计算机科学2002Vol.29№.2

一种动态角色委托代理授权模型

A RBAC Dynamically Delegating Grant Model

王小明[1,2] 赵宗涛[1,3] 冯德民[2]

(西北大学计算机系 西安710066)[1](陕西师范大学计算机科学学院 西安710062)[2]

(第二炮兵工程学院计算机与指挥自动化系 西安710025)[3]

Abstract RBAC(Role-Based Access Control) has been the most promising access control policy that is generally acknowledged currently, and becomes one of hot research topics in the area of information. The role grant management in traditional RBAC is only done by system(or security) administrator, it is static. Under some special circumstances, such as medical emergency treatment and fireproof, in current RBAC models, a serious problem exists: when the system administrator or the role that is being required is just absent, the other roles in the system have no ways to legally obtain a qualification so as to temporarily take the place of the absent role. In this paper, With a RBAC dynamic delegating grant (RBAC-DDG) mechanism being introduced, the new role-role relations and the novel grant model, RBAC-DDG, as well as its dynamic grant protocol are proposed, The above problem can be addressed better using this model. On the basis of an abstracted medical emergency system, the application scope of RBAC-DDG is discussed, and on RBAC-DDG, some future problems to be further studied are also given.

Keywords RBAC, RBAC-DDG, Role delegating agent, Protocol

1 引言

近年来,随着计算机信息系统安全问题愈来愈重要,对数据存取控制策略的研究已成为当前信息系统领域研究的热点问题之一。目前,存取控制主要有三种:自主存取控制(DAC, Discretionary Access Control),强制存取控制(MAC, Mandatory Access Control)和基于角色的存取控制(RBAC, Role-Based Access Control)[1~4]。在DAC方法中,同一用户对不同的数据对象有不同的存取权限;不同的用户对同一存取对象有不同的存取权限;用户可将自己拥有的存取权限不受任何限制地授予其他用户。因此DAC十分灵活,但安全性不强,授权管理复杂(用户,权限,存取对象)[1~3]。在MAC中,每一个存取对象被标以相应的密级,每一个用户也被授予某一个级别的访问许可证。只有持有许可证的用户才可以存取相应密级的对象,用户许可证和对象密级均由系统管理员管理,要求用户-权限关系具有良好的层次结构,是一种比较严格的存取控制方法,灵活性差[1~3]。

由于DAC和MAC的上述缺点,近年来,西方国家提出了基于角色的存取控制(RBAC, Role-Based Access Control)。在RBAC中,由系统(安全)管理员将一组权限赋给角色,角色分配给用户。一个用户可拥有多个角色,一个角色可授权给多个用户;一个角色可包含多个权限,一个权限可被多个角色包含。用户通过角色享有权限,它不直接与权限相关联;权限对存取对象的操作是通过活跃角色实现的。用户与角色,角色与权限,角色与存取对象之间的关系均为多对多关系(m:n)。角色之间也可能存在继承关系,即上级角色可继承下级角色的部分或全部权限,角色间可继承的权限称为它们的公共(public)权限,角色还可拥有私有(private)权限;角色继承关系形成了角色层次结构[3,4,6]。用户-角色关系的语义表达能力很强,它不仅能够表达复杂的社会组织机构中的"职-责"关系,而且能够降低存取控制管理的复杂度,已成为当前公认的最有发展潜力的存取控制策略[1~4,6]。但是,在特殊情况(例如医疗急救,火灾防护等)下,目前广泛使用的RBAC模型(如NIST RBAC和RBAC96[1~3])普遍存在一个严重问题:当系统管理员和某角色"缺席"时,"在场"的另一角色无法合法代理缺席角色[2,7],从而使紧急任务无法及时进行。本文提出了一种动态角色委托代理授权模型RBAC-DDG(RBAC Dynamic Delegation Grand),较好地解决了这个问题。

2 RBAC-DDG模型

为了形式化描述RBAC-DDG模型,首先给出下列基本概念。其中,以大写字母开头的字符串代表对象名称,大写字母(串)代表集合,小写字母(串)代表集合元素,"∧"符号代表逻辑"与"。

