第32卷 第21期 计 算 机 工 程 2006年11月
Vol.32 No.21 Computer Engineering November 2006
·安全技术· 文章编号：1000—3428(2006)21—0167—03 文献标识码：A 中图分类号：TP309

一种基于角色访问控制的新模型

杨文波，张 辉，刘 瑞
(北京航空航天大学计算机学院软件开发环境国家重点实验室，北京 100083)

摘 要：通过分析RBAC模型在信息量巨大、信息资源变化频繁的应用系统中表现出的不足——安全维护代价过于繁重，提出了一种新的基于角色的访问控制模型——ARBAC模型。该模型具有较低的安全维护代价，良好的安全控制效果，有效克服了RBAC模型在该类系统中存在的不足。

关键词：访问控制；抽象角色；许可分配；模型

A New Role-based Access Control Model

YANG Wenbo, ZHANG Hui, LIU Rui
(National Lab of Software Development Environment, Department of Computer Science,
Beijing University of Aeronautics and Astronautics, Beijing 100083)

【Abstract】 This paper analyzes the drawback of role-based access control model that is onerous secure maintenance costs in the systems which have a large amount of highly updating frequently information. Based on this point, a new role-based access control model——ARBAC has been proposed. This new model has lower secure maintenance costs, well secure control effects, and can effectively overcome the drawback of role-based access control model in the kind of the systems mentioned above.

【Key words】 Access control; Abstract role; Permission assignment; Model

随着计算机网络应用的不断普及，人们不仅要求系统的功能不断完善，而且对系统的安全性也提出了更高要求，访问控制正是实现这一要求的一个重要手段。

传统的访问控制模型有自主访问控制(Discretionary Access Control，DAC)和强制访问控制(Mandatory Access Control，MAC)。DAC[1]是计算机系统中实现最多的访问控制机制，其主要特征是：主体可以自主地把自己所拥有客体的访问控制权限授予其他主体或从其他主体收回所授予的权限；而MAC则根据客体的敏感数据和主体的许可级来限制主体对客体的访问，多用于军用系统。但是，随着应用系统的不断复杂化，传统的访问控制模型越来越不利于实现系统的统一和全局访问控制。于是，在20世纪90年代初，由美国国家标准和技术研究院提出了一种新的访问控制模型——基于角色的访问控制(Role-Based Access Control，RBAC)。

RBAC[3,4]的主要思想是将授权和角色联系在一起，可以分配角色到用户，使用户通过角色间接地访问资源。随着新的应用和系统的增加，角色可以分配更多的权限，也可以根据需要撤销相应的权限。其最大优点在于它能够灵活表达和实现组织的安全策略，使管理员从访问控制底层的具体实现机制中脱离出来，十分接近日常的组织管理规则。但另一方面，安全管理员除了需要维护用户和角色的关联关系外，还需维护信息资源访问权限和角色的映射。因此在信息量巨大、信息资源变化频繁的应用系统中，如何有效地减轻由于信息资源单点派生、演化和重组等带来的重新分配、设定角色权限的繁重工作量是系统开发中的一个关键问题。

1 基于抽象角色的访问控制模型

针对上面的问题，RBAC模型本身并没有给出相应的解决方法。基于此，本文提出了基于RBAC的新模型——ARBAC模型。

1.1 ARBAC模型相关术语的定义

定义1 实体(Entity，E)：一个计算机资源(物理设备、数据文件、内存或进程)、合法程序或用户。由实体构成的集合称为实体集，简称为EG＝{E_i|1=<i<=n}。

定义2 主体[4](Subjective，S)：一个可以对其它实体施加动作的主动实体。主体可以是用户或其它任何代理用户行为的实体(比如进程、作业或程序)。

定义3 客体[4](Object，O)：一个接受其它实体动作的被动实体。客体可以是一个可识别的资源。一个实体可以在某一时刻是主体，而在另一时刻是客体，这取决于该实体扮演的是动作执行者还是被执行者。

定义4 用户(User，U)：一个被授权使用计算机的人员。由用户构成的集合称为用户组，简称UG＝{U_i|1=<i<=n}。

定义5 操作(Operation，OP)：完成任务时施加在客体上的动作。由一连串操作组成的集合称为操作集OPG＝{OP_i|1=<i<=n}。

定义6 资源(Resource，R)：当执行一项功能时所使用或消耗的信息或实体。由一系列资源构成的集合称为资源集RG＝{R_i|1=<i<=n}。

定义7 许可(Permission，P)：表示对系统中的客体进行特定模式访问的操作许可。由许可组成的集合称为许可集PG＝{P_i|1=<i<=n}。

定义8 抽象角色(Abstract Role，AR)：是指可以具有一定操作的命名，它代表了一定的资格、权利和责任。由抽象角色组成的集合称为抽象角色集ARG＝{AR_i|1=<i<=n }。

