第 35 卷 第 11 期 计 算 机 工 程 2009 年 6 月
Vol.35 No.11 Computer Engineering June 2009
·安全技术· 文章编号：1000—3428(2009)11—0140—03 文献标识码：A 中图分类号：TP391.7

# 协同环境中基于 RBAC 模型的访问控制策略

付喜梅
(韶关学院数学系，韶关 512005)

**摘 要：**协同系统具有动态性和群体性的特点，其权限管理比传统软件系统复杂。传统数据库系统中的访问控制机制比较简单，远不能满足协同系统的要求。针对共享资源访问控制策略的授权方式复杂、授权粒度不细致的问题，引入角色机制，把角色访问并发控制策略应用到系统中，设计基于角色的系统功能权限的位映射算法。该算法降低了授权管理的复杂性，增强了系统安全性。
**关键词：**协同；角色控制；权限；访问控制

# Access Control Strategy Based on RBAC in Collaborative Environment

FU Xi-mei
(Mathematics Department, Shaoguan University, Shaoguan 512005)

**【Abstract】**Collaborative system has dynamic and groups characteristics, whose rights management is more complex than the traditional software system. Traditional database system access control mechanism is relatively simple, which is far from satisfying the collaborative system. Aiming at the problems of the complicated authorizing way of access control strategy to shared resource, authorized granularity's incompact, this paper introduces role mechanism. It applies role-based access control concurrent strategy to system, and puts forward role-based bit mapping system function right algorithm. The algorithm reduces complexity of authority management efficiently and strengthens system's safety.
**【Key words】**collaborative; role control; permission; access control

## 1 概述

协同设计系统是多用户、多机、多任务共享环境的集成体系，然而这样一个典型的多用户、多任务的分布式工作环境给系统带来了严重的安全性问题[1-2]。协同设计中协同用户需要对共享环境中的数据进行访问，然而谁能够以何种方式来访问何种数据是需要规定的。不同的身份、专业特长、任务分工的不同用户应有的访问权限是不同的；而且不同数据的共享范围和安全性级别要求也不同。因此，协同设计系统要制定有效的访问控制策略，防止非法用户的侵入和合法用户对资源的非法使用。

传统的访问控制直接将访问主体和客体相关联，将用户与访问权限直接相联系，当组织内人员新增或有人离开时，或者某个用户的职能发生变化时，需要进行大量授权更改工作。在访问控制中引入角色的概念[3]，可以减少授权管理的复杂性，降低管理开销，而且还为系统管理员提供一个易于实现的安全策略。一旦权限初始设置好之后，就不再需要作大的调整。因为权限直接与角色相对应，而不是用户。即使公司的人员调动频繁，但是工作岗位本身是很少变化的。由于权限控制是基于工作岗位，而不是基于职员的，因此人员的调动并不会太多地影响到权限控制机制。

## 2 基于角色访问控制的基本原理

基于角色的访问控制(Role-Based Access Control, RBAC)技术是 20 世纪 90 年代后出现的一种强制性访问控制技术。RBAC 模型的基本思想是：在用户和权限之间引入角色机制，建立用户和权限两者之间的间接映射，把角色分配给用户，将权限与角色绑定，根据需要赋予用户不同的角色，即用户和用户组的权限取决于用户的角色[4-5]。因此，可根据用户在系统中的角色进行访问授权与控制。

RBAC 通过在访问主体和客体中间加入角色来沟通主体与客体。RBAC 模型结构[6]如图 1 所示。

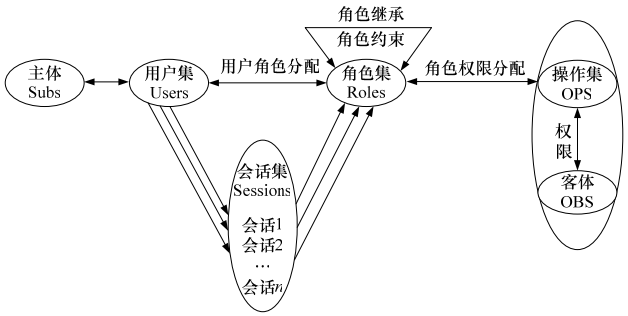


**图 1 基于 RBAC 控制模型结构**

在基于角色权限的访问控制模型中，角色作为一个桥梁，沟通于用户和资源之间。对用户的访问授权转变为对角色的授权，然后再将用户与特定的角色联系起来，并通过角色沟通主体与客体。一个用户可以同时担当多个角色，一个角色可被多个用户所拥有。每个角色可执行多种操作，每个操作也可以由不同的角色执行。一个用户可拥有多个主体，但每

