基于双层角色和组织的可扩展访问控制模型

熊厚仁*①② 陈性元①② 张 斌①② 杜学绘①③
①(解放军信息工程大学 郑州 450001)
②(河南省信息安全重点实验室 郑州 450001)
③(数学工程与先进计算国家重点实验室 郑州 450001)

摘 要：针对现有基于角色的访问控制(RBAC)研究存在角色设置单一使得适应性差、多域环境下角色或权限冗余、对资源管理关注不够等问题，论文提出支持资源管理的基于双层角色和组织的访问控制模型。通过双层角色划分，提出基于职能角色和任务角色的双层角色架构，使得模型更加符合实际，也更具适应性；引入组织的概念并与双层角色相结合，对角色和权限的概念加以扩展，形式化定义了提出的基于双层角色和组织的访问控制模型，描述了影响模型安全的职责分离约束和势约束。对模型的表达能力、复杂度进行了分析，分析表明该机制不仅保留了RBAC的特点与优势，且比RBAC具有较低的复杂度并更适合于由多个相似组织构成的分布式多域环境。

关键词：网络信息安全；基于角色的访问控制；双层角色；组织；角色继承；职责分离
中图分类号： TP393.08 **文献标识码**： A **文章编号**：1009-5896(2015)07-1612-08
DOI: 10.11999/JEIT141255

Scalable Access Control Model Based on Double-tier Role and Organization

Xiong Hou-ren①② Chen Xing-yuan①② Zhang Bin①② Du Xue-hui①③
①(*PLA Information Engineering University*, *Zhengzhou* 450001, *China*)
②(*Henan Key Laboratory of Information Security*, *Zhengzhou* 450001, *China*)
③(*State Key Laboratory of Mathematical Engineering and Advanced Computing*, *Zhengzhou* 450001, *China*)

Abstract: For tackling the deficiencies of weak adaptability due to the singleness of the role establishment method, role or privilege redundancy, and little attention on resource management in the existing Role-Based Access Control (RBAC) researches, a Scalable Access Control model Based on Double-Tier Role and Organization (SDTR-OBAC) is proposed. Through double role partition, a double-tier role architecture of function role and task role is presented, solving the problem that the traditional role can not cover the requirements of both organizational level and application level at the same time. The concept of organization is introduced to integrate with the double-tier role and form an organization-role pair assigned to user instead of role only in RBAC, making model suitable to cross-domain access as well as a single domain. Through extending privileges as an operation and resource type pair, the model and its constraints including separation of duty and cardinality constraint are defined formally. The discussion of expressive power and complexity indicates that SDTR-OBAC retains all the advantages of RBAC, and can effectively reduce the administration complexity with better scalability and universality.

Key words: Network information security; Role-Based Access Control (RBAC); Double-tier role; Organization; Role inheritance; Separation of duty

1 引言

通过引入角色的概念，基于角色的访问控制模型(Role-Based Access Control, RBAC)[1]中用户不是直接与权限相关联，而是将权限赋予角色，通过为用户分配合适的角色从而获得指定权限，极大地降低了授权管理复杂度，并具有策略中立、强扩展性、易于管理等特点，更适用于现代信息系统。自提出以来，RBAC得到了广泛的研究和应用，包括RBAC管理模型ARBAC研究[2]、基于时间和空间的RBAC研究[3,4]、RBAC中的职责分离约束研究[5]及结合其它约束研究[6]、基于工作流的RBAC研究[7]、基于RBAC的委托研究[8]、RBAC与信任管理结合的研究[9]，跨域访问控制研究[10]，角色工程研究[11]及围绕RBAC展开安全性分析研究[12,13]等。以

2014-09-25收到，2015-02-11改回，2015-05-08网络优先出版
国家863计划项目(2012AA012704)和2014年河南省基础研究计划项目(142300413201)资助课题
*通信作者：熊厚仁 xionghouren@163.com

上研究根据实际应用需要对经典RBAC进行改进或扩展，从而满足不同应用环境的特定需求。

现有针对 RBAC 模型的应用及研究继承了RBAC 的诸多优点，但仍存在一些适应性、安全性及复杂度等方面的问题，具体表现为：

(1)传统角色概念不能同时满足组织层面和应用层面的访问控制需求。RBAC 模型中的授权管理包括用户授权(user-role assignment)和角色授权(role-permission assignment)。在实际系统中，人员信息和资源信息往往交给不同人员管理，分别由人事部门和信息管理部门负责。在较大规模的组织或企业中，很难找到既熟知人员的职责分工，又熟悉应用系统业务流程的管理人员。现有研究中，RBAC模型单一的角色设置不符合现实世界的真实情况，其适用性较低。文献[14]提出将角色划分为职能角色和任务角色的思想以解决以上问题，但该方案用于具有多个相似组织的大型分布式系统时易带来角色和权限数量过多等问题。

(2)在由多个相似组织组成的大型分布式网络中，现有研究存在角色、权限数量过大和冗余等不足，易导致权限分配繁琐、管理复杂等问题。分布式网络呈现多域、动态等特点，由多个具有组织结构特点的域构成，且每个域都采用 RBAC 模型时，需要为每个域定义相应的角色及权限，不仅容易造成角色和权限冗余，也带来了较大的管理复杂度，特别是增加了跨域访问控制的难度和复杂性。文献[15]在 RBAC 的基础上引入组织的概念以解决该问题，但存在单一角色设置及私有权限得不到有效保护的问题。

(3)现有研究大多将权限视为一个整体，主要集中于用户权限分配、权限约束的管理，忽视了资源的重要性，或者未对面向 RBAC 的授权管理中的资源、资源操作及权限分配的管理进行深入具体的描述，或者其资源管理不具有通用性和可扩展性。

针对以上问题，本文对文献[14]提出的双层角色进行延伸，在文献[15]引入的基于组织和角色的访问控制的基础上，将双层角色与组织相结合，提出支持资源管理的基于双层角色和组织的可扩展访问控制模型(Scalable Access Control Model Based on Double-Tier Role and Organization, SDTR-OBAC)，解决现有研究中存在的角色设置单一使得适应性差、存在角色或权限冗余、资源管理得不到足够重视等问题。

2　基于角色区分的双层角色架构

RBAC 中用户授权和角色授权部分的管理由不同管理员完成。若角色与组织架构中的职能分工对应，用户授权工作比较直观方便，但角色授权工作就非常复杂；若角色按照应用系统内的业务划分制定，则角色授权工作可以由应用系统管理员完成，但用户授权则会变得比较繁琐。将授权管理工作按照组织层面和应用层面进行区分，可以极大地降低管理负担和复杂度。通过将角色概念进行拆分，提出基于角色区分的双层角色架构：在组织层面，按照组织架构中用户的职责分工情况，设置职能角色；在应用层面，按照应用系统内的业务划分和资源属性，设置任务角色，如图 1 所示。

