第33卷 第19期 计 算 机 工 程 2007年10月
Vol.33 No.19 Computer Engineering October 2007
·安全技术· 文章编号：1000—3428(2007)19—0147—03 文献标识码：A 中图分类号：TP311

RBAC和FBAC的适用条件与集成

潘德锋[1]，彭 霞[2]，吴信才[3]
（1. 华北电力大学计算机科学系，保定 071000；2. 北京大学遥感与地理信息系统研究所，北京 100871；
3. 中国地质大学信息工程学院，武汉 430074）

摘 要：用集合论的方法分析了模型选取的4个决定因素：需要授权的用户数量，功能权限集的基数，角色的权限变化情况，用户的角色变化情况。对RBAC和FBAC的适用情况进行了划分。论证了在复杂的大型系统中，综合采用多种访问控制模型，对权限进行分割合并，区分出公共权限和专门权限，并引入多级授权机制，才能够克服单一模型的不足。

关键词：存取控制；RBAC；FBAC

Suitable Conditions and Integration of RBAC and FBAC

PAN De-feng[1], PENG Xia[2], WU Xin-cai[3]
(1. Department of Computer Science, North China Electric Power University, Baoding 071000; 2. Institute of Remote Sensing and GIS, Peking University, Beijing 100871; 3. School of Information Engineering, China University of Geoscience, Wuhan 430074)

【Abstract】 This paper analyzes the 4 factors that determine which model should be chosen with the method of set: the quantity of users, the radix of function set, the change of role's permissions, the change of user's roles. It partitions the different conditions that are suitable for RBAC(role-based access control) model or FBAC(function-based access control) model. This paper demonstrates that it is necessary and feasible to adopt different models in complex large system. In this kind of system, it can deal with two methods: cut apart and combine the permissions; use the mechanism of multistage assignment.

【Key words】 access control(AC); role-based access control(RBAC); function-based access control(FBAC)

RBAC 模型被广泛接受并越来越受到重视。RBAC 的存取控制策略是，将访问权限封装在角色中，用户通过被赋予的角色来访问数据资源。FBAC 是另外一种实用的模型。其基本原理是，在对应实际权限的基础上，把执行某项功能的相关操作组合成一种权限，然后对用户授权。它将访问控制策略保存在矩阵中，矩阵的每个元素表示一个用户对一种功能的权限。任何直接或间接的访问都要进行权限验证。那么，RBAC 和 FBAC 各有什么特点，如何选取一种最优的访问控制模型，二者有无结合的必要性和可能性？

1 选取 RBAC 或 FBAC 的决定因素

将需要指定权限的功能组建成一个功能权限集FS，假设它的基数|FS| = n。把操作权限子集组成的集合定义为角色集RS。在RBAC中，RS的角色就是联系权限和用户的中介[1]。当RS的每个角色只包含 1 种权限，并且 |RS| = n时，RBAC模型就演变为间接的FBAC模型。

通过对各种存取控制模型的特征分析，可以归纳出具体选择哪种模型，取决于 4 个方面的因素：(1)需要授权的用户数量；(2)功能权限集的基数|FS|；(3)RS 中角色的权限变化情况，即角色集合中 10%角色(角色普查限 LR)的权限发生变化时，所经历的时间；(4)授权用户的角色变化情况，即授权用户中 10%用户(用户普查限 LU)的角色发生变化时，所经历的时间。

下面分析各种因素不同情况下的授权管理工作量。在图 1 中，y 轴为管理工作量，x 轴为功能权限集的基数|FS|，u 为授权用户数量。

假设功能权限集的基数|FS| = x，则它的幂集的基数|ρ(FS)| =2^x。只要|FS|稍大一点，|ρ(FS)|也会增长为一个天文数字。实际需要的操作权限子集并没有这么多，即RS⊆ρ(FS)，但权限规划要考虑最极端的可能，角色集必须是完全的，RS=ρ(FS)。假设每个用户只需一次授权，那么RBAC对2^x个角色进行x项功能的授权，再把角色授予u个用户，每个角色的授权操作平均值为x /2。则初始授权管理工作量，RBAC的$y1=2^x \cdot x/2+u$，FBAC的$y2=u \cdot x/2$。

