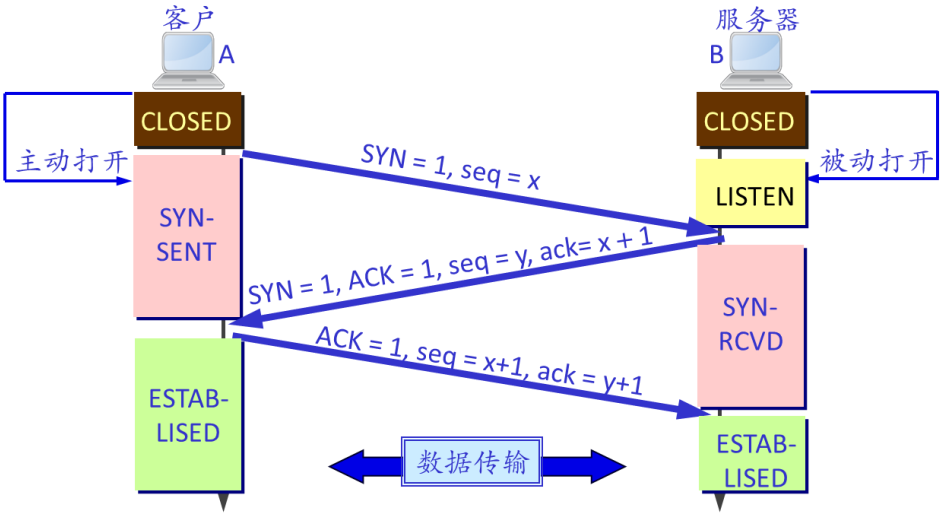
计算机网络第三次作业

李昊宸 2017K8009929044

1.简述 TCP 三次握手过程

答:



1）TCP 客户进程 A 主动打开连接：创建传输控制块 TCB，向 B 发送连接请求报文段，进入 SYN-SENT（同步已发送）状态。请求报文段首部的同步位 SYN 置 1，随机选择初始序号 SequenceNum 为 x。

2）B 收到请求后，应答确认报文段，进入 SYN-RCVD (同步收到)状态：确认报文段首部的标志位 SYN、ACK 都置 1，确认号 Acknowlegment = x+1，并随机选择自己的初始序号为 y。

3）A 收到 B 的确认后，向 B 应答确认，进入 ESTABLISHED (已建立连接)状态：该确认报文段首部的标志位 ACK 置 1，确认号为 y+1，序号为 x+1。B 收到 A 的确认后，也进入 ESTABLISHED (已建立连接)状态。

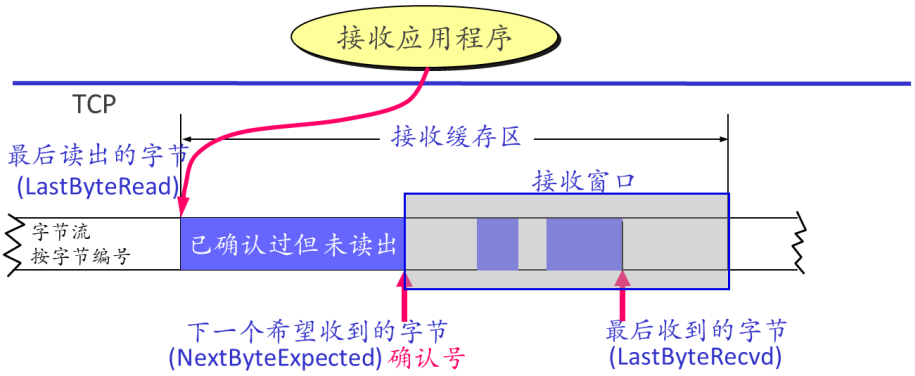
2. 请简述什么是流量控制和拥塞控制，TCP/IP 网络是如何解决这两个问题的?

