

可信可控网络中的 QoS 资源控制模型*

曲延盛[†], 李伟, 罗军舟, 王鹏

(东南大学 计算机科学与工程学院, 江苏 南京 210096)

Resource Control Model for QoS in the Trustworthy and Controllable Network

QU Yan-Sheng[†], LI Wei, LUO Jun-Zhou, WANG Peng

(School of Computer Science and Engineering, Southeast University, Nanjing 210096, China)

+ Corresponding author: E-mail: yansqu@seu.edu.cn

Qu YS, Li W, Luo JZ, Wang P. Resource control model for QoS in the trustworthy and controllable network. *Journal of Software*, 2011, 22(11): 2782-2794. <http://www.jos.org.cn/1000-9825/3914.htm>

Abstract: In this article, a Resource Control Model for QoS (RCMQ) is presented, based on the former work on Trustworthy and Controllable Network Architecture (TCNA). The QoS mechanisms are distributed into four logical planes in RCMQ: QoS decision plane, QoS discovery plane, QoS data plane, and QoS interface plane. Deployment of RCMQ includes independent and centralized resource control within an Autonomous System and consensus distributed control among ASes. RCMQ is scalable because it decouples all network level QoS control mechanisms from the data plane, and it is robust because RCMQ is a simple and effective close loop control structure. Finally RCMQ in SSFNet environment are implemented, and experimental results show that RCMQ can provide a more stable QoS transmission and maintain fewer flow states than InterServ, which from the other point of view proves that TCNA is more controllable.

Key words: quality of service (QoS); trustworthy and controllable network architecture; resource control model

摘要: 基于前期工作——可信可控网络体系结构 TCNA(trustworthy and controllable network architecture),提出了一种 QoS 资源控制模型(resource control model for QoS,简称 RCMQ)。该模型从网络可控角度将 QoS 控制分为 4 层: QoS 决策层面、QoS 观测层面、QoS 接口层面和 QoS 资源层面,模型部署包括独立集中的域内控制和一致性分布式控制。RCMQ 模型闭环控制结构保证了 QoS 控制的有效性,而独立集中的 QoS 决策层面使得模型具有可扩展性。最后,通过仿真实验验证了 RCMQ 资源控制模型与 InterServ 模型相比,能够提供更为稳定的 QoS 传输,并且极大地减少了 QoS 状态维护,同时也从侧面说明可信可控网络体系比传统 TCP/IP 网络具有更高的可控性。

关键词: 服务质量;可信可控网络体系结构;资源控制模型

中图法分类号: TP393 文献标识码: A

随着网络技术的飞速发展,多媒体和分布式数据分发应用逐渐增多,扩展“尽力而为”的服务模式使网络能

* 基金项目: 国家自然科学基金(60903161,60903162,61003257,61070161,61070158); 国家重点基础研究发展计划(973)(2010CB328104); 高等学校博士学科点专项科研基金(200802860031); 江苏省自然科学基金(BK2008030); 江苏省网络与信息安全重点实验室资助项目(BM2003201); 计算机网络和信息集成教育部重点实验室资助项目(93K-9)

收稿时间: 2009-11-19; 修改时间: 2010-04-27; 定稿时间: 2010-07-06

够提供端到端的服务质量保证,是网络发展迫切需要解决的问题.端到端的服务质量保证也被看作是下一代互连网络所具备的最重要的特征之一.

为网络提供服务质量保证是通过设计 QoS 资源控制模型来实现的.QoS 模型不仅包含多种复杂的 QoS 控制机制,还需要保证各种机制有效协作.文献[1]总结了现有 TCP/IP 网络体系下 QoS 模型,认为这些模型可以在逻辑上划分为以下 3 层:QoS 控制层面、QoS 管理层面和 QoS 数据层面,如图 1 所示.每个层面对应各自的网络功能,并包含一类 QoS 资源控制机制.路由机制和接纳控制机制是属于 QoS 控制层面,这些控制机制一般是嵌入到一些分布式协议当中;测量机制(metering)、服务水平协商机制(service level agreement)^[2]和服务恢复机制(service restoration)等属于 QoS 管理层面,QoS 管理层面通过配置管理来协调 QoS 控制层面和 QoS 数据层面;QoS 数据层面主要包含与数据转发调度相关的资源控制机制,如缓冲区管理机制、调度与排队机制、报文的整形与分类机制等^[1].这种 3 层控制模型反映了 TCP/IP 网络的基本资源控制结构,所以现有网络的 QoS 资源控制模型都可以看作是这种模型的一种具体应用.在综合服务(integrated services,简称 IntServ)模型中,QoS 控制层面、QoS 管理层面和 QoS 数据层面都被集中实现在交换设备上.IntServ 虽然能够带来比较灵活的服务质量保证,但它加重了交换设备的负担,因而模型可扩展性非常差,不容易在网络上实现.区分服务(differentiated services,简称 DiffServ)模型省略了 QoS 控制层面或者使用了隐式的 QoS 控制层面,并在数据层面采用了具有可扩展性的区分服务传输方式,使得这种方案容易在现有网络中部署实施.但由于缺乏有效接纳控制,区分服务没有摆脱传统网络“尽力而为”的服务模式,因而无法保证服务质量.

网络服务提供商们认为,现有的主干网络带宽容量是充足的,没有必要建立专门的机制来保证服务质量^[3].文献[4]通过对历史总结指出,主干网与接入网之间容量关系存在一定的周期性,并预言,在不久的将来,随着应用的升级以及接入带宽容量的快速增长,相对充足的主干网络带宽将不复存在.而文献[5]进一步指出,即使主干网络带宽充足,TCP/IP 控制层面的分布式特性也会使路由抖动剧烈,网络无法提供稳定有效的传输.

现有网络无法提供服务质量保证的主要原因是由于资源控制结构不合理^[6,7].在现有网络控制结构下,无论控制层面还是数据层面或者管理层面的控制机制都需要由路由器来处理,这使得路由器成为网络控制的瓶颈.实现服务质量保证需要在这 3 个层面添加复杂的控制机制,这些控制机制最终仍然需要部署在路由器上.而现有路由器无法胜任如此复杂的任务,这是现有服务质量保证模型无法大规模部署实施的直接原因;另外,由于网络控制分布式的特性以及管理层面和控制层面分离的控制结构,管理层面和控制层面的控制机制不得不各自采集网络状态信息并建立相应的视图以供决策.实现服务质量保证却需要各个层面控制机制密切合作,控制机制的合作必然要求他们交互各自的视图.但在 Internet 级别分布式网络中,视图交互是极为消耗资源并且容易出错的,视图之间交互一旦出现不一致,都可能导致网络抖动,进而影响服务质量,这是现有服务质量保证模型无法有效实施的根本原因.例如,现有网络控制层面路由机制分为域内路由机制(interior gateway protocol,简称 IGP)和域间路由机制(border gateway protocol,简称 BGP),域内路由机制只采集域内设备的关联信息以建立域内拓扑,同样,域间路由机制只收集域间可达信息.端到端的路由却需要 IGP 与 BGP 合作,而现有实现二者视图交互的机制,无论是全互联的通告方式还是路由反射器都被证明是极其耗费资源并且容易出错的^[8].