**基金项目：**广东省自然科学基金资助项目(7301275)
**作者简介：**付喜梅(1979－)，女，硕士，主研方向：计算机支持协同设计技术
**收稿日期：**2008-10-20 **E-mail：**afxmabc@yahoo.com.cn

个主体只对应一个用户。每个操作可施加于多个客体，每个客体也可以接收多个操作用户。为了对系统资源进行存取，用户需要建立会话。一个会话将一个用户与他所对应的角色集中的一部分建立映射关系。

## 3 基于角色的系统功能权限分配建模

系统的权限主要分为 2 类：数据权限和功能权限。其中，数据权限规定对数据对象有何种操作权限，通过对访问数据库对象的限制来保证数据库的安全性；而功能权限规定使用系统时哪些功能可用，哪些功能被禁止，通过限制用户使用系统的功能来间接保证数据的安全。

本系统建立了一种基于角色的系统功能权限分配模型。即把系统具有的功能分配给角色，从而间接地实现角色对数据对象的操作权限控制。原来用户、角色、权限是单纯地由后台数据库管理系统来管理，通过引进功能权限，用户可以经由界面可视化地控制数据库的数据权限，提高了系统管理员的透明度，扩充了角色的内涵，增强了系统的安全性。基于角色的系统功能权限分配模型如图 2 所示。

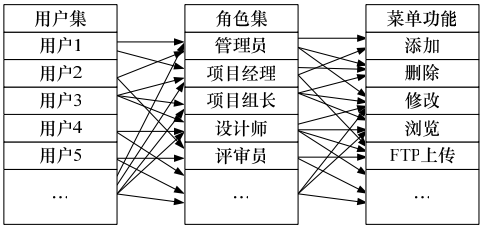


**图 2　基于角色的系统功能权限分配模型**

### 3.1 基于角色的系统功能权限位映射算法

本文把角色和权限合并到一个表中，基于角色的系统功能分配模型，采用位映射法来组织菜单功能权限信息。把所有的功能菜单项按照一定的顺序从零开始计数编码，每项菜单功能权限用一个布尔位 $p_i$ (用 $p_i$ 表示菜单功能权限项中的第 $i$ 个菜单功能权限)来表示，当 $p_i=1$ 时表示允许操作，用 $p_i=0$ 表示禁止操作。这些代表功能权限的位 $p_i$ ( $i=1,2,\cdots$ ) 组合成二进制串，形成角色权限码，用 $P$ 表示。当系统确认当前用户为合法用户时，便取出该用户所有的角色权限值，并根据该值和功能项编码的映射关系得出该用户每种功能权限的情况。表 1 给出了该算法原理。

**表 1　角色权限分配**

| 角色 | 权限 $p_1$ | 权限 $p_2$ | … | 权限 $p_j$ | … | 角色的权限集 |
|---|---|---|---|---|---|---|
| 角色 1($r_1$) | 1/0 | 1/0 | … | 1/0 | … | $P(r_1)=p_1p_2\cdots p_j\cdots$ |
| 角色 2($r_2$) | 1/0 | 1/0 | … | 1/0 | … | $P(r_2)=p_1p_2\cdots p_j\cdots$ |
| … | … | … | … | … | … | … |
| 角色 $i$($r_i$) | 1/0 | 1/0 | … | 1/0 | … | $P(r_i)=p_1p_2\cdots p_j\cdots$ |
| … | … | … | … | … | … | … |

