基于资源抽象的 RBAC 模型的研究及应用

Research and Application on RBAC models Base on Resource Abstract

周 维 郑金华 许海霞 （湘潭大学信息工程学院 湖南湘潭 411105）

摘要：本文提出了一种新的基于资源抽象的角色访问控制模型（RARBAC）：它在对角色授权以及实现系统安全策略的过程中，通过对资源的二次抽象，有效的减少冗余角色，降低管理复杂度；并在高校管理系统的实例中验证了本方案。与经典的 RBAC 相比，RARBAC 能自由控制权限粒度，具有更好的适应性和安全性。

关键词：RBAC 权限粒度 资源抽象 角色授权 资源区域

1 引言

1992 年 David Ferraiolo 等人提出的基于角色访问控制模型 RBAC[1]，是近年来在信息安全领域访问控制方面的研究重点。RBAC 的特点就是引入了角色的概念，简化了权限的管理。根据控制对象的粗细程度，访问控制可分为粗粒度和细粒度两种。通常粗粒度表示类（model）别级，即仅考虑对象的类别，不考虑对象的某个特定的实例。而细粒度表示实例（instance）级别，即需要考虑具体对象的实例。细粒度是在考虑粗粒度的对象类别之后才再考虑特定实例。

目前多数基于 RBAC 授权管理模型的研究尚不完善，关于权限粒度矛盾问题仍然没有解决。即权限粒度越小，角色数量越多，角色数量的增长与权限粒度的细化成指数增长。

从 RBAC 模型的形式化描述中可知角色实质是对主体权限的一种归类与抽象，但经典 RBAC 只针对主体的某一个属性（主要是操作）对主体进行了一次抽象，抽象度不高，导致在角色创建时候考虑多属性就会导致大量的角色数量，造成角色的冗余。因此，本文提出基于资源抽象的角色访问控制模型——RARBAC 模型，该模型在对角色授权以及实现系统安全策略的过程中，利用资源二次抽象，较好的克服了角色冗余的问题，减少了角色规模，降低角色管理的复杂性，能更好适应多级复杂权限系统的要求。

2 RBAC 模型

RBAC 与 DAC、MAC 称为三大访问控制策略[7]。相比较而言，RBAC 是实施面向企业的安全策略的一种有效的访问控制方式，其具有灵活性、方便性和安全性的特点，目前在大型数据库系统的权限管理中得到普遍应用。

RBAC 的基本思想是：用户被赋予角色，而权限不直接赋予用户而是赋予角色。用户通过担任某些角色而获得访问权限这样就能极大地简化权限管理，减少管理访问控制策略的开销，并且易于描述和理解。

在 RBAC96 模型中，角色和授权是多对多的关系，即对每个角色设置了多个授权关系，同时一个授权也可以赋予多个角色。RBAC96 规定：每个角色至少具备一个授权，而每个用户至少扮演一个角色。

授权机制通过特定的操作，如读、写、更新和执行等将角色和用户联结起来。通过授权管理机制，可以给予一个角色多个授权，而一个授权也可以赋予多个角色。相对于将用户和授权之间直接关联的方法，RBAC 授权机制用更加简单的方法向最终用户提供语义更加丰富和完整控制的存取功能[8,9]。

3 基于资源抽象的角色访问控制模型

3.1 资源抽象概念的提出

经典 RBAC 的抽象度不高，导致在权限分配与使用时候造成冗余，无疑增加了角色管理的复杂度。上节定义中说明，主体的权限是操作与资源的笛卡尔积，经典 RBAC 通常是对操作进行抽象来进行角色划分。我们可以在经典 RBAC 操作一次抽象的基础上对资源进行二次抽象，建立一种基于资源抽象的角色访问控

制模型——RARBAC(Resource Abstract – based RBAC)模型。

在该模型权限分配中,首先对操作进行抽象,提取出与职务联系的不同操作子集,即角色。然后对资源进行抽象,即根据实际需求以及相应规则或约束对资源进行类划分,使得每一类资源对应不同的区域。最后指定每一个角色一个资源的类,同时指定每个用户拥有的每个角色一个具体的资源的区域。RARBAC 通资源分解,对角色与用户同时授权,使得权限抽象度大大提高,冗余角色将大幅减少。模型示意图如图 1 所示。

