# 基于行为的访问控制模型及其行为管理

李凤华[1,2],王 巍[1],马建峰[1],梁晓艳[1]
(1. 西安电子科技大学计算机网络与信息安全教育部重点实验室,陕西西安 710071;
2. 北京电子科技学院研究生处,北京 100070)

**摘 要**: 访问控制模型是对信息资源进行授权决策的重要方法之一. 首先给出了环境的定义,结合角色、时态和环境的概念,给出了行为的定义. 在不同的信息系统中,行为可以用来综合角色、时态状态和环境状态的相关安全信息. 然后介绍了行为、时态状态和环境状态的层次结构,提出了基于行为的访问控制模型 ABAC(Action-Based Access Control Model),并讨论了在 ABAC 模型中角色、时态状态和环境状态之间的相互关系. 在此基础上,通过引入受限的时态状态和环境状态,给出了管理行为的定义和 ABAC 管理模型的结构;描述了 ABAC 管理模型下用户-管理行为、管理行为-管理权限的控制关系,使用 Z-符号形式化地描述了行为状态管理中使用的管理函数 AddAction、ModifyAction 和 DeleteAction,以及和 ABAC 管理模型相关的管理方法. 与已有其他模型相比,ABAC 模型更加适用于解决网络环境下支持移动计算的信息系统中的访问控制问题.

**关键词**: 访问控制; 行为; 环境状态; 时态状态; 管理行为
**中图分类号**: TP309.2 **文献标识码**: A **文章编号**: 0372-2112 (2008) 10-1881-10

# Action-Based Access Control Model and Administration of Actions

LI Feng-hua[1,2], WANG Wei[1], MA Jian-feng[1], LIANG Xiao-yan[1]
(1. *Key Laboratory of Computer Networks and Information Security (Ministry of Education), Xidian University, Xi'an, Shaanxi* 710071, *China*;
2. *Graduate School, Beijing Electronic Science and Technology Institute, Beijing* 100070, *China*)

Abstract: Access control is one of the powerful and generalized approaches of authorization decisions on information resources. Firstly, the environmental state is introduced and the term "action" is defined based on roles, temporal states and environmental states. Actions can be used to capture security-relevant aspects of roles, temporal states and environmental states in different information systems. Then, the action hierarchy, temporal hierarchy, environmental hierarchy and Action-Based Access Control (ABAC) model are presented. And the relationship among roles, temporal states and environmental states are analyzed. By introducing the limited temporal states and environmental states, the administrative action and administrative model for ABAC are described. The controlling relations of user-administrative action and administrative action-administrative permission are proposed. By Z-notation, the functions of AddAction, ModifyAction and DeleteAction are introduced. Moreover, the related methods for ABAC administrative model are presented. Compared with the existing models, the ABAC model can solve the problem of access control in information systems with mobile computation.

Key words: access control; action; environmental state; temporal state; administrative action

## 1 引言

随着网络化、信息化的高速发展,移动计算在电子政务等网络环境下大型信息系统中的应用日益普及,信息资源的安全访问问题日益突出,访问控制是对信息资源进行保护的重要措施之一. 访问控制的主要任务是保证信息资源不被非法使用和访问,其规定了主体对客体访问的限制,并在身份识别的基础上,根据身份对提出信息资源访问的请求加以控制.

R Sandhu 等人于 1996 年提出了著名的 RBAC96 模型,将传统的 RBAC 模型根据不同需要拆分成 4 种嵌套的模型并给出形式化定义,极大地提高了系统灵活性和可用性[1]. 1997 年他们更进一步提出了一种 RBAC 管理模型 ARBAC97,实现了在 RBAC 模型基础上的角色管理[2]. 2001 年,D F Ferraiolo、R Sandhu 等人制定了一个 RBAC 模型的美国国家标准草案[3]. 但在很多情况下,

收稿日期:2008-04-29;修回日期:2008-05-30
基金项目:国家 863 高技术研究发展计划(No. 2007AA01Z429, No. 2007AA01Z472, No. 2007AA01Z482);国家自然科学基金(No. 60633020, No. 60573036, No. 60702059);陕西省"13115"科技创新工程重大科技专项(No. 2007ZDKG-56)

单纯的基于角色的访问控制并不能完全适用于网络环境下支持移动计算的信息系统的管理. 文献[4]提出了一种应用于 Web 环境的访问控制模型 GenericWA-RBAC,文献[5]提出了一种应用于 VPN 环境的访问控制模型 DERBAC,文献[6]使用模糊关系对 RBAC 进行了改进,但这 3 个文献也没有考虑移动环境和时态对 RBAC 模型的影响.

文献[7]提出了时态访问控制模型(TRBAC,Temporal RBAC),但其没有考虑对用户-角色分配和角色-权限分配时的时态因素. J B D Joshi 等人考虑了更通用的时态约束的访问控制模型(GTRBAC,Generalized Temporal RBAC)[8,9],但 GTRBAC 没有考虑移动计算环境对访问控制的影响. 文献[10]通过将时态约束扩展到用户和权限两个方面提出了改进的访问控制模型.

为了控制在普适计算环境下对私有信息和资源的访问,Covington 等人通过引入环境角色提出了一种 GRBAC 模型[11]. 其中,每个分配的权限都与一个环境角色集合相关联,而环境角色根据不同的环境条件而变化. I Ray 和 L Yu 提出了一种基于位置的访问控制模型[12,13],但是此模型只讨论了 RBAC 模型中不同组件与位置的关系,而没有考虑到时态和环境对访问控制的影响.

GEO-RBAC 模型[14]使用空间实体来对客体、用户位置和有地理边界的角色进行建模,且此模型支持层次化、权限建模和职责隔离等. 张宏等人提出的 SG-RBAC 模型,通过引入空间角色的概念将空间上下文集成到角色中,可根据用户的当前位置来判断会话中哪些角色是有效的,并为受限的 SG-RBAC 模型确定空间职责隔离限制、基于位置的基数限制和基于位置的时序限制[15]. X Cui 等人在移动协作系统中通过增加身份限制和时空限制提出了一种 Ex-RBAC 模型[16]. 但文献[8～16]都没有考虑移动计算环境对访问控制的影响. 同时,文献[17～21]只考虑了管理角色的功能实现问题,但都没有考虑如何选择管理角色. 这可能对整个系统带来安全威胁.

虽然部分已有模型考虑了与访问控制相关的时态因素和位置因素,但现有的模型都没有对移动计算下角色所处环境(各种客观因素组成的环境要素,如场所物理位置、网络位置、逻辑位置、硬件平台、软件平台等)对访问控制的影响进行详细分析. 本文首先指出环境对访问控制的重要性,给出了"行为(Action)"的概念,并在此基础上提出了一种基于行为的访问控制模型 ABAC(Action-Based Access Control). 本文给出了管理行为的定义和 ABAC 管理模型的结构,描述了 ABAC 管理模型下的用户-管理行为和管理行为-管理权限的控制关系,形式化地描述了行为状态管理中使用的管理函数,同时给出了 ABAC 的相关管理方法.

## 2 基于行为的访问控制模型

本节首先综合角色、时态和环境的概念,给出"行为"的定义. 然后对"行为"进行形式化描述,并结合用户、角色及其层次结构、权限、时态和环境的概念,给出基于行为的访问控制模型.

### 2.1 相关概念

访问控制模型中涉及到以下几个概念:

用户:人或者自治代理,用户的集合记为 $U$;

角色:实现某种功能所需权限集合的描述,其与用户是多对多的关系,角色的集合记为 $R$;

会话:将用户与激活角色(集合)对应的映射. 一个用户可以有多个会话,但一个会话只能从属于一个用户;一个会话可以对应多个角色. 会话的集合记为 $S$;

权限:系统中对象的访问模式,权限的集合记为 $P$;

时态:时间约束集合,时态的集合记为 $T$.

