

HW3.

习题 15, 22, 23, 25, 27, 37, 40, 44, 45, 46

T15. 分组长1500字节, 即为12000比特

由信道利用率的公式: $U = \frac{L/R}{RTT + L/R}$

当设置窗口时 $U = \frac{n \cdot \frac{L}{R}}{RTT + \frac{n \cdot \frac{L}{R}}$ $U = 90\%$

$RTT = 30ms$

$$\frac{L}{R} = \frac{12000 \text{ bit/pkt}}{10^7 \text{ bit/s}} = 12 \mu s / \text{pkt.}$$

故求得 $n > 2250.9$

故应该有 2251 个数据包

T22. a. 窗口大小为4.

由接收方期待序号为 k 知发送方发送过 $(k-4), (k-3), (k-2), (k-1)$,

接收方均完整接收并传回 Ack, 因此接收方的预期序号为 k .

有2种极端情况: ① 这些 Ack 没有被发送方接收到, 故发送方窗口为 $[k-4, k-1]$

② 上述 Ack 全部被发送方接收到, 发送方更新基序号为 k ,

窗口为 $[k, k+3]$

综上, 发送方窗口内的报文序号为从 m 开始的 4 个连续序号,

且 $m \in [k-4, k]$

T23. 假设序号空间为 $0, 1, 2, \dots, k-1, 0, \dots$

对于 GBK, 由于 GBK 采用累积确认, 所以只需要窗口大小不大于 k 即可保证不会因失序分组而造成混乱。 $N \geq k$

对于 SR, 以接收方等待接收的最低序列号为 m 为例, 设窗口大小为 N , 则只有当 $N < \frac{k}{2}$ 时, 才可能出现接收方窗口的前端与发送方窗口的后端重叠的情况。

~~故 $N \geq \frac{k}{2}$~~ $N \geq \frac{k}{2}$

T25. a) 因为应用程序用 TCP 传输数据时会将数据写入缓存, 在发送时 TCP 不一定会传输段中的单个信息, 可能会将数据分成多个报文, 应用程序在一个报文段中具体发送的数据量、哪些数据无法控制。

而使用 UDP 时, UDP 会直接将数据分段, 打包进报文并立即传输。因此, 如果应用程序提供 UDP 应用消息, 这一消息会是 UDP 段的有效负载。对于 UDP, 应用程序对于报文段中发送什么数据有更多的控制。

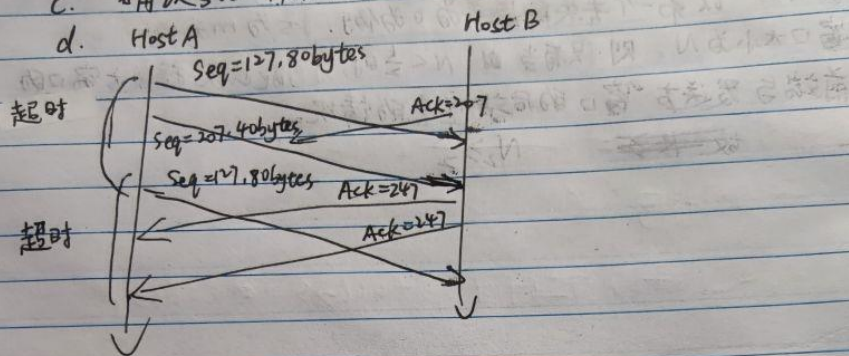
b) TCP 有流量控制和拥塞控制, 而 UDP 没有。
所以使用 UDP 时, 应用程序对何时发送报文段有更多的控制。

T27. a. 序号为 207, 源端口号为 302, 目的端口号为 80

b. 确认号为 ACK 207, 源端口号为 80, 目的端口号为 302

c. 确认号为 ACK 127

d. Host A



T37. a. GBN: A: 1 2 3 4 5 2 3 4 5 共 9 个

(因为 2 没有被成功接收所以 2, 3, 4, 5 重传)

B: 1 1 1 1 2 3 4 5 共 8 个

从 1 到 5, 只接收到了 1, 3, 4, 5, ACK 1;