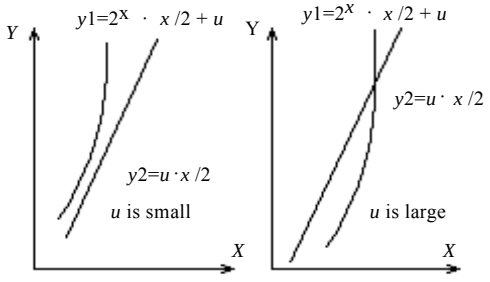

图 1 初始授权工作量

由图 1 可见，$y2$ 为线性函数，而 $y1$ 为指数函数与线性函数的积，所以在 u 较小的情况下，当 x 增加时，$y1$ 的增加要远远快于 $y2$。只有 $x<<u$，权限集的基数远小于用户数量时，

基金项目：国家“863”计划基金资助项目(2001AA135170)
作者简介：潘德锋(1978－)，男，讲师、博士，主研方向：MIS，AI；彭 霞，硕士；吴信才，博士、博士生导师
收稿日期：2006-11-16 **E-mail：**padefeng@etang.com

RBAC 的初始授权工作量才会小于 FBAC 的。

再来看对新增的用户授权或调整已有用户的权限。RBAC在计算授权工作量时，不能忽略检索角色的工作量。查看角色集的每一个元素，与一次授权的工作量相当。根据统计平均，要遍历ρ(FS)的一半才能找到合适的角色，再加上 1 次授予角色，RBAC的授权管理工作量$y1=2^{x}/2+1$。对于直接授权，要遍历功能权限集FS，根据统计平均，还要把FS的一半元素所代表的权限授予用户，FBAC的授权管理工作量$y2=x+x/2$。由图 2 可见，由于$y2$为固定斜率的线性函数，而$y1$为指数函数，当x增加时，$y1$的增加要远远快　　于$y2$。

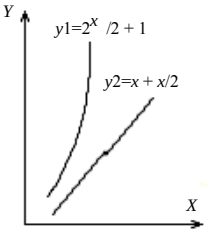

图 2　单个用户的权限授予或调整工作量

在RBAC模型中，为了系统安全，必须增加一些额外开销。从上一次所有角色的权限审核无误开始，当角色集合中累计有 10%角色的权限发生变化时，要重新对所有角色的权限进行普查。这里的角色普查限LR和下面的用户普查限LU的经验值都取 10%，在对用户的权限要求非常严格的场合，取值应更小一些。对于RBAC模型，审核单个角色的工作量为y＝x，审核所有角色的工作量为$y'=2^{x}\cdot x$，见图 3(a)。随着x增大，角色权限普查的工作量剧增。

从上一次所有用户的角色审核无误开始，当累计有 10%用户的角色发生了变化时，要重新对所有用户的角色进行普查。对于单个用户来讲，如果用户角色无误，核对的工作量为 1；如果用户角色错误，其工作量为调整已有用户角色的工作量，即查找合适角色的工作量，外加 1 次授权，工作量$=2^{x}/2+1$。用户角色普查的总工作量介于u和$(2^{x}/2+1)\cdot u$之间，见图 3(b)。

对 FBAC 模型，从上一次所有用户的权限审核无误开始，当累计有 10%用户的权限发生变化时，也要对所有用户的权限进行普查。对于单个用户来讲，如果用户权限无误，核对的工作量为 x；如果用户权限错误，其工作量为调整已有用户权限的工作量，工作量＝$x+x/2$。用户权限普查的总工作量介于 ux 和 $3ux/2$ 之间，见图 3(c)。