服务质量保证在 TCP/IP 网络上的部署失败,使研究人员转向通过改变网络控制结构来实现服务质量保证.这些模型包括基于覆盖网络(overlay network)的 QoS 控制模型^[9]、基于 BB(bandwidth broker)QoS 资源控制模型^[10]等.这些模型只关注了问题的一部分,比如通过从数据层面剥离部分控制层面功能来减轻路由器负担,但本质上这些模型的控制结构仍然基于传统 3 层控制模型,因而不能提供有效的服务质量保证.

从资源管理和控制角度,我们的研究小组提出了一种可信可控网络体系结构 TCNA(trustworthy and

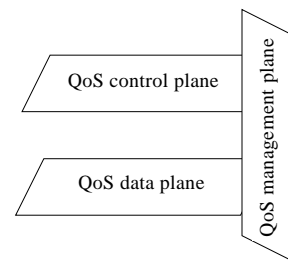


Fig.1 Three-Plane QoS resource control model of current network

图 1 现有网络的 3 层 QoS 资源控制模型

controllable network architecture)^[11],将网络控制分为决策层面、观测层面、可信可控接口层面和资源层面.基于可信可控网络,本文提出了一种 QoS 资源控制模型 RCMQ(resource control model for QoS).不同于传统的 3 层资源控制模型,RCMQ 是一种 4 层的资源控制模型,包括 QoS 决策层面、QoS 观测层面、QoS 接口层面和 QoS 资源层面.这种 4 层模型体现一种简单的闭环控制,通过集成传统的控制层面和管理层面,保证了决策过程的一致性;通过将管理控制层面的控制机制从传统的数据层面剥离,保证了资源层面的可扩展性;同时,QoS 观测层面和 QoS 控制接口能够使控制层面获得全网络的逻辑视图并实现直接控制,避免了传统控制机制之间的视图交互,保障网络传输服务的稳定性.

本文第 1 节简要介绍可信可控的网络体系结构.第 2 节提出 RCMQ 模型架构和模型的部署方式.第 3 节阐述 RCMQ 模型各个层的控制机制.第 4 节通过实验和仿真验证模型的性能.第 5 节总结全文,并提出后期的研究工作.

1 可信可控的网络体系

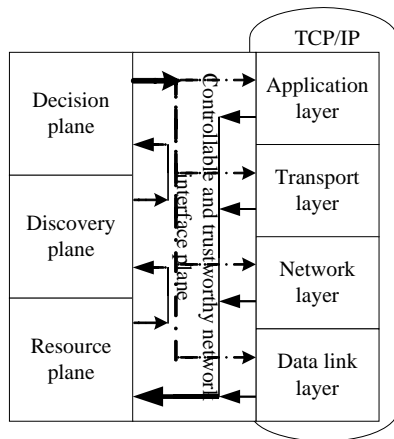


Fig.2 Trustworthy and controllable network architecture

图 2 可信可控网络体系

在我们前期工作中,基于 4D 网络结构^[12]提出了可信可控的网络体系结构.可信可控网络的主要目的是通过对网络进行有效的控制来实现用户对网络的信任.如图 2 所示,可信可控网络体系从控制角度分为 4 个部分,即决策层面、观测层面、资源层面和可信可控接口层面.其中,决策层面集中了传统网络的控制层面和管理层面的控制机制;观测层面检测资源层面的状态,为决策层面提供资源视图;资源层面从控制角度等价于传统网络的数据层面,只包含每包的控制;而可信可控接口层面负责维护各个层面的通信.

与传统网络相比,可信可控网络具有以下优势:(1) 可信可控网络能够实现网络级别的控制目标.由于可信可控网络具有集中的决策层面,这个决策层面集成了所有网络级别的控制机制,因而可以消除各个控制机制之间的决策冲突,并且加强各个控制机制之间的合作.(2) 可信可控网络能够容纳各种异构网络体系.可信可控网络并不涉及网络传输的细节,只是针对网络控制结构,因而可信可控网络能够在各种网络环境中实现.(3) 可信可控网络具有可扩展性.可信可控网络将复杂的网络控制,如路由控制等,从路由器等交换设备上

剥离并集中到决策层面,网络可以添加一些复杂的 QoS 控制机制并且不会对路由器造成负担,方便网络发展.

2 RCMQ 模型及部署

2.1 RCMQ 模型

RCMQ 模型基于可信可控网络的控制结构,从服务质量保证入手对 QoS 资源控制机制进行了 4 层划分,如图 3 所示,分别包括 QoS 决策层面、QoS 观测层面、QoS 接口层面和 QoS 资源层面.各层功能表述如下:

- QoS 决策层. QoS 决策层面是整个 RCMQ 模型的决策者,QoS 决策层面的功能可以简要概括为受理来自用户的 QoS 请求,根据 QoS 观测层面的网络视图确定是否接纳该请求,在肯定的情况下通过 QoS 接口层面对 QoS 资源层面做相应的资源预留,并处理传输过程中网络的异常.
- QoS 观测层. QoS 观测层面主要功能是为 QoS 决策层面提供实时、正确、有效的网络资源视图.QoS 观测层面包含拓扑发现、网络资源状态监测和状态维护机制.
- QoS 接口层. QoS 接口层面功能是为 QoS 决策层面和 QoS 资源层面提供可靠的通信会话,保证 QoS 决策层面的控制命令有效地传递到 QoS 资源层面.

- QoS 资源层. QoS 资源层面包含所有的网络设备,如路由器交换机、链路等.从控制功能角度,QoS 资源层面等同于传统的数据层面,QoS 资源层是服务质量保证的最终实施者.

与传统网络的 3 层构架相比,RCMQ 的 4 层结构反映了一种逻辑上简单有效的闭环控制过程.如图 3 所示,QoS 决策层面通过 QoS 观测层面提供的资源状态信息进行决策,通过 QoS 接口层面控制到达 QoS 资源层面实施决策.这种简单而有效的控制结构能够维护网络的稳定,并且有利于网络的扩展.

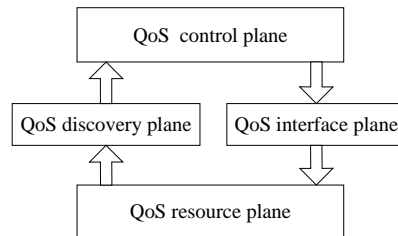


Fig.3 Resource control model of RCMQ

图 3 RCMQ 的资源控制模型

2.2 模型部署

现有互联网是受技术因素、经济因素以及社会因素多方面影响而形成的多个自治系统构成的巨型网络.为适应这种网络架构,RCMQ 模型的部署方式包括:

- 自治系统内独立集中的资源控制

独立的资源控制是指传统 QoS 控制层面和管理层面的资源控制机制从路由器等交换设备上剥离;集中的资源控制是指在自治系统内实现传统 QoS 控制层面和管理层面逻辑上的集中.现有网络自治系统内部网络拓扑较为简单,并且设备数量有限,同时,自治系统作为网络中独立的管理区域,对域内资源有绝对的管理权限,因而容易在自治系统内部实现独立集中的资源控制.若要实现这一目标,需要在自治系统内基于 RCMQ 模型建立完整的 4 层控制,即 QoS 决策层面、QoS 观测层面、QoS 接口层面和 QoS 资源层面,并且 QoS 决策层面需要在物理上独立于 QoS 资源层面.

