文章编号:1000 - 582X(2003)05 - 0055 - 04

企业信息集成平台电子文档安全存储管理技术*

王成良[1],曾 德[2]
(1. 重庆大学 软件学院,重庆 400044; 2. 重庆大学 计算机学院,重庆 400044)

摘 要:对企业中产生的重要敏感电子文档进行数据库管理,在安全性问题日益突出的今天,将具有重要意义。电子文档进行数据库管理后,运用基于用户 - 角色的授权访问机制进行访问虽然可行,但由于企业电子文档的动态生成特性,需要不断地进行授权维护,从而不能提供授权灵活性。提出了一种自主访问控制的实体授权关系的树模型,将其同用户 - 角色机制相结合简化了授权的复杂性,大大提高了企业电子文档授权管理的灵活性,使其更加符合企业的实际需求。

关键词:基于角色的访问控制; 自主访问控制; 电子文档; 授权访问; 实体授权
中图分类号:TP311.11 **文献标识码**:A

在企业信息化过程中,将会产生大量有关企业在管理、产品设计、制造、销售等各个环节的各类电子文档。将企业电子文档采用数据库方式进行管理除了具有查询快速方便;防止误删除;不会受病毒感染等优点外,对于企业重要、敏感文档通过数据库系统授权访问控制可提高其访问安全性,防止被窃取或遭到破坏[1-2]。

目前对于数据库管理系统的安全访问一般采用基于角色的访问控制(RBAC)来进行授权管理,对不同级别的用户授予不同的角色,每个角色都具有一组各不相同的系统操作权限。这种方式虽然简化了系统授权管理又保证了系统的安全性,但却无法满足企业电子文档管理的特点。作为企业电子文档,它在企业各部门内、各部门间流动共享是不可避免的,是保证企业内部协同工作的基础,对电子文档的授权应根据企业各部门内、部门间合作关系的变化而变化,而这种变化往往是动态的、是事先无法预知的,因此对电子文档的授权机制应是一种动态的授权机制,而不是相对固定的基于用户 - 角色的授权机制。

针对电子文档的动态授权特性,采用了一种将基于角色的访问控制(RBAC)[3]与自主访问控制(DAC)相结合的机制[4],提出了一种实体授权关系的树模型,在 RBAC 的基础上对电子文档单独采用 DAC 从而有效的解决了这一问题,在保证了系统安全性与授权管理简便性的同时,添加了对电子文档动态授权的灵活性。

1 基于角色的访问控制(RBAC)

RBAC 是近年来研究最多,思想最成熟的一种数据库权限管理机制。RBAC 的基本思想是根据企业组织视图中不同的职能岗位划分不同的岗位角色,岗位角色就是在一个信息系统上具有相同操作权限的用户组,不同的岗位角色只能享有由系统管理员或子系统管理员分配的操作权限[5-6]。它将整个系统的操作权限根据系统的菜单项划分成不同的功能原权限,功能原权限就是指不可再分的对系统的操作权限,在系统上表现为某一具体菜单项,而所有这些原权限构成一个功能原权限集合。分配给岗位角色的权限就是同工作岗位的职能相对应的功能原权限集合的子集,各个子集合彼此不相等,对所有子集合进行集合并运算得到的集合为原权限集合。每个用户根据需要可对应一个或几个岗位角色。同时在授予用户岗位角色时必须满足基本安全原则[4],即 1)最小特权原则,指用户拥有的权限不能超出能完成工作所需的权限。2)职责分离原则,指用户不能同时拥有这样两个岗位角色,它们在工作中的职能是互相监督与制约的。

基于角色的功能授权管理模型如图 1 所示。

图 1 中的原权限子集合包含各个岗位角色所具有

* 收稿日期:2002 - 11 - 20
基金项目:重庆市应用基础研究项目(20016809)
作者简介:王成良(1964 -),男,江苏丹阳人,重庆大学副教授,博士,主要从事网络及数据库应用技术研究。