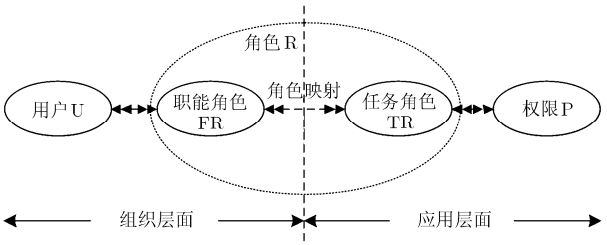

图 1 基于职能角色和任务角色的双层角色架构

与 RBAC 中的角色层次类似，两种角色均具有角色层次结构及与之对应的角色树。其中任务角色直接与权限相关联，是权限的集合，其层次结构能够体现权限的继承关系；职能角色树中上下级节点之间是部门间的层次关系和岗位对部门的隶属关系，没有权限继承关系。

3　基于双层角色和组织的可扩展访问控制模型 SDTR-OBAC

3.1 模型主要思想

通过引入组织的概念，将双层角色和组织相结合，并对权限概念加以扩展，改进经典 RBAC 模型，提出支持资源管理的基于双层角色和组织的可扩展访问控制模型 SDTR-OBAC，如图 2 所示。

SDTR-OBAC 模型的主要思想是：将传统的角色概念划分为职能角色和任务角色，解决传统角色概念不能同时满足组织层面和应用层面需求及适应性差的问题；引入组织的概念，代表进行协作的各个域，并将分配给用户的角色扩展为组织-职能角色二元组，将与权限关联的角色扩展为组织-任务角色二元组，通过组织-职能角色和组织-任务角色间的映射关系建立用户与权限之间的关联，解决由大量相似组织或域构成的分布式环境下角色、权限过多或冗余问题；经典 RBAC 模型中由操作和资源构成的权限扩展为由操作和资源类型构成的二元组，提高角色授权管理效率；对职能角色、任务角色分别定

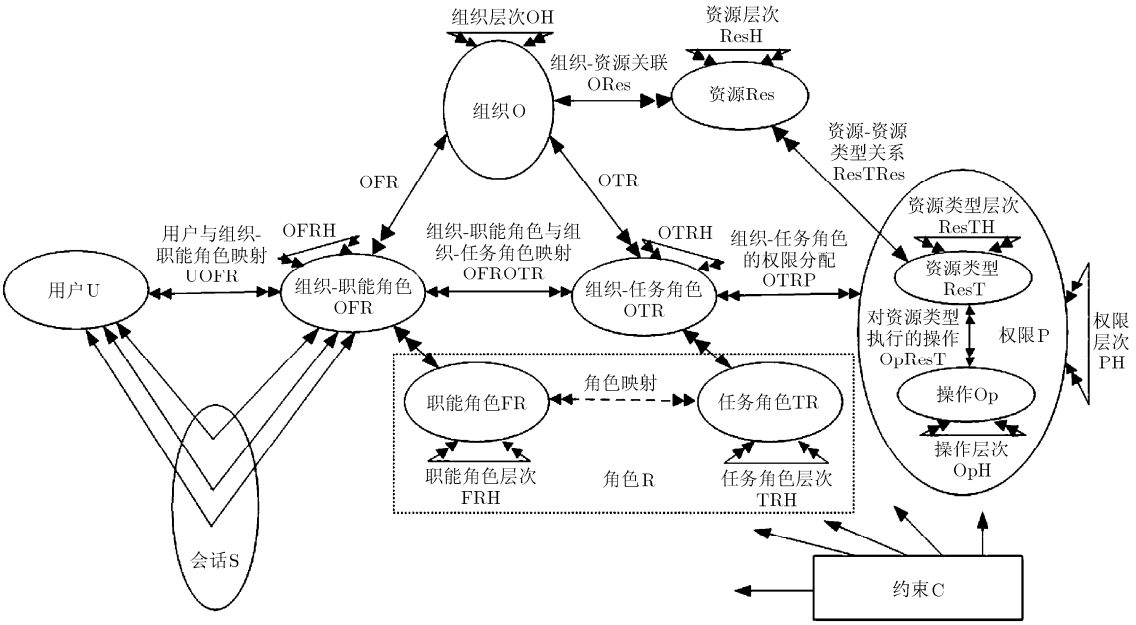

图 2 基于双层角色和组织的可扩展访问控制模型 SDTR-OBAC

义不同的继承关系，解决下层角色的敏感权限得不到有效保护的问题；分析描述了针对资源、操作和权限等的管理，解决现有研究缺乏面向授权管理的资源管理的问题；引入职责分离约束、势约束等授权管理安全约束，对权限继承、用户角色分配、角色权限分配和角色映射关系等问题进行限制，提高授权管理的安全性。

3.2 形式化描述

本文采用集合论和一阶逻辑分别对模型的元素、关系、函数和约束进行形式化定义。

3.2.1 模型元素与关系

定义 1 模型元素

(1) U，S，Op，Res：与经典 RBAC 定义相同，分别表示用户、会话、操作和资源的集合。

(2) O：组织集合，代表构成分布式系统的各个域；R：角色集合，包含职能角色 FR 和任务角色 TR，即 $\mathrm{R}=\mathrm{FR}\cup\mathrm{TR}$；ResT：资源类型集合。

(3) OFR，OTR：分别表示组织-职能角色集合和组织-任务角色集合，$\mathrm{ofr}\in\mathrm{OFR}$ 和 $\mathrm{otr}\in\mathrm{OTR}$ 是由组织分别与职能角色和任务角色构成的二元组，即 $\mathrm{ofr}=(o,\mathrm{fr})$，$\mathrm{otr}=(o,\mathrm{tr})$，其中 $\mathrm{fr}\in\mathrm{FR}$，$\mathrm{tr}\in\mathrm{TR}$。

(4) P：权限，对经典 RBAC 中权限的概念进行扩展，权限 $p\in\mathrm{P}$ 是由操作和资源类型构成的二元组，即 $p=(\mathrm{op},\mathrm{rt})$，其中 $\mathrm{op}\in\mathrm{Op}$，$\mathrm{rt}\in\mathrm{ResT}$。

(5) C：约束，对用户与组织-职能角色分配、组织-职能角色与组织-任务角色映射、组织-任务角色的权限分配、组织层次关系、角色层次关系等进行限制，主要包括职责分离约束 SoD，势约束 Cardinality 等。

定义 2 模型关系

(1) $\mathrm{UOFR}\subseteq\mathrm{U}\times\mathrm{O}\times\mathrm{FR}$：多对多的用户与组织-职能角色分配关系，类似于经典 RBAC 中的 UA；$\mathrm{US}\subseteq\mathrm{U}\times\mathrm{S}$：用户与会话关系，用户可激活多个会话，但一个会话只能属于一个用户；$\mathrm{SOFR}\subseteq\mathrm{S}\times\mathrm{O}\times\mathrm{FR}$：会话与组织-职能角色对之间的多对多映射关系；$\mathrm{OTRP}\subseteq\mathrm{O}\times\mathrm{TR}\times\mathrm{P}$：组织-任务角色与权限映射关系，类似于经典 RBAC 中的 PA。

