陕西师范大学

计算机网络课后习题指导

Solutions for Problem Sets of Computer Networking

王涛 秦石醉 张阳阳 张敏江等 编著

6/25/2015

说明

本书由王涛、秦石醉、张阳阳、张敏江等人编著，部分内容参考自网络公开资源。编写过程难免有错误，请谅解。

本书习题基于英文经典计算机网络教材“Computer Networking A Top-down Approach”第六版。

Table of Contents

Chapter 1 Computer Networks and the Internet

P1. Design and describe an application-level protocol to be used between an automatic teller machine and a bank's centralized computer. Your protocol should allow a user's card and password to be verified, the account balance (which is maintained at the centralized computer) to be queried, and anaccount withdrawal to be made (that is, money disbursed to the user). Your protocol entities should be able to handle the all-too-common case in which there is not enough money in the account to cover the withdrawal. Specify your protocol by listing the messages exchanged and the action taken by theautomatic teller machine or the bank's centralized computer on transmission and receipt of messages. Sketch the operation of your protocol for the case of a simple withdrawal with no errors, using a diagram similar to that in Figure 1.2. Explicitly state the assumptions made by your protocol about the underlying end-to-end transport service.

答：本题不止一个答案，很多协议都能解决这个问题，下面是一个简单的例子：

Messages from ATM machine to Server

```
Msg name              purpose
---------             --------
HELO <userid>         Let server know that there is a card in the
                      ATM machine
                      ATM card transmits user ID to Server
PASSWD <passwd>       User enters PIN, which is sent to server
BALANCE               User requests balance
WITHDRAWL <amount>    User asks to withdraw money
BYE                   user all done
```

Messages from Server to ATM machine (display)

```
Msg name              purpose
---------             --------
PASSWD                Ask user for PIN (password)
OK                    last requested operation (PASSWD, WITHDRAWL)
                      OK
ERR                   last requested operation (PASSWD, WITHDRAWL)
                      in ERROR
AMOUNT <amt>          sent in response to BALANCE request
BYE                   user done, display welcome screen at ATM
```

Correct operation:

```
client                          server

HELO (userid)     --------------> (check if valid userid)
                  <-------------  PASSWD
PASSWD <passwd>   --------------> (check password)
                  <-------------  OK (password is OK)
BALANCE           -------------->
                  <-------------  AMOUNT <amt>
WITHDRAWL <amt>   --------------> check if enough $ to cover
                                  withdrawl
                  <-------------  OK
ATM dispenses $
BYE               -------------->
                  <-------------  BYE
```

In situation when there's not enough money:

```
HELO (userid)     --------------> (check if valid userid)
                  <-------------  PASSWD
PASSWD <passwd>   --------------> (check password)
                  <-------------  OK (password is OK)
BALANCE           -------------->
                  <-------------  AMOUNT <amt>
WITHDRAWL <amt>   --------------> check  if  enough  $  to  cover
withdrawl
                  <-------------  ERR (not enough funds)
error msg displayed
no $ given out
BYE               -------------->
                  <-------------  BYE
```

P2. Equation 1.1 gives a formula for the end-to-end delay of sending one packet of length L over N links of transmission rate R. Generalize this formula for sending P such packets back-to-back over the N links.

答：由一个分组端到端时延公式：$d_{end-to-end} = N\frac{L}{R}$可得，当连续发送 P 个分组时，得到时延为：$d_{end-to-end} = N\frac{L}{R} \times P$。

P3. Consider an application that transmits data at a steady rate (for example, the sender generates an N-bit unit of data every k time units, where k is small and fixed). Also, when such an application starts, it will continue running for a relatively long period of time. Answer the following questions, briefly justifying your answer:

a. Would a packet-switched network or a circuit-switched network be more appropriate for this application? Why?

b. Suppose that a packet-switched network is used and the only traffic in this network comes from such applications as described above. Furthermore, assume that the sum of the application data rates is less than the capacities of each and every link. Is some form of congestion control needed? Why?

答：a. 电路交换网更适合所描述的应用，因为这个应用要求在可预测的平滑带宽上进行长期的会话。由于传输速率是已知，且波动不大，因此可以给各应用会话话路预留带宽而不会有太多的浪费。另外，我们不需要太过担心由长时间典型会话应用积累起来的，建立和拆除电路时耗费的开销时间。

b. 由于所给的带宽足够大，因此该网络中不需要拥塞控制机制。最坏的情况下（几乎可能拥

塞），所有的应用分别从一条或多条特定的网络链路传输。而由于每条链路的带宽足够处理所有的应用数据，因此不会发生拥塞现象（只会有非常小的队列）。

P4. Consider the circuit-switched network in Figure 1.13. Recall that there are 4 circuits on each link. Label the four switches A, B, C and D, going in the clockwise direction.

a. What is the maximum number of simultaneous connections that can be in progress at any one time in this network?

b. Suppose that all connections are between switches A and C. What is the maximum number of simultaneous connections that can be in progress?

c. Suppose we want to make four connections between switches A and C, and another four connections between switches B and D. Can we route these calls through the four links to accommodate all eight connections?

答：a. 因为这 4 对相邻交换机，每对之间可以建立 4 条连接，因此最多可以建立 16 条连接。

b. 我们可以通过右上角交换机有四条连接，通过左下角交换机有四条连接，一共有 8 个连接。

c. 是的。对于 A 和 C 之间的链接，我们可以选择两条路径通过 B 两条路径通过 D。对于 B 和 D 之间的链接，我们可以选择两条路径通过 A 两条路径通过 C。在这种情况下，任意一条链路至少有四个连接。

P5. Review the car-caravan analogy in Section 1.4. Assume a propagation speed of 100 km/hour.

a. Suppose the caravan travels 150 km, beginning in front of one tollbooth, passing through a second tollbooth, and finishing just after a third tollbooth. What is the end-to-end delay?

b. Repeat (a), now assuming that there are eight cars in the caravan instead of ten.

答：由于收费站间隔 150km，车速 100km/h，收费站以每 12s 通过一辆汽车的速度提供服务。

a. 10 辆车，第一个收费站要花费 120s，即 2 分钟来处理。每一辆车要达到第二个收费站都会有 75/100=0.75 小时，即 45 分钟的传播延时，因此所有的车在通过第一个收费站时花费 47 分钟，在通过第二个收费站和第二条链路时同样花费 47 分钟，通过第三个收费站花费 2 分钟。因此，（端到端）总延时为 96 分钟。

b. 每两个收费站之间的延时为 8×12 秒＋45 分＝46 分 36 秒， 端到端总延时是该时延两倍再加上第三个收费站传输时延，即：

$$(46min + 36sec) \times 2 + 8 \times 12sec = 94min + 38sec$$

P6. This elementary problem begins to explore propagation delay and transmission delay, two central concepts in data networking. Consider two hosts, A and B, connected by a single link of rate R bps. Suppose that the two hosts are separated by m meters, and suppose the propagation speed along the link is s meters/sec. Host A is to send a packet of size L bits to Host B.

a. Express the propagation delay, d_{prop}, in terms of m and s.

b. Determine the transmission time of the packet, d_{trans}, in terms of L and R.

c. Ignoring processing and queuing delays, obtain an expression for the endto-end delay.

d. Suppose Host A begins to transmit the packet at time $t = 0$. At time $t = d_{trans}$, where is the last bit of the packet?

e. Suppose d_{prop} is greater than d_{trans}. At time $t = d_{trans}$, where is the first bit of the packet?

f. Suppose d_{prop} is less than d_{trans}. At time $t = d_{trans}$, where is the first bit of the packet?

g. Suppose $s = 2.5 \cdot 10^8$, $L = 120$ bits, and $R = 56$ kbps. Find the distance m so that d_{prop} equals d_{trans}.

答：a.传播时延: $d_{prop} = m/s$秒；

b.传输时延 $d_{trans} = L/R$秒；

c.端到端时延 $d_{end-to-end} = \mathrm{m/s + L/R}$秒；

d.该分组的最后一个 bit 刚刚离开主机 A；

e.第一个比特距离主机 A 为$\mathrm{d}_{trans} \times s$米，还没有到达 B；

f.第一个比特已经到达 B；

g.由$\frac{m}{s} = \frac{L}{R}$得，$\mathrm{m} = \frac{L}{R} \times s = \frac{120}{56 \times 10^3} \times 2.5 \times 10^8 m \approx 536km$。

P7. In this problem, we consider sending real-time voice from Host A to Host B over a packet-switched network (VoIP). Host A converts analog voice to a digital 64 kbps bit stream on the fly. Host A then groups the bits into 56-byte packets. There is one link between Hosts A and B; its transmission rate is 2 Mbps and its propagation delay is 10 msec. As soon as Host A gathers a packet, it sends it to Host B. As soon as Host B receives an entire packet, it converts the packet's bits to an analog signal. How much time elapses from the time a bit is created (from the original analog signal at Host A) until the bit is decoded (as part of the analog signal at Host B)?

答：考虑分组第一个比特 bit。在该 bit 传输之前，需要将分组所有的 bit 都产生，这个分组产生时延为：

$$\frac{56 \times 8}{64 \times 10^3} s = 7 \times 10^{-3} s = 7ms$$

然后是分组传输时延为：

$$\frac{L}{R} = \frac{56 \times 8}{2 \times 10^6} s = 0.224ms$$

最后由题意可得分组在链路上的传播时延为：10ms

综上，节点总时延为上面三个时延之和：

$$\mathrm{d}_{total} = 7 + 0.224 + 10 = 17.224ms$$

P8. Suppose users share a 3 Mbps link. Also suppose each user requires 150 kbps when transmitting, but each user transmits only 10 percent of the time. (See the discussion of packet switching versus circuit switching in Section 1.3.)

a. When circuit switching is used, how many users can be supported?

b. For the remainder of this problem, suppose packet switching is used. Find the probability that a given user is transmitting.

c. Suppose there are 120 users. Find the probability that at any given time, exactly n users are transmitting simultaneously. (*Hint*: Use the binomial distribution.)

d. Find the probability that there are 21 or more users transmitting simultaneously.

答：a. 当使用电路交换时，由于每一个用户共享带宽，所以用户数量为：

$$\frac{3M\mathrm{bps}}{150kbps} = 20个；$$

b. 由题知每个用户仅有 10%的时间在传输，所以任意时刻某用户传输概率为 10%；

c. 当有 120 个用户，其中恰好有 n 个用户同时传输的概率为：

$$p = \binom{120}{n} \times p^n \times (1-p)^{120-n}$$

d. 有 21 个或者更多用户在传输的概率为：

$$P = 1 - \sum_{0}^{20} \binom{120}{n} p^n (1-p)^{120-n}$$

我们用中心极限定理来求此概率，令X_j为独立随机变量，则$P(X_j = 1) = p$。

$$P(\text{"21 个或更多的用户"}) = 1 - P\left(\sum_{j=1}^{120} X_j \le 21\right)$$

$$P\left(\sum_{j=1}^{120} X_j \le 21\right) = P\left(\frac{\sum_{j=1}^{120} X_j - 12}{\sqrt{120 \times 0.1 \times 0.9}} \le \frac{9}{\sqrt{120 \times 0.1 \times 0.9}}\right)$$

$$\approx P\left(Z \le \frac{9}{3.286}\right) = P(Z \le 2.74) = 0.997$$

当 Z 为标准正态随机变量时，所求概率$P(\text{"21 个或更多的用户"}) \approx 0.003$。

P9. Consider the discussion in Section 1.3 of packet switching versus circuit switching in which an example is provided with a 1 Mbps link. Users are generating data at a rate of 100 kbps when busy, but are busy generating data only with probability $p = 0.1$. Suppose that the 1 Mbps link is replaced by a 1 Gbps link.

a. What is N, the maximum number of users that can be supported simultaneously under circuit switching?

b. Now consider packet switching and a user population of M users. Give a formula (in terms of p, M, N) for the probability that more than N users are sending data.

