

计算机网络LAB4——TCP协议实验

1400012849 李岩昊

1. 实验概述

实验要求主要包括:

1) 了解TCP协议的主要内容, 并针对客户端角色的、“停-等”模式的TCP 协议, 完成对接收和发送流程的设计。

2) 实现 TCP 报文的接收流程, 重点是报文接收的有限状态机。

3) 实现 TCP 报文的发送流程, 完成 TCP 报文的封装处理。

4) 实现客户端 Socket 函数接口。

选做:

1) 滑动窗口。由于将 TCP 协议简化为“停-等”模式, 实际上发送窗口和接收窗口的大小都为1, 因此也就不存在乱序接收的问题。发送流程也得到了很大的简化, 不再需要复杂的状态控制。可以进一步实现复杂的滑动窗口控制机制。

2) 拥塞控制。“停-等”模式的 TCP 协议, 其发送和接收窗口都为1, 不存在拥塞控制的问题。可以在实现复杂的滑动窗口机制的基础上, 进一步设计实现拥塞控制算法。

3) 往返时延估计。TCP 协议超时重传的间隔时间长短会影响到 TCP 协议甚至整个网络的性能, 如何确定重传时间间隔也是当前计算机网络领域的一个研究热点。现在主要采用的是一种动态自适应的算法, 根据网络性能的连续测量情况, 来不断地调整超时时间间隔。

2. 实验实现思路

本次试验仅完成必须部分, 由于代码长度多于300行且代码逻辑很好读懂, 因此不再防止过多代码截图, 说明中会给出具体实现位置, 详见打包提交中所附代码。TCP头信息字段定义和TCP节点数据结构定义如下, 存储结构使用的是带形式头部的链表。

```
typedef struct tcphead{
    uint16_t srcPort;
    uint16_t destPort;
    uint32_t seqNo;
    uint32_t ackNo;
    uint8_t headLen;
    uint8_t flag;
    uint16_t windowSize;
    uint16_t checksum;
    uint16_t urgentPointer;
    char data[1024];
};

typedef struct TCB{
    unsigned int srcAddr;
    unsigned int dstAddr;
    unsigned short srcPort;
    unsigned short dstPort;
    unsigned int seq;
    unsigned int ack;
    int sockfd;
    BYTE state;
    unsigned char* data;
};

typedef struct tcb_node{
    TCB* current;
    struct tcb_node* next;
};
```

1. 校验和和伪首部

具体涉及字段见教科书和TCP协议定义。此处不再赘述，分别在代码中实现为getChecksum()和getChecksum2()两个函数。

2. TCP状态机的设计

1. 设计思路：每当调用stud_tcp_send()发送数据时，直到收到ack并将ack递交给stud_tcp_input()后返回退出。每当stud_tcp_recv()收到一个数据包时，发送一个ack后再退出。
2. 接受流程状态机：查看当前TCB状态，检验FLAG，若合法则进入相应状态处理，否则报错退出。
3. 发送封装状态机：检查TCB是否为空，封装数据（设置各个字段），改变TCB状态，变换字节序并发送。
4. stud_tcp_socket()：检查domain, type等要求字段是否合法，新建TCB并初始化其各个字段值。更新sockfd, srcPort等。
5. stud_tcp_connect()：TCB状态更新为SYN_SEND，发送SYN等待应答。若收到ACK+SYN则将状态变为ESTABLISHED并返回ACK应答。
6. stud_tcp_send()：若sockfd合法且状态为ESTABLISHED，则调用tcp_stud_output()并一直等待ack，收到ack后调用tcp_stud_input()处理ack。
7. stud_tcp_recv()：若sockfd合法且状态为ESTABLISHED，则一直等待数据包，收到后调用tcp_stud_output()发送ack包。
8. stud_tcp_close()：删除当前TCB。若状态为ESTABLISHED，则发送FIN_ACK包并转状态为FINWAIT1，等待ACK。收到后转移到FINWAIT2并等待FIN_ACK。收到后状态转移为TIMEWAIT。

3. 实验中遇到的困难

1. 因为刚刚在许辰人老师的网络实习实验班LAB2上已经自主实现过一次自定义的TCP协议，对于TCP的工作原理和协议设计有了很深刻的印象。因此本次试验并未遇到非常困难的bug和错误的理解
2. 没了

4. 附录代码

见提交tar包中附的代码以及在线测试系统上的代码