信任管理中间件研究与设计*)

尹 刚 王怀民 贾 焰 吴泉源

(国防科技大学计算机学院 长沙410073)

摘 要 在访问控制领域,代理(delegation)对于提高授权管理活动的灵活性和伸缩性具有重要意义。如何在中间件中提供代理机制一直是研究热点和难点。本文将信任管理思想引入中间件框架中,提出一种通用信任管理中间件体系结构,在此基础上基于分布计算环境StarBus设计实现了信任管理中间件Star-TMM。同其他典型中间件平台相比,Star-TMM能够有效地支持授权代理和能力代理两种重要代理机制。

关键词 信任管理,能力,权威,中间件,CORBA

Research and Design of Trust Management Middleware

YIN Gang WANG Huai-Min JIA Yan WU Quan-Yuang

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

Abstract In the field of access control, delegation is important to improve the flexibility and scalability of authorization management activities in distributed systems. How to provide delegation mechanism in middleware is a hot but difficult problem. This paper introduces the idea of trust management into the framework of middleware, proposing a generic architecture of trust management middleware. According to this architecture, we develop a trust management middleware named as Star-TMM based on StarBus, a distributed computing environment. Compared with other classic middleware, Star-TMM can provide two important delegation mechanisms, which are called delegation of authority and delegation of capability.

Keywords Trust management, Capability, Authority, Middleware, CORBA

1 引言

代理(delegation)是分布计算领域的重要概念,其基本思想是通过权力或任务在计算实体间的传递,实现灵活可伸缩的控制和管理。代理在访问控制领域主要表现为安全相关的权力或任务的传递,这对提高授权的灵活性和伸缩性具有重要意义。随着Internet和大规模Intranet的出现和广泛普及,软件系统正从面向封闭的、熟识用户群体和相对静态的形式向开放的、公共可访问的和高度动态的服务模式转变。开放协同软件环境[21]作为一种新型软件系统形态得到迅速发展,其中的授权管理问题对传统安全技术提出挑战。如何基于代理技术为开放协同软件环境提供授权管理设施是具有重要理论和实际意义的研究课题。本文提出的信任管理中间件(Trust Management Middleware)是一种面向开放协同软件环境的新型授权管理框架,结合了信任管理和中间件技术各自的优势,能够为软件系统透明地提供不同层次(管理层和请求层)的代理机制。

信任管理(TM)是1996年M. Blaze等人提出的面向Internet的新型访问控制思想和方法[4],其本质上是一种基于代理的多中心(multicentric)访问控制模型。TM系统的语言描述机制、本地化控制和通用TM引擎等特性对开放系统安全的理论和实践提供了引人注目的新思路。目前已提出的TM系统有KeyNote[5]、DL[6]、RT[7]和OASIS[8]等,大多侧重于TM语言和原型系统的研究。其中OASIS同事件中间件设施紧密集成,但是并没有从体系结构的角度深入研究TM系统同中间件的集成问题,因而难以得到广泛应用和标准化。中间件技术是当前实现分布式软件系统安全的主流技术。在中间件系统中合理地嵌入安全机制(如SSL[9]、Kerberos[10]和SESAME[11]等)可以使中间件为软件系统透明地提供消息保护、认证和访问控制等安全服务。目前典型的中间件如CORBA[14]、Java[12]和DCOM[13]等都在底层通信设施和服务框架中集成了相应的安全机制,其中CORBA还给出了完整的安全体系结构和较为详尽的接口规范[19],但是上述中间件系统普遍缺乏对代理机制的支持,如何在中间件中提供代理机制一直是研究热点和难点。CORBA和EJB先后推出的CSIv2规范(Common Secure Interoperability V2)定义了请求层的代理机制。几个重要CORBA厂商开发的安全ORB,如MICOSec[15]和ORBAsec[16]等仅实现了CSIv2中的身份代理机制。

从上述TM系统和中间件技术的特点和发展现状来看,两种技术的结合是一种必然的趋势。TM系统的语言描述机制可以为中间件的代理机制提供良好的描述手段;TM引擎模型非常适于以软件服务的形式纳入到中间件的服务框架中。中间件技术的许多特性和机制,如互操作性和跨平台特