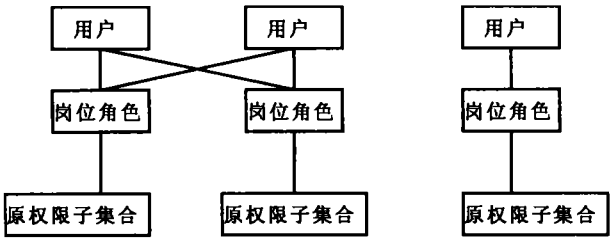


图 1 基于角色的功能授权管理模型

的对系统的操作权限,各子集互不相等且与岗位角色一一对应。岗位角色不与企业中具体的岗位相对应,对系统具有相同操作权限的岗位可归并为同一岗位角色,每个岗位角色对应唯一的原权限子集合。用户是指企业中有权使用系统的员工,由于一个员工可以身兼数职所以一个用户可能同时拥有一个或多个岗位角色,几个员工也可在同一类岗位上工作,具有相同的岗位角色。通过采用这一模型就避免了对系统每一用户逐项授权,使得权限管理更为清晰。

2 基于实体的自主访问控制(DAC)

尽管 RBAC 提供了较好的安全管理机制,但由于企业内部是一个有机的整体,要提高企业竞争力必须保证各部门之间有机的协同合作,因此各类电子文档在不同部门、不同级别的岗位上流动、共享是不可避免的,而且部门间、部门内的协作关系也不是一成不变的。由于 RBAC 根据岗位来确定岗位角色,岗位角色被分配的权限是相对固定的,而且对应一组用户而不是单一用户,不可能根据某一用户权限需要的改变而改变分配给角色的权限,如果将电子文档权限的管理设计成一种固定的模式则不能反映各部门之间、岗位之间合作的不确定性,因此设计出一种能反映部门间、岗位间动态合作关系的电子文档动态授权模型并和 RBAC 相结合就显得十分必要。尽管电子文档种类繁多,不妨将各种电子文档统称为实体,对各种电子文档的授权称为实体授权。实体授权采用 DAC 机制是指文档创建者可以根据用户而不是角色来授予其文档的操作权限,而拥有对该文档操作权限的用户又可以将自身所拥有的操作权限授予其他不具有该操作权限的用户。把对实体不可再分的各种操作权限称为实体原权限,这些原权限构成的集合称为实体原权限集合。每个主体(用户)对某一实体的操作权限的集合都是实体原权限集合的子集。实体的创建者拥有对该实体所有的权限,并作为实体授权关系的起点。由于 DAC 是一种自主式授权,整个授权过程难以控制可能造成企业敏感信息的泄露,因此必须禁止重复授权,即当授权用户(向其他用户授权的用户)授予被授权用户某一实体的某项操作权限时应保证授权用户对该实体具有该项操作权限,而被授权用户不具有该项操作权限,并且该授权关系必须记录在授权日志当中。当要消除某用户对某一实体的某项操作权限时必须由其上级授权用户来执行,消权信息也必须记录在日志内,一旦发生敏感信息泄露可根据授权日志进行授权关系的追踪。虽然 DAC 提供了授权的灵活性但同时也使授权关系更为复杂,图 2 描述了这种授权关系。

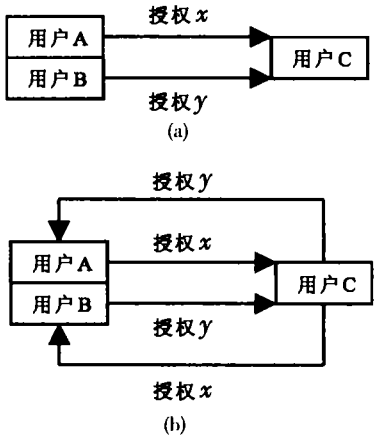


图 2 实体授权模型

图 2 中,用户 A 和 B 起始对某实体 e 各拥有权限 x 和 y,用户 C 对 e 不拥有任何权限,但授权后 C 对 e 却同时拥有权限(x,y),反而比授权者 A,B 更高。当用户 C 拥有权限(x,y)后又可能分别授予用户 A 对 e 的 y 权限,用户 B 对 e 的 x 权限。

