

# 基于规则的受限委派框架

尹 刚 王怀民 史殿习 滕 猛

(国防科学技术大学计算机学院 长沙 410073)

**摘 要** 委派(delegation)允许特权在主体间传播,是信任管理系统实现跨域授权的核心机制,但不加限制的委派可导致特权扩散,削弱了信息系统的安全性.现有信任管理系统的委派机制缺乏有效的特权传播控制能力,委派机制的安全性也有待于严格的分析和证明.文中提出了基于角色的受限委派模型 RCDM,能够支持灵活的特权委派策略,并采用一种范围约束(scope constraint)结构控制特权传播的深度范围和广度范围.面向 RCDM 提出一种基于规则的满足性验证算法 C3A,基于逻辑程序语义理论分析了 C3A 算法关于 RCDM 的可靠性和完备性问题,从理论上证明了 RCDM 的安全性和可用性.

**关键词** 信任管理;委派;范围约束;规则;满足性验证

**中图法分类号** TP311

## Rule Based Constrained Delegation Framework

YIN Gang WANG Huai-Min SHI Dian-Xi TENG Meng

(School of Computer Science, National University of Defense Technology, Changsha 410073)

**Abstract** Delegation allows privilege propagation between principals, which is the core mechanism of trust management systems to enable multi-domain authorization. But unrestricted delegation may lead to privilege proliferation and breach the security of information systems. The delegation mechanisms in existing trust management systems are short of effective controllability on privilege propagation and their security need to be formally analyzed and proved. In this paper, a role-based constrained delegation model named RCDM (Role-based Constrained Delegation Model) is proposed, which supports flexible policies for delegation of authority and uses a scope constraint structure to control the depth scope and width scope of privilege propagation. A rule-based compliance checking algorithm named C3A is proposed for RCDM, the soundness and completeness of C3A with respect to RCDM are analyzed using the semantic theory of logic programs, which theoretically prove the security and availability of RCDM.

**Keywords** trust management; delegation; scope constraint; rule; proof of compliance

## 1 引 言

信任管理(Trust Management, TM)是面向多

域环境(multi-domain environment)的安全技术,旨在解决不同安全域的主体间的跨域授权问题.自 Blaze 于 1996 年提出 PolicyMaker<sup>[1]</sup>以来,出现了 KeyNote<sup>[2]</sup>, SPKI<sup>[3]</sup>, DL<sup>[4]</sup>, RT<sup>[5]</sup> 和 Cassandra<sup>[6]</sup>

收稿日期:2005-04-09;最终修改稿收到日期:2007-02-16. 本课题得到国家“九七三”重点基础研究发展规划项目基金(2005CB321804)、国家自然科学基金(90412011)和国家“八六三”高技术研究发展计划项目基金(2003AA115210, 2004AA112020)资助. 尹 刚,男,1975 年生,博士,讲师,研究方向为分布计算与信息安全. E-mail: jack\_nudt@yahoo.com.cn 王怀民,男,1962 年生,博士,教授,博士生导师,主要研究领域为分布计算、智能软件与信息安全. 史殿习,男,1966 年生,博士,副教授,研究方向为分布计算与自主计算. 滕 猛,男,1968 年生,博士,讲师,研究方向为分布计算与信息安全.

等诸多 TM 系统. 委派 (delegation) 是 TM 系统中特权传播的核心机制, 其基本思想是一个主体将某种特权传递给其它主体, 由其代表自己执行某种操作. 特权在主体间的多级委派形成委派链, 可实现灵活和可伸缩的跨域授权机制. 但是, 委派链中特权的可信传播依赖于主体间信任的传递性, 而信任关系的传递性假设过于乐观, 因此委派链可能导致特权扩散 (privilege proliferation), 使不可信主体获得越权访问的机会, 从而削弱甚至破坏信息系统的安全性. 现有 TM 系统中的委派机制难以避免特权扩散隐患, 原因主要包括两个方面: (1) 委派约束机制的控制粒度有限; (2) 缺乏对委派约束机制的严格的安全性分析.

现有 TM 系统主要通过布尔型约束和整型约束实现对委派链长度的控制. 布尔型控制包括两种策略: 不允许后续委派或对后续委派不加限制. 多数 TM 系统支持布尔型约束, 如 SPKI<sup>[3]</sup>, RT<sup>[5]</sup> 等. 例如, 对于 SPKI 凭证  $(I, S, A, D, V)$ , 表示主体  $I$  将特权  $A$  授予主体  $S$ , 有效期为  $V$ , 其中  $D$  有两种取值:  $T$  表示允许  $S$  继续将  $A$  传递给其它主体,  $F$  表示  $S$  不能再向外传播  $A$ . RT 则以较为隐含的方式支持布尔型约束, 例如 RT 凭证  $A.r \leftarrow B.r$  表示主体  $A$  将角色  $A.r$  的管理权威授予  $B$ , 且允许  $B$  以类似方式将其传播给其它主体, 而 RT 凭证  $B.r \leftarrow C$  表示  $B$  将  $B.r$  授予  $C$ , 但不允许  $C$  将  $B.r$  授予其它主体. 整型约束则可以控制委派链的长度, 提供了较细的约束粒度, DL<sup>[4]</sup> 和 Cassandra<sup>[6]</sup> 都支持整型约束. 但整型约束同布尔型约束一样, 假设某一深度范围内的信任关系可以传递 (布尔型约束可理解为取值 1 或  $\infty$  的整型约束). 这种假设过于乐观, 难以控制特权的实际传播范围 (不难分析, 深度超过 2 时对应的信任传递模型在现实社会中已较为少见). Cassandra 虽然支持整型约束, 但其需要在角色中扩展委派深度信息, 需要用户编写较为复杂的逻辑程序. 此外, DL 还支持一种宽度约束机制<sup>[4]</sup> 以限制后续委派主体的范围, 但其必须引入临时密钥签名宽度约束策略, 这可能会增加安全风险和策略复杂性.

另一方面, TM 系统用凭证 (经过签名的安全断言) 表达委派策略, 并采用通用的满足性证明 (Proof Of Compliance, POC) 算法回答授权查询: “凭证集  $C$  能否证明操作请求  $Q$  满足本地策略?” POC 算法的核心任务是在凭证集中寻找一条从权威源 (source of authority) 到操作请求者的有效的委派链, 其有效性包括多个方面, 例如委派链必须满足各级委派主体设置的委派约束等. 虽然 POC 算法提供