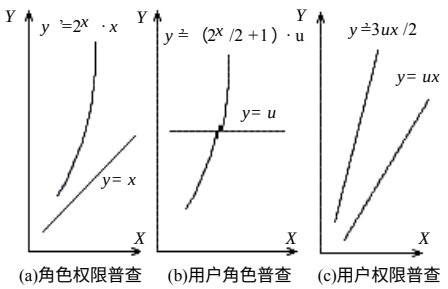

图 3　各类普查的工作量曲线

对上述 4 个决定因素各分 3 种不同情况进行组合，得到一个包含了 81 种情况的表格(见表 1)。表 1 中，S 列表示编号；U 列用 0、1、2 表示授权用户数量的少、中等、多；|FS| 列用 0、1、2 表示 |FS| 的小、中等、大；CR 列用 0、1、2 表示角色权限变化的小、中等、频繁；CU 列用 0、1、2 表示用户角色变化的小、中等、频繁；Su 列表示最适用的模型，列值 R 表示 RBAC，列值 F 表示 FBAC。

表 1　不同情况下最适宜的授权模型

N	U	S	CR	CU	Su	N	U	S	CR	CU	Su
1	0	0	0	0	RF	55	2	0	0	0	R
…	…	…	…	…	RF	56	2	0	0	1	R
9	0	0	2	2	RF	57	2	0	0	2	(R)
10	0	1	0	0	F	58	2	0	1	0	R
…	…	…	…	…	F	59	2	0	1	1	(R)
27	0	2	2	2	F	60	2	0	1	2	(R)
28	1	0	0	0	RF	61	2	0	2	0	R
…	…	…	…	…	RF	62	2	0	2	1	(R)
34	1	0	2	0	RF	63	2	0	2	2	(R)
35	1	0	2	1	F	64	2	1	0	0	(R)
36	1	0	2	2	F	65	2	1	0	1	C
37	1	1	0	0	RF	66	2	1	0	2	C
38	1	1	0	1	F	67	2	1	1	0	B
…	…	…	…	…	F	…	…	…	…	…	B
46	1	2	0	0	F	72	2	1	2	2	B
47	1	2	0	1	(F)	73	2	2	0	0	A
…	…	…	…	…	(F)	…	…	…	…	…	A
54	1	2	2	2	(F)	81	2	2	2	2	A

模型选取规则是，用户数量很大时，FBAC 的访问控制矩阵成几何级数增长，只能采用 RBAC；|FS| 较大或 RS 中各角色的权限变化频繁时，RBAC 的角色管理变得非常烦琐，容易出现安全漏洞，只能采用 FBAC；上述 4 个因素都不等于 2 时，两种模型的相对优势并不明显，理论上采用任何一种都可以，但实际应用中并不推荐(表 1 中用括号注明)。表 1 中 A、B 和 C 的含义，请参照本文第 3 节单一访问控制模型不足的 3 种典型情况。

下面是几个应用例子。文件共享管理中，对每个文件的每种操作，都是一种权限，但是共享是对整个文件夹进行的操作，|FS| 并不大，属于情况 1～9，用RBAC或FBAC都可以[4]。在数据库对象访问中，用户数量不多，|FS|不大，各角色具备的权限比较固定，属于情况 1～9，用RBAC或FBAC都可以。例如Oracle提供了 90 多种可选的系统级权限，它采用的是RBAC模型，DBA先创建角色，然后把一些权限授予角色，最后再把这些角色授予用户[5]。在数据记录访问中，访问者数量通常很多，但是一般都是对有限的几个表进行操作，|FS| 很小，各角色的权限在设计时已经约定好，比较固定，属于情况 55～61，适合用RBAC模型。