答:

流量控制：防止快发送方给慢接收方发送数据造成接受崩溃，缓冲区溢出。

流量控制解决方案:

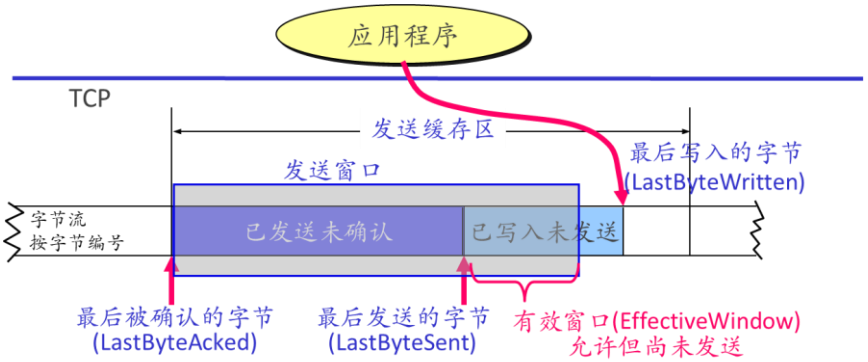
接收方：确定 AdvertisedWindow 大小



1）必须保持：LastByteRecvd – LastByteRead ≤ MaxRcvBuffer

2）AdvertisedWindow = MaxRcvBuffer – ((NextByteExpected – 1) - LastByteRead)

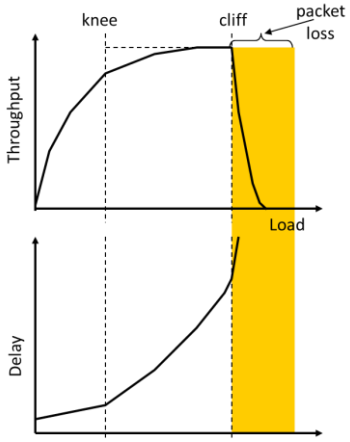
发送方：根据 AdvertisedWindow 值确定有效窗口，限制发送速率



1）有效窗口(EffectiveWindow) = AdvertisedWindow – (LastByteSent - LastByteAcked)，有效窗口大于 0 才能发送更多数据

2）发送方还必须同时保证发送缓存区不溢出：LastByteWritten – LastByteAcked ≤ MaxSendBuffer

拥塞控制：



网络中负载过大时，网络的性能会下降，当负载超过某阈值后，性能会急剧下降：

Knee 之后，吞吐率增长缓慢，延迟增长很快

Cliff 之后，吞吐率下降很快（到零），延迟增长很快（到无穷大）

为避免网络拥塞，TCP 采用端到端的拥塞控制策略：端设备通过丢包、延迟变化等推测网络拥塞状况。

TCP 采用两种手段进行拥塞控制：慢启动 + 拥塞避免

修改窗口大小：

MaxWindow = min (cwnd, AdvertisedWindow)

1）通知窗口（AdvertisedWindow）

接收方决定，可以同时发出的最大字节数以防止超出接收方的接收能力

2）拥塞窗口 cwnd (Congestion Windows)

拥塞控制算法决定，可以同时发出的最大字节数以防止造成网络拥塞

慢启动：

基本思想：主机开始发送数据（连接刚建立）或当判断拥塞发生时，不确定网络状况，应避免注入大量数据而引起拥塞。拥塞窗口大小从很小的初始值开始，发送成功则快速增大，以探测网络的负载能力。

拥塞窗口 cwnd 的初始值：

旧的规定：初始拥塞窗口 cwnd 设置为 1 至 2 个发送方的最大报文段 SMSS (Sender Maximum Segment Size) 的字节数

新的 RFC 5681：把初始拥塞窗口 cwnd 设置为不超过 2 至 4 个 SMSS

拥塞窗口 cwnd 的增长:

在每收到一个对新的报文段的确认后，把 cwnd 增加 1 个 SMSS 的数值数

拥塞避免:

拥塞窗口 cwnd 的增长:

在每收到一个对新的报文段的确认后，把 cwnd 增加 1 个 SMSS/cwnd 的数值数

慢启动与拥塞避免的界限:

