基于安全策略的一种数据保护方法及实现技术*)

张 孝 王 珊 彭朝晖
(中国人民大学信息学院 北京 100872)

摘 要 防止非法访问一直是数据库安全的重要内容,但随着防火墙、权限检查等技术的成熟,应用中越来越多的安全威胁开始来自数据管理者。本文给出了在数据库上采用安全策略的一种数据保护方法和基本原则,它能够防范系统外的入侵者,也能够防止数据库管理者的数据泄漏。同时我们还根据安全策略的定义,给出相应的查询优化指导方法。简单分析表明,执行引擎根据数据库操作的代价特点优化或重写查询能显著减小安全检查带来的性能影响。

关键词 安全策略,安全标签,查询优化

One Method and Implementation of Security-Policy-based Data Protection

ZHANG Xiao WANG Shan PENG Zhao-Hui
(School of Information, Renmin University of China, Beijing 100872)

Abstract Preventing unauthorized access is always one of the main issues to database security, however, internal administrator becomes one threat more and more while facilities like firewall are maturing and widely laid out. A security-policy-based method and its principles are introduced in this paper, which can either defense the external intruders or the leakage of sensitive data because of the administrator(s). We also present several guidelines for cost-based query optimization according to the filter functions of the security policy. The basic analyses show it is feasible to effectively reduce the overhead of security checking by query rewriting and optimization in a query engine.

Keywords Security policy, Security label, Query optimization

1 引言

数据库安全一般是指数据系统通过各种技术和管理安全措施保护数据库中的数据免遭破坏、更改或泄露。各种数据库系统在保障安全时,主要考虑如何防止非法侵入和越权访问来达到标准[1~3]规定的保护级别。随着数据库权限检查技术、各种防火墙的成功运用,复杂的外部攻击都能受到有效遏制,但是电子商务等应用领域的快速发展,数据库安全的重要威胁正由外部转向数据库管理者,管理员的疏忽或故意泄漏正造成越来越多的安全事故,给客户和公司造成损害。

例 1 信用卡号被窃案。某个网页上曾出现一个广告:只要点击页面按钮,就能够获得一个免费信用卡号。有人真的使用得到的卡号成功完成了网上电子交易或消费,但信用卡的真正持有者报了警,最后美国 FBI 介入了该案件。

例子中信用卡号被窃就是由于系统实现者和维护人员的疏忽,没有启用数据库最基本的口令验证保护而使用了系统缺省用户和密码,导致事实上的不设防,而给人以可乘之机。

例 2 冒领信用卡案。这是一个更严重的例子。美国某个公司的数据库管理员在进行系统维护过程中,私自备份了公司客户的关键个人信息,包括客户的姓名、生日、社会安全号 SSN(Social Security Number)和驾照号。在离开该公司后,利用客户信息以他人的名义从网上申领信用卡后恶意透支。

上述案例中的信息是保证电子商务有序、有效、合法完成的关键数据,作为敏感信息,必须得到安全保护。特别是例 2 的安全问题是系统维护人员利用高级权限访问客户的敏感信息,而相应的数据库系统却没有提供限制 DBA 的有效手段,即使在支持 B1 级安全的数据库系统[4,5]中,由于特权或共同作案的可能,仍然可能被泄漏。

本文贡献之一是给出了在数据库(C 级或 B 级安全)内支持安全策略的一种模型、原则和实现方法,开发人员可以定义面向应用的安全策略,在防范系统外入侵(如例 1)的同时也能够防止数据库维护者的数据泄漏或非法访问(如例 2)。

另外,我们将安全策略的元数据作为查询优化的启发信息,给出基于代价的查询优化指导方法,特别是执行引擎能够根据过滤函数中数据操作的代价来优化查询。

本文第 2 节介绍支持安全策略的安全模型,给出主要原则和实现方法;第 3 节是查询处理的优化方法和代价分析;第 4 节是相关工作,最后是全文总结。

2 安全策略模型及实现方法

安全策略是任何人访问计算机资源都必须遵守的一组规则和过程用以确保数据和资源的机密性、完整性和可用性。在安全策略基础上,数据库安全策略是指数据库中定义的、用来描述对数据对象进行访问时的控制规则。它与常规权限无关,任何访问(即使通过权限检查和安全级别验证)如果不符合它的规则,仍无法获得相关的数据。应用系统开发者通过制定面向应用的安全策略,保障数据安全。

本节首先描述具有安全策略的一种数据库安全架构,然后给出维护策略定义的元数据模式和实现举例。

*)国家自然科学基金资助项目 604473069,60496325。张 孝 副教授,主要研究方向为数据库体系结构,数据网络;王 珊 教授,博导,主要研究方向为数据库与知识库等;彭朝晖 博士研究生,主要研究方向为数据库关键词检索。

