**BGP状态机实验**

计21 2012011401 张梦豪

1. **实验目的**

通过BGP协议状态机实验，加深对协议状态机描述的理解，并掌握协议状态机的设计实现方法。本实验状态机取材于BGP路由协议，可以加深对于BGP路由协议的理解。

1. **实验说明**

实验原理性说明见实验书的实现说明部分。（P197）

1. **实验内容及实现思路**

（1）实验要求

本实验要求根据系统的各种输入事件，进行BGP状态的变迁，并根据BGP协议进行相应的处理。

（2）接口函数说明

本实验中需要实现的接口函数包括收到Open消息事件处理函数、收到KeepAlive消息事件处理函数、收到Notification消息事件处理函数、收到Update消息事件处理函数、TCP连接异常事件处理函数、计时器超时事件处理函数、BGP开始事件处理函数、BGP结束事件处理函数和收到连接结果事件处理函数。

1）收到Open消息事件处理函数：

BOOLEAN stud\_bgp\_FsmEventOpen (BgpPeer \*pPeer, BYTE \*pBuf, unsigned int len)

2）收到KeepAlive消息事件处理函数：

BOOLEANstud\_bgp\_FsmEventKeepAlive (BgpPeer \*pPeer, BYTE \*pBuf, unsigned int len)

3）收到Notification消息事件处理函数：

BOOLEAN stud\_bgp\_FsmEventNotification (BgpPeer \*pPeer, BYTE \*pBuf, unsigned int len)

4）收到Update消息事件处理函数：

BOOLEAN stud\_bgp\_FsmEventUpdate (BgpPeer \*pPeer, BYTE \*pBuf, unsigned int len)

5） TCP连接异常事件处理函数：

Void stud\_bgp\_FsmEventTcpException (BgpPeer \*pPeer, BYTE msgType)

6） 计时器超时事件处理函数：

Void stud\_bgp\_FsmEventTimerProcess (BgpPeer \*pPeer, BYTE msgType)

7） BGP开始事件处理函数：

Void stud\_bgp\_FsmEventStart (BgpPeer \*pPeer)

8）BGP结束事件处理函数：

Void stud\_bgp\_FsmEventStop (BgpPeer \*pPeer)

9）收到连接结果事件处理函数：

Void stud\_bgp\_FsmEventConnect (BgpPeer \*pPeer)

系统提供的函数如下：

1） 试图建立连接函数：

Voidbgp\_FsmTryToConnectPeer()

2）TCP段发送函数：

Void bgp\_FsmSendTcpData (char \*pBuf, DWORD dwLen)

（3）数据结构定义

本实验中，实验系统定义了如下BGP对等体结构

Struct BgpPeer

{

DWORD bgp\_dwMyRouterID; //本路由器的路由器ID

WORD bgp\_wMyAS; //本路由器所属的AS

DWORD bgp\_dwCfgHoldtime; //设置的holdtime时间值

BYTE bgp\_byState; //协议状态机

BYTE bgp\_bAuth; //是否有认证信息

}

具体实现见BGP.cpp。

1. **思考问题**
2. 查找相关文献，思考BGP头部域应该如何填写，为何这样填写。

解：BGP头部域由三部分组成，Marker，Length和Type。Marker字段应该全部为1，Length为报文总长度，Type是消息类型。这样填写，便于使每个消息在到来时即可检查其相关信息，防止出错。

1. 实现状态变迁可以以事件为索引，也可以以状态为索引，试比较这两种方式。

解：本题采取的是事件为驱动的方法，当然也可以采取以状态为驱动的方法。本质上这都是描述状态机状态转换的方法，无本质区别，具体哪种更简洁需要因题而异。特别的，在本题中，个人觉得以状态为驱动描述起来更方便，因为在实验说明里是以状态为驱动描述的，而且本题的状态数相对较少，事件数相对较多，所以以状态描述的话将更加简洁。

1. **实验总结**

通过本次实验，让我对BGP状态机有了更加深刻的理解，本实验做起来非常纷繁复杂，6种状态，13种事件，一不小心就会忽略某个状态转换而出错，实在需要细心，耐心和用心，总体做下来还是收获蛮大的。