除了以上提到的概念之外,在移动计算或分布式计算环境下,角色访问系统时的位置和操作平台等环境信息会影响到角色访问系统的权限. 当一个角色所处的位置不同时,其所能够得到的权限可能是不同的. 如某个角色是跨国公司中分公司主管,当其处于分公司内部时可以享有主管的权限,而当其在总公司时享有的权限可能和一般职员相同. 此外,其出差时享有的权限可能比在分公司内部享有的权限低,同时其权限又高于一般职员. 同样,若此角色访问信息资源时使用的操作平台不同,其权限也可能不同. 如使用公用计算机时其只能享有最低的权限、当使用系统内部计算机时其可以访问公司向内部人员公开的信息,当使用系统内部专用机时其可以访问公司机密信息等. 此外,使用不同的软件也可能会影响到角色享有的权限. 如使用公用软件时只能访问一些公开资源,而使用专用软件时可以访问一些机密资源. 所以在对访问控制机制进行建模时有必要引入"环境"因素. 下面介绍"环境"的概念.

**定义 1** 环境指用户访问系统时的客观因素,如位置(场所物理位置、网络位置、逻辑位置等)、平台(硬件平台、软件平台、密码系统等)和其他与访问控制相关的外部客观信息等. 系统可使用与安全相关的环境信息来限制对系统资源的访问. 环境状态对用户在何种外部客观因素下的权限进行约束,将环境状态的集合记为 $E$.

**定义 2** 对于相同的时态状态,若环境 $e_i$、$e_j$ 是 $E$ 中的元素,角色 $r$ 在环境 $e_i$ 中得到的权限 $p_i$ 和角色 $r$ 在环境 $e_j$ 中得到的权限 $p_j$ 满足 $\{p_i\} \supseteq \{p_j\}$,则称 $e_j$ 为 $e_i$ 的子环境状态,记为 $e_i \geq e_j$.

**定义 3** 对于相同的环境状态,若 $t_i$、$t_j$ 是 $T$ 中的元素,角色 $r$ 在 $t_i$ 中得到的权限 $p_i$ 和角色 $r$ 在 $t_j$ 中得到的权限 $p_j$ 满足 $\{p_i\} \supseteq \{p_j\}$,则称 $t_j$ 为 $t_i$ 的子时态状态,记

为 $t_i \geq t_j$.

**定义 4**　行为 $a$ 指角色 $r$ 在某种环境 $e$ 下某段时间 $t$ 内实现某个功能所需权限集合的描述,即用户 $u$ 在启动会话 $s$ 获得权限 $p$ 时所需的角色、时态和环境信息,其中 $u \in U$、$s \in S$、$p \in P$、$r \in R$、$t \in T$、$e \in E$. $a$ 可以表示为三元组 $(r, t, e)$.

记 $a$ 对应的 $r$ 的集合为 $\{r^a\}$, $t$ 的集合为 $\{t^a\}$, $e$ 的集合为 $\{e^a\}$, $p$ 的集合为 $\{p^a\}$.

**定义 5**　行为约束 *Constraints-a* 指启动会话 $s$ 之后,用户 $u$ 只能够通过行为 $a$ 才能得到权限 $p$.

**定义 6**　用户-行为分配 $ua$ 指对用户 $u$ 分配行为 $a$ 的过程, $ua$ 的集合记为 $UA$.

**定义 7**　行为-权限分配 $ap$ 指对行为 $a$ 分配权限 $p$ 的过程, $ap$ 的集合记为 $AP$.

表 1 给出了行为约束的描述,其中可用 enable、不可用 disable、激活 active 的定义与文献[8]类似. 时态状态 $T$ 可以表示为 $[(TI, TP) | TD | TC]$,其中二元组 $(TI, TP)$ 表示事件的时间区间, $TD$ 表示事件的持续时间, $TC$ 表示事件发生的周期;环境状态 $E$ 可以表示为 $[EL | EN | EH | ES | EC]$,其中 $EL$ 表示事件发生的物理位置, $EN$ 表示事件发生的网络位置, $EH$ 表示事件发生时用户使用的硬件平台, $ES$ 表示事件发生时用户使用的软件平台, $EC$ 表示事件发生时用户使用的密码系统; $S$ 和 $R$ 分别为上述的会话集合和角色集合; assign/ deassign 表示分配和解分配操作; enable/ disable 表示行为的可用/不可用状态; $N_{active}$ 表示激活数量; $N_{max}$ 表示所能激活的最大数量; $active_{U\_total}$ 表示用户当前激活的所有行为的数量; $active_{P\_total}$ 表示得到某个权限的所有激活行为的数量. 表 1 只给出了行为约束的例子,根据不同的应用环境可以对行为约束做出更多的表述.

**表 1　行为约束的描述**

| 约束的分类 | 约束 | | 描述 |
|---|---|---|---|
| 行为可用约束 | 用户-行为分配 | | $(T, E, S, R, assign_U/\ deassign_U\ a\ to\ u)$ |
| | 行为可用 | | $(T, E, S, R, enable/\ disable\ a)$ |
| | 行为-权限分配 | | $(T, E, S, R, assign_P/\ deassign_P\ p\ to\ a)$ |
| 行为激活约束 | 激活行为的数量 | 用户 | $(T, E, S, R, N_{active}, active_{U\_total})$ |
| | | 权限 | $(T, E, S, R, N_{active}, active_{P\_total})$ |
| | 当前系统中激活行为的总数量 | 用户 | $(T, E, S, R, N_{max}, active_{U\_total})$ |
| | | 权限 | $(T, E, S, R, N_{max}, active_{P\_total})$ |

## 2.2　行为的层次结构及其继承机制

由于行为综合考虑了角色、时态和环境,所以也具有层次结构.

**定义 8**　环境层次结构 $EH \subseteq E \times E$ 是环境集合 $E$ 上的偏序关系. 对于任意的 $e_i$、$e_j \in E$, $(e_i, e_j) \in EH$ 当且仅当 $e_i \geq e_j$ 成立. 如果 $(e_i, e_j) \in EH$,则称 $e_i$ 是 $e_j$ 的高级环境, $e_j$ 是 $e_i$ 的低级环境,记为 $e_i \geq e_j$. 如果 $(e_i, e_j) \in EH$ 且不存在 $e_k$ 使得 $e_i \geq e_k$ 与 $e_k \geq e_j$ 成立,则称 $e_i$ 是 $e_j$ 的直接高级环境,记为 $e_i > e_j$.

**定义 9**　时态层次结构 $TH \subseteq T \times T$ 是时态集合 $T$ 上的偏序关系. 对于任意的 $t_i$、$t_j \in T$, $(t_i, t_j) \in TH$ 当且仅当 $t_i \geq t_j$ 成立. 如果 $(t_i, t_j) \in TH$,则称 $t_i$ 是 $t_j$ 的高级时态, $t_j$ 是 $t_i$ 的低级时态,记为 $t_i \geq t_j$. 如果 $(t_i, t_j) \in TH$ 且不存在 $t_k$ 使得 $t_i \geq t_k$ 与 $t_k \geq t_j$ 成立,则称 $t_i$ 是 $t_j$ 的直接高级时态,记为 $t_i > t_j$.