除此之外还会出现如图 3 一样的多个用户之间的循环授权。由于上述两种关系使得整个实体授权关系会形成一个网状结构,如果不对这个网状结构进行分解简化会使得整个授权关系逻辑混乱,层次不清大大降低管理效率。通过追踪某一具体实体原权限的授权关系可以发现,如果在这个网状结构上将仅具有该权限的结点(用户)以及这些结点间的授权关系从整个实体授权关系中独立出来,会形成一个以实体创建者为根结点的树结构,假设实体原权限共有 n 个就会产生 n 棵授权关系树,一棵授权关系树就表示实体上一项权限的授权关系,树上的每一个结点表示对该实体具有该项操作权限的用户。如果 n 值不大则通过对这 n 棵树的管理就可以达到对整个实体授权的管理。

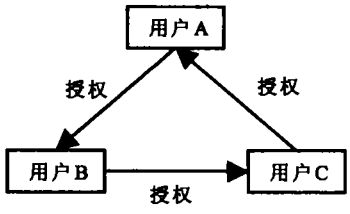


图 3 多个用户之间的循环授权

图 4 为授权关系树图。该图表示在一个有 n 个实体原权限的实体上 n 棵授权关系树中的一棵，其它的 $n-1$ 棵授权关系树都与此类似。假设这是实体权限为 x 的授权树，那么有且只有该树上的用户才具有对实体创建者所创建的实体享有 x 操作权限。由于电子文档的原权限只有有限的几种，所以可以采用这种方法。当对某一用户授予对实体的一种操作权限时，就在该权限树上添加该用户作为授权用户的子结点。当对某一用户消除权限时，必须且只能由其上级结点进行并采用一种级联式消权，即在删除该结点的某项权限之前必须扫描子结点，找出具有该权限的子结点并删除该权限。通过这种递归式扫描一直到叶结点，这样以该结点为起点的子树将被删除。

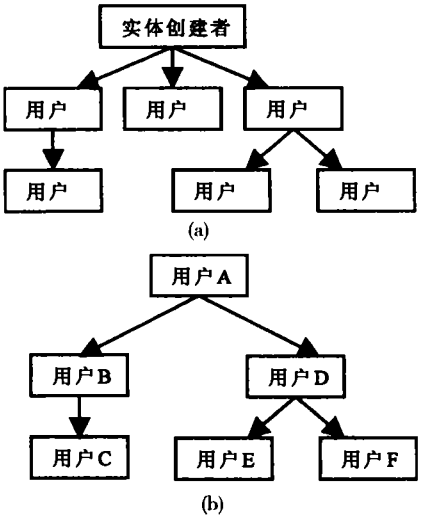


图 4　授权关系树

图 4 是一个从一个授权关系网中分离出来的关于权限 x 的一棵授权关系子树，用户 B，D 的 x 权限只能由用户 A 取消，并且在这棵树上用户 A 也仅能取消其子树的 x 权限（假设用户 B、C 还拥有其它权限），当用户 A 取消 B 的 x 权限时，用户 C 的 x 权限也会被取消，同理当用户 A 取消用户 D 的 x 权限时用户 E、用户 F 的 x 权限也会被取消。其结果分别如图 5（a）和图 5（b）所示。

