第 36 卷 第 3 期 计 算 机 工 程 2010 年 2 月
Vol.36 No.3 Computer Engineering February 2010
·安全技术· 文章编号：1000—3428(2010)03—0152—04 文献标识码：A 中图分类号：N945.12

基于 T-RBAC 的 PMI 模型

刘晓冰，白朝阳，王 霄，李忠凯
(大连理工大学 CIMS 中心，大连 116024)

摘 要：在工作流系统应用中，权限管理基础设施(PMI)模型存在数据冗余、动态适应性差的缺陷。针对该问题，提出一个基于任务-角色的访问控制的 PMI 模型。该模型通过增加任务规范角色证书与任务分配属性证书、PMI 任务管理器与策略库，将访问权限与任务关联。应用结果证明该模型能够对访问控制进行动态、灵活的管理，实现基于角色、任务、角色和任务这 3 种访问控制，为企业信息安全管理提供保障。
关键词：基于任务-角色的访问控制；上下文约束；权限管理基础设施

PMI Model Based on T-RBAC

LIU Xiao-bing, BAI Zhao-yang, WANG Xiao, LI Zhong-kai
(CIMS Center, Dalian University of Technology, Dalian 116024)

【Abstract】Aiming at the problems of Privilege Management Infrastructure(PMI) model application in the work flow system that data redundancy, weakly dynamic adaptability, this paper proposes a PMI model based on Task-Role Based Access Control(T-RBAC). The model can make the relationship between access right and task by adding task specification access certificate, task assignments access certificate and task management agent. Application result proves that the model can support information security management with three ways of access control in dynamic work flow systems, it consists of access control based on role, task and role and task.
【Key words】Task-Role Based Access Control(T-RBAC); context constrain; Privilege Management Infrastructure(PMI)

1 概述

权限管理基础设施(Privilege Management Infrastructure, PMI)是一种以 PKI 可信身份认证服务为基础，通过属性证书的形式实现授权管理的基础设施。由于 PMI 能够提供跨应用、跨系统、跨企业、跨安全域的用户属性管理和交互手段，使 PMI 模型自 2000 年提出以来，被广泛应用在电子政务、电子商务、银行金融等网络服务系统。

随着数据库、网络和分布式计算的发展，组织任务进一步自动化，与服务相关的信息进一步计算机化，使得以工作流管理为核心的系统成为网络服务系统的发展方向。在工作流系统中，PMI 在应用时暴露出一些缺陷，表现为：数据冗余，动态适应性差，没有达到真正的最小权限约束。

目前，在 PMI 中主要采用基于角色的访问控制(Role Based Access Control, RBAC)技术进行授权管理和访问控制，以角色为核心，直接将权限与角色相关，建立对象与操作、权限、角色、组织结构、系统结构的层次化关系的描述和管理框架。在任务驱动的工作流系统中，当数据流动时，数据处理的上下文环境变化导致执行操作的用户、用户的权限都动态改变。如果采用 RBAC 进行访问控制，那么就需要频繁地更换角色，导致系统数据冗余、动态适应性差；如果围绕角色设置其最小权限，那么系统负担就将大幅增大，降低系统运行效率。

本文在研究系统上下文约束集成模型和任务实例化映射关系的基础上，从约束、角色和任务出发建立基于任务-角色的访问控制(Task-Role Based Access Control, T-RBAC)模型，并将 T-RBAC 模型引入到 PMI 角色模型中。改进的 PMI 模型解决了上述问题，能够动态、灵活地进行访问控制管理。

2 相关研究工作

目前，关于 PMI 访问控制模型和策略一直是研究的热点。文献[1]研究 PMI 的安全架构，文献[2-3]研究 PMI 角色模型，文献[4]研究系统上下文约束模型，文献[5]研究 PMI 模型在工作流系统中的应用。

上述研究大多是从系统角度出发保护资源。工作流系统环境需要动态的访问控制和应用层面的安全约束，在权限控制时需要综合考虑任务和上下文。

在文献[1-3]中，PMI 访问控制没有涉及访问实例的动态权限管理；文献[4]提出基于动态上下文的访问控制模型，但没有在应用层面对上下文环境规范化，以及给出相应的层次化上下文环境约束模型；文献[5]建立了基于任务的访问控制，但没有给出如何在应用层面建立角色和任务间的映射关系，以及如何将角色-任务模型引入 PMI 角色模型，使其能够动态地进行访问控制。

