万海东　王红卫　(华中科技大学系统工程研究所　430074)

Mangement of User's Privilege on the Basis of Web Information System

基于Web的信息系统的权限设置方式

摘要: 本文分析了基于角色的信息系统权限管理机制,介绍了角色系统管理模型及在此基础上的两种实现方式,并结合SQL SERVER7.0数据库系统以及笔者正在开发的三峡财务公司信息系统进行说明。

关键词: 权限　角色　基于角色的权限访问控制

1　概述

全球信息网的发展,带动了信息系统的变革和发展,越来越多的传统的MIS向Web平台移植。这些与企业MIS系统密切相关的信息系统,多数都需要有数据库的支持,如何确保Web与数据库系统有机的结合,规范合法用户的使用权限与范围,防止故意的非法使用以及无意的误用,已成为在Web上建立信息系统所需要的关键技术和必须重视的问题。文章分析了目前常见的面向用户权限的信息安全技术,提出了Web与数据库集成系统的用户权限管理结构和机制。

一个信息系统的安全管理可分为两个方面:一个是系统级(system)的数据库安全管理,这种安全管理机制是由数据库系统提供的;另一个是应用级(application)安全管理,其主要决定于应用系统的构成。一个大型的信息系统具有许多功能和许多用户,这些用户处于不同的岗位、不同的级别,他们从信息系统中获取信息与处理信息的职权也不同,这就要求应用系统提供一种权限管理机制,控制各种用户使用系统的权力访问。

但是,由于企业具有大量的信息和复杂的组织结构以及人员的流动性,使得用户权限管理也变得异常复杂与繁琐。笔者根据基于角色的访问控制基本模型,结合正在开发的三峡财务公司信息系统,提出了两种基于角色的用户权限设置方式。

2　基于角色的权限控制模型

数据库的权限管理是一项十分艰巨的工作,系统中的用户要有一定的权限集,才能进行工作。权限管理的传统做法是将相关权限直接授予用户,所谓权限就是对数据库执行某种操作的许可,这里用户与权限之间是一种多对多的关系,这种机制对于一个具有诸多用户、诸多权限的数据库系统来说,管理的任务是相当繁重的,且不灵活,一旦组织结构或安全需求有所变动,管理员必须跟着进行复杂而繁琐的授权变动,且容易出现一些意想不到的安全漏洞。

权限管理的另一种做法是在用户和权限之间设置相应角色。一个用户可以赋予若干角色,一个角色也可以被赋予给若干用户,用户与角色之间是多对多的关系。一个角色可以拥有若干权限,一个权限也可以被赋予给若干角色,角色与权限之间也是多对多的关系。其模型结构如图1所示:

基于角色的权限管理的基本特征就是根据安全策略划分出不同的角色,对于每个角色分配不同的权限,并为用户指派不同的角色,用户通过角色间接地对信息资源进行许可的相应操作。简单地说,角色就是一个已命名的权限集合,将角色授予某用户,那么分派给角色的所有权限将同时都授予该用户。

基于角色的访问控制将极大地减少权限管理的负担和代价。不妨做一个简单计算:假设有M个权限及N个要被授予权限的用户。如果把每个权限直接授予每个用户,那么需要进行M×N次授权。如果把每个权限授予某个角色,再把该角色授予每个用户,那么只需进行M+N次授权。若有100个权限及50个用户,那么将是5000:150,当M、N数量增加时差距将更加显著。

另外,基于角色的权限管理使得系统的安全机制具有很大的灵活性。如果为机构的每一职务定义一个数据库角色,角色具有相应职务所拥有

©《计算机系统应用》编辑部 http://www.c-s-a.org.cn

的操作权限，然后根据需要把这些角色授予具有该职责的各用户，即使职务相对应的权限发生了改变，也只需简单地修改角色的权限，而不必修改该职务的每一个用户的权限；或者某用户的职务改变了，也只需报消该用户原来的角色，重新授予新职务所对应的角色。

3 角色权限设置的两种方式

一种是利用SQL SERVER 7.0数据库自身的权限控制机制，直接在数据库系统中设定用户、角色及权限。另一种是通过应用程序判断登录用户的权限，这种方式需要在数据库中建立若干信息表。下面将分别说明这两种方式。