答：a. 链路带宽为 1Gbps，用户传输速率为 100Kbps，则可支持最大的用户数量为：

$$N = \frac{1Gps}{100Kbps} = \frac{10^9 bps}{100 \times 10^3 bps} = 10000 \text{ 台；}$$

b. 令传输用户数量为 n，则多于 N 用户发送数据的概率为：

$$\sum_{n=N+1}^{M} \binom{n}{M} P^n (1-p)^{M-n}$$

或者

$$1 - \sum_{n=0}^{N} \binom{n}{M} p^n (1-p)^{M-n}$$

P10. Consider a packet of length L which begins at end system A and travels over three links to a destination end system. These three links are connected by two packet switches. Let d_i, s_i, and R_i denote the length, propagation speed, and the transmission rate of link i, for $i = 1, 2, 3$. The packet switch delays

each packet by d_{proc}. Assuming no queuing delays, in terms of d_i, s_i, R_i, ($i = 1,2,3$), and L, what is the total end-to-end delay for the packet? Suppose now the packet is 1,500 bytes, the propagation speed on all three links is 2.5 ·10*8 m/s, the transmission rates of all three links are 2 Mbps, the packet switch processing delay is 3 msec, the length of the first link is 5,000 km, the length of the second link is 4,000 km, and the length of the last link is 1,000 km. For these values, what is the end-to-end delay?

答：由题意，设两个端系统为 A 和 B，分组交换器为S_1和S_2，链路传输速率为 R，由题意总时延由 A，S_1和S_2上的传输时延，在交换器S_1和S_2的处理时延以及在三条链路上的传播时延构成，则传输时延为：

$$d_{trans}=\frac{L_A+L_{S_1}+L_{S_2}}{R}=\frac{1500\times 8\times 3bit}{2\times 10^6 bps}=18\text{ms}$$

传播时延为：

$$d_{prop}=\frac{d_1+d_2+d_3}{s}=\frac{(5000+4000+1000)\times 10^3 m}{2.5\times 10^8 m/s}=40ms$$

则总时延为：

$$\text{d}_{total}=d_{tran}+d_{prop}+\text{d}_{proc}=18+40+3+3=64ms$$

P11. In the above problem, suppose $R_1 = R_2 = R_3 = R$ and $d_{\text{proc}} = 0$. Further suppose the packet switch does not store-and-forward packets but instead immediately transmits each bit it receives before waiting for the entire packet to arrive. What is the end-to-end delay?

答：由题意可知，分组链路传输速率相同，节点处理时延为 0，路由器不再存储转发而是在接收到后立即发送每一个比特 bit，所以在两个路由器上就不存在传输时延，类似的，传输时延为：

$$d_{trans}=\frac{L_A}{R}=\frac{1500\times 8bit}{2\times 10^6 bps}=6\text{ms}$$

传播时延不变：

$$d_{prop}=\frac{d_1+d_2+d_3}{s}=\frac{(5000+4000+1000)\times 10^3 m}{2.5\times 10^8 m/s}=40ms$$

端到端的总时延为：$6+40=46\text{ms}$

P12. A packet switch receives a packet and determines the outbound link to which the packet should be forwarded. When the packet arrives, one other packet is halfway done being transmitted on this outbound link and four other packets are waiting to be transmitted. Packets are transmitted in order of arrival.

Suppose all packets are 1,500 bytes and the link rate is 2 Mbps. What is the queuing delay for the packet? More generally, what is the queuing delay when all packets have length L, the transmission rate is R, x bits of the currently-being-transmitted packet have been transmitted, and n packets are already in the queue?

答：由题意，新到达的分组必须等待它之前到达的分组传输完成才能传输，之前分组包括等待传输的分组和正在传输的分组两部分，则：

a. 新的分组需要等待$1500\times 4+1500\div 2=6750\text{bytes}$的传输，即其排队时延为：

$$\frac{(1500\times 4+1500\div 2)\times 8bit}{2\times 10^6 bps}=27ms$$

b. 更一般的情况，当前分组已传输 x bits，队列有 n 个分组等待传输，则新到达的分组排队

时延为：

$$d_{queue}=\frac{n\times L+(L-x)}{R}$$

P13. (a) Suppose *N* packets arrive simultaneously to a link at which no packets are currently being transmitted or queued. Each packet is of length *L* and the link has transmission rate *R*. What is the average queuing delay for the *N* packets?

(b) Now suppose that *N* such packets arrive to the link every *LN/R* seconds. What is the average queuing delay of a packet?

答：a. 由题意，第一个分组没有排队时延，第一个分组排队时延为 0，第二个分组排队时延为$\frac{L}{R}$，第 N 个分组排队时延为$\frac{(N-1)\times L}{R}$，则平均排队时延为：

$$\left[0+\frac{L}{R}+\frac{2L}{R}+\cdots+\frac{(N-1)\times L}{R}\right]\times\frac{1}{N}=\frac{(N-1)L}{2R}$$

b. 由题意， N 个分组的传输时间为$\frac{LN}{R}$秒，第 1，2，3，…，N 个分组排队时延分别为：$0,\frac{L}{R},\frac{2L}{R}\cdots,\frac{(N-1)L}{R}$，则其平均排队时延为：

$$\frac{1}{N}\sum_{n=1}^{N}\frac{(N-1)\times L}{R}=\frac{1}{N}\times\frac{L}{R}\times\sum_{n=0}^{N-1}n=\frac{L}{R}\times\frac{N-1}{2}$$

P14. Consider the queuing delay in a router buffer. Let *I* denote traffic intensity; that is, *I* = *La/R*. Suppose that the queuing delay takes the form *IL/R* (1 – *I*) for *I* < 1.

a. Provide a formula for the total delay, that is, the queuing delay plus the transmission delay.

b. Plot the total delay as a function of *L/R*.

答：a. 传输时延为 L/R，总时延为：

$$d_{total}=d_{queue}+d_{trans}=\frac{IL}{R(1-I)}+\frac{L}{R}=\frac{L}{R(1-I)}$$

b. 令 L/R=x，则

$$d_{total}=\frac{x}{1-ax}$$

当 x=0 时，总时延为 0；当 x 逐渐增大是，总时延d_{total}也随之增大，当 x 接近 1/a 时，d_{total}趋近于无穷大。

P15. Let *a* denote the rate of packets arriving at a link in packets/sec, and let *μ* denote the link's transmission rate in packets/sec. Based on the formula for the total delay (i.e., the queuing delay plus the transmission delay) derived in the previous problem, derive a formula for the total delay in terms of *a* and *μ*.

答：由题，*a* 表示链路到达速率，单位为 packets/sec；*μ* 表示传输速率，单位为 packets/sec，其倒数 1/*μ* 即传输时延；由 14 题可知，总时延为：$d_{total}=\frac{L/R}{1-I}$，流通强度$I=\frac{aL}{R}$，带入 *a* 和 *μ* 得到总时延为：

$$d_{total} = \frac{L/R}{1-I} = \frac{L/R}{1-\frac{aL}{R}} = \frac{1/\mu}{1-a/\mu} = \frac{1}{\mu - a}$$

P16. Consider a router buffer preceding an outbound link. In this problem, you will use Little's formula, a famous formula from queuing theory. Let *N* denote the average number of packets in the buffer plus the packet being transmitted. Let *a* denote the rate of packets arriving at the link. Let *d* denote the average total delay (i.e., the queuing delay plus the transmission delay) experienced by a packet. Little's formula is $N = a \cdot d$. Suppose that on average, the buffer contains 10 packets, and the average packet queuing delay is 10 msec. The link's transmission rate is 100 packets/sec. Using Little's formula, what is the average packet arrival rate, assuming there is no packet loss?

答：分组的总数量等于缓存的等待传输的队列分组和正在传输的那一个分组，所以：N=10+1=11。

因为 $N = a \cdot d$，所以

$$10 + 1 = \mathrm{a} \times (d_{queue} + d_{trans})$$

$$11 = \mathrm{a} \times \left(0.01 + \frac{1}{100}\right)$$

解得$\mathrm{a} = 550\mathrm{packets/sec}$。

P17. a. Generalize Equation 1.2 in Section 1.4.3 for heterogeneous processing rates, transmission rates, and propagation delays.

b. Repeat (a), but now also suppose that there is an average queuing delay of d_{queue} at each node.

答：a. 有 Q 个节点（包括源节点和 Q-1 个路由器节点），用d_{proc}^{q}表示第 q 个节点的处理时延，R^q表示第 q 条链路的传输速率，分组由 L 个比特，则，第 q 个节点的传输速率为$d_{trans}^{q} = \mathrm{L}/R^q$，用$\mathrm{d}_{prop}^{q}$表示在第 q 条链路的传播速率，则端到端时延为：

$$\mathrm{d}_{end-to-end} = \sum_{q=1}^{Q} \left[\mathrm{d}_{proc}^{q} + d_{trans}^{q} + \mathrm{d}_{prop}^{q}\right]$$

b. 用d_{queue}^{q}表示在第 q 个节点的平均处理时延，则端到端时延为：

$$\mathrm{d}_{end-to-end} = \sum_{q=1}^{Q} \left[\mathrm{d}_{proc}^{q} + d_{trans}^{q} + \mathrm{d}_{prop}^{q} + \mathrm{d}_{queue}^{q}\right]$$

P18. Perform a Traceroute between source and destination on the same continent at three different hours of the day.

a. Find the average and standard deviation of the round-trip delays at each of the three hours.

b. Find the number of routers in the path at each of the three hours. Did the paths change during any of the hours?

c. Try to identify the number of ISP networks that the Traceroute packets pass through from source to destination. Routers with similar names and/or similar IP addresses should be considered as part of the same ISP. In your experiments, do the largest delays occur at the peering interfaces between adjacent ISPs?

d. Repeat the above for a source and destination on different continents. Compare the intra-continent and inter-continent results.