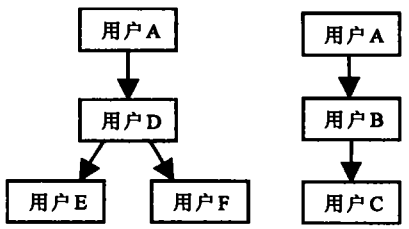


图 5　消权处理结果

这样通过一个基于权限树的 DAC 模型就能灵活地根据企业运作的需要对相关文档进行授权管理、跟踪，弥补了 RBAC 的不足。

3　授权管理实现过程

企业电子文档授权管理数据表结构关系如图 6 所示。方案中每个表的主键 PK 可保证数据表记录的唯一性。每个用户用自己的用户名和密码作为唯一的登录标识，用户登录后得到的用户编码和用户 - 角色表中的用户编码相关联，会得到一个或几个岗位角色编码（如果用户身兼数职就会有几个岗位角色编码），每个角色都有一组固定的系统操作权限，用户登录后这

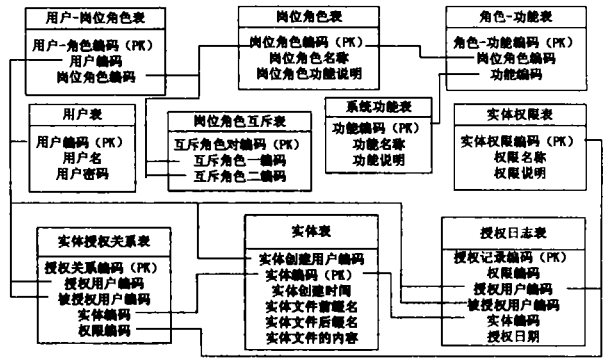
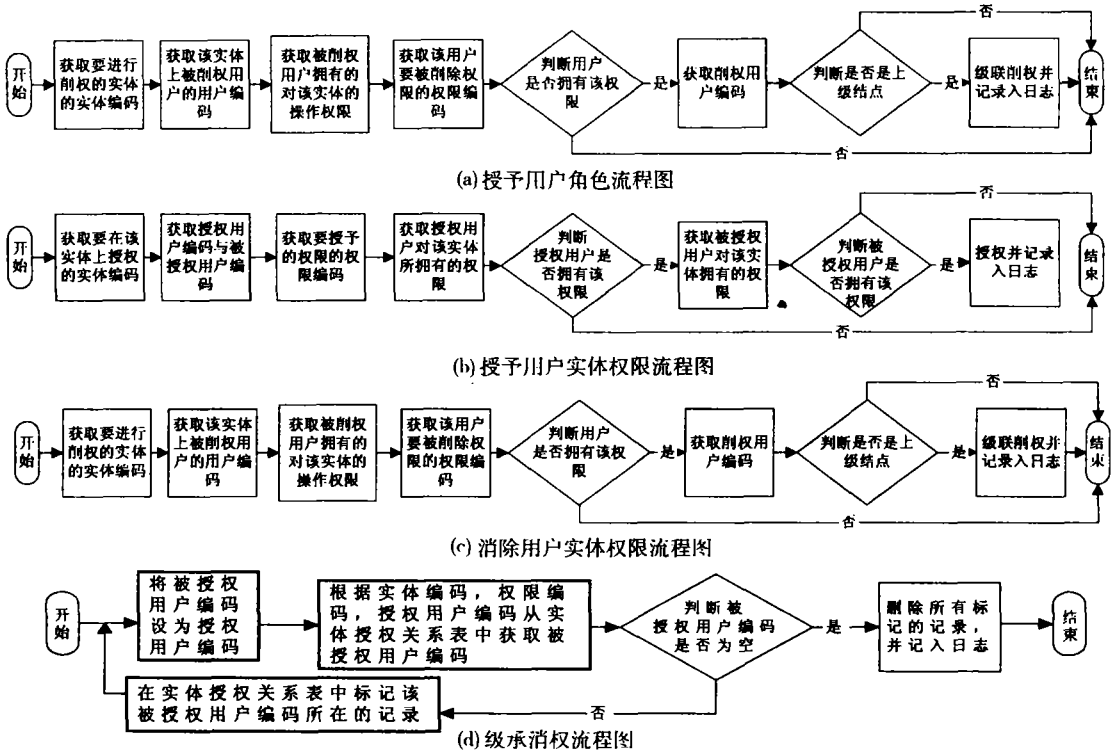


