VPD 细粒度访问控制系统①

王振辉[1], 王振铎[2], 张 敏[1], 王艳丽[1]

[1](西安翻译学院 工程技术学院, 西安 710105)
[2](西安思源学院 电子信息工程学院, 西安 710038)

摘 要: 为解决日益严峻的 Web 数据库安全现况和适应企业信息管理系统中对访问控制灵活多变的需求, 分析了目前访问控制方案, 对虚拟专用数据库技术进行研究, 提出基于行数据的细粒度访问控制系统模型, 为数据库管理员设计了后台访问控制系统, 并使用Oracle虚拟专用数据库技术进行实施和部署, 该方案不用修改原应用系统, 同时提高了系统安全性和灵活性.

关键词: 虚拟专用数据库; 细粒度; 访问控制; 安全策略

Fine-Grained Access Control System by VPD

WANG Zhen-Hui[1], WANG Zhen-Duo[2], ZHANG Min[1], WANG Yan-Li[1]

[1](School of Technology and Engineering, Xi'an Fanyi University, Xi'an 710105, China)
[2](Electronic Information Engineering, Siyuan University, Xi'an 710038, China)

Abstract: In order to solve the increasingly serious web database security and adapt flexible demand of enterprise information management system, analysis current access control scheme and study the virtual database technology, coming up a fine-grained access control model based row data, design the database control system, implementation and deployment by used Oracle virtual private database, the scheme need not modify the original application system and improve the system security and flexibility.

Key words: virtual privacy database; Fine-Grained; access control; security policy

随着 Internet 的普及和电子商务与电子政务的蓬勃发展, Web 信息系统已经成为软件系统开发的主要部分, 这也使得企业面临更为复杂的信息安全问题. Web 数据库是基于 Web 信息系统的核心组成部分, 面临来自企业外部和内部的双重威胁, 2011 年末 CSDN 等网站的“泄密门”事件说明 Web 数据库信息泄露已成为威胁企业安全的主流因素之一. Web 信息系统前期“重应用, 轻安全”使得黑客可以轻松的通过脆弱的 Web 信息系统漏洞窃取数据.

访问控制是对用户进行安全审核的重要手段, 其健壮性和粒度直接影响到数据的安全性. 传统的访问控制分为系统级和应用级. 系统级管理需要专门DBA或网络管理员人工定义数据库访问用户的权限或角色[1,2], 优点是可以根据用户工作场景随时修改用户权限, 缺点是人工使用命令或第三方软件操作, 办公效率低下.

在应用级的安全管理方面, 基于角色的访问控制方法(Role-Based policies Access Contro1, RBAC)是公认的解决大型企业统一资源访问控制的有效方法[3-5], 但是控制粒度为表级, 粒度过于宽泛.

为了避免上面两种方法的缺点, 本文结合 Oracle 虚拟专用数据库技术, 提出了一种细粒度访问控制方法, 并设计了 DBA 使用的访问控制系统, 解决了用户对访问控制细粒度的要求.

1 设计思想

对于数据访问要求高且灵活的企业信息管理系统,

① 基金项目:陕西省教育厅科研计划项目(12JK1055)
收稿时间:2013-02-04;收到修改稿时间:2013-03-26

不仅不同用户需要不同的授权, 同一用户在不同工作场景也需要进行不同授权. 即不同用户执行同一 SQL 语句, 看到的数据是不同的. 传统做法是采用视图技术, 但视图定义过多, 给 DBA 管理带来不便. 在不修改应用软件代码的情况下, 如临时需要定义用户权限, DBA 必须使用命令操作, 十分不便, 为了不修改前端数据库应用程序同时又可以提高 DBA 工作效率, 我们决定使用虚拟专用数据库技术开发访问控制系统, 以适应企业灵活多变的业务需求.

1.1 虚拟专用数据库

虚拟专用数据库 (VPD) 提供了角色和视图无法提供的行级访问控制. 对于互联网访问, 虚拟专用数据库可以确保在线客户只能看到他们自己的数据. Web 托管公司可在同一 Oracle 数据库中维护多个公司的数据, 但只允许每个公司查看其自身数据.

在企业内部, 虚拟专用数据库可在应用程序部署方面降低拥有成本, 也可以在数据库服务器一次实现安全性, 而不需要在访问数据的每个应用程序中分别实现安全性. 因为是在数据库中实施安全性, 所以不管用户访问数据的方式如何, 安全性均较以前更高. 客户端工具或新报表生成程序的用户不再能绕过安全环节[6].

