

The Convergence Data Flow Implementation Mechanism of IntServ in Ipv6

Liufang Wang

Department of Electronic Engineering, Hebi Automotive Engineering Career Academy, Hebi
Email: 172740089@qq.com

Received: Jun. 10th, 2014; revised: Jul. 8th, 2014; accepted: Jul. 16th, 2014

Copyright © 2014 by author and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

This article has analyzed the IntServ model, DiffServ model and MPLS model, and it absorbs their essence. All data flows from the same subnet to the same destination subnet will be gathered into a class. The class is acted as the unit of resource application, and the virtual circuit is established in the link meeting the QoS requirements; the virtual circuit is canceled until all the Ipv6 packets have been sent. This method not only guarantees the QoS requirements of data, but also reduces the number of the application resources; it has reduced the network cost suiting for the application in large-scale networks.

Keywords

Convergence, Routing Server, Subnet, Ipv6QoS

Ipv6中利用汇聚数据流实现QoS的新机制

王留芳

鹤壁汽车工程职业学院电子工程系，鹤壁
Email: 172740089@qq.com

收稿日期：2014年6月10日；修回日期：2014年7月8日；录用日期：2014年7月16日

摘要

本文通过对IntServ模型、DiffServ模型和MPLS模型分析,汲取了它们思想精髓,将同一子网发向同一目的子网的所有数据流聚集成类,以类为单位进行资源申请,在满足QoS要求的链路上建立虚电路,直到该类中所有的Ipv6分组都发送完后,取消虚电路。这种方法既保证数据的QoS要求,也减少了资源申请的次数;降低网络成本,适合在大型网络中应用。

关键词

汇聚, 路由服务器, 子网, Ipv6QoS

1. 引言

1.1. 集成服务(IntServ)

集成服务[1](Integrated Service, IntServ)是 IETF 针对服务质量(Quality of Service, QoS)提出的一种模型。该模型是一种基于流的、状态相关的结构。集成服务体系的核心协议是资源预留协议(resource reservation protocol, RSVP), RSVP 通过信令[2]在应用程序和网络节点间进行 QoS 协商。源端按照选定的路径向接收端发送 PATH 消息和业务特性信息,并在所经过的链路上搜集各个节点的信息,这些信息包括最小的可用带宽和最小的路径延迟等。

这些消息传送到接收端后,接收端按照应用的延迟要求,计算沿途允许的排队延迟。然后选择满足要求所需要的带宽。此时,接收端可发出 RESV 消息申请资源预留。RESV 消息沿原路返回,并在沿途路由器申请资源预留,包括带宽和延迟预留。中间路由器安装每个流的资源预留软状态(soft state),如果资源预留信息成功返回源端,源端可以在这条预留的路径发送数据流,否则无法发送。

集成服务模型的主要缺点是,中间的路由器必须是 RSVP 路由器,只要一个路由器不是 RSVP 路由器,资源预留无法实现,对中间路由器要求高。面对单流进行资源预留,使路由器计算处理和内存消耗大大增加,所以此模型不适合用在大型网络中。

1.2. 区分服务[3](DiffServ)

集成服务可扩展性很差,并没有获得广泛的应用,难以在 Internet 核心网络实施,DiffServ 主要解决网络的扩展性问题,用于骨干网。

DiffServ 把复杂性放在网络的边界,网络的边界路由器按照 QoS 的要求进行分类,Internet 服务供应商提供给每个用户不同等级和质量的服务。网络边界路由器把每个 Ipv6 分组中的通信类(Traffic Class)字段标记为区分服务(DS),DS 字段被称为区分服务码点(DSCP)。边界路由器把 DS 映射到类转发行为 PHB((Per-Hop Behavior, 每跳行为)中去。核心路由器查看 DSCP,并根据 PHB 值进行转发。

DiffServ 是聚集后的每一类 Ipv6 分组进行的 QoS 控制,而不是像 IntServ 那样针对单个流,因此,DiffServ 是粗粒度的 QoS 控制系统。容易出现拥塞。但具有扩展性,可在大型网站中使用。

1.3. 多协议标记交换[3](MPLS)

MPLS 在网络的入口路由器为每个 Ipv6 分组加上一个固定长度的标签,核心路由器根据标签直接转发,在出口的边缘路由器再恢复到原来的 Ipv6 分组。实现路由选择和数据转发相分离。