委派链搜索能力, 但没有验证搜索到的委派链的正确性, 对于“POC 算法导出的授权决策是否违反了委派约束条件”, 或者“满足委派约束条件的合法操作请求是否一定得到授权”等问题尚未得到回答. 对于安全系统而言, 这类问题必须得到严格的分析和证明. 下面以支持布尔型约束的 RT<sup>[2]</sup> 为例说明这类工作的必要性, 考虑以下 3 个 RT 凭证:

$$(C_1) A.r \leftarrow B.r;$$

$$(C_2) B.r \leftarrow C;$$

$$(C_3) C \xrightarrow{C \text{ as } B.r} D.$$

其中  $A$  允许  $B$  将  $A.r$  的管理权威传播给其它主体 (凭证  $C_1$ );  $B$  则不允许  $C$  传播  $B.r$  (凭证  $C_2$ );  $C$  把活跃角色“ $C \text{ as } B.r$ ”委派给  $D$ , 且允许  $D$  将“ $C \text{ as } B.r$ ”传播给其它主体 (凭证  $C_3$ ). 根据 RT 的语义<sup>[2]</sup>,  $C_1$ ,  $C_2$  和  $C_3$  将被映射为以下 DATALOG 规则:

$$(R_1) \text{ forRole}(\text{?}z, \text{?}y, A.r) \leftarrow \text{forRole}(\text{?}z, \text{?}y, B.r);$$

$$(R_2) \text{ forRole}(C, C, B.r);$$

$$(R_3) \text{ forRole}(D, C, B.r) \leftarrow \text{forRole}(C, C, B.r).$$

规则  $R_1, R_2, R_3$  可以导出  $\text{forRole}(D, C, A.r)$ , 这表明  $D$  能够以角色  $A.r$  访问资源. 这个例子说明, 序列“ $A \rightarrow B \rightarrow C \rightarrow D$ ”对应的委派链中, 虽然  $B$  不允许  $C$  继续传播角色  $B.r$ , 但 RT 的 POC 算法 (其 DATALOG 语义) 仍允许  $D$  使用  $A.r$ . 这是因为 RT 允许  $C$  把“ $C \text{ as } B.r$ ”自主地委派给  $D$ , 使得  $D$  可以间接地获得  $B.r$  角色 (规则  $R_2$ ), 并进而获得  $A.r$  角色 (规则  $R_1$ ). 虽然  $C_1$  和  $C_3$  分别属于权威委派和能力委派两种具有不同功能、不同层次的委派行为<sup>[5]</sup>, 但 RT 的 POC 算法并没有针对两类委派分别执行委派链搜索过程, 而是搜索涉及两类委派的混合委派链, 这可能使权威委派的约束机制 (如凭证  $C_2$ ) 失效, 很容易导致特权扩散. 出现这种结果的根本原因是 RT 缺乏形式化的委派约束模型, 难以对其 POC 算法的安全性进行严格分析. 其它支持委派约束的 TM 系统, 如 DL 和 Cassandra 等系统的 POC 算法也有待于进一步分析.

为充分利用委派机制的灵活性并尽可能避免委派引起的特权扩散隐患, 本文着重研究更为可控的委派模型及更为安全可靠的 POC 算法. 本文第 2 节首先提出一种基于角色的受限委派模型 RCDM, 基于范围约束控制其特权传播; 第 3 节给出 RCDM 的 POC 算法 C3A 并分析 C3A 的安全性和可用性; 第 4 节通过典型用例说明 RCDM 的用法; 第 5 节与相关工作进行比较分析; 第 6 节总结全文.

## 2 角色受限委派模型

### 2.1 基本模型

RCDM 定义了面向多域环境的访问控制模型, RCDM 包含 6 个基本集合:

- (1)  $E$  是主体(principal)集合;
- (2)  $N$  是名字(name)集合;
- (3)  $P \subseteq N$  是权限(permission)集合;
- (4)  $R \subseteq E \times N$  是角色(role)集合;
- (5)  $Nat$  是自然数集合;
- (6)  $SC \subseteq \wp(E) \times \wp(E) \times Nat$  是范围约束集合.

基于以上集合, RCDM 模型定义了 4 种授权关系:

- (1)  $PRA$  是权限角色指派关系;
- (2)  $URA$  是用户角色指派关系;
- (3)  $SRA$  是会话角色指派关系;
- (4)  $CDA$  是受限权威委派关系.

下面介绍基本的授权关系( $PRA, URA, SRA$ ), 并引入定义  $CDA$  所需的若干辅助性概念, 如权威委派和授权链等,  $CDA$  将在 2.2 节描述.

**定义 1(权限角色指派).** 权限角色指派  $(x, p, x.r) \in PRA \subseteq E \times P \times R$ , 表示主体  $x$  把权限  $p$  授予角色  $x.r$ . 角色  $x.r \in E \times N$  表示主体  $x$  定义的名字为  $r$  的角色. 主体  $x$  称为角色  $x.r$  的权威源. 令  $\omega = x.r$ ,  $\omega$  的权威源  $x$  记为  $\omega.soa$ .

**定义 2(用户角色指派).** 用户角色指派  $(x, y, \omega) \in URA \subseteq E \times E \times R$ , 表示主体  $x$  把角色  $\omega$  授予  $y$ .

$PRA$  和  $URA$  在传统 RBAC 模型的同名关系中扩展了策略声明人(指派的第 1 个元素).  $PRA$  中各指派的声明人与其角色的权威源相同, 因此  $PRA$  是一种局部授权关系, 而  $URA$  则是分布的.

**定义 3(会话角色指派).** 会话角色指派  $(x, y, x.r) \in SRA \subseteq E \times E \times R$ , 表示主体  $x$  把角色  $x.r$  授予主体  $y$  持有的会话.

$SRA$  用于描述系统的授权状态, 表示会话主体已经激活了某个角色.  $SRA$  是一种局部授权关系, 指派的声明人与其中角色的权威源相同.

在多域环境中, 权威源的分散性导致各权威源独立自治, 权威源需要以某种方式将域内的特权传递给其它权威源以实现协作授权. TM 系统多采用权威委派(delegation of authority)实现跨域授权<sup>[1-6]</sup>, 本文在此基础上提出一种基于角色的权威

委派机制, 作为定义  $CDA$  的概念基础.

**定义 4(权威委派).** 权威委派  $(x, y, \omega) \in DRA \subseteq E \times E \times R$  表示主体  $x$  把角色  $\omega$  的管理权威传递给主体  $y$ , 此后  $y$  可以把  $\omega$  指派给任何主体.

**定义 5(权威委派链).** 令  $\zeta$  是长度为  $n$  的权威委派序列, 当  $n > 0$  时, 令  $\zeta = \delta_1, \delta_2, \dots, \delta_n$ , 其中  $\delta_i = (x_{i-1}, x_i, \omega) \in DRA$ ,  $n \geq i \geq 1$ , 则  $\zeta$  是权威委派链, 此时  $\zeta$  记为  $(x_0 \rightarrow x_1 \rightarrow \dots \rightarrow x_n, \omega)$ , 简记为  $(x_0^n, \omega)$ <sup>①</sup>; 当  $n = 0$  时,  $\zeta$  是空序列, 记为  $\epsilon$ ,  $\epsilon$  是权威委派链.

**定义 6(授权链).** 令  $\xi$  是长度为  $n+1$  的序列. 当  $n > 0$  时, 令  $\xi = \zeta, \delta$ , 其中  $\zeta = (x_0^n, \omega)$  是权威委派链,  $\delta = (x_n, u, \omega) \in URA$ , 则  $\xi$  是授权链, 简记为  $(x_0^n \rightsquigarrow u, \omega)$ ; 当  $n = 0$  时, 如果  $\xi \in URA$ , 则  $\xi$  是授权链.

### 2.2 受限委派

RCDM 基于范围约束(scope constraint)控制特权传播范围, 其核心原理是根据授权链的授权踪迹控制特权传播范围.

**定义 7(授权踪迹).** 授权踪迹集合  $AT \subseteq \wp(E) \times \wp(E) \times Nat$ , 其中  $\wp(E)$  是  $E$  的幂集,  $Nat$  是自然数集. 授权踪迹  $t \in AT$  记为  $\text{trace}(da, du, dd)$ , 其中  $t.da, t.du, t.dd$  分别称为  $t$  的委派中介集、委派用户集和委派深度. 对于授权链  $\xi = (x_0^n \rightsquigarrow u, \omega)$ ,  $n > 0$ ,  $\xi$  的授权踪迹记为  $\text{trace}(\{x_0, x_1, \dots, x_n\}, \{u\}, n+1)$ ; 特别的, 对于授权链  $\xi = (x, u, \omega) \in URA$ , 其授权踪迹记为  $\text{trace}(\{\}, \{u\}, 0)$ .