**定义 10**　将行为集合记为 $A$,行为层次结构 $AH \subseteq A \times A$ 是行为集合 $A$ 上的偏序关系. 对于任意的 $a_i = (r_i, t_i, e_i)$, $a_j = (r_j, t_j, e_j) \in A$, $(a_i, a_j) \in AH$ 当且仅当 $r_i \geq r_j$、$t_i \geq t_j$、$e_i \geq e_j$ 同时成立. 如果 $(a_i, a_j) \in AH$,则称 $a_i$ 是 $a_j$ 的高级行为, $a_j$ 是 $a_i$ 的低级行为,记为 $a_i \geq a_j$. 如果 $(a_i, a_j) \in AH$ 且不存在 $a_k$ 使得 $a_i \geq a_k$ 与 $a_k \geq a_j$ 成立,则称 $a_i$ 是 $a_j$ 的直接高级行为,记为 $a_i > a_j$.

在定义 10 的基础上,可以构造系统中的行为层次结构图. 为了讨论的方便性,我们采用类似文献[10]的方法对行为层次结构图上的路径做出如下定义.

**定义 11**　对任意的 $a_i \in A$,在行为层次结构 $AH$ 上,以 $a_i$ 为起点的路径为行为序列 $l = (a_i, \cdots, a_k, \cdots, a_j)$,其中 $a_i \geq a_k$, $a_k > a_j$,并且不存在 $a' \in A$,使得 $a_j > a' > a_k$. 从 $a_i$ 出发的所有不同路径全集记为 $L(a_i)$. $l$ 包含行为 $a$,记为 $a \in l$. $a_i$ 依赖于路径 $l$ 继承的低级行为集合为:

$$Subset\_action_l^{a_i} = \{a \mid a \in l\}$$

对任意的 $a \in Subset\_action_l^{a_i}$, $a$ 的有效角色集合、有效时态集合、有效环境集合和有效权限集合分别为:

$$Set\_role_l^a = \bigcup_{a' \in l, a' \leq a} R^{a'}$$

$$Set\_temporal_l^a = \bigcup_{a' \in l, a' \leq a} T^{a'}$$

$$Set\_environment_l^a = \bigcup_{a' \in l, a' \leq a} E^{a'}$$

$$Set\_permission_l^a = \bigcup_{a' \in l, a' \leq a} P^{a'}$$

$a_i$ 依赖于路径 $l$ 继承的有效角色集合、有效时态集合、有效环境集合和有效权限集合分别为:

$$Set\_validrole_l^{a_i} = \{r \mid a \in Subset\_action_l^{a_i}, r^a \in Set\_role_l^a\}$$

$$Set\_validtemporal_l^{a_i} = \{t \mid a \in Subset\_action_l^{a_i}, t^a \in Set\_temporal_l^a\}$$

$$Set\_validenvironment_l^{a_i} = \{e \mid a \in Subset\_action_l^{a_i}, e^a \in Set\_environment_l^a\}$$

$$Set\_validpermission_l^{a_i} = \{p \mid a \in Subset\_action_l^{a_i}, p^a \in Set\_permission_l^a\}$$

在行为层次结构 $AH$ 上, $a_i$ 的有效角色全集、有效时态全集、有效环境全集和有效权限全集分别为:

$Set\_validrole^{a_i} = \bigcup_{l \in L(a_i)} Set\_validrole_l^{a_i}$

$Set\_validtemporal^{a_i} = \bigcup_{l \in L(a_i)} Set\_validtemporal_l^{a_i}$

$Set\_validenvironment^{a_i} = \bigcup_{l \in L(a_i)} Set\_validenvironment_l^{a_i}$

$Set\_validpermission^{a_i} = \bigcup_{l \in L(a_i)} Set\_validpermission_l^{a_i}$

对任意的 $p \in Set\_validpermission^a$, $p$ 的有效角色集合、有效时态集合和有效环境集合分别为:

$$Set\_validrole^p = \{ r \mid r \in Set\_validrole^a, (a, p) \in AP\}$$

$$Set\_validtemporal^p = \{ t \mid t \in Set\_validtemporal^a, (a, p) \in AP\}$$

$$Set\_validenvironment^p = \{ e \mid e \in Set\_validenvironment^a, (a, p) \in AP\}$$

由上述公式可以得到用户 $u$ 在启动会话 $s$ 获得权限 $p$ 时所需的行为 $a_{usp}$ 满足下式:

$$a_{usp} \in \{ a \mid (u, a) \in UA, (a, p) \in AP, r^a \in Set\_validrole^p, e^a \in Set\_validenvironment^p, t^a \in Set\_validtemporal^p\}$$
$$a = (r, t, e)$$

图 1 给出了一个行为层次结构图的例子. 对于 $l = (a_1, a_5, a_8, a_{10})$, $a_1$ 依赖于路径 $l$ 继承的有效角色集合、有效时态集合和有效环境集合分别为 $(r_1, r_5, r_8, r_{10})$、$(t_1, t_5, t_8, t_{10})$、$(e_1, e_5, e_8, e_{10})$. 在行为层次结构图上, $a_1$ 的有效角色全集、有效时态全集和有效环境全集分别为 $(r_1, r_3, r_5, r_7, r_8, r_{10})$、$(t_1, t_3, t_5, t_7, t_8, t_{10})$、$(e_1, e_3, e_5, e_7, e_8, e_{10})$. 假设 $a_7$ 是可以享有权限 $p$ 的最低级的行为, 则权限 $p$ 的有效角色集合、有效时态集合和有效环境集合分别为 $\{r^{a_7}, r^{a_6}, r^{a_4}, r^{a_3}, r^{a_2}, r^{a_1}\}$、$\{t^{a_7}, t^{a_6}, t^{a_4}, t^{a_3}, t^{a_2}, t^{a_1}\}$、$\{e^{a_7}, e^{a_6}, e^{a_4}, e^{a_3}, e^{a_2}, e^{a_1}\}$.

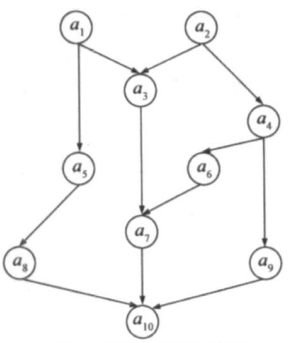

图1 行为层次结构图

### 2.3 基于行为的访问控制模型 ABAC

根据"行为"的概念, 下面借鉴文献[1]的形式给出了基于行为的访问控制模型 ABAC 的形式化定义.

**定义 12** ABAC 模型具有以下组件:

·$U$、$A$、$P$ 和 $S$(用户、行为、权限、会话), 其中 $A = (R, T, E)$, $R$、$T$、$E$ 分别为角色、时态和环境, $A$ 是一个多元因素相互交叉影响的复杂关系, 一般用 Action 层次树或 ACL(Access Control List)描述

·$UA \subseteq U \times A$, 表示多对多的用户-行为的分配关系

·$AP \subseteq A \times P$, 表示多对多的行为-权限的分配关系

·$AH \subseteq A \times A$, 表示行为集合 $A$ 上的偏序关系, 记为 $\geq$

·Constraints, 表示约束条件

·$user: S \rightarrow U$, 将会话 $s_i$ 映射到单个用户 $user(s_i)$ 的函数(会话生命期内保持不变)

·$actions: S \rightarrow 2^A$, 将会话 $s_i$ 映射到行为集合 $actions(s_i) \subseteq \{a \mid (\exists a' \geq a)[(user(s_i), a') \in UA]\}$ 的函数, 会话 $s_i$ 具有权限 $\bigcup_{a \in actions(s_i)} \{p \mid (\exists a'' \leq a)[(a'', p) \in AP]\}$

在行为集合 $A = (R, T, E)$ 中, 环境 $E$ 十分重要. 例如, 我们可以使用如下方法获得不同的环境状态: 对于 $EL$, 可以使用 GPS 等定位系统; 对于 $EN$, 可以通过网络布线和设备配置, 使用 IP 地址和服务器中的地址映射表等网络信息; 对于 $EH$, 可以选用配有 TPM 等安全芯片的硬件平台; 对于 $ES$, 可以选用能够访问特殊数据的专用软件; 对于 $EC$, 可以选用不同密码算法或不同群组密钥. 由此, 可以对不同的环境 $E$ 加以区分.