- 自治系统之间有效的分布式控制

出于经济、安全和社会因素考虑,自治系统不会向其他网络交出域内资源的控制权,因而不可能实现整个网络的集中式管理.所以,自治系统之间的分布式管理仍然是必需的.造成传统域间控制抖动剧烈、收敛速度慢的根本原因,是由于自治系统内域间控制机制没有一个完整的域间可达视图,并且控制机制对网络状态变迁强调响应性而忽略一致性.在各个自治系统之间实现一致性的分布式 QoS 控制的关键是,在自治系统内建立一个完整统一的域间资源视图,并且改变域间资源控制机制的响应.

RCMQ 模型的部署方式如图 4 所示.本文将自治系统内的 QoS 决策层面和 QoS 观测层面的机制部署在控制节点中,这个控制节点从自治系统内部获得拓扑信息,并且从其他自治系统获得相关资源的状态和路由信息共同构成全网络视图,QoS 决策层面的控制机制根据这个视图做出决策并实施.这种域内集中控制域间分布控制的部署模式一方面提高了 QoS 网络控制的有效性:RCMQ 模型可以看成是 InterServ 模型和 DiffServ 模型的结合.从决策层面来看,由于控制的集中和网络的抽象,AS 之间实施基于每流的资源预留;而从资源层面来看,每个交换节点仍然使用具有扩展性的区分服务.另一方面,这种部署方式降低了网络控制的复杂性:由于域内的集中控制,桩自治系统(stub-AS)和一些不传递过渡流量的自治系统可以看作是高级的用户(用户的集合),而传输自治系统(transit-AS)可以看作是高级的路由交换设备.这样,整个网络由高级的“用户”和高级的“交换设备”构成,而“交换设备”在数量上只有 2 万多个^[13],远远小于网络中路由器设备的数量.

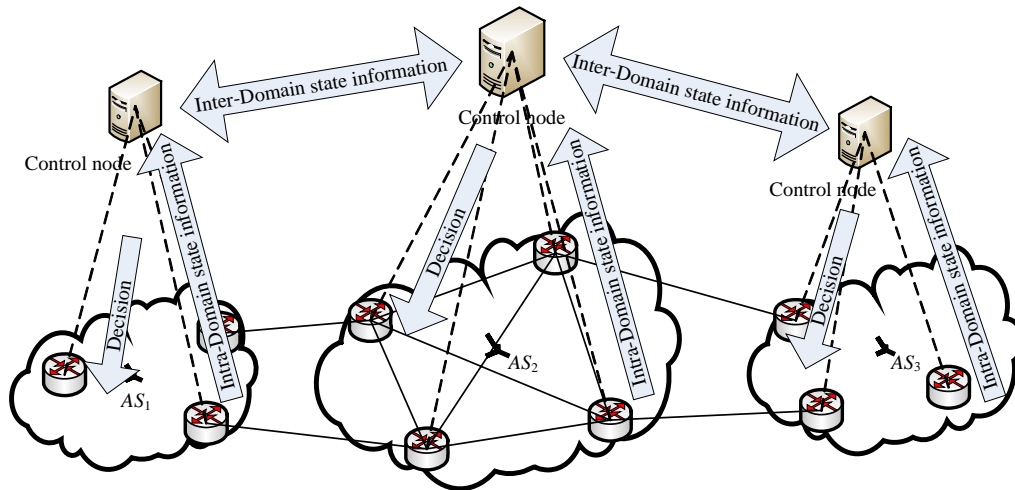


Fig.4 Deployment of RCMQ

图 4 RCMQ 部署方式

3 RCMQ 的控制机制

由于 RCMQ 模型修改了传统网络的资源控制结构,因而本文需要重新设计或者修改一些控制协议,使其能够符合域内独立集中和域间分布的资源控制部署方式,以便更有效地完成 QoS 资源控制。

QoS 资源层面负责从 QoS 决策层面接收网络控制决策,而 QoS 资源层面对于数据报的处理仍然沿用传统网络数据层面的各种机制,如数据报的封装格式以及缓冲管理、数据报排队和调度等.为了实现服务质量保证并保持 QoS 资源层面的扩展性,QoS 资源层面采用区分服务的方式,即通过数据报的分类调度实现有限数量的 QoS 服务模式来匹配用户的服务.QoS 资源层面路由器需要接收并执行来自 QoS 决策层面的决策,如路由表,并根据数据报的标记类型执行不同的数据报调度服务;而边界路由器不仅从 QoS 决策层面获得路由表等决策,而且需要接收和维护从 QoS 决策层面发送的 QoS 流与服务类型对应表,以便在入口处对数据报进行标记;同时,边界路由器如果长时间没有收到某个 QoS 传输报文,则向决策层面申请删除资源预留。

QoS 接口层面功能是保证域内控制命令和域间控制信息及时、准确、有效地传递,而这本身就是服务质量保证的目标.在本文中,其功能是通过在域内各个层面之间的通信链路上预留部分带宽来实现的。

3.1 QoS观测层面的网络视图

QoS 观测层面的功能是为 QoS 决策层面提供决策依据,而这个功能是通过构建和维护网络视图 NV (network view)来实现的.网络视图包含网络拓扑视图 TV(topology view)、域内网络资源的状态视图 SV(state view)和路由视图 RV(route view)。

网络拓扑反映网络的连接状态,它是 QoS 决策层面决策的主要依据.路由机制根据网络拓扑信息选择合适的链路和路由器为用户传输服务,观测层面可以通过侦听域内路由器的之间的 LSA(link state advertisement)信息来获得域内的拓扑,通过收集边界路由器接收的域间路由通告采集域间拓扑;同时,路由机制将选择的路由生成路由视图 RV(route view).这样,接纳控制可以确定哪些网络资源为用户 QoS 申请服务.域内资源状态信息体现域内资源的服务能力,接纳控制根据资源状态判断是否接纳用户 QoS 申请.域内的资源状态信息主要指域内路由的服务能力,它以链路为单位,包含时延、抖动、可用带宽等.由于底层资源层面采用的是基于区分服务的数据报调度传输方式,观测层面并不需要实时测量这些信息,而只需要知道对于特定资源层面数据报调度机制其传输的时延、抖动、丢包的范围和允许传输的最大带宽.这种网络视图一方面能够为 QoS 决策层面的控制机制提供完整、及时的网络状态信息,保证决策的有效性;另一方面能够使得 QoS 决策层面控制机制对网络状

态变化响应趋于一致,并避免了复杂的视图的交互,从而实现控制机制有效协作。

域间的可用路由仍然采用传统 BGP 的存储方式,以它们需要经过的域内边界路由器分类存储在 Loc-RIB-In 中;域内拓扑仍然存储在链路状态数据库中.路由机制根据域内拓扑集中选择路由,并在观测层面构造域内路由视图.域内的路由视图以可达树的形式存储,即以域内每个边界路由器为根,其他边界路由器或者端系统是这棵树的叶节点,根节点到叶节点之间路径就是路由机制选择的路由;同时,接纳控制机制在这个视图上标记资源占用情况以及资源状态.