在上述研究的基础上，本文从 3 个方面改进 PMI 模型：

(1)规范和建立上下文约束集成模型；

(2)细化访问控制过程中的显性、隐性权限以及相关实例，建立基于角色和任务映射关系的 T-RBAC 访问控制模型；

(3)改进 PMI 角色模型，增加任务管理器，并根据 T-RBAC 模型改进证书内容和相应管理机制。

基金项目：国家自然科学基金资助项目“面向 APS 的能力需求计划和能力平衡算法研究”(70772086)
作者简介：刘晓冰(1956－)，男，教授、博士生导师，主研方向：信息安全管理；白朝阳、王 霄、李忠凯，博士研究生
收稿日期：2009-06-17 **E-mail：**baizhaoyang2000@yahoo.com.cn

3 系统上下文约束集成模型

系统上下文约束包括用户约束、会话约束、角色约束、任务约束和权限约束等，它是一个非常复杂的约束集合，可通过分层法将其简化。通过上下文数据(Context Data, CD)，上下文数据获取机(Context Data Acquisition, CDA)和上下文条件(Context Condition, CC)的组合来进行定义。CD 是描述当前运行环境的数据。CDA 是获取 CD 的实际机制。CC 分为 2 类: 简单上下文条件(Simple Context Condition, SCC)和复杂上下文条件(Composite Context Condition, CCC)。

SCC 是一个布尔运算式，由 CD 组合而成，用来判定某个环境的有效性，CCC 由 SCC 和 CCC 组成。图 1 显示了由 CD, CDA, CC 组成的多级上下文约束集成模型。

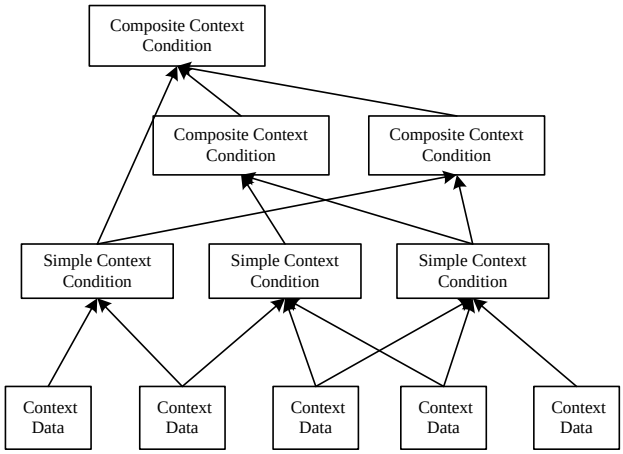

图 1　上下文约束集成模型

4 T-RBAC模型

定义 T-RBAC 模型为一个五元组(*U, R, T, C, P*)，其中，*U* 表示用户；*R* 表示角色；*T* 表示任务；*C* 表示上下文约束；*P* 表示权限，是由角色 *R*、任务 *T* 和约束 *C* 激活的权限。下文对该模型进行建模。

任务是一个工作的逻辑单位，它不可分割且必须完整执行。当用户运行工作流系统时，执行任务实例。本文定义了任务和任务实例 2 个概念，任务和任务实例通过映射函数相对应。

权限是用户对信息系统中的资源对象执行某种特定模式操作的操作许可，任务由一系列操作组合而成。本文将操作层看作透明，只考虑任务层。任务对资源的操作许可定义为显性权限(Explicit Permission, EP)，任务实例对资源的操作许可定义为隐性权限(Implicit Permission, IP)。

显性权限和隐性权限构成了权限集。EP 通过显性权限授予(Explicit Permission Assignment, EPA)赋予任务，IP 通过隐性权限授予(Implicit Permission Assignment, IPA)赋予任务实例，即通过任务与任务实例的映射关系以及 EPA 产生任务实例的权限，这样任务和任务实例就具备了相应权限。权限通过任务间接地赋予角色。