对于行为-权限的分配关系 $AP$, 由于行为由角色、时态和环境三者决定, 所以我们可以得到以下关系:

$\forall r_i \in R, t_j \in T, e_k \in E, i, j, k \in N$, $\exists p_{s_1}, p_{s_2}, \cdots, p_{s_n} \in P$ 满足 $(a_{ijk} = (r_i, t_j, e_k), p_{s_l}) \in AP$, $l = 1 \sim n$. 如果有 $i', j', k' \in N$ 和 $p_{s_1'}, p_{s_2'}, \cdots, p_{s_n'} \in P$ 满足 $(a_{i'j'k'} = (r_{i'}, t_{j'}, e_{k'}), p_{s_l}) \in AP$, $l = 1 \sim n'$, 且 $r_{i'} \leq r_i$, $t_{j'} \leq t_j$, $e_{k'} \leq k_k$, 则 $\{p_{s_1'}, p_{s_2'}, \cdots, p_{s_n'}\} \subseteq \{p_{s_1}, p_{s_2}, \cdots, p_{s_n}\}$.

参考 RBAC 的结构图[1], 图 2 给出了基于行为的访问控制模型 ABAC 的结构图, 其中包含了行为层次、角色层次、环境层次、时态层次. 从图中可以看出 ABAC 对 RBAC 进行了扩展, 主要体现在行为(A)的结构上, 即行为(A)包含角色(R)、环境状态(E)和时态状态(T). 行为(A)的状态随着角色、时态和环境的不同而动态变化. 其中, 环境状态和时态状态对角色所能享有的权限

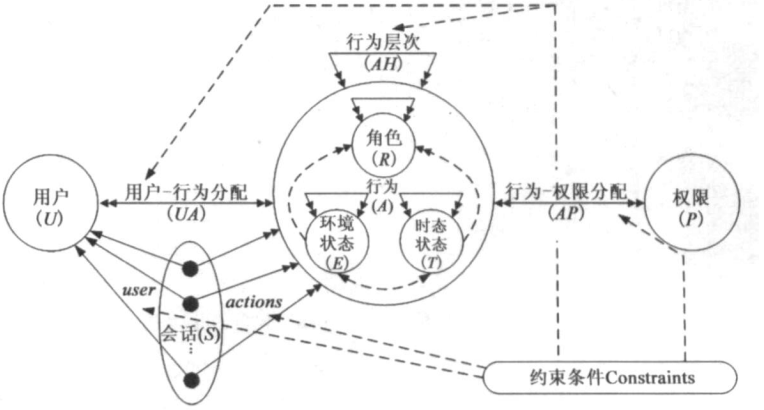

图2 ABAC模型

具有直接影响,如不同的物理位置、网络位置、硬件平台、软件平台和密码系统等外部环境可以对角色产生影响;同时不同的时态状态如事件发生的起始时间、终止时间、持续时间、周期等也会对角色产生影响. 此外,对于相同的角色,环境状态和时态状态之间也存在相互影响. 考虑不同的地理位置,如公司内部网络和外部网络;不同的时间段,如上班时间和下班时间等. 我们可以给出 4 种不同的环境-时态组合:内部网络-上班、内部网络-下班、外部网络-上班、外部网络-下班等. 这时,角色可能由于所处环境-时态组合的不同而享有不同的权限,如内部网络-上班对应访问机密资源的权限、内部网络-下班和外部网络-上班对应访问公司内部公开资源的权限、外部网络-下班对应访问公司对外公开资源的权限. 通过将角色、时态和环境综合考虑,可以使 ABAC 灵活地处理各种信息系统中的访问控制问题.

## 3　ABAC 管理模型

文献[2]通过 URA97、PRA97 和 RRA97 提出了解决 RBAC 中角色管理问题的方法. 但其没有考虑选择何种角色作为管理角色的问题. 文献[17～22]也对角色管理问题进行了一些研究,但这些工作都只是对文献[2]中工作的简单扩展,同时也没有考虑管理角色 $AR$ 的安全问题.

### 3.1　管理行为与管理模型

在 ABAC 中,管理行为用来对其他的行为进行安全管理,所以管理行为本身的安全性是关系到 ABAC 模型整体安全性的核心问题. 已有的方案只关注管理角色的功能,而没有对管理角色本身进行限定. 下面,设 $AR$ 表示管理角色集合,我们通过引入受限的时态状态和环境状态给出管理行为的定义.

**定义 13**　管理行为 $ada$ 是一种特殊的行为,满足行为的所有属性,但其环境状态和时态状态是受限的. 如环境状态中的物理位置为单位内部、网络位置为内部专用网络、软硬件平台支持密码操作、时态状态为上班时间等. $ada$ 可以表示为 $(ar, limt, lime)$,其中 $ar \in AR$、$limt \in T$、$lime \in E$. 记 $ada$ 的集合为 $ADA$、$limt$ 的集合为 $LIMT$、$lime$ 的集合为 $LIME$.

通过在受限的环境状态中引入可信平台模块,ABAC 中的管理行为可以提供更安全的服务,从而对一般行为进行可信的管理. 令 $ADP$ 表示管理权限的集合,下面借鉴文献[2]的形式对 ABAC 的管理模型进行描述.

**定义 14**　ABAC 管理模型具有以下组件:

·$U$:用户集合;$A$:一般行为集合;$ADA$:管理行为集合;$P$:权限集合;$ADP$:管理权限集合;$S$:会话集合

其中 $A = (R, T, E)$,$ADA = (AR, LIMT, LIME)$,$R$、$T$、$E$ 分别为角色、时态和环境,$AR$ 为管理角色,$LIMT$、$LIME$ 分别为受限的时态和受限的环境

·$AA \subseteq A \times A$,表示多对多的行为-行为的分配关系

·$UA \subseteq U \times A$,表示多对多的用户-行为的分配关系

$UADA \subseteq U \times ADA$,表示多对多的用户-管理行为的分配关系

·$AP \subseteq A \times P$,表示多对多的行为-权限的分配关系

$ADAP \subseteq ADA \times P$,表示多对多的管理行为-权限的分配关系

·$AH \subseteq A \times A$,表示行为集合 $A$ 上的偏序关系,记为 $\geq$

$ADAH \subseteq ADA \times ADA$,表示管理行为集合 $ADA$ 上的偏序关系,记为 $\geq$

·Constraints,表示约束条件

·$user: S \rightarrow U$,将会话 $s_i$ 映射到单个用户 $user(s_i)$ 的函数(会话生命期内保持不变)

·$actions: S \rightarrow 2^{A \cup ADA}$,将会话 $s_i$ 映射到行为集合 $actions(s_i) \subseteq \{a \mid (\exists a' \geq a)[(user(s_i), a') \in UA \cup UADA]\}$ 的函数,会话 $s_i$ 具有权限 $\cup_{a \in actions(s_i)} \{p \mid (\exists a'' \leq a)[(a'', p) \in AP \cup ADAP]\}$.