(2) $\mathrm{OFR}\subseteq\mathrm{O}\times\mathrm{FR}$，$\mathrm{OTR}\subseteq\mathrm{O}\times\mathrm{TR}$，$\mathrm{ORes}\subseteq\mathrm{O}\times\mathrm{Res}$：分别表示多对多的组织与职能角色、组织与任务角色、组织与资源之间的关联关系。

(3) $\mathrm{FRTR}\subseteq\mathrm{FR}\times\mathrm{TR}$，$\mathrm{OFROTR}\subseteq\mathrm{OFR}\times\mathrm{OTR}$：分别表示多对多的职能角色与任务角色映射关系和多对多的组织-职能角色与组织-任务角色映射关系。OFROTR 主要由组织之间的信任关系和 FRTR 决定，若两个组织 o_1，o_2 间的信任关系表示为 $o_1\leftrightarrow o_2$，则 $\mathrm{OFROTR}=\{((o_1,\mathrm{fr}),(o_2,\mathrm{tr}))\mid(o_1,\mathrm{fr})\in\mathrm{OFR}\wedge(o_2,\mathrm{tr})\in\mathrm{OTR}\wedge o_1\leftrightarrow o_2\wedge(\mathrm{fr},\mathrm{tr})\in\mathrm{FRTR}\}$。

(4) $\mathrm{OpResT}\subseteq\mathrm{Op}\times\mathrm{ResT}$，$\mathrm{PResT}\subseteq\mathrm{P}\times\mathrm{ResT}$，$\mathrm{POp}\subseteq\mathrm{P}\times\mathrm{Op}$，$\mathrm{ResTRes}\subseteq\mathrm{ResT}\times\mathrm{Res}$：分别表示操作与资源类型关联关系、权限和资源类型的关联关系、权限和操作的关联关系和资源与资源类型的隶属关系。

(5) $\mathrm{OH}\subseteq\mathrm{O}\times\mathrm{O}$：组织层次关系，指组织之间的上下级隶属关系和管理关系，具有自反性、反对称性和可传递性，是偏序关系，两个组织 $o_1\in\mathrm{O}$，

$o_2 \in \mathrm{O}$ 间的层次关系定义为 $(o_1, o_2) \in \mathrm{OH}$，表示 $o_1 \le o_2$。

(6) $\mathrm{RH} \subseteq \mathrm{R} \times \mathrm{R}$：角色层次关系，与经典 RBAC 相同，包含职能角色层次关系 $\mathrm{FRH} \subseteq \mathrm{FR} \times \mathrm{FR}$ 和任务角色层次关系 $\mathrm{TRH} \subseteq \mathrm{TR} \times \mathrm{TR}$，即 $\mathrm{RH} = \mathrm{FRH} \cup \mathrm{TRH}$。与组织层次关系 OH 类似，角色层次关系 RH 也具有自反性、反对称性和传递性，是偏序关系。为了保护下层角色的敏感权限，两类角色采取不同的继承策略，即职能角色层次中不存在权限继承和用户继承关系，上下级角色间只存在管理关系，即 $(\mathrm{fr}_1, \mathrm{fr}_2) \in \mathrm{FRH}$ 表示为 $\mathrm{fr}_1 \le \mathrm{fr}_2$；而任务角色层次中存在权限继承但不存在职能角色继承，即 $(\mathrm{tr}_1, \mathrm{tr}_2) \in \mathrm{TRH}$ 表示为 $\mathrm{tr}_1 \preceq \mathrm{tr}_2$。

(7) $\mathrm{OFRH} \subseteq \mathrm{OFR} \times \mathrm{OFR}$：组织-职能角色层次关系，依赖于组织层次关系和职能角色层次关系，是偏序关系，即 $(\mathrm{ofr}_1, \mathrm{ofr}_2) \in \mathrm{OFRH}$ 或 $\mathrm{ofr}_1 \le \mathrm{ofr}_2$ 当且仅当 $o_1 \le o_2 \wedge \mathrm{fr}_1 \le \mathrm{fr}_2$，其中 $\mathrm{ofr}_1 = (o_1, \mathrm{fr}_1)$，$\mathrm{ofr}_2 = (o_2, \mathrm{fr}_2)$。

(8) $\mathrm{OTRH} \subseteq \mathrm{OTR} \times \mathrm{OTR}$：组织-任务角色层次关系，依赖于组织层次关系和任务角色层次关系，是偏序关系，即 $(\mathrm{otr}_1, \mathrm{otr}_2) \in \mathrm{OTRH}$ 或 $\mathrm{otr}_1 \preceq \mathrm{otr}_2$ 当且仅当 $o_1 \le o_2 \wedge \mathrm{tr}_1 \preceq \mathrm{tr}_2$，其中 $\mathrm{otr}_1 = (o_1, \mathrm{tr}_1)$，$\mathrm{ofr}_2 = (o_2, \mathrm{tr}_2)$。

(9) $\mathrm{ResH} \subseteq \mathrm{Res} \times \mathrm{Res}$：资源层次关系，主要指资源间的包含关系，是偏序关系，$(\mathrm{re}_1, \mathrm{re}_2) \in \mathrm{ResH}$ 表示 $\mathrm{re}_1 \preceq \mathrm{re}_2$；$\mathrm{ResTH} \subseteq \mathrm{ResT} \times \mathrm{ResT}$：资源类型层次关系，是资源类型间的包含关系，是偏序关系，$(\mathrm{rt}_1, \mathrm{rt}_2) \in \mathrm{ResTH}$ 表示 $\mathrm{rt}_1 \preceq \mathrm{rt}_2$；$\mathrm{OpH} \subseteq \mathrm{Op} \times \mathrm{Op}$：操作层次关系，主要指操作间的蕴含关系，如读写操作蕴含只读操作，是偏序关系，$(\mathrm{op}_1, \mathrm{op}_2) \in \mathrm{OpH}$ 表示 $\mathrm{op}_1 \preceq \mathrm{op}_2$。

(10) $\mathrm{PH} \subseteq \mathrm{P} \times \mathrm{P}$：权限层次关系，由操作蕴含关系和资源类型包含关系决定，是偏序关系，即 $(p_1, p_2) \in \mathrm{PH}$ 或 $p_1 \preceq p_2$ 当且仅当 $\mathrm{op}_1 \preceq \mathrm{op}_2 \wedge \mathrm{rt}_1 \preceq \mathrm{rt}_2$，其中 $p_1 = (\mathrm{op}_1, \mathrm{rt}_1)$，$p_2 = (\mathrm{op}_2, \mathrm{rt}_2)$。