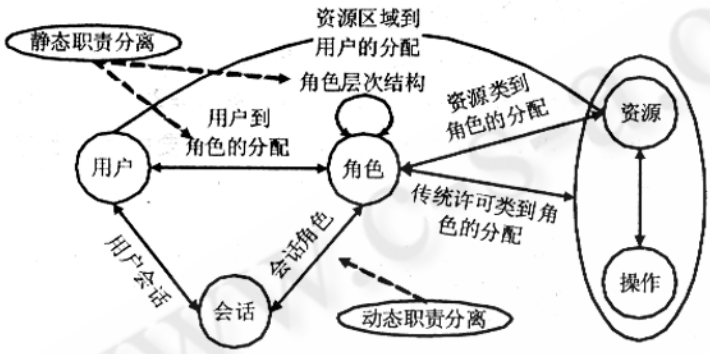

图 1　基于资源抽象的 RBAC 模型

3.2 RARBAC 模型追加定义

资源类(Resource Class:RC):是指对对象 OBJ 进行不同层次划分时的资源子集,代表一定类型的资源。由资源类组成的集合称为资源类集合 $RCG=\{RC_i|1\leqslant i\leqslant n\}$;

资源区域(Resource District:RD):是指以 RC 为标准,对对象 OBJ 进行划分之后所形成的区域集合。由资源区域组成的集合称为资源区域集合 $RDG=\{RD_i|1\leqslant i\leqslant n\}$;

由定义可知,不同的 RC 对应不同的 RD,且 $OBJ=RC_i\times RD_i(1<=i<=n)$,则:

$RA=RC\times R$, RA 是资源类到角色的多对多关系;

$DA=RD\times U$, DA 是资源区域到用户的多对多关系;

$operations:R\times RC_i\times RD_i\rightarrow 2^{OP}$,一个角色映射到客体对象的多个操作集合;

$object:P\rightarrow 2^{RC_i\times RD_i}$,一个权限映射到一组对象集合;

$PA\subseteq P\times R=OP\times RC_i\times RD_i\times R$, 表示权限和角色之间的多对多指定关系;

其中:

$roles(u_i)=\{r\in R|(u_i,r)\in UA\}$ 任意一个用户 u_i 对应一个角色集合;

$class\_resource(r_i)=\{rc\in RC|(rc,r_i)\in RA\}$ 任意一个角色对应一个资源类;

$district\_resource(u_i,r_i)=\{rd_i\in RD|rc_i\in RC,(rd,u_i)\in DA,(rc,r)\in RA\}$ 任意一个用户 u_i 对一个角色对应的一个资源区域集合;

由以上定义与推论,我们可以推导出对一个用户的某一角色的权限:

$operations(u_i,r_i)=\{op\in OP,rc_i\in RC,rd_i\in RD|(op,class\_resource(r_i)_i,district\_resource(u_i,r_i),r_i)\in PA\}$

3.3 RARBAC 模型中资源抽象涉及到的关系与规则

新模型中资源的关系主要是继承,继承关系包括资源类间的继承和资源区域的继承。

资源类间的继承关系:

定义 $RCH\in RC\times RC$ 上 RC 的二元关系,记为 $\geqslant_{rc}$,且有:

$\forall A,B\in RC,A\geqslant_{rc}B\Leftrightarrow A$ 包括 B 所有的资源,还可拥有自己的资源,这种继承关系满足自反、传递和反对称关系,是一种偏序关系。

资源区域间的继承:

定义 $RDH\in RD\times RD$ 上 RD 的二元关系,记为 $\leqslant_{rd}$,且有:

$\forall A,B\in RC,A\leqslant_{rd}B\Leftrightarrow A$ 是 B 子区域,B 还可拥有其他子区域,

这也是一种偏序关系。

根据以上定义的关系可以给出相应规则:

$\forall A\in RC_i,B\in RC_i\Rightarrow A\cap B=\phi$,同层次的资源类互不相交。

$\forall i,1\leqslant i\leqslant n,\bigcup_{A\in RC_i}A=OBJ$,资源在每一层都必须归属一个资源类。

3.4 RARBAC 模型中权限的分配与使用

RARBAC 模型中通过对资源进行抽象,将资源分解成资源类和资源区域,使得模型在角色的创建时能更好的对权限进行细粒度抽象,解决了角色冗余问题。

RARBAC 模型中权限分配与使用与经典 RBAC 模型有一定的区别。

经典 RBAC 中，权限直接赋予角色，用户通过担任某些角色而获得访问权限，这样相同角色具有相同权限。但在 RARBAC 中，由于资源的分解，权限 P 相应的分解为 P1，P2 两部分，其中 $P1 = OP \times RC_i$，$P2 = RD_i$，如图 2，权限的分配分为两步：

