第43卷 第6期 西 安 交 通 大 学 学 报 Vol. 43 №6
2009年6月 JOURNAL OF XI′AN JIAOTONG UNIVERSITY Jun. 2009

分布式环境中高效信任管理的研究

官尚元，伍卫国，董小社，梅一多
（西安交通大学电子与信息工程学院，710049，西安）

摘要：针对分布式环境中信任管理缺乏统一定义和一致性验证算法效率较低等问题，给出了信任管理的形式化定义. 信任管理为六元组，包括可数的主体集、信任类型集、信任属性集、上下文集，以及主体之间存在的信任关系和定义在信任关系上的且封闭于此关系的函数. 同时，讨论了形式化定义与描述性定义之间的关系，由此提出了高效的信任管理模型 NUMEN. 模型的一致性验证算法基于格不动点理论，其时间复杂度和空间复杂度与授权证书集的势 n 有关，均为 $O(n)$. 实验结果表明，NUMEN 以较小的开销能够获取较高的安全性，其一致性验证算法优于 SPKI/SDSI 和 KeyNote 模型，并得出了授权证书数和权限委托节点数是影响系统运行时间的关键因素的结论.

关键词：访问控制；信任管理；一致性验证；格不动点理论
中图分类号：TP393 **文献标志码**：A **文章编号**：0253-987X(2009)06-0015-05

Research on Efficient Trust Management in Distributed Environments

GUAN Shangyuan, WU Weiguo, DONG Xiaoshe, MEI Yiduo
(School of Electronics and Information Engineering, Xi′an Jiaotong University, Xi′an 710049, China)

Abstract: Many trust management (TM) systems have been proposed, but some issues still remain to be addressed, e. g. there is no consensus on the definition of TM in the literature, and algorithms for proof of compliance are inefficient. To address these problems, a formal definition of TM is proposed in this paper, which is composed of a set of countable principals, a set of trust classes, a set of trust attributes, a set of contexts, a set of trust relationships and a set of rules. The relationship between the formal TM and the descriptive TM is discussed. Based on the formal definition of TM, an efficient TM, called NUMEN, is presented, and the algorithm for PoC is based on the lattice-theoretical fix-point theorem. The time complexity and the space complexity of the algorithm are both $O(n)$ where n is the cardinality of the set of authorization credentials. Experimental results show that NUMEN can effectively protect sensitive resources at the cost of little performance of systems, and the PoC algorithm for NUMEN is more efficient than those for the existing TM systems such as SKPI/SDSI and KeyNote. It is observed that the numbers of authorization brokers and of delegation credentials are crucial factors in determining the runtime.

Keywords: access control; trust management; proof of compliance; lattice-theoretical fix-point

访问控制是安全服务的重要组成部分，通过某种方法显式地允许或限制访问能力及范围，可以限制对关键资源的访问，防止非授权用户访问或者授权用户非法访问[1]. Blaze 等人在文献[2]中提出信任管理(TM)的概念，以解决分布式系统中的权限管理问题，同时确定 PolicyMaker 模型的一致性验证(PoC)为不可判定问题，其只解决了受限的 PoC 问题，而 KeyNote 的 PoC 算法所解决的是 PolicyMaker 模型的子集问题[3]. REFEREE 模型可自动收集、验证凭证[4]，但其 PoC 过程复杂且允许的安

收稿日期：2008-11-10. 作者简介：官尚元(1979—)，男，博士生；董小社(联系人)，男，教授，博士生导师. 基金项目：国家自然科学基金资助项目(60773118)；国家高技术研究发展计划资助项目(2006AA01A109).

全策略和安全凭证程序的自主调用存在安全隐患. SPKI/SDSI 模型利用非集中式命名空间解决了分布式系统的权限管理问题[5],但其 PoC 算法需先执行路径发现算法,再执行证书化简算法. 可见,已有工作在 TM 上未达成统一的系统性认识[6].

针对 TM 缺乏统一定义以及已有 TM 系统 PoC 算法效率较低等问题,本文给出了 TM 的形式化定义,并讨论其与描述性定义之间的关系,同时基于形式化定义提出了高效的 TM 模型(NUMEN).

