基于事件驱动的边界网关协议 BGP-4 的设计与实现*

徐 恪，吴建平，范晓勃

（清华大学 计算机科学与技术系，北京 100084）

E-mail：xuke@mail.cs.tsinghua.edu.cn

http://www.tsinghua.edu.cn

摘要：边界网关协议 BGP(border gateway protocol)是 Internet 上用于自治系统之间交换路由信息的动态路由协议.介绍了在清华大学研制的国产高性能路由器中分布式路由协议 BGP-4 的具体实现.为了实现这一复杂的动态协议，提出基于事件驱动的设计和实现方法.事件驱动机制能够精确地反映网络报文交换行为的实现机制，因而很适合 Internet 高层协议的实现.该方法也可以用于实现其他的 Internet 高层协议.

关键词：高性能路由器；路由协议；BGP；事件驱动

中图法分类号：TP393 **文献标识码**：A

清华大学计算机科学与技术系已经研制完成一种高性能的路由器，该路由器主要面向主干网互联的高端市场.高性能路由器是国家"九五"攻关重点课题，设计目标要求支持多种路由协议，并且实现高性能，在性能上要求达到 cisco7000 的水平.为了能够支持主干网互连，高性能路由器必须支持用于自治系统 AS(autonomous system)之间交换路由信息的 BGP-4 路由协议.BGP(border gateway protocol)是一个用于自治系统之间的路由协议，它的主要功能是在各个实现了 BGP 协议的系统之间交换网络可达性信息[1].

BGP-4 协议是一个比较复杂的分布式动态路由协议，到目前为止，各类国产路由器都还没有实现这一协议.因此，在国产高性能路由器中实现 BGP-4 协议不仅是设计的需要，也有利于我们掌握先进的 Internet 路由技术.

本文介绍了 BGP-4 协议的基于事件驱动机制的具体实现.第 1 节介绍了 BGP-4 协议，并分析了协议实现中的难点.第 2 节讨论了协议的基于事件驱动的设计与实现.第 3 节总结了全文.

1 边界网关协议 BGP-4 简介

BGP-4 是用于自治系统 AS 之间的路由协议，它的主要功能是在各个实现了 BGP-4 协议的系统之间交换网络可达性信息[1].这些信息包括一个路由所穿越的自治系统的列表，它们足以建立一个表示连接状态的图.这样便很容易地解决了路由环路问题，同时也使得在 AS 基础上的路由选择策略成为可能.从这一点上讲，BGP-4 是一个综合了向量-距离算法和链路-状态算法的协议.BGP-4 运行于可靠的传输协议之上，采用传输控制协议 TCP(transfer control protocol)作为其底层协议，这样便无须显式地进行分片、重传、确认和排序.

BGP-4 的路由选择刷新消息是由(网络号/AS 路径)对组成的，AS 路径是一系列 AS 名字的字符串，它记录了通向最终目标所经过的所有网络.两个运行 BGP-4 协议的路由器之间的初始数据交换是整个路由表，随着路由表的变化，发送的刷新消息也越来越多.与其他路由协议不同，BGP-4 不要求对整个路由表进行周期性地刷新，而是保持每一个路由表的最新版本.尽管 BGP-4 保持通往特定目标的所有路径的路由表，但是刷新消息中

* 收稿日期：1999-09-28；修改日期：1999-10-13

基金项目：国家自然科学基金资助项目(69725003)；国家 863 高科技项目基金资助项目(863-306-01-03-99；863-306-ZD-07-01)

作者简介：徐恪(1974－)，男，江苏洪泽人，博士生，主要研究领域为计算机网络体系结构，计算机系统性能评价；吴建平(1953－)，男，山西太原人，博士，教授，博士生导师，主要研究领域为计算机网络体系结构，网络协议测试；范晓勃(1973－)，男，山西太原人，硕士，主要研究领域为分布式路由协议.

© 中国科学院软件研究所 http://www.jos.org.cn

只发送主要(最佳)路径.BGP-4 的度量方法可以是一个任意单位的数,用它指明某一个特定路径的可参考程度,这些度量方法通常由网络管理人员通过配置文件来设置.可参考程度可以基于最终系统计数(计数越小路径越佳)和数据链路类型(链路速度、稳定性和可靠性)等.