MPLS 域的边界路由器(LER)根据一定的路由协议选择路径, 选好路径后, 使用标签分发协议(LDP)为每个端口分发标签, 建立标签交换路径(LSP), 核心路由器按照标签沿路径进行转发。这样提高转发的速度。

其主要优点是减少了网络复杂性, 能降低网络成本, 在提供 Ipv6 业务时能确保 QoS 和安全性, 具有流量工程[4]能力。但它不能选择路由, 因此需要和 CBR 合作, 以便根据网络的流量特性和网络容量限制选择路由。

以上三种模型都是实现 QoS 的方法。IntServ 模型实现复杂, 成本高, 适用在小型网络中, 扩展性差。DiffServ 模型实现粗粒度的 Ipv6 分组汇聚的数据流, 容易出现拥塞。MPLS[5]不能选择路由。因此, 本文从网络的可扩展性, 减少网络拥塞[6], 可选择路由等方面提出了一种新型 Ipv6QoS 的实现方法。

本文的系统在边界路由器将同一子网发向同一目的子网的所有数据流聚集[7]成一个类, 以类为单位进行资源申请, 申请到资源后, 该类中所有 Ipv6 分组按照一定调度方法调度发送, 直到该类中所有 Ipv6 分组都发送完以后才取消预留的资源。

该系统以类为单位进行资源申请, 减少了资源申请的次数, 通过在系统中安装一个专用路由服务器[8]来控制资源的预留。

2. 实现 QoS 系统的结构

该系统的组成: 由边界路由器(入口路由器或出口路由器)、内部路由器和专用路由服务器组成。如图 1 所示。

2.1. 入口路由器(或出口路由器)

入口路由器的功能主要有以下三方面:

1) 把单个的流按照源地址的子网号和目的地址的子网号, 映射[9]到类。如图 2 所示。

当 Ipv6 分组到达入口路由器时, 入口路由器先判断 Ipv6 分组是 QoS 分组还是非 QoS 分组, 如果是非 QoS 分组[10]按尽力而为的方式发送, 如果是 QoS 分组, 从该分组的源地址中提取出源地址的子网号(如果没有划分子网就提取出网络号), 从该分组的地址中提取出目的地址的子网号。根据源地址的子网号和目的地址的子网号划分类。源地址的子网号和目的地址的子网号相同的分组为一类。这样入口路由器就把单个的 Ipv6 分组汇聚成一个流聚集[11]。

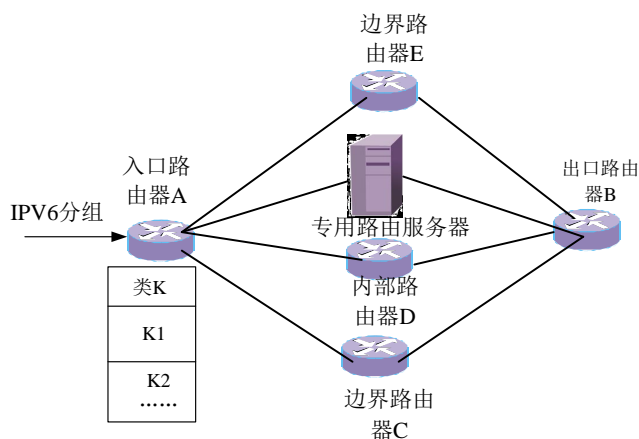


Figure 1. QoS system structure
图1. QoS系统的结构

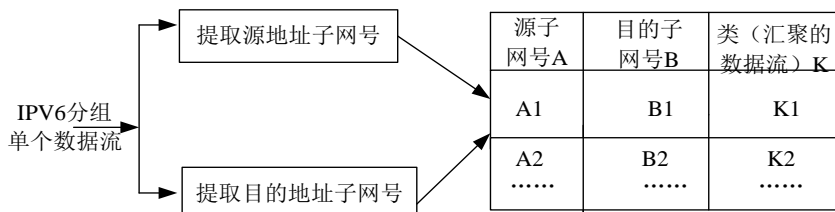


Figure 2. Ipv6 packets are mapped to the classes

图 2. Ipv6 分组映射到类

- 2) 资源申请, 入口路由器照数据流 QoS 要求, 对链路[12]进行资源申请。
- 3) 接入控制, 只有当链路的资源满足 QoS 要求时, 入口路由器才能接收发来数据流。