2　RBAC 的局限性

在适用条件下，RBAC 的权限规划清晰，权限管理直观，安全性好。但是它并非适用于所有情况，在某些条件下的局限性也日益明显。

(1)|FS| 只能很小[2]。RBAC把对用户的授权分成两部分，以增加对角色的授权管理，来减少对用户的授权管理。但是当 |FS| 较大时，这一替换就得不偿失了。角色增多也加重了检索负担。

(2)授权灵活性差。权限跟角色之间、角色跟用户之间都是多对多关系。中介授权隐含的问题是：一个用户本身需要的权限构成一个集合，当他被被赋予多个角色后，多个角色与权限关联构成另一个集合，保证两个集合相等很不容易[1]。

(3)角色的权限变化容易造成安全漏洞。在修改角色的权限时，很容易发生这样的情况，在减少角色权限后，拥有该角色的用户可能丧失了应有的权限；在增加角色权限后，拥有该角色的用户可能被赋予了额外的权限，导致越权操作，这无疑降低了系统的安全性[1]。

(4)对于用户规模不大的系统，RBAC比直接授权还繁琐[2]。尤其是角色的权限和用户的角色不稳定时，要不断地创建新角色，撤销旧角色，给用户重新授权。管理角色的负担远远超出管理访问控制矩阵的负担。

3　RBAC 与 FBAC 的集成

3.1 单一模型的不足和解决方法

上面列举了单一访问控制模型的典型。然而对于下列情形，单纯地采用任何一种模型，都无法满足需要。

(1)需要授权的用户很多，|FS| 很大。如表 1 中 73～81。无论角色的权限变化、用户的角色变化是否频繁，管理的负担都无法承受。

(2)需要授权的用户很多，|FS| 中等，角色的权限变化中等或频繁。诸如表 1 中 67～72。由于角色权限变化的可扩散性，极易出现安全漏洞。

(3)需要授权的用户很多，|FS| 中等，用户的角色变化中等或频繁。诸如表 1 中 65、66。由于需要进行角色替换的用户很多，管理的负担无法承受。

在大型的MIS、OA或电子政务等多用户的网络软件开发中，信息访问控制时经常会面临上述问题，即 |FS| 很大，而且角色界限非常模糊。在这些系统中，模型选取时须通盘考虑。对上述 3 类情况，一般用下面 2 种方法来处理(通常同时使用)[3]：

(1)分割合并 RS 中的各角色的权限，区分出公共权限和专门权限，公共权限的管理采用 RBAC，专门权限的管理采用 FBAC。RBAC 最擅长的是用户数量较大，|FS| 较小，角色的权限和用户的角色都很稳定的情况；FBAC 最擅长的是用户数量较少，角色的权限和用户的角色经常变化的情况。

(2)引入多级授权机制，将权限框架分解为多个相对独立的子系统，降低 |FS| 的最大值[4]。以OA系统为例，特定的功能只与特定部门的少数用户有关，在这类软件的开发中，总是依照实际部门将用户分组，而不是构造一个包含了所有用户和所有权限的访问控制矩阵。大型系统包含的用户和操作，被多个部门分割后，最终考虑的是中等或少量用户在|FS| 不大的情况下的授权。

3.2 RBAC 与 FBAC 集成实例

以国外某商业银行的OA为例，它管理员工 1 420 人，结算中心 11 个，储蓄所 83 个，结算储蓄业务 67 种。其特点是：用户数量大，人员角色不稳定，权限管理的权责要求严格。原先采用单一的RBAC模型，每年由于权限管理漏洞，错账金额高达数百万元。改用集成模式后，错账金额大幅度下降，而且能够迅速准确地找出过失原因。见表 2 [5]。

表 2 银行 OA 的权限管理模型改变前后的明细对照(部分)

明细对照	RBAC	集成	集成的优势
权限分配员人数	1	11 + 83	管理分散，不易出错
管理员管理权限	67	0	不涉及业务授权
授权业务种类	67	9 + 58	只对专门业务授权
地方主管管理用户	—	平均 20	主管对他管辖的业务员
地方主管管理权限	—	平均 10	
账目不平衡笔数	18	2	可以追踪的责任人
账目不平衡金额	270 万	21 万	更容易追踪明细