2.1 安全架构和非形式化原则

支持安全策略的数据库系统安全架构如图1所示。虚线是可选的功能模块，其中MAC是B1级安全以上的数据库系统必需的，而安全审计是C2级或以上安全需要的。

2.1.1 查询处理过程

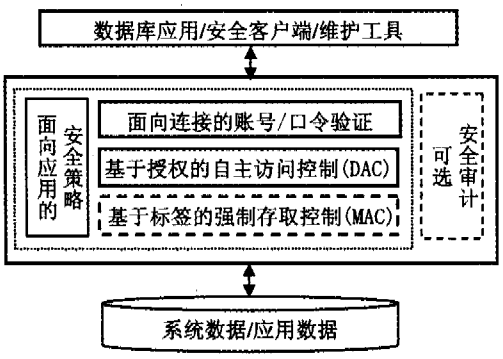

图1 安全架构

图1的安全架构中，我们给出的安全策略可以在各种级别的数据库系统内实施。数据库检索处理如下：

a)连接请求验证，建立内部运行UID；
b)应用关于连接的安全策略；
c)多次交互提交数据库操作请求：
1)对操作进行DAC检查；
2)[可选(如果支持MAC则有该项检查)]对每个元组进行MAC检查；
3)对符合请求的元组应用安全策略，滤除违反策略规则的；
d)[可选(如果支持审计且设置必要的审计选项，则执行)]按配置选项记录审计结果；

处理中，b)检查连接请求的合法性。c)-3)对查询结果集根据内容，屏蔽违反安全策略的数据。对DDL/DML语句，安全策略在请求影响原有数据或生成新数据前被应用。

2.1.2 安全策略基本原则

通过分析应用领域的典型案例，我们设计如下基本原则：

1)用户普适性或不可规避性。任何安全策略不会因某个特定用户而失效，即使DBA也没有特权。

2)空口令禁用。禁止用户不指定登录口令，拒绝无口令连接。该原则能很简单地预防例1发生的问题。

3)精确匹配。修改安全策略信息必须由策略创建者完成，修改策略时必须准确提供原始用户名、口令。

4)定义不可逆。访问控制策略的定义要以加密方式存在。同时要求加密定义不能以明文导出。

5)多粒度对象控制。安全策略可以应用到不同粒度的对象，如数据库、模式、表/视图、元组、列等。

6)数据加密。数据加密是防止DBA直接检索应用敏感数据的有效手段。简单加密都能在一定程度上制止案例2的发生。

上述规则1~4建议是强制性的，安全策略管理器必须提供；规则5是可配置的；规则6是推荐性规则。

规则1是安全策略模型的基础，如果安全策略不具有普适性，特权用户就可能对应用敏感数据造成威胁。空口令或缺省口令都是产生口令漏洞的基本形式，规则2的严格形式是禁止缺省用户访问应用数据。规则3限制超级用户通过修改安全策略来访问应用敏感信息。规则4特别针对具有备份权限的用户，避免他们从导出数据中发现原始敏感数据。规则5要求策略可以应用到不同粒度的对象，安全策略管理器可以选择不同的粒度。规则6仅作为推荐性原则，因为有些系统支持数据加密，另外加密也可以通过应用完成。

策略定义也以关系表的形式存在。

2.2 策略元数据模式

安全策略用于对象访问控制，策略定义主要包括适用的数据对象和控制过程。我们给出定义的一种元数据模式：

```
ALL_SECURITY_POLICIES(
  SEC_POLICY_ID, SEC_POLICY_NAME, /* Policy Caption */
  SEC_POLICY_CREATOR, /* Creation Information */
  SEC_POLICY_CREATETIME,
  OBJECT_CATALOG, OBJECT_SCHEMA, /* Applied Object */
  OBJECT_RELATION, OBJECT_COLUMN,
  ACCESS_OPER_CLASS, /* Access Operation Type */
  ACCESS_OPERATION,
  FILTER_FUNC_NAME, /* Filter Function */
  FILTER_FUNC_NARG, FILTER_FUNC_ARGS );
```

模式定义分为5节，分别记录着：

1)策略标题。保存策略的标识信息，都是候选码。

2)创建信息。创建者必须具有策略管理权限，这些信息在创建策略时自动维护，不可以修改。

3)适用对象。本模式给出可选择支持的4种粒度。对象可以是表、索引、过程或触发器等。具体对象和存取操作类型相关。

4)存取操作类型。根据适用的对象类型分为：SESS、CATA、SCHEM、REL、COL、IDX等多个组。如SESS_CONN操作检查连接空口令，拒绝无效请求。

5)过滤函数。又称响应函数或句柄函数。在约定的场合调用，实参与执行上下文相关。可以是内置函数、自定义函数等。对于自定义函数，系统必须支持定义加密，并且在数据导出时不能将逆转为明文。