2.2. 专用路由服务器

在路由器服务器上动态地维护两个数据库。链路带宽利用率[13]数据库(LUDB)和链路延迟[14]数据库(LDDB)。

在图 1 中, 当 Ipv6 分组到达入口路由器 A, 先归入某一类(比如 K1), 在发送之前, 入口路由器 A 为类 K1 进行资源申请[15], 入口路由器 A 向专用路由服务器发送资源申请信息 request[16](请求), request 中包括入口路由器[17]地址, 出口路由器地址以及类 K1 中 Ipv6 分组 QoS 要求(包括链路带宽 r 和延迟 d)。

如果类 K1 中各个 Ipv6 分组 QoS 要求链路带宽分别为 $r_1, r_2, r_3, r_4, r_5, \dots, r_n$, 那么 r 是 $r_1, r_2, r_3, r_4, r_5, \dots, r_n$ 中最大的带宽; 如果类 K1 中各个 Ipv6[18]分组 QoS 要求延迟分别为 $d_1, d_2, d_3, d_4, d_5, \dots, d_n$, 那么 d 是 $d_1, d_2, d_3, d_4, d_5, \dots, d_n$ 中最小的延迟。

专用路由服务器收到 request 后, 计算从入口路由器 A 到出口路由器 B 之间所有路径的带宽和延迟。从入口路由器 A 到出口路由器 B 之间共有三条路径。分别是路径 1: A→E→B, 路径 2: A→D→B, 路径 3: A→C→B。路径 1 的带宽和延迟分别为: r_1 和 d_1 ; 路径 2 带宽和延迟分别为: r_2 和 d_2 ; 路径 3 带宽和延迟分别为: r_3 和 d_3 。

如何选择合适的路径? 一条路径符合 QoS 要求必须具备的条件: 链路实际带宽大于或等于分组 QoS 要求带宽, 链路的实际延迟小于或等于 QoS 要求带宽的延迟。如表 1 所示。

当有多条链路都满足 QoS 要求时, 如图 1、图 2, 那么就选择带宽最大, 延迟最小的那一条链路。如表 1、表 2 实际链路带宽、延迟的计算结果, 选择路径根据表 1~3 路径的选择。

表 1、表 2 中, 如果选择路径 1: A→E→B, 专用路由服务器就在路径 A→E→B 上建立一条虚电路, 同时向入口路由器 A 发送一个 answer[19](应答)消息, 其中, answer 消息中包括路径 A→E→B, 以及路径的带宽的数值 r_1 和延迟的数值 d_1 , 入口路由器 A 收到 answer 消息后, 就开始发送类 K1 中的数据分组。

当入口路由器 A 发送完类 K1 中的 Ipv6 分组后, 就向专用路由服务器发送一个穿上结束的 end 信息。专用路由服务器收到 end 信息后, 释放 A→E→B 虚电路[20]。

如果计算实际带宽和延迟的结果是没有符合要求的路径, 如表 4 所示, 就按表 3 的方法选择路径。

此时, 入口路由器 A 收到 answer 消息时, 只发送类 K1 中带宽 $< r_1$, 延迟 $> d_1$ 的 Ipv6 分组, 带宽 $> r_1$, 延迟 $< d_1$ 的 Ipv6 分组无法发送。

3. 系统的优势与改进

该系统中的内部路由器的功能比较简单, 当专用路由服务器选好路径, 建立虚拟电路以后, 按照虚拟电路发送数据就可以了。所以内部路由器是一般的路由器, 不需要是 RSVP 路由器, 这样减低了网络

Table 1. Calculating the actual link bandwidth and the delay (1)
表 1. 计算实际链路带宽、延迟的结果(1)

路径	实际带宽	实际延迟	判断 r, d	是否符合要求
1	r_1	d_1	$r_1 \geq r, d_1 \leq d$	符合
2	r_2	d_2	$r_2 < r, d_2 > d$	不符合
3	r_3	d_3	$r_3 < r, d_3 > d$	不符合

Table 2. Calculating the actual link bandwidth and the delay (2)
表 2. 计算实际链路带宽、延迟的结果(2)

路径	实际带宽	实际延迟	判断 r, d	链路是否符合要求
1	r_1	d_1	$r_1 \geq r, d_1 \leq d$	符合
2	r_2	d_2	$r_2 \geq r, d_2 \leq d$	符合
3	r_3	d_3	$r_3 \geq r, d_3 \leq d$	符合