定义9 具体角色(Concrete Role，CR)：是指可以具有一定许可

基金项目：国家科技基础条件平台基金资助项目(2003DKA5G015)
作者简介：杨文波(1980－)，男，硕士生，主研方向：计算机网络与应用；张 辉、刘 瑞，副教授
收稿日期：2006-01-23 **E-mail：**yangwb@amazon.com

的命名，同样也代表了一定的资格、权利和责任。由具体角色组成的集合称为具体角色集

CRG＝{CR_i|1=<i<=n }。

定义 10 操作分配(Operation Assignment，OA)：OA ⊆ OPG×ARG 是一个多对多的操作到抽象角色的授权关系。

定义 11 许可分配(Permission Assignment，PA)：PA ⊆ PG×ARG 是一个多对多的许可到抽象角色的授权关系。

定义 12 用户分配[1](User Assignment，UA)：UA ⊆ UG×CRG 是一个多对多的用户或用户组到具体角色的授权关系。对一个用户组完成用户分配后，该组成员将自动获得其所在组的所有授权。

定理 1 PG＝OPG×RG，即许可集等价于操作集和资源集的笛卡尔积，其中如果序偶<OPi,Rj> OPi ∈ OPG, Rj ∈ RG, 1≤i,j≤n 不存在，则在 PG 中记<OPi,Rj>为 NULL。对某一特定的非空许可 P 记作：P＝OP@R。

证明：由定义(5)和定义(7)得知：操作是指完成任务时施加在客体上的动作，与资源毫无关联，是一个抽象的、纯粹的操作。但许可则不同，它表示对系统中的客体进行特定模式访问的操作许可，是某一特定的施加在资源上的操作。可见许可等价于和资源关联在一起的操作，因而许可集也就等价于操作集和资源集的笛卡尔积。

推论 1 CRG＝ARG×RG，即具体角色集等价于抽象角色集和资源集的笛卡尔积，其中如果序偶<ARi, Rj> ARi ∈ ARG, Rj ∈ RG, 1≤i, j≤n 不存在，则在 CRG 中记<ARi, Rj>为 NULL。对某一特定的非空具体角色 CR 可记作：CR＝AR@R。

证明：由定理 1、定义(8)和定义(9)容易得出推论 1 的结论。

推论 2 PA＝OA×RG，即许可分配等价于和资源集关联在一起的操作分配。

证明：由定理 1、定义(10)和定义(11)可得出推论 2 的结论。

1.2 ARBAC 模型中角色间的关系

新模型中角色间的关系主要涉及到角色的继承和阻塞。

(1)继承关系：包括抽象角色间的继承和具体角色间的继承，这种继承关系满足自反、传递和反对称关系，因此用偏序关系≥来表示。抽象角色间的继承关系记为 ARH，ARH ⊆ ARG × ARG 是关于 ARG 的偏序关系。$AR_i \geq AR_j$, $AR_i \in ARG$，$AR_j \in ARG$，1≤i, j≤n表示ARi继承ARj，ARi包括AR_j的所有操作之外，还可以拥有自己专门的操作。具体角色间的继承记为CRH，这种继承针对存在包含关系的不同资源之间对同一抽象角色的继承，CRH ⊆ CRG × CRG＝(ARG×RG)×(ARG×RG)，$CR_i \geq CR_j$且$R_i \subseteq R_j$，其中CR_i＝AR@Ri，CR_j＝AR@R_j，$CR_i \in CRG$，$CR_j \in CRG$，AR ∈ ARG，1≤i, j≤n表示CR_i继承CR_j。具体角色间继承的一个典型例子就是在以Portlet为核心的门户应用系统中，一个页面可以包含多个Portlet，其中每个Portlet和页面间的关系满足包含关系，因而就可以使用具体角色间的继承关系。

(2)阻塞继承包括对抽象角色间继承关系的阻塞和对具体角色间继承关系的阻塞，分别用ARUB和CRDB来表示。前者可以指定命题：$AR_i \geq AR_j$，$AR_i \in ARG$，$AR_j \in ARG$，1≤i, j≤n不成立，从而可以阻止 2 个抽象角色间的继承关系。同样CRDB可以阻止 2 个具体角色间继承关系的产生。

1.3 ARBAC 模型中的约束关系

一个系统中的操作类型相对于系统资源和访问者的变更来说是相对稳定的，在 ARBAC 模型中，根据这一特点并利用角色间的关系可以由操作集构造抽象角色集，模型对此提出了第一约束条件。

定义 13 从操作集到抽象角色集的构造关系记为$R_{OP} \rightarrow_{AR}$(OPG, ARG)，该关系的输入为操作集OPG，输出为抽象角色集ARG。对于任意一个$AR_i \in ARG$,1≤i≤n其所拥有的操作表示为$OP(AR_i)$。