1989 年,Internet 工程任务组 IETF(Internet Engineering Task Force)发表了 BGP 协议的第 1 个版本[2],次年发表了第 2 版[3],1991 年发表了第 3 版[4].这些版本被相应地称为 BGP-1,BGP-2 和 BGP-3.到目前为止,最新的版本是 1995 年制定的 BGP-4[1],我们的工作主要是针对 BGP-4 完成的.其他 BGP-4 的相关文献还有文献[5~13].BGP-4 的状态较多,交换的报文结构复杂,因此我们需要一种能够高效率地实现的机制.而事件机制是一种能够精确地反映网络报文交换行为的实现机制,因此,我们选择了事件机制来实现 BGP-4.

2 基于事件驱动的 BGP-4 协议软件的设计与实现

2.1 事件驱动机制概述

所谓事件驱动(event-driven)[14]机制,是指程序按照事件发生的次序随机执行而不是按照编程时就定义好的顺序执行,当某个事件发生时,程序将找到相应的事件处理程序来处理该事件.这种方式很适合网络协议软件的实现,在网络协议软件中,网络的行为是很难预测的,我们不可能预知事件的精确的发生次序,因此,一种比较好的实现方法就是采用事件驱动的机制来进行软件设计.

事件驱动软件的核心部分有两个:事件生成器和事件调度器.事件生成器负责根据网络的行为生成相应的事件,还要负责维护事件队列和时钟链,提供对这些数据结构进行操作的函数接口.事件调度器负责根据事件调度相应的事件处理程序.事件调度器也可能调用事件生成器生成其他事件.

下面,我们将详细讨论基于事件驱动的 BGP-4 协议软件的实现.

2.2 BGP-4 协议软件的总体设计

如图 1 所示,我们从功能上把协议软件分为初始化、事件生成器、有限状态机、报文处理、路由信息处理、协议报文成文与发送、输入/输出这 7 个模块,另外,还有配置信息和路由信息两个数据库.为了提高处理效率,减轻系统负担,整个软件设计为单进程结构,各个模块之间的交互采用函数调用和数据交换的方式.下面,简单叙述各个模块的功能.

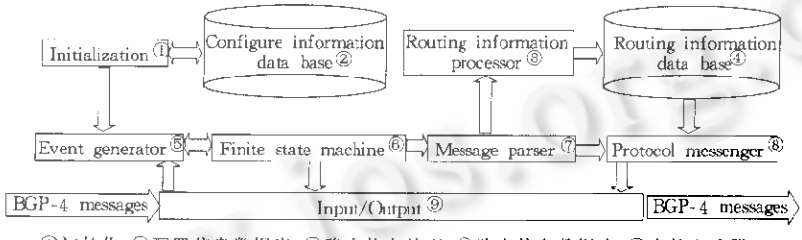


Fig. 1 The total architecture of BGP-4 implementation
图1 BPG-4协议实现总体结构图

• 初始化模块:对用户配置文件或控制台输入进行分析,从中读取 BGP-4 协议软件运行时所需的各项参数,同时进行语法和语义检查,把正确的参数写入配置信息数据库中,供其他模块在运行时检索引用;之后,初始化全局数据结构(如事件队列、时钟链等),并为某些结构分配空间;最后,向有限状态机 FSM(finite state machine)模块发送 Start 事件,启动 FSM 模块,进而使整个协议软件开始运行.

• 事件生成器:生成事件,从而驱动 FSM 模块.这些事件包括通过输入/输出模块收到 BGP-4 报文以及检测到 TCP 连接成功或失败,通过检查时钟链得到时钟超时事件.本模块还负责维护事件队列和时钟链,提供对这些数据结构进行操作的函数接口.

• 有限状态机 FSM:接收初始化模块和事件生成器送来的事件,作为有限状态机的输入,产生状态变迁及相应动作,控制报文处理模块的运行.本模块还通过输入/输出模块负责 TCP 连接的建立,同时还通过事件生成

© 中国科学院软件研究所 http://www.jos.org.cn

器产生新的事件.

• 报文处理模块:对收到的4种BGP-4报文进行正确性检查和分析.如果是UPDATE报文,还要通过调用路由信息处理模块更新路由信息数据库RIB(routing information base),调用模块协议报文成文与发送模块向相邻的内部BGP-4网关发送协议报文.本模块还在FSM的控制下,周期性地扫描路由器中的全局路由表,向相邻的外部BGP-4网关广播本地路由表的变化情况.