任务授予(Task Assignment, TA)将任务指派给角色，这样该角色就具备了该任务对应的权限。

任务实例授予(Task Instance Assignment, TIA)将任务实例指派给角色，这样该角色就具备了该任务实例具有的权限。若任务具备某权限，则该任务所映射的所有任务实例都具备该权限。如果一个任务被赋予某个角色，那么该任务的所有实例都可被赋予此角色。

在 T-RBAC 模型中，基于任务实例映射的 T-RBAC 模型如图 2 所示。在用户启动一个会话后，不直接激活对应的角色，而是采用拉动式的授权访问方式。当用户执行某个任务实例时，根据具体的上下文环境，构建上下文约束，生成任务实例的具体要求和权限约束。任务实例根据流程动态到达角色，权限随之赋予角色，当任务完成时，角色权限收回，角色在工作流中不赋予权限。这保证了角色在任务执行期间的权限赋予和任务非执行期间的权限撤销。在图 2 中，虚线表示在工作流系统运行过程中，权限的授予过程。每个任务的执行都被看作是主体使用相关访问权限访问客体的过程。在任务执行过程中，权限被消耗，当权限用完时，主体就不能再访问客体。

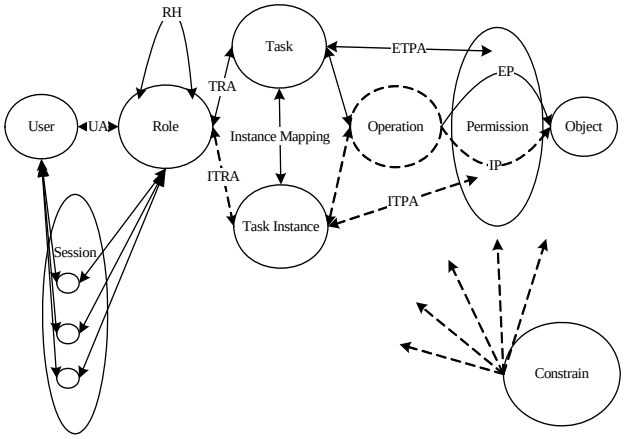

图 2　基于任务实例映射的 T-RBAC 模型

下文给出 T-RBAC 模型的形式化描述:

(1)*U*: 用户集合;

(2)*R*: 角色集合;

(3)*S*: 会话集合;

(4)*OB*(Object): 资源集合;

(5)*T*(Task): 任务集合。对于任务有 $T_i \to T_j$，表示 T_i 是 T_j 的直接前序任务; $T_i \to^* T_j$: 表示 T_i 是 T_j 的前序任务，可能是直接也可能是非直接;

$$\{T_1,T_2,\cdots,T_n\} \to T_i : activate(T_i) \Rightarrow$$
$$completed(T_1,T_2,\cdots,T_n) \vee activated(T_1,T_2,\cdots,T_n)$$

(6)*TI*(Task Instance): 任务实例集合;

(7)*T* 和 *TI* 之间的映射函数:

$$Mapping\ \varsigma : T \to 2^{TI}, \varsigma(a) \cap \varsigma(b) = \varnothing$$
$$\text{if } a \neq b \text{ and } a,b \in T$$

(8)OP(Operation): 操作集合;

(9)*UA*(User Assignment): 用户角色分配集合: $UA \subseteq U \times R$;

(10) $RH \subseteq R \times R$: 角色的偏序关系，称为角色层次关系或等级关系，记做“≤”，其中，$R_i \leqslant R_j$: R_i 和 R_j 是直接偏序关系; $R_i \leqslant^* R_j$: R_i 和 R_j 是偏序关系，可以是非直接; $r_2 \leqslant r_1 \forall T_i \in r_2 \Rightarrow r_1\ inherits\ \{T_i | T_i \in r_2\}$;

(11)*P*(*Permission*): 权限集合，$P = EP \cup IP$，$Permissions \to 2(OPs \times OBs)$;

(12)*EP*: 显性权限集合，$EP \subseteq OP(T) \times OB$;

(13)*IP*: 隐性权限集合，$IP \subseteq OP(TI) \times OB$;

(14)*EPA*: 显性权限授予集合，$EPA \subseteq T \times EP$;

(15)*IPA*: 隐性权限授予集合，由 EPA 产生的隐性权限授予集合，$IPA = \{(r_i,t_i) | [\![\exists (r_i,t) \in EPA]\!] \wedge t_i \in \varsigma(t)\}$;