1 TM 的形式化定义

信任是指信任者在给定的上下文中基于自身经验和相关知识相信被信任者的某些属性或行为[2,6]. TM 是描述、建立、验证和维护主体之间信任关系的方法,以支持系统决策,提高系统性能. 信任描述研究了主体之间信任关系的描述;信任建立研究了如何在主体之间建立初始信任关系;信任验证研究了主体之间信任关系的存在性及其属性;信任维护研究了各主体之间信任关系的衍化.

定义 1 信任管理 $M_T=\langle P,K,A,C,F,R\rangle$,其中 P 为可数主体集,K 为可数信任类型集,A 为可数信任属性集,C 为可数上下文集,在此基础上定义如下关系.

(1)$t\subseteq P\times P$ 表示主体之间存在信任关系.

(2)$F\subseteq t\times K\times A\times C$ 描述主体之间的信任关系.

(3)$R:F^m\to 2^F$ 是定义在 F 上的函数且对 F 封闭,用于建立、验证和维护信任关系.

在定义 1 中,主体是指能够颁发授权证书、发出资源请求或提供资源的进程、程序、个人或组织等,信任类型可参考文献[7],上下文和信任属性通常与具体应用相关;"×"表示笛卡尔积.

公理 1 给定信任管理系统 $M_T=\langle P,K,A,C,F,R\rangle$,其中 F 满足:$(\forall f_1)(\exists f_2)((f_1\in F)\wedge(f_2\in F)\wedge((f_1)_t=(f_2)_t)\wedge((f_1)_K=(f_2)_K)\wedge((f_1)_C=(f_2)_C)\Rightarrow((f_1)_A=(f_2)_A))$.

公理 1 表明:在给定环境中,如果主体之间存在某类信任关系,则此关系惟一. 设 $f_1\in F,f_2\in F$,如果 f_1 和 f_2 满足$((f_1)_t=(f_2)_t)\wedge((f_1)_K=(f_2)_K)\wedge((f_1)_C=(f_2)_C)$,则 f_1 与 f_2 相同,记为 $f_1=f_2$. 根据公理 1,可以对 R 进行分类.

设 $r\in R$,则 $\mathrm{dom}(r)=\{f_1,f_2,\cdots,f_m\mid(\exists\gamma)((\gamma\in 2^F)\Rightarrow(r(f_1,f_2,\cdots,f_m)=\gamma))\}$,而 $\mathrm{cod}(r)=\{f_1,f_2,\cdots,f_k\mid(\exists f_{21},f_{22},\cdots,f_{2m})((\forall i)((f_i\in F)\wedge(i\in[1\cdots m]))\Rightarrow(r(f_{21},f_{22},\cdots,f_{2m})=\{f_1,f_2,\cdots,f_k\}))\}$.

定义 2 给定 $M_T=\langle P,T,A,C,F,R\rangle$,设 $r\in R$,则 R 分类如下.

(1)如果 $\mathrm{dom}(r)\supset\varnothing$ 且 $\mathrm{cod}(r)=\varnothing$,则 r 为撤销规则,所有撤销规则组成的集合为 R_W;如果 $\mathrm{dom}(r)=\varnothing$ 且 $\mathrm{cod}(r)\supset\varnothing$,则 r 为原生规则,所有原生规则组成的集合为 R_P.

(2)如果 $\varnothing\subset\mathrm{dom}(r)\subset(\mathrm{dom}(r)\cup\mathrm{cod}(r))$,则 r 为衍生规则,所有衍生规则组成的集合为 R_D. 如果 $1\leqslant|\mathrm{cod}(r)|<|\mathrm{dom}(r)|$,则 r 为聚合规则.

(3)如果 $\varnothing\subset\mathrm{dom}(r)=\mathrm{cod}(r)$,则 r 为衍化规则,所有衍化规则组成的集合为 R_E.

R_W、R_P、R_D 和 R_E 满足如下性质.

(1)$R=R_W\cup R_P\cup R_D\cup R_E$.