3.2 QoS决策层面的路由机制

保证服务质量最为关键的是保证传输稳定,本文在引言部分已经分析了现有 QoS 资源控制模型路由机制存在的问题.本文中,路由机制的工作模式基于 RCP^[14],它根据 QoS 观测层面提供的网络视图集中选择路由,并将路由表分发到域内的被控交换设备上.但是,RCP 对于域间网络状态变化过于强调响应性^[15],会造成传输抖动等网络动荡,因而不利于服务质量保证.RCMQ 模型中,路由机制修改了 RCP 的路由响应方式,提出一种网络域间路由更新算法(NRUA),这种更新方式会抑制网络动荡、缩短网络收敛时间,从而为服务质量保证提供稳定的传输.为了说明方便,本文采用使用以下定义:

定义. 某条到目的地址 D 可用路由上自治系统 A 的上游自治系统,是指一旦这条可用路由被采用,这些自治系统的数据报就需要流经自治系统 A ;反之,则称为自治系统 A 的下游自治系统.

本文将造成网络域间路由重定向的原因分为两种.一种是网络拓扑变化是由故障等一些不可预测的客观因素造成的,在这个过程中,路由通告信息被称为紧急状态信息(emergent message).这些消息不仅包含路由变化信息,而且包含故障位置.另一种则是由于自治系统出于主观的管理要求(如流量工程需要)重新选择或者取消某些路由.由于第 2 种情况造成网络路由重定向的时间是可以预测的,在这个过程中,路由通告信息被称为安全状态信息(stable message).不同于传统的 BGP 响应方式,本文中路由控制对这两种原因导致的路由重定向的响应方式是不同的,进而提出了一种域间路由更新算法(NRUA),如图 5 所示.

Network-Wide route update algorithm (NRUA)

```

1. WHEN Receive message  $M$ 
2.   IF  $M \in$  emergent message
3.     Update (TV);
4.      $M \leftarrow$  SelectRoute (TV);
5.     ChangeRouteTable ();
6.     Update (RV);
7.     ACK ( $M$ );
8.     Send ( $M, uAS$ );
9.   ELSE
10.    Update (TV);
11.     $M \leftarrow$  SelectRoute (TV);
12.    Send ( $M, uAS$ );
13.    IF all  $uAS$  acked  $M$  or TimeOut
14.      ChangeRouteTable ();
15.      Update (RV);
16.      ACK ( $M$ );
17.    ENDIF
18.  ENDIF
19. ENDWHEN

```

Fig.5 Inter-Domain route update algorithm

图 5 域间路由更新算法

在 NRUA 算法中,若被通告的路由存在环路,本文也将其归为紧急状态消息.对于紧急状态消息,路由更新方式采用 BGP-RCN^[16]的更新方式:自治系统一旦感知或者被通知网络故障,则首先将路由通告信息存储在 Loc-RIB-In 中,然后,路由决策时过滤通过故障设备的路由,并且将新决策传递给上游节点.对于故障等客观原因

引起的网络路由重定向,文献[16]已经证明,在通告内添加故障位置信息,可以使网络在故障等客观因素引起网络重定向的收敛速度达到 $2D$,其中, D 是所有网络的直径时延,远远高于传统 BGP 协议对于故障的响应速度.对于安全状态信息的响应,本文将路由选择和执行分开.一旦自治系统需要主观重定向路由或者收到其他自治系统路由更新消息,它首先根据路由消息更新网络拓扑视图 TV,并根据 TV 进行路由选择(SelectRoute),然后将决策结果以路由更新消息发送给该自治系统的上游自治系统(uAS).直到所有的上游自治系统都回复该路由更新消息,它才修改路由表(ChangeRouteTable).在对安全状态信息的响应造成的路由重定向过程中,由于下游自治系统先通告路由变化,在上游自治系统决策后才执行路由更改,可以实现域间路由的无损更新(即不影响数据传输),并且本文可以证明,这个过程中 NRUA 算法不会造成域间路由环路.

定理. 处于收敛状态的网络,即域间路由无环路,如果由主观原因引起网络域间路由重定向并且期间没有网络故障发生,NRUA 算法在路由重定向过程中不会导致网络域间路由环路.

证明:由 NRUA 算法可知,由主观原因造成域间路由重定向过程中,任意自治系统执行路由策略时,所有它的上游自治系统都执行了路由策略.假设 AS_k 执行路由策略后数据流传输出现环路,并且环路上存在以下自治系统: $AS_m, AS_{m+1}, AS_{m+2}, \dots, AS_{m+n}, AS_m$.环路上自治系统和 AS_k 之间存在如下 3 种关系:

- (i) $AS_{m+j}(-1 < j < n+1, j \in \mathbf{Z})$ 都是 AS_k 上游自治系统;
- (ii) $AS_k = AS_{m+j}(-1 < j < n, j \in \mathbf{Z})$;
- (iii) $AS_{m+j}(-1 < j < n+1, j \in \mathbf{Z})$ 都是 AS_k 下游自治系统;

对于关系(i),任意 $AS_{m+j}(0 < j < n)$ 是 AS_m 的上游自治系统,由 NRUA 算法推出, $AS_{m+j}(0 < j < n)$ 在 AS_m 执行路由策略前执行其路由策略;同时, $AS_{m+j}(0 < j < n)$ 又是 AS_m 的下游自治系统,又可以推出 $AS_{m+j}(0 < j < n)$ 在 AS_m 执行路由策略后执行其路由策略,因而存在矛盾.所以, AS_k 上游自治系统之间数据流传输不存在环路.

对于关系(ii),若 $AS_k = AS_{m+j}(0 < j < n)$,同理可以推出 AS_k 既要在 AS_m 执行路由策略之前执行自己的路由策略又要在 AS_m 执行路由策略之后执行自治的路由策略,因而也存在矛盾;否则 $AS_k = AS_m$,可以推出 AS_k 既要在 $AS_{m+j}(0 < j < n)$ 执行路由策略之前执行自己的路由策略又要在 $AS_{m+j}(0 < j < n)$ 执行路由策略之后执行自治的路由策略,同样也存在矛盾.所以,路由环路中不可能包含 AS_k .

对于关系(iii),可以推出自治系统 AS_k 的下游自治系统之间存在路由环路.由 NRUA 算法可知,自治系统 AS_k 执行路由策略时,所有下游自治系统仍然保持原有路由传输数据,下游自治系统存在域间路由环路可以推出路由更新之前网络未收敛.这与已知条件矛盾,因而 AS_k 的下游自治系统之间不存在路由环路.

综上所述,由主观原因引起域间路由重定向过程中,NRUA 算法不会造成域间路由环路.证毕. \square

3.3 QoS决策层面的接纳控制

决策层面的接纳控制首先确定网络中的资源是否满足用户需要,同时,接纳控制还负责处理 QoS 传输过程中出现的网络的故障或者异常.传统的 3 层 QoS 资源控制模型中,由于接纳控制与路由机制在域内都是分布式执行的,因而它们之间难以协作,接纳控制只能被动地定时向网络发送更新消息以防止路由更新带来的资源预留失效,这使得网络状态变化过程中难以保证服务质量并增加网络负载.在 RCMQ 模型下,基于全网络视图,本文采用主动的方式响应路由重定向.自治系统一旦执行路由重定向,传输自治系统便直接通知相邻的 QoS 端系统要求其重新发送 PATH 消息.这样可以更好地保证服务质量,并大大降低 QoS 控制造成的网络负载.

