

计算机网络阶段测试

老师：朱敏

日期：2022.11.9

姓名：_____

学号：_____

I. 首字母缩略词与描述匹配，每个缩略词仅使用一次：Congestion Control, CNAME, FIN, Flow Control, IEEE, RFC, RST, TCP, TLD, UDP。(10 分，每题 2 分)

- 1) 最广泛使用的可靠传输协议。
- 2) 用于突然终止 TCP 连接的控制标志。
- 3) 发送方如何避免 TCP 连接中的接收方过载。
- 4) 拥塞控制是一种用来调整传输控制协议 (TCP) 连接单次发送的分组数量算法。

答案：1) TCP 2) RST 3) Flow Control
4) Congestion Control

解析：

1) TCP：传输控制协议(Transmission Control Protocol, TCP)是一种面向连接的、**可靠的**传输层协议。

干扰项：UDP，即用户数据报协议(User Datagram Protocol, UDP)，是一种无连接、**不可靠的**传输层协议。

2) RST，即 TCP 协议字段中的复位位 (Reset)。RST=1 时，表明 TCP 连接中出现严重差错（如主机崩溃或其他原因），必须**立即释放连接**，然后再重新建立运输连接。

干扰项：FIN，即 TCP 协议字段中的终止位(Finish)。用来释放一个连接。FIN=1 表明此报文段的发送方的数据已发送完毕，并要求释放传输连接。

3) Flow Control，即流量控制。TCP 提供流量控制服务来消除发送方使接收方缓存区溢出的可能性，因此可以说流量控制是一个速度匹配服务（匹配发送方的发送速率与接收方的读取速率）。

干扰项：Congestion Control，即拥塞控制，指防止过多的数据注入网络，以使网络中的路由器或链路不致过载。

二者区别：拥塞控制是让网络能够承受现有的网络负荷，是一个**全局性**的过程，涉及所有的主机、所有的路由器，以及与降低网络传输性能有关的所有因素。相反，流量控制往往是指**点对点**的通信量的控制，即**接收端控制发送端**，它所做的是抑制发送端发送数据的速率，以便使接收端来得及接收。

4) Congestion Control，即拥塞控制，指防止过多的数据注入网络，以使网络中的路由器或链路不致过载。

II. 单项选择题。(30 分，每题 3 分)

- 解析：**当 t 时刻发生超时，把 ssthresh 设为 8 的一半，即 4，把拥塞窗口设为 1KB。然后经历 10 个 RTT 后，拥塞窗口的大小依次为 2、4、5、6、7、8、9、10、11、12，然后发送窗口取当时的拥塞窗口和接收窗口的最小值，接收窗口初始为 10KB，所以此时的发送窗口为 10KB，选 A。

III. 判断正误 (T or F)。(10 分, 每题 2 分)

- 解析：仅需建立一个 TCP 连接。

- 解析：**发送窗口上限值 = $\min\{\text{接收窗口大小, 拥塞窗口大小}\}$ ，因此发送窗口的大小可以与接收方窗口的大小相同。

- 解析：TCP 并不支持广播。

- 4) A 和 B 建立了 TCP 连接, 当 A 收到确认号为 100 的确认报文段时, 表示末字节序号为 99 的报文段已收到。()

解析：TCP 的确认号是指明接收方下一次希望收到的报文段的数据部分第一个字节的编号，可以看出，前一个已收到的报文段的最后一个字节的编号为 99，所以此题是正确的。

IV. 综合题。(36 分)

1. UDP 报头的信息（十六进制）如表 1 所示。UDP 数据报的格式如图 2 所示。请回答以下问题：(8 分)

表 1: UDP 报头的信息 (十六进制)

Number	1	2	3	4	5	6	7	8
Data	F7	21	00	45	00	2C	E8	27

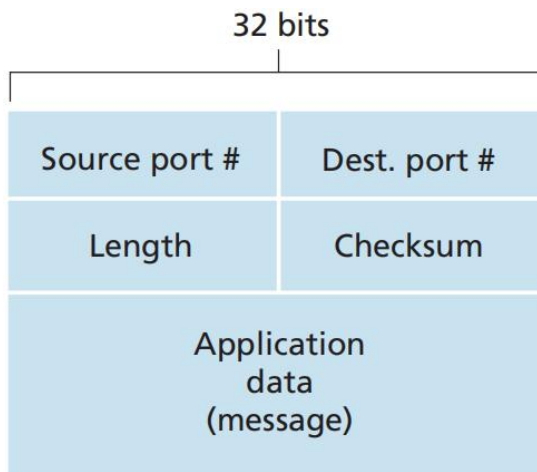


图 2: UDP 数据报的格式

- 1) 源端口、目的端口分别是什么? (4 分)
- 2) 数据报总长度、数据部分长度分别是什么? (4 分)