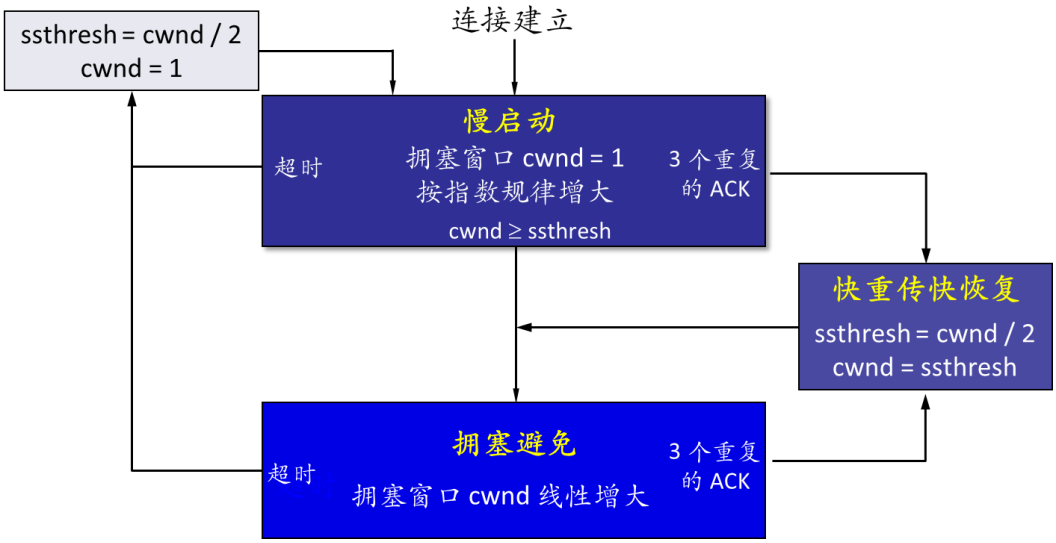
当 cwnd < ssthresh 时，使用慢启动

当 cwnd > ssthresh 时，使用拥塞避免

无论处于哪种算法，一旦超时，ssthresh 减小为当前 cwnd 值的一半，cwnd 置为最小值，执行慢启动。

在慢启动与拥塞避免的基础上，又引入了快重传+快恢复的算法:

收到 3 个重复 ACK 立即触发重传，在快重传之后，ssthresh 减小为当前 cwnd'的一半，cwnd 置为新的 ssthresh，执行拥塞避免。



3. 试用具体例子说明为什么 TCP 在进行连接建立时要采用“三次握手”，若客户端不向服务器端应答“三次握手”中的最后一个确认报文段，可能出现什么问题?

答:

假设 A 为客户端，B 为服务器。A 首先发出一个连接请求，但是在某些网络结点滞留；随后 A 又发出一个连接请求，B 收到，随后建立连接，事务处理结束后连接被释放。之后，被滞留的第一个请求到达 B, B 收到这个早应该失效的请求后, 误认为 A 又发送了一次连接请求，就像 A 发送确认报文段，同意建立连接。如果没有第三次握手，此时 B 单方面建立了连接，但 A 甚至不知道连接的建立，也不会向 B 发送数据，造成 B 的空等、资源浪费。

4. 主机 A 向主机 B 连续发送了两个 TCP 报文段，其序号分别为 60 和 100。试问:

(1) 第一个报文段携带了多少个字节的数据?

(2) 主机 B 收到第一个报文段后发回的确认中的确认号应当是多少?

(3) 如果主机 B 收到第二个报文段后发回的确认中的确认号是 150，试问 A 发送的第二

个报文段中的数据有多少字节?

（4）如果 A 发送的第一个报文段丢失了，但第二个报文段到达了 B。B 在第二个报文到达后向 A 发送确认。试问这个确认号应为多少?

（5）针对上述第 4 个问题描述的情况，主机 B 可以采取选择确认的方式，减少重复数的发送，请描述选择确认机制的基本原理。

答:

