

## 第五章 传输层 TCP可靠传输

可靠：保证接收方进程从缓存区读出的字节流与发送方发出的字节流是完全一样的。

TCP实现可靠传输的机制：

- 校验：与UDP相同
- 序号
- 确认
- 重传

**1、序号：**TCP首部的序号字段用来保证数据能有序提交给应用层，TCP把数据视为一个无结构但有序的字节流。序号建立在传送的字节流之上。

TCP连接传送的数据流中的每个字节都编上一个序号。序号字段的值是指本报文段所发送的数据的第一个字节的序号。

如下图所示，假设A和B之间建立了一条TCP连接，A的发送缓存区中共有10B，序号从0开始标号，第一个报文包含第0~2个字节，则该TCP报文段的序号是0，第二个报文段的序号是3。



图 5.9 A 的发送缓存区中的数据划分成 TCP 段

**2、确认：**TCP首部的确认号是期望收到对方的下一个报文段的数据的第一个字节的序号。

TCP默认使用累计确认，即TCP只确认数据流中至第一个丢失字节为止的字节。

例如，接收方B收到了A发送的包含字节0~2及字节6~7的报文段。

由于某种原因，B还未收到字节3~5的报文段，此时B仍在等待字节3，因此B到A的下一个报文段将确认号字段置3。

### 3、重传：

导致TCP对报文段进行重传：**超时**和**冗余ACK**。

#### (1) 超时

TCP每发送一个报文段，就对这个报文段**设置一次计时器**。计时器设置的**重传时间到期但还未收到确认时，就要重传这一个报文段**。

TCP采用**自适应算法**，**动态改变重传时间RTTs**（加权平均往返时间）

#### (2) 冗余ACK（冗余确认）：

每当比期望序号大的失序报文段到达时，发送一个冗余ACK，指明下一个期待字节的序号。

冗余确认过程：

##### 冗余ACK（冗余确认）

每当比期望序号大的失序报文段到达时，发送一个**冗余ACK**，指明下一个期待字节的序号。

发送方已发送1，2，3，4，5报文段

接收方收到1，返回给1的确认（确认号为2的第一个字节）

接收方收到3，仍返回给1的确认（确认号为2的第一个字节）

接收方收到4，仍返回给1的确认（确认号为2的第一个字节）

接收方收到5，仍返回给1的确认（确认号为2的第一个字节）

发送方收到**3个对于报文段1的冗余ACK** ➡ 认为2报文段丢失，重传2号报文段 **快速重传**

**TCP规定当发送方收到对同一个报文段的3个冗余ACK时，就可以认为跟在这个被确认报文段之后的报文段已经丢失。这种技术称为快速重传。**

