基于双边意愿的委托授权协商模型*

高荣瑞, 孙宇清+

(山东大学 计算机科学与技术学院,山东省济南市 250101)

A Delegation Agreement Model Based on Bilateral Intention

Gao Rong-Rui, Sun Yu-Qing+

(Department of Computer Science and Technology, Shandong University, Jinan 250101, China)

+ Corresponding author: Phn +86-531-88391810, Fax +86-531-88392498, E-mail:sun_yuqing@sdu.edu.cn, http://www.sdu.edu.cn

Abstract: Delegation is a major mechanism to achieve flexible authorization. As an important way to realize delegation, bilateral agreement based delegation takes into account the intentions of both delegator and delegatee, which can effectively increase the utilization ratio of system resources, balance the system load and improve work efficiency. In response to the requirements for bilateral agreement based delegation in practical application, a bilateral intention based delegation agreement model is proposed on the basis of the intentions of the delegator and delegatee. It considers essential factors during the process to set up delegation and defines the conception of intention expression. Then, the intention matching algorithm is proposed which realizes delegation agreement based on bilateral intention by matching the intention expressions using the Boolean expression conjunction and reduction. At last, the way to apply bilateral intention based delegation agreement model to both centralized and distributed system is discussed.

Key words: delegation; agreement; access control

摘 要: 委托授权是实现灵活授权管理的一种重要机制.双边委托授权作为委托授权的一种重要方式,在委托授权的建立过程中考虑了委托者和受托者双方的意愿,能够有效的提高系统资源的利用率,平衡系统负载,提升系统工作效率.针对实际应用中对双边委托的需求,首先建立了基于双边意愿的委托授权协商模型,探讨了在委托授权建立过程中需要满足的因素,提出了意愿表达式的概念.其次,给出了意愿匹配算法, 通过意愿表达式的合取和约简对委托意愿和受托意愿进行匹配,实现了基于双边意愿的委托授权协商.最后,讨论了在集中式和分布式环境下基于双边意愿的基于双边意愿的委托授权协商模型的实现机制.

关键词: 委托授权;协商;访问控制

中图法分类号: **TP309** **文献标识码**: A

1 引言

授权是一种控制信息资源访问行为的机制,通过限定用户的访问行为达到保护敏感信息的目的,实施合适的授权管理是构建安全的信息系统的有效保证.委托授权是一种重要的授权方式,指用户将自己拥有的权限转移给他人,从而使后者可以替代自己或者协助自己执行有关操作,其中将权限委托出去的用户被称为委托者(delegator),接受委托授权的用户被称为受托者(delegatee).在实际应用中,委托授权有着十分重要的作用[1],如当系统中特定用户因故无法执行任务时,为保证任务的顺利执行,可以将其拥有的权限委托授权给合适的替代者从而保证工作的继续;在另外的一些情况下,为实现各部门间或者各用户间的协作,可能需要通过委托授权的方式将权限临时委托授权给协作用户.

为了满足对委托授权进行管理的需求,许多研究工作对如何管理委托授权进行了研究.现有的委托授权研究主要关注如何委托授权规则[1,2,3,4],如何保证委托授权安全性[5,6]和如何制定委托授权条件[7,8]等几个方面的问题.Jacques Wainer[7]给出了一种基于RBAC 实现用户到用户委托授权的方法,该方法不仅考虑了较高层次上的组织安全约束,并且允许委托者制定委托条件.Atluri and Warner[8]针对工作流系统中的委托授权问题研究了如何在工作流系统中进行委托授权,提出了条件委托授权的概念,允许用户基于时间,工作量和任务属性等制定委托条件.但是现有的工作大多是基于单边委托对委托授权进行研究.

委托授权协商是指在委托授权建立的过程中,委托者和受托者确定执行委托授权需要满足的条件和约束的过程.根据协商机

* Supported by the Science Foundation of Shandong Province under Grant No. Y2008G28, 山东省自然科学基金;

作者简介: 高荣瑞(1985－),男,山东枣庄人,硕士研究生,主要研究领域为信息系统安全,访问控制; 孙宇清(1967－),女,博士,副教授,主要研究领域为安全策略与机制,访问控制,协同计算.