(16)*PA*(Permission Assignment): 权限授予集合，$PA =$

$EPA \cup IPA$；

(17)TA：任务授予集合，$TA \subseteq T \times R$；

(18)TIA：任务实例授予集合，$TIA \subseteq TI \times R$；

(19)TTA：总体任务授予集合，$TTA = TA \cup TIA$；

(20)ContextConstrain：约束。

$ContextConstrain = \{cd, scc, ccc\}$

依据 RBAC96 模型中的函数定义，定义以下函数：

(1)函数 $user(S_i)$： $User: S \to U$，将每一个会话 S_i 映射到一个用户；

(2)函数 $role: S \to 2^k \in UA$，将各个会话 S_i 与一个角色集合连起来映射，其中，$role(s_i)\{r \mid \exists (r' \geqslant r)[(user(s_i), r' \in UA)]\}$，$U_{r \in role(s_i)}\{p \mid \exists (r' \geqslant r)[(p, r' \in PA)]\}$；

(3)函数 $permission: R \to 2^P$ 角色与权限的映射函数(同 RBAC96)；

(4)函数 $permission^*: R \to 2^P$ 由于角色的偏序关系而产生的权限(同 RBAC96)；

在 T-RBAC 模型中，满足上下文环境约束 CC 的可以执行的任务实例 $t_i(r_i)$：

$$t_i(r_i) = \{(t_i) \mid [\exists (r_i, t) \in EPA] \wedge t_i \in \varsigma(t) \wedge CC\}$$

执行任务实例 t_i 的角色 r_i 具有权限：

$$permission(r_i) = \{(CC, t_i) \mid [\exists (r_i, t) \in EPA] \wedge t_i \in \varsigma(t) \wedge CC\}$$

$$permission^*(r_i) = \{(CC, t_i) \mid (\exists r_i \geqslant r)[(r, t) \in EPA] \wedge t_i \in \varsigma(t) \wedge CC\}$$

5 PMI模型的改进和实现

5.1 PMI角色模型分析

PMI 角色模型能实现基于角色的访问控制，角色模型如图 3 所示。

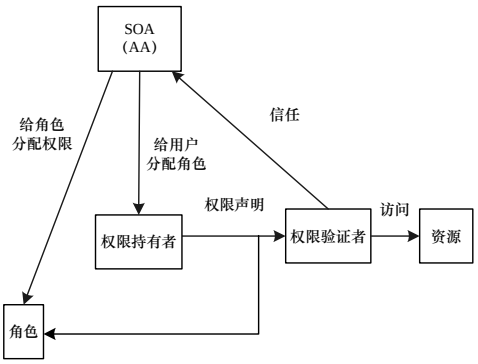

图 3 PMI 的角色模型

角色提供一种间接分配特权给个体的方式。PMI 角色模型分发角色分配证书给个体，通过证书中的角色属性使他们能够扮演一个或多个角色。明确的特权通过角色规范证书(Role Specification Certificates, RSC)赋给角色。角色分配证书(Role Assignments Certificate, RAC)可以是公钥证书，也可以是属性证书，但角色规范证书只能是属性证书。

在角色模型中，信任源点(Source of Authority, SOA)不将特权直接分配给特权持有者，而是建立若干角色将权限授予角色。特权持有者在申请特权时，SOA 通过角色分配证书为个体赋予角色身份，从而使个体获得角色所具有的特权集合。在角色模型中，SOA 是整个 PMI 系统的最终信任源和最高管理机构，SOA 保有角色规范证书，属性授权(Attribute Authority, AA)是授权管理体系的核心服务节点，SOA 授权 AA 管理一个部分或全部属性的权限。特权持有者只持有角色分配证书类似于一个指向特权的指针。在进行访问请求时，特权验证者根据特权持有者提供的角色分配证书，在本地的角色规范证书库中查找对应者；如果没有，则根据证书中的信息到 SOA 查找。