其中，“$\le$”和“$\preceq$”的区别为前者不存在权限继承和用户继承关系，而后者存在权限继承关系。

3.2.2 模型函数

定义 3　模型函数　模型包含与以上关系对应的相关函数，限于篇幅，主要给出以下关键函数。

(1) user: $\mathrm{S} \to \mathrm{U}$：会话到用户的映射，通过该函数查找与会话 s 关联的用户 u。

(2) restypes: $\mathrm{Res} \to 2^{\mathrm{ResT}}$：资源到其所属资源类型集合的映射，某具体资源可属于多种资源类型。

(3) resorgs: $\mathrm{Res} \to 2^{\mathrm{O}}$：资源到其所隶属的组织集合的映射，资源可隶属于多个组织。

(4) assigned_orgs-trole: $\mathrm{OFR} \to 2^{\mathrm{OTR}}$：组织-职能角色到组织-任务角色集合的映射，可为某组织-职能角色对映射多个组织-任务角色对。

(5) assigned_orgs-frole: $\mathrm{U} \to 2^{\mathrm{OFR}}$：用户到为其分配的组织-职能角色集合的映射，形式化表示为

$$\begin{aligned}&\text{assigned\_orgs-frole}(u)\\&\quad=\{(o,\mathrm{fr}) \mid \exists o' \in \mathrm{O}, \mathrm{fr}' \in \mathrm{FR},\\&\qquad(o \le o' \wedge \mathrm{fr} \le \mathrm{fr}' \wedge (u,(o',\mathrm{fr}')) \in \mathrm{UOFR})\}\end{aligned}$$

(6) assigned_users: $\mathrm{OFR} \to 2^{\mathrm{U}}$：组织-职能角色到用户集合的映射，形式化表示为：$\text{assigned\_users}((o, \mathrm{fr})) = \{u \mid (u,(o, \mathrm{fr})) \in \mathrm{UOFR}\}$。

(7) active_orgs-frole: $\mathrm{S} \to 2^{\mathrm{OFR}}$：会话到其可用的组织-职能角色集合的映射，形式化表示为：$\text{active\_orgs-role}(s) \subseteq \text{assigned\_orgs-frole}(\text{user}(s))$。

(8) assigned_privilege: $\mathrm{OTR} \to 2^{\mathrm{P}}$：组织-任务角色到权限集合的映射，包含直接分配的权限和继承而来的权限，形式化描述为：$\text{assigned\_privilege}(o,\mathrm{tr}) = \{p \in \mathrm{P} \mid \exists \mathrm{tr}' \preceq \mathrm{tr} \wedge \exists o' \le o \wedge ((o',\mathrm{tr}'), p) \in \mathrm{OTRP}\}$。

(9) $\text{can\_access}(\mathrm{U},\mathrm{S},\mathrm{Op},\mathrm{Res})$：用于判断某用户是否可通过激活会话对资源执行特定的操作。$\text{can\_access}(u,s,\mathrm{op},\mathrm{re}) = \text{true}$ 表示用户 u 可通过激活会话 s 对资源 re 执行 op 操作。$\text{can\_access}(u, s,\mathrm{op},\mathrm{re}) = \text{true}$ 成立当且仅当下式成立：$u = \text{user}(s) \wedge (o,\mathrm{fr}) \in \text{active\_orgs-frole}(s) \wedge (\exists o' \le o, o' \in \text{resorgs}(\mathrm{re})) \wedge (\exists (o'', \mathrm{tr}) \preceq \text{assigned\_orgs-trole}(o', \mathrm{fr}) \wedge (\exists \mathrm{op}', \mathrm{op} \preceq \mathrm{op}') \wedge (\exists \mathrm{rt}, \text{restype}(\mathrm{re}) \preceq \mathrm{rt}) \wedge (\mathrm{op}', \mathrm{rt}) \in \text{assigned\_privilege}(o'', \mathrm{tr}))$。

3.2.3 安全约束

定义 4　模型约束　约束是模型中用于限制 UOFR, OFROTR, OTRP 和 RH 等关系的重要内容，本文主要给出 UOFR 和 OFROTR 的约束，其他如 OTRP, RH 等的约束可类似定义，主要考虑职责分离约束 SoD 和势约束 Cardinality。

为了职责分离约束和势约束进行定义，引入通配符“?”和“*”表示组织 O 中任意一个组织 o 和 O 中不同的两个或多个组织，则可对职责分离约束和势约束进行如下形式化定义。

职责分离约束 SoD：$\mathrm{SoD} \subseteq (2^{\mathrm{RO}^+} \times N)$。其中，$\mathrm{RO}^+ \subseteq \mathrm{R} \times \mathrm{O}^+$；$\mathrm{O}^+ = \mathrm{O} \cup \{?, *\}$，$N$ 是一个自然数集且满足 $\forall (\mathrm{ro}, n) \in \mathrm{SoD}$，$|\mathrm{ro}| \ge n \ge 2$，$n \in N$。

职责分离约束 $(\mathrm{ro}, n) \in \mathrm{SoD}$ 表示不能将 n 个或更多存在互斥关系的组织-角色对 ro 指派给某用户。

通配符“?”和“*”的区别是：“?”指组织 O 中的相同组织，而“*”则指组织 O 中的任意不同

取值。当只有一种取值时，“?”和“*”具有相同的含义。

势约束 Cardinality：cardinality:$\mathrm{RO}^+ \to N$，其中 $\mathrm{RO}^+ \subseteq \mathrm{R}\times\mathrm{O}^+$，$\mathrm{O}^+ = \mathrm{O}\cup\{?, *\}$，$N$ 是一个自然数集合，$\forall(r,o)\in \mathrm{RO}^+, |\,\mathrm{assigned\_users}((r,o))| \leqslant \mathrm{cardinality}((r,o))$。

势约束 $n=\mathrm{cardinality}((r,o))$ 表示能被指派组织 o 中角色 r 的用户数量是 $n=\mathrm{cardinality}((r,o))$。

当 $(o,r)\in \mathrm{OTR}$ 时，$\mathrm{assigned\_users}((r,o))$ 指经组织-职能角色映射获得组织-任务角色的用户。通配符“?”和“*”与职责分离约束 SoD 中定义相同。因为势约束中只有一种取值，因此在势约束中，“?”和“*”没有区别。例如，$\mathrm{cardinality}((r,?))=10$ 与 $\mathrm{cardinality}((r,*))=10$ 的含义是相同的，表示能够获得任意组织 o 中角色 r 的最大用户数量是 10。

3.2.4 授权管理操作

定义 5　授权管理操作　模型包含 3 类 32 种授权管理操作，分别为添加、删除用户，添加、删除职能角色，添加、删除任务角色，添加、删除组织，添加、删除操作，添加、删除资源类型，添加、删除资源，创建、删除静态互斥组织-职能角色集，创建动态互斥组织-职能角色集，创建静态互斥组织-任务角色集，创建动态互斥组织-任务角色集，添加、删除约束等 24 种系统要素管理操作；为用户分配、撤销组织-职能角色，创建、撤销组织-职能角色与组织-任务角色的映射关系，为组织-任务角色分配、撤销权限等 6 种权限授予与撤销操作；创建、结束会话 2 种用户访问行为管理操作。