制的不同,委托授权可分为单边委托和双边委托[9].

单边委托是指由委托者单方决定委托条件的委托授权,一旦委托者决定将权限委托,受托者必须接受这一委托.在单边委托的情况下,当委托者 Alice 请求将所承担的任务委托授权给 Bob 时,Bob 必须接受 Alice 的委托. 采用单边委托, 可以使得系统能够高效的执行命令,比较适用于军事部门等要求执行力度高的组织,但是可能会导致系统中出现部分用户空闲而部分用户过度繁忙的情况.

在实际应用中,除单边委托外,还有很多情况需要双边委托.在双边委托中委托授权的双方共同决定执行委托授权需要满足的条件和约束.在上面的例子中,如果采用双边委托, Bob 可以根据自己的工作负载和能力选择接受,有条件接受或者拒绝接受该委托授权.双边委托,可以充分的利用资源,保证各个用户的工作负载均衡,提高工作效率.

为了满足对双边委托的需求,本文将对委托条件的概念进行扩展,提出意愿表达式的概念,建立基于双边意愿的委托授权协商模型.委托意愿和受托意愿由委托者和受托者分别制定,并可使用意愿匹配算法进行匹配.利用基于双边意愿的委托授权模型,我们可以在建立委托授权的过程中,实现基于双边意愿的委托授权.

本文结构组织如下:第 2 节介绍了基于双边意愿的委托授权协商模型,定义了意愿表达式并给出了例子.在第 3 节,讨论了基于素数的意愿表达式的表示方法并给出了委托授权协商算法.第 4 节讨论了委托授权协商模型的实现机制.对本文工作的总结和展望在第 5 节给出.

2 基于双边意愿的委托授权协商模型

2.1 模型描述

委托授权协商模型主要包括四个组成部分:管理节点,委托授权协调器,委托者节点,受托者节点.

系统管理员定义的高层次安全约束是系统在执行委托授权时必须满足的客观约束,即任何委托授权的执行都不得违反高层次的安全约束.

在双边委托授权过程中,委托者和受托者分别对委托授权的时间,地点,用户的身份和工作负载等因素做出约束和限制.当且仅当委托者和受托者对委托授权提出的约束和限制均被满足时,委托授权才能够顺利执行.我们将委托者和受托者对委托授权做出的约束和限制称作意愿.

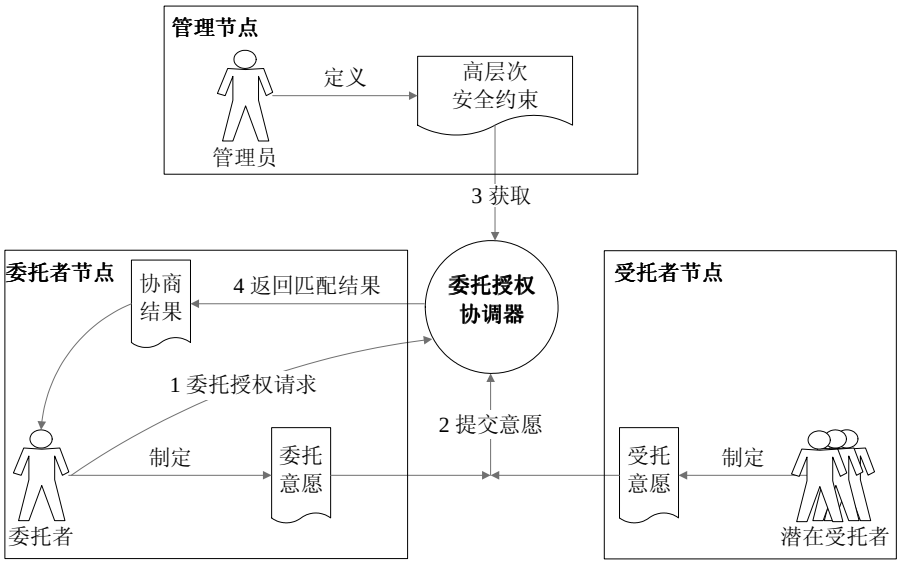

图1 基于双边意愿的委托授权协商模型

因此,委托授权的成功执行需要考虑三方面的条件:1)高层次安全约束;2)委托者的委托意愿;3)受托者的受托意愿.