*)本文的资助项目包括:武器装备预研基金"多数据库系统安全技术"(编号51415030203KG01),863重点课题(编号2001AA113020)和863课题(编号2003AA115410)。尹 刚 博士生,主要研究领域为数据库与分布对象技术,分布系统安全。王怀民 教授,博士生导师,主要研究方向为分布对象技术与网络安全。贾 焰 教授,博士生导师,主要研究方向为数据库与分布对象技术。吴泉源 教授,博士生导师,主要研究方向为智能软件与分布对象技术。

性、异步消息机制、事件机制等能够对 TM 系统的凭证分发、凭证链发现以及凭证撤销等核心功能提供直接有效的支持；中间件的底层安全机制(如消息加密)能够保证 TM 系统中的凭证在传递过程中的机密性和完整性。

本文首先提出一种基于代理的授权系统模型 ALM(Authorization Loop Model)，深入分析了 ALM 中 TCB 模型的特点，并在此基础上给出 TM 中间件的定义；然后提出一种较为通用的 TM 中间件体系结构，详细介绍其中的核心组件。最后介绍了我们基于自主研发的中间件平台 StarBus 设计实现的 TM 中间件 Star-TMM。

2 基于代理的授权系统模型

2.1 特权与代理

信息安全领域中有关权力的概念尚没有统一和标准化[2]，例如权限、授权、特权等词汇常见于计算机软件系统和安全系统中，往往具有多种含义且彼此经常混淆。本文用特权(privilege)统一描述系统中有关权力的概念，并根据特权在授权过程中表现出的不同权力内涵，将其划分为权限(permission)、权威(authority)和能力(capability)，分别用以表达不同的安全概念和策略。

定义1 权限(Permission) 是访问系统中客体(如文件、对象、服务等)的特权。

在开放协同环境中，权限的传播可能通过多种方式。权威和能力是两种可以传递的特权，因此适合表达权限的传播。下面基于权限给出两者的定义，我们也试图从两者同权限的关系揭示其内在联系和区别。

定义2 权威(Authority) 是管理权限的特权。管理权限是指建立实体到权限间的映射关系(直接授权)。在开放协同环境中，权威往往表现为管理其他实体的权限的特权。权威可以在实体间传递，例如实 A 允许 B 管理 A 定义的某些权限(如阅读 A 中文档的权力)，B 就具有管理 A 的这些权限的权威。在某种限制下 B 还可能将这种权威传递给实体 C，由 C 间接地管理 A 的权限。显然，A 缺省具有管理自身定义的权限的特权，并且是这些权限的权威源(source of authority)。

定义3(能力，Capability) 是使用权限的特权。在开放协同环境中，能力往往表现为对其他实体的权限的使用。能力也可以在实体间传递，例如实体 A 允许 B 以 A 的身份访问服务器，B 就具有使用 A(被服务器授予)的权限的能力。在某种限制下 B 还可能将这种能力传递给实体 C。特别地，如果服务器授予 A 权限 p，那么 A 就缺省使用 p 的能力。

代理从本质上讲就是权威或能力在计算实体间的传递。本文分别称之为权威代理和能力代理。

定义4(权威代理，Delegation of Authority) 是指一个实体把权威传递给另一实体，由后者代表前者管理权限的行为。

定义5(能力代理，Delegation of Capability) 是指一个实体把能力传递给另一实体，由后者代表前者访问资源的行为。现有 TM 系统大都仅支持简单的权威代理策略，目前只有 RT[9]在一定程度上同时支持权威代理和能力代理。

2.2 授权回路模型(ALM)

基于上述概念，我们提出面向开放协同系统的授权系统模型 ALM(Authorization Loop Model)，如图1。ALM 的出发点是权限管理和请求访问是有机整体(如下所述，两个层次的代理形成一个回路)。ALM 的基本思想是：(1)基于权威代理实现灵活可伸缩的授权管理；(2)基于能力代理实现请求的调用链中计算实体间灵活可控的身份和权限的委托。

在 ALM 中，服务实体(SE)拥有需要保护的客体(objects)，对客体的访问权限的授权由安全管理员(SA)实施。SA 将部分或全部权威代理给管理中介实体(MAE)，MAE 将对用户实体(UE)直接授权。被授予某种权限的 UE 常常通过请求中介实体(RAE)访问 SE 的资源，RAE 最终把代理来的能力提交给 SE 的本地服务进程(由 SE 本地基准监视器 RM 进行访问控制)。从本质上讲，RAE 到 SE 的能力传递也是一种能力代理。这样，特权在整个授权活动中从“权威”→“权限”→“能力”的变化轨迹，形成了从 SE 出发又终止于 SE 的回路，我们称之为授权回路(authorization loop)。ALM 模型没有限制管理层和请求层的代理深度，因此 MAE(或 RAE)可以进行多级代理。