限于篇幅，本文不对授权管理操作展开详细描述。

4　模型分析

本节主要对 SDTR-OBAC 模型的表达能力和复杂度进行分析。

4.1 表达能力分析

表达能力是评价访问控制模型优劣的一个重要指标，通过采用构造模型系统的方法及以下定理分析模型的表达能力，证明模型具有与经典 RBAC 相同的表达能力。

引理 1　任何基于经典 RBAC 的系统均可采用基于 SDTR-OBAC 的系统实现。

证明　给定任一经典 $\mathrm{RBAC}=(\mathrm{U,S,R,RH,Op,Res,P,UA,PA,user})$，构造一个 $\mathrm{SDTR-OBAC}=(\mathrm{U',S',R',RH',FR,FRH,TR,TRH,O,OH,OFR,OFRH,OTR,OTRH,OFROTF,Op',OpH,ResT,ResTH,Res',ResH,P',PH,UOFR,OTRP,user',restypes,resorgs})$，其中：

(1) $\mathrm{U'},\mathrm{S'},\mathrm{R'},\mathrm{RH'},\mathrm{Op'},\mathrm{Res'}$ 和 $\mathrm{user'}$ 与 RBAC 中的 $\mathrm{U},\mathrm{S},\mathrm{R},\mathrm{RH},\mathrm{Op},\mathrm{Res}$ 和 user 相同。

(2)对于组织 O，定义 $\mathrm{O}=\{o\,|\,o=\mathrm{resorgs}(\mathrm{rs}_i), i=1,2,\cdots,n\}$，即 RBAC 中的所有资源属于同一个组织。

(3) 对于 FR 和 TR，定义 $\mathrm{R}=\mathrm{FR}=\mathrm{TR}$ 且 $\mathrm{RH}=\mathrm{FRH}=\mathrm{TRH}$，与 RBAC 相同；对于 OFR，OTR，定义 $\mathrm{OFR}=\mathrm{OTR}=\mathrm{OR}=\{(o,r)\,|\,o\in\mathrm{O},r\in\mathrm{R}\}$；对于 $\forall(o,r)\in\mathrm{OR}$，定义 $((o,r),(o,r))\in\mathrm{ORH}$ 且 $\mathrm{ORH}=\mathrm{OFRH}=\mathrm{OTRH}$；对于 OFROTF 及 $\forall(o,r)\in\mathrm{OR}$，由于 $\mathrm{OFR}=\mathrm{OTR}=\mathrm{OR}$，定义 $((o,r),(o,r))\in\mathrm{OFROTF}$。

(4) 对于资源类型 ResT，定义 $\mathrm{ResT}=\{\mathrm{rt}_i\,|\,\mathrm{rt}_i=\mathrm{restypes}(\mathrm{re}_i), i=1,2,\cdots,n\}$；对于 $\forall\mathrm{re}_i\in\mathrm{Res}$ 和 $\forall\mathrm{rt}_i\in\mathrm{ResT}$，分别定义 $(\mathrm{re}_i,\mathrm{re}_i)\in\mathrm{ResH}$ 和 $(\mathrm{rt}_i,\mathrm{rt}_i)\in\mathrm{ResTH}$。

(5) 对于 $\forall\mathrm{op}_i\in\mathrm{Op}$，定义 $(\mathrm{op}_i,\mathrm{op}_i)\in\mathrm{OpH}$；$\mathrm{P'}\subseteq\mathrm{Op}\times\mathrm{ResT}$，对于 RBAC 中的任一 $(\mathrm{op},\mathrm{re}_i)\in\mathrm{P}$，定义 $(\mathrm{op},\mathrm{rt}_i)\in\mathrm{P'}$，其中 rt_i=restypes(re_i)；对于 $\forall(\mathrm{op},\mathrm{rt}_i)\in\mathrm{P'}$，定义 $((\mathrm{op},\mathrm{rt}_i),(\mathrm{op},\mathrm{rt}_i))\in\mathrm{PH}$。

(6) $\mathrm{UOFR}\subseteq\mathrm{U'}\times\mathrm{OFR}$，对于 RBAC 中的 $\forall(u,r)\in\mathrm{UA}$，定义 $(u,(o,r))\in\mathrm{UOFR}$；$\mathrm{OTRP}\subseteq\mathrm{OTR}\times\mathrm{P'}$，对于 RBAC 中的 $\forall(r,p)=(r,(\mathrm{op},\mathrm{re}_i))\in\mathrm{PA}$，定义 $((o,r),(\mathrm{op},\mathrm{rt}_i))\in\mathrm{OTRP}$。

由此可见，基于经典 RBAC 的系统可通过构造基于 SDTR-OABC 的系统实现。　　证毕

引理 2　基于 SDTR-OBAC 的系统可通过基于经典 RBAC 的系统实现。

证明　对于任一 $\mathrm{SDTR-OBAC}=(\mathrm{U,S,R,RH,FR,FRH,TR,TRH,O,OH,OFR,OFRH,OTR,OTRH,OFROTF,Op,OpH,ResT,ResTH,Res,ResH,P,PH,UOFR,OTRP,user,restypes,resorgs})$，构造经典 $\mathrm{RBAC}=(\mathrm{U',S',R',RH',Op',Res',P',UA',PA',user'})$，其中：

(1) $\mathrm{U'},\mathrm{S'},\mathrm{Op'},\mathrm{Res'}$ 和 $\mathrm{user'}$ 与 SDTR-OBAC 中的 $\mathrm{U},\mathrm{S},\mathrm{Op},\mathrm{Res}$ 和 user 相同。

(2)对于 SDTR-OBAC 中的 $\forall o_i\in\mathrm{O},\forall o_j\in\mathrm{O},\forall\mathrm{fr}_k\in\mathrm{FR},\forall\mathrm{tr}_l\in\mathrm{TR}$ 和 $\forall((o_i,\mathrm{fr}_k),(o_j,\mathrm{tr}_l))\in\mathrm{OFROTF}$，定义角色 $r_{\mathrm{ik}}\in\mathrm{R'}$，即 SDTR-OBAC 模型中的每个组织-角色对映射关系根据组织-职能角色对 (o_i,fr_k) 定义成 RBAC 模型中单独的角色。

(3) 对于 $\mathrm{OH}\subseteq\mathrm{O}\times\mathrm{O}$，$\mathrm{RH}\subseteq\mathrm{R}\times\mathrm{R}$ 和 $\mathrm{OFRH}\subseteq\mathrm{OFR}\times\mathrm{OFR}$，定义 $\mathrm{RH'}=\{(r_{\mathrm{ik}},r_{\mathrm{jl}})\,|\,(r_k,r_l)\in\mathrm{RH}\wedge(o_i,o_j)\in\mathrm{OH}\wedge((o_i,\mathrm{fr}_k),(o_j,\mathrm{fr}_l))\in\mathrm{OFRH}\}$。