约束 1 对于$R_{OP} \rightarrow_{AR}$(OPG, ARG)，ARBAC模型要求如果构造关系$R_{OP} \rightarrow_{AR}$的输入OPG一旦确定，那么其输出ARG也必须确定；|ARG|>=|OPG|，$\bigcup_{i=1}^{m} OP(AR_i)$＝OPG，其中|s|表示集合s所包含的元素个数，m=|ARG|。

约束 1 中|ARG|≥|OPG|且$\bigcup_{i=1}^{m} OP(AR_i)$＝OPG能够防止构造关系$R_{OP} \rightarrow_{AR}$在产生抽象角色集的过程中发生操作遗漏现象。对于一个应用系统而言，其操作集一般是相对固定的，根据约束 1 的要求，通过构造关系$R_{OP} \rightarrow_{AR}$得出的抽象角色集也是相对固定的，而且产生的抽象角色集所包含的抽象角色数其上限不会超过$(2^{|OPG|}-1)$，而且一般往往小于这个数字，否则就会形成抽象角色的冗余；同时由推论 1 可知具体角色集相对于资源集也是相对固定的，这点是与RBAC模型之最大不同。在RBAC模型中，角色可以由系统管理员根据需要随时创建，在理论上角色的创建可以是无限制的。

约束 2 命题∃UG(Ui∈UG)(1=<i<=n)的真值为真，而且一般情况下要求|UG|>1。

约束 2 表明对于任意一个用户存在一个所属用户组，当然这条约束并不限制一个用户可以处在多个用户组。一般情况下，约束 2 要求用户组的成员人数至少有 2 个，同时也不排斥某个用户组只有一个成员的特殊情况。约束 2 的核心思想就是尽量将一个用户归在用户组之中。

约束 3 在进行用户分配时，用户组具有优先授予权，只有当组内的某个用户具有特殊许可要求时才对该用户进行单独的用户分配。

约束 2 要求尽量将一个用户归在用户组之中，同时根据定义(12)可知在对一个用户组进行用户分配时，该组成员将自动进行用户分配。显然，在进行用户分配时以用户组优先能够大大减少系统的管理开销，节省代价。当然，用户组的一次授权不能满足其中个别用户的许可要求时可以对其单独进行用户分配。这也正是约束 3 提出的主要目的。

1.4 ARBAC 模型中的授权关系

ARBAC 模型的授权关系主要涉及以下 4 大要素：用户组或用户(UG/U)，抽象角色集(ARG)，具体角色集(CRG)，资源集(RG)。其关系如图 1 所示。

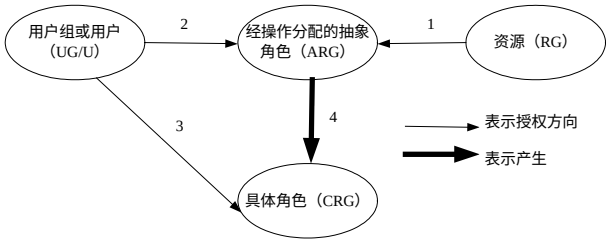


图 1 ARBAC 授权关系

图 1 中的授权关系主要存在 2 条授权路径：1－>2 和 1－>4－>3。前者表示首先经过第 1 步将资源和抽象角色关联在一起，然后经第 2 步将用户(组)和经操作分配的抽象角色

关联在一起，此时用户(组)就可以获得对资源的许可权了。第 2 条授权路径经第 1 步后，再经第 4 步产生具体角色，然后通过第 3 步用户分配将用户(组)和具体角色关联在一起，同样用户(组)也可以获取对资源的许可权。从表面上看似乎是 2 种授权路径，但本质上这 2 条授权路径是等价的。

2 ARBAC 模型分析

ARBAC 与 RBAC 模型最大的不同在于∶ARBAC 提出了抽象角色和具体角色的概念，对操作分配和许可分配作了区分，针对模型给出了具体的约束条件，使模型中的抽象角色相对于操作类型而言固定化；但是 RBAC 模型中的角色则不同，角色可以由系统安全管理员来创建，从理论上讲，角色的创建是无限制的。RBAC 模型中角色创建的任意性使得在信息量巨大、信息资源变化频繁的应用系统中，维护大量的信息资源访问权限和角色之间的映射成为安全管理员的一项繁重工作。而 ARBAC 模型能够在很大程度上减轻安全管理员的繁重工作量，是解决该问题的一个很好方法。下面给出证明过程：

假设：|OPG|=m, |RG|=n, |ARG|=m+δ, m、n、δ均为一正整数。

证明：|CRG|<<|R_0G|,其中|R_0G|表示RBAC模型中可能的角色数量。

证：∵CRG=ARG×RG(推论 1)