3.1 数据库权限控制方式

(1) 建立login帐号及赋予固定的服务器角色。一个用户要访问SQL SERVER数据库首先要登录到数据库服务器，这需要登录账号login。SQL SERVER7.0提供存储过程sp-addlogin (@login，@password，@database，@language，@sid，@encryption-option)添加登录帐号。

用户连接到数据库服务器后，可以将系统的固有角色分配给他，如生成和修改数据库的权利、增加和删除用户的权利等等。相应的存储过程为sp_addsrvrolemember (@login，@role)。

(2) 建立数据库用户。对于已经登录到服务器上的用户，还必须对相应数据库建立数据库访问用户user，并赋予该用户某些角色。系统提供的存储过程为sp_adduser(@login，@user，@role)。

(3) 建立和设置新的数据库角色。各个数据库自身都提供一些初始角色，例如对数据库的查询、数据库的读写控制等。但是MIS系统往往需要控制用户对某些具体表的访问权限，如对表A可以使用select、insert语句，但不能使用delete、update语句。因此，系统自身带有的少数角色无法满足需要，我们必须建立自己的角色。建立新角色的存储过程为sp-addrole (@role,@user)。

由于给新建角色赋予权限没有系统提供的存储过程，需要自己编写脚本，因此系统管理员可以直接在SQL SERVER7.0环境下对各个角色设定permission。

以上为建立用户权限的过程。若要删除用户、角色、登录名等，可用sp-droplogin (@login)、sp-dropuser (@ user)、sp-droprole (@role) 等相应的存储过程，其操作步骤与建立时的步骤相反。

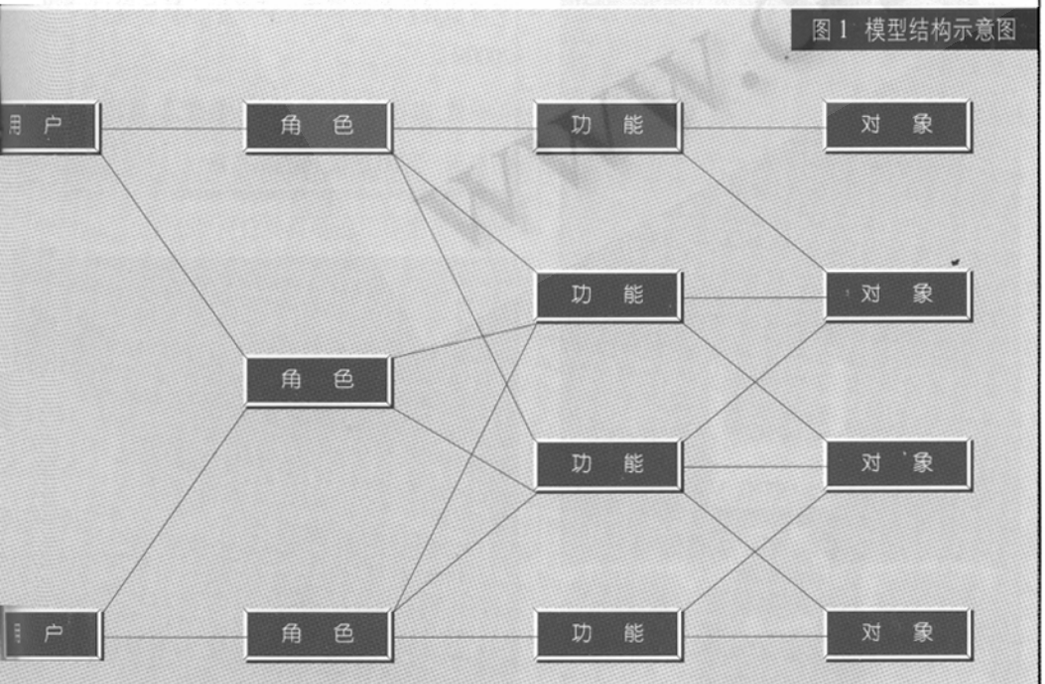


图1 模型结构示意图

在笔者开发的系统中，权限的控制和分配由系统管理员维护，用户只能通过提供的客户端程序增加新用户和修改自己的密码。用户登录系统后，只能进行对指定表、视图和存储过程的操作。