**定义 8(范围约束).** 范围约束结构是三元组  $(SC, AT, \supseteq)$ , 其中  $SC \subseteq \wp(E) \times \wp(E) \times Nat$  是范围约束集合,  $AT$  是授权踪迹集合,  $\wp(E)$  是  $E$  的幂集,  $Nat$  是自然数集. 范围约束  $c \in SC$  记为  $\text{trace}(da, du, dd)$ , 其中  $c.da$  是  $c$  的委派中介范围,  $c.du$  是  $c$  的委派用户范围,  $c.dd$  是  $c$  的委派深度范围, 范围约束  $c$  也简称为约束  $c$ . 特别的,  $c.da$  和  $c.du$  取值为  $*$  时不产生任何约束效力.  $\supseteq \subseteq SC \times AT$  是  $SC$  和  $AT$  上的二元关系. 对于  $c \in SC, t \in AT, c \supseteq t$  当且仅当  $c.da \supseteq t.da, c.du \supseteq t.du, c.dd \geq t.dd$ .

下面结合范围约束将权威委派扩展为受限权威委派(constrained delegation of authority).

**定义 9(受限权威委派).** 受限权威委派  $\delta = (x, y, \omega, c) \in CDA \subseteq E \times E \times R \times SC$  表示主体  $x$  把

① 本文将序列  $\alpha_i \rightarrow \alpha_{i+1} \rightarrow \dots \rightarrow \alpha_j$  简记为  $\alpha_i^j$ , 其中  $0 \leq i \leq j$ .

角色  $\omega$  的权威委派给主体  $y$ , 并以  $c$  限制  $y$  的后续授权活动。

为了明确刻画范围约束对委派活动的控制机理, 这里引入受限权威委派链和受限授权链。

**定义 10**(受限权威委派链). 令  $\zeta$  是长度为  $n$  的受限权威委派序列. 当  $n > 0$  时, 令  $\zeta = \delta_1, \delta_2, \dots, \delta_n$ , 其中  $\delta_i = (x_{i-1}, x_i, c_i, \omega) \in CDA, n \geq i \geq 1$ , 则  $\zeta$  是权威委派链, 简记为  $(x_0^n, c_1^n, \omega)$ ; 当  $n = 0$  时,  $\zeta$  是空序列, 记为  $\epsilon$ ,  $\epsilon$  是受限权威委派链。

**定义 11**(受限授权链). 令  $\xi$  是长度为  $n+1$  的序列. 当  $n > 0$  时, 令  $\xi = \zeta, \delta$ , 其中  $\zeta = (x_0^n, c_1^n, \omega)$ ,  $\delta = (x_n, u, \omega) \in URA$ , 则  $\xi$  是受限授权链, 记为  $(x_0^n \rightsquigarrow u, c_1^n, \omega)$ ; 当  $n = 0$  时, 如果  $\xi \in URA$ , 则  $\xi$  是受限授权链。

给定受限授权链  $\xi = (x_0^n \rightsquigarrow u, c_1^n, \omega)$ . 令  $\xi' = (x_0^n \rightsquigarrow u, \omega)$ , 易知  $\xi'$  是授权链, 本文称  $\xi'$  是  $\xi$  的授权链,  $\xi'$  的授权踪迹称为  $\xi$  的授权踪迹。

**定义 12**(可信授权链). 令  $\xi$  是长度为  $n+1$  的受限授权链, 且 (1), (2) 成立:

(1) 当  $n > 0$  时, 令  $\xi = (x_0^n \rightsquigarrow u, c_1^n, \omega)$ ,  $\xi_i = (x_i^n \rightsquigarrow u, \omega)$ ,  $t_i$  是  $\xi_i$  的授权踪迹,  $\omega.soa = x_0$ , 如果  $c_i \supseteq t_i, i = 1, 2, \dots, n$ , 则称  $\xi$  是可信授权链;

(2) 当  $n = 0$  时, 令  $\xi = (x, u, \omega) \in AS.URA$ , 如果  $\omega.soa = x$ , 则称  $\xi$  是可信授权链。

可信授权链的直观含义是指: 授权链中的各级委派的声明人设置的约束都能得到满足。

### 2.3 授权系统

基于以上元素可以构造一个授权系统, 以表达授权策略和系统状态。

**定义 13**(授权系统). 授权系统  $AS$  是四元组  $(PRA, URA, CDA, SRA)$ , 本文将  $AS$  的各集合分别记为  $AS.PRA, AS.URA, AS.CDA$  和  $AS.SRA$ 。

$AS.PRA, AS.URA$  和  $AS.CDA$  主要由安全管理员维护, 例如添加或删除策略的操作.  $AS.SRA$  则由系统的授权服务维护, 用户登录或注销时将被更新.  $AS$  在收到授权查询时必须能够给出授权决策,  $AS$  支持两种授权查询: 角色激活查询和访问控制查询。

**定义 14**(授权查询). 角色激活查询  $raq = (x, y,$

$x.r) \in RAQ \subseteq E \times E \times R$  用于判断  $x$  是否允许  $y$  激活角色  $x.r$ . 访问控制查询  $acq = (x, y, p) \in ACQ \subseteq E \times E \times P$  用于判断  $x$  是否允许  $y$  访问权限  $p$  对应的资源。

RCDM 依据  $AS$  的授权公理回答角色激活查询和访问控制查询。

**公理 1**(角色激活公理). 对于角色激活请求  $raq = (s, u, s.r) \in RAQ \subseteq E \times E \times R$ ,  $AS \vdash raq$  表示  $s$  可以对  $raq$  授权.  $AS \vdash raq$  当且仅当  $AS$  中存在可信授权链  $(s \rightarrow x_1^n \rightsquigarrow u, c_1^n, s.r)$  或  $(s, u, s.r) \in AS.URA$ 。

给定授权系统  $AS$  和角色激活查询  $raq$ , 如果  $AS \vdash raq$ , 则用户激活相应角色后, 系统将把会话角色指派  $(s, u, s.r)$  添加至  $AS.SRA$ , 当用户注销时, 相应的指派信息将从  $AS.SRA$  中删除。

**公理 2**(资源访问公理). 对于访问控制请求  $acq = (s, u, p) \in ACQ \subseteq E \times E \times P$ ,  $AS \vdash acq$  表示  $s$  可以对  $acq$  授权.  $AS \vdash acq$  当且仅当  $(s, p, s.r) \in AS.PRA$  且  $(s, u, s.r) \in AS.SRA$ 。

资源访问公理用于控制主体对资源的访问, 给定授权系统  $AS$  和访问控制查询  $acq$ , 如果  $AS \vdash acq$ , 则  $acq$  对应的资源访问请求将被授权。

## 3 策略满足性验证算法

RCDM 可以通过授权系统  $AS$  描述授权策略和授权状态, 并给出了直观的授权决策判定公理, 但 RCDM 没有给出相应的判定算法. 本节给出一种基于规则的满足性检查算法 C3A (Constrained Compliance Checking Algorithm)。