集成模式特点是：权限管理员只负责初级权限的授予和修改，例如让某个用户属于或主管某个结算中心或储蓄所。二级权限的授予和修改则由地方主管负责。这不仅减轻了权限管理员的负担，而且更符合业务运行的现实原型，降低了出错可能。

RBAC 在适用条件下，通过角色中介降低了权限管理负担；FBAC 在适用条件下，将权限直接指派给用户，提高了管理的灵活性，增强了系统的安全性。在复杂的多用户的大型系统中，采用多种方法，让 RBAC 和 FBAC 两种模型相互补充，是一种非常有效的访问控制策略。它不仅简化了用户授权管理，降低了权限管理的负担和代价，而且减少了程序开发与维护的负担和代价。

参考文献

1 Covington M J. Generalized Role-based Access Control for Securing Future Applications[C]//Proc. of National Information Systems Security Conference. 2000: 187-196.

2 Tidswell J, Potter J. Adynamically Typed Access Control Model[C]//Proc. of the 3rd Australian Conference on Information Security and Privacy. 1998-07: 308-319.

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

4 Khayat E J, Abdallah A E. A Formal Model for Flat role-based Access Control[J]. Computer Systems and Applications, 2003, 44(3): 75-87.

5 Sandhu R, Bhamidipati V. An Oracle Implementation of the PRA97 Model for Permission-role Assignment[C]//Proc. of the 3rd ACM Workshop on RBAC. 1998: 13-21.

（上接第 133 页）

表 1 常见各种攻击下水印图像的性能测试结果

攻击方式	参数	本文方法	文献[1]方法
		NC	*NC*
JPEG 压缩	质量因子 50	0.999	0.965
	质量因子 30	0.949	0.751
叠加噪声	椒盐噪声(0.02)	0.925	0.885
	乘性噪声(0.02)	0.962	0.931
	均匀分布(5%)	0.996	0.943
	高斯分布(5%)	0.961	0.813
平滑滤波	高斯滤波[7,7]	0.998	0.993
	中值滤波[3,3]	0.961	0.732
图像增强	均衡化处理	0.999	0.985
	锐化 2 次	0.996	0.991
	对比度+50%	0.970	0.916
几何攻击	缩放 4 倍	0.994	0.883
	剪切 1/2	0.865	0.792

该算法对这一系列图像处理具有很好的鲁棒性，其性能优于利用“多小波域子块之间的系数相关性”的水印嵌入算法[1]。

4 结束语

本文利用支持向量机的学习优势，建立了小波域中小波系数方向树结构的SVM模型，根据这种关系模型来嵌入和提取水印，获得了很好的效果。与基于小波零树和重要系数树的小波域算法[2~3]相比，该算法不仅水印容量大、算法简单，而且水印的鲁棒性良好；与基于神经网络的多小波域水印算法[1]相比，该算法具有很好的理论基础。

参考文献

1 Zhang J, Wang N C, Xiong F. A Novel Watermarking for Images Using Neural Networks[C]//Proc. of International Conference on Machine Learning and Cybernetics. 2002: 1405-408.

2 Inoue H, Miyazaki A. A Digital Watermark Based on the Wavelet Transform and Its Robustness on Image Compression[C]//Proc. of International Conference on Image Processing. 1998: 391-395.

3 Hsieh M S, Tseng D C, Huang Y H. Hiding Digital Watermarks Using Multiresolution Wavelet Transform[J]. IEEE Transactions on Industrial Electronics, 2001, 48(5): 875-882.

4 Li C H, Lu Z D. SVR-parameters Selection for Image Watermarking[C]//Proc. of the 17th IEEE International Conference on Tools with Artificial Intelligence. 2005: 466-470.