(2)$(\forall i)(\forall j)((i\in\{W,P,D,E\})\wedge(j\in\{W,P,D,E\})\wedge(i\neq j)\wedge(R_i\cap R_j=\varnothing))$成立.

在定义 2 中,原生规则用于建立信任关系,撤销规则和衍化规则用于维护信任关系,衍生规则用于验证信任关系. 因此,描述性定义和形式化定义可从不同角度对 TM 进行定义,但二者本质上相同.

2 NUMEN

本文的 NUMEN 框架如图 1 所示. 在 NUMEN 中,利用授权证书来描述主体之间的信任关系,资源提供者或权限委托者根据相关策略可主动向其他主体颁发授权证书,授权证书的存储和收集方式参考 RT(Role-Based Trust Management)[8]实现.

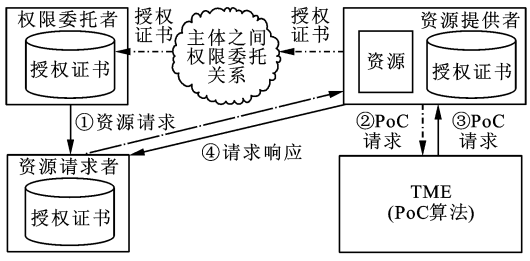

图 1 NUMEN 框架

资源请求者在发出访问资源请求之前,先根据持有的授权证书判断自己是否具有可以请求的权限,如果不具备,则取消访问请求,否则向资源提供者发送资源请求(图 1 中①)并提供所持有的授权证书. 资源提供者在收到资源请求之后,收集相关的授权证书并向信任管理引擎(TME)发出 PoC 请求(图 1 中②),TME 执行 PoC 并返回 PoC 结果(图 1 中

③)，资源提供者根据 TME 返回结果，以决定接受或拒绝请求(图 1 中④).

3　信任描述

研究信任描述即为研究定义 1 中的 F. NUMEN 利用授权证书表示信任关系，故给出权限与授权证书的相关定义.

定义 3　设有主体集 P、S-表达式集 S'、时间集 T，则权限 $r\in R=\mathrm{Pow}(P\times S'\times T)$，其中 R 为权限集. 权限 $r=\langle p,s,t\rangle$ 的语义为：主体 p 在时刻 t 可执行 s 所表示的操作.

R 是 $P\times S'\times T$ 的幂集，所以$(R,\subseteq)$是格子.

定义 4　设有主体集 P、权限集 R，则权限映射 $m\in R_{\mathrm{Map}}=P\to R$，其中 R_{Map} 为权限映射集. 权限映射 $m(p)$的语义为：主体 p 授予特定主体的权限.

因$(R,\subseteq)$是格子，故$(R_{\mathrm{Map}},\subseteq)$在点式顺序下仍然是格，即 $m\subseteq m'$成立，当且仅当$(\forall p)((p\in P)\wedge(m(p)\subseteq m'(p)))$成立，同时$(R_{\mathrm{Map}},\subseteq)$是完备格.

定义 5　设有主体集 P、整数集 D、操作集 O、时间集 T、时间区间集 $T_{\mathrm{Interval}}=T\times T$、权限映射集 R_{Map}，则授权证书 $c\in C=P\times P\times D\times O\times T_{\mathrm{Interval}}$. 授权证书 $c=\langle p_1,\ p_2,\ d,x,\langle t_1,t_2\rangle\rangle$的语义为：主体 p_1 授权主体 p_2 在 $t\in[t_1,t_2]$时执行 x 所代表的操作，如果 $d>0$，则允许 p_2 对 x 所代表的操作进一步授权. 该语义的形式化描述如下

$$S_C(\langle p_1,p_2,d,x,\langle t_1,t_2\rangle\rangle)=\langle p_1,f\rangle \text{ where } f(m)=\{\langle p,y,t\rangle\mid(y\in S_O(x))\wedge(t\in[t_1,t_2])\wedge(\text{if } p=p_2 \text{ then true else (if } d>0 \text{ then } p\in m(p_2) \text{ else false}))\}\quad(1)$$

在式(1)中，$S_O:O\to\mathrm{Pow}(S')$则表示将操作映射成 S-表达式. 对于定义 5 需说明：①授权证书单调，$f(m)$单调(在第 4 节中将利用到此性质)；②为了减小授权网络的复杂度，利用变量 d 控制授权深度，当授权证书 $d\geqslant 0$ 时，证书有效，否则无效；③在实现时，p_1 和 p_2 为对应主体的公钥或其散列值；④授权主体在发放授权证书之前，应利用私钥对授权证书进行签名，在执行 PoC 之前须验证签名.