用户(U)是被许可进入系统并对系统中的数据对象进行存取的主体,它具有系统可验证的特征。存取对象(O)是系统中能够被合法用户进行存取的实体(entity)。权限(P)是施加在存取对象上的最小操作算子。例如数据库系统中的插入(insert),更新(update),查询(select)等都是操作权限。角色(R)是一组权限的集合。会话(S)是用户的一个活跃进程,它代表用户与系统交互。角色休眠是指用户 u_i 拥有的角色 r_j 处于不可用状态,用 $^*r_j$ 表示。角色活跃是指用户 u_i 在其会话 s_i 执行期间使其角色 r_k 处于可用状态,用 $^*r_j$ 表示。

2.1 角色-角色关系及授权规则

定义1(角色包含关系 Ω) 给定两个不同角色 r_i 和 r_j,若 r_j 中的任意权限 p_m 都包含在 r_i 中,则称 r_i 包含 r_j,用 $r_i \Omega r_j$ 表示。即 $r_i \Omega r_j \Rightarrow \forall\, p_m(p_m \in r_j \rightarrow p_m \in r_i)$

定理1 关系 Ω 是偏序关系。

王小明 博士研究生,副教授,主要研究领域为数据库与知识库,信息系统。赵宗涛 教授,博士生导师,主要研究领域为数据库与知识库,决策支持系统。冯德民 教授,主要研究领域为软件算法与理论,操作系统。

证明：(1)自反性。对任意 r_i，$r_i\Omega r_i$ 显然成立。

(2)反对称性(用反证法)。假设关系 Ω 不是反对称的，即存在两个不同的角色 r_i，r_j，若 $r_i\Omega r_j$，并且 $r_i\Omega r$，成立，则由定义1得 $r_i\subseteq r_j$，并且 $r_j\subseteq r_i$，根据集合性质得 $r_i=r_j$，与假设矛盾。因此，关系 Ω 是反对称的。

(3)传递性。对任意三个不同的角色 r_i，r_j，r_k，若 $r_i\Omega r_j$，并且 $r_j\Omega r_k$，则由定义1得 $r_i\subseteq r_j$，并且 $r_j\subseteq r_k$，由集合包含关系得 $r_i\subseteq r_k$，于是 $r_i\Omega r_k$ 成立。

定义2(角色相交关系 ξ) 给定两个角色 r_i 和 r_j，若 r_i 中至少存在一个权限 p_m，它也被包含在 r_j 中，则称 r_i 和 r_j 相交，用 $r_i\xi r_j$ 表示，即 $r_i\xi r_j\Rightarrow\exists\ p_m(p_m\in r_i\wedge p_m\in r_j)$。

定义3(角色独立关系 Φ) 给定两个角色 r_i 和 r_j，若 r_i 中的任意权限 p_m 都不包含在 r_j 中，则称 r_i 与 r_j 独立，用 $r_i\Phi r_j$ 表示，即 $r_i\Phi r_j\Rightarrow\forall\ p_m(p_m\in r_i\rightarrow p_m\notin r_j)$。

定义4(角色互斥关系 Γ) 给定两个角色 r_i 和 r_j，若 r_i 和 r_j 授予同一用户或同时处于活跃状态时会造成不合法的用户存取，则称 r_i 与 r_j 互斥。用 $r_i\Gamma r_j$ 表示，即 $r_i\Gamma r_j\Rightarrow(r_i\in u_k\wedge r_j\in u_k)\vee((^*r_i\wedge{}^*r_j)\rightarrow\text{false})$。

规则1 授权给用户 u_i 的角色 r_j 不允许同 u_i 当前拥有的活跃角色互斥。

规则2 将要激活的用户 u_i 的角色 r_j 不允许同 u_i 当前拥有的活跃角色互斥。

定义5(角色协同关系 φ) 给定两个角色 r_i 和 r_j，若 r_i 和 r_j 通过彼此合作才能完成某一用户操作，则称 r_i 与 r_j 协同，用 $r_i\varphi r_j$ 表示，即 $r_i\varphi r_j\Rightarrow{}^*r_i\wedge{}^*r_j\rightarrow\text{true}$，若 n 个角色协同，则表示为 $r_1\varphi r_2\varphi\cdots\varphi r_n$ 如银行金库的多个管理员角色之间存在协同关系。