图 6　实体授权数据表结构关系

组权限便作为用户对该系统的操作权限。岗位角色表，角色 - 功能表，用户 - 角色表，岗位角色互斥表由系统管理员或子系统管理员维护，前 3 个表是对角色的基本管理，必须使其满足最小特权原则使得每个角色所拥有的权限不超过完成岗位工作所需的范围，而岗位角色互斥表可以保证职责分离原则的满足，当授予用户角色时必须使得该角色与用户已有的角色不会互斥。任一用户创建了电子文档，该电子文档就会在实体表中对应一个记录，而创建者则拥有对该文档的全部操作权限，他可以向其他用户授予该实体的操作权限，获得该实体的操作权限的用户也可以向其他用户授于自身拥有的权限。每次授权都应在实体授权关系表中插入一条相应的记录，具有相同实体编码和权限编码的所有记录就构成一棵授权关系树。当用户要对某一实体操作时，根据实体编码和被授权用户编码在实体授权关系表中就可以找出该用户对该实体的所有操作权限，从而判断该用户对该实体是否具有该项操作权限。当消权时就必须从进行消权的用户结点开始搜索，找出对应子树以及子树对应的实体授权关系表中的记录，删除所有记录。所有授权、消权操作都必须记录在授权日志表中以备后查。图 7（a）~（c）是电子文档授权访问实现过程的流程图。图 7（d）为消除用户实体权限流程图中的递归过程流程图。



(a) 授予用户角色流程图

(b) 授予用户实体权限流程图

(c) 消除用户实体权限流程图

(d) 级承消权流程图

图 7 授权管理实现过程流程图

4 结束语

RBAC 具有简便的授权管理，强大的角色管理与划分，能满足已有的安全需求原则，而 DAC 则使得授权更为灵活，有利于企业内部运作，大大方便了授权管理。通过将 RBAC 与 DAC 相结合使得对企业电子文档的管理相对地更加安全和灵活。本系统已运用于实际的项目中，运行良好。

参考文献：

[1] 林琪，卢昱. 电子文档安全存储技术研究[J]. 计算机应用. 2000，20(9)：62－65.

[2] 陈庆章，洪宁. 协同工作环境下网上公文传送的安全机制[J]. 计算机工程，2000，26(5)：5－8.

[3] 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)：16－22.

[4] 罗雪平，郑奕莉，徐定国. 一种扩展的基于角色的访问控制模型[J]. 计算机工程，2001，27(6)：106－107.

[5] 张晓辉，王培康. 大型信息系统用户权限管理[J]. 计算机应用，2000，20(11)：36－39.

[6] 洪帆，余祥宣，倪晓俊. 多级安全 RDBMS 的安全策略[J]. 华中理工大学学报，1996，24(1)：41－43.

Authorized Administration of Electronic Documents in Enterprises

WANG Cheng-liang[1] *, ZENG De*[2]

(1. College of Software Engineering , Chongqing University, Chongqing 400044, China;
2. College of Computer, Chongqing University, Chongqing 400044, China)

Abstract: When prominence of security problems increasing, it is of importance to save the sensitive electronic documents into database. The user－role mechanism is employed in the process of managing documents. However, it cant provide the flexibility of authorized access because of the dynamic generating particularity of the electronic documents which need maintaining continually. A tree model of entity authorization is introduced based on discretionary access control mechanism. The approach can grealty simplify the authorization, increase the flexibility and is more accordant with the practical requirement of enterprises.

Keywords; role－based access control(RBAC); discretionary access control(DAC); electronic document,; authorized access; Entity Authorization

(责任编辑 吕赛英)