完成接纳控制,网络需要向端系统通告其服务能力,即 QoS 路由上资源的服务能力,本文中用 QoS 路由约束表示.文献[17]认为,QoS 约束本质上可以分为两大类:最大最小性约束和可加性约束.对于最大最小性约束,如带宽需求,各个自治系统不需要了解全局信息,只需要根据自己域内的服务能力就可以判断是否能够满足该约束;而可加性约束则需要全局信息,必然需要查询路由上各个控制域的服务能力综合才能判断.

本文中,接纳控制机制包含 3 种基本的控制消息:PATH 消息、RESV 消息和 TEARDOWN 消息.PATH 消息的功能是描述本次申请的 QoS 要求,并收集网络设备的服务能力信息.PATH 消息包含 TSPEC 参数、QSPEC 参数和 PSPEC 参数,其中,TSPEC 参数描述了流的属性,如流协议类型、发送与接收端地址和端口等;QSPEC 参数功能描述本次 QoS 申请所要求的流量和期望的性能,如需要的带宽、容忍的时延抖动与丢包率等;PSPEC 参数

与 QSPEC 参数类型是一致的,其功能是沿着 PATH 路径收集网络资源的服务能力.RESV 消息用于确认网络是否接纳该申请.TEARDOWN 消息用于提示自治系统拆除资源预留。

具有 QoS 要求数据流的发送者和接收者是 QoS 传输的两个端系统,它们一起确定该 QoS 传输的各项路由性能指标,并且一次 QoS 申请是单向的,只为从发送者到接收者的数据流服务.发送者发起申请,而接收者确定网络是否具备足够的资源满足该申请.QoS 申请详细的处理过程如下:

发送端处理流程:

- (1) 与接收端协商 QoS 传输的具体参数要求.
- (2) 将 QoS 传输要求添加到 PATH 信息中发送.
- (3) 若收到 RESV 消息,则开始 QoS 传输,传输过程中,保证在一段时间内有该 QoS 传输的报文(可以是内容为空的报文)在网络中传递.若收到 TEARDOWN 消息,则终止通信.若收到路由重定向消息,则决定是否继续通信(是否重发 PATH 消息).
- (4) 当传输结束后或者发送 PATH 消息一段时间内没有收到 RESV 消息时,发送 TEARDOWN 消息,拆除自治系统建立的资源预留.

接收端处理流程:

- (1) 与发送端协商 QoS 传输的具体参数要求.
- (2) 如果接收到 PATH 消息,检查 PATH 消息中的 PSPEC 各项参数是否满足该 QoS 申请要求(QSPEC).如果满足,向该路由发送 RESV 消息;否则,发送 TEARDOWN 消息.

传输 AS 处理流程:

- (1) 接收到 PATH 消息,决策层面从消息中得到 QoS 申请的 QSPEC 参数匹配服务域内服务类型,并检查域内资源是否满足最大最小约束:如果满足,则根据域内路由性能参数对该 QoS 传输进行资源预留并更新 PATH 消息中的 PSPEC 参数;同时,控制节点保存该流传输的状态信息(TSPEC 中的协议类型、地址和端口号)以及上一跳的自治系统的控制节点地址和服务类型,并向该 QoS 传输的边界路由器发送流信息和服务类型对应表.上述工作完毕后,将 PATH 转发给它路由上下一跳自治系统的控制节点;否则,直接向发送端发送 TEARDOWN 消息.
- (2) 若接收到 RESV 消息,则确认资源预留,将 RESV 消息发送给该申请上一跳自治系统.
- (3) 当收到 TEARDOWN 消息时,或者在 QoS 传输过程中一定时间内没有收到该流的报文,删除资源预留.

在上述处理过程中,PSPEC 参数的更新方式如下:对于带宽等最大最小性参数初始赋值为无穷大,如果现有自治系统路由中相应资源剩余量小于该参数,则更新;对于时延、丢包率等可加性参数初始赋值为 0,自治系统将域内路由由相关参数与原有值进行相加(或者取对数相加)后对应更新.

QoS 传输过程中若发生路由重定向,端系统的处理流程不变,传输自治系统的处理步骤如下:

传输 AS 处理流程(异常):

- (1) 传输自治系统如果收到路由变更信息,并且路由变化原因是由网络故障等客观原因造成的,则自治系统迅速通告使用这条路由 QoS 应用的端系统,由它们决定是否取消 QoS 传输.
- (2) 否则,路由变化是由主观原因造成的,自治系统的处理方式如下:
 - a) 自治系统一旦收到路由变化信息并且要执行决策路由,它首先检查与它直接相邻的端系统中哪些 QoS 传输在本次路由变换中受到影响,并立即通知它们重新发送 PATH 消息.
 - b) 自治系统收到 PATH 消息并且接纳控制已经接纳了该 QoS 传输,如果该自治系统有影响该 QoS 传输的路由决策还没有执行并且执行路由策略后该 QoS 传输下一跳 AS 改变,自治系统接纳控制先保留该 PATH 消息后转发该消息,直到路由决策执行后再转发;否则,直接转发该 PATH 消息.
 - c) 自治系统控制节点收到来自下游自治系统的 TEARDOWN 消息(RESV 消息),如果决策层面存在该流状态信息并且现有的 QoS 传输需要经过这个下游自治系统,则拆除资源预留并向前转发(直接转发);否则,忽略该消息.

图 6 显示了网络非故障原因引起网络路由重定向时,传输自治系统的详细处理过程,图中显示了由 sender 到 receiver 的一个 QoS 链接(如图中粗线所示).传输过程中,AS₃ 需要调整策略,不再向 AS₂ 提供到 receiver 的路由服务.路由消息从 AS₃ 开始,经 AS₂ 传递到 AS₁.AS₁ 收到控制消息后查找网络视图,发现经过 AS₅,AS₇,AS₄ 可以到达 receiver,它向 Sender 端发送消息,要求其重新发送 PATH 消息.由于 AS₁ 没有关于该路由的上游自治系统,所以它直接回复路由控制消息给 AS₂,提示其执行路由决策(发送 PATH 消息和回复路由通告消息是同步的).AS₂ 接收到来自 AS₁ 的回复后回复 AS₃,AS₃ 收到来自 AS₂ 的回复后执行自己的控制策略.