本文假定目标数据库系统支持PL/SQL或T-SQL，支持自定义函数。另外为保持系统独立性，未给出属性类型。

下面举例说明如何利用安全策略防止例2的案件发生。

2.3 实现举例

这里给出安全策略如何控制DBA存取敏感数据能力。

例3 防止例2中冒领信用卡案。假定数据库系统为X-DBS，单独泄漏SSN已经特别危险，因此保护SSN。

具有保护能力的应用系统可以有如下情形：

a)应用层将加密后的数据存入数据库，数据库将加密数据作为普通数据，DBA也无法看到原始数据。

b)支持数据加密，同时不支持完全可逆导出(数据导出是很多系统支持数据迁移的重要工具，但有些未提供数据加密)，DBA只能看到加密后的数据，原始数据得到保护。

c)符合规则4和6的联合，而a)相当于没有安全策略下b)的变形，增加了应用开发的复杂度。

定义客户信息在X-DBS中的关系：

X-DBS. INSURA. CUSTOMER(

CID, CNAME, CDOB, CSSN, CLICENSE, ……);

在CSSN上定义安全策略。假定存在自定义函数CINFO_ENCRYPT。它接受1个和CSSN同类型的参数，返回加密的同类型数据。

应用创建者使用扩展SQL语法创建策略：

```
CREATE POLICY CUST_SSN_ENCR
ON X-DBS. INSURA. CUSTOMER. CSSN
WHEN COL_INSERT, COL_UPDATE
BY CINFO_ENCRYPT
```

定义策略后，插入或修改CSSN都会启用它，执行引擎用CINFO_ENCRYPT加密后的数据自动替换该列原值作为将写入数据库的真实数据。如元组

(101, 'Tim', '1970-03-12', '334-25-9023', 'DF425', …)

执行引擎内部转化后，相当于没有策略时插入的元组：

(101，'Tim'，'1970-03-12'，CINFO_ENCRYPT('334-25-9023')，'DF425'，…)

DBA既不能直接检索到'334-25-9023'，也不能通过数据导出来获得原始信息。至于破解密钥，那是加密的问题。

限于篇幅，这里不再列举各种操作的具体实现方法。

3 基于代价的查询重写/优化

支持安全策略，数据库执行引擎不仅扩展语法(另外一种不修改语法的实现是系统将安全策略作为抽象数据类型来维护，我们在KingbaseES中就是采用类似技术支持B1级安全性)，同时还要优化策略影响的执行计划，特别在结果集元组很多的时候。

例4 根据SSN确认客户的保单数的查询Q1：

```
SELECT COUNT(*) FROM CUSTOMER
  WHERE CSSN='334-25-9023';
```

因为CSSN上有数据加密，相应的有安全策略对操作REL_SELECT和COL_SELECT进行解密。假设过滤函数是CINFO_DECRYPT，则最简单的查询执行过程是：

1)抽取所有元组形成快照；

2)策略执行CINFO_DECRYPT，如果匹配则计数。

该方法相当于对加密数据没有策略时的等价查询Q2：

```
SELECT COUNT(*) FROM CUSTOMER
  WHERE CINFO_DECRYPT(CSSN)='334-25-9023';
```

假设有N个客户记录，每页M个元组，估略代价为：

$Cost_{i/o}(N/M)+N*Cost(CINFO\_DECRYPT)$

观察Q2，很容易得到一个等价查询Q3：

```
SELECT COUNT(*) FROM CUSTOMER
  WHERE CSSN=CINFO_DECRYPT('334-25-
        9023');
```

检索函数附加代价为1！这时我们有下面的优化建议：

O1、将过滤函数应用到常量但保持查询等价；

O2、常量上的应用结果要缓存；

对于例4，如果CSSN经常被检索，可以建立索引，由于该列需要加密，索引键值也需要安全策略，这时有：

O3、索引键值直接使用执行策略后的元组列值。

这时的检索代价大略是：

$Cost_{i/o}(IdxPg+RelPg)+RelPg*p*Cost(CINFO\_DECRYPT)$

$Cost_{i/o}(IdxPg+RelPg)$是读入的索引和关系的页面I/O，而p是页面中有效元组的选择率。

此外，各种通用的数据库查询重写和优化方法仍然适用。

4 相关工作

视图是最常见的安全保护机制，通过选择输出列达到屏蔽信息的目的[6]，但视图不能影响对基本表的访问。

触发器主要作用在操纵表的DML语句，如PG[7]、SQL Server 2000[8]、MySQL 5.0[9]等。它能控制元组操作但不能预防例1的问题。类似有PG的规则系统，在执行SQL前根据规则重写查询。触发器和规则系统都可以增强以支持安全策略。SQL Server 2005[10]、Oracle 9i/10g[11,12]都提供了更强的触发器技术，监测服务器、数据库级的事件，但效率需改进。9i/10g[13]支持返回动态谓词的安全策略，但不支持在安全策略中使用任意过滤函数，包括数据加密。