数据库权限控制方式的缺点在于：系统管理员必须给每一个角色赋予对每一个表的四种具体操作行为，如果表的数目比较多，新增角色初始化的工作将比较繁琐。

3.2 应用程序控制方式

在该控制方式中，所有用户都使用数据库系统管理员 (并非MIS系统管理员) 的帐号登录，可以访问数据库中的一切资源，省去了数据库内部的权限设置工作。只要在数据库中建立若干用户信息表及权限信息表，通过应用程序判断该用户是否具有相应权限即可。

在笔者开发的信息系统中，共建立了五个表用来存储用户和权限的信息：

(1) 用户代码表 (t-yhdm)：保存用户登录名、登录密码、用户真实姓名等信息。用于判断用户是否具有登录MIS系统的权限。

(2) 用户组别代码表 (t-yhzbdm)：相当于数据库系统中角色的地位，处于同一组中的用户具有相同的权限。一般是一个部门分配一个组别。如果部门中存在具有特殊权限的用户，可以将他分离出来，同时添加一个新的组别。

(3) 用户组别对照表 (t-yhzbdz)：用于将用户和用户组别联系起来。同一用户可以对应不同的用户组，是多对多的关系。

(4) 用户组别权限表 (t-yhzbqx)：对各个用户组分配权限，也是多对多的关系。该表包含用户组别代码和权限代码两个字段。

(5) 权限代码表 (t-gndxdm)：该表将系统的所有权限细分成许多具体的操作，例如，财务公司的信贷部由台帐录入和报表查询两大

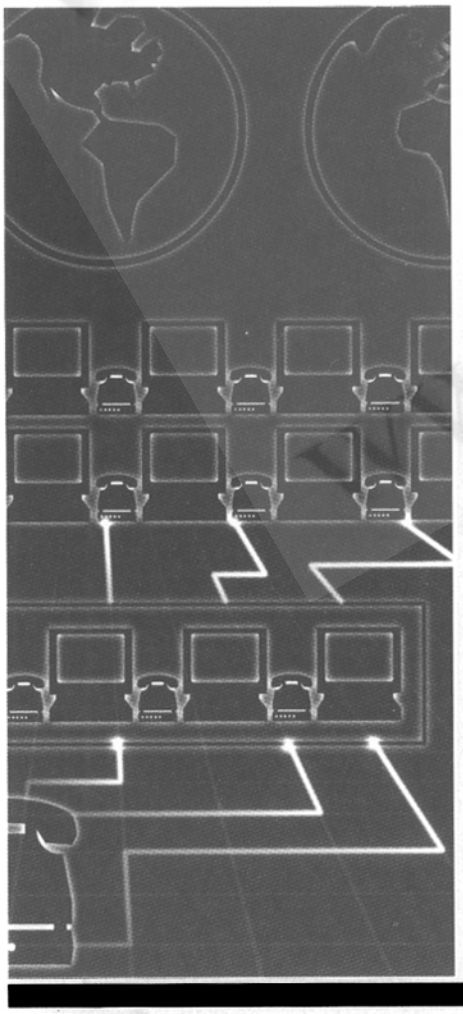
©《计算机系统应用》编辑部 http://www.c-s-a.org.cn

部分组成，录入部分又可分为五种台帐的录入，查询部分包括十五张报表，所以共可以细分为二十种权限对象。

这五个表都是由系统管理员维护的，该控制方式的缺点是所有用户都通过数据库管理员的帐号登录，因此，该帐号和密码都要集成到应用程序中，这样就不方便管理员事后修改，程序开发人员如果在开发过程中掌握了密码，他在开发完程序后就有可能对数据库进行破坏。