### 3.1 C3A 算法

C3A 算法是一种基于无函数确定逻辑程序 (Function-free Definite Logic Program, FDLP) 理论<sup>[7]</sup> 的 POC 算法, 如图 1. 给定授权系统  $AS$  和授权查询  $Q$ , C3A 可以判断“ $AS \vdash Q$  是否成立”。C3A 是由规则组成的推导算法, 为了基于  $AS$  进行授权决策, 必须将  $AS$  映射为逻辑程序. 本文将  $AS$  对应的逻辑程序称为授权策略集。

```
( $\pi_1$ )  $x.canActivate(y, x.r) \leftarrow x.ura(y, x.r).$ 
( $\pi_2$ )  $x.canActivate(y, x.r) \leftarrow z.ura(y, x.r), x.rcda(z, x.r, trace(\{\}, \{y\}, 0)).$ 
( $\pi_3$ )  $x.rcda(y, x.r, t) \leftarrow x.cda(y, x.r, c), c \supseteq n-trace(t, y).$ 
( $\pi_4$ )  $x.rcda(y, x.r, t) \leftarrow z.cda(y, x.r, c), c \supseteq n-trace(t, y), x.rcda(z, x.r, n-trace(t, y)).$ 
( $\pi_5$ )  $x.canAccess(y, p) \leftarrow x.active(y, x.r), x.pra(p, x.r).$ 
```

图 1 基于规则的满足性验证算法 C3A

**定义 15(授权策略集  $\Delta$ ).** 给定  $AS$ ,  $\Delta$  是一段 FDLF 程序且  $\Delta = PS1 \cup PS2 \cup PS3 \cup PS4$ , 其中,

$$PS1 = \{x.pra(p, x, r) \mid (x, p, x, r) \in AS.PRA\};$$

$$PS2 = \{x.ura(y, z, r) \mid (x, y, z, r) \in AS.URA\};$$

$$PS3 = \{x.active(y, x, r) \mid (x, y, x, r) \in AS.SRA\};$$

$$PS4 = \{x.cda(y, \omega, c) \mid (x, y, \omega, c) \in AS.CDA\}.$$

$\Delta$  中各谓词的第一个参数作为前缀置于谓词名字的前面, 这种记法强调谓词的声明人, 适于表达多域环境中的授权断言. 例如  $x.ura(y, z, r)$  表示  $x$  将角色  $z.r$  授予  $y$ ,  $x$  是该授权断言 ( $x$  对其签名后则称为凭证) 的声明人.

C3A 将角色激活查询和访问控制查询映射为  $canActivate$  和  $canAccess$  谓词:  $x.canActivate(y, x, r)$  表示主体  $x$  允许主体  $y$  激活角色  $x, r$ ;  $x.canAccess(y, p)$  表示主体  $x$  允许主体  $y$  访问权限  $p$  对应的资源.

通过以上映射, C3A 的功能可以等价地表述为: 给定授权策略集  $\Delta$  和授权查询谓词  $Q$  ( $canActivate$  谓词或  $canAccess$  谓词), C3A 可以判断“策略集  $\Delta$  能否证明查询  $Q$ ”.

此外, 算法 C3A 还引入一个特殊的谓词符号  $rcda$  来递归的搜索可信授权链. 谓词  $x.rcda(y, z, r, t)$  表示主体  $x$  将角色  $z, r$  的权威传递给主体  $y$ ,  $t$  是授权踪迹, 其中  $t.da$  是从  $x$  到  $y$  的授权链的委派中介集 (包括  $y$ ),  $t.du$  包含得到  $y$  授权的主体 (即  $y$  将角色  $z, r$  指派给  $t.du$  中的主体),  $t.dd$  是这些授权链的共同的委派深度 (因为  $t.dd$  与  $t.du$  无关, 所以这些授权链具有相同的委派深度). C3A 中的谓词符号  $\sqsubseteq$  的实现算法参见定义 8, 该谓词可以分解为基本的集合关系谓词和自然数比较谓词.

C3A 包括 5 个核心规则, 本质上是一种可信授权链的搜索算法, 各规则参数都是变量 (用斜体小写字母表示). 其中  $\pi_1$  和  $\pi_2$  是  $canActivate$  的入口规则. 规则  $\pi_1$  的含义是: 如果  $x$  为  $y$  授予角色  $x, r$ , 则  $x$  允许  $y$  激活角色  $x, r$ . 规则  $\pi_2$  允许通过受限权威委派间接地获得激活角色的权力, 其直观含义可理解为, 如果满足下面两个条件  $x$  将允许  $y$  激活角色  $x, r$ : (1) 存在某个主体  $z$  为  $y$  授予角色  $x, r$ ; (2)  $x$  将  $x, r$  的权威可信的委派给  $z$  (直接或间接), 授权踪迹是  $trace(\{\}, \{y\}, 0)$ , 这相当于一个长度为 1 的授权链的授权踪迹.

规则  $\pi_3$  和  $\pi_4$  递规地定义了  $rcda$  的判定算法, 这是一种反向深度优先的可信授权链的搜索算法 (从授权链的最后一个委派中介主体开始, 逐步向前搜索委派中介), 在搜索过程中强制实现了可信授权链的性

质.  $\pi_4$  中的算子  $n\text{-trace}$  定义为:  $n\text{-trace}(t, y) \triangleq \text{trace}(t, da \cup \{y\}, t.du, t.dd + 1)$ . 算子  $n\text{-trace}$  将新搜索到的委派中介 ( $y$ ) 增加至授权踪迹中, 并将委派深度加 1. 规则  $\pi_5$  定义了访问控制查询的判定算法, 其中谓词  $x.active(y, x, r)$  对应于会话角色指派.

### 3.2 算法分析

C3A 是否符合 RCDM 模型是算法分析的核心问题: 一方面, 由 C3A 导出的授权决策符合  $AS$  授权公理; 同时, 符合  $AS$  授权公理的授权决策可由 C3A 导出. 由于  $AS$  角色激活公理涉及委派链的约束检查, 而资源访问公理较为简单, 本文主要分析 C3A 与角色激活公理的一致性问题, 这也称为 C3A 关于 RCDM 的可靠性和完备性问题.

本文将算法 C3A 包含的规则集合记为  $\Pi$ . 对于授权策略集  $\Delta$ ,  $\Delta \cup \Pi$  是一段无函数确定逻辑程序, 其中  $\Pi$  中规则是产生逻辑结论的基本机制. 本文将  $\Delta \cup \Pi$  称为策略基.

**定义 16(策略基  $\Sigma$ ).** 策略基  $\Sigma = \Delta \cup \Pi$ .

逻辑程序的语义是其所能推导出来的所有结论,  $\Sigma$  的语义可以由其最小 Herbrand 模型表示.

**定义 17( $\Sigma$  的模型及推论).** 策略基  $\Sigma$  的模型是指  $\Sigma$  的最小 Herbrand 模型, 记为  $M_\Sigma$ . 对任意事实  $f$ , 如果  $f \in M_\Sigma$ , 则  $f$  称是  $\Sigma$  的推论, 记为  $\Sigma \models f$ .

$M_\Sigma$  可由直接推论算子  $T_\Sigma$  迭代得到, 算子  $T_\Sigma$  定义如下<sup>[7]</sup>.