委托协调器是基于双边意愿的委托授权协商模型的核心部件. 如图 1,当委托者提出委托授权请求时,委托授权协调器会检查委托授权请求是否违反系统的高层次安全约束,同时将委托者提交的委托意愿和所有潜在受托者提交的受托意愿进行匹配,将匹配成功的潜在受托者加入候选受托者集合,并作为响应返回给委托者.

2.2 意愿表达式

本文采用意愿表达式描述委托者和受托者的意愿.谓词是构成意愿表达式的基本元素,具有 Pred(t)的形式,如 Role(u)表示用户 u 所拥有的角色, 谓词表 M 存储意愿表达式中可用的所有谓词.

在实际应用中,系统管理员在系统初始化的时候定义谓词表.在意愿的制定过程中,委托者和受托者只能使用谓词表中已经确定的谓词,谓词表的大小直接影响到意愿表达式的表达能力.

原子条件表达式是构成意愿表达式的基本语义单元, 每个原子条件表达式表示委托者或者受托者对于委托授权的一个约束条件.

定义 1(原子条件表达式). 原子条件表达式是具有形式 $w_1\ op\ w_2$ 的表达式.其中,w_i, i =1,2,是一个常量,变量或者谓词,op 是一个关系运算符,$op \in \{>, <, =\}$.

定义 2(意愿表达式). 意愿表达式是建立在原子条件表达式集合 B 上的布尔表达式,其中 B={bi| bi 是原子条件表达

式,i=1,2,…n }.意愿表达式具有如下几种形式:

- **b** 其中 **b**∈**B**;
- **¬F$_i$** 其中 **F$_i$** 是一个意愿表达式,¬表示逻辑非;
- **F$_i$**∧**F$_j$** 其中 **F$_i$, F$_j$** 是意愿表达式,∧表示逻辑与
- **F$_i$**∨**F$_j$** 其中 **F$_i$, F$_j$** 是意愿表达式,∨表示逻辑或
- **(F$_i$)** 其中 **F$_i$** 是一个意愿表达式;

我们定义优先级:¬>**()**>∧>∨.

例 1(委托意愿):Role(delegatee)=RA∧SystemTime()>8:00am∧¬(Location()=office).

Role(delegatee)表示接受委托授权的用户必须拥有角色 RA. SystemTime()>8:00am 限制委托授权发生的时间晚于上午八点.¬(Location()=office)则指明委托授权发生的地点不得在办公室以内.

例 2(受托意愿):Role(delegator)=Prof∧**(**Delegated()=teach∨Delegated()=research**)**∧**(**Location()=office∨Location()=meetingroom**)**.

Role(delegator)=Prof 表明制定该受托意愿的用户只愿意接受来自拥有角色 Prof 的委托者的委托授权.Delegated()=teach∨Delegated()=research 限制了委托授权的内容只能是教学任务和研究任务,而不接受关于其他任务的委托授权,如作报告等.Location()=office∨Location()=meetingroom 则要求委托授权必须发生在办公室或者会议室,当用户身处除办公室和会议室以外的其他地方时,用户不接受委托授权.

2.3 意愿表达式标准化

根据布尔表达式的性质,每一个意愿表达式可以转化为等价的析取范式形式,即每一个意愿表达式 **F** 都可以表示为 $\bigvee_{i=1}^{t} F_i$,t≥1 的形式,其中 **F$_i$**, i=1,2,…,t,不包含∨,称作简单意愿表达式.因此用户的意愿可以表示为简单意愿表达式的集合,当且仅当集合中至少一个简单意愿表达式可满足时,用户的意愿可满足.

在本文中，意愿表达式的标准化是指将一个意愿表达式转化为等价的简单意愿表达式的集合的过程.

3 委托授权协商

刘光远等[10]提出了一种利用自然数唯一分解定理表示布尔表达式的方法,并给出了相应的布尔运算规则和基于此规则进行布尔表达式约简的算法.我们采用基于素数的方法表示意愿表达式,并通过对意愿表达式进行合取和约简提出了意愿匹配算法,对委托意愿和受托意愿进行匹配,判断委托意愿和受托意愿是否存在交集,并给出匹配结果.