(4)对于 $\mathrm{P}\subseteq\mathrm{Op}\times\mathrm{ResT}$ 及 $\forall(\mathrm{op}_i,\mathrm{rt}_j)\in\mathrm{P}$，定义

$\mathrm{P}' = \mathrm{P}' \cup \{(\mathrm{op}_i, \mathrm{re}_j) \mid \mathrm{restypes}(\mathrm{re}_j) = \mathrm{rt}_j\}$。

(5) 对于 $\mathrm{UOFR} \subseteq \mathrm{U} \times \mathrm{OFR}$ 及 $\forall (u,(o_i,\mathrm{fr}_k)) \in \mathrm{UOFR}, \forall ((o_i,\mathrm{fr}_k),(o_j,\mathrm{tr}_l)) \in \mathrm{OFROTF}$，定义 $(u, r_{\mathrm{ik}}) \in \mathrm{UA}$。

(6) 对于 $\mathrm{OTRP} \subseteq \mathrm{OTR} \times \mathrm{P}$ 及 $\forall ((o_j,\mathrm{tr}_l),(\mathrm{op}_m, \mathrm{rt}_n)) \in \mathrm{OTRP}, \forall ((o_i,\mathrm{fr}_k),(o_j,\mathrm{tr}_l)) \in \mathrm{OFROTF}$，定义 $\mathrm{PA}' = \mathrm{PA}' \cup \{(r_{\mathrm{ik}},(\mathrm{op}_m,\mathrm{re}_n) \mid \mathrm{restypes}(\mathrm{re}_n)=\mathrm{rt}_n)\}$。

可见，基于经典 SDTR-OBAC 的系统可通过构造基于 RBAC 的系统实现。　　证毕

定理 1　SDTR-OBAC 模型具有与 RBAC 相同的表达能力。

证明　引理 1 说明 SDTR-OBAC 模型的表达能力比 RBAC 强；引理 2 说明 RBAC 的表达能力比 SDTR-OBAC 强。因此，根据引理 1 和引理 2 可知，SDTR-OBAC 模型与 RBAC 模型具有相同的表达能力。　　证毕

4.2 复杂度分析

RBAC 通过在用户和权限之间引入角色极大地降低了授权管理的负担和操作复杂度，但在由多个组织或域构成的大型分布式网络环境下，其优势就不再明显。SDTR-OBAC 通过引入组织的概念并将其与角色相结合，可在 RBAC 的基础上进一步降低复杂度，特别适合用于具有多个相似组织或域的分布式环境。

在分析 SDTR-OBAC 模型的操作复杂度之前，先定义以下同构度的概念。

定义 6　**同构度**　给定一个 SDTR-OBAC 模型，定义以下概念。

(1)对于 $o_i, o_j \in \mathrm{O}, r \in \mathrm{R}, \mathrm{R} = \mathrm{FR} \cup \mathrm{TR}$，当且仅当 $(o_i,r) \in \mathrm{OR} \wedge (o_j,r) \in \mathrm{OR}$ 时，o_i 与 o_j 对于 r 是同构的，记为：$o_i \equiv_r o_j$；对于 $o_i, o_j \in \mathrm{O}, \mathrm{Rc} \subseteq \mathrm{R}, \mathrm{R} = \mathrm{FR} \cup \mathrm{TR}$，当且仅当 $\forall r \in \mathrm{Rc}, (o_i,r) \in \mathrm{OR} \wedge (o_j,r) \in \mathrm{OR}$ 时，o_i 与 o_j 对于集合 Rc 是同构的，即对于任意 $r \in \mathrm{Rc}, o_i$ 与 o_j 是同构的，记为：$o_i \equiv_{\mathrm{Rc}} o_j$。

(2) $\mathrm{compatible\_O}^*: 2^{\mathrm{Rc}} \to 2^{\mathrm{O}}$ —— 该函数将某个角色集 Rc 映射到对于该角色集同构的所有组织的集合，形式化表示为：对于 $\mathrm{Rc} \subseteq \mathrm{R}, \mathrm{R} = \mathrm{FR} \cup \mathrm{TR}$ 且 $\mathrm{Rc} \neq \varnothing$，$\mathrm{compatible\_O}^*(\mathrm{Rc}) = \{o \mid \forall r \in \mathrm{Rc}, (o,r) \in \mathrm{OR}\}$，特别地，定义 $\mathrm{compatible\_O}^*(\varnothing) = \varnothing$；若 $1 < |\mathrm{compatible\_O}^*(\mathrm{Rc})| < |\mathrm{O}|$，则称 SDTR-OBAC 模型对于 Rc 是部分同构的，记为 Rc -部分同构；若 $\mathrm{compatible\_O}^*(\mathrm{Rc}) = \mathrm{O}$，则称 SDTR-OBAC 模型对于 Rc 是完全同构的，记为 Rc -完全同构；若 $|\mathrm{compatible\_O}^*(\mathrm{Rc})| = 1$，则称 SDTR-OBAC 模型对于 Rc 是异构的，记为 Rc -异构。

(3)同构度 hindex: $2^{\mathrm{Rc}} \to [0,1]$ —— 将角色集映射到区间 [0,1] 中某个实数的函数，形式化表示为：$\mathrm{hindex}(\mathrm{Rc}) = |\mathrm{compatible\_O}^*(\mathrm{Rc})| / |\mathrm{O}|$。

同构度函数 hindex(Rc) 用于衡量 SDTR-OBAC 模型系统中多个组织对于某特定角色集的同构程度，即该角色集在这些组织中的相似程度。若 SDTR-OBAC 模型是 Rc -完全同构的，则 $\mathrm{hindex}(\mathrm{Rc}) = 1$；若 SDTR-OBAC 模型是 Rc -异构的，则 $\mathrm{hindex}(\mathrm{Rc}) = 1/|\mathrm{O}|$；若 SDTR-OBAC 模型是 Rc -部分同构的，则 $1/|\mathrm{O}| < \mathrm{hindex}(\mathrm{Rc}) < 1$。

根据以上同构度的定义，可得出 hindex(R) 的以下两个性质。

定理 2　对于角色集 R 中任意两个非空子集 $\mathrm{Rc}_1, \mathrm{Rc}_2$，若 $\mathrm{Rc}_1 \subseteq \mathrm{Rc}_2$，则 $\mathrm{hindex}(\mathrm{Rc}_1) \geq \mathrm{hindex}(\mathrm{Rc}_2)$。

证明　给定条件 $\mathrm{Rc}_1 \neq \varnothing, \mathrm{Rc}_1 \subseteq \mathrm{Rc}_2$，$\forall r \in \mathrm{Rc}_1 \Rightarrow r \in \mathrm{Rc}_2$。