后面从 2 到 5, 接收到了 4 个, ACK 2, ACK 3, ACK 4, ACK 5

SR: A: 1 2 3 4 5 2 共 6 个

2 没被收到, 2 重传

B: 1 3 4 5 2 共5个

2没接收到, 不影响其它确认号的返回。

TCP: A: 1 2 3 4 5 2 共6个

B: 2 2 2 2 6 共5个

2丢失后, B期待收到2, 所以发送2

直至接收到2后, 由于3, 4, 5已收到, 期待6。

b. TCP, 因为TCP有快速重传功能, 无需等待SRTT。

P40. a. TCP在 $[1, 6]$ 和 $[23, 26]$ 的时间间隔中慢启动。

b. TCP在 $[6, 16]$ 、 $[17, 22]$ 的时间间隔中拥塞避免运行。

c. 是由3个冗余ACK检测出的, 因为若是由于超时, 则拥塞窗口被置为1。

d. 是根据超时检测出的, 因为拥塞窗口被设置为1。

e. ssthresh 初始值为32, 这个值的窗口长度时,

慢启动停止, 拥塞避免开始。

f. 拥塞避免停止时, 窗口大小为42, 后慢启动, 窗口大小减半为21。

g. 与f原理相同, 取29的一半, 向下取整为14

h. 每一个传播轮回的窗口大小累加得, 在第7个发送传播轮回, 发送第7个报文段。

i. 阈值为拥塞窗口(8)的当前值的一半, 即为4。
窗口设置为 $(4+3)=7$

j. 阈值为21, 拥塞窗口大小为1

k) 17次, 1个分组; 18次, 2个分组; 19次, 4个分组;

20次, 8个分组; 21次, 16个分组; 22次, 21个分组。

总数为52。

P44. a. ^{cwnd.} 每个RTT增加1个MSS

则从6MSS到12MSS要花费6RTT

b. ~~on~~ cwnd 依次为 6MSS, 7MSS, 8MSS, 9MSS, 10MSS, 11MSS

$$\frac{6+7+8+9+10+11}{6} = 8.5 \text{ MSS/RTT}$$

P45 从连接速率为 $\frac{W}{2RTT}$ 到 $\frac{W}{RTT}$ 的过程中

发送的总分组数为:

$$\frac{W}{2} + (\frac{W}{2} + 1) + \dots + W = \frac{3}{8}W^2 + \frac{3}{4}W$$

$$\text{丢包率} = \frac{1}{\frac{3}{8}W^2 + \frac{3}{4}W}$$

b. W足够大时, $\frac{3}{8}W^2 \gg \frac{3}{4}W$ $L = \frac{1}{\frac{3}{8}W^2} = \frac{8}{3W^2}$

$$\text{即 } W = \sqrt{\frac{8}{3L}}$$

平均速率

$$\approx \frac{\frac{1}{2}(W + \frac{1}{2}W)}{RTT} \approx \frac{1.22 \times \text{MSS}}{RTT \sqrt{L}}$$

P46. a. 用W表示段中测量的最大窗口大小。

$$\text{则有 } \frac{W \times \text{MSS}}{RTT} = 10 \text{ Mbps}$$

$$\text{MSS} = 1500 \times 8 = 12000 \text{ byte}$$

$$RTT = 150 \text{ ms} = 0.15 \text{ s}$$

$$\frac{W \times 1500 \times 8}{0.15} = 10 \times 10^6 \quad W = 125$$

最大窗口长度为125个片段。

b. 当拥塞窗口在 $\frac{W}{2}$ 到 W 之间变化时, 平均窗口大小为 $0.75 W = 93.75$ 。

$$\text{平均吞吐量为 } \frac{94 \times 1500 \times 8}{0.15} = 7.52 \text{ Mbps}$$

c. 数据包丢失后, 拥塞窗口由 W 变为 $\frac{W}{2}$ 。

由于每个RTT, 窗口增加1, 故需要 $\frac{125}{2} \times 0.15 = 9.375$ 秒

$$(125 - L \frac{125}{2}) \times 0.15 = 9.45 \text{ s}$$