3.1 基于素数表示的意愿表达式

由于任何自然数都可以被唯一分解为若干素数的乘积,因此可以用素数唯一标识一个原子条件表达式.设 $B=\{b_1,b_2,\cdots b_n\}$是委托意愿和受托意愿中所有原子条件表达式的集合,$P=\{p_1,p_2,\cdots p_n\}$是由前 n 个素数构成的集合,且对任意 $1\leqslant i<j\leqslant n$, 有 $p_i<p_j$.

由定义可知,简单意愿表达式可以表示为 $(\bigwedge_{i=1}^{k} b_i)\wedge(\bigwedge_{j=k+1}^{m}\neg b_j)$ 的形式,通过在 B 和 P 之间建立映射,可以用 p_i 表示原子条件表达式 b_i, i=1,2,…,n.定义 $R=\prod_{i=1}^{k} p_i$, $S=\prod_{j=k+1}^{m} p_j$,则简单意愿表达式 $(\bigwedge_{i=1}^{k} b_i)\wedge(\bigwedge_{j=k+1}^{m}\neg b_j)$ 表示为<R,S>.因此,通过标准化,可以将任何意愿表达式用$\{<R_1,S_1>,<R_2,S_2>,\ldots,<R_t,S_t>\}$的形式表示.

3.2 意愿匹配算法

意愿匹配算法首先将意愿使用基于素数的布尔表达式表示,然后将委托意愿和受托意愿两两合取,并对合取的结果进行约简,最后算法根据匹配的结果判断委托授权是否可以执行.算法输入委托意愿 dele_inte,受托意愿 rec_inte,委托意愿和受托意愿均采用析取范式的形式表示,算法输出为匹配结果集 Result,具体算法如图 2 所示.

算法(Algorithm Intentmatch).

输入:委托意愿 dele_inte,受托意愿 rec_inte

输出:匹配结果集 Result

/*基于素数表示委托意愿与受托意愿*/

1. 建立原子条件表达式与素数之间的映射表
2. 基于素数对委托意愿和受托意愿进行转化,得到 S_1 和 S_2

/*委托意愿与受托意愿两两合取并约简*/

```
3.   foreach(<a1,b1>∈S1)
4.    foreach(<a2,b2>∈S2)
                  //S1 中每一个简单意愿表达式依次和 S2 中所有简单意愿表达式合取
5.      if (mcd(a1,b2)>1 或 mcd (a2,b1) > 1)
          <a1,b1>和<a2,b2>匹配失败;                    //<a1,b1>和<a2,b2>不存在交集
6.      else
7.        foreach(<a,b>∈Result)
8.        Reduct(<a1*a2/mcd (a1, a2), b1*b2/mcd (b1, b2) >,<a,b>); //将合取结果与 Result 中的每一条规则进行约简
9.        将<a1*a2/mcd (a1, a2), b1*b2/mcd (b1, b2) >加入 Result;
          //<a1,b1>和<a2,b2>匹配结束
  /*匹配结果处理*/
10.   if(<1,1>∈Result)
11.     Return  TRUE;     //匹配结果集中出现永真式,委托授权请求可无条件执行
12.   if(Result=NULL)
13.     Return  FALSE;    //委托意愿和受托意愿不存在交集,委托授权不可执行
14.   else
15.     Return  Result;   //委托授权可以在 Result 中约束满足的情况下执行
```

图 2 意愿匹配算法

在算法的第一阶段,首先将委托意愿和受托意愿基于素数表示为集合 S_1 和 S_2.算法第 3 到 9 步利用布尔表达式的合取对委托意愿和受托意愿中的所有简单意愿表达式进行两两合取,将合取的结果以一个约束规则的形式加入匹配结果集 Result, Reduct 利用布尔表达式约简对匹配结果集合进行约简.算法第 10 到 15 步对匹配结果进行了处理:当匹配结果集中出现永真式时,表示委托意愿和受托意愿可以完美匹配;当匹配结果集为空时,表明委托意愿和受托意愿不存在交集;当匹配结果集不为空且不包含永真式时,表明委托授权可在匹配集中约束规则被满足的情况下被执行.