角色 $i$ 的功能权限集用 $P(r_i)$ 表示，其对应的第 $j$ 项功能权限用 $P(r_i)_j$ 表示。若 $P(r_i)_j=1$，说明角色 $i$ 拥有第 $j$ 项功能权限；若 $P(r_i)_j=0$，则角色 $i$ 不享有第 $j$ 种功能权限。设登录用户拥有 $k$ 种角色，则计算该用户第 $j$ 种功能权限的算法为 $F(j)=\bigcup_i^k P(r_i)_j$。当一个用户具有多个角色时，其操作权限是他所拥有的所有角色的操作权限之和(并集)。权限的位映射算法用 ASP 实现，其具体实现过程如下：

```
<%
'--------角色权限专用环境变量，求取用户角色的权限值------
'从用户角色表中查询出当前用户所具有角色的 ID
Set rs1= Server.CreateObject("ADODB.Recordset")
sql="select roleid from userrole where id ="&session("Uid")
rs1.open Sql,Conn,1,3
i=0
'从数据库的角色表中查找出用户所拥有的角色的权限码并依次导入到数组 b(20)中
dim b(20)
do while not rs1.EOF
  Set rs2= Server.CreateObject("ADODB.Recordset")
    Sql2="select xian from role where roleid="&rs1("roleid")
  rs2.open Sql2,Conn,1,3
  b(i)=rs2("xian")
  i=i+1
  rs1.MoveNext
loop
'把用户拥有所有角色的权限码求取并集赋给权限 session("xian")的会话变量
dim c(80)
for j=1 to 80
  for i=0 to 9
  if mid(b(i),j,1)="1" then
    c(j)=1
    exit for
  else
    c(j)=0
  end if
  next
next
for k=0 to 60
  xian=cstr(xian)&cstr(c(k))
next
session("xian")=xian    '建立用户权限的会话变量
…
Quanxian=session("xian")
right=Mid(quanxian,i,1)    '提取出用户权限码的第 i 位
If right="1" then
   '说明该用户拥有第 i 项系统功能权限
   …
Else
   '说明该用户不具有第 i 项系统的功能权限
   …
End if
…
%>
```

采用可靠的访问控制技术是系统安全的必要保证。基于角色的访问控制技术与传统的访问控制技术相比具有明显的优势，它可以实现细粒度的访问控制，能够控制具体的资源项。而基于角色的系统功能权限分配策略又把系统的功能权限和数据权限紧密结合起来，增强角色管理的透明度，减轻系统管理员的负担，提高数据库的安全性。

### 3.2 基于角色权限的访问控制实现

将基于角色的系统功能权限控制策略应用到系统中，较好地解决了协同环境中用户授权问题，提高其灵活性和管理效率，具有非常细致的访问控制粒度，提高了系统安全性。图 3 是角色管理界面。

