

计算机网络第三次作业

1711361 刘炼

南开大学网络空间安全学院

2019 年 11 月 22 日

1 第一题

1.1 问题

使用UDP 中校验和的计算方法计算下面三个16 位二进制数值的校验和（给出计算过程）

1011010011101000

0110111011000111

1110011100111000

1.2 解答

首先将此三个16位二进制数进行相加，具体为 $1011010011101000 + 0110111011000111 + 1110011100111000 = 100000101011100111$

可以看到，相加之后变为18位，将高两位右移16位后与低16位相加，如果将加法后的结果记为 n ，那么相应的操作为：

$$n \gg 16 + n \& 0x0000FFFF$$

因此，得到的结果为 $0000101011100111 + 10 = 0000101011101001$

对得到的结果取反，得到最终计算得到的校验和为 1111010100010110

2 第二题

2.1 问题

在城市A 和城市B 之间有一条Internet 主干网链路，其数据率为1Gb/s，往返时间（RTT）为100 毫秒，城市A 中的一台主机通过TCP 连接向城市B 中的一台主机发送数据，接收端通告的窗口从未大于1 兆字节，那么发送端可以达到的最大吞吐率是多少？

2.2 解答

以传输的窗口大小为1M进行计算，那么可以得到，传输中的最大吞吐率为：

$$throughput = \frac{W}{RTT} = \frac{1MB}{0.1s} = 10MBytes/s = 80Mbits/s$$

3 第三题

3.1 问题

分析下面捕获的TCP 报文片段，请回答如下问题：

```
TCP 1026 >http[ACK]Seq=51231 Ack=1 Win=65535 Len=1460
TCP 1026 >http[ACK]Seq=52691 Ack=1 Win=65535 Len=1460
TCP 1026 >http[ACK]Seq=54151 Ack=1 Win=65535 Len=1460
TCP http >1026[ACK]Seq=1 Ack=51231 Win=62780 Len=0
TCP 1026 >http[ACK]Seq=55611 Ack=1 Win=65535 Len=1460
TCP 1026 >http[PSH,ACK] Seq=57071 Ack=1 Win=65535 Len=892
TCP http >1026[ACK] Seq=1 Ack=52691 Win=62780 Len=0
TCP [TCP Dup ACK 98#1] http >1026 [Ack]Seq=1 Ack=52691 Win=62780
TCP [TCP Dup ACK 98#2] http >1026 [Ack]Seq=1 Ack=52691 Win=62780
TCP TCP 1026 >http[ACK] Seq=52691 Ack=1 Win=65535 Len=1460
TCP TCP 1026 >http[ACK] Seq=55611 Ack=1 Win=65535 Len=1460
```

3.2 解答

3.2.1 问题1

请问哪些是重传报文（写出其发送序列号），重传的原因分别是什么？

在发送端中，序列号Seq = 52691 和序列号 Seq = 55611 这两个报文为重传报文，其中序列号为Seq = 52691 的是因为接连收到了三次ACK，因此假设该序列的号所指示的数据丢失，需要重传；序列号为55611 没有收到相应的ACK，因此需要超时重传

3.2.2 问题2

ACK 报文中Win 字段的作用是什么？

Win字段的作用是为了显示在receiver的缓冲区中还可以接受多少个字节的数据，这控制了发送端对于数据传输的字节大小。

3.2.3 问题3

当接口层为不可靠的无线链路时（出错率较高），TCP 的拥塞控制机制对网络性能有何影响？简单进行解释。

(1)无线链路构造的无线网络的出错率较高：无线网络相比于有线有较高的出错率，会导致超时、重传等，这会启动拥塞控制机制降低TCP 的性能；

(2)TCP无法区分无线链路产生丢包的类型，可能是随即丢包，也有可能是拥塞丢包，所以在这都会让拥塞控制机制启动降低性能；

(3)无线链路的带宽有限：带宽较小导致发送端判断超时等的次数增多，启动拥塞控制从而降低性能。

4 第四题

4.1 问题

两台主机A和B，主机A上运行的Web服务器进程试图向主机B上的浏览器进程发送数据。对于每个TCP连接，主机A上的TCP维护一个512字节的发送缓存，主机B上的TCP维护一个1024字节的接收缓存。为了简单起见，假设TCP序列号从0开始。

4.1.1 问题1

主机B的TCP层从主机A按顺序接收到第560字节，浏览器进程只从中读出前60字节，那么在主机B发送给主机A的TCP段首部中的确认序列号（ACK#）和接收窗口大小（RcvrWindow Size）分别为多少？

4.1.2 问题2

在同一个TCP连接中，如果主机A的拥塞窗口设置成1个MSS（Maximum Segment Size, 536字节），主机B通告的流控窗口为560字节，主机A从主机B接收到的最后确认序列号为第700字节，主机A发送给主机B的最后字节为900。

假设主机A没有再收到ACK，它的窗口大小没有改变，那么主机A能够发送的最大字节号是多少？

4.1.3 问题3

假设主机A没有再收到另外的ACK，则运行在主机A上的Web服务器进程在阻塞前可以再向Socket写入多少字节？

4.2 解答

4.2.1 解答1

因为接收到的560个字节小于主机B缓存区的大小（1024字节），因此，主机B可以继续接收主机A中传来的数据，其确认的序列号（ACK#）应该为：561，主机B中由于已经读取了60字节的数据，那么，接受窗口大小通过计算，应为： $1024 - 560 - 60 = 524bytes$