答案:

- 1) 源端口: F7 21, 转换成十进制数为 63265; 目的端口: 00 45, 转换成十进制数为 69。
- 2) 数据报总长度: 44B; 数据部分长度: 36B。

解析:

- 1) 第 1、2 个字节为源端口, 即 F721, 转换成十进制数为 63265。第 3、4 个字节为目的端口, 即 00 45, 转换成十进制数为 69。
- 2) 第 5、6 个字节为 UDP 长度(包含首部和数据部分)即 00 2C, 转换成十进制数为 44。数据报总长度为 44B, 减去 UDP 固定首部长度(8B)即为数据部分长度, 即 $44-8=36B$ 。

2. 主机 A 基于 TCP 向主机 B 连续发送 3 个 TCP 报文段。第 1 个报文段的序号为 90, 第 2 个报文段的序号为 120, 第 3 个报文段的序号为 150。(8 分)

- 1) 第 1、2 个报文段中有多少数据? (4 分)
- 2) 假设第 2 个报文段丢失而其他两个报文段到达主机 B, 在主机 B 发往主机 A 的确认报文中, 确认号应是多少? (4 分)

答案:

- 1) 第 1 个报文段: 30B; 第 2 个报文段: 30B。
- 2) 确认号: 120。

解析:

1) TCP 传送的数据流中的每个字节都有一个编号, 而 **TCP 报文段的序号为其数据部分第一个字节的编号**。因此第 1 个报文中的数据有 $120-90=30B$, 第 2 个报文中的数据有 $150-120=30B$ 。

2) **TCP 使用累计确认**, 因此当第 2 个报文段丢失后, 第 3 个报文段就成了失序报文, B 期望收到的下一个报文段是第 2 个报文段, 所以确认号为 120。

3. 假设 TCP Reno 是一个经历如图 3 所示行为的协议, 回答下列问题。(20 分)