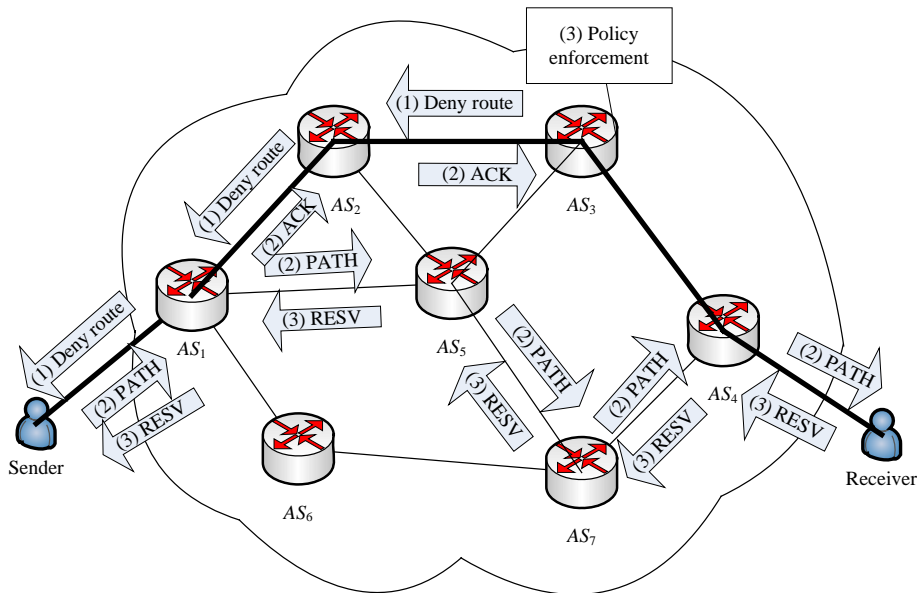


Fig.6 Demonstration of admission control

图 6 接纳控制处理演示

4 仿真实验

4.1 实验部署

本文使用 SSFNet 验证 RCMQ 模型的性能.在实验中,本文采用了类似美国 Tier-1 网络的拓扑结构(该拓扑配置已经在 SSFNet 中实现).如图 7 所示,该拓扑中包含 25 个自治系统,拓扑中资源(链路、交换设备和端系统)的属性(带宽、可靠性等)采用原有脚本的配置,自治系统之间链接关系包含了服务提供关系(customer-provider)和对称关系(peer-to-peer)^[18].在这个网络中,每个子网对应一个自治的控制域.而在上层控制域中,低层子网只抽象为一个节点.按照前述的 RCMQ 模型及控制机制设计,本文在自治内部实现了 RCMQ 的 4 层模型.如上文所述,QoS 决策层面和 QoS 观测层面统一实现在自治系统内部的控制节点服务器中.控制节点的各个控制机制如图 8 所示,其中,QoS 决策层面包含路由机制和接纳控制机制,而 QoS 观测层面包含全网络视图的维护机制.QoS 资源层面仍然采用了传统 TCP/IP 网络的数据传递模式和报文格式.

InterServ 模型是较为完整的 QoS 模型.它基于传统网络的路由协议并设计了完整的资源预留机制.为了便于比较,在另外相同的网络拓扑中,本文仍然采用传统的 TCP/IP 网络 3 层控制体系,并且每个网络节点支持 RSVP 协议,域间路由采用 BGP,域内路由统一采用 OSPF,从而实现 InterServ 资源控制.

RCMQ 模型的 QoS 资源层面的资源控制机制仍然是基于传统资源控制模型的 QoS 数据层面,因而本文没有设计新的数据报调度处理机制.前期很多研究工作集中在 QoS 数据层面的报文分类调度和队列管理的算法,提出了一些实用的算法并验证了其性能.同时,为了简化实验部署,在 RCMQ 模型和 InterServ 模型只模拟实现了

质量保证型服务,因为质量保证型服务是一种“硬实时”服务,利于决策层面对其运行性能的估计。

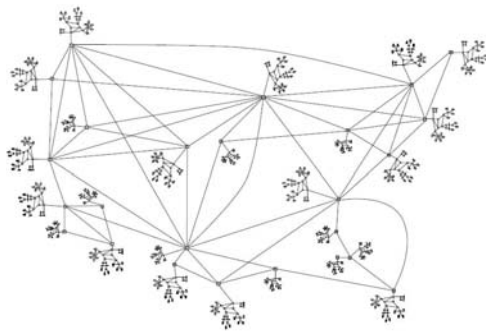


Fig. 7 Experiment topology

图 7 实验拓扑结构

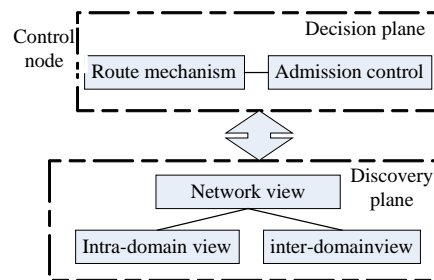


Fig. 8 Relationship of control mechanisms in control node

图 8 控制节点中各个控制机制之间关系

在实验中,本文路由机制和接纳控制机制的相关参数设置如下:InterServ 模型中,BGP 路由机制的 MRAI 设置为 5s,MASOI 设置为 1s;RSVP 协议中,更新信息发送时间间隔设置为 2s,状态信息保存时间设置为 10s,即 10s 内没有收到更新则删除预留.相应的,在 RCMQ 模型的接纳控制机制中,如果 QoS 传输不连续,而且需要自治系统维护其申请,那发送端系统必须保证 2s 内有数据报在网络中传输(可以是空报文),同样,控制节点状态信息保存时间设置为 10s;RCMQ 模型中,路由机制的 MRAI 设置为 5s,接纳控制协议设置与 RSVP 协议一致.两个模型拓扑的每个端自治域 QoS 申请速率平均是 2 次/秒,并且相邻 QoS 申请的时间间隔服从指数分布.其目的端自治系统是随机的,每个 QoS 申请要求带宽都是一样的 250Kbps;端到端时延要求随机生成,范围在 150ms~250ms 之间.端到端抖动为 1ms,分组丢失率上限为 10^{-2} ,持续时间为 20s.

4.2 实验结果

本文实验主要验证 RCMQ 模型的有效性和扩展性.RCMQ 模型有效性主要体现在:(1) 网络能够提供稳定的路由;(2) 网络保证为 QoS 传输提供足够的资源.这两个条件在网络状态恒定的条件下是容易满足的,有效性实验主要验证在网络状态变化过程中,两种模型是否能够保证有效的 QoS 决策.由于 RCMQ 模型将网络决策控制从路由器等交换设备剥离,路由器负担大大减轻,不再需要处理 QoS 申请,它们甚至不需要计算路由,因而在资源层面,RCMQ 比 InterServ 模型更加具有扩展性.模型的扩展性还体现在:对于一定数量的 QoS 传输,QoS 决策层面需要处理的 QoS 消息和维护的 QoS 流状态信息的数量多少.

由上文分析可知,造成网络状态变化的原因有两种,本文首先测试主观原因更改路由策略过程对 QoS 传输的影响.实验中,传输自治系统出于流量工程或者管理的原因,可以取消全部或者部分目的地址的路由服务.本文通过实验验证这种流量工程配置对于 QoS 传输的影响.在每次实验中,本文选择一定数量的 AS 及其服务商,并执行这种路由变化策略.每次实验中,本文增加实施路由策略变化的自治系统数量.图 9 和图 10 分别显示了随着实验的进行,两个模型 QoS 应用出现传输异常比例和传输自治系统中,QoS 资源预留量与实际使用量之间的关系.

图 9 的实验结果显示,InterServ 模型中,随着参与主观路由策略变化的自治系统的增加,越来越多的 QoS 应用出现数据报文大量丢失或者被拒绝服务.实验中,16%的自治系统参与路由更新会造成测试传输过程中超过 30%的 QoS 应用出现丢包、异常或者拒绝服务.这主要归因于路由机制对网络状态改变的响应方式.下游自治系统已经更改路由信息,而上游系统在没有收到路由消息之前无法做出及时回应而造成丢包;并且,现有网络 3 层控制结构也会造成 QoS 传输异常.由于各个控制机制自己形成的网络视图信息不完整,容易出现路由环路和路由黑洞^[19-21].在 RCMQ 模型中,由于 QoS 决策层面拥有完整及时的网络视图,并且改变了决策层面的控制机制对网络状态变化的响应,因而只有很少的 QoS 应用出现数据报丢失.而发生丢包的原因主要是由于域内路由器的路由由表更新不同步造成的,传输过程中会通知一些 QoS 应用取消传输,被取消的原因是现有网络没有路

由为其服务,因而不能提供用户所要求的服务质量,并且这种通知方式是友好的.