ALM 在基准监视器模型[1]的基础上扩展了可信计算基(Trust Computing Base)的内涵，TCB 是确保系统安全运行的基本组件。TCB 必须确保其运行过程的正确性、不可篡改和旁路，并且规模尽可能小以易于完整性分析(本文将这些要求简称为 TCB 特性)。ALM 将系统 TCB 分为 LTCB 和 GTCB 两部分。其中 LTCB 同基准监视器模型中的 TCB 一致，主要保证访问控制过程不被旁路，是系统本地的安全核心；GTCB 则必须确保授权回路中各种代理行为能够安全可靠地实施，由各中介实体合作实现。因此仅确保本地计算环境的可信性是不够的，各自治软件系统之间协作过程的 TCB 特性也必须得到保证，这是设计面向开放协同计算环境的授权系统的重点和难点，也是我们采用中间件结构设计授权系统的重要依据。

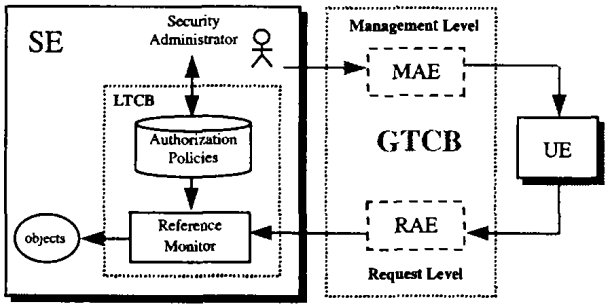

图1 基于代理的授权系统模型 ALM

ALM 的 GTCB 是一种分布式 TCB 结构[19]。在中间件内部实现分布式 TCB 可以大大增强系统的安全性和可靠性。从本质上讲，中间件是在操作系统之上且独立于应用逻辑的软件部分，因此 ALM 的 LTCB 和 GTCB 都可以纳入到中间件一级实现。理想情况下，软件系统仅通过制定和修改授权策略即可基于中间件实现授权管理的目标。

定义6(信任管理中间件，Trust Management Middleware) 是一组同现有通信中间件集成的可重用软件设施，能够透明地为软件系统提供基本安全机制(消息保护、认证、访问控制)和代理机制(包括管理层代理和请求层代理)，并基于公钥密码技术保证自身的 TCB 特性。

ALM 的另一重要内容是能够描述上述代理策略的授权语言。我们设计了一种基于角色的可扩展授权语言 REAL(Role-based Extensible Authorization Language)。REAL 可以看成是带约束的 RT[7]，是一种凭证语言(凭证＝策略＋策

略颁发者对策略的签名)。目前我们正在完善 REAL 的语法及其满足性验证(Proof Of Compliance)算法。本文着重讨论体系结构问题,授权策略均封装在凭证(Credentials)中由 TM 中间件进行管理和操作。ALM 包括三类凭证:直接授权凭证、权威代理凭证和能力代理凭证。

3 通用 TM 中间件体系结构

基于上述信任管理中间件的思想,我们在自行设计开发的中间件平台[20]上构筑了面向开放协同软件环境的可信计算平台。平台的中间件体系结构包括5个主要组件:传输层通信协议、Session Controller、Credential Manager、Reference Monitor 和服务适配器,如图2。该体系结构的通用性表现在这些组件同样适用于其他主流分布计算环境(如 Web Services,OGSA Services[3])。

