

# H1 第五章： 运输层

---

运输层为相互通信的应用进程提供逻辑通信

## H2 5.1 运输层协议概述

---

### 进程之间的通信

通信的真正端点不是主机而是主机中的进程，运输层就是搞这个的。运输层提供应用进程之间的逻辑通信，看起来通信就像一条笔直的路一样，实际上并不是，他只是向高层隐藏了细节。

H3 运输层需要对收到的报文进行差错检验，网络层只检验首部。

TCP--全双工的可靠信道

## H2 5.2 用户数据报协议UDP

---

### UDP概述

1. UDP无连接
  2. UDP尽最大努力交付
  3. UDP面向报文，上层传输过来的数据直接向下层交付，因此上层要控制合适的长度。
  4. UDP没有拥塞控制，接受端根本不发信息回来，根本没有办法拥塞。
  5. UDP支持一对一、一对多、多对多的交互通信。
  6. UDP的首部开销小，仅仅使用8字节。
- H3

UDP的首部格式：2字节的源端口，2字节的目的端口，2字节的长度，2字节的检验和字段，在检验的时候会有12个字节的伪首部。

## H2 5.3 控制传输协议TCP

---

## TCP最主要的特点

H3

1. TCP面向连接，上层使用TCP的时候需要先建立连接。
2. 每条TCP连接只能有两个端点，套接字。
3. TCP提供可靠交付。
4. TCP提供全双工通信。
5. TCP面向字节流，TCP不关心上层交给他的是怎么样的块数据，TCP都将视为字节流。会根据缓存和拥塞的程度来决定发送的字节。

H2

## 5.4可靠传输的工作原理

---

理想的传输条件有以下两个特点：

1. 传输信道不产生差错
2. 不管发送方以多大的速度传输数据，接收方总是来得及处理数据

但是显然，现实的网络不具备这些条件，但是可以通过协议来实现可靠传输。猜测TCP对第一点的实现在于错误重传，第二点的实现在于缓存窗口。

### 停止等待协议

简单的来讲，这个协议就是，发送出去的数据，接收方一定要发个序号回来确认，长时间没有确认返回的话，发送端认为丢失了，于是会进行重传。

需要注意的几点：

H3

1. 发送端在发送完之后保存一个副本，用于之后可能的重传。
2. 分组和确认分组要进行编号，毕竟是要比较的。
3. 重传时间设置应当比数组在分组传输的平均往返时间更长些。

确认丢失和确认迟到：

1. 确认丢失之后就会重传。
2. 确认迟到发送方不断重传，接收方不断接受并丢弃。

### 连续ARQ协议

简单来讲，连续ARQ协议就是，将发送的数据按编号放在缓存里，一个缓存最多放n个数据块。如果数据传送成功，发送端接收到接受端的信号，就将原先缓存中发送成功的块丢弃掉，加入新的块。如果发送失败，就要重新发送失败的数据块，这个称之为回退。

H3

## H2 TCP首部及后续其他

