Liu P. Architectures for Intrusion Tolerant Database Systems. In: Proc. of 18th Annual Computer Security Applications Conf. Dec. 2002. Las Vegas, Nevada

3 Ammann P, Jajodia S, McCollum C D, et al. Surviving information warfare attacks on databases. In: Proc. of the IEEE Symposium on Security and Privacy, Oakland, CA, May 1997. 164～174

4 Carter, Katz. Computer crime: an emerging challenge for law enforcement. FBI Law Enforcement Bulletin, Dec. 1996

5 Ellison Rl, et al. Survivability: Protecting your critical systems. IEEE Internet Computing 3, 6Nov.-Dec. 1999. 55～63

6 Liu P, Jajodia S. Multi-phase damage confinement in database systems for intrusion tolerance. In: Proc. 14th IEEE Computer Security Foundations Workshop, June 2001. 191～205

7 Liu P, Jajodia S, McCollum C D. Intrusion Confinement by Isolation in Information Systems. Journal of Computer Security, 2000, 8(4): 243～279

8 Jajodia S, Liu P, McCollum C D. Application-Level Isolation to Cope With Malicious Database Users, ACSAC'98. In: Proc. 14th Annual Computer Security Applications Conf., Phoenix, AZ, December 1998. 73～82

9 Liu P. DAIS: A Real-Time Data Attack Isolation System for Commercial Database Applications, 17th Annual Computer Security Applications Conference, Dec. New Orleans, 2001

10 Fayad A, Jajodia S, McCollum C D. Application-level isolation using data inconsistency detection. In: Proc. 15th Annual Computer Security Applications Conf, Phoenix, AZ, December 1999. 119～126

11 National Computer Security Center. A Guide to Understanding Trusted Recovery in Trusted Systems. NCSC-TG-022, 30 Dec. 1991

12 Jajodia S, McCollum C D, Ammann P. Trusted Recovery. Communications of the ACM, 1999, 42(7): 71～75

13 Ammann P, Jajodia S, Liu P. Recovery from Malicious Transactions. IEEE Trans. on Knowledge and Data Engineering, 2002, 15(2)

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