4　信任验证

4.1　PoC 定义

信任验证研究衍生规则，即如何利用已有的事实 F 和推理规则 R 衍生出更多的事实 F，进而判断指定主体之间信任关系的存在性及属性. 如果请求者欲访问资源，则向资源提供者发出请求并提供所持有的授权证书，NUMEN 将进行信任验证，包括：①授权证书的收集(授权证书的存储和收集参考 RT[8] 的实现方法)；②授权证书的有效性检查，即验证授权证书是否过期，数字签名是否正确，授权深度 d 是否小于 0 等；③TME 根据授权证书集和请求，执行 PoC 并返回验证结果.

定义 6　TME 根据请求者 p_2 发出的请求 r_{req} 和收集到的证书集 C，确定资源提供者 p_1 是否授予 p_2 权限 r_{req}，并返回验证结果，其形式化描述如下

$$S_{\mathrm{TME}}:P\times R\times\mathrm{Pow}(C)\to B$$
$$S_{\mathrm{TME}}(p_1,r_{\mathrm{req}},C)=(r_{\mathrm{req}}\subseteq G_{\mathrm{lfp}}(p_1))$$
$$\text{where } G(m,p_1)=\mathrm{Sup}\{f(m)\mid\langle p_1,f\rangle\in C\}\quad(2)$$

在定义 6 中，$B=\{\mathrm{true},\mathrm{false}\}$. 由定义 5 知，主体 p 颁发的所有证书集为$\{\langle p_1,f\rangle\mid\langle p_1,f\rangle\in C\}$. 给定权限映射 m，则 p 授予 p_2 的权限集为$\{f(m)\mid\langle p_1,f\rangle\in C\}$，通过取上确界将 p_1 授予 p_2 的所有权限合并成一个权限 $G(m,p_1)=\mathrm{Sup}\{f(m)\mid\langle p_1,f\rangle\in C\}$. 通过求 $G(m)$的最小不动点 $G_{\mathrm{lfp}}(m)$，可求出各主体授予 p_2 的所有权限. 由定义 4 和定义 5 知，$(R_{\mathrm{Map}},\subseteq)$为完备格且 $f(m)$单调，故由格最小不动点理论知 $G_{\mathrm{lfp}}(m)$必定存在.

$G_{\mathrm{lfp}}(p_1)$表示 p_1 授予 p_2 的权限，如果 $r_{\mathrm{req}}\subseteq G_{\mathrm{lfp}}(p_1)$，则表示 p_2 的请求包含于 p_1 授予 p_2 的权限中，$S_{\mathrm{TME}}(p_1,r_{\mathrm{req}},C)$将返回 true；否则，$p_1$ 未授予 p_2 权限 r_{req}且 $S_{\mathrm{TME}}(p_1,r_{\mathrm{req}},C)$返回 false. 与定义 1 中的 R 相比，$G_{\mathrm{lfp}}(p_1)$的求解依据聚合规则，即将授权证书集 C 中的证书合并成单个授权证书 $c=\langle p_1,p_2,d,x,\langle t_1,t_2\rangle\rangle$，表示 p_1 授予 p_2 的权限为 $G_{\mathrm{lfp}}(p_1)$，其形式化描述如下

$$\langle p_1,p_2,d,x,\langle t_1,t_2\rangle\rangle=\langle p_1,f\rangle \text{ where } f(m)=\{\langle p_2,y,t\rangle\mid\langle p_2,y,t\rangle\in G_{\mathrm{lfp}}(p_1)\}\quad(3)$$