借鉴文献[2],图 3 给出了 ABAC 管理模型的结构图,其中包含了一般行为层次、管理行为层次、角色层次、环境层次、时态层次. 从图 3 中可以看出,ABAC 管理模型通过管理行为集合 $ADA$ 对用户-行为分配、行为-权限分配和行为状态进行管理. ABAC 管理模型利用 Constraints 对用户-行为、行为-权限、用户-管理行为、管理行为-管理权限的分配进行限制,还通过用户-管理行为分配、管理行为-管理权限分配对管理行为进行控制. 由于管理行为在受限的环境状态和时态状态下进行一般行为的管理,所以可以保证管理行为的安全性.

### 3.2　ABAC 管理模型的功能

ABAC 管理模型下用户-行为分配、用户-行为撤销的控制关系可以借用文献[2]的定义 2 和定义 4;行为-权限分配、行为-权限撤销的控制关系可以借用文献[2]的定义 6. 有了 ABAC 管理模型之后,可以得到 ABAC 模型下用户-管理行为和管理行为-管理权限的控制关系的定义,以及行为状态管理中的相关函数.

**定义 15**　先决条件是通过 $\wedge$和 $\vee$操作符对 $x$ 和 $\bar{x}$ 进行操作的布尔表达式. 其中,

(1) $x \in A$ 是一般行为. 对于用户 $u$ 来说,若 $x$ 为真则 $(\exists x' \geq x)((u, x') \in UA \wedge (u, x') \notin UADA)$;若 $\bar{x}$ 为真则 $(\forall x' \geq x)((u, x') \notin UA \wedge (u, x') \notin UADA)$. 对于给定的一般行为集合 $A$,令 $CA$ 表示使用 $A$ 中行为可能得到的所有先决条件;

(2) $x \in ADA$ 是管理行为. 对于用户 $u$ 来说,若 $x$ 为

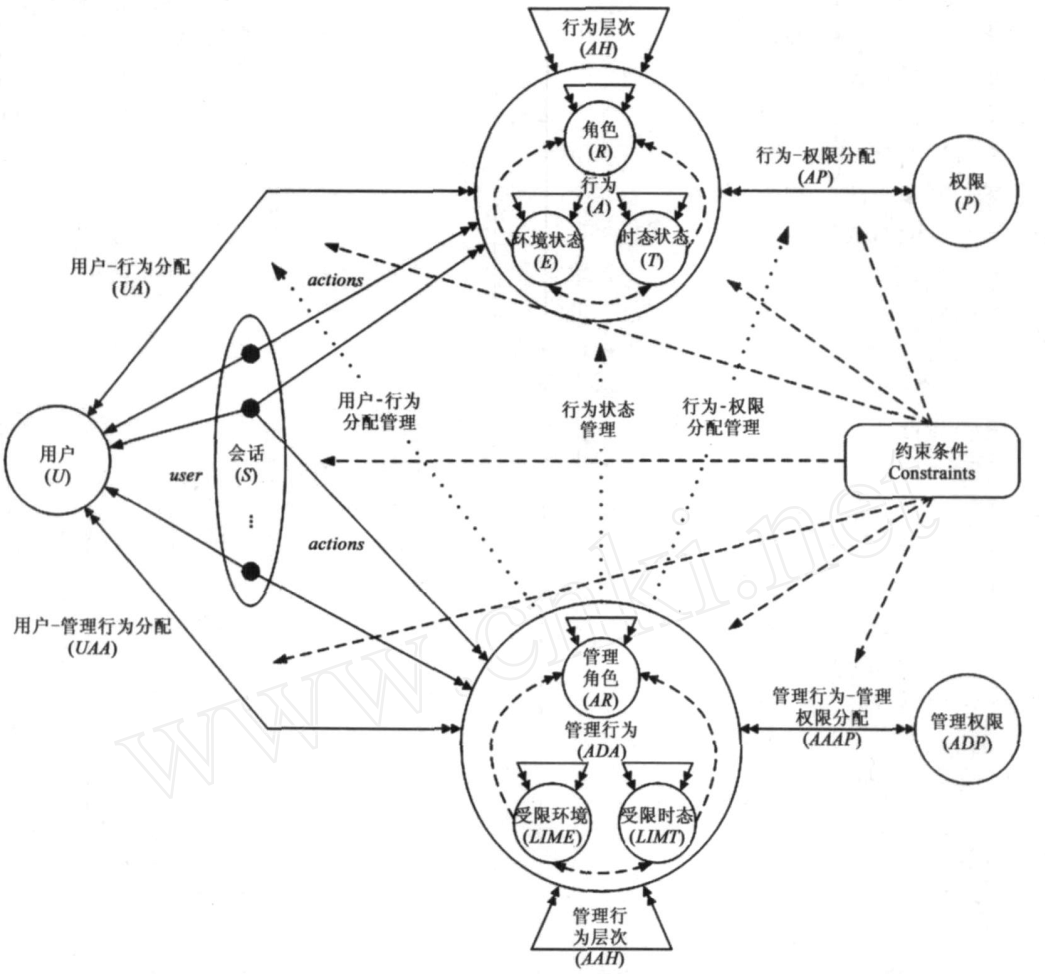


图3 ABAC管理模型

真则$(\exists x'\geq x)((u,x')\in UADA\wedge(u,x')\notin UA)$;若$\overline{x}$为真则$(\forall x'\geq x)((u,x')\notin UADA\wedge(u,x')\notin UA)$.对于给定的管理行为集合$ADA$,令$CADA$表示使用$ADA$中管理行为可能得到的所有先决条件.

**定义16** ABAC管理模型分别使用如下关系对用户-管理行为进行分配、对已经分配的用户-管理行为进行撤销:

$$can\_assignsu\subseteq ADA'\times CADA\times 2^{ADA}$$

$$can\_revokesu\subseteq ADA'\times 2^{(ADA\setminus\{superada\})}$$

这里$ADA'=ADA\setminus\{a|\nexists a'\in ADA,a'<a\}$,$superada$是系统设定的超级管理行为,位于管理行为层次结构的顶层,且不能被撤销.设$Z$表示"可被分配的管理行为"的集合,$can_assignsu(x,y,Z)$表示管理行为$x$(或$\forall x'\geq x$)可以对满足先决条件$y$的用户分配管理行为$z\in Z$.设$Z$表示撤销的管理行为集合,则$can_revokesu(x,Z)$表示管理行为$x$可以撤销分配给用户的管理行为集合$Z$.

**定义17** ABAC管理模型分别使用如下关系对管理行为-管理权限进行分配、对已经分配的管理行为-管理权限进行撤销:

$$can\_assignsp\subseteq ADA'\times CADA\times 2^{ADP}$$

$$can\_revokesp\subseteq ADA'\times 2^{ADP}$$

这里,$ADA'=ADA\setminus\{a|\nexists a'\in ADA,a'<a\}$,$ADP=\{userad,actoinad,permissionad,uaad,apad,asad\}$表示管理权限集合,包括用户管理userad、行为管理actoinad、权限管理permissionad、用户-行为分配管理uaad、行为-权限分配管理apad、行为状态管理asad等.设$Z$表示"可被分配的管理权限"的集合,$can_assignsp(x,y,Z)$表示管理行为$x$(或$\forall x'\geq x$)可以对满足先决条件$y$的管理行为分配管理权限$z\in Z$.设$Z$表示撤销的管理权限集合,则$can_revokesp(x,Z)$表示管理行为$x$可以撤销分配给管理行为的管理权限集合$Z$.

下面采用$Z$-符号对行为状态管理中的添加行为、修改行为和删除行为进行形式化描述.其中,$NAME$是一个抽象数据类型,可以表示ABAC模型中的行为、角色、时态状态、环境状态、用户、会话、权限等组件,以及证书类型;$ACTIONS$为行为集合,$ROLES$为角色集合,$TSTATES$为时态状态集合,$ESTATES$为环境状态集合,

*USERS* 为用户集合; *SESSIONS* 为会话集合, *OPS* 为操作集合, *OBS* 为对象集合.