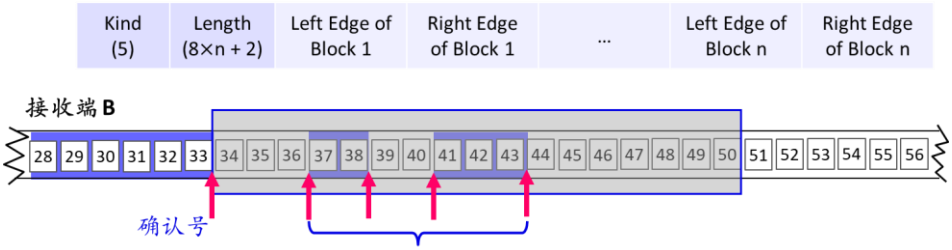
1）第一个报文段的范围为 60~99，共 40 字节的数据

2）主机 B 收到第一个报文段的最后一个字节的序号为 99，确认号应为 100

3）第二个报文段从 100 开始，到 150-1=149 结束，那么第二个报文端有 50 字节的数据

4）确认号为 70

5）选择确认（Selective ACK）：确认接收到的不连续的数据块的边界。使用首部的 SACK 选项



Left Edge 记录连续数据块的左边界，Right Edge 记录连续数据块的右边界

5. （1）长度为 250 字节的应用层数据交给运输层传送，需加上 25 字节的 TCP 首部。再交给网络层传送，需加上 25 字节的 IP 首部。最后交给数据链路层的以太网传送，加上首部和尾部 18 字节。试求数据的传输效率。若应用层数据长度为 1500 字节，数据的传输效率是多少?

（2）长度为 40 字节的应用层数据交给传输层传送，需要加上 25 字节的 TCP 首部。再交给网络层传送，需要加上 25 字节的 IP 首部。最后交给数据链路层的以太网传送，加上首部和尾部共 18 字节。试求数据的传输效率。若应用层数据长度为 400 字节，数据的传输效率是多少?

（3）一个 TCP 连接总是以 1KB 的最大段发送 TCP 段，发送方有足够多的数据要发送。当拥塞窗口为 16KB 时发生了超时，如果接下来的 4 个 RTT（往返时间）时间内的 TCP 段的传输都是成功的，那么当第 4 个 RTT 时间内发送的所有 TCP 段都得到肯定应答时，拥塞窗口大小是多少?

答:

1）250 /（250 + 25 + 25 + 18）* 100% = 78.61%

1500 /（1500 + 25 + 25 + 18）* 100% = 95.66%

2）40 /（40 + 25 + 25 + 18）* 100% = 37.04%

400 /（400 + 25 + 25 + 18）* 100% = 85.47%

3）cwnd 为 16KB 时发生了超时，慢启动设置阈值 cwnd 为 8KB

第一个 RTT：1KB -> 2KB

第二个 RTT：2KB -> 4KB

第三个 RTT：4KB -> 8KB

第四个 RTT：8KB -> 9KB

所以 cwnd 大小为 9KB

6. 假设一个 TCP 连接总是以 1KB 的最大段发送 TCP 报文段，且发送方有足够多的数据要发送，接收方有足够的接收能力（接收窗口足够大）。发送方以拥塞窗口为 1KB 开始发送，当拥塞窗口为 32KB 时发生了数据丢失而超时，在这之后的连续的 10 个 RTT（往返时间）时间内的 TCP 报文段的传输都是成功的，接着再往后因为一个 TCP 报文段传输时延过大而导致发送方接收到三个连续的重复确认。

（1）试画出拥塞窗口和传输轮次（1 个 RTT 时间为 1 个轮次）的时间曲线图。

（2）分别指明 TCP 工作在慢启动阶段、拥塞避免阶段的时间段（时间以轮次为单位）。

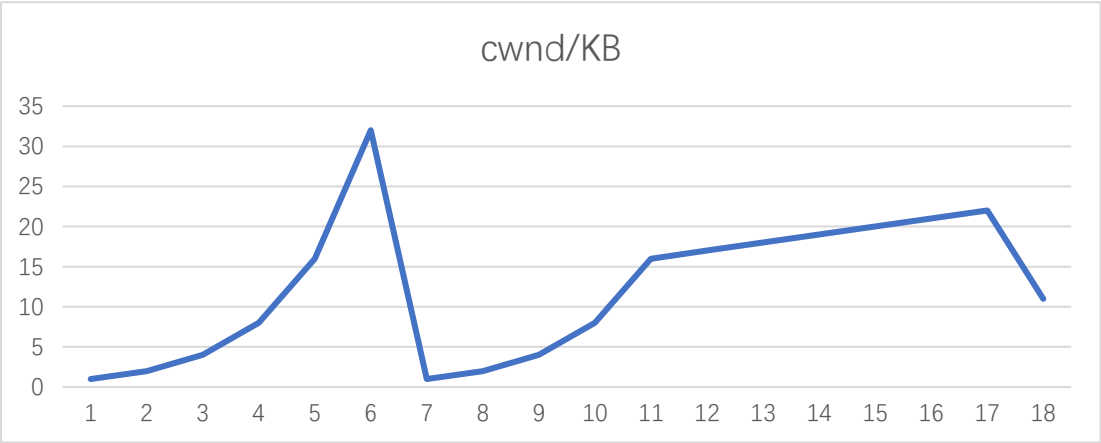
（3）在第 7 轮次，第 18 轮次发送时，拥塞窗口 cwnd 和门限 ssthresh 分别被设置为多大？