授权证书 $c=\langle p_1,p_2,d,x,\langle t_1,t_2\rangle\rangle$与定义 1 中 F 相比，$\langle p_1,p_2\rangle\in t$ 表示主体 p_1 与 p_2 之间存在信任关系，即授权关系，而 d、x、$\langle t_1,t_2\rangle\in A$ 为信任关系的属性：主体 p_1 授权主体 p_2 在 $t\in[t_1,t_2]$执行 x 所代表的操作，如果 $d>0$，则允许 p_2 对 x 所代表的操作进一步授权. 因为 NUMEN 以分布式环境中的访问控制为研究背景，所以无须考虑信任类型 K 和上下文 C.

4.2　PoC 算法

由定义 6 知，PoC 的关键问题在于求解 $G_{\mathrm{lfp}}(m)$，而求解 $G_{\mathrm{lfp}}(m)$即为聚合定义 2 中的规则. 根据完备格不动点的求解思路，本文给出 NUMEN

中的求解 $G_{1fp}(m)$以及判断请求是否合法的算法(见图2),其本质上是通过迭代求得各主体授予特定主体的权限.因为授权证书集有限,所以上述迭代过程会在有穷步内完成(见定理1).

为了证明定理1,先定义证书链.设授权证书 $c_1=\langle p_1,p_2,d_1,x_1,\langle t_1,t_2\rangle\rangle$ 和 $c_2=\langle p_3,p_4,d_2,x_2,\langle t_3,t_4\rangle\rangle$,如果 $p_2=p_3$ 或 $p_4=p_1$,则 c_1 和 c_2 构成证书链.多个授权证书构成证书链的定义与此类似.

输入:资源提供者p',资源请求者p'',授权证书集C,资源请求r_{req}.
输出:根据C,如果p'授予p''权限$r_{req}\subseteq r$,则返回true,否则返回false.

设C涉及到的授权主体集为P,C和P的势分别为n和$l(l\leqslant n)$,$p\in P$颁发的授权证书集为$C_p\subseteq C$.
步骤1:计算$G(m)$,并对P中每个主体p执行如下步骤.
(1)根据定义5计算p通过授权证书$c_i\in C_p$授予p''的权限r_i.设c_i的授权客体为p_i'',p授予p_i''的权限为r_i'.如果p_i''与p''相同,则$r_i=r_i'$;否则,$r_i=\mathrm{Inf}\{r_i',m(p_i'')\}$.
(2)p授予p''的权限为$G(m,p)=\mathrm{Sup}\{r_1,r_2,\cdots,r_k\}$,其中$k$为$C_p$的势.
步骤2:计算G_{1fp}.
(1)$m_0=\mathrm{Sup}\{\}$,即$m_0(p_0)m_0(p_1)=m_0(p_l)=\mathrm{Sup}\{\}$.
(2)将m_i代入$G(m,p)$中求得m_{i+1}.
(3)如果$m_{i+1}=m_i$,则下转步骤2(4);否则,令$m_i=m_{i+1}$,上转步骤2(2).
(4)$G_{1fp}=m_{i+1}$,$G_{1fp}(p')=m_{i+1}(p')$.
步骤3:如果$r_{req}\subseteq G_{1fp}(p')=m_{i+1}(p')$,则返回true;否则,返回false.

图2 NUMEN中的PoC算法

定理1 设授权证书集 C 的势为 n,C 涉及到的授权主体集的势为 $l(l\leqslant n)$,则 NUMEN 中的 PaC 算法的时间复杂度和空间复杂度均为 $O(n)$.

证明 设算法中步骤 1(1)、1(2)、2(1)、2(2)、2(3)、2(4)及步骤 3 语句分别需要执行 c_1、c_2、c_3、c_4、c_5、c_6 及 c_7,故算法总运行时间为 $T(n)=(c_1+c_2)l+c_3+(c_4+c_5)z+c_6+c_7$,其中 z 为步骤 2(2)、2(3)的执行次数.