*AddAction( action : NAME)* ◁
　*action* ∉ *ACTIONS*
　if *action. role* ∉ *ROLES* then *ROLES* ′= *ROLES* ∪{ *role*}
　if *action. temporalstate* ∉ *TSTATES* then *TSTATES* ′= *TSTATES* ∪{ *temporalstate*}
　if *action. environmentalstate* ∉ *ESTATES* then *ESTATES* ′= *ESTATES* ∪{ *environmentalstate*}
　*ACTIONS* ′= *ACTIONS* ∪{ *action*}
　*UA* ′= *UA* ∪{ *action*| →∅ }
　*AP* ′= *AP* ∪{ *action*| →∅ } ▷

*ModifyAction( action , role , temporalstate , environmentalstate : NAME)* ◁
　*action* ∈*ACTIONS*
　if *role* ∉ *ROLES* then *ROLES* ′= *ROLES* ∪{ *role*}
　if *temporalstate* ∉ *TSTATES* then *TSTATES* ′= *TSTATES* ∪{ *temporalstate*}
　if *environmentalstate* ∉ *ESTATES* then *ESTATES* ′= *ESTATES* ∪{ *environmentalstate*}
　[ ∀*s* ∈*SESSIONS* ·*action* ∈*session* _ *actions*( *s*) ⇒*DeleteSession*( *s*) ]
　*action* ′= ( *role* , *temporalstate* , *environmentalstate*)
　*ACTIONS* ′= *ACTIONS* \ { *action*} ∪{ *action* ′}
　*UA* ′= *UA* \ { ∀*u* ∈*USERS* ·*action*| →*u*} ∪{ *action* ′| →∅ }
　*AP* ′= *AP* \ { ∀*op* ∈*OPS*; ∀*ob* ∈*OBS* ·*action*| →( *op* , *ob*)} ∪{ *action* ′| →∅ }

*DeleteAction( action : NAME)* ◁
　*action* ∈*ACTIONS*
　[ ∀*s* ∈*SESSIONS* ·*action* ∈*session* _ *actions*( *s*) ⇒*DeleteSession*( *s*) ]
　[ ∀*a* ∈*ACTIONS* \ { *action*} ·*a. role* ≠*action. role* ⇒*ROLES* ′= *ROLES* \ { *action. role*} ]
　[ ∀*a* ∈*ACTIONS* \ { *action*} ·*a. temporalstate* ≠*action. temporalstate* ⇒ *TSTATES* ′= *TSTATES* \ { *action. temporalstate*} ]
　[ ∀*a* ∈*ACTIONS* \ { *action*} ·*a. environmentalstate* ≠*action. environmentalstate* ⇒*ESTATES* ′= *ESTATES* \ { *action. environmentalstate*} ]
　*UA* ′= *UA* \ { ∀*u* ∈*USERS* ·*action*| →*u*}
　*AP* ′= *AP* \ { ∀*op* ∈*OPS*; ∀*ob* ∈*OBS* ·*action*| →( *op* , *ob*) }
　*ACTIONS* ′= *ACTIONS* \ { *action*} ▷

通过识别用户身份、角色信息、物理位置、网络位置、硬件平台、软件平台、密码系统和访问资源时的起始时间、终止时间、持续时间、周期等, 利用表 2 中的相关函数可以实现对 ABAC 的用户-行为、行为-权限和行为状态进行管理.

**表 2　管理方法使用的相关函数**

| 函数名 | 描述 |
|---|---|
| verifyid | 若用户身份和证书合法则返回 True, 否则返回 False<br>*verifyid( userid , certification : NAME; out result : BOOLEAN)* ◁<br>*result* = ( *userid* ∈*U*) ∧( *isvalid*( *certification*) ) ▷ |
| isenable | 若行为 action 可用则返回 True, 否则返回 False<br>*isenable( action : NAME; out result : BOOLEAN)* ◁<br>*action* ∈*ACTIONS*<br>*result* = *action* ∈*ENABLEA* * ▷ |
| globalt | 若请求的时态状态 *reqt* 可用且可以忽视所有的环境状态则返回 True, 否则返回 False<br>*globalt( reqt : NAME; out result : BOOLEAN)* ◁<br>*reqt* ∈*TSTATES*<br>*result* = ( ∀$e_i$ , $e_j$ ∈*ESTATES*; ∀*r* ∈*ROLES*; ∀*p* ∈*PERMISSIONS*| $e_i$ ≠$e_j$ ·( $a_i$ = ( *r* , *reqt* , $e_i$) , *p*) = ( $a_j$ = ( *r* , *reqt* , $e_j$) , *p*) ∈*AP*) ∧ ( ∀$e_i$ , $e_j$ ∈*ESTATES*; ∀*r* ∈*ROLES*; ∀*a* ∈*ACTIONS*| $e_i$ ≠$e_j$ ·( *u* , $a_i$ = ( *r* , *reqt* , $e_i$) ) = ( *u* , $a_j$ = ( *r* , *reqt* , $e_j$) ) ∈*UA*) ▷<br>例如, 在一个证券公司中, 股票交易系统只能在特定的时间开放, 且其开放时所有的信息都是公开的, 但在系统关闭时不能进行买入或卖出操作. |
| globale | 若请求的环境状态 *reqe* 可用且可以忽视所有的时态状态则返回 True, 否则返回 False<br>*globale( reqe : NAME; out result : BOOLEAN)*<br>*reqe* ∈*ESTATES*<br>*result* = ( ∀$t_i$ , $t_j$ ∈*TSTATES*; ∀*r* ∈*ROLES*; ∀*p* ∈*PERMISSIONS*| $t_i$ ≠$t_j$ ·( $a_i$ = ( *r* , $t_i$ , *reqe*) , *p*) = ( $a_j$ = ( *r* , $t_j$ , *reqe*) , *p*) ∈*AP*) ∧ ( ∀$t_i$ , $t_j$ ∈*TSTATES*; ∀*r* ∈*ROLES*; ∀*a* ∈*ACTIONS*| $t_i$ ≠$t_j$ ·( *u* , $a_i$ = ( *r* , $t_i$ , *reqe*) ) = ( *u* , $a_j$ = ( *r* , $t_j$ , *reqe*) ) ∈*UA*) ▷<br>其中, 对于 *reqe* = [ *el*| *en*| *eh*| *es*| *ec* ], 可能只需部分元素就可以确定 *reqe* 是否满足 *global* 属性. 例如, 在一个应急指挥信息管理系统中, 操作者需要使用特定的安全平台(如 *eh*、*es*、*ec* 可以达到足够的安全性) 对系统数据进行处理, 且其访问系统的时间和位置状态是没有限制的, 即操作者可以在任意时间内通过安全平台访问应急指挥信息管理系统中的数据. |
| verifyt | 若请求的时态状态 *reqt* 满足时态状态 *validt* 的要求则返回 True, 否则返回 False<br>*verifyt( reqt , validt : NAME; out result : BOOLEAN)* ◁<br>*reqt* ∈*TSTATES*; *validt* ⊆ *TSTATES*<br>*result* = ( ∃$t_1$ , $t_2$ ∈*validt* ·$t_1$ ≤*reqt* ≤$t_2$) ∧<br>( *r* ∈*ROLES*; *e* ∈*ESTATES*; *p* ∈*PERMISSOINS* ·( *u* , *a* = ( *r* , *reqt* , *e*) ) ∈*UA* ∧( *a* = ( *r* , *reqt* , *e*) , *p*) ∈*AP*) ▷ |
| verifye | 若请求的环境状态 *reqe* 满足环境状态 *valide* 的要求则返回 True, 否则返回 False<br>*verifye( reqe , valide : NAME; out result : BOOLEAN)* ◁<br>*reqe* ∈*ESTATES*; *valide* ⊆ *ESTATES*<br>*result* = ( ∃$e_1$ , $e_2$ ∈*valide* ·$e_1$ ≤*reqe* ≤$e_2$) ∧<br>( *r* ∈*ROLES*; *t* ∈*TSTATES*; *p* ∈*PERMISSIONS* ·( *u* , *a* = ( *r* , *t* , *reqe*) ) ∈*UA* ∧( *a* = ( *r* , *t* , *reqe*) , *p*) ∈*AP*) ▷ |