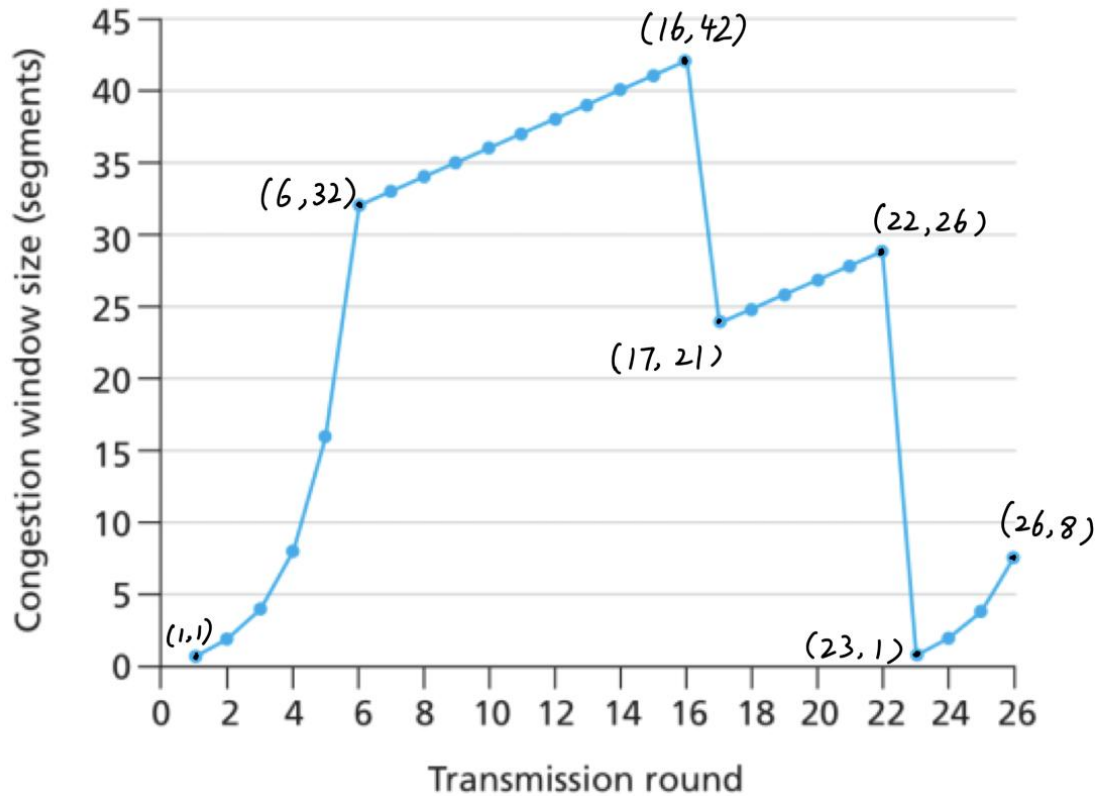


图 3: TCP 窗口大小与时间的关系

- 1) 在第 1 个传输轮回里, ssthresh 的初始值设置为多少? (2 分)
- 2) 在第 18 个传输轮回里, ssthresh 的值设置为多少? (2 分)
- 3) 在第 24 个传输轮回里, ssthresh 的值设置为多少? (2 分)
- 4) 在哪个传输轮回内发送第 70 个报文段? (2 分)
- 5) 假定在第 26 个传输轮回后, 通过收到 3 个冗余 ACK 检测出有分组丢失, 拥塞窗口长度和 ssthresh 的值应当是多少? (4 分)
- 6) 假定使用 TCP Tahoe (而不是 TCP Reno), 并假定在第 16 个传输轮回收到 3 个冗余 ACK。在第 19 个传输轮回, ssthresh 和拥塞窗口长度是什么? 如果在第 22 个传输轮回有一个超时事件, 从第 17 个传输轮回回到第 22 个传输轮回, 一共发送了多少分组? (8 分)

答案:

- 1) 32。
- 2) 21。
- 3) 13。
- 4) 第 7 个传输轮回。
- 5) 拥塞窗口长度为 4(或拥塞窗口长度为 7), ssthresh=4。
- 6) 拥塞窗口长度为 4, ssthresh=21; 共发送 52 个分组。

解析:

- 1) ssthresh 最初是 32。因为在此窗口大小下, 慢启动算法停止并开始执行拥塞避免算法。

2) 当检测到分组丢失时, 将 `ssthresh` 设置为拥塞窗口值的一半。当在传输回合 16 中检测丢失时, 拥塞窗口大小为 42。因此, 在第 18 个传输回合中, `ssthresh` 为 21。

3) 当检测到分组丢失时, 将 `ssthresh` 设置为拥塞窗口值的一半。在传输回合 22 中检测到丢失时, 拥塞窗口大小为 26。因此, 在第 24 传输回合中, `ssthresh` 为 13。

4) 在第一轮发送中, 发送分组 1; 在第二轮发送中, 发送分组 2~3; 在第三轮发送中, 发送分组 4~7; 在第四轮发送中, 发送分组 8~15; 在第五轮发送中, 发送分组 16~31; 在第六轮发送中, 发送分组 32~63。分组 64~96 在第七轮发送中发送。因此, 分组 70 在第七个传输轮回中被发送。

5) 当发生丢失时, `ssthresh` 将被设置为拥塞窗口 8 的一半, 并且拥塞窗口将被设置为新的 `ssthresh` 或设置为新的 `ssthresh+3`(详见计算机网络[第 7 版][谢希仁]P235)。因此, `ssthresh` 和拥塞窗口分别为 4 和 4(或 7)。

6) 使用 TCP Tahoe 意味着发生拥塞时将始终采用拥塞避免算法, 即 `ssthresh` 减半, 并将拥塞窗口重置为 1。在第 16 个传输轮回检测到拥塞, `ssthresh` 设置为 21。第 17 个传输轮回时, 拥塞窗口为 1。此后执行慢开始算法, 拥塞窗口依次变为 2、4。因此, 在第 19 个传输轮回 `ssthresh` 为 21, 拥塞窗口大小为 4。

第 17 轮回, 1 个分组; 第 18 轮回, 2 个分组; 第 19 轮回, 4 个分组; 第 20 轮回, 8 个分组; 第 21 轮回, 16 个分组; 第 22 轮回, 继续执行慢开始算法拥塞窗口大小将超过 `ssthresh`(32>21), 故此时停止慢开始算法而改用拥塞避免算法, 传送 21 个分组。因此, 从第 17 个传输轮回回到第 22 个传输轮回, 一共发送了 52 个分组。