如果 C 中证书均由资源提供者 p' 颁发给资源请求者 p'',则会出现最佳情况.此时,步骤 2(2)、2(3)语句需要执行 2 次,故最佳运行时间为 $T(n)=(c_1+c_2)l+c_3+2(c_4+c_5)+c_6+c_7\leqslant(c_1+c_2)\ n+c_3+2(c_4+c_5)+c_6+c_7$.

如果 C 中所有证书构成证书链且证书链长度为 n,则出现最坏情况.此时,步骤 2(2)、2(3) 语句需要执行 $n+1$ 次,运行时间为 $T(n)=(c_1+c_2)\ l+c_3+(c_4+c_5)\ (n+1)+c_6+c_7\leqslant(c_1+c_2+c_4+c_5)n+c_3+c_4+c_5+c_6+c_7$.

在最佳和最坏情况下,算法的运行时间都是 n 的一次线性函数,故其时间复杂度为 $O(n)$.

在算法中,各授权证书只需存储一次,故其空间复杂度为 $O(n)$,因此算法的时间复杂度和空间复杂度均为 $O(n)$.

5 实验及分析

本文利用 Java 语言开发了 NUMEN 原型,并在曙光 4000L 平台上对 NUMEN 原型进行了实验,测试环境配置见表 1.在各节点上安装了 NUMEN 软件,在资源请求者和资源提供者上分别安装了资源访问软件和资源提供软件,在资源请求者上安装了软件 httperf(http://www.hpl.hp.com/research/linux/httperf/),以模拟多个请求者.实验中,NUMEN 的授权证书通过预置的方式保存在各节点上.为了确保实验的准确性,每项实验单独执行 20 次并分析其平均值.

表 1 测试环境配置

平台	配置描述
曙光 4000L	22 个节点:曙光天阔 I610r-V, RedHat 9.0 (Kernel 2.4.20) 1 个内网:1 000 Mb/s 交换机

实验 1 测试 NUMEN 的性能开销.利用 httperf 软件在 1～10 个资源请求者上进行了模拟,且每个资源请求者的授权证书集相同,用测试系统配置 NUMEN 和未配置 NUMEN 时的访问资源时间总和来分析 NUMEN 的通信和处理时间开销.实验结果如图 3 所示,可见:①NUMEN 的开销基本恒定,图中的 2 条曲线的间距变化不大;②随着并发资源请求数的增加, NUMEN 的开销增大,但变化幅度并不大.从图中还可以看出,配置 NUMEN 虽然增加了开销,但对系统性能影响不大,即以较小的开销可以获取较高的安全性.

实验 2 测试 NUMEN、SPKI/SDSI 和 KeyNote 对同一授权证书集(证书格式不同,但操作与授权逻辑关系相同)的PoC时间开销.实验建立了4个不同

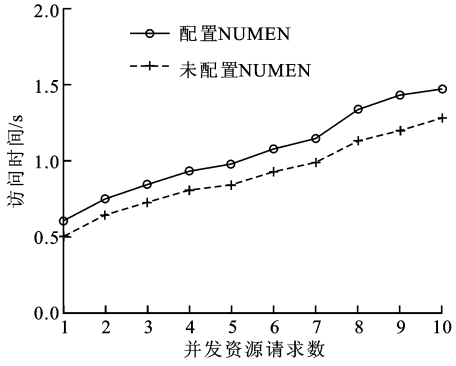

图3 NUMEN 开销测试

的授权证书集，结果见图 4. 可见：NUMEN 的时间开销小于 SPKI/SDSI 和 KeyNote，随着授权证书数的增加，时间开销差值增大. 其主要原因在于：SPKI/SDSI 工作在有序证书集上，即需先执行证书路径发现算法，再对授权证书进行排序，然后调用证书化简算法计算请求者所具有的权限[5]. KeyNote 的 PoC 算法比 SPKI/SDSI 的更为复杂[3]，而 NUMEN 能够有效地发现路径并直接计算资源提供者授予资源请求者的权限.