假设$|B|=n,|S_1|=|S_2|=m$, 由于每次合取最多向 Result 中插入一个约束规则,因此$|Result|\leq m^2$.意愿匹配算法中,建立原子条件表达式集合与素数集合之间的映射关系的时间复杂度为 O(n),进行合取和约简操作的时间复杂度为 $O(|S_1|*|S_2|*|Result|)=O(m^4)$,算法最后进行匹配结果处理的时间复杂度为 O(1),因此整个意愿匹配算法的时间复杂度为 $O(n+m^4)$.考虑到在实际应用中 m<<n,所以委托协调器能够在有效的时间内迅速的对委托者的委托授权请求做出响应.

3.3 实例分析

为了更好地说明意愿匹配算法,我们以例 1 和例 2 中的委托意愿和受托意愿进行匹配为例给出一个实例分析.

例 1 和例 2 中的意愿进行标准化后,可以得到委托意愿 Role(delegatee)=RA ∧ SystemTime()>8:00am ∧ ¬(Location()=office)和受托意愿 **(**Role(delegator)=Prof ∧ Delegated()=teach ∧ Location()=office**)** ∨ **(**Role(delegator)=Prof ∧ Delegated()=teach ∧ Location()=meetingroom**)** ∨ **(**Role(delegator)=Prof ∧ Delegated()=research ∧ Location()=office**)** ∨ **(**Role(delegator)=Prof∧Delegated()=research∧Location()=meetingroom**)**.

建立原子条件表达式集合 B 和素数表 P 之间的对应关系如表 1 所示,采用基于素数的表示方式,将委托意愿表示为{<6,5>},将受托意愿表示为{<385,1>,<1309,1>,<455,1>,<1547,1>}.

B	P
Role(delegatee)=RA	2
SystemTime()>8:00am	3
Location()=office	5
Role(delegator)=Prof	7
Delegated()=teach	11
Delegated() =research	13
Location()=meetingroom	17

表 1 原子条件表达式与素数对应表

将委托意愿中的元素和受托意愿中的元素一一匹配,由于 mcd(5,385)和 mcd(5,455)均大于 1,因此<6,5>与<385,1>,<455,1>均不存在交集.<6,5> 与 <1309,1> 匹配后得到 <7854,5>,<6,5> 与 <9289,1> 匹配后得到 <10374,5>.因此意愿匹配的结果集 Result={<7854,5>,<9289,5>}.

根据自然数唯一分解定理可知,7854 和 9289 可以唯一分解为 2*3*7*11*17 和 2*3*7*13*17,结合表 1 中原子条件表达式与素数之间的对应关系,我们可以得到匹配结果集中的两条规则,Role(delegatee)=RA∧SystemTime()>8:00am∧Role(delegator)=Prof∧Delegated()=teach ∧ Location()=meetingroom ∧ ¬ **(**Location()=office**)** 和 Role(delegatee)=RA ∧ SystemTime()>8:00am ∧ Role(delegator)=Prof∧Delegated()=research∧Location()=meetingroom∧¬**(**Location()=office**)**.当且仅当两条规则中至少一条为真时,委托授权可以被执行.

4 委托授权协商模型实现机制

根据系统所处环境的不同,基于双边意愿的委托授权协商模型可以采取两类不同的实现机制.

在集中式环境中,系统的访问控制由管理节点(或者是管理员)负责,委托授权协调器位于管理节点中,意愿匹配和处理委托授权请求的工作主要在管理节点中执行,因此对管理节点的计算能力和存储能力提出了较高的要求.而系统中所有的委托授权都必须经过管理节点进行审核,保证了较好的安全性.

在分布式环境中,不存在一个统一的管理节点负责系统的维护和管理,在这种情况下,可以由委托者节点或者受托者节点承担委托授权协调器的工作.当委托授权协调器位于委托者节点时,要求系统中每个节点定期与其他节点交换受托意愿信息,并保存记录有所有其他节点的受托意愿的受托意愿表.委托者节点发出委托授权请求时,根据自身存储的委托意愿表查找所有候选受托者并从中选择受托者执行委托授权.当委托授权协调器位于受托者节点时,委托者需要将委托授权请求和委托意愿进行广播,由所有节点根据自身的受托意愿执行意愿匹配算法.匹配失败的节点,忽略该请求;匹配成功的候选受托者节点,发送响应消息给委托者并由委托者从所有响应的节点中选择受托者执行委托授权.