• 路由信息处理模块:从报文处理模块中得到要撤销的和声明为有效的路由,更新路由信息库RIB.为了减小RIB的规模,提高处理效率,这里还对RIB中的路由进行合并和压缩.另外,本模块还提供了对RIB中的数据结构进行操作的所有函数.

• 协议成文与发送模块:对4种BGP-4报文进行格式化,并调用输入/输出模块将其发出.

• 输入/输出模块:直接调用操作系统提供的TCP服务接口,完成TCP连接的建立和释放;接收相邻BGP-4网关送来的报文,提交给上层模块;接收上层模块发来的格式化为字节流的BGP-4报文,发送给相应的BGP-4网关.此外,对这些操作中产生的相应事件,本模块还通过事件生成器(event generator)发送给FSM.

• 配置数据库:存放BGP-4协议软件运行时所需的各项配置参数.这些参数可以来自配置文件,也可以由用户通过控制台输入.

• 路由信息数据库RIB:存放所有协议软件所产生和要利用的路由信息的数据库.实际上分为3个相互独立的数据结构:Adj-RIBs-In存放从其他BGP-4网关收到的路由信息;Loc-RIB存放路由器中本地路由表的映射;Adj-RIBs-Out存放向相邻BGP-4网关广播过的路由信息.

在以上模块中与事件机制相关的主要是事件生成器模块和FSM模块,其中事件生成器模块主要负责产生各类事件,FSM模块是整个软件的总控模块,它根据系统发生的事件调度相应的事件处理程序.下面,我们将详细讨论这两个模块.

2.3 事件生成器模块的设计与实现

如图1所示,整个BGP-4协议软件的运行核心是事件驱动型的,即由事件生成器产生各种事件,驱动FSM这一事件循环地执行.因而本模块实现了发送事件函数bgp_send_event(),供其他模块(如初始化、输入/输出)产生事件时调用;还实现了接收事件函数bgp_get_event(),供FSM模块调用,从而获取事件.另外,为了管理时钟及产生超时事件,这里还实现了各种时钟处理函数.

2.3.1 时钟链及时钟处理函数

时钟链是一个按照超时顺序由近到远排序的时钟链表,各模块启动的时钟都按序插入到此表中.若判断是否有时钟超时,只要从链表头节点(最先超时的时钟)开始,把当前时刻与此节点超时时刻比较,若比它小,则说明此时钟已超时,发送超时事件.为了加快处理速度和减轻插入负担,我们把时钟链设计为静态链表的结构,其程序描述如下:

```
typedef struct timerItem   /* 时钟项类型定义 */
{ bgpTime timeout_val;   /* 时钟超时值 */
  bgpPeer *peer;   /* 相邻网关 */
  int ev;   /* 超时后发送的事件 */
  int next;   /* 链表中的前一个节点 */
  int last;   /* 链表中的后一个节点 */
} timerItem;
typedef struct bgpTimerChain   /* 时钟链定义 */
{ int av_head;   /* 空闲链表头节点 */
  timerItem timer[BGPMAXTIMER];
} bgpTimerChain;
```

每个时钟用一个timerItem结构表示,其中timeout_val记录超时时刻,peer记录此时钟对应于哪个网关,ev是超时后要发送的事件,last和next分别指向链表中的前后节点.时钟链结构bgpTimerChain中的av_head指

© 中国科学院软件研究所 http://www.jos.org.cn

向当前未分配时钟构成的空闲链表的头节点.初始时,所有未分配时钟构成一个大的空闲链表.启动时钟时,只要把此节点摘下,插入时钟链的相应位置,并返回时钟号.因为时钟在数组 timer 中的位置不会变化,所以关闭时钟时可以通过时钟号把此节点直接摘下,放入到空闲链表中.

2.3.2 事件队列与事件处理函数

事件和事件队列的定义如下所示:

```
typedef struct bgpEvent   /* BGP 事件定义 */
{ int ev_type;   /* 事件类型 */
  int err_no;   /* 事件代号 */
  bgpPeer *peer;   /* 事件对应的网关 */
} bgpEvent;
typedef struct_bgpEventQueue   /* 事件队列定义 */
{ int head;   /* 队列头 */
  int tail;   /* 队列尾 */
  bgpEvent ev[BGPMAXEVENT];   /* 事件队列 */
} bgpEventQueue;
```