在图 3 中，具有权限的用户可以根据系统需要来定义角色并为角色设置相应的权限，另外也可以修改、删除、浏览

角色的权限。图 4 就是修改角色权限的界面，通过填写各级菜单中的复选框就可以重新给角色授权。


**图 3　角色授权管理界面**

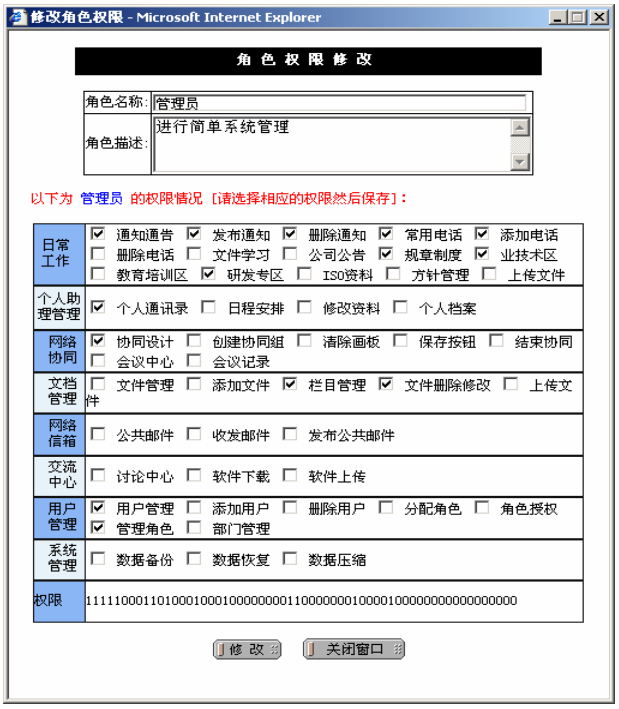


**图 4　角色权限修改界面**

根据基于角色的系统功能的位映射算法，把所有系统功能项按主次分类做成 ASP 中的复选框(CheckBox 控件)。同时对每一项系统功能按顺序从零开始计数编码，通过填写系统功能项的复选框就可以设置角色的菜单功能权限范围。由于复选框的值是一个布尔值，每一个布尔位根据功能编码映射一个系统功能项，因此所有复选框的值按顺序组合起来就形成了角色的权限码(用变量 *quanxian* 表示)。当用户登录系统，把权限码赋给用户建立的权限会话变量 *session*("*xian*")中，利用函数 *mid*("*quanxian*", *j*, 1)在角色权限码中求取第 *j* 位的数值。若为 1 则表示拥有第 *j* 个权限访问功能；若为 0 则表示没有此访问功能。因此，当用户点击菜单时，进行 *mid*("*quanxian*", *j*, 1)值的判断，就可以实现角色权限的控制。当用户登录时根据用户的角色来初始化菜单，同时在用户点击菜单时触发访问权限检查，用 *mid*("*quanxian*", *j*, 1)方法从权限码中取出相应的权限位来控制访问许可与否。

## 4　基于角色权限的访问控制流程

对于用户的权限控制，通过主动权限控制和被动权限控制 2 种方式来具体实现。所谓主动权限控制，就是在用户登录系统时，系统主动获取该用户所拥有的功能权限。在用户进入系统后，只显示那些用户有权限访问的功能菜单，将那些用户不具有访问权限的功能菜单设置为对该用户不可见。这样，用户在进入系统后只能看到他拥有访问权限的那些功能菜单。被动权限控制是指当用户在具体提出访问某一资源的请求时(如要打开某一文档文件)，系统再被触发去检查该用户是否具有访问该资源的权限。访问控制流程如图 5 所示。

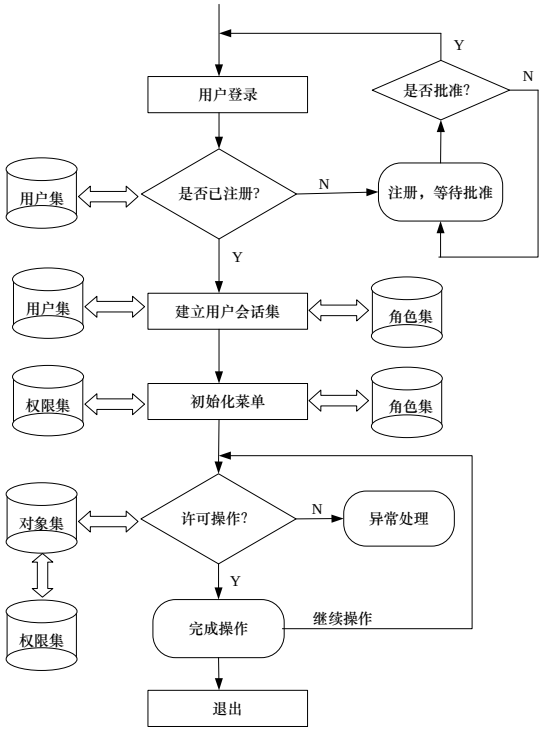


**图 5　用户访问控制流程**

## 5　结束语

本文基于角色访问控制的原理，建立了基于角色的系统功能权限分配模型，给不同角色赋予不同权限，使得在用户与权限之间建立间接的映射。提出基于角色的功能权限的位映射算法来实现角色权限的访问控制策略，有效地降低了授权管理的复杂性，提高了权限管理的效率与灵活性，便于系统安全策略的制定。在分布式环境下易于管理，具有良好的可扩展性，是目前实现分布式环境下访问控制策略的有效技术。

## 参考文献

[1] Suleiman M, Cart M, Ferrie J. Serialization of Concurrent Operations in a Distributed Collaborative Environment[C]//Proc. of ACM SIGGROUP Conference on Supporting Group Work. Phoenix, USA: ACM Press, 1997: 435-445.

[2] 李伟琴, 杨亚平. 基于角色的访问控制系统[J]. 计算机应用, 2000, 20(2): 16-21.

[3] 刘怀宇. 基于角色的细粒度访问控制系统的研究与实现[D]. 北京: 北京航空航天大学, 2002: 26-29.

[4] 严　悍, 张　宏, 许满武. 基于角色访问控制对象建模及实现[J]. 计算机学报, 2000, 23(10): 1064-1071.

[5] 叶锡君, 许　勇, 吴国新. 基于角色的访问控制在 Web 中的实现技术[J]. 计算机工程, 2002, 28(1): 170-172.

[6] 李成错, 詹永照, 茅　兵, 等. 基于角色的 CSCW 系统访问控制模型[J]. 软件学报, 2000, 11(7): 931-937.

编辑　顾逸斐