答：在 Linux 你可以用如下命令：

traceroute www.targethost.com

在 Windows 环境你可以用如下命令：

tracert www.targethost.com

不论在哪一种环境下做实验，你可以得到三组实验结果。在任意的三组实验中，你可以计算平均往返时延的均值和方差。在不同时间段内

做实验并且评论是否有所变化。

以下是实验截图：

```
traceroute to www.poly.edu (128.238.24.40), 30 hops max, 40 byte packets
 1  thunder.sdsc.edu (132.249.20.5)  2.802 ms  0.645 ms  0.484 ms
 2  dolphin.sdsc.edu (132.249.31.17)  0.227 ms  0.248 ms  0.239 ms
 3  dc-sdg-agg1--sdsc-1.cenic.net (137.164.23.129)  0.360 ms  0.260 ms  0.240 ms
 4  dc-riv-core1--sdg-agg1-10ge-2.cenic.net (137.164.47.14)  8.847 ms  8.497 ms  8.230 ms
 5  dc-lax-core1--lax-core2-10ge-2.cenic.net (137.164.46.64)  9.969 ms  9.920 ms  9.846 ms
 6  dc-lax-px1--lax-core1-10ge-2.cenic.net (137.164.46.151)  9.845 ms  9.729 ms  9.724 ms
 7  hurricane--lax-px1-ge.cenic.net (198.32.251.86)  9.971 ms  16.981 ms  9.850 ms
 8  10gigabitethernet4-3.core1.nyc4.he.net (72.52.92.225)  72.796 ms  80.278 ms  72.346 ms
 9  10gigabitethernet3-4.core1.nyc5.he.net (184.105.213.218)  71.126 ms  71.442 ms  73.623 ms
10  lightower-fiber-networks.10gigabitethernet3-2.core1.nyc5.he.net (216.66.50.106)  70.924 ms  70.959 ms  71.072 ms
11  ae0.nycmnyzrj91.lightower.net (72.22.160.156)  70.870 ms  71.089 ms  70.957 ms
12  72.22.188.102 (72.22.188.102)  71.242 ms  71.228 ms  71.102 ms
```

```
traceroute to www.poly.edu (128.238.24.40), 30 hops max, 40 byte packets
 1  thunder.sdsc.edu (132.249.20.5)  0.478 ms  0.353 ms  0.308 ms
 2  dolphin.sdsc.edu (132.249.31.17)  0.212 ms  0.251 ms  0.238 ms
 3  dc-sdg-agg1--sdsc-1.cenic.net (137.164.23.129)  0.237 ms  0.246 ms  0.240 ms
 4  dc-riv-core1--sdg-agg1-10ge-2.cenic.net (137.164.47.14)  8.628 ms  8.348 ms  8.357 ms
 5  dc-lax-core1--lax-core2-10ge-2.cenic.net (137.164.46.64)  9.934 ms  9.963 ms  9.852 ms
 6  dc-lax-px1--lax-core1-10ge-2.cenic.net (137.164.46.151)  9.831 ms  9.814 ms  9.676 ms
 7  hurricane--lax-px1-ge.cenic.net (198.32.251.86)  10.194 ms  10.012 ms  16.722 ms
 8  10gigabitethernet4-3.core1.nyc4.he.net (72.52.92.225)  73.856 ms  73.196 ms  73.979 ms
 9  10gigabitethernet3-4.core1.nyc5.he.net (184.105.213.218)  71.247 ms  71.199 ms  71.646 ms
10  lightower-fiber-networks.10gigabitethernet3-2.core1.nyc5.he.net (216.66.50.106)  70.987 ms  71.073 ms  70.985 ms
11  ae0.nycmnyzrj91.lightower.net (72.22.160.156)  71.075 ms  71.042 ms  71.328 ms
12  72.22.188.102 (72.22.188.102)  71.626 ms  71.299 ms  72.236 ms
```

```
 1  thunder.sdsc.edu (132.249.20.5)  0.403 ms  0.347 ms  0.358 ms
 2  dolphin.sdsc.edu (132.249.31.17)  0.225 ms  0.244 ms  0.237 ms
 3  dc-sdg-agg1--sdsc-1.cenic.net (137.164.23.129)  0.362 ms  0.256 ms  0.239 ms
 4  dc-riv-core1--sdg-agg1-10ge-2.cenic.net (137.164.47.14)  8.850 ms  8.358 ms  8.227 ms
 5  dc-lax-core1--lax-core2-10ge-2.cenic.net (137.164.46.64)  10.096 ms  9.869 ms  10.351 ms
 6  dc-lax-px1--lax-core1-10ge-2.cenic.net (137.164.46.151)  9.721 ms  9.621 ms  9.725 ms
 7  hurricane--lax-px1-ge.cenic.net (198.32.251.86)  11.365 ms  10.048 ms  13.844 ms
 8  10gigabitethernet4-3.core1.nyc4.he.net (72.52.92.225)  71.920 ms  72.977 ms  77.264 ms
 9  10gigabitethernet3-4.core1.nyc5.he.net (184.105.213.218)  71.273 ms  71.247 ms  71.291 ms
10  lightower-fiber-networks.10gigabitethernet3-2.core1.nyc5.he.net (216.66.50.106)  71.114 ms  82.516 ms  71.136 ms
11  ae0.nycmnyzrj91.lightower.net (72.22.160.156)  71.232 ms  71.071 ms  71.039 ms
12  72.22.188.102 (72.22.188.102)  71.585 ms  71.608 ms  71.493 ms
```

从 San Diego Super Computer Center 到 www.poly.edu 的三次 Traceroute

a. 在三个时间段内的平均往返时延分别是**71.18ms**，**71.38ms**和**71.55ms**，标准方差分别为**0.075ms**，**0.21ms**，**0.05ms**。

b. 在这个例子中，在三个时段各自路由器数量均为 12 个，在这些时段中路径未发生变化。

c. Traceroute 分组在源 IP 地址和目的 IP 地址之间经过了 4 个 ISP 网络。是的，在相邻的 ISP 网络间对等接口处出现了最大的时延。

```
traceroute to www.poly.edu (128.238.24.40), 30 hops max, 60 byte packets
 1  62-193-36-1.stella-net.net (62.193.36.1)  0.500 ms  0.415 ms  0.440 ms
 2  62.193.33.29 (62.193.33.29)  0.910 ms  1.065 ms  1.026 ms
 3  bg1.stella-net.net (62.193.32.254)  0.972 ms  1.026 ms  1.078 ms
 4  62.193.32.66 (62.193.32.66)  1.021 ms  0.988 ms  0.947 ms
 5  10gigabitethernet-2-2.par2.he.net (195.42.144.104)  1.537 ms  1.752 ms  1.714 ms
 6  10gigabitethernet7-1.core1.ash1.he.net (184.105.213.93)  80.273 ms  80.103 ms  79.971 ms
 7  10gigabitethernet1-2.core1.nyc4.he.net (72.52.92.85)  86.494 ms  85.872 ms  86.223 ms
 8  10gigabitethernet3-4.core1.nyc5.he.net (184.105.213.218)  85.248 ms  85.424 ms  85.388 ms
 9  lightower-fiber-networks.10gigabitethernet3-2.core1.nyc5.he.net (216.66.50.106)  86.194 ms  85.864 ms  86.116 ms
10  ae0.nycmnyzrj91.lightower.net (72.22.160.156)  85.796 ms  85.823 ms  85.766 ms
11  72.22.188.102 (72.22.188.102)  87.717 ms  86.817 ms  86.774 ms
```

```
traceroute to www.poly.edu (128.238.24.40), 30 hops max, 60 byte packets
 1  62-193-36-1.stella-net.net (62.193.36.1)  0.375 ms  0.397 ms  0.355 ms
 2  62.193.33.29 (62.193.33.29)  0.810 ms  0.877 ms  0.836 ms
 3  bg1.stella-net.net (62.193.32.254)  1.098 ms  0.991 ms  1.055 ms
 4  62.193.32.66 (62.193.32.66)  0.994 ms  0.960 ms  1.157 ms
 5  10gigabitethernet-2-2.par2.he.net (195.42.144.104)  1.679 ms  1.816 ms  1.768 ms
 6  10gigabitethernet7-1.core1.ash1.he.net (184.105.213.93)  80.416 ms  90.573 ms  90.659 ms
 7  10gigabitethernet1-2.core1.nyc4.he.net (72.52.92.85)  85.933 ms  95.987 ms  96.087 ms
 8  10gigabitethernet3-4.core1.nyc5.he.net (184.105.213.218)  90.268 ms  90.229 ms  90.030 ms
 9  lightower-fiber-networks.10gigabitethernet3-2.core1.nyc5.he.net (216.66.50.106)  85.833 ms  85.448 ms  85.418 ms
10  ae0.nycmnyzrj91.lightower.net (72.22.160.156)  87.067 ms  86.025 ms  85.962 ms
11  72.22.188.102 (72.22.188.102)  86.542 ms  86.369 ms  86.170 ms
```

```
traceroute to 128.238.24.40 (128.238.24.40), 30 hops max, 60 byte packets
 1  62-193-36-1.stella-net.net (62.193.36.1)  0.396 ms  0.284 ms  0.239 ms
 2  62.193.33.29 (62.193.33.29)  0.817 ms  0.786 ms  0.848 ms
 3  bg1.stella-net.net (62.193.32.254)  1.150 ms  1.216 ms  1.265 ms
 4  62.193.32.66 (62.193.32.66)  1.002 ms  0.963 ms  0.923 ms
 5  10gigabitethernet-2-2.par2.he.net (195.42.144.104)  1.573 ms  1.534 ms  1.643 ms
 6  10gigabitethernet7-1.core1.ash1.he.net (184.105.213.93)  88.738 ms  82.866 ms  82.783 ms
 7  10gigabitethernet1-2.core1.nyc4.he.net (72.52.92.85)  94.888 ms  90.936 ms  90.877 ms
 8  10gigabitethernet3-4.core1.nyc5.he.net (184.105.213.218)  90.498 ms  90.543 ms  90.482 ms
 9  lightower-fiber-networks.10gigabitethernet3-2.core1.nyc5.he.net (216.66.50.106)  85.716 ms  85.408 ms  85.637 ms
10  ae0.nycmnyzrj91.lightower.net (72.22.160.156)  85.779 ms  85.290 ms  85.252 ms
11  72.22.188.102 (72.22.188.102)  86.217 ms  86.652 ms  86.588 ms
```

从 www.stella-net.net (France) 到 www.poly.edu (USA)的 Traceroutes

在三个时间段内的平均往返时延分别是**87.09ms**，**86.35ms**和**86.48ms**，标准方差分别为**0.53ms**，**0.18ms**，**0.23ms**。在这个例子中，在三个时段各自路由器数量均为 11 个，在这些时段中路径未发生变化。Traceroute 分组在源 IP 地址和目的 IP 地址之间经过了 3 个 ISP 网络。是的，在相邻的 ISP 网络间对等接口处出现了最大的时延。

P19. (a) Visit the site www.traceroute.org and perform traceroutes from two different cities in France to the same destination host in the United States. How many links are the same in the two traceroutes? Is the transatlantic link the same?

(b) Repeat (a) but this time choose one city in France and another city in Germany.

(c) Pick a city in the United States, and perform traceroutes to two hosts, each in a different city in China. How many links are common in the two traceroutes? Do the two traceroutes diverge before reaching China?

