简单的路由器项目报告

1. 项目概述

这个项目用提供的模拟环境实现并模拟了一个简单的路由器. 路由器支持IPv4, ICMP, ARP和RIP协议, 能够(1)正确转发IPv4数据包; (2)正确回复ICMP echo请求; (3)在ARP协议下获取相邻设备的物理地址; (4)在RIPv2下生成路由表; (5)在以上过程出错时正确发送ICMP数据包.

1. 代码结构和功能

总体上我们实现的代码分为2个模块, (1)数据层, 处理IPv4, ICMP和ARP; (2)控制层, 处理RIP.

1. IPv4.

收到以太网数据包后, 若为IPv4数据包, 则执行以下步骤: 若目标非自身, 则检查校验和, 如正确则转发该数据包(此时先查询路由表获取下一跳的目标IP地址, 如ARP缓存不知道IP地址对应的MAC地址, 则通知ARP并等待); 若目标为自身(或广播), 若是RIP数据包, 则交由控制层处理, 若是ICMP echo数据包, 则回复之(同样先查询路由表, 再查询MAC地址).

1. ICMP.

发送ICMP数据包的过程和上面回复ICMP echo数据包的过程一样. 不同情况下发送的类型和代码按照SimpleRouter.pdf中的实现. 此处注意到发送ICMP数据包时出错不应再引起ICMP数据包的发送, 否则可能导致死循环.

1. ARP.

若有数据包等待ARP, 在ARP请求被回复时, 发送这些数据包.

1. RIP.

init(): 对每一个网关出口, 发送一个RIP request的广播包.

run(): 每隔10s向所有网关出口发送一个RIP response. 此外，对于一个路由表项，若30s没有收到RIP包有对应的entry, 则从路由表中删除.

handlePacket(): 收到一个RIP包, 对每个entry先判断距离是否大于16, 如果大于则意味着超出网络边界, 不处理. 然后查路由表, 若路由表中没有对应的IP地址则加入一个新的表项. 否则与原来的路由表项作比对, 若距离更短则更新gateway.  若是收到一个request请求应回复一个response.

1. 制作过程和测试

在制作过程中, 我(高宇)负责数据层, 周耀达负责控制层, 最后合并调试, 大约工作了1人·天. 调试结束后, 能通过给出的数据点, 表明该路由器模拟实现了一个基本的路由器功能. 它距离我们实际使用的路由器还需要添加安全控制, 自动分配IP, 自动登陆宽带上网等功能.