结论和未来工作 数据库受到来自内外的安全威胁，DBA也不可靠。安全策略面向应用，对任何数据库用户都有效。本文的安全策略模型有6个基本原则：4个强制性、1个可配置和1个推荐原则，它们或控制操作，或加工数据。

完全支持安全策略基本规则的系统不要求应用修改原来的SQL语句。按照建议模型，查询引擎能够自动根据策略定义对各种SQL语句进行基于代价的重写和优化。

我们提出的安全策略模型只是一种防卫手段，随着面向服务和自定义能力的增强，数据库系统会受到更多安全危险。我们的策略模型要继续完善，还要考虑意外补救措施和多类管理员权限划分等。

参考文献

1 CCIMB-99-031& 032 & 033. Common Criteria for Information Technology Security Evaluation, Version 2.1, August 1999

2 GB/T 18336-2001. 信息技术/安全技术/信息技术安全性评估准则. 2001

3 Department Of Defense. Trusted Database Management System Of TCSEC, 1991

4 Elliott B D, LaPadula Leonard J. Secure computer systems: unified exposition and MULTICS interpretation, 1976

5 张孝. 可信COBASE的系统强制存取控制的设计与实现:[中国人民大学硕士学位论文]. 1998

6 萨师煊，王珊. 数据库系统概论(第三版). 高教出版社，2000

7 PostgreSQL8.2. http://developer.postgresql.org/docs/postgres/

8 Microsoft Corp. SQL Server 2000, http://msdn.microsoft.com/library/default.asp?url=/library/en-us/tsqlref/ts_create2_7eeq.asp

9 MySQL 5.0. http://dev.mysql.com/doc/refman/5.0/en

10 Microsoft Corp. SQL Server 2005. http://msdn2.microsoft.com/en-us/library/ms189799.aspx

11 Oracle Corp. Oracle9i Application Developer's Guide, 2002

12 Oracle Corp. Database SQL Reference 10g, 2003

13 Oracle Corp. Label Security Administrator's Guide 10g, 2003

(上接第103页)

8 Han Yan-Bo, et al. CAFISE: An Approach to Enabling Adaptive Cofiguration of Service Grid Applications. Journal of Computer Science & Technology, 2003, 18(4):484～494

9 张尧学，方存好. 主动服务——概念、结构与实现. 北京：科学出版社，2005

10 李建强，范玉顺. 基于Petri网模型化简的工作流模型验证. 信息与控制，2001, 30(6):492～497

11 孙瑞志，史美林. 一种基于Petri网化简的工作流过程语义验证方法. 软件学报，2005, 16(7):1242～1251

12 许安国，蒋昌俊. P/T网的化简运算及其性质研究. 软件学报，1997,8(7): 493～504

13 Jiang C J. A Polynomial-time Algorithm for the Legal Firing Sequences Problem of A Type of Synchronous Composition Petri Nets. Science in China (Series E), 2001,44(3):226～233

14 Jiang C J. Testing of Functions of Complex Systems Based on Synchronous Composition Nets. Int J of Studied on Information Control, 2000, 9(4):321～328

15 Jiang C J, et al. Behaviour Relations in Synthesis Process of Petri Net Models. IEEE Trans on RA, 2000, 16(4):400～407

16 Jiang C J, et al. Urban Traffic Information Service Application Grid. J of Comp Sci & Tech, 2005, 20(1):134～140

17 Jiang C J, Wang H Q, Liao S Y. Behavior Relativity of Petri Nets. J of Comp Sci & Tech, 2002,17(6):770～780

18 Du Y Y, Jiang C J. Formal Representation and Analysis of Batch Stocks Trading Systems Through Logical Petri Net Workflows. In:Proc. of 4th Int Conf. on Formal Engineering Methods, Lecture Notes in Computer Science, No. 2495, 2002. 221～225

19 蒋昌俊. 离散并发系统的PN机理论. 北京：科学出版社，2000

20 Peterson L L. Petri网理论与系统模拟. 吴哲辉译. 中国矿业大学出版社，1981

21 Hopcroft J E, et al. 自动机理论、语言和计算. 徐美瑞译. 北京：科学出版社，1990