实验结果如下：

```
traceroute to www.poly.edu (128.238.24.30), 30 hops max, 60 byte packets
 1  62-193-36-1.stella-net.net (62.193.36.1)  0.426 ms  0.329 ms  0.284 ms
 2  62.193.33.25 (62.193.33.25)  0.810 ms  0.771 ms  0.878 ms
 3  62.193.32.66 (62.193.32.66)  0.815 ms  0.840 ms  0.801 ms
 4  10gigabitethernet-2-2.par2.he.net (195.42.144.104)  1.387 ms  1.506 ms  1.467 ms
 5  10gigabitethernet7-1.core1.ash1.he.net (184.105.213.93)  85.402 ms  85.553 ms  85.353 ms
 6  10gigabitethernet1-2.core1.nyc4.he.net (72.52.92.85)  94.360 ms  96.220 ms  96.355 ms
 7  10gigabitethernet3-4.core1.nyc5.he.net (184.105.213.218)  90.279 ms  87.459 ms  87.709 ms
 8  lightower-fiber-networks.10gigabitethernet3-2.core1.nyc5.he.net (216.66.50.106)  85.474 ms  85.450 ms  85.983 ms
 9  ae0.nycmnyzrj91.lightower.net (72.22.160.156)  86.160 ms  85.768 ms  86.016 ms
10  72.22.188.102 (72.22.188.102)  124.111 ms  89.340 ms  89.556 ms
```

```
 1  v1200.hs01.mar01.jaguar-network.net (85.31.192.253)  0.552 ms  0.414 ms
 2  ae1.cr01.mar01.jaguar-network.net (85.31.194.9)  0.340 ms  0.213 ms
 3  xe2-0-0.cr01.par02.jaguar-network.net (78.153.231.201)  9.933 ms  9.841 ms
 4  te1-3.er01.par02.jaguar-network.net (85.31.194.14)  9.828 ms  9.962 ms
 5  10gigabitethernet-2-2.par2.he.net (195.42.144.104)  10.456 ms  10.332 ms
 6  10gigabitethernet7-1.core1.ash1.he.net (184.105.213.93)  88.793 ms  96.781 ms
 7  10gigabitethernet1-2.core1.nyc4.he.net (72.52.92.85)  94.651 ms  99.654 ms
 8  10gigabitethernet3-4.core1.nyc5.he.net (184.105.213.218)  94.786 ms  94.755 ms
 9  lightower-fiber-networks.10gigabitethernet3-2.core1.nyc5.he.net (216.66.50.106)  91.935 ms  91.776 ms
10  ae0.nycmnyzrj91.lightower.net (72.22.160.156)  91.909 ms  91.784 ms
11  72.22.188.102 (72.22.188.102)  93.791 ms  93.515 ms
```

从法国一个城市到美国纽约市的 Traceroute

a. 在这些从法国一个城市到美国纽约市的 Traceroute 的实验中包括大洋彼岸的链路在内一共有 7 条链路。

traceroute to www.poly.edu (128.238.24.30), 30 hops max, 60 byte packets

1				*	*	*
2	hos-tr3.juniper2.rz10.hetzner.de	213.239.224.65	de	**0.224** ms		
	hos-tr2.juniper1.rz10.hetzner.de	213.239.224.33	de	**0.174** ms	**0.176** ms	
3	hos-bb1.juniper1.ffm.hetzner.de	213.239.240.224	de	**4.746** ms	**4.780** ms	
	hos-bb1.juniper4.ffm.hetzner.de	213.239.240.230	de	**4.823** ms		
4	20gigabitethernet4-3.core1.fra1.he.net	80.81.192.172	de	**5.462** ms	**5.461** ms	**5.456** ms
5	10gigabitethernet1-4.core1.ams1.he.net	72.52.92.94	us	**12.899** ms		
	10gigabitethernet5-3.core1.ams1.he.net	72.52.92.77	us	**13.197** ms		
	10gigabitethernet5-3.core1.lon1.he.net	184.105.213.145	us	**26.110** ms		
6	10gigabitethernet1-4.core1.lon1.he.net	72.52.92.81	us	**18.720** ms	**18.871** ms	**18.862** ms
7	10gigabitethernet7-4.core1.nyc4.he.net	72.52.92.241	us	**86.677** ms	**85.580** ms	**86.560** ms
8	lightower-fiber-networks.10gigabitethernet3-2.core1.nyc5.he.net	216.66.50.106	us	**118.500** ms		
	10gigabitethernet3-4.core1.nyc5.he.net	184.105.213.218	us	**90.346** ms		
	lightower-fiber-networks.10gigabitethernet3-2.core1.nyc5.he.net	216.66.50.106	us	**118.500** ms		
9	ae0.nycmnyzrj91.lightower.net	72.22.160.156	us	**85.289** ms	**85.552** ms	**85.283** ms


```
traceroute to www.poly.edu (128.238.24.30), 30 hops max, 60 byte packets
 1  62-193-36-1.stella-net.net (62.193.36.1)  0.426 ms  0.329 ms  0.284 ms
 2  62.193.33.25 (62.193.33.25)  0.810 ms  0.771 ms  0.878 ms
 3  62.193.32.66 (62.193.32.66)  0.815 ms  0.840 ms  0.801 ms
 4  10gigabitethernet-2-2.par2.he.net (195.42.144.104)  1.387 ms  1.506 ms  1.467 ms
 5  10gigabitethernet7-1.core1.ash1.he.net (184.105.213.93)  85.402 ms  85.553 ms  85.353 ms
 6  10gigabitethernet1-2.core1.nyc4.he.net (72.52.92.85)  94.360 ms  96.220 ms  96.355 ms
 7  10gigabitethernet3-4.core1.nyc5.he.net (184.105.213.218)  90.279 ms  87.459 ms  87.709 ms
 8  lightower-fiber-networks.10gigabitethernet3-2.core1.nyc5.he.net (216.66.50.106)  85.474 ms  85.450 ms  85.983 ms
 9  ae0.nycmnyzrj91.lightower.net (72.22.160.156)  86.160 ms  85.768 ms  86.016 ms
10  72.22.188.102 (72.22.188.102)  124.111 ms  89.340 ms  89.556 ms
```


从法国一个城市到德国一个城市的 Traceroute

b. 在这些从法国一个城市到德国一个城市的 Traceroute 的实验中一共有 7 条链路，没有大

洋彼岸的链路。

```
Tracing route to www.autoisp.shu.edu.cn [27.115.83.251]
over a maximum of 30 hops:

  1     9 ms     8 ms    10 ms  10.40.32.1
  2    12 ms    12 ms     9 ms  gig-3-0-4-nycmnyj-rtr1.nyc.rr.com [24.29.119.189]
  3    21 ms    20 ms    22 ms  tenge-0-6-0-0-nyquny91-rtr001.nyc.rr.com [24.29.100.122]
  4    19 ms    21 ms    22 ms  bun6-nyquny91-rtr002.nyc.rr.com [24.29.148.254]
  5    11 ms    11 ms    19 ms  ae-3-0.cr0.nyc20.tbone.rr.com [66.109.6.76]
  6    14 ms    18 ms    14 ms  ae-0-0.pr0.nyc30.tbone.rr.com [66.109.6.159]
  7    14 ms    11 ms    10 ms  xe-9-0-0.edge2.Newark1.Level3.net [4.59.20.29]
  8    12 ms    10 ms    13 ms  ae-31-51.ebr1.Newark1.Level3.net [4.69.156.30]
  9    10 ms    15 ms    13 ms  ae-2-2.ebr1.NewYork1.Level3.net [4.69.132.97]
 10    11 ms    17 ms    14 ms  ae-81-81.csw3.NewYork1.Level3.net [4.69.134.74]
 11    12 ms    14 ms    11 ms  ae-82-82.ebr2.NewYork1.Level3.net [4.69.148.41]
 12    83 ms    83 ms    88 ms  ae-2-2.ebr4.SanJose1.Level3.net [4.69.135.185]
 13    91 ms    87 ms    84 ms  ae-71-71.csw2.SanJose1.Level3.net [4.69.153.6]
 14    83 ms    83 ms    88 ms  ae-2-70.edge3.SanJose1.Level3.net [4.69.152.82]
 15   595 ms   593 ms   600 ms  CHINA-NETCO.edge3.SanJose1.Level3.net [4.79.54.6]
 16   594 ms   591 ms   592 ms  219.158.96.213
 17   539 ms   540 ms   540 ms  219.158.11.173
 18   593 ms   586 ms   585 ms  219.158.19.93
 19   585 ms   585 ms   584 ms  219.158.21.246
 20   568 ms   587 ms   569 ms  112.64.243.62
 21   570 ms   566 ms   568 ms  112.64.243.146
 22   342 ms   341 ms   347 ms  112.65.183.106
 23   574 ms   571 ms   573 ms  27.115.83.251

Trace complete.
```

```
Tracing route to www.lb.pku.edu.cn [162.105.131.113]
over a maximum of 30 hops:

  1     8 ms     8 ms     8 ms  10.40.32.1
  2    14 ms     9 ms    10 ms  gig-0-3-0-18-nycmnyj-rtr1.nyc.rr.com [24.168.138.85]
  3    21 ms    10 ms    11 ms  tenge-0-6-0-0-nyquny91-rtr001.nyc.rr.com [24.29.100.122]
  4    13 ms    22 ms    22 ms  bun6-nyquny91-rtr002.nyc.rr.com [24.29.148.254]
  5    11 ms    18 ms    12 ms  ae-3-0.cr0.nyc20.tbone.rr.com [66.109.6.76]
  6    43 ms    38 ms    41 ms  ae-8-0.cr0.chi10.tbone.rr.com [66.109.6.25]
  7    86 ms    88 ms    88 ms  ae-6-0.cr0.sjc30.tbone.rr.com [66.109.6.14]
  8    86 ms    89 ms    91 ms  ae-1-0.pr0.sjc10.tbone.rr.com [66.109.6.137]
  9    87 ms    86 ms    86 ms  66.109.10.210
 10   257 ms   258 ms   258 ms  ge3-0-0.gw4.hkg3.asianetcom.net [61.14.157.250]
 11   298 ms   296 ms   295 ms  CER-0002.gw4.hkg3.asianetcom.net [203.192.137.198]
 12   297 ms   305 ms   305 ms  202.112.61.13
 13   295 ms   296 ms   296 ms  202.112.61.157
 14     *        *        *     Request timed out.
 15   298 ms   302 ms   298 ms  202.112.41.178
 16   308 ms   300 ms   300 ms  202.112.41.182
```

美国的一台主机对中国两个不同城市的两台主机进行 Traceroute

c.在两个 traceroute 试验中，均是 5 条链路。这两个 traceroute 在抵达中国之前已分开。

P20. Consider the throughput example corresponding to Figure 1.20(b). Now suppose that there are *M* client-server pairs rather than 10. Denote R_s, R_c, and R for the rates of the server links, client links, and network link. Assume all other links have abundant capacity and that there is no other traffic in the network besides the traffic generated by the *M* client-server pairs. Derive a general expression for throughput in terms of R_s, R_c, R, and M.