规则3 具有协同关系的角色必须作为一个整体授权给用户。

定义6(角色继承关系 Ψ) 给定两个角色 r_i 和 r_j，若 r_j 中的部分或全部权限是对 r_i 的权限继承，则称 r_i 与 r_j 之间存在继承关系。其中，r_i 为父角色，r_j 为子角色。父、子角色之间可继承的权限称为它们的公共权限(public)，父、子角色也可以拥有各自的私有权限(private)，用 $r_i\Psi r_j$ 表示，即

$r_i\Psi r_j\Rightarrow\exists\ p_m(p_m\in r_j\wedge(\text{TYPE}(p_m)=\text{public}))\rightarrow\text{MANIPULATE}(r_i,p_m)$。

其中，TYPE 为类型判断函数，MANIPULATE(r_i，p_m)表示 r_i 可操纵(控制)p_m。

角色继承关系也允许多重继承，从而形成灵活多样的角色层次结构，它能够实现社会组织机构中普遍存在的复杂的层级结构映象。

定理2 关系 Ψ 是偏序关系。

证明：(略)

规则4 具有继承关系的角色，当父角色被激活、休眠或撤消时，子角色被自动激活、休眠或撤消。

由以上定义及定理很容易得出以下定理：

定理3 授予用户 u_i 的角色是安全的，当且仅当 u_i 的角色集中不存在互斥角色关系。

定理1和定理2保证了 RBAC-DDG 授权的非循环性，定理3保证了 RBAC-DDG 授权的职责分离性[1,2]。

定义7 RBAC-DDG 模型是一个11元组(U，R，P，S，ϑ，η，$\leqslant$，$F_{S\rightarrow U}$，$F_{S\rightarrow 2^R}$，CONS，agent)。

其中，U，R，P，S 分别是用户，角色，权限，会话集合。$\vartheta\subseteq P\times R$ 是权限空间到角色空间的多对多赋值关系。$\eta\subseteq U\times R$ 是用户空间到角色空间的多对多赋值关系。$\leqslant\subseteq R\times R$ 是角色层次偏序关系。$F_{S\rightarrow U}$是会话空间到用户空间的函数映射：$S\rightarrow U$。$F_{S\rightarrow 2^R}$是会话空间到角色子集空间的映射：

$F_{S\rightarrow 2^R}(S_i)\subseteq\{r\mid(\exists\ r_m(r_m\leqslant r\wedge(F_{S\rightarrow U}(S_i),r_m)\in\eta))\}$ 使得 s_i 有权限集合 $\bigcup_{r\in F_{S\rightarrow 2^R}(S_i)}\{p\mid(\exists\ r_k(r_k\leqslant r\wedge(p,r)\in\vartheta))\}$。

CONS 是授权约束集合，包括用户-角色关系，角色-角色关系，角色-权限关系和授权规则及其它约束关系。agent 是角色委托授权代理机构。

2.2 RBAC-DDG 基本架构

RBAC-DDG 模型的基本架构是：由角色委托代理申请者(Applicant)、角色委托代理协调者(delegator)、角色委托代理接受者(reciever)、角色委托代理协议(protocol)和数字证书(certificate)构成一个角色委托代理授权机构(agent)，如图1所示。