5.2 基于T-RBAC的PMI角色模型

根据 PMI 角色模型特点和要求，从以下 4 个方面对 PMI 角色模型进行改进，将 T-RBAC 引入到 PMI 角色模型中。

(1)确定角色规范证书和角色分配属性证书的内容。为了实现基于 T-RBAC 模型的授权，对角色规范证书的属性(attribute)域做了稍许修改。SOA(或 AA)通过角色规范属性证书为角色指定任务组或者分配权限，角色规范属性证书的持有者(holder)域是角色，attribute 域是角色或者一系列的任务组(如果是任务组，那么该证书要配合任务证书一起使用)；SOA(或 AA)通过角色分配属性证书为用户分配角色，角色分配属性证书的 holder 域是用户，attribute 域是一系列的分配给用户的角色。

(2)为了实现基于 T-RBAC 模型的授权，本文增加 2 类证书：任务组规范属性证书(Task Specification Attribute Certificate, TSAC)和任务组分配属性证书(Task Assignment Attribute Certificate, TAAC)。任务组规范属性证书负责描述任务组所拥有的权限，其 holder 域表示任务组，attribute 描述一系列分配给任务组的权限。任务组分配属性证书用来说明用户所在的任务组，其 holder 域表示用户，attribute 域表示用户所在组的信息。SOA(或 AA)给用户授权时可以选择是颁发角色分配属性证书还是 TAAC。

(3)增加任务管理器，负责对任务进行管理。通过任务管理器，生成授权策略库，SOA(AA)基于该授权策略库实现权限的授予。

(4)为权限验证者增加本地角色规范属性证书库和访问控制策略库。权限验证者在本地角色规范属性证书库查询用户是否具备访问资源的权限。若查询不到某角色的角色规范属性证书，则在轻型目录访问协议(Lightweight Directory Access Protocol, LDAP)目录中查询。从 LDAP 目录中查询到的角色规范属性证书作为副本存放在本地角色规范属性证书库中供以后使用，提高证书的查询效率。基于 T-RBAC 的 PMI 角色模型如图 4 所示。

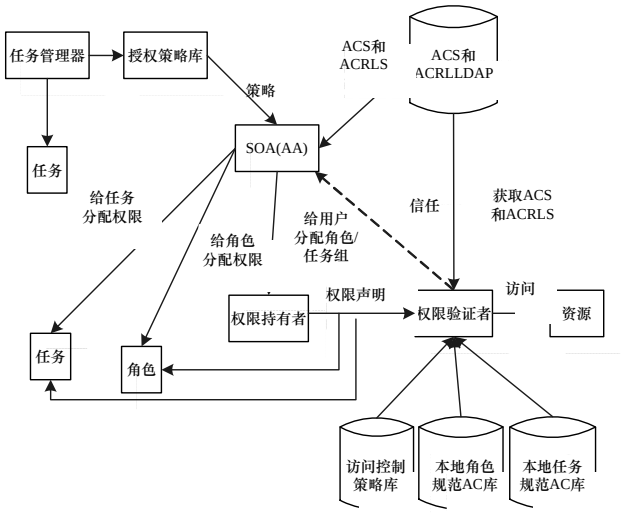

图 4 基于 T-RBAC 的 PMI 角色模型

5.3 访问控制过程

当用户向 SOA(AA)申请权限时，SOA(AA)根据情况决定为用户颁发角色分配属性证书或任务分配属性证书，颁发任何一种属性证书时都必须满足授权策略库定义的授权规则，颁发任务分配证书的前提是存在任务规范证书。

用户发出访问请求时，权限验证者实施访问控制的过程如下:

(1)根据流程运行到某个任务实例，权限验证者到 LDAP 目录中去查询用户的属性证书，得到用户拥有的角色集合和任务组。

(2)权限验证者在本地任务规范属性证书库查询权限。

根据用户拥有的角色查询其对应的角色规范属性证书，如果查询到的角色规范属性证书的 attribute 域的取值为权限，则直接获取权限，实现传统的基于角色的访问控制；如果查询到的角色规范属性证书的 attribute 域的取值为任务组，则继续查找该任务组的任务规范属性证书以获取其对应的权限，实现基于角色和任务的访问控制。

根据任务组查询任务规范属性证书，确定相应的权限，实现基于任务的访问控制。

若查询任务规范属性证书，则在 LDAP 目录中查询。从 LDAP 目录中查询到的任务规范属性证书作为副本存放在本地任务规范属性证书库中供以后使用。这样可以提高属性证书的查询效率。