Table 3. Choosing the path
表 3. 路径的选择

最大带宽	最小延迟	选择路径
r_1	d_1	路径 1: A→E→B
r_2	d_2	路径 2: A→D→B
r_3	d_3	路径 3: A→C→B

Table 4. Calculating the actual link bandwidth and the delay (3)
表 4. 计算实际链路带宽、延迟的结果(3)

路径	实际带宽	实际延迟	判断 r, d	链路是否符合要求
1	r_1	d_1	$r_1 \leq r, d_1 \geq d$	不符合
2	r_2	d_2	$r_2 \leq r, d_2 \geq d$	不符合
3	r_3	d_3	$r_3 \leq r, d_3 \geq d$	不符合

的成本; 以类为单位申请资源, 减少了资源申请的次数。

改进之处: 1) 如果当网络的规模比较大的时候, 组成多个不同的域。在每个域中安装专用路由服务器。2) 当类 K 中的 Ipv6 分组比较多时, 可以把类 K 再细分二级类, 每一类和一条链路绑定。

参考文献 (References)

- [1] 魏乐, 赵秋云, 舒红平 (2012) 云制造环境下基于 QoS 的组合云服务自适应调整. *兰州大学学报(自然科学版)*, **4**, 98-104.
- [2] 顾军, 夏士雄, 张瑾 (2007) IPv6 环境下端到端 QoS 模型. *计算机工程与设计*, **5**, 2037-2040.
- [3] 徐迎晓, 张根度 (2002) MPLS 和区分服务的集成. *计算机工程*, **4**, 154-156.
- [4] 谢锐, 顾一众, 汪为农 (2006) 基于 IPV6 源路由的一种 QoS 系统的研究与设计. *计算机应用与软件*, **2**, 84-133.
- [5] 刘念伯, 刘明, 吴磊, 曾家智 (2010) 一种在 MPLS 网络中提供单流 QoS 保障的区分服务标记方法. *计算机应用研究*, **27**, 1422-1426.
- [6] 王海涛 (2002) Internet 区分服务模型及其实现策略. *现代有线传输*, **1**, 47-50.

- [7] 王晓武, 党小超 (2006) 对于 IPv6 网络中服务质量的研究. *现代电子技术*, **16**, 70-74.
- [8] 戴慧珺, 曲桦, 赵季红 (2013) 一种覆盖网多 QoS 约束均衡的路由算法. *计算机工程*, **39**, 65-69.
- [9] Liu, K. and Lee, V.C. (2010) RSU-based real-time data access in dynamic vehicular systems. *IEEE Transactions on Vehicular Technology*, **56**, 3337-3347.
- [10] 葛连升, 江林, 秦丰林 (2010) QoS 组播路由算法研究综述. *山东大学学报(理学版)*, **45**, 55-65.
- [11] 刘丽, 方金云 (2013) 一种考虑 QoS 的 Web 服务组合 Petri 网建模方法. *计算机科学*, **40**, 37-39.
- [12] 肖芳雄, 等 (2011) Web 服务组合功能与 QoS 的形式化统一建模和分析. *软件学报*, **11**, 2698-2715.
- [13] 刘志忠, 王勇, 等 (2013) 服务组合中面向端到端用户 QoS 需求的 QoS 聚合机制研究. *计算机科学*, **40**, 18-21.
- [14] 李英壮, 孙梦, 等 (2011) 基于 OpenFlow 技术的 QoS 管理系统的设计与实现. *广西大学学报: 自然科学版*, **1**, 42-46.
- [15] 申利民, 陈真, 李峰 (2013) 一种考虑 QoS 数据不确定性的服务选取方法. *计算机集成制造系统*, **19**, 2652-2663.
- [16] 魏乐, 赵秋平, 等 (2012) 云制造环境下基于 QoS 的组合云服务自适应调整. *兰州大学学报: 自然科学版*, **4**, 98-104.
- [17] 申利民, 陈真, 李峰, 等 (2013) 一种考虑 QoS 数据不确定性的服务选取方法. *计算机集成制造系统*, **10**, 2652-2663.
- [18] 刘昕民, 桂卫华, 等 (2013) 一种基于 D-S 证据理论的 QoS 可信度评估方法. *哈尔滨工业大学学报*, **3**, 96-101.
- [19] 孙志刚 (2013) 一种 QoS 需求的资源优化算法. *微电子学与计算机*, **5**, 153-156.
- [20] 邓水光, 黄龙涛, 等 (2013) 一种 QoS 最优的语义 Web 服务自动组合方法. *计算机学报*, **5**, 1015-1030.