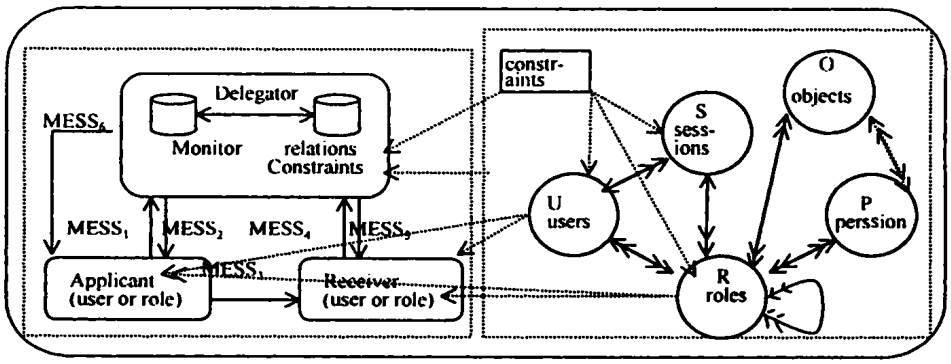

图1 角色委托代理授权机构(agent)架构

当某用户(或某角色)需要进行角色委托代理时，Applicant 向 delegator 发送角色委托代理申请消息 $MESS_1$，请求将某用户的某角色(或角色包含的权限)授予另一用户(或角色)，$MESS_1$ 发送之前须使用 Applicant 和 delegator 之间的共享密钥 $K_{a\text{-}d}$ 进行加密。delegator 收到 $MESS_1$ 之后，根据数字证书、用户-角色关系、角色-权限关系、角色-角色关系和授权规则及其它约束验证其合法性和可行性。如果 $MESS_1$ 合法并且可行，则发送给 Applicant 一个申请被批准的消息 $MESS_2$，$MESS_2$ 发送之前须使用 delegator 的私钥 K_d 加密，再用 $K_{a\text{-}d}$ 加密；否则发送一个申请被拒绝的消息(REJ_1)。Applicant 收到 $MESS_2$ 之后，向 receiver 发送一个角色委托代理执行消息 $MESS_3$。reciever 收到 $MESS_3$ 之后，向 delegator 发

送一个请求验证 $MESS_3$ 的消息 $MESS_4$。delegator 收到 $MESS_4$ 并对 $MESS_3$ 的真假进行验证后，向 receiver 发送一个确认消息 $MESS_5$，并向 applicant 发送一个 receiver 已收到 $MESS_3$ 的消息 $MESS_6$。Receiver 收到 $MESS_5$ 之后，开始角色委托代理执行（$MESS_3$，$MESS_4$，$MESS_5$，$MESS_6$ 均需加密发送）。至此，agent 完成了一次动态角色委托代理授权过程。

2.3 动态角色委托代理授权协议

根据第2.2节所讨论的 RBAC-DDG 模型角色委托代理授权的基本思想，设计下列 RBAC-DDG 授权协议。其中，K_{i-j} 为 i 和 j 之间的共享密钥，K_i 为 i 的私有密钥，certificate 为数字证书，包括接受者标识号，需要代理的角色名，限制条件等。

①Applicant SENDS $MESS_1$ to delegator://发送请求角色代理消息，//
$K_{a-d}\{MESS_1(ApplicantID, r_j, u_i, condition)\}$ //加密发送//

②delegator RECEIVES $MESS_1$ from Applicant: //接收请求代理消息并处理//
VERIFIES the legality of $MESS_1$//验证 $MESS_1$ 的合法性//
JUDGES the possibility of delegation in terms of CONS //根据 CONS 中的约束评判请求代理的可能性//
IF $MESS_1$ is true and possible THEN SENDS aproval $MESS_2$ to Applicant:
$K_{a-d}\{K_d\{(certificate)\}, K_{a-r}\}$ //加密发送请求批准消息//
ELSE SENDS K_{a-d}{'reject message'} to Applicant //发送请求被拒绝消息//

③Applicant SENDS $MESS_3$ to receiver: //发送代理角色执行消息//
$K_{a-r}\{certificate\}$ //加密发送//

④receiver RECEIVES $MESS_3$ from Applicant: //接收要求执行角色代理消息//
SENDS $MESS_4$ to delegator: //发送请求验证 $MESS_3$ 消息//
$K_{r-d}\{receiverID, K_d\{certificate\}\}$ //加密发送//

⑤delegator RECEIVES $MESS_4$ from receiver: //接收验证请求并处理//
CONFIRMS $MESS_3$
SENDS $MESS_5$ to receiver:
$K_{r-d}\{certificate\}$ //加密发送确认消息//
SENDS $MESS_6$ to applicant:
$K_{a-d}\{MESS_6(DelegatorID, 'Receiver\ see\ MESS_3')\}$ //发送 Receiver 收到了 $MESS_3$ 的消息//
□Receiver EXECUTE r //Receiver 执行代理角色//
END.