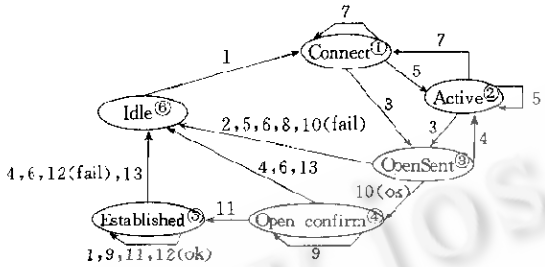
当获取事件时,只要检查事件队列是否为空,不空则返回队列头表示的事件.发送事件时,只要把事件加入队列尾即可.但是,由于整个软件采用了单进程结构,同时要与多个 BGP 4 网关交互,因而在获取事件时,要进行一次对所有网关 TCP 连接状态的轮询,查看是否有报文到达、是否有连接请求到达或是否以前的连接请求得到响应.函数 bgp_get_event 的算法流程如下所示.算法中的 bgpPeer 是一个描述 BGP-4 对等体的结构.

```
Bgp_get_event()
{
    获取 BGP 事件
    if 事件队列不空
        return 队列头部的事件;
    if 有数据到来
        for (i=0;i<当前的 BGP 连接数;i++)
            if 连接 i 有数据到来
            {
                由连接映射到 bgpPeer *pp,将数据读入报文缓冲区;
                判断报文类型,向 pp 发送相应的报文到达事件;
            }
    if 有连接建立好
    {
        while(所有建立好的连接)
            由连接映射到 bgpPeer *pp,向 pp 发送连接建立好(OPEN)事件;
    }
    if 有连接请求
        由连接映射到 bgpPeer *pp,向 pp 发送连接建立好(OPEN)事件;
    if 事件队列不空
        return 队列头部事件;
    if 时钟链头节点超时
        return 超时事件;
    return IDLE 事件;   /*表示当前没有需要处理的事件*/
}
```

© 中国科学院软件研究所 http://www.jos.org.cn

2.4 有限状态机 FSM 模块的设计与实现

从图 1 中可以看到,FSM 模块是整个协议软件的控制核心.它根据获取的事件及各有限状态机的当前状态,向其他模块发送控制流,从而协调整个软件的运行.前面提到,由于效率的原因,整个软件采用了单进程结构,这样便增加了程序设计的复杂性,即与所有网关的通信交互以及它们的状态变迁必须混合处理.因此,几乎所有的函数接口都有一个 bgpPeer * pp 这样的指针参数,用以指明当前的处理是针对哪个网关的.于是,FSM 模块便也不仅仅是代表一个状态机,而是代表一个状态机数组,只不过获取的事件、发出的控制(对其他模块的函数调用)及状态的转换都用一个参数 pp 加以区分.图 2 是根据 BGP 4 协议得到的有限状态机模型.



BGP events:⑦
1. BGP start,
2. BGP stop,
3. Establish transfer connection of BGP,
4. Close transfer connection of BGP,
5. Get error when establish transfer connection,
6. Transfer connection of BGP gets error,
7. Time out of connection retry clock,
8. Time out of waiting clock,
9. Time out of Keepalive clock,
10. Receive OPEN packet,
11. Receive KEEPALIVE packet,
12. Receive UPDATE packet,
13. Receive NOTIFICATION packet.

①连接,②活动,③发送启动,④发送证实,⑤连接建立,⑥空闲,⑦BGP流程:1. BGP启动,2. BGP停止,3. BGP传输连接建立,4. BGP传输连接关闭,5. BGP传输连接建立失败,6. BGP传输连接出错,7. 连接重试时钟超时,8. 等待时钟超时,9. Keepalive时钟超时,10. 收到OPEN报文,11. 收到Keepalive报文,12. UPDATE报文,13. 收到NOTIFICATION报文.

Fig. 2 The state machine of BGP protocol
图2 BGP协议状态机

图 2 所示的协议模型是相应于每一个网关运行状态的互相独立的状态机,所有事件和作出的动作都是针对某一个网关(某个 bgpPeer * pp)的.但是在具体实现时,还有一些事件的产生和响应是不依赖于某个网关的.例如,为了在 Active 状态监听连接请求,需要定期 bind TCP 的 179 端口,直到 bind 成功.这就需要在 bind 不成功时启动 REBIND 时钟,以便当它超时后发送 REBIND 事件而重新 bind.为了定期扫描本地路由表,需要启动 IGPDUMP时钟,它超时后发送 IGPDUMP 事件,FSM 收到后便扫描一次本地路由表,并根据其变化向外部相邻 BGP 网关发送路由更新报文.另外,如果当前事件队列空,bgp_get_event 返回 IDLE 事件,FSM 收到后便睡眠一段时间.由于 BGP-4 协议的报文交互不很频繁,计算负载很小,所以没有事件处理时适当地睡眠一段时间而让出 CPU,不会影响协议软件的响应,而且还解决了单进程轮询所带来的过多占用 CPU 的问题.