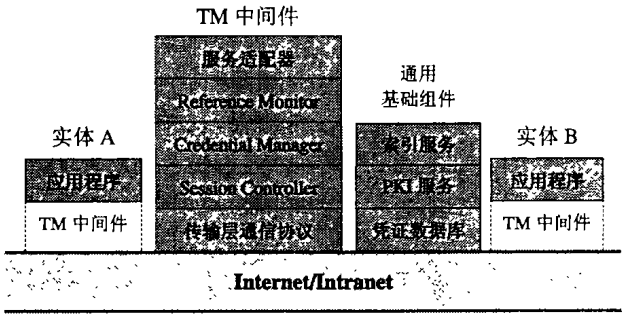

图2 通用可信计算平台

平台中的实体(具有明确身份的人或计算实体)通过应用程序和 TM 中间件同其他实体交互。在上述平台中,实体间的交互关系可以分为4类:(1)实体 *A* 访问实体 *B* 的资源(资源访问);(2)实体 *A* 委托实体 *B* 代表 *A* 访问其他实体的资源(能力代理);(3)实体 *A* 为实体 *B* 授予某种权限(直接授权);(4)实体 *A* 委托实体 *B* 管理自己的权限(权威代理)。实体间的所有通信都基于传输层通信协议(如 IIOP,HTTP,SOAP 等)实现。TM 中间件可以将安全机制(如 Kerberos,SSL 等)集成到通信协议中,为整个平台提供消息保护和基本的身份认证。其中 CORBA 安全规范给出的 SSLIOP 协议插件就是 IIOP 协议同 SSL 协议的有机结合。

实体将自己定义的授权凭证存放在凭证数据库中,并通过 Credential Manager(CM)发布给其他实体的 CM,接收方经过验证将基于收到的凭证更新本地凭证数据库。CM 之间传递的凭证包括两类:直接授权凭证和权威代理凭证。当一个实体对请求进行策略满足性验证时,CM 还可以根据一定的算法在 CM 组成的网络中搜索相关凭证。其中通用基础组件中的索引服务能够辅助 CM 对其它相关 CM 的搜索与发现。CM 是管理层 GTCB 的核心组件。

当实体发送资源访问请求时,Session Controller(SC)根据本地策略将有关凭证(可以是上述三类凭证)封装到请求消息中。请求方可以通过 SC 在请求中声明本次请求需要激活的能力(能力申请)。当目标实体收到请求后,目标方 SC 从请求消息(如 IIOP 消息)中解析出凭证并进行基本的验证(如签名验证)。目标方 SC 将调用 CM 和 Reference Monitor 验证请求方能力申请的合法性。"能力申请"机制能够促使 CM 以目标驱动的方式搜索凭证(CM 只能搜索直接授权凭证和权威代理凭证)。SC 的另一重要作用是管理有状态安全会话,缓存经过验证的凭证(甚至满足性验证结果),这对提高系统的效率具有重要意义。SC 是请求层 GTCB 的核心组件。

Reference Monitor(RM)是 TM 引擎,对 CM 和 SC 提供的二元组(凭证集合,请求)进行策略满足性验证。经过 RM 验证的合法请求将派发给服务适配器,以激活相应的服务对象对请求进行响应。R 是 LTCB 的核心组件。

通用可信平台是一种可以灵活配置的开放式计算环境。通用基础组件(索引服务、PKI 服务和凭证数据库)可以有多种实现形式。实体可以单独具有自己的 PKI,也可以同其他实体形成联盟共享一个 PKI。索引服务和凭证数据库都可以根据实际的需求以共享或自治的形式部署。

4 Star-TMM 的设计与实现

Star-TMM 是基于中间件平台 StarBus[20]的 TM 中间件实例。本节就 Star-TMM 中的凭证模型和几种关键技术介绍有关设计与实现细节。

4.1 凭证模型

凭证(Credentials)是 Star-TMM 的核心数据结构。在上述 TM 中间件模型中,Session Controller(SC)和 Credential Manager(CM)的主要功能都以凭证为操作对象。

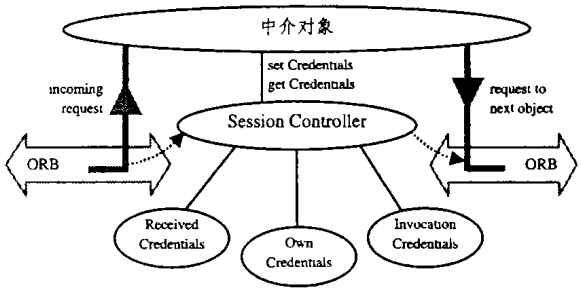

图3 SC 中的凭证传递模型

目前的主流计算环境都属于分布对象系统。多级对象调用[19]是软件实体间的常见现象。能力代理是解决对象间身份和权限委托的有效方法。SC 将分布对象(如图3中的中介对象)处理的凭证划分为 ReceivedCredentials、Own Credentials 和 Invocation Credentials,分别用来表示请求方"push"过来的凭证,中介对象自身的凭证以及中介对象进一步发送请求时提交给下一级对象的凭证。这种凭证分类方法便于实施各种能力代理策略,也能为应用程序提供清晰的调用接口。