答：吞吐量 Throughput = $min\{R_s, R_c, R/M\}$。

P21. Consider Figure 1.19(b). Now suppose that there are *M* paths between the server and the client. No two paths share any link. Path k ($k = 1, \ldots, M$) consists of N links with transmission rates R_{k1}, R_{2k}, . . ., R_{Nk}. If the server can only use one path to send data to the client, what is the maximum throughput that the server can achieve? If the server can use all *M* paths to send data, what is the maximum throughput that the server can achieve?

答：如果只能用一条路径，最大吞吐率为：

$$\max\left(min(R_1^1,\ R_2^1,\ \cdots,\ R_N^1),\ min(R_1^2,\ R_2^2,\ \cdots,\ R_N^2),\ \cdots,\ min(R_1^M,\ R_2^M,\ \cdots,\ R_N^M)\right),$$

如果使用所有路径，最大吞吐率为：

$$\sum_{k=1}^{M} min\{R_1^k,\ R_2^k, \cdots,\ R_N^k\}$$

P22. Consider Figure 1.19(b). Suppose that each link between the server and the client has a packet loss probability *p,* and the packet loss probabilities for these links are independent. What is the probability that a packet (sent by the server) is successfully received by the receiver? If a packet is lost in the path from the server to the client, then the server will re-transmit the packet. On average, how many times will the server re-transmit the packet in order for the client to successfully receive the packet?

答：成功接收到一个分组的概率为：$P_s = (1-p)^N$；

传输在一直进行中，直到客户机成功接收到分组为止，传输的数量是几何随机变量，概率为 P_s，因此，平均需要传输的数量为：$1/P_s$；然后，平均需要重传的数量为$1/P_s - 1$。

P23. Consider Figure 1.19(a). Assume that we know the bottleneck link along the path from the server to the client is the first link with rate R_s bits/sec. Suppose we send a pair of packets back to back from the server to the client, and there is no other traffic on this path. Assume each packet of size L bits, and both links have the same propagation delay d_{prop}.

a. What is the packet inter-arrival time at the destination? That is, how much time elapses from when the last bit of the first packet arrives until the last bit of the second packet arrives?

b. Now assume that the second link is the bottleneck link (i.e., $R_c < R_s$). Is it possible that the second packet queues at the input queue of the second link? Explain. Now suppose that the server sends the second packet T seconds after sending the first packet. How large must T be to ensure no queuing before the second link? Explain.

答：我们把第一个分组称为 A，第二个为 B。

a. 瓶颈链路是第一条链路，所以分组 B 在第一条链路上等待分组 A 的传输，分组在目的地的到达间隔为：L/R_s。

b. 如果第二条链路是瓶颈链路，两个分组是连续传输的，必须保证：在第一个分组在路由器传输进入第二条链路之前，第二个分组到达第二条链路的的输入队列，需要满足以下条件：

$$\frac{\mathrm{L}}{R_s} + \frac{\mathrm{L}}{R_s} + d_{prop} < \frac{\mathrm{L}}{R_s} + d_{prop} + \frac{\mathrm{L}}{R_c}$$

其中左边表示第二个分组到达传第二条链路的传输队列时间，右边表示第一个分组传输到第二条链路的时间。

如果第二个分组在第一个分组 T 时间间隔之后传输，我们需要确保第二个分组在第二条链路没有排队时延，需要保证：

$$\frac{\mathrm{L}}{R_s} + \frac{\mathrm{L}}{R_s} + d_{prop} + \mathrm{T} \geq \frac{\mathrm{L}}{R_s} + d_{prop} + \frac{\mathrm{L}}{R_c}$$

因此，T 的最小值为：

$$\frac{L}{R_c} - \frac{L}{R_s}$$

P24. Suppose you would like to urgently deliver 40 terabytes data from Boston to Los Angeles. You have available a 100 Mbps dedicated link for data transfer. Would you prefer to transmit the data via this link or instead use FedEx overnight delivery? Explain.

答：如果使用 100 Mbps 专用链路(dedicated link)传输，需要花费的时间为：

$$\frac{40TB}{100Mbps}=\frac{40\times10^{12}\times8bits}{100\times10^{6}bits/s}=3.2\times10^{6}s=37\text{days}$$

如果使用联邦快递(FedEx overnight delivery)，只需要一天，并且少花费 100 美元，综上可知，紧急传输大容量数据快递公司较好。

P25. Suppose two hosts, A and B, are separated by 20,000 kilometers and are connected by a direct link of R = 2 Mbps. Suppose the propagation speed over the link is $2.5 \cdot 10^8$ meters/sec.

a. Calculate the bandwidth-delay product, $R \cdot d_{prop}$.

b. Consider sending a file of 800,000 bits from Host A to Host B. Suppose the file is sent continuously as one large message. What is the maximum number of bits that will be in the link at any given time?

c. Provide an interpretation of the bandwidth-delay product.

d. What is the width (in meters) of a bit in the link? Is it longer than a football field?

e. Derive a general expression for the width of a bit in terms of the propagation speed *s,* the transmission rate *R,* and the length of the link *m*.

答：a. 由题意，传播时延为：$d_{prop}=\frac{20000\times10^{3}}{2\times10^{6}m/s}=0.08s$，则宽带时延积(bandwidth-delay product)为：

$$\mathrm{R}\cdot d_{prop}=0.08s\times2\times10^{6}bps=160000bits$$

b. 在链路上的最大比特数量为 160000，小于要传输的 800000bits。

c. 宽带时延积指的是在任意时刻，链路上能具有的最大比特数量。

d. 一个比特的宽度即传播距离 m 与宽带时延积的商，为：

$$\frac{20000\times10^{3}\mathrm{m}}{160000bit}=125m/bit$$

e. 比特宽度为：

$$\frac{\mathrm{m}}{R\cdot d_{prop}}=\frac{m}{R\cdot\frac{m}{s}}=\frac{s}{R}$$

P26. Referring to problem P25, suppose we can modify R. For what value of R is the width of a bit as long as the length of the link?

答：由题，比特宽度大小等于链路传播速率大小，即 s/R=20000km，则：

$$\mathrm{R}=\frac{s}{20000km}=\frac{2.5\times10^{8}}{2\times10^{7}}=12.5bps$$

P27. Consider problem P25 but now with a link of R = 1 Gbps.

a. Calculate the bandwidth-delay product, $R \cdot d_{prop}$.

b. Consider sending a file of 800,000 bits from Host A to Host B. Suppose the file is sent continuously as one big message. What is the maximum number of bits that will be in the link at any given time?

c. What is the width (in meters) of a bit in the link?

答：a. 由 25 题可知，传播时延为 0.08 秒，则宽带时延积为：

$$R \cdot d_{prop} = 0.08s \times 1 \times 10^9 bps = 8 \times 10^7 bits$$

b. 由于宽带时延积远远大于发送文件的大小，所以在任意给定的时间，链路能具有最大比特数为 80000bits。

c. 比特宽度为传播距离与宽带时延积的商，即：

$$\frac{20000\text{km}}{8 \times 10^7 bit} = 0.25m/b$$

P28. Refer again to problem P25.

a. How long does it take to send the file, assuming it is sent continuously?

b. Suppose now the file is broken up into 20 packets with each packet containing 40,000 bits. Suppose that each packet is acknowledged by the receiver and the transmission time of an acknowledgment packet is negligible. Finally, assume that the sender cannot send a packet until the preceding one is acknowledged. How long does it take to send the file?

c. Compare the results from (a) and (b).

答：a. 由题意总时延为：

$$d_{trans} + \mathrm{d}_{proc} = \frac{800\ 000 bits}{2 \times 10^6 bps} + \frac{20000 \times 10^3 m}{2.5 \times 10^8 m/s} = 0.4 + 0.08 = 0.48s$$

b. 由题意，原来的 800000bits 大文件分成了 20 个小分组来传输，并且一个分组在发送方收到确认之前不会发送下一个分组，且忽略 receiver 接收时间和确认分组的传输时间，每一个分组包括三部分，在 sender 的传输时间，链路的传播时间，确认 ACK 的传播时间，一共需要传输 20 次，时间为：

$$20(\mathrm{d}_{trans} + 2 \times d_{proc}) = 20 \times \left(\frac{40000b}{2 \times 10^6 bps} + 2 \times \frac{2 \times 10^4 \times 10^3 m}{2.5 \times 10^8 m/s}\right) = 3.6s$$

c. 将一个文件分解成多个小的分组发送将花费更多时间，因为每一个分组都要传播时间以及确认 ACK 的传播时间，多出很多时间开销。

P29. Suppose there is a 10 Mbps microwave link between a geostationary satellite and its base station on Earth. Every minute the satellite takes a digital photo and sends it to the base station. Assume a propagation speed of 2.4 · 108 meters/sec.

a. What is the propagation delay of the link?

b. What is the bandwidth-delay product, $R \cdot d_{prop}$?

c. Let x denote the size of the photo. What is the minimum value of x for the microwave link to be continuously transmitting?

答：a. 地球同步卫星与地球的距离为 36000km，传播时延为：

$$\mathrm{d}_{prop} = \frac{36000 \times 10^3 m}{2.4 \times 10^8 m/s} = 0.15s = 150ms$$

b. 宽带时延积为：

$$R \cdot d_{prop} = (10 \times 10^6 bps) \times 0.15\mathrm{s} = 1.5 \times 10^6 bits$$

c. 由题意，每分钟传输的最小的数据大小为：

$$60\mathrm{R} = 60\mathrm{s} \times 10 \times 10^6 bps = 6 \times 10^8 bits$$

P30. Consider the airline travel analogy in our discussion of layering in Section 1.5, and the addition of headers to protocol data units as they flow down the protocol stack. Is there an equivalent notion of header information that is added to passengers and baggage as they move down the airline protocol stack?

答：有与首部信息等价的概念。比如旅客和行李对应到达协议栈顶部的数据单元，当旅客检票的时候，他的行李也被检查了，行李和机票被加上标记，当旅客稍后通过安检，通常会另外添加一个标记，指明该旅客已经通过了安检，之后的行程中，每到一个节点时，票上或行李上都会做相应的标记以保证旅客和行李的正确运输。正如协议的首部一样，协议的数据单元向协议的底层流动时，都要增加相应的首部信息使其能够保证数据单元下一步的正确进行。

P31. In modern packet-switched networks, including the Internet, the source host segments long, application-layer messages (for example, an image or a music file) into smaller packets and sends the packets into the network. The receiver then reassembles the packets back into the original message. We refer to this process as *message segmentation*. Figure 1.27 illustrates the end-to-end transport of a message with and without message segmentation. Consider a message that is $8 \cdot 10^6$ bits long that is to be sent from source to destination in Figure 1.27. Suppose each link in the figure is 2 Mbps. Ignore propagation, queuing, and processing delays.