（4）在第几轮次发送出第 70 个报文段？

答：

1）& 2） & 3）

n	cwnd/KB	ssthresh	seq	status
1	1		1	慢启动
2	2		2~3	慢启动
3	4		4~7	慢启动
4	8		8~15	慢启动
5	16		16~31	慢启动
6	32		32~63	慢启动
7	1	16	重传 32	慢启动
8	2	16	33~34	慢启动
9	4	16	35~38	慢启动
10	8	16	39~46	慢启动
11	16	16	47~62	慢启动
12	17	16	63~79	拥塞避免
13	18	16	80~97	拥塞避免
14	19	16	98~116	拥塞避免
15	20	16	117~136	拥塞避免
16	21	16	137~157	拥塞避免
17	22	16	158~179	拥塞避免
18	11	11	重传 158~168	拥塞避免



4）如表格，在第 12 轮次发送出第 70 个报文段

7. 如下为 UDP 数据报首部格式。假设一个 UDP 用户数据报的首部的十六进制为 06 32 00 45 00 1C E2 17，请回答一下问题:

（1）请问源端口、目的端口、用户数据报的总长度、数据部分长度

（2）这个用户数据报是从客户发送给服务器还是从服务器发送给客户?

字节	2	2	2	2
	源端口	目的端口	长度	校验和

答:

注：所有回答不考虑网络字节序到本地字节序的转换!

1） 源端口：0x0632 = 1586

目的端口：0x0045 = 69

用户数据报总长度：0x001C = 28

数据部分长度：28-8 = 20

2)

熟知端口(well-known port)：0~1023

每个服务器进程在某个固定的熟知端口接收消息

定期在 RFC 公布，大多数可在 UNIX 系统的/etc/services 文件中得到

应用程序	FTP	SMTP	DNS	TFTP	HTTP	SNMP
熟知端口号	21	25	53	69	80	161

一般，仅仅是通信的起点，客户、服务器端在该端口达成一致，在另一端口进行后续通信，以释放该端口给其它客户进程使用

登记端口号：1024~49151

供服务提供商使用，需在 IANA 登记，防止重复

客户端端口：49152-65535

供客户端使用，动态选择

该数据包目的端口为 69，为熟知端口 TFTP，因此是从客户发给服务器的

8. 在如下图所示的应用场景中，用户 A 向视频服务器 B 申请下载大小为 100MB 的视频资源。A 和 B 之间的链路带宽为 100Mbps，往返时延为 100ms。下载流程为：在下载前，用户 A 需要向服务器 B 发送 ping 命令，探测服务器 B 是否有效；其后才会发起下载请求。针对上述网络参数和应用流程，计算用户 A 下载视频资源所能够获得的吞吐量（忽略各种消息在节点中的处理时延，忽略消息的头部开销）。