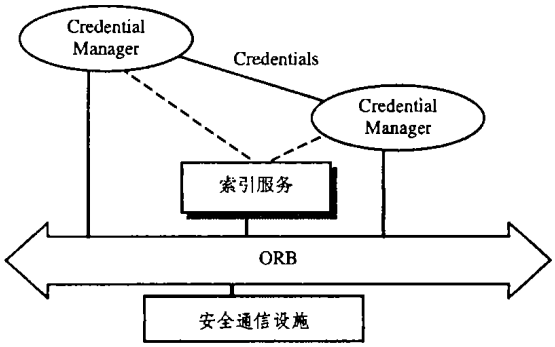

图4 CM 中的凭证发现模型

CM 则关注凭证的发布和搜索。为此 Star-TMM 基于名字服务(NamingService)设计了索引服务。各 CM 将自己需要发布的凭证信息和自身引用(如 IOR)发布到索引服务器上(如图4),其他 CM 可以基于凭证的主题(subject)或颁发者(issuer)快速定位有关 CM。凭证发现模型可以作为一种通用的凭证获取机制推广到其他凭证系统中(如 SPKI[17]、PGP[18]

等系统)。

4.2 关键技术

本节概要介绍 Star-TMM 中采用的关键技术,这些技术的实现细节将在后续的文章中介绍。

截获器技术:截获器(Interceptor)是一种重要的软件成长性设计方法,CORBA 开发商需要扩展 ORB 功能时,可以在 ORB 内核的基础上以 ORB 服务的形式对其进行扩展,截获器技术可以使 ORB 服务便携地"挂接"到 ORB 内核上。Session Controller 在结构上同 ORB 紧耦合,我们将其设计为一种 ORB 服务,由相应的截获器调用。

主动安全技术:在开放协同软件环境中,实体之间的授权关系具有较强的动态性。凭证的有效性需要不断刷新,这对系统造成很大开销。为此我们基于 StarBus 中的异步回调机制[20]实现一种凭证的主动撤销制:当一个 CM 从其它 CM(目标 CM)获取凭证时,随即发送一个异步请求以"订阅"该凭证的撤销事件。当目标 CM 撤销该凭证的时候,就会检查是否存在相应的"订阅"请求。如果存在,则对所有请求产生相应,以达到主动撤销的目的。

凭证缓存技术:授权决策的响应时间对系统的效率会产生重要影响。现有 TM 系统的响应时间完全取决于凭证发现算法和策略满足性验证(Proof Of Compliance)所耗费的时间。虽然对 POC 算法本身的直接优化或采用更好的计算模型是提高效率的直接方法,但 POC 算法复杂性往往随着 TM 语言描述能力的增强迅速增加。凭证发现算法因为涉及网络延迟,更加难以控制。因此从系统结构的角度出发,尽可能减少对 POC 算法的调用次数是一种有效的方法。Star-TMM 的 Session Manager 采用 Cache 技术,将输入凭证、本地策略状态和相应 POC 结果作为记录写入"凭证 Cache"中,如果后续请求提交的凭证集同"凭证 Cache"中缓存的记录具有"一致性",则直接采用该记录中的 POC 结果作为对本次请求的授权决策。

结论 本文提出的通用 TM 中间件体系结构将信任管理思想引入到中间件框架中。同其他中间件平台的安全体系结构相比,该体系结构能够支持授权代理和能力代理两种重要的代理机制。这对提高授权管理的活性和伸缩性具有重要意义,对解决企业联盟(enterprise coalitions)系统,多中心协作系统以及 Grid 计算平台的授权管理问题具有较强的理论和实用价值。

Star-TMM 能够支持 CORBA 安全规范中定义的代理模型,并在此基础上实现了管理层代理和多级代理机制。为解决跨安全域的授权问题给出新的解决方法,是对 CORBA 安全体系结构的有意义的补充。另外,Star-TMM 在开发过程中提出的主动安全技术和凭证缓存技术也具有一定的普遍意义。