(续表 2)

| 函数名 | 描述 |
|---|---|
| n_activeabyu | 返回用户 *user* 激活的行为数量<br>*n_activeabyu(user:NAME; out result:N)* ◁<br>*user* ∈*USERS*<br>*result* = $N_{active_{U\_total}}(user)$ ▷ |
| maxn_activeabyu | 返回用户 *user* 最多能激活的行为数量<br>*maxn_activeabyu(user:NAME; out result:N)* ◁<br>*user* ∈*USERS*<br>*result* = $N_{max_{U\_total}}(user)$ ▷ |
| n_activeabyp | 返回激活权限 *permission* 的行为数量<br>*n_activeabyp(permission:NAME; out result:*N) ◁<br>*permission* ∈*PERMISSIONS*<br>*result* = $N_{active_{P\_total}}(permission)$ ▷ |
| maxn_activeabyp | 返回最多能激活权限 *permission* 的行为数量<br>*maxn_activeabyp(permission:NAME; out result:*N) ◁<br>*permission* ∈*PERMISSIONS*<br>*result* = $N_{max_{P\_total}}(permission)$ ▷ |

* ENABLEA 表示可用行为集合

## 4 ABAC 提供的特性

本节将 ABAC 与已有的几种访问控制模型进行比较,结果如表 3 所示. 从表 3 可以看到,ARBAC97、GenericWA-RBAC、DERBAC、Fuzzy RBAC 不能提供时态约束和环境约束,所以其应用受到了限制. 而 TRBAC 和文献[10]中的模型虽然涉及到时态约束的概念,但由于这两个模型比较简单,也不适用于解决网络环境下信息系统的访问控制问题. GTRBAC 对时态约束进行了详细的分析,因此其应用方式比较灵活,但 GTRBAC 没有考虑到环境约束. LRBAC、GEO-RBAC、SC-RBAC、Ex-RBAC 和文献[11]中的模型只简单涉及到角色位置信息的概念,而且这几种模型没有涉及用户的操作平台,同时也没有对位置与访问控制的关系进行分析,所以这些模型的应用范围也受到了限制. 上述几种模型也没有考虑到在移动计算环境下访问控制模型的建立. 文献[17～21]只考虑了管理角色的功能,但没有考虑管理角色的安全问题. ABAC 除了可以提供传统的角色、角色控制和时态约束之外,还提供环境约束,支持移动计算的接入用户、接入的具体业务需求、接入位置、接入时间和接入平台是随机的、事先不可预知等典型特性. 因此,ABAC 具有广泛的应用范围、方便的应用方式,能够更加有效地解决网络环境下支持移动计算的信息系统中的访问控制问题.

表 3 ABAC 与已有的几种访问控制模型的特性比较

| 性质 / 模型 | 角色 | 时态状态 | 环境状态* | | 应用 | 方便性 | 移动计算 | 管理的安全性 |
|---|---|---|---|---|---|---|---|---|
| | | | 位置 | 平台 | | | | |
| ARBAC97[2] | 支持 | 不支持 | 不支持 | 不支持 | 较少 | 低 | 不支持 | 不支持 |
| GenericWA-RBAC[4] | 支持 | 不支持 | 不支持 | 不支持 | 较少 | 中 | 不支持 | 不支持 |
| DERBAC[5] | 支持 | 不支持 | 不支持 | 不支持 | 较少 | 中 | 不支持 | 不支持 |
| Fuzzy RBAC[6] | 支持 | 不支持 | 不支持 | 不支持 | 一般 | 中 | 不支持 | 不支持 |
| TRBAC[7] | 支持 | 部分支持 | 不支持 | 不支持 | 较少 | 低 | 不支持 | 不支持 |
| GTRBAC[8,9] | 支持 | 全部支持 | 不支持 | 不支持 | 一般 | 高 | 不支持 | 不支持 |
| 文献[10]中的模型 | 支持 | 复杂 | 不支持 | 不支持 | 较少 | 低 | 不支持 | 不支持 |
| 文献[11]中的模型 | 支持 | 不支持 | 支持单一 | 不支持 | 一般 | 低 | 部分支持 | 不支持 |
| LRBAC[12,13] | 支持 | 不支持 | 支持单一 | 不支持 | 一般 | 中 | 部分支持 | 不支持 |
| GEO-RBAC[14] | 支持 | 全部支持 | 支持单一 | 不支持 | 一般 | 中 | 部分支持 | 不支持 |
| SC-RBAC[15] | 支持 | 全部支持 | 支持单一 | 不支持 | 一般 | 中 | 部分支持 | 不支持 |
| Ex-RBAC[16] | 支持 | 部分支持 | 支持单一 | 不支持 | 较少 | 低 | 部分支持 | 不支持 |
| EARBAC[17] | 支持 | 不支持 | 不支持 | 不支持 | 较少 | 低 | 不支持 | 不支持 |
| X-GTRBAC[18] | 支持 | 不支持 | 不支持 | 不支持 | 较少 | 低 | 不支持 | 不支持 |
| 文献[19]中的模型 | 支持 | 不支持 | 不支持 | 不支持 | 较少 | 低 | 不支持 | 不支持 |
| 文献[20]中的模型 | 支持 | 不支持 | 不支持 | 不支持 | 较少 | 低 | 不支持 | 不支持 |
| DRBAC[21] | 支持 | 不支持 | 不支持 | 不支持 | 较少 | 低 | 部分支持 | 不支持 |
| ABAC | 支持 | 全部支持 | 支持多种 | 支持 | 较多 | 高 | 支持 | 支持 |

*为了简洁,只比较了位置和平台因素

## 5 结束语

访问控制的主要任务是保证信息资源不被非法使用和访问,是对信息资源进行保护的重要措施,也是计算机系统最重要和最基础的安全机制.据我们所知现有的研究工作既没有形式化定义环境的概念,也没有详细讨论环境与访问控制的关系.本文首先介绍了包含位置、操作平台等要素的环境的概念并对其进行形式化描述,然后将角色、时态和环境的概念相结合给出了行为的定义.在介绍行为层次结构及其继承机制的基础上提出了基于行为的访问控制模型 ABAC;通过引入受限的时态状态和环境状态,给出了管理行为的定义和 ABAC 管理模型的结构;介绍了 ABAC 管理模型下的控制关系 *can_assignsu*、*can_revokesu*、*can_assignsp*、*can_revokesp*,并采用 Z 符号对行为状态管理中的添加行为、修改行为和删除行为进行形式化描述,给出了 ABAC 管理方法使用的相关函数.与已有其他模型相比,ABAC 更加适用于解决网络环境下支持移动计算的信息系统中的访问控制问题.

**参考文献**:


[1] R Sandhu, E Coyne, H Feinstein, et al. Role-based access control models[J]. IEEE Computer, 1996, 29(2): 38 - 47.

[2] R Sandhu, V Bhamidipati, Q Munawer. The ARBAC97 model for role-based administration of roles[J]. ACM Transactions on Information and System Security, 1997, 2(1): 105 - 135.