1.2 VPD 工作原理

将一个或多个安全策略与表或视图关联后, 就可以实现虚拟专用数据库. 对带安全策略的表进行直接或间接访问时, 数据库将调用一个实施该策略的函数. 策略函数返回一个访问条件(Where 子句), 即谓词. 应用程序将它附加到用户的 SQL 语句, 从而动态修改用户数据访问权限.

通过编写一个存储过程, 我们可以将 SQL 谓词附加到每个 SQL 语句来实施 VPD. 例如, 如果张三(他属于9311701班)输入 SELECT * FROM tbstudent 语句, 则可以使用 VPD 添加 Where classname = ‘9311701’子句, 这样, 就实现了对该语句的行级别访问权限的控制.

VPD 技术除了可以实现行级访问控制, 也可对列数据进行访问控制, 我们只需将策略函数与一个列相关联, 另外 VPD 允许一张表与多策略函数对应, 此时函数返回的所有谓词以 AND 作连接操作.

2 访问控制系统模型

企业管理信息系统运行过程中, 不同工作场景对数据表的数据有不同的访问控制要求, 而这仅通过数据库管理系统和应用程序中定义的角色和权限机制是很难实现的. 目前, 大多数访问控制仍使用应用软件的登录子系统和用户访问控制子系统完成的, 访问控制模型如图 1 所示.

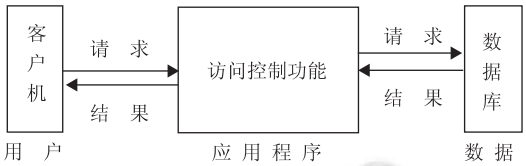

图 1 应用软件数据访问控制方法

该访问控制模型即使使用权限矩阵[7]来合理分配用户对数据的访问, 但还存在如下不可避免的缺陷.

(1) 目前基于应用程序的角色定义, 权限定义粒度过于宽泛.

(2) 用户更换工作场景, 增加授权必须更改程序代码, 成本高, 维护不便.

(3) 不能随时提升、撤销用户的某类权限.

为了实现基于行级或列级数据的屏蔽, 避免用户使用客户端软件 SQL Plus 操作 Oralce 数据库, 我们采用细粒度访问控制方案, 使用 VPD 技术为数据库管理员开发了访问控制系统. 相对于图 1 模型中提供的访问控制功能(一般由登录、权限、角色定义功能实现), 图 2 中的访问控制系统模型可以实现用户管理、政策制定、政策分配和回收功能, 从而可以按需随时定义各用户的数据访问策略. 同时, 在原 RBAC 访问控制模型三个实体: 用户、角色和权限管理基础上增加了表和表策略函数的管理.

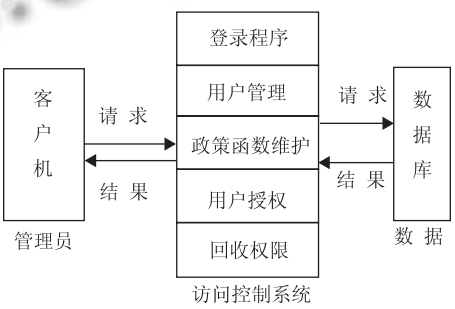

图 2 访问控制系统模型

该模型的处理流程是创建用户、指定目标表、定义策略函数、分配策略函数给用户, 由于是细粒度控制, 分配和回收的不再是角色或权限, 而是策略函数.

访问控制系统由DBA用户操作, 通过图形化的界面把用户与表之间的数据的访问关系显示出来. 这样,

DBA 不必去使用数据库管理系统提供的客户端软件和复杂的语法就可以定义每个用户针对数据表的政策函数, DBA 用户可随时按照企业需求, 分配和回收用户访问权限.

该访问控制系统的用户管理功能还实现了最终用户与数据库用户一一映射, 应用系统最终用户同时也是数据库用户, 因此能有效利用 Oracle 数据库内置的强大安全机制, 如基于角色的用户权限管理和安全审计等功能特性. 采用“最终用户与数据库用户一一映射”的用户管理设计模型与 Oracle VPD 安全特性相结合的方法进行访问控制系统设计, 将传统的在应用程序中定义和实施的安全策略转移到数据库中, 从根本上消除某些软件能绕过应用程序中安全策略而直接访问数据库所带来的安全隐患, 因而有效地提高系统的安全性. 同时, 用户的身份认证、访问控制在由数据库中自动实现, 因而能显著提高软件开发效率.