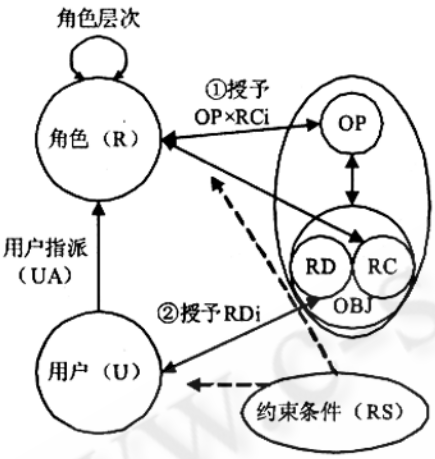

图 2　RARBAC 中的权限的分配

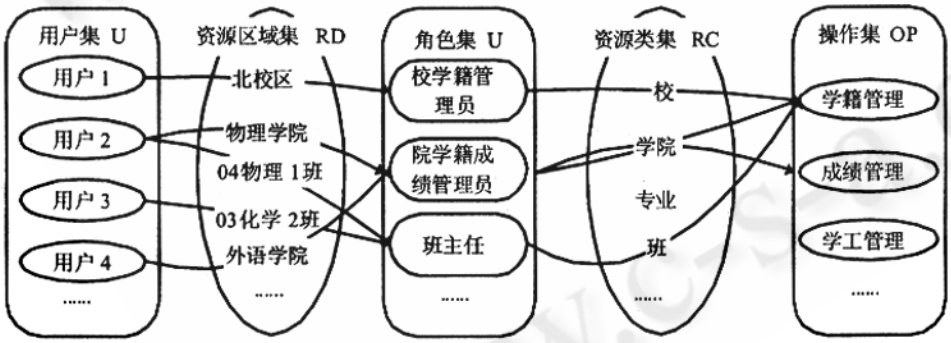

图 3　用户—资源—角色关系实例图

（1）创建角色时，对权限进行两次抽象，然后将操作与资源类，即 P1 赋给角色；

（2）给用户赋予角色时，根据角色与实际情况赋给用户对应的资源区域，即 P2；

RARBAC 模型中，角色与用户都没有独立权限，用户与角色结合在一起才能的到完整的权限。这样，同一角色的用户根据资源区域的不同拥有的是不同的权限。而且，RARBAC 模型可以通过对角色选取不同的资源类从而自由控制访问控制粒度。RARBAC 模型的权限分配分为两步，同样，用户权限使用也分为两步：

（1）根据用户当前所处的角色得到 P1

（2）根据用户相应角色的区域属性的到 P2，从而得到权限 $P = P1 \times P2 = P \times RC_i \times RD_i$.

3.5 RARBAC 模型的性能分析

RARBAC 在经典 RBAC 一次抽象的基础上对资源进行二次抽象，使的对权限的抽象度大大提高，冗余角色将大幅减少。其角色规模减小的定性分析如下。

假设：一大型系统中，有 m 种操作，n 种资源（$n = (2\alpha)^x$），其资源呈层次分布，管理人员将资源分为 x 层，每层的资源区域为 2^{i-1}（i 为所在层），m，n，i 均为一正整数，而且 $1 \leqslant i \leqslant x$ 则：

由定义 1 与追加定义 1 可知

$|OP| = m$，$|RD_i| = n/(2^{i-1})$，$|RC_i| = 2^{i-1}$：那么，使用经典 RBAC 进行完全的权限管理所需要的角色数量为：

$$|R_0G| = \sum_{i=1}^{x}\left(\sum_{i=1}^{m} C_m^i \times \sum_{k=1}^{2^{i-1}} C_{2^{i-1}}^k\right),$$

$$= \sum_{i=1}^{m} C_m^i \times \sum_{i=1}^{x}\left(\sum_{k=1}^{2^{i-1}} C_{2^{i-1}}^k\right)$$

$$= (2^m - 1) \times \sum_{i=1}^{x}(2^{2^{i-1}} - 1)$$

$$= (2^m - 1) \times [(2^1 + 2^2 + 2^4 + \cdots 2^{2^{i-1}} - x]$$

$$= (2^m - 1) \times [(4^x - 1)/3 + 1 - x]$$

而使用 RARBAC 模型进行完全权限管理所需要的角色数量为：

$$|RDG| = \sum_{i=1}^{x}\left(\sum_{i=0}^{m} C_m^i \times 1\right)$$

（每一层对于资源只有一种划分，那么一层只有一个角色）

$$= x(2^m - 1)$$

令 $T = |RDG|/|R_0G|$ 则

$$T = x(2^m - 1)/(2^m - 1) \times [(4^x - 1)/3 + 1 - x]$$

$$T = x/[(4^x - 1)/3 + 1 - x]$$

当 $x = 1$ 时 $T = 1$，这意味着当资源权限为粗粒度，即不对资源做划分时，RARBAC 与经典 RBAC 需要的角色数目是一样的。

一旦系统对资源实行细粒度管理，即 $x > 1$ 时，$T = x/\times[(4^x - 1)/3 + 1 - x] < 1$，则 RARBAC 需要的角色数

