OSPFv2协议所需要的驱动接口定义

2016/04/01

1. **当收到一个数据包（IP包）时，需要底层驱动拆掉IP包头后，同时提供IP包头的信息，收到该包的端口的端口号以及数据包信息（包括数据包及数据包长度）。**

void Receive(struct Ipv4Header\* header, int interfaceId, char\* packet, int packetSize);

其中header为IP包头指针，interfaceId为端口号，packet为数据包指针，packetSize为数据包长度（字节数）。

【IP包头的结构体可以直接使用您所使用的SDK中提供的结构体，并希望能够给我们提供一份结构体定义，谢谢。】

【如果底层驱动对传入的数据包格式有要求，也请提供一份结构体的定义代码。但是希望能够尽力封装一下，让协议只提供必要的信息，谢谢。】

1. **发送数据包时，希望通过传入某端口的唯一标识（一般是端口号）来向该端口发包，从而通过该端口将包发送出去。另外，协议送出的包是没有IP包头的，需要底层驱动打上IP包头。**

bool Send(int interfaceId, char\* packet, int packetSize);

其中返回值为是否发送成功，packet为数据包指针，packetSize为数据包长度。

【返回类型也可以改为void】

1. **协议生成的路由表应该提供给IP层以供查询，这里需要看IP层的需求来定义更新的方式。**

【之前在仿真平台中的做法是IP层和OSPF协议模块中各有一个ForwardingTable类的指针，指向同一个ForwardingTable对象。因此OSPF协议更新完路由表后IP层可以直接使用最新的路由表。】

1. **端口的IP地址和子网掩码等信息不论由哪个模块来维护，最终都必须要使全系统统一才行。**

【因此如果由协议软件来维护的话，需要提供一个方法来通知IP层。如果由底层驱动维护的话，需要在配置完毕后、OSPF启动之前通知OSPF协议。】