**定义 18(直接推论算子  $T_\Sigma$ ).** 给定策略基  $\Sigma$ , 映射  $T_\Sigma(I): \wp(H_\Sigma) \rightarrow \wp(H_\Sigma)$  是  $I$  的直接推论算子, 其中  $I$  是  $\Sigma$  的 Herbrand 解释,  $H_\Sigma$  是  $\Sigma$  的 Herbrand 基,  $\wp(H_\Sigma)$  是  $H_\Sigma$  的幂集.  $T_\Sigma$  定义为

$$T_\Sigma(I) = \{x.a(\vec{e}) \mid x.a(\vec{e}) \leftarrow x_1.a_1(\vec{e}_1), \dots, x_n.a_n(\vec{e}_n) \in G(\Sigma) \text{ and for all } i \in [1..n], x_i.a_i(\vec{e}_i) \in I\},$$

其中,  $G(\Sigma)$  是  $\Sigma$  中规则在  $U_\Sigma$  上的所有基例 (ground instance),  $U_\Sigma$  是  $\Sigma$  的 Herbrand 域. 算子  $T_\Sigma$  的作用通过序数幂 (ordinal power) 实现.

**定义 19(算子  $T_\Sigma$  的序数幂).**  $T_\Sigma$  的序数幂是一种迭代操作, 定义如下:

$$T_\Sigma \uparrow 0 = \emptyset,$$

$$T_\Sigma \uparrow n = T_\Sigma(T_\Sigma \uparrow (n-1)),$$

$$T_\Sigma \uparrow \omega = \text{lub}\{T_\Sigma \uparrow n \mid n = 0, 1, 2, \dots\},$$

其中,  $T_\Sigma$  是  $\wp(H_\Sigma)$  上的单调二元关系, 又  $\wp(H_\Sigma)$  关于集合包含关系为完备格 (complete lattice), 因此有以下一般性结论<sup>[7]</sup>.

**引理 1.**  $T_\Sigma$  是连续单调映射且  $T_\Sigma$  存在最小不动点, 记为  $\text{lfp}(T_\Sigma)$ . 此外,  $T_\Sigma$  具有以下性质:

- (1) 最小不动点  $lfp(T_{\Sigma}) = M_{\Sigma}$ ;
- (2) 对任意  $n > 0$ ,  $T_{\Sigma} \uparrow n \subseteq M_{\Sigma}$ ;
- (3) 如果  $f \in M_{\Sigma}$ , 则存在  $n > 0$ , 使得  $f \in T_{\Sigma} \uparrow n$ .

给定策略基  $\Sigma$ , 其授权决策的判定规则为: 如果  $\Sigma \models x.canActivate(y, x, r)$ , 则  $x$  允许  $y$  激活角色  $x.r$ ; 如果  $\Sigma \models x.canAccess(y, p)$ , 则  $x$  允许  $y$  访问  $p$  对应的资源. 为了分析 C3A 的可靠性和完备性, 这里先引入语义授权系统的概念.

**定义 20**(语义授权系统  $AS_{\Sigma}$ ). 策略基  $\Sigma$  的语义授权系统  $AS_{\Sigma}$  定义如下:

$$\begin{aligned} AS_{\Sigma}.PRA &= \{(x, p, x.r) \mid x.pra(p, x.r) \in M_{\Sigma}\}; \\ AS_{\Sigma}.URA &= \{(x, y, \omega) \mid x.ura(y, \omega) \in M_{\Sigma}\}; \\ AS_{\Sigma}.SRA &= \{(x, y, x.r) \mid x.active(y, x.r) \in M_{\Sigma}\}; \\ AS_{\Sigma}.CDA &= \{(x, y, \omega, c) \mid x.cda(y, \omega, c) \in M_{\Sigma}\}. \end{aligned}$$

不难分析, 根据上述构造方法,  $M_{\Sigma}$  唯一对应一个语义授权系统  $AS_{\Sigma}$ .  $M_{\Sigma}$  中可能包括  $canActivate$  和  $canAccess$  事实, 这两类事实是否与  $AS_{\Sigma}$  的授权决策公理一致(参见公理 1 和 2), 是验证 C3A 正确性的关键(如前所述, 本文主要考虑公理 1). 下面先给出算子  $T_{\Sigma}$  的几个性质.

**引理 2.**  $T_{\Sigma} \uparrow 1 \cap \{\beta \mid \beta \in H_{\Sigma}, \beta.name \in \{rcda, canActivate, canAccess\}\} = \emptyset$ , 其中  $H_{\Sigma}$  是  $\Sigma$  的 Herbrand 基,  $\beta.name$  是基例  $\beta$  的谓词符号.

**证明.** 根据 C3A,  $rcda$  事实由  $cda$  事实导出( $\pi_3$  和  $\pi_4$ ), 又因为  $T_{\Sigma} \uparrow 0 = \emptyset$ , 根据  $T_{\Sigma}$  定义可知  $T_{\Sigma} \uparrow 1$  中不存在  $rcda$  事实. 同理可证算子  $T_{\Sigma}$  关于  $canActivate$  和  $canAccess$  具有同样的性质. 证毕.

引理 2 说明经过一次迭代,  $T_{\Sigma} \uparrow 1$  中不包含  $rcda, canActivate$  和  $canAccess$  事实.

**引理 3.** 给定策略基  $\Sigma$ , 令  $\delta = s.rcda(a, s.r, t)$ , 则  $\delta \in M_{\Sigma}$  当且仅当  $M_{\Sigma}$  中存在序列  $\xi = s.cda(a_1, s.r, c_1), \dots, a_{m-1}.cda(a_m, s.r, c_m)$  且  $c_i \sqsupseteq t_i$ , 其中  $t_i = trace(t.da \cup A_i, t.du, t.dd + (m - i + 1))$ ,  $A_i = \{a_i, \dots, a_m\}$ ,  $a_m = a$ ,  $m \geq 1$ .

**证明.**

( $\Rightarrow$ ) 由引理 1 和  $\delta \in M_{\Sigma}$  可知存在  $n > 0$ ,  $\delta \in T_{\Sigma} \uparrow n$ , 并且根据引理 2 可知  $n \geq 2$ , 下面对  $n$  归纳证明. 对于  $n=2$ , 如果  $\delta \in T_{\Sigma} \uparrow 2$ , 则根据  $T_{\Sigma}$  定义知  $G(\Sigma)$  中存在基例  $g = s.rcda(a, s.r, t) \leftarrow x_1.\alpha_1(\bar{e}_1), \dots, x_s.\alpha_s(\bar{e}_s)$  且  $x_i.\alpha_i(\bar{e}_i) \in T_{\Sigma} \uparrow 1, i=1, 2, \dots, s$ . 由引理 2 可知  $\alpha_i \neq rcda$ , 因此  $g$  是  $\pi_3$  的基例且  $g = s.rcda(a, s.r, t) \leftarrow s.cda(a, s.r, c), c \sqsupseteq n-trace(t, a)$ . 因此  $s.cda(a, s.r, c) \in T_{\Sigma} \uparrow 1 \subseteq M_{\Sigma}$ . 令  $c_1 = c, t_1 =$

$trace(t.da \cup \{a\}, t.du, t.dd + 1) = n-trace(t, a)$ , 即  $c_1 \sqsupseteq t_1$ , 必要性得证.