3 访问控制系统设计实例

3.1 应用描述

为了验证访问控制系统方案的可行性和性能. 笔者结合校级学生成绩管理系统进行了开发和测试. 学生成绩管理系统操作人员分为管理员、教务、教师和学生四类用户. 以教师为例, 学校的政策允许所有教师可以访问本学院所有班级学生的成绩信息, 但在最近, 对该政策做了改变只允许教师访问特定的班级和课程, 为了使应用程序符合新的政策, 必须对原应用程序进行修改, 我们有下面三种选择:

(1) 修改应用程序代码, 使所有 SQL 语句都包含一个 WHERE 子句. 但如果将来政策又有变化, 则必须再一次修改代码, 所以从长远考虑这不是一个好的解决方案.

(2) 保持应用程序不动, 用判定词创建表的视图, 并用与表名一样的名字为这些视图创建同义词. 从应用程序不变更和安全性的角度来看这种方法比较好, 但可能难于管理, 因为有大量潜在的视图需要跟踪和管理.

(3) 保持应用程序不动, 结合 VPD 技术编写细粒度访问控制系统.

3.2 在 ORACLE 数据库上创建 VPD 的步骤

VPD 包含两个要素: 策略(Policy), 策略函数(Policy Function). 其中策略(Policy)用于管理(如添加, 删除, 修改)对哪些对象(表或视图)执行行级别安全控制. 为了实现 VPD, 可以使用如下步骤创建 VPD, 具体 PL/SQL 代码, 请参考[8].

(1) 创建称为 VPD 的用户账户

创建称为 VPD 的用户账户, 该账户具有创建上下文和维护策略函数的权限, 也是控制管理系统内置的用户.

(2) 创建应用程序上下文

使用 create context 命令, 可以创建应用程序定义的属性的名称, 这些属性用于实施安全策略. 此外, 还可以定义函数和过程的程序包名称, 这些函数和过程用于设置用户会话的安全上下文.

(3) 创建登录触发器

为了确保针对每个会话设置上下文变量, 可以使用登录触发器来调用与该上下文关联的过程.

(4) 策略函数定义

策略函数就是定义用于生成谓词的函数, 这些谓词将附加到受保护表的每个 select 语句或 DML 命令. 用于实现谓词生成的函数有 2 个参数: 受保护对象的拥有者、拥有者模式中对象的名称. 一个函数只可以处理一种操作类型的谓词生成, 例如 select, 或者可以适用于所有的 DML 命令, 这取决于该函数如何关联受保护的表.

(5) 添加策略

在 Oracle 中应用 dbms_rls 包实现数据访问控制, 可以使用该包 add_policy 函数将策略函数加入到该包中以获得执行的能力.

3.3 访问控制系统部署

该访问控制系统的应用环境如图 3 所示, 一台 Linux 服务器作为数据库服务器, 一台 Window2003 Server 作为 Web 服务器. 业务用户通过 Internet/Internat 访问 Web 应用. 访问控制系统安装在 DBA 的工作站上, DBA 通过访问控制系统来管理用户和制定、分配和回收用户数据访问策略.

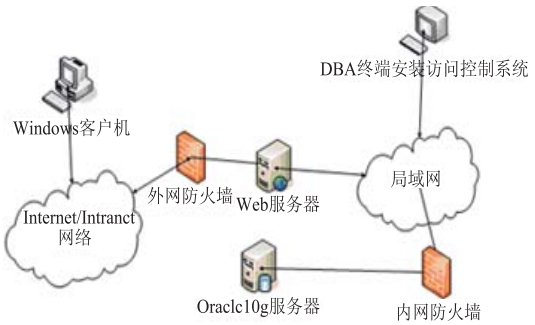

图 3　访问控制系统部署图

3.4 性能分析

通过使用访问控制系统建立 VPD 访问控制策略，并将其应用到成绩表上，实现预期结果：当教师登录数据库后，发出查询 select * from tbscore，只能查询该教师所授课的学生成绩记录. 在实际应用中，除了考虑数据访问安全性，还须考虑应用 VDP 技术对数据操作执行效率的影响. 为探讨这种基于行策略的安全机制是否会影响查询等数据操作的执行效率，我们模拟了三种 SQL 查询应用场景，并统计其在不同记录数下的平均执行时间. 这三种查询请求分别为:

(1) 不使用访问控制策略，模拟应用程序中按教师姓名查询学生成绩的 SQL 语句:

select * from tbscore where tid in (select * from tbteacher where tname='王宏');

(2) 不使用访问控制策略，模拟应用程序中的按教师编号查询学生成绩的 SQL 语句:

select * from tbscore where tid=’20114’;

(3) 应用 VPD 行访问控制策略，直接发出查询语句:

select * from tbscore; 实验结果如图 4 所示.

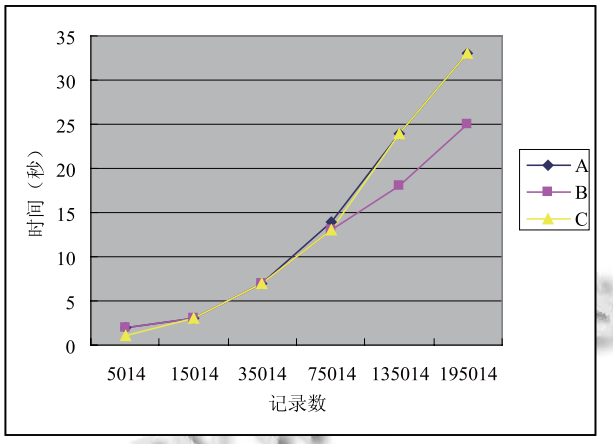

图 4 VPD 执行策略效率统计图

实验中我们使用 DataFactory 数据自动生成工具产生数据，分别使用上述三种方法进行数据查询，实验可以看出，在查询记录数小于 10 万条以下的情况下，在表中建立策略方法的查询效率略优于应用程序中用查询条件限定的方法的查询效率. 随着查询记录数的增加，基于策略的访问控制的执行效率开始低于应用程序的实现效率，但在实际应用中，学生成绩表(加载策略的表)是以分院系存放的，按学院 2000 人 1 学年 10 门课程，一届学生 4 年成绩记录数在 8 万条左右，所以用基于访问控制策略的应用从安全性和执行效率上都要优于基于应用程序的访问控制.

4 结语

数据安全是一个不断研究和发展的课题，本文结合 Oracle VPD 技术实现了细粒度的访问控制系统，数据的安全性大为提高，DBA 工作效率也大为提升，从而有效缓解了安全与灵活应用要求的矛盾. 实际应用中的实例要比上述实现模型的实例更为复杂、譬如经过授权教师允许查询其它教师授课班级学生的成绩信息以及一些系统的超级用户可绕过控制策略等问题都是需要进一步研究的内容.

参考文献

1 吴孝丽,周焱,耿惊涛.ORACLE 数据库安全策略和方法.煤炭科技,2011,30(5):113–115.

2 魏立峰,孟凯凯,何连跃.面向用户角色的细粒度自主访问控制机制,2009,29(10):2809–2811.

3 吴江栋,李伟华,安喜锋.基于 RBAC 的细粒度访问控制方法.计算机工程,2008,34(20):52–54.

4 范明虎.樊红.伍孝金.ASP.net 中基于 RBAC 的通用权限管理系统.计算机工程,2010,6(1):143–145.

5 王成良,姜黎.B/S 应用系统中的细粒度权限管理模型.计算机系统应用,2010,19(7):79–82.

6 Oracle VPD 实现数据细粒度访问.[2010-09-03]. http://www.233.com/oracle/jishu/20100903/110418103.html

7 韩言妮,刘国华,沈兵红.数据库层上的细粒度访问控制技术.燕山大学学报,2006,30(4):345–348.

8 余钢.数据库细粒度访问控制技术应用研究.软件导刊,2009,8(9):145–147.

9 Jeong DW, Jeong H, Park SH. A Security Model Based on Relational Model for Semantic Sensor Networks. Wireless Pers Commun,2011,56:131–146.

10 Sehta N, Jain S. A Fine Grained Access Control Model for Relational Databases. IJCSIT, 2012,3(1):3183–3186.

11 Dhage VN, Shelke RR. Analysis of Fine-Grained Access Control in Database.International Journal of Computer Applications,2012(4):19–21.