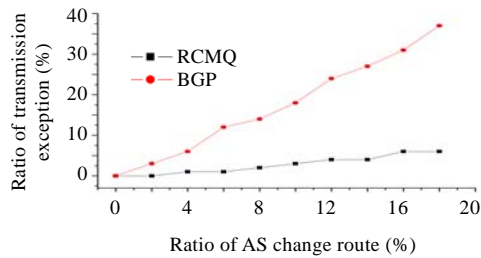


Fig.9 Ratio of exception of QoS application in transition state of network

图9 网络状态变化过程中 QoS 应用传输异常比率

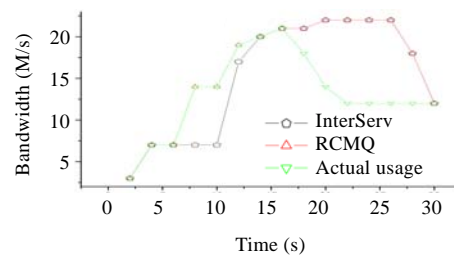


Fig.10 Resource usage and reservation in transit AS

图10 传输自治系统资源使用量与预留量的关系

图 10 显示在网络状态变化过程中,Tier-1 网络的一个自治系统某个链路两个模型资源预留量与网络实际 QoS 传输量之间的关系.在路由策略的变化过程中,InterServ 模型资源预留量和 QoS 应用实际使用量出现了严重的背离.在第 6s,由于其他自治系统的决策层面变更路由,导致大量使用其他路由的 QoS 传输借道这个自治系统,而这个自治系统的路由器并没有及时更新预留量,直到新的 PATH 消息被处理后才与实际使用量保持一致.而在这个过程中,QoS 传输的实际使用量远远大于预留量是很难保证传输服务质量的.在第 16s,由于 QoS 申请数量突然增加,网络 QoS 决策层面取消了部分路由,InterServ 模型和 RCMQ 模型资源预留量没有及时更新,直到自治系统长时间没有收到更新报文而删除资源预留.这个过程由于预留量大于实际使用量,因而只是暂时地影响了网络吞吐量,并不影响网络 QoS 传输.

接下来,在实验进行过程中不断有链路或者路由器故障产生.RCMQ 模型中,自治系统控制收敛的标志是 QoS 决策层面路由机制选择了正确的路由;而相应的 InterServ 模型中,统计对象换成了路由器,即路由器收敛标志是路由器选择了正确的路由.图 11 显示对于某个自治系统内一个边界路由器失效,两个模型的收敛速度.由结果可以看出,RCMQ 模型对于错误或者异常的响应速度远远高于传统的 InterServ 模型.对于同样的网络故障,RCMQ 模型只要不到 1s 所有的自治系统都达到了收敛的状态,而 InterServ 模型需要近 7s 才能收敛.这主要是由于传统网络对于网络异常相应采用了类似穷举的测试方法,往往需要尝试所有可能的情况才会知晓网络到底发生了什么异常.RCMQ 的高响应速度主要得益于可信可控网络具有完整的全网络视图,从而很大程度上抑制了的网络病态抖动,为服务质量保证提供了一个相对稳定的数据传输层面.

图 12 显示整个实验运行期间,两个模型控制节点平均每秒处理的 QoS 消息维护的 QoS 传输状态数量.在 RCMQ 模型中,QoS 消息主要包括 PATH 消息、RESV 消息和 TEARDOWN 消息;而 InterServ 模型除了包含 RCMQ 模型的 3 种消息外,还包含一些 Error 消息.实验结果显示,对于同样数量的 QoS 申请,RCMQ 模型处理的 QoS 消息数目远远小于 InterServ 模型.两种模型自治系统平均需要维护 51 条 QoS 传输状态,InterServ 模型自治系统平均每秒需要处理 26 条 PATH 消息和 RESV 消息以及 13 条其他 QoS 消息,而 RCMQ 模型平均每秒只需处理 2 条 PATH 消息和 RESV 消息以及 1 条其他 QoS 消息(统计时进行了四舍五入).这主要得益于 RCMQ 模型主动响应网络状态变化,因而在 QoS 传输期间不需要发送 PATH 消息维护 QoS 申请状态.但是由于 RCMQ 模型的接纳控制仍然是基于每流的,因而和 InterServ 模型一样,网络维护的 QoS 应用的状态仍然比较多.

综合以上实验,可信可控网络中的 RCMQ 模型与传统网络体系下的 InterServ 资源控制模型相比,能够减少网络状态抖动对 QoS 传输的影响,并且能够保证 QoS 资源控制的有效实施;同时对于相同数量的 QoS 传输,RCMQ 模型相比 InterServ 模型只需处理少量的 QoS 消息就能维护这些应用的状态,因而具有更高的可扩展性.

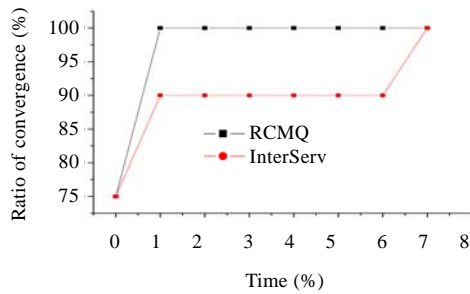


Fig. 11 Convergence speed of two models

图 11 两种模型的收敛速度

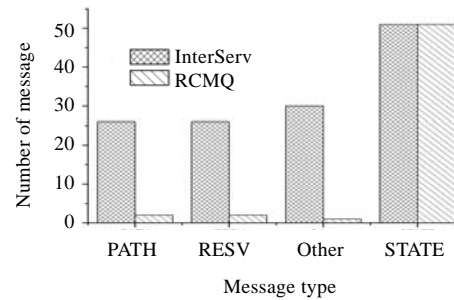


Fig. 12 Number of QoS message and state processed in AS per second

图 12 自治系统每秒处理 QoS 消息和维护状态的数目

5 总结与展望

服务质量保证难以实现一直是阻碍网络发展的重要因素.改进网络体系,使其能够提供服务质量保证已经迫在眉睫.我们认为,现有网络不能提供服务质量保证的根本原因是传统网络资源控制体系结构的不合理.结合最新的可信可控网络体系结构,本文提出了一种 QoS 资源控制模型 RCMQ,并通过论述和实验证明了它比传统的 QoS 资源控制体系具有更好的可扩展性和有效性.同时,由于可信可控网络体系只是修改了 TCP/IP 网络的控制结构而保留了其数据层面,因而 RCMQ 模型有利于在现有网络结构中实现部署.

由于对可信可控网络的研究还处于起步阶段,相应的控制结构和设计原则一直未能在大规模真实网络中进行验证.而服务质量保证正是评价新型网络体系的必要因素,因而本文后期的工作将会集中在对该模型在大规模级别实验床的测试上.