假设  $n = k > 2$  时, 必要性成立, 考虑  $\delta \in T_{\Sigma} \uparrow (k+1)$  的情况. 由  $T_{\Sigma}$  定义可知  $G(\Sigma)$  中存在  $g = s.rcda(a, s.r, t) \leftarrow x_1.\alpha_1(\bar{e}_1), \dots, x_s.\alpha_s(\bar{e}_s)$  且  $x_i.\alpha_i(\bar{e}_i) \in T_{\Sigma} \uparrow k, i=1, 2, \dots, s$ . 显然  $g$  是  $\pi_3$  或  $\pi_4$  的基例. 如果  $g$  是  $\pi_3$  的基例, 则与  $n=2$  的情况类似, 必要性可证. 如果  $g$  是  $\pi_4$  的基例, 令  $t' = n-trace(t, a)$ , 则  $g = s.rcda(a, s.r, t) \leftarrow e.cda(a, s.r, c), c \sqsupseteq t', s.rcda(e, s.r, t')$  且  $\{e.cda(a, s.r, c), c \sqsupseteq t', s.rcda(e, s.r, t')\} \subseteq T_{\Sigma} \uparrow k \subseteq M_{\Sigma}$ , 由  $s.rcda(e, s.r, t') \in M_{\Sigma}$  和归纳假设知  $M_{\Sigma}$  中存在序列  $s.cda(e_1, s.r, c_1), \dots, e_{h-1}.cda(e_h, s.r, c_h)$ , 其中  $e_h = e, c_i \sqsupseteq t_i, t_i = trace(t'.da \cup E_i, t'.du, t'.dd + (h - i + 1))$ ,  $E_i = \{e_i, \dots, e_h\}, h \geq i \geq 1$ . 令  $m = h + 1, a_j = e_j, a_m = a, m - 1 \geq j \geq 1$ , 则有  $t_i = trace(t.da \cup \{a_i, \dots, a_m\}, t.du, t.dd + (m - i + 1)), t_m = t', m - 1 \geq i \geq 1$ . 再令  $c_m = c$ , 由  $\{e.cda(a, s.r, c), c \sqsupseteq t'\} \subseteq M_{\Sigma}$  可知  $M_{\Sigma}$  中存在  $\xi$  且  $c_i \sqsupseteq t_i, m \geq i \geq 1$ , 必要性得证.

( $\Leftarrow$ ) 已知  $S = \{s.cda(a_1, s.r, c_1), \dots, a_{m-1}.cda(a, s.r, c_m)\} \subseteq M_{\Sigma}$  可对  $m$  进行归纳证明. 当  $m=1$  时, 由  $S = \{s.cda(a, s.r, c_1)\} \subseteq M_{\Sigma}$  可知存在  $n > 0, s.cda(a, s.r, c_1) \in T_{\Sigma} \uparrow n$ . 令  $t' = n-trace(t, a)$ , 因为  $G(\Sigma)$  中存在  $\pi_3$  的基例  $s.rcda(a, s.r, t) \leftarrow s.cda(a, s.r, c_1), c_1 \sqsupseteq t'$ , 且由已知条件  $t_1 = trace(t.da \cup \{a\}, t.du, t.dd + 1) = t'$  且  $c_1 \sqsupseteq t_1$ , 可得  $c_1 \sqsupseteq t'$ , 因此  $s.rcda(a, s.r, t) \in T_{\Sigma} \uparrow (n+1) \subseteq M_{\Sigma}$ , 充分性得证.

假设  $m = k > 1$  时, 充分性成立, 考虑  $m = k + 1$  的情况. 由题设知  $S' = \{s.cda(a_1, s.r, c_1), \dots, a_k.cda(a_{k+1}, s.r, c_{k+1})\} \subseteq M_{\Sigma}$  且  $c_i \sqsupseteq t_i$ , 其中  $t_i = trace(t.da \cup A_i, t.du, t.dd + (k + 1 - i + 1))$ ,  $A_i = \{a_i, \dots, a_{k+1}\}, a_{k+1} = a, k + 1 \geq i \geq 1$ . 令  $t' = n-trace(t, a)$ , 显然有  $t_i = trace(t'.da \cup A_i, t'.du, t'.dd + (k - i + 1)), k + 1 \geq i \geq 1$ , 由归纳假设可得  $s.rcda(a_k, s.r, t') \in M_{\Sigma}$ . 又因为  $G(\Sigma)$  中存在  $\pi_4$  的基例  $g = s.rcda(a, s.r, t) \leftarrow a_k.cda(a_{k+1}, s.r, c_{k+1}), c_{k+1} \sqsupseteq t', s.rcda(a_k, s.r, t')$ , 由  $c_{k+1} \sqsupseteq t_{k+1}$  和  $t_{k+1} = t'$  可知  $c_{k+1} \sqsupseteq t'$ , 因此有  $s.rcda(a, s.r, t) \in M_{\Sigma}$ , 充分性得证. 证毕.

**定理 1**(C3A 的可靠性). 给定策略基  $\Sigma$ , 如果  $\Sigma \models s.canActivate(u, s.r)$  则  $AS_{\Sigma} \models (s, u, s.r)$ .

**证明.** 令  $\delta = s.canActivate(u, s.r)$ . 由  $\Sigma \models \delta$

知  $\delta \in M_{\Sigma}$ , 由引理 1 知存在自然数  $w, \delta \in T_{\Sigma} \uparrow w$ , 根据公理 1, 可靠性可表述为: 如果  $\delta \in T_{\Sigma} \uparrow w$ , 则  $AS_{\Sigma}$  中存在可信授权链  $\xi, \xi = (s \rightarrow x_1^m \dots u, c_1^m, s, r)$  或  $\xi = (s, u, s, r), m \geq 1$ .

由引理 2 知  $w \geq 2$ . 由  $\delta \in T_{\Sigma} \uparrow w$  可知  $G(\Sigma)$  中存在基例  $g = s.canActivate(us.r) \leftarrow x_1. \alpha_1(\bar{e}_1), \dots, x_s. \alpha_s(\bar{e}_s)$  且  $x_i. \alpha_i(\bar{e}_i) \in T_{\Sigma} \uparrow (w-1), i=1, 2, \dots, s$ . 显然  $g$  是  $\pi_1$  或  $\pi_2$  的基例. 如果  $g$  是  $\pi_1$  的基例, 可令  $g = s.canActivate(u, s, r) \leftarrow s.ura(u, s, r)$ , 且  $s.ura(a, s, r) \in T_{\Sigma} \uparrow (w-1) \subseteq M_{\Sigma}$ , 由  $AS_{\Sigma}$  定义和定义 12 易知  $AS_{\Sigma}$  中存在可信授权链  $(s, u, s, r)$ , 命题得证. 如果  $g$  是  $\pi_2$  的基例, 令  $g = s.canActivate(u, s, r) \leftarrow a.ura(u, s, r), s.rcda(a, s, r, t)$ , 其中  $t = trace(\{\}, \{u\}, 0)$ . 由  $T_{\Sigma}$  定义知  $\{a.ura(u, s, r), s.rcda(a, s, r, t)\} \subseteq T_{\Sigma} \uparrow (w-1) \subseteq M_{\Sigma}$ . 由  $s.rcda(a, s, r, t) \in M_{\Sigma}$  和引理 3 可知  $M_{\Sigma}$  中存在序列  $a_0.cda(a_1, s, r, c_1), \dots, a_{m-1}.cda(a_m, s, r, c_m)$  且  $c_i \supseteq t_i$ , 其中  $t_i = trace(t.da \cup \{a_i, \dots, a_m\}, t.du, t.dd + (m-i+1))$ ,  $a_0 = s, a_m = a, m \geq i \geq 1$ . 由  $AS_{\Sigma}$  定义知  $(a_k, a_{k+1}, s, r, c_{k+1}) \in AS_{\Sigma}.CDA, (a, u, s, r) \in AS_{\Sigma}.URA, m-1 \geq k \geq 0$ . 因此  $AS_{\Sigma}$  中存在受限授权链  $\xi = (s \rightarrow a_1^m \dots u, c_1^m, s, r)$ . 令  $\xi_i = (x_i^m \dots u, s, r), m \geq i \geq 1$ , 易知  $t_i$  是  $\xi_i$  的授权踪迹, 由  $c_i \supseteq t_i$  和定义 12 可知  $\xi$  是可信授权链, 命题得证.