对于 $\forall o \in \mathrm{compatible\_O}^*(\mathrm{Rc}_2) \Rightarrow \forall r \in \mathrm{Rc}_2$，$(o,r) \in \mathrm{OR} \Rightarrow \forall r \in \mathrm{Rc}_1, (o,r) \in \mathrm{OR} \Rightarrow o \in \mathrm{compatible\_O}^*(\mathrm{Rc}_1)$。即 $\mathrm{compatible\_O}^*(\mathrm{Rc}_2)$ 中的任一组织 o 也是 $\mathrm{compatible\_O}^*(\mathrm{Rc}_1)$ 中的元素，因此 $|\mathrm{compatible\_O}^*(\mathrm{Rc}_2)| \leq |\mathrm{compatible\_O}^*(\mathrm{Rc}_1)|$，从而 $\mathrm{hindex}(\mathrm{Rc}_1) = |\mathrm{compatible\_O}^*(\mathrm{Rc}_1)| / |\mathrm{O}| \geq |\mathrm{compatible\_O}^*(\mathrm{Rc}_2)| / |\mathrm{O}| = \mathrm{index}(\mathrm{Rc}_2)$。　　证毕

定理 3　若 SDTR-OBAC 模型是 Rc -完全同构的，则对于 Rc 中的所有非空子集，SDTR-OBAC 模型也是完全同构的。

证明　根据同构度的定义，模型是 Rc -完全同构的，则 $\mathrm{compatible\_O}^*(\mathrm{Rc}) = \mathrm{O}$，从而 $\mathrm{hindex}(\mathrm{Rc}) = |\mathrm{compatible\_O}^*(\mathrm{Rc})| / |\mathrm{O}| = 1$。

从定理 2 可知，$\forall \mathrm{Rc}' \subseteq \mathrm{Rc} \wedge \mathrm{Rc}' \neq \varnothing$，则 $\mathrm{hindex}(\mathrm{Rc}') \geq \mathrm{hindex}(\mathrm{Rc}) = 1$ 成立，根据定义，$\mathrm{hindex}(\mathrm{Rc}') \leq 1$。

从而可得 $\mathrm{hindex}(\mathrm{Rc}') = 1$，则 $\mathrm{compatible\_O}^*(\mathrm{Rc}') = \mathrm{O}$，即模型是 Rc -完全同构的。　　证毕

基于以上定义，将 SDTR-OBAC 与经典 RBAC 进行对比分析。SDTR-OBAC 与 RBAC 中角色数量的关系可描述为 $|\mathrm{Rc}|_{\mathrm{RBAC}} = |\mathrm{O}| \times [1 + (|\mathrm{Rc}|_{\mathrm{SDTR-OBAC}} - 1) \times \mathrm{hindex}(\mathrm{Rc})]$，则对于完全同构的系统，由于 $\mathrm{hindex}(\mathrm{R}) = 1$，从而 $|\mathrm{R}|_{\mathrm{RBAC}} = |\mathrm{O}| \times [1 + (|\mathrm{R}|_{\mathrm{SDTR-OBAC}} - 1) \times 1] = |\mathrm{O}| \times |\mathrm{R}|_{\mathrm{SDTR-OBAC}}$，即经典 RBAC 中所需的角色数量是完全同构的 SDTR-OBAC 模型中角色数量的 $|\mathrm{O}|$ 倍；对于异构系统，由于 $\mathrm{hindex}(\mathrm{R}) = 1/|\mathrm{O}|, |\mathrm{O}| = 1$，从而 $|\mathrm{R}|_{\mathrm{RBAC}} = |\mathrm{O}| \times [1 + (|\mathrm{R}|_{\mathrm{SDTR-OBAC}} - 1) \times 1] = |\mathrm{R}|_{\mathrm{SDTR-OBAC}}$，即异构系统中 RBAC 和 SDTR-OBAC 所需角色数量相同。

因此，应用于同构系统中，与 RBAC 相比，在保持其灵活、易于管理等优点的同时，SDTR-OBAC 模型所需角色数量将明显减少，特别是在由多个具有相似业务功能的组织构成的大型分布式系统中，SDTR-OBAC 模型的优势是显而易见的。但当系统中所有组织均没有相似的业务功能时，由于引入了组织的概念且角色被划分成职能角色和任务角色，使用 SDTR-OBAC 将增加额外的管理负担。

5 实例分析

下面通过实例说明本文模型的合理性和有效性。假设有一公司 com，其包含 3 个子公司 com_1，com_2，com_3；6 类职能角色：总经理 fr_1，业务经理 fr_2，主管 fr_3，会计 fr_4，出纳 fr_5 及其他普通职员 fr_6；4 个任务角色：系统管理员 tr_1，普通管理员 tr_2，高级用户 tr_3，普通用户 tr_4；公司向所有人员提供了一个公司业务处理系统，包含 3 种类型资源：数据库类资源 DB, Web 服务类资源 WS，网站类资源 WB，相应类型的资源分别包括 db_{11}，db_{12}，db_{13}，ws_{21}，ws_{22}，ws_{23}，wb_{31}，wb_{32}，wb_{33}，wb_{34}；5 类操作：更新 u，下载 d，浏览 b，查询 q，调用 i。讨论该公司用户 li，wang，liu，zhang，zhao 进行授权和访问控制的过程。

该实例中，部分关键的元素，关系和函数定义如下：资源 Res={db_{11},db_{12},db_{13},ws_{21},ws_{22},ws_{23},wb_{31}, wb_{32},wb_{33},wb_{34}}；权限 P ={p_1,p_2,p_3,p_4,p_5,p_6,p_7,p_8, p_9, p_{10}} ={(u,DB),(u,WS),(u,WB),(d,WB),(b,WS), (b,WB),(q,DB), (q,WS),(q,WB),(i,WS)}；用户-角色关系 UOFR={(li,com,fr_1),(wang,com,fr_2),(liu,com_1, fr_3),(zhang,com_3,fr_6),(zhao, com_2,fr_5)}；角色-权限关系 OTRP={(com_1,tr_1, p_1),(com_3,tr_1, p_3),(com_2,tr_1, p_2), (com_1,tr_2, p_7),(com_3, tr_2, p_8),(com_2,tr_2, p_9),(com_2,tr_3, p_4),(com_3,tr_3, p_5),(com_3, tr_3, p_{10}),(com_2,tr_4, p_6)}；职能角色-任务角色关系：FRTR={(fr_1,tr_1),(fr_2,tr_2),(fr_3, tr_3),(fr_4,tr_4),(fr_5,tr_4),(fr_6,tr_4)；组织层次关系 OH= {(com,com_1),(com,com_2),(com,com_3)}；职能角色层次关系 FRH={(fr_1,fr_2),(fr_2,fr_3),(fr_3,fr_4),(fr_3,fr_5),(fr_3, fr_6)}；任务角色层次关系 TRH={(tr_1,tr_2),(tr_2,tr_3), (tr_3,tr_4)}；资源层次关系 ResH={(db_{11},db_{12})(db_{12}, db_{13}),(ws_{21},ws_{22}),(ws_{22},ws_{23}),(wb_{31},wb_{32}),(wb_{32},wb_{33}), (wb_{33},wb_{34})}；权限层次关系 PH = {(p_1, p_7),(p_2, p_8), (p_8, p_{10}),(p_{10}, p_5),(p_3, p_4),(p_4, p_9),(p_9, p_6)}；资源类型与组织关系 resorgs(DB)={com_1}，resorgs(WS)= {com_3}，resorgs(WB)={com_2}；职责分离约束 SoD = {({(fr_4,*),(fr_5,*)},2)}；势约束：cardinality ((fr_1,*)) = 1，cardinality((tr_1,*)) = 1。