4.2.2 解答2

A的发送缓冲区大小为512字节，而由于收到的确认序列号为700，故而应该从700字节往前发，至多发送512个字节，故而最大的字节号应该为： $700 + 512 - 1 = 1211$

4.2.3 解答3

由于700 900这201个字节是已经发出但是还没有确认的数据，故而需要在sender的缓存区中进行存储，故而，还可以写入的数据为： $512 - 201 = 311bytes$

5 第五题

5.1 问题

如下图5-1所示，纵轴表示TCP拥塞窗口大小，横轴为时间轴。请根据下图回答如下问题：

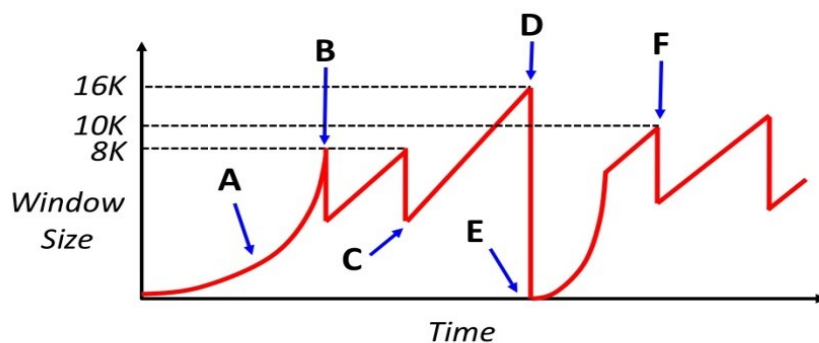


图 5-1 题目所示图

5.1.1 问题1

在图中B、D处分别发生了什么事件，B、D事件的发生表明在网络中一定有数据包被丢弃吗？

5.1.2 问题2

考虑图中A段曲线，为什么TCP拥塞窗口采取此种增长方式而非线性增长？

5.1.3 问题3

假设发送端在 $t=0$ 时刻开始建立了一个TCP连接，TCP连接的MSS为1000字节，发送端到接收端的往返延时（RTT）为100ms。那么到达B、C、D、F点所用的时间分别为多少？（假设发送端有充足的数据等待发送）

5.2 解答

5.2.1 解答1

(1)在B处, $ssthresh$ (阈值) 变为 $cwnd$ 的一半, $cwnd = ssthresh + 3$ 。因为慢启动和拥塞控制之间的阈值 $ssthresh$ 定义为 $cwnd$ 的一半, $cwnd = ssthresh + 3$, 故这是因为收到三次ACK, 检测到丢失的情况, 这种情况下发生了丢包。

(2)在D处, $ssthresh$ 变为 $cwnd$ 的一般, 且 $cwnd = 1$ 。产生这种情况的原因是系统检测ACK超时, 所以是在传输过程中的数据或者ACK loss 掉, 或者传输过程中, 被系统丢弃, 因此, 在D处可能的情况与B处不同。

5.2.2 解答2

这是因为在拥塞窗口 $cwnd$ 还未到达 $ssthresh$ (阈值) 之前, 拥塞控制机制采取的方法为慢启动, 慢启动的实质就是初始速率慢但以指数增长, 所以出现了非线性增长。

5.2.3 解答3

(1)从开始到达B点, 由于是从初始值1个MSS开始, 可以发现B处为8个MSS, 最开始为指数增长, 计算为: $cwnd = 2^{\frac{t}{RTT}} = 8$, 可以知道 $t_B = 3 \times RTT = 300ms$

(2)从B点开始到C点的过程中, 相当于是从 $ssthresh = 4, cwnd = 7$ 开始继续增长, 进行的是线性增长, 但是在传输之前, 从B点下降之前是接收到了三次冗余的ACK, 故需要的是 $\Delta t_1 = RTT$, 之后, 由 $cwnd = 7$ 到 $cwnd = 8$ 需要的是一个RTT, 故而需要的是 $\Delta t_2 = RTT = 100ms$, 之后相当于又是三次冗余ACK, 故而还需要 $\Delta t_3 = RTT = 100ms$, 故而最终的 $t_C = t_B + \Delta t_1 + \Delta t_2 + \Delta t_3 = 600ms$

(3)从C点到D点的过程中, 由之前的分析可知, 在C点, $cwnd = 7$, 那么相应的, 从 $cwnd = 7 \rightarrow cwnd = 16$ 需要的时间为 $\Delta t = 9 \times RTT = 900ms$, 因而, 可以得到 $t_D = t_C + \Delta t = 1500ms$

(4)之后E点 $ssthresh = 8, cwnd = 1$, 从D点到E点过程中需要的时间为: $\Delta t_1 = RTT = 100ms$, 然后E点慢启动过程需要的时间, 如(1)中所计算的那样, 需要 $\Delta t_2 = 3 \times RTT = 300ms$, 最后, 线性增加环节需要的时间为: $\Delta t_3 = 2 \times RTT = 200ms$, 故而, 到F点需要的时间为: $t_F = t_D + \Delta t_1 + \Delta t_2 + \Delta t_3 = 2100ms$