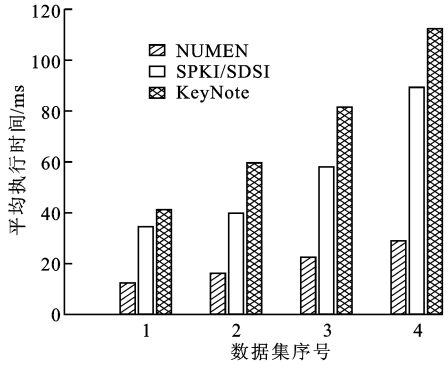

图 4　3 种模型的 PoC 平均执行时间比较

实验 3　测试了 NUMEN 总运行时间（包括证书收集时间和 PoC 的执行时间）与授权证书和权限委托节点之间的关系. 在构建授权证书时，尽可能保证授权证书覆盖所有的权限委托节点. 实验结果如图 5 所示，可见：①在授权证书数相同的情况下，总运行时间随权限委托节点数的增加而增加，主要原因在于收集授权证书的通信开销增加；②在权限委托节点数相同的情况下，总运行时间随证书数的增加而增加，主要原因在于 PoC 开销增加；③在高速局域网环境中，相比较而言，PoC 对 NUMEN 的运行时间影响更大，故研究高效的 PoC 算法对提高 NUMEN 效率至关重要.

综合实验表明，NUMEN 以较小的开销可以获

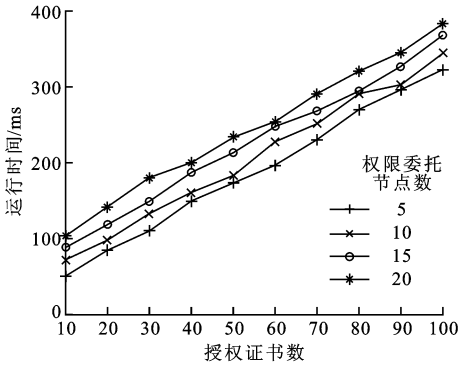

图 5　运行时间、节点数与证书数的关系

取较高的安全性，其一致性验证算法优于 SPKI/SDSI 和 KeyNote，授权证书数和权限委托节点数是影响系统运行时间的关键因素. NUMEN 运行的通信开销在于收集授权证书，而 NUMEN 中授权证书的存储和收集参考了 RT[8] 的实现方法，故其通信开销接近 RT. NUMEN 的扩展性与授权证书的存储和收集方式有关.

6　结束语

本文给出了 TM 的形式化定义及信任管理模型 NUMEN，并通过实验对 NUMEN 进行了验证. 在未来的工作中我们将研究：①NUMEN 如何颁发授权证书；②NUMEN 的证书存储和收集方式；③NUMEN 中授权证书撤销方法.

参考文献：

[1] TOLONE W, AHN G J, PAI T. Access control in collaborative systems [J]. ACM Computing Surveys, 2005, 37(1): 29-41.

[2] BLAZE M, FEIGENBAUM J, LACY J. Decentralized trust management [C]// IEEE Symposium on Security and Privacy. Los Alamitos, NJ, USA: IEEE Computer Society, 1996: 164-173.

[3] BLAZE M, IOANNIDIS J, KEROMYTIS A D. Experience with the KeyNote trust management system: applications and future directions, iTrust 2003 [M]// LNCS 2692. Berlin, Germany: Springer-Verlag, 2003: 284-300.

[4] CHU Y H, FEIGENBAUM J, MACCHIA B L, et al. REFEREE: trust management for web applications [J]. World Wide Web Journal, 1997, 2(3): 127-139.

[5] LI N, MITCHELL J C. Understanding SPKI/SDSI using first-order logic [J]. International Journal of Information Security, 2006, 5(1): 48-64.

[6] WEEKS S. Understanding trust management systems [C]// IEEE Symposium on Security and Privacy. Los Alamitos, NJ, USA: IEEE Computer Society, 2001: 94-105.

[7] JOSANG A, ISMAIL R, BOYD C. A survey of trust and reputation systems for online service provision [J]. Decision Support Systems, 2007, 43(5): 618-644.

[8] LI N, WINSBOROUGH W H, MITCHELL J C. Distributed credential chain discovery in trust management [J]. Journal of Computer Security, 2003, 11(1): 35-86.

（编辑　苗凌）