References:

- [1] Lan HL, Faynberg LI. An architectural framework for support of quality of service in packet networks. *IEEE Communication Magazine*, 2003,41(6):98–105. [doi: 10.1109/MCOM.2003.1204754]
- [2] Lamanna DD, Skene J, Emmerich W. SLAng: A language for defining service level agreements. In: *Proc. of the 9th IEEE Workshop on Future Trends in Networks*. Washington: IEEE, 2003. 100–107. [doi: 10.1109/FTDCS.2003.1204317]
- [3] Li YF, Qiu H, Lan JL, Wang BQ. A survey of buffer sizing for Internet routers. *Journal of Software*, 2008,19(3):733–743 (in Chinese with English abstract). <http://www.jos.org.cn/1000-9825/19/733.htm> [doi: 10.3724/SP.J.1001.2008.00733]
- [4] Crowcroft J, Hand S, Mortier R. QoS's downfall: At the bottom, or not at all! In: *Proc. of the ACM SIGCOMM Workshops*. Karlsruhe: ACM, 2003. 109–115. [doi: 10.1145/944592.944594]
- [5] Wang LJ, Wu JP. Time window mechanism to improve BGP routing convergence. *Journal of Software*, 2008,19(11):2979–2989 (in Chinese with English abstract). <http://www.jos.org.cn/1000-9825/19/2979.htm> [doi: 10.3724/SP.J.1001.2008.02979]
- [6] Lin C, Ren FY. Controllable: Trustworthy and scalable new generation Internet. *Journal of Software*, 2004,15(12):1815–1821 (in Chinese with English abstract). <http://www.jos.org.cn/1000-9825/15/1815.htm>
- [7] Lin C, Lei L. Research on next generation Internet architecture. *Chinese Journal of Computers*, 2007,30(5):693–711 (in Chinese with English abstract).
- [8] Lu LY, Varmanand P, Doshi K. Graduated QoS by decomposing bursts: Don't let the tail wag your server. In: *Proc. of the 29th IEEE Int'l Conf. on Distributed Computing Systems*. Montreal: IEEE, 2009. 11–21. [doi: 10.1109/ICDCS.2009.55]
- [9] Subramanian L, Stoica I, Balakrishnan H, Katz RH. OverQoS: Offering Internet QoS using overlays. *ACM SIGCOMM Computer Communication Review*, 2003,33(1):11–16. [doi: 10.1145/774763.774764]
- [10] Zhang ZL, Duan ZH, Gao LX, Hou YT. Decoupling QoS control from core routers: A novel bandwidth broker architecture for scalable support of guaranteed services. In: *Proc. of the ACM Conf. on Applications, Technologies, Architectures, and Protocols*

- for Computer Communication. ACM, 2000. 71–83. [doi: 10.1145/347059.347403]
- [11] Luo JZ, Han ZG, Wang LM. Trustworthy and controllable network architecture and protocol framework. Chinese Journal of Computers, 2009,32(3):391–404 (in Chinese with English abstract). [doi: 10.3724/SP.J.1016.2009.00391]
- [12] Greenberg A, Hjalmtysson G, Maltz DA, Myers A, Rexford J, Xie G, Yan H, Zhan JB, Zhang H. A clean slate 4D approach to network control and management. ACM Computer Communication Review, 2006,35(5):41–54. [doi: 10.1145/1096536.1096541]
- [13] Yan H, Maltz DA, Eugene TS, Gogineni H, Zhang H, Zhang C. Tesseract: A 4D network control plane. In: Proc. of the 4th USENIX Symp. on Networked Systems Design & Implementation (NSDI). USENIX, 2007. 369–382.
- [14] Caesar M, Caldwell D, Feamster N, Rexford J, Shaikh A, van der Merwe J. Design and implementation of a routing control platform. In: Proc. of the 2th USENIX Symp. on Networked Systems Design & Implementation (NSDI). USENIX, 2005. 15–28.
- [15] John JP, Katz-Bassett E, Krishnamurthy A, Anderson T, Venkataramani A. Consensus routing: the Internet as a distribute system. In: Proc. of the 5th USENIX Network Symp. on Design and Implementation. USENIX, 2008. 351–364.
- [16] Pei D, Azuma M, Messay D, Zhang LX. BGP-RCN: Improving BGP convergence through root cause notification. Computer Networks, 2005,48(2):175–194. [doi: 10.1016/j.comnet.2004.09.008]
- [17] Lin C, Shan ZG, Ren FY. Quality of Service of Computer Networks. Beijing: Tsinghua University Press, 2004. 187–195 (in Chinese).
- [18] Wang L, Saranu M, Gottlieb JM, Pei D. Understanding BGP session failures in a large ISP. In: Proc. of the IEEE Int'l Conf. on Computer Communications (INFOCOM). Anchorage: IEEE, 2007. 348–356. [doi: 10.1109/INFCOM.2007.48]
- [19] Wang F, Gao LX, Wang J, Qiu J. On understanding of transient inter-domain routing failures. In: Proc. of the IEEE Int'l Conf. on Network Protocol (ICNP). Boston: IEEE, 2005. 10–21. [doi: 10.1109/ICNP.2005.29]
- [20] Kushman N, Kandula S, Katabi D. Can you hear me now?! It must be BGP. ACM SIGCOMM Computer Communication Review, 2007,37(2):75–84. [doi: 10.1145/1232919.1232927]
- [21] Fu J, Sjodin P, Karlsson G. Loop-Free updates of forwarding tables. IEEE Trans. on Network and Service Management, 2008,5(1): 1–13. [doi: 10.1109/TNSM.2008.080103]

附中文参考文献:

- [3] 李玉峰,邱菡,兰巨龙,汪斌强.路由器缓存需求.软件学报,2008,19(3):733–743. <http://www.jos.org.cn/1000-9825/19/733.htm> [doi: 10.3724/SP.J.1001.2008.00733]
- [5] 王立军,吴建平.改善 BGP 路由收敛的时间窗口机制.软件学报,2008,19(11):2979–2989. <http://www.jos.org.cn/1000-9825/19/2979.htm> [doi: 10.3724/SP.J.1001.2008.02979]
- [6] 林闯,任丰原.可控可信可扩展的新一代互联网.软件学报,2004,15(12):1815–1821. <http://www.jos.org.cn/1000-9825/15/1815.htm>
- [7] 林闯,雷蕾.下一代互联网络体系结构研究.计算机学报,2007,30(5):693–711.
- [11] 罗军舟,韩志耕,王良民.一种可信可控的网络体系及协议结构.计算机学报,2009,32(3):391–404. [doi: 10.3724/SP.J.1016.2009.00391]
- [17] 林闯,单志广,任丰原.计算机网络的服务质量.北京:清华大学出版社,2004.348–356.



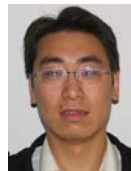
曲延盛(1982—),男,山东莱州人,博士生,主要研究领域为下一代互联网,资源控制.



李伟(1978—),男,博士,副教授,CCF 会员,主要研究领域为下一代网络体系结构,服务计算.



罗军舟(1960—),男,博士,教授,博士生导师,CCF 高级会员,主要研究领域为下一代网络体系结构,网络安全,网格与云计算,服务计算,无线局域网.



王鹏(1983—),男,博士生,主要研究领域为下一代互联网.