∴|CRG|=|ARG|×|RG|=(m+δ)n

令 m×n＝k，则 k 表示 RBAC 模型中可能的资源访问权限的数量。

∵RBAC 模型的定义(见参考文献[1])

$\therefore |R_0G| = \sum_{i=1}^{k} C_k^i$

因为 $(x+y)^k = \sum_{i=0}^{k} C_k^i x^i y^{(k-i)}$，令x＝y＝1，

则 $2^k = \sum_{i=0}^{k} C_k^i$

又∵ $\sum_{i=0}^{k} C_k^i = 1 + \sum_{i=0}^{k} C_k^i$

$\therefore |R_0G| = \sum_{i=0}^{k} C_k^i = 2^k - 1 = 2^{mn} - 1$

(1)不妨令 m=5，n＝20，δ＝4，则

$|CRG|/|R_0G| = (m+\delta)n/2^{mn} - 1 = (5+4) \times 20/2^{2 \times 50} = 180/2^{100} \approx 0$

(2)令 $n \to +\infty$，则

$\lim_{n\to+\infty} (|CRG|/|R_0G|) = \lim_{n\to+\infty} (m+\delta)n/(2^{mn} - 1) = (m+\delta) \lim_{n\to+\infty} n/(2^{mn} - 1) = 0$

综上所证：|CRG|<<|R_0G| 证毕。

由上面的证明可以看出，当一个系统的信息资源数量不断增大时，RBAC 模型可能产生的角色以及需要维护的资源访问权限和角色间的映射与 ARBAC 模型相比会迅速增大，远远超过了 ARBAC 模型可能产生的具体角色和需要维护的资源访问权限和角色间映射的数量。另一方面，为了避免因抽象角色的相对固定性而导致 ARBAC 模型在灵活性方面的明显损失，模型通过定义角色间的继承和阻塞关系使得抽象角色和具体角色具有一定的可定制性，尤其是在具有包含关系的信息资源所在的应用系统中。

3 总结

本文提出的 ARBAC 新访问控制模型已经在国家重点建设项目——国家科技基础条件平台应用系统中实现并投入运行，试验结果表明：在信息量巨大、信息资源变化频繁的应用系统中，ARBAC 模型相比于 RBAC 模型付出的安全维护代价是相对比较低的，取得的安全控制效果是良好的，有效克服了 RBAC 模型在该类系统中所表现出的不足。

参考文献

1 Sandhu R S, Coyne E J, Feinstein H L, et al. Role-based Access Control Models[J]. IEEE Computer, 1996, 29(2): 38-47.

2 Joshi J B D, Bertino E, Latif U, et al. A Generalized Temporal Role-based Access Control Model[J]. IEEE Trans. on Knowledge and Data Engineering, 2005, 17(1): 4-23.

3 Duan Haixin, Wu Jianping, Li Xing. Policy-based Access Control Framework for Large Networks[C]. Proceedings of IEEE International Conference on Networks, 2000-09-05: 267-272.

4 Moyer M J, Abamad M. Generalized Role-based Access Control[C]. Proc. of the 21st International Conference on Distributed Computing Systems, 2001: 391-398.

（上接第 161 页）

{MessageBox("该考生可能为非法考生!", "系统提示", MB_ICONINFORMATION | MB_OK);} }

5 结束语

本文设计的基于指纹特征的考生身份认证系统，可在考试现场采集考生指纹并根据考生提供的考号进行 1:1 匹配，对考生身份进行验证，能保证报名者与考生身份完全一致。整个系统软件开发采用 Visual C++6.0 设计开发，运行速度快，以 500 位考生做测试，平均每位考生现场验证身份时间为 4.67s。由于考号（一般为准考证号）的输入采用小键盘，其输入速度取决于操作者的熟练程度，若能将考号以条形码的形式制作在准考证上，利用条形码识别器直接读入考生号码，将能进一步提高系统应答速度，并能减少出错的可能性。

随着指纹识别产品的不断开发，指纹识别产品价格不断下降，运用指纹识别技术进行考生身份认证是今后考试管理发展的必然趋势。利用指纹识别技术鉴别考生的真实身份，可以有效地杜绝替考现象，从而保证考试的公平、公正。

参考文献

1 傅德胜，王水平，李仕强. 指纹中心自动定位诸方法研究及应用[J]. 计算机应用与软件, 2004, 21(3): 10-13.

2 米晓萍，曹 富. 指纹识别技术在考试管理系统中的应用[J]. 科技情报开发与经济, 2004, 14(6): 206-207.

3 于 焱. 指纹识别技术在考试管理信息系统应用的研究[J]. 长春大学学报, 2003, 13(1): 61-63.

4 王水平，傅德胜. 基于指纹特征的网络身份认证系统[J]. 信息安全与保密, 2005, 1(4): 23-26.