3 结 论

目前,BGP-4 协议已经设计完成,国家"九五"攻关课题——高性能路由器也已经通过鉴定.在实现完成之后,我们利用清华大学计算机科学与技术系研制开发的协议集成测试系统 PITS(protocol integrated test system)[15]对我们的 BGP 实现进行了一致性测试和性能测试,结果是令人满意的.由于 BGP 是一种分布式的路由协议,一致性测试和性能测试的过程比较复杂,具体的测试方法和测试结果将另文叙述.这里,我们只列出高性能路由器和 cisco7500 路由器的性能测试结果.我们研制的高性能路由器单板转发速率为 80kpps(64 字节长的分组),而同等条件下 cisco7500 的转发速率为 50kpps,因此,我们的实现在性能上达到并超过了 cisco 高端产品的性能,圆满地实现了设计要求.其中 BGP-4 协议软件的实现由于采用了单进程体系结构和基于事件机制的实现方法,因此,对系统的总体负载影响很小,对系统总体性能的提高起到了重要的作用.

References:

[1] Rekhter, Y., Li, T. A border gateway protocol 4 (BGP-4). RFC 1771, 1995.
[2] Lougheed, K., Rekhter, Y. A border gateway protocol (BGP). RFC 1105, 1989.
[3] Lougheed, K., Rekhter, Y. A border gateway protocol (BGP). RFC 1163, 1990.
[4] Lougheed, K., Rekhter, Y. Border gateway protocol 3 (BGP-3). RFC 1267, 1991.

© 中国科学院软件研究所 http://www.jos.org.cn

[5] Rekhter, Y., Gross, P. Application of the border gateway protocol in the Internet. RFC 1772, 1995.
[6] Traina, P. Experience with the BGP-4 protocol. RFC 1773, 1995.
[7] Traina, P. BGP-4 protocol analysis. RFC 1774, 1995.
[8] Haskin, D. A BGP/IDRP route server alternative to a full mesh routing. RFC 1863, 1995.
[9] Traina, P. Autonomous system confederations for BGP. RFC 1965, 1996.
[10] Bates, T., Chandra, R. Route reflection an alternative to full mesh IBGP. RFC 1966, 1996.
[11] Chandra, R., Traina, P., Li, T. BGP communities attribute. RFC 1997, 1996.
[12] Chen, E., Bates, T. An application of the BGP community attribute in multi-home routing. RFC 1998, 1996.
[13] Manning, B. Registering new BGP attribute types. RFC 2042, 1997.
[14] George, C. P. Software design guidelines for event-driven programming. The Journal of Systems and Software, 1998, 41(2): 79~91.
[15] Wu, Jian-ping, Chen, Xiu-huan, Hao, Rui-bin. Protocol integrated test system based on formal technology——PITS. Journal of Tsinghua University, 1998, 38(S1): 26~29 (in Chinese).

附中文参考文献:

[15] 吴建平,陈修环,郝瑞斌. 基于形式化技术的协议集成测试系统——PITS. 清华大学学报,1998,38(S1):26~29.

Design and Implementation of Border Gateway Protocol BGP-4 Based on Event-Driven Programming

XU Ke, WU Jian-ping, FAN Xiao-bo

(*Department of Computer Science and Technology, Tsinghua University, Beijing* 100084, *China*)

E-mail: xuke@mail.cs.tsinghua.edu.cn

http://www.tsinghua.edu.cn

Received September 28, 1999; accepted October 13, 1999

Abstract: Border Gateway Protocol (BGP) is a distributed dynamic routing protocol for exchanging routing information among autonomous systems. In this paper, the implementation of distributed routing protocol BGP-4 in high performance router developed by Tsinghua University is presented. A method based on event-driven programming is presented to implement the complicated protocol. Event driven programming can reflect the real situation of networks, so it is suitable for the network programming. This method can also be used to implement other higher layer protocols of Internet.

Key words: high-performance router; routing protocol; BGP (border gateway protocol); event-driven programming

© 中国科学院软件研究所 http://www.jos.org.cn