证毕.

**定理 2**(C3A 的完备性). 给定策略基  $\Sigma$ , 如果  $AS_{\Sigma} \not\vdash (s, u, s, r)$ , 则  $\Sigma \models s.canActivate(u, s, r)$ .

**证明.** 令  $\delta = s.canActivate(u, s, r)$ . 由  $AS_{\Sigma} \not\vdash (s, u, s, r)$  可知  $(s, u, s, r) \in AS.URA$  或者  $AS_{\Sigma}$  中存在可信授权链  $(s \rightarrow a_1^m \dots u, c_1^m, s, r), m \geq 1$ . 对于第 1 种情况, 由  $AS_{\Sigma}$  定义知  $s.ura(u, s, r) \in M_{\Sigma}$ , 由  $\pi_1$  立得  $\Sigma \models s.canActivate(u, s, r)$ . 对于第 2 种情况, 由定义 12 可知  $S = \{(s, a_1, s, r, c_1), \dots, (a_{m-1}, a_m, s, r, c_m)\} \subseteq AS_{\Sigma}.CDA, (a_m, u, s, r) \in AS_{\Sigma}.URA$  且  $c_i \supseteq t_i$ , 其中  $t_i = trace(\{a_i, \dots, a_m\}, \{u\}, m-i+1), m \geq i \geq 1$ . 由  $AS_{\Sigma}$  定义知  $M_{\Sigma}$  中存在序列  $s.cda(a_1, s, r, c_1), \dots, a_{m-1}.cda(a_m, s, r, c_m), a_m.ura(u, s, r)$ . 令  $t = trace(\{\}, \{u\}, 0)$ , 则  $t_i = trace(t.da \cup \{a_i, \dots, a_m\}, t.du, t.dd + (m-i+1)), m \geq i \geq 1$ . 由引理 3 可知  $s.rcda(a_m, s, r, t) = \delta \in M_{\Sigma}$ . 因为  $G(\Sigma)$  中存在  $\pi_2$  的基例  $g = s.canActivate(u, s, r) \leftarrow a_m.ura(u, s, r), s.rcda(a_m, s, r, t)$ , 可知  $\delta \in M_{\Sigma}$ , 即  $\Sigma \models s.canActivate(u, s, r)$ , 命题得证.

证毕.

C3A 的可靠性和完备性定理说明 C3A 与

RCDM 模型是一致的. 由于篇幅限制, 本文分析了 C3A 与较为关键的 AS 角色激活公理的一致性问题. 直观上讲, C3A 的可靠性是指推导出来的授权查询一定满足授权系统 AS 的授权公理; C3A 的完备性是指 AS 授权公理的授权查询请求必能得到证明. POC 算法的可靠性分析有助于验证委派约束是否真正得到满足, 从而增强了系统安全性; 完备性分析则有助于确保授权系统的可用性, 系统应授权合理的安全请求.

## 4 应用举例

本节通过典型跨域授权案例介绍 RCDM 的使用方法和流程. 为了便于描述, 这里直接采用逻辑程序描述授权策略(参见 3.1 节).

假设 ELib 是基于 Internet 的综合性数字图书馆(如 CNKI 等), ELib 同大学 TechU 达成如下合作协议: ELib 为 TechU 的学生提供优惠(ELib.disc), 但要求学生是 ELib 引用文献的作者(ELib.author). ELib 认可 TechU 各分校(TechU.branch)的学籍管理资格. ELib 的上述策略可以描述如下:

(1) ELib.cda(TechU, ELib.disc,  $c_1$ ).

其中  $c_1 = trace(TechU.branch, ELib.author, 2)$ . 这里用 TechU.branch 和 ELib.author 表示 TechU 的分校集合和 ELib 作者的集合(由于主体的抽象表示不是 RCDM 关注的问题, 本文假设定义策略时 TechU.branch 和 ELib.author 代表的主体集合可以枚举出来, 这可以借助外部机制实现).

SchA 是 TechU 的分校, TechU 授权 SchA 管理 TechU 学生学籍, 但不允许 SchA 授权其它组织管理其学籍. TechU 将学生优惠的指派权力委派给 SchA, 且也允许 SchA 将此特权传播给其它机构或个人, 上述策略描述如下:

(2) TechU.cda(SchA, TechU.student,  $c_2$ ).

(3) TechU.cda(SchA, ELib.disc,  $c_3$ ).

其中  $c_2 = c_3 = trace(*, *, 1)$ . Alice 是 SchA 的学生且被 SchA 授权享有 ELib 的优惠. 此外, Alice 还有文章被 ELib 引用.

(4) SchA.ura(Alice, TechU.student).

(5) SchA.ura(Alice, ELib.disc).

(6) ELib.ura(Alice, ELib.author).

ELib 通过规则(1)把 ELib.disc 的管理权威传递给 TechU, 并附加委派约束: (i) 不允许 TechU 把该特权委派给其分校以外的机构; (ii) 委派深度

限制为 2; (iii) 最终获得优惠的用户只能是 ELib 收录文献的作者 (ELib.author). 规则 (2), (3) 分别定义了 TechU 对其分校 SchA 的两个权威委派. 因为规则 (1) 中的委派约束, TechU 和 SchA 的授权活动将无法超出上述约束的范围. TechU 将深度约束设置为 1, 不允许 SchA 进一步委派, 此时委派中介范围和委派用户范围无需设置 (置为 \* 即可).

当 Alice 希望从 ELib 下载文献时, 为了享受优惠, 她必须先向 ELib 注册, 以激活 ELib.disc 角色. 授权服务将注册请求转换为角色激活查询:

(7) ?  $\leftarrow$  ELib.canActivate(Alice, ELib.disc).

根据 C3A 算法, ELib 将允许 Alice 激活 ELib.disc 角色的请求, 此时 ELib 的授权服务将在策略集中增加事实: ELib.active(Alice, ELib.disc), 表示 Alice 已激活 ELib.disc 角色.

Alice 成功登录 ELib 后, 便可以下载选定的电子文献, 但是下载操作必须得到 ELib 授权服务的许可. 假设下载操作为 download\_disc, 下载文献的名称没有限制, 则授权服务将该操作转换为访问控制查询:

(8) ?  $\leftarrow$  ELib.canAccess(Alice,  $p$ ).

其中权限  $p = \text{download\_disc}$ , 表示以某种折扣下载文献. 这里假定 ELib 已定义如下策略:

(9) ELib.pra(download\_disc, ELib.disc).

根据 C3A 语义, 访问控制查询 (8) 可被证明, ELib 将允许 Alice 的下载操作.