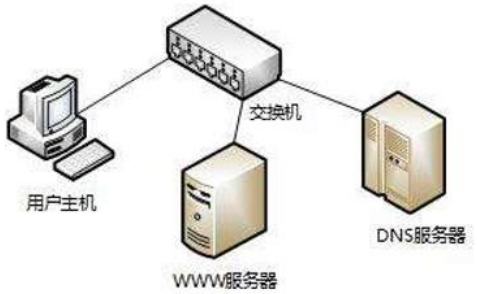
答：吞吐量 = 传输数据量 / 总时间

Ping 命令，花费 100ms

传输时间 = 100ms + （1/100Mbps）*100MB = 8.1s

总时间 = 8.1s + 100ms = 8.2s
吞吐量 = 100MB / 8.2s = 97.56Mbps

9. 在如下图所示的局域网中，交换机连通了用户主机、DNS 服务器、WWW 服务器，用户主机的 IP 地址是 192.168.1.1，DNS 服务器的 IP 地址是 192.168.1.2，WWW 服务器的域名为 www.2020kaoshi.org。假设用户主机刚刚启动，已经配置了 DNS 服务器的地址，请按步骤简要描述用户主机访问 www.2020kaoshi.org 主页的过程。要求：仅需描述每一步骤中所涉及的协议名称，以及该步骤的目的（不考虑主机自动配置 IP 地址的过程）。



答：

1）用户主机想要访问 www.2020kaoshi.org，但是既不知道目标网页的 IP 地址，也不知道 MAC 地址，也不知道 DNS 服务器的 MAC 地址，只知道 DNS 服务器的 IP。注意，此处用户主机应该发现了目的服务器与自己处于同一个局域网，所以第一步，主机广播 ARP 请求报文，询问 DNS 服务器的地址。协议：ARP
2）请求报文到达 DNS 服务器。DNS 服务器回应 ARP 应答报文。协议：ARP
3）主机收到 ARP 应答报文，获得 DNS 服务器的 MAC 地址，发送 DNS 查询请求报文。协议：DNS
4）DNS 检索本地缓存，查询到 www.2020kaoshi.org 对应的 IP，向用户主机发送 DNS 应答报文。协议：DNS
5）主机收到 DNS 应答报文，获得服务器的 IP。注意，此处用户主机应该发现了目的服务器与自己处于同一个局域网，所以，主机广播 ARP 请求报文，询问 WWW 服务器的 MAC 地址。协议：ARP
6）WWW 服务器收到请求报文。WWW 服务器回应 ARP 应答报文。协议：ARP
7）主机收到 ARP 应答报文，获得 WWW 服务器的 MAC 地址。最后，主机发送 TCP 报文请求建立连接。协议：TCP/IP

此题目中是主机与服务器同处同一个局域网，所以才会直接发送 ARP 报文询问 MAC 地址。如果不在同一局域网的话，一般会选择询问下一跳网关的 MAC 地址，设置目的 IP 为目标 IP，目的 MAC 为下一跳网关，将包发送出去。

10. DNS 服务（域名解析）、Web 服务、DHCP 服务（动态主机配置）是三种常见的 C/S（客户/服务器）模式的服务。在靠近客户端的一侧，这三种服务中分别定义了本地域名服务器、Web 代理、DHCP 代理等实体角色类型，请简要说明这三种实体的主要功能。

答：

本地域名服务器：又称递归服务器、默认域名服务器。本地域名服务器距离用户侧较近，保存了用户常用的 DNS 解析数据。当一个用户主机发送出 DNS 查询请求时，这个查询请求报文就发送给本地域名服务器，本地域名服务器相当于一个代理：如果主机询问的本地域名服

务器知道查询的目标域名映射，就将查询结果直接返回给用户主机；如果不知道，就代替用户主机，向根域名服务器、顶级域名服务器、权威域名服务器查询，最后将查询结果返回给客户主机。

Web 代理：把最近的一些请求和响应暂存在本地磁盘中，当与暂时存放的请求相同的新请求到达时，万维网高速缓存就把暂存的响应发送出去，而不需要按 URL 的地址再去互联网访问该资源，不仅可减少网络流量，同时提升传输性能。

DHCP 代理：基本上每个网络都需要 DHCP 服务，但是又不能让 DHCP 服务器数量太多，所以产生了 DHCP 中继代理。每个网络至少有一个 DHCP 中继代理，它配置了 DHCP 服务器的 IP 地址信息，并协助配置：

1）当 DHCP 中继代理收到主机发送的发现报文（DHCPDISCOVER）后，就以单播方式向 DHCP 服务器转发此报文，并等待其回答

2）收到 DHCP 服务器的提供报文（DHCPOFFER）后，DHCP 中继代理再将此报文发回给主机