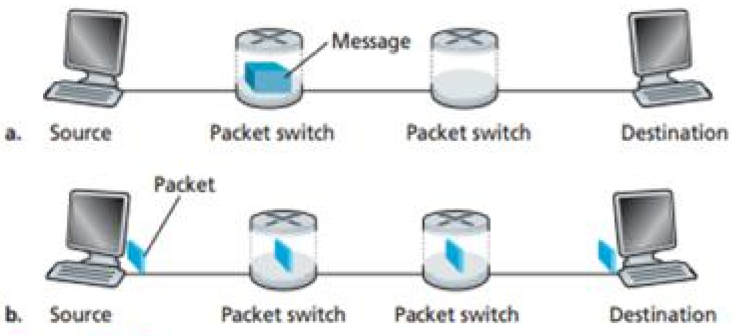


Figure 1.27 ♦ End-to-end message transport: (a) without message segmentation; (b) with message segmentation

a. Consider sending the message from source to destination *without* message segmentation. How long does it take to move the message from the source host to the first packet switch? Keeping in mind that each switch uses store-and-forward packet switching, what is the total time to move themessage from source host to destination host?

b. Now suppose that the message is segmented into 800 packets, with each packet being 10,000 bits long. How long does it take to move the first packet from source host to the first switch? When the first packet is being sent from the first switch to the second switch, the second packet is being sent from the source host to the first switch. At what time will the second packet be fully received at the first switch?

c. How long does it take to move the file from source host to destination host when message segmentation is used? Compare this result with your answer in part (a) and comment.

d. In addition to reducing delay, what are reasons to use message segmentation?

e. Discuss the drawbacks of message segmentation.

答：a. 由题意，忽略传播时延、排队时延和处理时延，在没有分组时，从源主机到第一个分组交换机的时间即传输时间为：

$$\frac{8\times10^6b}{2\times10^6bps}=4s$$

一共要传输 3 次到链路，从源主机到目的主机总时延为：

$$4\times3=12s$$

b. 报文段一共分成 800 个组，每个组大小为 10000bits，从源主机到第一个分组交换机，第一个分组的时延为：

$$\frac{10000b}{2\times10^6bps}=0.005s=5ms$$

第一个分组发送到第二台交换机，时延为 5×2=10ms；第二个分组从源主机发送到第一台交换机时延为 5ms；第二个分组被第一个交换机收到时间为 5×2=10ms。

c. 由题意，分组没有等待前一个分组 ACK 就连续发送，时间间隔为 5ms，当第一个分组到达目的地后，经过 5ms 第二个分组就到达，以此类推得到分组发送总时延为：

$$5\times3+799\times5=4010\text{ms}=4.01\text{s}$$

d. 如果没有报文分段，则不允许比特错误，如果有一个比特发生错误要重传整个报文段；如果没有报文分段，巨大的分组(如 HD videos)被发送到网络中，路由器必须容纳这些巨大的分组，小的分组必须排队在这些巨大的分组后面，忍受不公平的时延。

e. 分组在目的地必须按序排列；报文分组在许多小的分组中起作用，对于所有的分组而言，不管其大小，头部长度一样，报文分组以后在首部会有很大的开销。

P32. Experiment with the Message Segmentation applet at the book's Web site. Do the delays in the applet correspond to the delays in the previous problem? How do link propagation delays affect the overall end-to-end delay for packet switching (with message segmentation) and for message switching?

答：是的，在 applet 实验中的时延和 31 题的时延一致，传播时延影响整个端到端的时延，对交换机的分组交换和报文段分组影响一样。

P33. Consider sending a large file of F bits from Host A to Host B. There are three links (and two switches) between A and B, and the links are uncongested (that is, no queuing delays). Host A segments the file into segments of S bits each and adds 80 bits of header to each segment, forming packets of L = 80 + S bits. Each link has a transmission rate of R bps. Find the value of S that minimizes the delay of moving the file from Host A to Host B. Disregard propagation delay.

答：由题意，忽略排队时延，处理时延和传播时延，只需要考虑传输时延。分组个数为F/S，每个分组报文段大小为$(\text{S}+80)$bits，分组传输进入一条链路时间为$(\text{S}+80)/R$，第一个分组报文段传输到主机 B 时延为$3\times(\text{S}+80)/R$，以后每经过$(\text{S}+80)/R$时延一个分组报文段到达目的主机 B，则总时延为：

$$d_{delay}=(\text{S}+80)\times3+(S+80)\times\left(\frac{F}{S}-1\right)=(S+80)\times\left(\frac{F}{S}+2\right)$$

要找到合适的 S 使时延最小，对上式进行求导得：

$$\frac{\text{d}}{dS}d_{delay}=0$$

解得$\text{S}=\sqrt{40F}$。

P34. Skype offers a service that allows you to make a phone call from a PC to an ordinary phone. This means that the voice call must pass through both the Internet and through a telephone network. Discuss how this might be done.

答：电路交换电话网路和互联网两个网络在“网关”处连接在一起。当一个“Skype”用户(连接到互联网的)打一个电话给普通用户时，通过电路交换网络，在电话用户和网关之间建立了电路。由多个分组构成的“Skype”的语音通过互联网发送到网关。在网关那里，语音信号被重整处理，然后发送到电话交换网络。另一方面，语音信号通过电路交换网络发送到网关，网关对语音信号进行分组化处理，再发送给“Skype”用户。

Chapter 2 Application Layer

P1. True or false?

a. A user requests a Web page that consists of some text and three images. For this page, the client will send one request message and receive four response messages.

b. Two distinct Web pages (for example, www.mit.edu/research.html and www.mit.edu/students.html) can be sent over the same per- sistent connection.

c. With nonpersistent connections between browser and origin server, it is possible for a single TCP segment to carry two distinct HTTP request messages.

d. The Date: header in the HTTP response message indicates when the object in the response was last modified.

e. HTTP response messages never have an empty message body.

答：a. F，文本和三张图片各需要一个请求和响应报文，所以一共有四个请求报文和四个响应报文；

b. T，由题意，这两个 web 页面主机名相同均为：www.mit.edu，只是路径名不同。

c. F，一方面，不同的 HTTP 请求报文经过封装后形成的 TCP 报文段不同，所以一个 TCP 报文段不可能携带不同的 HTTP 请求报文；另一方面是非持久连接，有可能在一个请求响应报文回合结束就断开了连接，所以不正确。

d. F，上次修改时间在 HTTP 首部字段，即："Last-Modified"，而非在数据字段。

e. F，响应报文可能实体为空，比如"404 Not Found"，表示请求的对象在服务器上没有找到。

P2. Read RFC 959 for FTP. List all of the client commands that are supported by the RFC.

答：访问控制命令(Access control commands)：USER, PASS, ACT, CWD, CDUP, SMNT, REIN, QUIT；

传输参数命令(Transfer parameter commands)：PORT, PASV, TYPE STRU, MODE；

FTP 服务命令(Service commands)：RETR, STOR, STOU, APPE, ALLO, REST, RNFR, RNTO, ABOR, DELE, RMD, MRD, PWD, LIST, NLST, SITE, SYST, STAT, HELP, NOOP。

P3. Consider an HTTP client that wants to retrieve a Web document at a given URL. The IP address of the HTTP server is initially unknown. What transport and application-layer protocols besides HTTP are needed in this scenario?

答：应用层协议：HTTP 和 DNS，运输层协议：TCP 用于封装 HTTP，UDP 用于封装 DNS。

P4. Consider the following string of ASCII characters that were captured by Wireshark when the browser sent an HTTP GET message (i.e., this is the actual content of an HTTP GET message). The characters <cr><lf> are carriage return and line-feed characters (that is, the italized character string <cr> in the text below represents the single carriage-return character that was con- tained at that point in the HTTP header). Answer the following questions, indicating where in the HTTP GET message below you find the answer.

GET /cs453/index.html HTTP/1.1*<cr><lf>*Host: gaia.cs.umass.edu*<cr><lf>*User-Agent: Mozilla/5.0 (Windows;U; Windows NT 5.1; en-US; rv:1.7.2) Gecko/20040804 Netscape/7.2 (ax)

*<cr><lf>*Accept:ext/xml, application/xml, application/xhtml+xml, text/html; q=0.9, text/plain; q=0.8, image/png, */*; q=0.5*<cr><lf>*Accept-Language: enus, en; q=0.5
*<cr><lf>*AcceptEncoding: zip,deflate*<cr><lf>*Accept-Charset: ISO-8859-1, utf-8; q=0.7, *; q=0.7*<cr><lf>*Keep-Alive: 300*<cr><lf>*Connection:keep-alive*<cr><lf><cr><lf>*

a. What is the URL of the document requested by the browser?
b. What version of HTTP is the browser running?
c. Does the browser request a non-persistent or a persistent connection?
d. What is the IP address of the host on which the browser is running?
e. What type of browser initiates this message? Why is the browser type needed in an HTTP request message?

答：a. 请求的 URL 是：http://gaia.cs.umass.edu/cs453/index.html，其中 gaia.cs.umass.edu 是主机名，/cs453/index.html 是主机上的路径名。
b. HTTP/1.1。
c. 由字段“Connection:keep-alive”可以得出是持久连接。
d. 在 HTTP 首部没有 IP 信息。
e. Mozilla/5.0。浏览器类型信息是必须的，因为对于服务器而言，它要将同一对象的不同版本发送给不同类型的浏览器。

P5. The text below shows the reply sent from the server in response to the HTTP GET message in the question above. Answer the following questions, indicat- ing where in the message below you find the answer.

HTTP/1.1 200 OK<cr><lf>Date: Tue, 07 Mar 2008 12:39:45GMT<cr><lf>Server: Apache/2.0.52 (Fedora)
<cr><lf>Last-Modified: Sat, 10 Dec2005 18:27:46 GMT<cr><lf>ETag: “526c3-f22-a88a4c80”<cr><lf>Accept- Ranges: bytes<cr><lf>Content-Length: 3874<cr><lf> Keep-Alive: timeout=max=100<cr><lf>Connection: Keep-Alive<cr><lf>Content-Type: text/html; charset= ISO-8859-1<cr><lf><cr><lf><!doctype html public “- //w3c//dtd html 4.0 transitional//en”><lf><html><lf>
<head><lf> <meta http-equiv=”Content-Type” content=”text/html; charset=iso-8859-1”><lf> <meta name=”GENERATOR” content=”Mozilla/4.79 [en] (Windows NT 5.0; U) Netscape]”><lf> <title>CMPSCI 453 / 591 /
NTU-ST550A Spring 2005 homepage</title><lf></head><lf>
<much more document text following here (not shown)>

a. Was the server able to successfully find the document or not? What time was the document reply provided?
b. When was the document last modified?
c. How many bytes are there in the document being returned?
d. What are the first 5 bytes of the document being returned? Did the server agree to a persistent connection?

答：a. 由“HTTP/1.1 200 OK”状态码 200 可以看出，服务器成功找到文档；回应文档时间为：“Tue, 07 Mar 2008 12:39:45GMT”。

b. 由“Last-Modified: Sat, 10 Dec2005 18:27:46 GMT”得到上次修改时间为：Sat, 10 Dec2005 18:27:46 GMT。

c. 由“Content-Length: 3874”得知，文档大小为 3874bytes。

d. “<cr><lf><cr><lf>”两个回车换行符是标志，之后是文档内容，前五个字节是：<!doc 。

e. 由“Connection: Keep-Alive”可知，是持久连接。

P6. Obtain the HTTP/1.1 specification (RFC 2616). Answer the following questions:

a. Explain the mechanism used for signaling between the client and server to indicate that a persistent connection is being closed. Can the client, the server, or both signal the close of a connection?

b. What encryption services are provided by HTTP?

c. Can a client open three or more simultaneous connections with a given server?

d. Either a server or a client may close a transport connection between them if either one detects the connection has been idle for some time. Is it possible that one side starts closing a connection while the other side is transmitting data via this connection? Explain.