5 相关工作分析

委派约束是 TM 系统面临的难点问题, 现有 TM 系统主要关注委派深度的约束机制 (见第 1 节), 难以适应多域环境中的应用系统. RCDM 提出的范围约束根据授权踪迹限制委派过程中的特权传播范围, 不仅支持委派深度约束 (纵向范围约束), 而且还可以控制特权向委派中介和委派用户的传播 (横向范围约束), 是一种细粒度且较为灵活的委派约束机制. 此外, RCDM 也提供了较为灵活的委派机制, 其特权传播方式类似于 KeyNote<sup>[2]</sup> 和 SPKI<sup>[3]</sup> 等能力委派系统, 但委派的特权是角色的管理权威, 而不是权限. RCDM 明确区分角色的管理权力与角色的使用权力, 且仅允许角色的管理权力在主体间传递, 这有助于避免由于特权类型的混淆造成的安全隐患.

委派策略在传统 RBAC 领域也一直得到关注, 如 RDF<sup>[8]</sup>, RDM2000<sup>[9]</sup>, CRDM<sup>[10]</sup> 等. RDF 是

Barka 和 Sandhu 等人提出的一种概念性角色委派框架, 对委派特性进行系统性研究和归类, 这对委派约束机制的设计具有一定的参考价值. RDM2000 支持多级角色委派, 并支持委派深度约束, 但是 RDM2000 并没有给出深度约束的验证算法. CRDM 支持面向平坦角色的临时性限制和常规角色关联性限制, 尚不支持多级委派约束<sup>[10]</sup>. 角色委派主要面向组织内部, 安全管理员和用户修改中心策略库来表达不同层次的委派策略. 角色委派模型是否适用于多域环境还有待进一步研究. Bandmann 等人提出的 CDM 模型<sup>[11]</sup> 面向规模较大、具有多级管理层次的组织. CDM 基于链式约束 (chain constraint) 限制委派树的结构, 但链式约束过于复杂, 作者尚未给出合适的计算模型. 目前, RCDM 模型尚未考虑时间因素, 由于时间与其它元素相对独立, 我们将在今后的研究中探讨时间约束问题.

现有 TM 系统主要关注策略描述机制和 POC 算法<sup>[1-6]</sup>, 尚没有严格刻画 POC 算法的核心计算模型和原理, 因此 POC 算法的正确性和有效性难以得到验证. 事实上, 一种提高系统安全性和可靠性的有效途径是采用两种不同的方法描述系统, 然后证明这两种方法的一致性. 本文给出的 RCDM 模型给出了直观的授权模型, 有助于清晰地刻画系统的工作原理, 这对系统合理性的评价具有重要作用. 本文给出了基于逻辑程序的 RCDM 策略描述方法, 据此提出的 C3A 算法本质上是一种面向 RCDM 的 POC 算法, 这两部分内容可以构成一个独立的 TM 系统. C3A 的可靠性和完备性定理证明了 C3A 与 RCDM 的一致性, 这对确保 C3A 的安全性和可用性具有重要作用 (事实上, 作者在分析 C3A 的过程中发现了使用中尚未发现的问题), 这也是对 TM 系统 POC 算法进行安全性分析的一种新的探索.

6 结 论

“信任未必完全可信”. 现有 TM 系统难以有效避免因“可信实体”的非预期授权而导致的特权扩散. 本文提出的 RCDM 模型为限制特权扩散提供了灵活有效的控制机制, 基于逻辑程序语义理论深入研究了 RCDM 的 POC 算法 C3A 的安全性问题; 严格证明了 C3A 算法关于 RCDM 模型的可靠性和完备性定理. 目前我们已将 C3A 算法集成到 StarBus<sup>+</sup> 的分布式授权服务<sup>[12]</sup> 中. 初步实践表明, RCDM 能够为网络环境下跨域授权和访问控制提供可



靠有效的支持.

参 考 文 献

[1] Blaze M, Feigenbaum J, Lacy J. Decentralized trust management//Proceedings of the 17th Symposium on Security and Privacy. Oakland, 1996: 164-173

[2] Blaze M, Feigenbaum J, Ioannidis J, Keromytis A D. The keynote trust-management system. Version 2. IETF RFC 2704, September 1999

[3] Ellison C M, Frantz B, Lampson B, Rivest R, Thomas B M, Ylonen T. SPKI certificate theory. RFC 2693, 1998

[4] Li Ning-Hui, Grosf B N, Feigenbaum J. Delegation logic: A logic-based approach to distributed authorization. ACM Transactions on Information and System Security, 2003, 6 (1): 128-171

[5] Li Ning-Hui, Mitchell J C, Winsborough W H. Design of a role-based trust management framework//Proceedings of the IEEE Symposium on Security and Privacy. Berkeley, California, USA, 2002: 114-130

[6] Becker M Y, Sewell P. Cassandra: Flexible trust management, applied to electronic health records//Proceedings of the 17th IEEE Computer Security Foundations Workshop (CSFW'04). Pacific Grove, California, USA, 2004: 139-

154

[7] Lloyd J W. Foundations of Logic Programming. 2nd, Extended Edition. Berlin: Springer, 1987

[8] Barka E, Sandhu R S. Framework for role-based delegation models//Proceedings of the 16th Annual Computer Security Applications Conference (ACSAC 2000), New Orleans, Louisiana, USA. IEEE Computer Society 2000, 2000: 168-176

[9] Zhang L H, Ahn G-J, Chu B-T. A rule-based framework for role-based delegation//Sandhu R S, Jaeger T eds. Proceedings of the 6th ACM Symposium on Access Control Models and Technologies. New York: ACM Press, 2001: 153-162

[10] Xu Zhen, Li Lan-Feng, Feng Deng-Guo. A constrained role-based delegation model. Journal of Software, 2005, 16(5): 970-978(in Chinese with English abstract)  
(徐震, 李澜, 冯登国. 基于角色的受限委托模型. 软件学报, 2005, 16(5): 970-978)

[11] Bandmann O, Damy M, Firozabadi Babak Sadighi. Constrained delegation//Proceedings of the 2002 IEEE Symposium on Security and Privacy (S&P'02). Berkeley, California, USA, 2002: 131-142

[12] Wang Huai-Min, Wang Yu-Feng, Tang Yang-Bin. StarBus<sup>+</sup>: Distributed object middleware practice for Internet computing. Journal of Computer Science and Technology, 2005, 20 (4): 542-551



**YIN Gang**, born in 1975, Ph. D. , lecturer. His research interests include distributed computing and information security.

**WANG Huai-Min**, born in 1962, Ph. D. , professor, Ph. D. supervisor. His research interests include distributed computing, intelligent software and information security.

**SHI Dian-Xi**, born in 1966, Ph. D. , associate professor. His research interests include distributed computing and autonomic computing.

**TENG Meng**, born in 1968, Ph. D. , lecturer. His research interests include distributed computing and information security.

Background

This work is supported by the National Basic Research Program(973 Program) of China under grant No. 2005CB321804, National Natural Science Foundation of China under grant No. 90412011, and the National High Technology Research and Development Program (863 Program) of China under grant Nos. 2003AA115210, 2004AA112020. Most of the new ideas in this paper arise from the development of the security service in StarBus<sup>+</sup>, a middleware platform developed by the authors during last 10 years and configured as general distributed computing platform for many projects and software products in China. Besides providing secure communication and identity-based access control, the security service

also aims to enable authorization across different security domains with different authorities. Delegation is one of the key mechanisms for decentralized authorization which is also hot spot in trust management(TM) systems. It is very important to control the privilege proliferation during delegation, which is still a difficult problem in existing TM systems. The authors are trying to propose more flexible and controllable delegation models to fulfill those special requirements, as shown in this paper. To integrate the delegation model into StarBus<sup>+</sup> security service, the authors designed a novel security architecture named middleware access control management, which will be reported in near future.