若用户拥有声称的权限，且满足访问控制策略库中的访问控制规则，则权限验证者允许用户执行访问，否则访问请求被拒绝。

6 应用实例

基于 T-RBAC 的 PMI 模型已被应用于东北特钢集团物资采购审批系统。该分布系统运行于大连、抚顺、齐齐哈尔 3 地，通过网络协作完成原材料的计划、预算、采购、实施等工作的审批。整个审批过程涉及多个动态因素，如市场价格波动、合同执行情况、生产能力状况、各分厂物流、库存、原材料战略重要度等，同时审批过程具有层次性和动态性特征。

整个系统架构如图 5 所示。在图 5 中，Context 获取模块从资源层得到 CD，根据上下文约束集成模型，Context Information 模块生成由 CD, SCC, CCC 组成的上下文约束定义。

任务执行节点(Task Executing Point, TEP)根据 T-RBAC 模型，TEP 主要功能是：将任务实例化；根据生成的上下文约束、任务实例的具体要求和权限约束，将任务实例赋予角色。角色决策节点(Role Decision Point, RDP)基于图 4 改进的 PMI 角色模型，RDP 节点根据上下文约束、安全策略定义，实现动态访问控制。

当用户执行任务申请权限时，首先 TEP 将任务实例化，建立任务与任务实例的映射关系，RDP 根据任务实例信息，从策略服务器和 Context Information 服务器得到上下文约束，任务实例相关约束，RDP 根据身份认证信息、任务请求和相关约束执行 5.3 节阐述的访问控制，TEP 从 RDP 得到任务实例的角色控制信息，将任务实例赋予角色，完成整个授权过程，从而在动态的工作流系统中实现基于角色、任务、角色和任务 3 种访问控制管理。

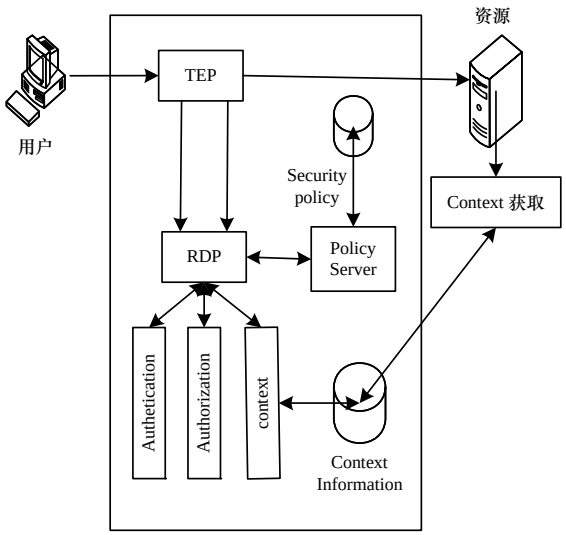

图 5 系统架构

7 结束语

对于动态的任务驱动的工作流系统，基于角色访问控制的 PMI 模型在应用时暴露的缺陷，本文提出一个基于 T-RBAC 的 PMI 模型。该模型在 RBAC 模型的基础上，加入任务层，建立 T-RBAC 模型，将其引入 PMI 模型。

通过增加 PMI 任务规范角色证书和任务分配属性证书，增加 PMI 任务管理器和策略库，实现从角色、任务和约束 3 个角度对动态的工作流系统进行授权和访问控制，改进了原有模型的缺陷。

参考文献

[1] Kandala S, Sandhu R. Secure Role Based Workflow Models[C]// Proc. of the 15th Annual Working Conference on Database and Application Security. Ontario, Canada: [s. n.], 2002: 45-58.

[2] Sejong O, Seog P. Task-role Based Access Control Model[J]. Information Systems, 2003, 28(6): 533-562.

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

[4] Kapssalis V, Hadellis L, Karelis D, et al. A Dynamic Context-aware Access Control Architecture for E-services[J]. Computers Security, 2006, 25(1): 507-521.

[5] Lopez J, Mana A, Ortega J J, et al. Integrating PMI Services in CORBA Applications[J]. Computer Standards Interfaces, 2003, 25(3): 391-409.

编辑 陆燕菲