答：a. 在 RFC 2616 第 8 部分讨论的是持久连接(这个问题的真正目的是要求你去获取并且阅读 RFC)，RFC 的 8.1.2 和 8.1.2.1 章节指出无论是客户机还是服务器都可以关闭持久连接。通过在在请求报文或者响应报文中，Connection 字段的值为：Close 来关闭连接。

b. HTTP 没有提供任何加密服务。

c. 不可以，最多两条。(From RFC 2616) “Clients that use persistent connections should limit the number of simultaneous connections that they maintain to a given server. A single-user client SHOULD NOT maintain more than 2 connections with any server or proxy.”

d. 在一端发送消息，而另一端却正在关闭连接这是有可能。(From RFC 2616) “A client might have started to send a new request at the same time that the server has decided to close the "idle" connection. From the server's point of view, the connection is being closed while it was idle, but from the client's point of view, a request is in progress.”

P7. Suppose within your Web browser you click on a link to obtain a Web page. The IP address for the associated URL is not cached in your local host, so a DNS lookup is necessary to obtain the IP address. Suppose that n DNS servers are visited before your host receives the IP address from DNS; the successive visits incur an RTT of RTT1, . . . , RTTn. Further suppose that the Web page associated with the link contains exactly one object, consisting of a small amount of HTML text. Let RTT0 denote the RTT between the local host and the server containing the object. Assuming zero transmission time of the object, how much time elapses from when the client clicks on the link until the client receives the object?

答：获得网页 IP 地址所花费的时间为：

$$RTT_1 + RTT_2 + RTT_3 + \cdots + RTT_n = \sum_{i=1}^{n} \mathrm{RTT}_i$$

一旦已知 IP 地址，一个RTT_0表示建立 TCP 连接时间，另一个RTT_0表示发送请求报文和响应报文的时间，综上总时间为：

$$2RTT_0 + \sum_{i=1}^{n} \mathrm{RTT}_i$$

P8. Referring to Problem P7, suppose the HTML file references eight very small objects on the same server. Neglecting transmission times, how much time elapses with

a. Non-persistent HTTP with no parallel TCP connections?

b. Non-persistent HTTP with the browser configured for 5 parallel connections?

c. Persistent HTTP?

答：获得网站 IP 并且进入主界面所需时间为：

$$\mathrm{RTT}_1 + \cdots + RTT_n + 2RTT_0.$$

a. 没有并行 TCP 连接的非持久 HTTP，在 TCP 连接建立以后，每一个对象都要建立 TCP 连接并进行传输，一共需要建立 8 次连接，所需要的时间为：

$$RTT_1 + \cdots + RTT_n + 2RTT_0 + 8 \times 2RTT_0 = RTT_1 + \cdots + RTT_n + 18RTT_0$$

b. 由 5 条并行 TCP 连接的非持久 HTTP，并行的意思即同时可以建立 5 条连接，所以只需要建立两次连接，第一次 5 条，第二次 3 条，所需时间为：

$$\mathrm{RTT}_1 + \cdots + RTT_n + 2RTT_0 + 2 \times 2RTT_0 = RTT_1 + \cdots + RTT_n + 6RTT_0$$

c. 有流水线的持久 RTT，在服务器和客户机建立连接以后，可以在此连接中请求并且接收所有 8 个对象，无需建立新的连接，所需时间为：

$$\mathrm{RTT}_1 + \cdots + RTT_n + 2RTT_0 + RTT_0 = RTT_1 + \cdots + RTT_n + 3RTT_0$$

P9. Consider Figure 2.12, for which there is an institutional network connected to the Internet. Suppose that the average object size is 850,000 bits and that the average request rate from the institution's browsers to the origin servers is 16 requests per second. Also suppose that the amount of time it takes from when the router on the Internet side of the access link forwards an HTTP request until it receives the response is three seconds on average (see Section 2.2.5). Model the total average response time as the sum of the average access delay (that is, the delay from Internet router to institution router) and the average Internet delay. For the average access delay, use $\Delta/(1-\Delta\beta)$, where Δ is the average time required to send an object over the access link and β is the arrival rate of objects to the access link.

a. Find the total average response time.

b. Now suppose a cache is installed in the institutional LAN. Suppose the miss rate is 0.4. Find the total response time.

答：a. 跨越访问链路发送一个对象，即通过因特网访问一个对象所需要的时间为：

$$\Delta = \frac{850000bits}{15Mbps} = \frac{850000bits}{15 \times 10^6 bps} \approx 0.0567s$$

链路的流量强度即对象对该访问链路的平均到达率为：

$$\beta = (16\mathrm{request/sec}) \times (0.0567 sec/request) \approx 0.9$$

平均访问时延为：

$$\frac{\Delta}{1-\Delta\beta} = \frac{0.0567s}{1-0.9} = 0.567s$$

访问一个对象总的平均响应时间为：

$$3 + 0.567 = 3.567\mathrm{s}$$

b. “miss rate”为 0.4，即命中率为 0.6，意思是请求的对象有60%由机构网络满足，40%由因特网满足，流量强度减少60%，则平均访问时延为：

$$\frac{\Delta}{1-\Delta\beta}=\frac{0.0567}{1-0.9\times 0.4}=0.089s$$

通过因特网访问一个对象所需要的时间为：$3+0.089=3.089\text{s}$，本地机构服务器满足40%的对象请求，访问时间为 0，因特网满足60%，时间为 3.123s 则总响应时间为：

$$0.4\times 3.089+0.6\times 0\approx 1.24s$$

由计算可知，在安装了缓存器之后，一个对象总平均响应时间由 3.567s 减少到了 1.24s，提高了效率。

P10. Consider a short, 10-meter link, over which a sender can transmit at a rate of 150 bits/sec in both directions. Suppose that packets containing data are 100,000 bits long, and packets containing only control (e.g., ACK or hand- shaking) are 200 bits long. Assume that N parallel connections each get 1/N of the link bandwidth. Now consider the HTTP protocol, and suppose that each downloaded object is 100 Kbits long, and that the initial downloaded object contains 10 referenced objects from the same sender. Would parallel downloads via parallel instances of non-persistent HTTP make sense in this case? Now consider persistent HTTP. Do you expect significant gains over the non-persistent case? Justify and explain your answer.

答：由题意可知，每下载一个对象可以完整的将一个分组投入链路，用T_p表示在服务器和客户机之间的单向传播时延。

首先考虑非持久连接的并行下载，并行下载允许 10 条连接共享 150bits/sec 的带宽，每个是 15bits/sec。非持久连接并行下载包括两个部分，第一部分是：TCP 三次握手以及下载整个对象，两台主机之间建立连接；第二部分是：因为是非持久连接，在下载每一个对象时，要再一次进行 TCP 三次握手，完成请求和响应过程。因此接收所有对象所花费的时间为：

$$\left(\frac{200}{150}+\mathrm{T}_p+\frac{200}{150}+\mathrm{T}_p+\frac{200}{150}+\mathrm{T}_p+\frac{100000}{150}+\mathrm{T}_p\right)+\left(\frac{200}{150/10}+\mathrm{T}_p+\frac{200}{150/10}+\mathrm{T}_p+\frac{200}{150/10}+\mathrm{T}_p+\frac{10000}{150/10}+\mathrm{T}_p\right)=7377+8T_p(seconds)$$

现在考虑持久 HTTP 连接，在服务器和客户机之间的连接建立以后，无需建立新的连接，在此连接上可以传输所有对象，花费总时间为：

$$\left(\frac{200}{150}+\mathrm{T}_p+\frac{200}{150}+\mathrm{T}_p+\frac{200}{150}+\mathrm{T}_p+\frac{100000}{150}+\mathrm{T}_p\right)+10\times\left(\frac{200}{150}+\mathrm{T}_p+\frac{10000}{150}+\mathrm{T}_p\right)=7351+24T_p(seconds)$$

假定光的传播速度是$3\times 10^8\mathrm{m/s}$，那么$\mathrm{T}_p=10m\div(3\times 10^8 m/s)=0.03ms$，因此传播时延$\mathrm{T}_p$相对于传输时延而言，可以忽略不计。

由上面计算我们可以看出，持久的 HTTP 连接不是明显的比非持久的并行下载快，快的不到1%。

P11. Consider the scenario introduced in the previous problem. Now suppose that the link is shared by Bob with four other users. Bob uses parallel instances of non-persistent HTTP, and the other four users use non-persistent HTTP without parallel downloads.

a. Do Bob's parallel connections help him get Web pages more quickly? Why or why not?

b. If all five users open five parallel instances of non-persistent HTTP, then would Bob's parallel connections still be beneficial? Why or why not?

答：a. 是的，因为 Bob 有更多的连接，他可以共享更大的链路带宽。

b. Bob 仍然需要执行并行下载，否则，对于其他四个用户来说，他将获得更少的链路带宽。

P12. Write a simple TCP program for a server that accepts lines of input from a client and prints the lines onto the server’s standard output. (You can do this by modifying the TCPServer.py program in the text.) Compile and execute your program. On any other machine that contains a Web browser, set the proxy server in the browser to the host that is running your server program; also con- figure the port number appropriately. Your browser should now send its GET request messages to your server, and your server should display the messages on its standard output. Use this platform to determine whether your browser generates conditional GET messages for objects that are locally cached.

答：如下：

```
TCPServer.java
 import java.io.*;
 import java.net.*;
 class TCPServer {
     public static void main(String argv[]) throws Exception
         {
                 String clientSentence;
                 ServerSocket welcomeSocket = new ServerSocket(6789);
                 while(true)
           {
                       Socket connectionSocket = welcomeSocket.accept();
                       BufferedReader inFromClient = new BufferedReader(new
                     InputStreamReader(connectionSocket.getInputStream( ) ) );
                       clientSentence = inFromClient.readLine();
                       System.out.println(“RECEIVED FROM CLIENT : ” +
               clientSentence    + “\n”);
           }
         }
   }
```

P13. What is the difference between MAIL FROM: in SMTP and From: in the mail message itself?

答：SMTP 中的 MAIL FROM：是 SMTP 握手协议的一部分，是来自于客户端的消息，标识发送到 SMTP 服务器的邮件消息的发送者。而邮件消息本身中的 From：是邮件报文实体的一部分，而不是一个 SMTP 消息。

P14. How does SMTP mark the end of a message body? How about HTTP? Can HTTP use the same method as SMTP to mark the end of a message body? Explain.

答：SMTP 用仅仅只包含时间的一行，来标记作为信息主体的末尾。HTTP 用“Content-Length header field (首部字段长度) ”来指出信息主体的长度。两者不能用同一样的方法来标记末尾，因为 HTTP 信息可以用 2 进制数据来表示，然而，SMTP 必须用 7 比特的 ASCII 格式来表示。

P15. Read RFC 5321 for SMTP. What does MTA stand for? Consider the follow- ing received spam email (modified from a real spam email). Assuming only the originator of this spam email is malacious and all other hosts are honest, identify the malacious host that has generated this spam email.