参 考 文 献

1 Lampson B. Protection. ACM Oper. Syst. Rev, 1974, 8(1): 18～24
2 Firozabadi B S, Sergot M. Power and Permission in Security Systems. In: Christianson B, Crispo B, Roe M, eds. Security Protocols, number 1796 in Lecture Notes of Computer Science, Cambridge, UK, Springer Verlag, 1999. 48～53
3 Foster I, Kesselman C, Nick J. Tuecke S. The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration. 4th Global Grid Forum, Toronto, Canada, 2002
4 Blaze M, Feigenbaum J, Lacy J. Decentralized trust management. In: Proc. of 17th Symposium on Security and Privacy, Oakland, IEEE, 1996. 164～173
5 Blaze M, Feigenbaum J, Keromytis A D. Keynote: Trust management for public-key infrastructures. In: Cambridge 1998 Security Protocols Intl. Workshop, Cambrige, Springer-Verlag, 1999. 59～63
6 Li Ninghui, Grosof B N, Feigenbaum J. Delegation logic: A logic-based approach to distributed authorization. ACM Transaction on Information and System Security (TISSEC), Feb. 2003
7 Li Ninghui, Mitchell J C, Winsborough W H. Design of a role-based trust management framework. In: Proc. of the 2002 IEEE Symposium on Security and Privacy. IEEE Computer Society Press, May 2002, 114～130
8 Yao W, Moody K, Bacon J. A Model of OASIS Role-Based Access Control and its Support for Active Security. In: Proc. Sixth ACM Symposium on Access Control Models and Technologies. SACMAT 2001, Chantilly, VA, USA, May 2001. 171～181
9 Transport Layer Security Working Group. The SSL Protocol. Version 3.0, March 1996
10 IETF, "RFC 1510, The Kerberos Network Authentication Service, V5," Internet Engineering Task Force, 1993
11 Kaijser P. A Review of the SESAME Development. Lecture Notes in Computer Science, 1998, 1438: 1～8
12 Lai C, Gong L, Koved L, Nadalin A, Schemers R. User Authentication And Authorization In The Java Platform. In: Proc. of Annual Computer Security Applications Conf. Phoenix, Arizona, USA, 1999. 285～290
13 Microsoft. "DCOM Architecture." Microsoft, 1998
14 The Common Object Request Broker: Architecture and Specification, Version 3.0. July 2002
15 http://www.objectsecurity.com/csiv2.html
16 http://www.adiron.com/InterOpTest.html
17 Clarke D, Elien J-E, Ellison C, Fredette M, Morcos A, Rivest R L. Certificate Chain Discovery in SPKI/SDSI. 1999
18 Zimmerman. Pgp user's guide: [Technical report]. MIT Press, 1994
19 Object Management Group. Security Service Specification, Version 1.8. March 2002
20 张志伟,吴泉源,王怀民,贾焰. 支持时间无关激活的分布对象中间件异步模型. 计算机学报, 2004
21 徐锋. 开放协同软件环境中信任管理研究:[博士论文]. 2003

(上接第41页)

期算法(RED)参数配置产生的分叉和混沌现象,从新的角度得出具有非线性丢包函数的 RED 算法在相同参数配置下,比线性的 RED 算法具有更好的鲁棒性。本文通过实验仿真比较,分别给出了自适应的 RED 算法和比例-微分 RED 算法控制分叉与混沌的效果。

参 考 文 献

1 Floyd S, Jacobson V. Random Early Detection Gateways for congestion avoidance. IEEE/ACM Transaction on Networkin, 1993, 1(4)
2 Hollot C V, Misra V, Towsley D, Gong W B. Analysis and design of controllers for AQM routers supporting TCP flows. In: Proc. of IEEE Infocom 2001. Volume: 3, 1510～1519
3 Mathis M, Semke J, Mahdavi J, Ott T. The Macroscopic Behavior of the TCP Congestion Avoidance Algorism. Computer Communications Review, 1997, 27
4 Johari R, Tan D K H. End-to-End Congestion Control for the Internet: Delays and Stability. IEEE/ACM Transactions on Networking, 2001, 9(6)
5 Veres A, Boda M. The Chaotic Nature of TCP Congestion Control. In: Proc. of IEEE INFOCOM 2000. Volume 3, 1715～1723
6 Ranjan P, Abed E H. Nonlinear Instabilities in TCP-RED. In: Proc. of IEEE Infocom 2002. Volume 1, 249～258
7 Erich P. On the Non-linearity of the RED Drop Function. In: Proc. of Intl. Conf. on Computer Communication, 2002
8 Sun J S, Ko K T, Chen G R, et, al. PD-RED: to Improve the Performance of RED. IEEE Communications Letters, 2003, 7 (8): 406～408
9 Firoiu V, Borden M. A Study of Active Queue Management for Congestion Control. In: Proc. of IEEE INFOCOM 2000
10 Wang X F. Controlling bifurcation and chaos in Internet congestion control system. In: Proc. of Intelligent Control and Automation, 2002, 1: 573～576