5 总结与展望

双边委托授权能够充分的利用系统资源,保证系统中用户的工作负载均衡,提高工作效率,在很多实际应用中有着重要的作用.针对这一需求,本文基于委托授权双方的意愿,建立了基于双边意愿的委托授权协商模型,提出了意愿表达式的概念.文章给出了意愿匹配算法,通过意愿表达式的合取和约简对委托意愿和受托意愿进行匹配,从而实现了基于双边意愿的委托授权协商.基于双边意愿的委托授权协商模型可以有效支持双边委托授权的实施.我们还针对集中式环境和分布式环境的特点,分别讨论了基于双边意愿的委托授权协商模型在两种不同环境下的实现机制.在未来的工作中,我们将结合意愿表达式的匹配,进一步研究如何在候选受托者集合中根据不同的策略选取受托者执行委托授权.此外,扩充原子条件表达式的语义表达能力从而使基于双边意愿的委托授权协商模型能够支持语义匹配和模糊匹配也是我们未来工作的一个重点.

致谢 在此,向系统安全讨论班上的老师和同学们对本文工作给予的意见和建议表示感谢.

References:

[1] Zhang XZ, Oh S, Sandhu R. PBDM: A flexible delegation model in RBAC. In: Ferrari E, Ferraiolo D, eds. Proc. of the 8th ACM Symposium on Access Control Models and Technologies. New York: ACM Press, 2003.149~157.

[2] Zhang LH, Ahn GJ, Chu BT. A rule-based framework for role-based delegation. In: Sandhu RS, Jaeger T, eds. Proc. of the 6th ACM Symposium on Access Control Models and Technologies. New York: ACM Press, 2001.153~162.

[3] Tuan-Anh Nguyen, Linying Su, George Inman, David Chadwick. Flexible and Manageable Delegation of Authority in RBAC. In: Proc. of the 21st International Conference on Advanced Information Networking and Applications Workshops. Ontario: 2007. 453~458.

[4] Meriam Ben Ghorbel-Talbi, Frederic Cuppens, Nora Cuppens-Boulahia, Adel Bouhoula. Managing Delegation in Access Control Models. In: Proc. of the 15th International Conference on Advanced Computing and Communications. IEEE Computer Society Press, 2007. 744~751.

[5] Ye Chunxiao, Wu Zhongfu, Fu Yunqing, Zhong Jiang, Feng Yon. An Attribute-Based Extended Delegation Model. Journal of Computer Research and Development, 2006, 43(6):1050~1057.

[6] QihuaWang, Ninghui Li, Hong Chen. On the Security of Delegation in Access Control Systems. In: Proc. of 13th European Symposium on Research in Computer Security. Malaga: 2008. 317~332.

[7] Jacques Wainer, Akhil Kumar. A fine-grained, controllable, user-to-user delegation method in RBAC, In: Proc. of the 10th ACM Symposium on Access Control Models and Technologies. New York: ACM Press, 2005. 59~66.

[8] Atluri V, Warner J. Supporting conditional delegation in secure workflow management systems, In: Proc. of the 10th ACM Symposium on Access Control Models and Technologies. New York: ACM Press, 2005. 49~58.

[9] Barka E, Sandhu R. Framework for role-based delegation models. In: Proc. of the 16th Annual Computer Security Application Conf. IEEE Computer Society Press, 2000. 168~176.

[10] LIU Guang-yuan; YUAN Sen-miao; DONG Li-yan. Tool for reduct of Boolean functions based on number computing. Computer Engineering and Applications, 2007, 43(12):83~85.

附中文参考文献:

[5] 叶春晓,吴中福,符云清,钟将,冯永.基于属性的扩展委托模型. 计算机研究与发展, 2006,43(6):1050~1057.

[9] 刘光远, 苑森淼, 董立岩. 基于数值计算的布尔表达式约简工具. 计算机工程与应用, 2007,43(12):83~85.