[3] D F Ferraiolo, R Sandhu, S Gavrila, et al. Proposed NIST standard for role-based access control[J]. ACM Transactions on Information and System Security, 2001, 4(3): 224 - 274.

[4] P H Bammigatti. GenericWA-RBAC: role based access control model for web applications[A]. Proceedings of the 9th International Conference on Information Technology (ICIT '06) [C]. Bhubaneswar, India: IEEE Computer Society, 2006. 237 - 240.

[5] L Dong, S Yu, K Ouyang. The dynamic endpoint-based access control model on VPN[A]. Proceedings of the International Conference on Networking, Architecture, and Storage (NAS 2007) [C]. Guilin, China: IEEE Computer Society, 2007. 44 - 54.

[6] H Takabi, M Amini, R Jalili. Enhancing role-based access control model through fuzzy relations[A]. Proceedings of the Third International Symposium on Information Assurance and Security [C]. Manchester, UK: IEEE Computer Society, 2007. 131 - 136.

[7] E Bertino, P A Bonatti. TRBAC: a temporal role-based access control model[J]. ACM Transactions on Information and System Security, 2001, 4(3): 191 - 223.

[8] J B D Joshi, E Bertino, U Latif, et al. A generalized temporal role-based access control model[J]. IEEE Transactions on Knowledge and Data Engineering, 2005, 17(1): 4 - 23.

[9] J B D Joshi, E Bertino, A Ghafoor. An analysis of expressiveness and design issues for the generalized temporal role-based access control model[J]. IEEE Transactions on Dependable and Secure Computing, 2005, 2(2): 157 - 175.

[10] 王小明,赵宗涛.基于角色的时态对象存取控制模型[J].电子学报,2005,33(9):1634 - 1638.
Wang Xiao-ming, Zhao Zong-tao. Role-based access control model of temporal object[J]. Acta Electronica Sinica, 2005, 33(9): 1634 - 1638. (in Chinese)

[11] M J Covington, W Long, S Srinivasan. Securing context-aware applications using environment roles[A]. Proceedings of the 6th ACM Symposium on Access Control Models and Technologies[C]. Chantilly, Virginia, USA: ACM Press, 2001. 10 - 20.

[12] I Ray, L Yu. Short paper: towards a location-aware role-based access control model[A]. Proceedings of the First International Conference on Security and Privacy for Emerging Areas in Communications Networks [C]. Athens, Greece: IEEE Computer Society, 2005. 234 - 236.

[13] I Ray, M Kumar, L Yu. LRBAC: a location-aware role-based access control model[A]. Proceedings of the Second International Conference on Information Systems Security (ICISS 2006) [C]. Kolkata, India: Springer-Verlag, 2006. 147 - 161.

[14] M L Damiani, E Bertino, B Catania. GEO-RBAC: A spatially aware RBAC[J]. ACM Transactions on Information and System Security, 2007, 10(1): 1 - 42.

[15] 张宏,贺也平,石志国.一个支持空间上下文的访问控制形式模型[J].中国科学 E 辑:信息科学,2007,37(2):254 - 271.
Zhang Hong, He Ye-ping, She Zhi-guo. A formal model for access control with supporting spatial context[J]. Science in China Series F: Information Sciences, 2007, 50(3): 419 - 439. (in Chinese)

[16] X Cui, Y Chen, J Gu. Ex-RBAC: An extended role based access control model for location-aware mobile collaboration system[A]. Proceedings of the Second International Conference on Internet Monitoring and Protection (ICIMP 2007) [C]. Silicon Valley, USA: IEEE Computer Society, 2007. 36 - 41.

[17] 龙勤,刘鹏,潘爱民.基于角色的扩展可管理访问控制模型研究与实现[J].计算机研究与发展,2005,42(5):868 - 876.
Long Qin, Liu Peng, Pan Ai-min. Research and implementation of an extended administrative role-based access control model[J]. Journal of Computer Research and Development, 2005, 42(5): 868 - 876. (in Chinese)

[18] R Bhatti, B Shafiq, J B D Joshi, et al. X-GTRBAC Admin: A

decentralized administration model for enterprise wide access control[J]. ACM Transactions on Information and System Security, 2005, 8(4): 388 - 423.

[19] S Oh, R Sandhu, X Zhang. An effective role administration model using organization structure[J]. ACM Transactions on Information and System Security, 2006, 9(2): 113 - 137.

[20] J Crampton. Understanding and developing role-based administrative models[A]. Proceedings of the 12th ACM Conference on Computer and Communications Security[C]. Alexandria, VA, USA: ACM Press, 2005. 158 - 167.

[21] Q Li, J Shi, S Qing. An administration model of DRBAC on the web[A]. Proceedings of the IEEE International Conference on e-Business Engineering[C]. Beijing, China: IEEE Computer Society, 2005. 364 - 367.

**作者简介**:

**李凤华** 男, 1966 年 3 月出生于湖北省浠水县, 西安电子科技大学博士生, 北京电子科技学院教授, 主要研究方向为网络安全与可信计算. E-mail: lfh@besti.edu.cn

**王 巍** 男, 1980 年 2 月出生于河北省张家口市, 西安电子科技大学博士生, 主要研究方向为群组密钥管理、协议形式化证明. E-mail: wei_wang@mail.xidian.edu.cn

**马建峰** 男, 1963 年 10 月出生于陕西省西安市, 西安电子科技大学计算机学院院长、博士、教授、博士生导师, 主要研究方向为密码学与网络安全. E-mail: jfma@mail.xidian.edu.cn

**梁晓艳** 女, 1985 年 4 月出生于湖南省益阳市, 西安电子科技大学硕士生, 主要研究方向为网络安全.

(上接第 1872 页)

[4] Ateniese G, Steiner M, Tsudik G. New multiparty authentication services and key agreement protocols[J]. IEEE Journal on Selected Areas in Communications, 2000, 18(4): 628 - 640.

[5] Boneh D, Franklin M. Identity-based Encryption from the weil Pairing[A]. In Proceedings of Crypto '2001[C]. Berlin: Springer-Verlag, 2001. 213 - 229.

[6] Joux A. One round protocol for tripartite Diffie-Hellman[A]. Proceedings of Algorithmic Number Theory Symposium[C]. Berlin: Springer-Verlag, 2000. 385 - 394.

[7] Smart N P. An Identity based authenticated Key Agreement protocol based on the Weil Pairing. Cryptography [R/OL]. eprint Archive, http://eprint.iacr.org/2001/111.

[8] Boneh D, Franklin M. Identity-based encryption from the Weil Pairing[A]. Advances in Cryptography-CRYPTO 2001[C]. Berlin: Springer-Verlag, 2001: 213 - 229.

[9] Cocks C. An Identity based encryption scheme based on quadratic residues[A]. Advances in Cryptography and Coding[C]. Berlin: Springer-Verlag, 2001. 360 - 363.

[10] Sattam S, Kenneth A. Parterson G. Authenticated Three Party Key Agreement Protocols from Pairings[OL]. http://eprint.iacr.org/2002/035.

[11] Boneh D, Silverberg A. Application of Multilinear forms to Cryptography[OL]. http://eprint.iacr.org/2002/080.

[12] H K Lee, H S Lee, Y R Lee. Multi-party Authenticated Key Agreement Protocols from Multilinear Forms[R/OL]. Cryptology ePrint Archive: http://eprint.iacr.org/2002/166.

[13] H M Lee, K J Ha, K M Ku. ID-based Multi-party Authenticated Key Agreement Protocols from Multilinear Forms[A]. Information Security, 8th International Conference, ISC 2005[C]. Berlin: Springer-Verlag, 2005. 104 - 117.