3 RBAC-DDG 模型实例

医疗急救系统通常由三个工作组组成（简化模型）：医生组，护士组，药剂师组。其中，医生工作组有三种角色：专家（specialist）、主治医师（resident）、见习医生（intern），并构成一个具有角色继承关系的层次结构；护士工作组有二种角色：护士长（chiefnurse）、护士（nurse），并构成一个具有角色继承关系的层次结构；假设药剂师工作组只有一个角色：药剂师（pharmacist），并约定专家继承主治医师和见习医生的权限，主治医师继承见习医生的权限；护士长继承护士的权限。医生工作组与护士工作组和药剂师工作组之间分别是多对多关系，护士工作组和药剂师工作组之间也是多对多关系，如图2所示，并规定在紧急情况下，医师可以代理药剂师角色。现假设有一个需要急救的病人，可是没有一个药剂师在岗，专家（$specialist_1$）为了能够及时对病人进行治疗，向 agent 的 delegator 发送请求允许见习医生（intern）在2001年9月25日晚7点至晚10点代理药剂师（pharmacist）角色委托代理授权消息 $MESS_1$：

K_{a-d}{specialist, pharm., intern, [2001/9/25: 18: 00—22: 00]}

遵循第2.3节讨论的动态角色委托代理授权协议，即可完成急救任务。医疗急救系统是 RBAC-DDG 模型应用的一个典型实例，与其类似的问题均可使用 RBAC-DDG 模型得到良好的解决。

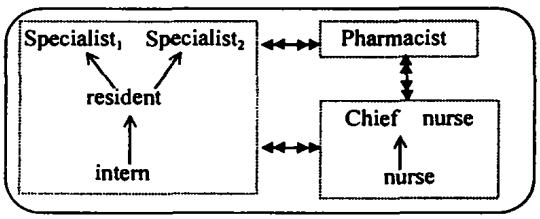

图2 医疗急救系统角色关系

结束语 为了使 RBAC 支持在某些角色（用户）"缺席"情况下，其它"在场"角色（用户）通过合法途径暂时代理缺席角色（或用户），以使特殊（紧急）任务及时进行，我们在 RBAC 模型中引入了动态角色委托代理授权机制（agent），扩展了 RBAC 模型，提出了一种动态角色委托代理授权模型（RBAC-DDG），给出了新的角色关系和授权规则及动态角色委托代理授权协议，并以抽象的医疗急救系统为原型，讨论了 RBAC-DDG 模型的适用性。初步的实验结果表明，RBAC-DDG 适用于 MIS，OA，DSS 等系统的前端用户的动态角色存取控制授权管理，在理论上也适用于系统软件（如 DBMS，OS 等）中基于角色的存取控制。分布式环境下的 RBAC-DDG 模型、高效的角色委托代理请求验证方法和授权规则知识库组织等问题有待进一步研究。

参 考 文 献

1 Department of Defence (USA). Department of Defense Trusted Computer system evaluation criteria. DoD 5200-78-STD, DoD, 1985

2 Sandhu R, Ferraiolo D, Kuhn R. The NIST model for role-based access control: towards a unified standard. In: Proc. of 5th ACM Workshop on Role-Based Access Control, ACM, Berlin, Germany, July, 2000

3 Osborn S, Sandhu R, Munawer M. Configuring role-based access control to enforce mandatory and discretionary access control policies. ACM Transactions on Information and System Security, 2000, 3(2)

4 Sandhu R, et al. Role-baseed access control model. IEEE Computer, 1996, 29(2)

5 Sandhu R. Role activation hierarchies. In: Proc. of 3rd ACM Workshop on Role-Based Access Control, ACM, Fairfax, Oct. 1998

6 Sandhu R, et al. Role-based access control models. IEEE Computer, 1996, 29(2): 38～47

7 Ahn G J, Sandhu R. The RSL99 language for role-based separation od duty constraints. In: proc. of 4th ACM Workshop on Role-Based Access Control, Fairfax, VA, Oct. 1999. 43～54

8 Sandhu R, Bhamidipati V, Munawer Q. The ARBAC97 model for role-based administration of roles. ACM Transactions on Information and system Security, 1999, 2(1): 105～135

9 Schneier B. Applied Cryptography-protocols, algorithims and source code in C. John Wiley & Sons, Inc. Second Edition, 1996