该 5 个用户分别提出以下访问请求：q_1 = can_access(li, s_1, u, db_{13})，q_2 = can_access(wang, s_2, d, wb_{33})，q_3=can_access(liu, s_3, i, ws_{23})，q_4=can_access (zhang, s_4, i, ws_{21})，q_5 = can_access(zhao, s_5, b, wb_{32})，对这些访问请求进行判决。根据模型定义可分析得出 q_1 = true，q_2 = true，q_3 = false，q_4 = false，q_5 = true，即 li，wang，zhao 的访问合法，允许访问；liu 和 zhang 的请求非法，拒绝访问。

以上实例可以看出，本文提出的 SDTR-OBAC 模型是合理的，可行的，可有效应用于多个组织中用户请求访问特定资源的授权管理和访问控制。以上实例涉及 4 个具有相似业务需求的组织，每个组织包含 6 个职能角色，4 个任务角色和 3 类资源。为满足访问控制需求，应用本文所提模型时，以上实例仅需 10 个角色，10 个权限即可实现。然而，若采用经典 RBAC，所需的角色数为 24，权限总数为 34。可见，经典 RBAC 所需的角色数和权限数比 SDTR-OBAC 模型多。

6 结束语

授权与访问控制是继身份认证后兴起的又一重要信息安全技术。RBAC 以其灵活、便于管理和策略中立等优点成为解决授权与访问控制问题的研究热点并取得了很多研究成果，但仍存在传统角色设置单一使得适应性较差、下级角色的私有权限难以得到有效保护、易带来角色或权限冗余及对资源管理关注不够等问题。针对这些问题，本文提出支持资源管理的基于双层角色和组织的访问控制模型并进行形式化定义。将传统角色划分为职能角色和任务角色，提出基于角色区分的双层角色架构，提高模型的适应性，并对两种角色分别定义不同的继承策略，解决下级角色的私有权限难以得到有效保护问题；引入具有域思想的组织的概念并与双层角色相结合，解决由大型分布式系统中易存在的角色、权限冗余问题；对资源、资源操作和权限等进行分析，将授权管理和资源管理相结合，解决现有研究缺乏面向授权管理的资源管理的问题；将经典 RBAC 中权限的概念扩展为操作和资源类型构成的二元组，提高授权管理效率。从表达能力、复杂度两个方面分析了模型的特点，表明该模型不仅保留了 RBAC 的特点与优势，且比 RBAC 具有较低的复杂度和较高的效率和适应性。下一步的工作是对模型的安全性进行分析证明。

参 考 文 献

[1] ANSI. 2004. American national standard for information technology-role based access control[S]. ANSI INCITS 359, 2004.

[2] Gofman M I and Yang Ping. Effecient policy analysis for evolving administrative role based access control[J]. *International Journal of Software Informatics*, 2014, 8(1): 95–131.

[3] Liu Meng and Wang Xuan. Alternative representation of periodic constraint on role enabling in TRBAC and GTRBAC[J]. *Journal of Computational Information Systems*, 2013, 9(24): 9909–9918.

[4] Abdunabi R, Al-Lail M, Ray I, *et al*.. Specification, validation, and enforcement of a generalized spatio-temporal role-based access control model[J]. *IEEE Systems Journal*, 2013, 7(3): 501–515.

[5] Muhammad Asif-habib. Mutually exclusive permissions in RBAC[J]. *International Journal of Internet Technology and Secured Transactions*, 2012, 4: 207–220.

[6] Ma Li, Zhou Yan-jie, and Duan Wei. Extended RBAC model with task-constraint rules[C]. Proceedings of 8th Future Information Technology: Lecture Notes in Electrical Engineering, Gwangju, Korea, 2014, 276: 245–250.

[7] Zu Xiang-rong, Liu Lian-zhong, and Bai Yan. A role and task-based workflow dynamic authorization modeling and enforcement mechanism[C]. The 1st International Conference on Information Science and Engineering (ICISE2009), Nanjing, China, 2009: 1593–1596.

[8] Sohr K, Kuhlmann M, and Gogolla M. Comprehensive two-level analysis of role-based delegation and revocation policies with UML and OCL[J]. *Information and Software Technology*, 2012, 54(12): 1396–1417.

[9] Liu Xin-xin and Tang Shao-hua. Analysis of role-based trust management policy using description logics[J]. *Journal of Computational Information Systems*, 2012, 8(13): 5445–5452.

[10] Unal D and Caglayan M U. A formal role-based access control model for security policies in multi-domain mobile networks[J]. *Computer Networks*, 2013, 57(1): 330–350.

[11] Zhang Da-na, Ramamohanarao K, Zhang Rui, *et al*.. Efficient graph based approach to large scale role engineering[J]. *Transactions on Data Privacy*, 2014, 7(1): 1–26.

[12] Ranise S, Truong A, and Armando A. Scalable and precise automated analysis of administrative temporal role-based access control[C]. SACMAT'14, London, Ontario, Canada, 2014: 103–114.

[13] 崔鸿飞. ARBAC 权限泄漏分析及改进[D]. [硕士论文], 天津大学, 2012.

Cui Hong-fei. Analysis of permission leakage in ARBAC and improvement[D]. [Master dissertation], Tianjin University, 2012.

[14] 任志宇, 陈性元, 单棣斌. 基于双层角色映射的跨域授权管理模型[J]. 计算机应用, 2013, 33(9): 2511–2515.

Ren Zhi-yu, Chen Xing-yuan, and Shan Di-bin. Cross-domain authorization management model based on two-tier role mapping[J]. *Joumal of Computer Applications*, 2013, 33(9): 2511–2515.

[15] Zhang Zhi-xiong. Scalable role & organization based control and its administration[D]. [Ph.D. dissertation], George Mason University, 2008.

熊厚仁： 男，1986 年生，博士生，研究方向为授权、访问控制与资源管理.

陈性元： 男，1963 年生，教授，博士生导师，研究方向为网络信息安全.

张　斌： 男，1969 年生，教授，博士生导师，主要研究方向为网络信息安全.

杜学绘： 女，1968 年生，教授，博士生导师，主要研究方向为网络信息安全.