

传输层

功能

- 向上层应用层提供通信服务
- 实现
 - 可靠传输
 - 差错控制
 - 顺序控制
 - 拥塞控制
- 区别
 - 网络层实现主机之间的通信
 - 传输层实现应用进程之间的通信

端口

- 区分上层应用的手段
- 复用和分用
 - 应用层多个应用进程通过传输层发送数据，传输层收到的数据必须交付给致命的应用进程
- HTTP服务80，FTP服务21

UDP

- 特点
 - 不可靠传输协议
 - 无连接
 - 没有进程接收数据，则返回ICMP“目的不可达”
- DNS, RIP

特点

- 可靠传输协议
- 面向连接一对一
- 全双工

传输管理

- 连接
 - 两个端点，套接字 (socket) (IP地址: 端口号)
- 首部格式
 - 源端口 2字节
 - 目的端口 2字节
 - 序号 4字节
 - 确认号ack 4字节
 - SYN: 同步 1bit
 - FIN: 终止 1bit
 - 窗口大小 2字节

TCP

连接管理

- 客户/服务器方式
 - 客户: 主动
- 步骤
 - 建立连接
 - 三次握手
 - 数据传送
 - 连接释放
 - 由任一方发起
- HTTP, FTP

可靠传输的实现

- 滑动窗口
 - 发送窗口的宽度
- 校验和+超时重传
- 流量控制
 - 接收窗口rwnd
- 拥塞控制
 - 由tcp重传机制引起
 - 拥塞窗口cwnd
 - 慢启动门限sssthresh
 - 慢开始
 - 拥塞避免
 - 快重传
 - 快恢复
 - cwnd < sssthresh 慢启动
 - cwnd加倍
 - cwnd > sssthresh 拥塞避免
 - cwnd+1
 - cwnd = sssthresh 都可
 - 拥塞后sssthresh减半
 - 发送窗口上限值为rwnd和cwnd中较小的一个

要点一 (2分): 主机根据对方的接收窗口大小和自身拥塞控制算法来计算发送窗口大小
要点二 (2分): 在对方接收窗口宽度和自身拥塞窗口宽度中取较小的一个
即: 发送窗口的上限值 = Min[rwnd, cwnd]
要点三 (1分): 每当收到对方发来的数据段, 都根据 TCP 报文段头部中的接收窗口大小重新计算以决定是否调整发送窗口; 另外, 传输过程中, 每经过一个传输轮次, 拥塞窗口变化时, 也将重新计算以决定是否调整发送窗口