对于传统的C/S两层结构的信息系统，例如用PB开发的系统，由于程序只有客户端程序和服务器端程序两部分，帐号和密码信息必须集成到程序中，因而无法解决上述问题。

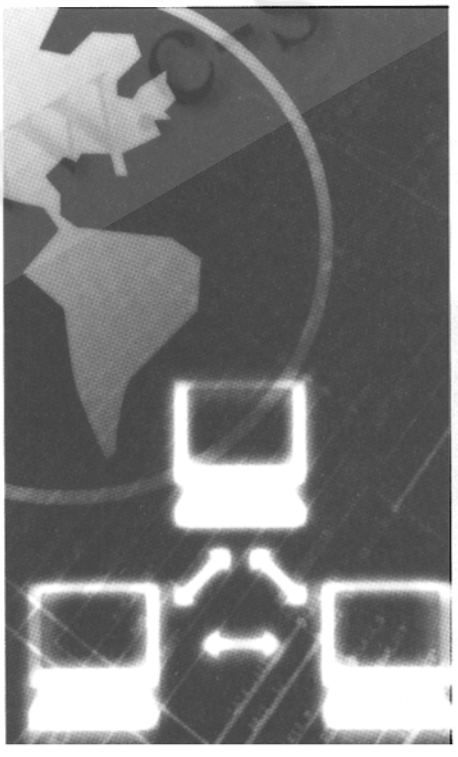
在笔者开发的财务公司信息系统中，我们采用的是B/S三层结构模型，使用JAVA语言进行设计。由于是三层结构，我们可以将数据库管理员的帐号和密码信息置于第二层，即应用服务层。



具体解决方式如下：

(1) 将数据库管理员的帐号和密码保存为资源文件的形式，而不是集成到程序中。将该资源文件置于应用服务层。该文件包括服务器地址、端口号、JDBC驱动程序 (如 driver = sun.jdbc.odbc.JdbcOdbcDriver)、连接字符串 (conStr = jdbc：odbc：sxcw；uid = sa；password = webtrade) 等信息。

(2) 每次执行访问数据库的操作时，首先从该资源文件中读取帐号信息，并连接到数据库。然后根据登录用户的信息，从五个用户信息表中获取用户的权限，判断用户是否具有执行该操作的权利。判断用户权限的SQL语句如下：" SELECT Count(*) FROM t-gndxdm, t-yhzbdz, t_yhzbqx WHERE t_yhzbqx.gndxdm = t_gndxdm.gndxdm And t_yhzbdz.yhzbdm = t_yhzbqx.yhzbdm and t_yhzbdz.yhdm = ´YHDM´ And LOWER (t_gndxdm.bz) = ´GNQXDM´"，其中´YHDM´和´GNQXDM´分别为从网页上获得的用户名称和该用户将要进行的操作。



(3) 在服务器端可以提供仅供系统管理员使用的数据库密码修改程序。一旦管理员修改了数据库密码，他还必须相应的修改资源文件中的帐号和密码信息。该资源文件可以用任何文本编辑器编辑。

经过上述操作后，前面提到的客户端用户和程序开发人员都不能对资源文件进行访问，从而很好的实现了对重要信息的保密。

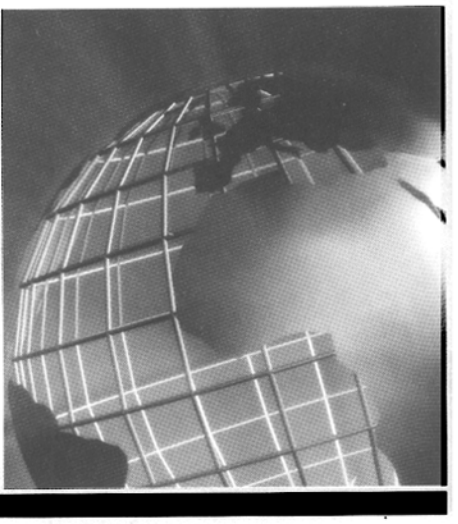
4 结语

本文提出的两种权限设置方式，能够使信息系统的实际使用部门中的岗位角色，很好地映射到信息系统中，而且这种权限管理模型便于统一、敏捷地管理各岗位角色和用户的权限，大大简化了用户授权管理。尤其是第二种方式在笔者开发的三峡财务公司信息系统中得到了使用，实践证明该方式适合企业管理的特点，提高了开发效率，降低了系统权限管理的复杂性，系统维护容易，获得了很好的应用效果。

参考文献

1 欧阳星明、张华哲，大型网络MIS系统中基于角色的权限管理，计算机工程与应用，2000.4。

2 蔡菁，基于角色的数据库权限管理，微型电脑应用，2000年第16卷第八期。



©《计算机系统应用》编辑部 http://www.c-s-a.org.cn