量要少于 RBAC 的角色数量。

层次越多，粒度越小，即 x 越大，T 越小，当 $x \to +\infty$，则 $T \to 0$，可见 RARBAC 模型在多级复杂权限系统中比经典 RBAC 模型具有更好的适应性。

3.6 RARBAC 模型在 PSMSS 中的应用

PSMSS (Permission Security Management Subsystem) 是在高校教务管理系统的研制和开发过程中，实现的一种基于 RARBAC 模型的权限安全管理子系统。PSMSS 实现了角色与资源的结合，使得安全管理更加完善，它在 DBMS/MIS 中，解决具有大量用户和权限分配的复杂性问题时很有成效，同时系统允许用户可以根据本身的具体情况实施安全策略。PSMSS 适用于大中型信息系统，实现企业级的安全管理策略。

在高校教务管理系统中，系统用户有校教务工作人员、学院教务工作人员，另外涉及其他相关单位，如财务处、学工处、评估办、研究生处等等。在各院各系中，用户又可分为不同的级别，各自执行不同的职责。不同级别、不同职责的用户，他们所能够访问的后台数据库中的信息是不同的。因此角色权限粒度的划分是高校教务管理系统实施安全策略的一个关键问题。一般情况下粒度过大，将会影响数据和数据库系统的安全性，不容易实现最小特权原则；粒度过小，角色数量的增加，提高了角色管理的复杂性。前文已经提到，RARBAC 模型利用对资源的二次抽象解决了这个矛盾。

在 PSMSS 设计方案中，采用了 RARBAC 模型：首先集中地为角色授予相应的操作权限，并为每一个角色赋予一个资源类；然后根据用户所在区域、单位、部门不同的职责范围，确定其一个或多个角色，并相应的为每个角色赋予角色一个资源区域。这使得用户可以根据自己所拥有的角色和相应的资源区域合法地执行相应的数据访问和系统操作权利，确保了数据信息的安全。图 3 是 PSMSS 应用资源二次抽象的实例图。

4 结束语

本文在 RBAC96 模型基础上，针对资源进行的二次抽象，提出一种新的访问控制模型——基于资源抽象的 RBAC 模型 (RARBAC)。该模型在对角色授权以及实现系统安全策略的过程中，利用对资源的再划分，减少冗余角色，降低管理复杂度。和经典 RBAC 模型比较，RARBAC 模型能自由控制访问控制粒度，具有更好的适应性和安全性。目前 RARBAC 模型已经在高校管理系统 UEAMS 系统中成功实现，下一步将对访问控制效率的提高、角色代理模型[10]的实现等做进一步研究。

参考文献

1 Ferraiolo David, Kuhn Richard. Role – based access controls[C]. Proceedings of the 15th NIST – NCSC National Computer Security Conference. Baltimore: NIST – NCSC, 1992: 554 – 563.

2 Ravi Sandhu, Edward Coyne, Hat Feinstein and Charles Youman. Role – Based Access Control Models[J]. IFEE Computer, 1996, 29(2): 38 – 47.

3 钟华、冯玉琳、姜洪安，扩充角色层次关系模型及其应用[J]，软件学报，2000，11(6)：779 – 784.

4 Freudenthal E, Pesin T, Port L, et al. dRBAC: Distributed Role – based Access Control for Dynamic Coalition Environments[R]. Technical Report TR2001 – 819, New York University, 2001.

5 梁彬、孙玉芳，一种改进的以基于角色的访问控制实施 BLP 模型及其变种的方法[J]，计算机学报，2004，27(5)：636 – 644.

6 杨亚平、李伟琴、刘怀宇，基于角色的细粒度的访问控制系统的研究与实现[J]，北京航空航天大学学报，2001，27 (2)：178 – 181.

7 Sandhu, R. S. and Samarati, P. Access Control: Principles and Practice[J]. IEEE Communications, 1994, 32(9): 40 – 48.

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

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

10 Longhua Zhang, Gail – Joon Ahn, Bei – Tseng Chu, A Role – Based Delegation Framework for Healthcare Information Systems[C]. In: SACMAT' 02, 2002.