

# RIP 协议实验报告

2013011371 计 34 沈哲言

## 一、 实验内容

- (1) 通过交互实验和编程实验，深入理解 RIP 协议的工作原理，了解 RIP 协议的分组接收和发送流程以及路由表的维护
- (2) 对客户端接收到的 RIP 报文进行有效性检查：对客户端接收到的 RIP 协议报文进行合法性检查，丢弃存在错误的报文并指出错误原因
- (3) 处理 Request 报文：正确解析并处理 RIP 协议的 Request 报文，并能够根据报文的内容以及本地路由表组成相应的 Response 报文，回复给 Request 报文的发送者，并实现水平分割
- (4) 处理 Response 报文：正确解析并处理 RIP 协议的 Response 报文，并根据报文中携带的路由信息更新本地路由表
- (5) 路由表项超时删除：处理来自系统的路由表项超时消息，并能够删除指定的路由
- (6) 路由表项定时发送：实现对本地路由的定时发送功能，并实现水平分割

## 二、 实验环境

Windows XP 操作系统、NetRiver 实验系统、客户端软件模拟一个网络中的路由器，在其中两个接口运行 RIP 协议，接口编号为 1 和 2，每个接口均与其他路由器连接，通过 RIP 协议交

## 互路由信息

### 三、 交互实验

交互实验主要通过界面互动的方式让我们能对 RIP 协议有一个直观的了解，而且交互实验能够更加直观地看到 RIP 分组报文的具体内容以及每一个字段的含义与取值，便于我们理解协议的规则与内容

实验的主要收获是深入理解了路由表项的修改规则，包括判断路由表项的相同与不同，在哪些情况下新增、在哪些情况下修改哪些字段等等，为后面的编程实验打下了基础。

### 四、 编程实验

#### (1) 合法性检查

合法性检查主要判据是 RIP 报头的命令字段和版本字段，分别占一个字节，**命令字段取值只能为 1 和 2，版本号只能为 2**，如果不符合这两个条件就调用 ip\_Discard()函数报错

#### (2) Response 分组

根据分组中携带的路由表项对本地路由表进行更新，主要注意事项为**判断路由表项的相同需要目的 IP 地址和子网掩码同时相同，更新 metric 值的时候 nexthop 也要一同更新，metric 值上限为 16**

#### (3) Request 分组

将本地的路由表(由全局变量 g\_rip\_route\_table)组成 response 分组发出去，注意事项为**根据收到分组的接口号将另一个接口的路由表发出去，本接口号的路由表不发，遵循水平分裂算法**

#### (4) 处理超时和定时发送

根据 `rip_route_timeout` 函数传入的参数来确定是超时还是定时发送, 超时只要简单地置表项为无效即可 (metric 置 16), 定时发送的话**需要注意在每个接口发送另一个接口的路由表**

### 五、 思考问题

- (1) 填写为 16 表示该路由表项无效, 因为最大的允许跳数为 15, RIP 协议的特点是“好消息传播快, 坏消息传播慢” 如果仅仅通过删除路由表项, 会使得该路由表项已失效的信息不能很快为周围路由器所知晓, 加大了数据丢包的可能性 (因为其实一条路已经不通了, 但路由器还不知道), 也同时浪费了带宽的资源
- (2) rip 协议支持的数据跳数很短, 只有 15 跳, 故先天就只适用于较小型的网络中, 且收敛速度比较慢, 还需要周期性的广播整个路由表, 这在大型网络中显然是不能实现的 (不论是效率还是缓存等问题), 相比之下 ospf 没有跳数的限制, 且收敛速度比较快, 划分了区域、节省了路由信息, 比较适合与较大型的网络之中。