From - Fri Nov 07 13:41:30 2008
Return-Path: <tennis5@pp33head.com>
Received: from barmail.cs.umass.edu (barmail.cs.umass.edu [128.119.240.3]) by cs.umass.edu (8.13.1/8.12.6) for <hg@cs.umass.edu>; Fri, 7 Nov 2008 13:27:10 -0500
Received: from asusus-4b96 (localhost [127.0.0.1]) by barmail.cs.umass.edu (Spam Firewall) for <hg@cs.umass.edu>; Fri, 7 Nov 2008 13:27:07 -0500 (EST)
Received: from asusus-4b96 ([58.88.21.177]) by barmail.cs.umass.edu for <hg@cs.umass.edu>; Fri, 07 Nov 2008 13:27:07 -0500 (EST)
Received: from [58.88.21.177] by inbnd55.exchangeddd.com; Sat, 8 Nov 2008 01:27:07 +0700
From: "Jonny" <tennis5@pp33head.com>
To: <hg@cs.umass.edu>
Subject: How to secure your savings

答：MTA 指“Mail Transfer Agent”。一个主机将报文信息发送给 MTA，然后这个信息在大量的 MAT 报文处有一个序号，以便到达接收方的邮箱。我们看到这个 MTA 报文跟随在一连串 MTA 报文后面。一个有信用的 MTA 应该报告它在何处收到这个 MTA。在这个报文中我们注意到，“asusus-4b96 ([58.88.21.177])”，它没有报告它在何处收到这份电子邮件，因此我们仅仅认为这份邮件的发起者是不诚实的，因此“asusus-4b96 ([58.88.21.177])”一定是发起者。

P16. Read the POP3 RFC, RFC 1939. What is the purpose of the UIDL POP3 command?

答：UIDL 是唯一识别码列表的缩写。当一个 POP3 客户端发出一个 UIDL 命令，服务器返回储存在用户邮箱里的所有邮件的唯一邮件识别码。这个命令对下载并保留方式有用。通过保留上次收取的邮件的列表信息，客户能够使用 UIDL 命令来确定在服务器上的哪些邮件是已经被阅读过的。

注：

UIDL POP3 命令格式: •UIDL[msg]

【参数】信件数（可选）。如果给出信件数，不包括被标记为删除的信件。

【限制】仅在"操作"状态下使用。

【说明】

如果给出了参数，且 POP3 服务器返回包括上述信息的"确认"，此行称为信息的"独立-ID"。如果没有参数，服务器返回"确认"响应，此响应便以多行给出。在最初的+OK 后，对于每个信件，服务器均给出相应的响应。此行叫做信件的"独立-ID 表"。为简化语法分析，所有服务器要求使用独立-ID 表的特定格式。它包括空格和信件的独立-ID。信件的独立-ID 由 0x21 到 0x7E 字符组成，这个符号在给定的存储邮件中不会重复。

P17. Consider accessing your e-mail with POP3.

a. Suppose you have configured your POP mail client to operate in the download-and-delete mode. Complete the following transaction:

C: list
S: 1 498

S: 2 912

S: .

C: retr 1

S: blah blah ... S: ....blah S: .

?

?

b. Suppose you have configured your POP mail client to operate in the download-and-keep mode. Complete the following transaction:

C: list

S: 1 498

S: 2 912

S: .

C: retr 1

S: blah blah ... S: ....blah S: .

?

?

c. Suppose you have configured your POP mail client to operate in the download-and-keep mode. Using your transcript in part (b), suppose you retrieve messages 1 and 2, exit POP, and then five minutes later you again access POP to retrieve new e-mail. Suppose that in the five-minute interval no new messages have been sent to you. Provide a transcript of this second POP session.

答：a. 如下：

C: dele 1

C: retr 2

S: (blah blah …

S: ………..blah)

S: .

C: dele 2

C: quit

S: +OK POP3 server signing off

b. 如下：

C: retr 2

S: blah blah …

S: ………..blah

S: .

C: quit

S: +OK POP3 server signing off

c. 如下：

C: list

S: 1 498

S: 2 912

S: .

C: retr 1
S: blah …..
S: ….blah
S: .
C: retr 2
S: blah blah …
S: ………..blah
S: .
C: quit
S: +OK POP3 server signing off

P18. a. What is a whois database?

b. Use various whois databases on the Internet to obtain the names of two DNS servers. Indicate which whois databases you used.

c. Use nslookup on your local host to send DNS queries to three DNS servers: your local DNS server and the two DNS servers you found in part (b). Try querying for Type A, NS, and MX reports. Summarize your findings.

d. Use nslookup to find a Web server that has multiple IP addresses. Does the Web server of your institution (school or company) have multiple IP addresses?

e. Use the ARIN whois database to determine the IP address range used by your university.

f. Describe how an attacker can use whois databases and the nslookup tool to perform reconnaissance on an institution before launching an attack.

g. Discuss why whois databases should be publicly available.

答：a. 对于一个给定的域名，IP 地址或网络管理员名的输入，whois 数据库能被用来定位相应的登记人，whois 服务器，DNS 服务器等。

b. 获得两台 DNS 服务器的名字：NS4.YAHOO.COM 从 www.register.com 得到；NS1.MSFT.NET 从 www.register.com 得到。

c．Local Domain: www.mindspring.com

Web servers : www.mindspring.com
207.69.189.21, 207.69.189.22,
207.69.189.23, 207.69.189.24,
207.69.189.25, 207.69.189.26, 207.69.189.27,
207.69.189.28

Mail Servers : mx1.mindspring.com (207.69.189.217)
mx2.mindspring.com (207.69.189.218)
mx3.mindspring.com (207.69.189.219)
mx4.mindspring.com (207.69.189.220)

Name Servers: itchy.earthlink.net (207.69.188.196)
scratchy.earthlink.net (207.69.188.197)

www.yahoo.com

Web Servers: www.yahoo.com (216.109.112.135, 66.94.234.13)

Mail Servers: a.mx.mail.yahoo.com (209.191.118.103)
b.mx.mail.yahoo.com (66.196.97.250)

c.mx.mail.yahoo.com (68.142.237.182, 216.39.53.3)
d.mx.mail.yahoo.com (216.39.53.2)
e.mx.mail.yahoo.com (216.39.53.1)
f.mx.mail.yahoo.com (209.191.88.247, 68.142.202.247)
g.mx.mail.yahoo.com (209.191.88.239, 206.190.53.191)

Name Servers: ns1.yahoo.com (66.218.71.63)
ns2.yahoo.com (68.142.255.16)
ns3.yahoo.com (217.12.4.104)
ns4.yahoo.com (68.142.196.63)
ns5.yahoo.com (216.109.116.17)
ns8.yahoo.com (202.165.104.22)
ns9.yahoo.com (202.160.176.146)

www.hotmail.com
Web Servers: www.hotmail.com (64.4.33.7, 64.4.32.7)

Mail Servers: mx1.hotmail.com (65.54.245.8, 65.54.244.8, 65.54.244.136)
mx2.hotmail.com (65.54.244.40, 65.54.244.168, 65.54.245.40)
mx3.hotmail.com (65.54.244.72, 65.54.244.200, 65.54.245.72)
mx4.hotmail.com (65.54.244.232, 65.54.245.104, 65.54.244.104)

Name Servers: ns1.msft.net (207.68.160.190)
ns2.msft.net (65.54.240.126)
ns3.msft.net (213.199.161.77)
ns4.msft.net (207.46.66.126)
ns5.msft.net (65.55.238.126)

nslookup –type=A … 该查询请求类型为 A 的该主机名对应的 IP 地址。
nslookup –type=NS … 该查询请求类型为 NS 的该域的权威 DNS 服务器的主机名。
nslookup –type=MX … 该查询请求类型为 MX 的该别名的邮件服务器的规范主机名。

d．雅虎（yahoo）的 Web 服务器具有多个 IP 地址， www.yahoo.com (216.109.112.135, 66.94.234.13)

e. 我所在的大学，陕西师范大学的(ShannXi Normal University)网络地址范围为：202.117.144.0 - 202.117.159.255。

f. 一个入侵者能使用 whois 数据库和 nslookup 工具来检测目标机构的 IP 地址范围，DNS 服务器地址等。

g．通过分析攻击包的源地址信息，受害者能够使用 whois 来掌握有关于攻击来源的域的信息，并能够通知来源域的管理员。

P19. In this problem, we use the useful dig tool available on Unix and Linux hosts to explore the hierarchy of DNS servers. Recall that in Figure 2.21, a DNS server higher in the DNS hierarchy delegates a DNS query to a DNS server lower in the hierarchy, by sending back to the DNS client the name of that lower-level DNS server. First read the man page for dig, and then answer the following questions.

a. Starting with a root DNS server (from one of the root servers [a-m].root- servers.net), initiate a sequence of queries for the IP address for your department’s Web server by using dig. Show the list of the names of DNS servers in the delegation chain in answering your query.

b. Repeat part a) for several popular Web sites, such as google.com, yahoo.com, or amazon.com.

答：a. 下面以 gaia.cs.umass.edu 为例：

a.root-servers.net

E.GTLD-SERVERS.NET

ns1.umass.edu(authoritative)

First command:

dig +norecurse @a.root-servers.net any gaia.cs.umass.edu

;; AUTHORITY SECTION:

edu.	172800	IN	NS	E.GTLD-SERVERS.NET.
edu.	172800	IN	NS	A.GTLD-SERVERS.NET.
edu.	172800	IN	NS	G3.NSTLD.COM.
edu.	172800	IN	NS	D.GTLD-SERVERS.NET.
edu.	172800	IN	NS	H3.NSTLD.COM.
edu.	172800	IN	NS	L3.NSTLD.COM.
edu.	172800	IN	NS	M3.NSTLD.COM.
edu.	172800	IN	NS	C.GTLD-SERVERS.NET.

Among all returned edu DNS servers, we send a query to the first one.

dig +norecurse @E.GTLD-SERVERS.NET any gaia.cs.umass.edu

umass.edu.	172800	IN	NS	ns1.umass.edu.
umass.edu.	172800	IN	NS	ns2.umass.edu.
umass.edu.	172800	IN	NS	ns3.umass.edu.

Among all three returned authoritative DNS servers, we send a query to the first one.

dig +norecurse @ns1.umass.edu any gaia.cs.umass.edu

gaia.cs.umass.edu.	21600	IN	A	128.119.245.12

b. google.com 网站的答案可能包括：

a.root-servers.net

E.GTLD-SERVERS.NET

ns1.google.com(authoritative)

P20. Suppose you can access the caches in the local DNS servers of your department. Can you propose a way to roughly determine the Web servers (outside your department) that are most popular among the users in your department? Explain.

答：我们可以周期性的对本地 DNS 服务器的 DNS 缓存进行快照(即访问)，最常见出现在 DNS 缓存的网站服务器是最受欢迎的服务器。这是因为，如果更多的用户这这个网站感兴趣，用户会发送对这个网站服务器发送更频繁的请求。因此，这个网站会更频繁的出现在这 DNS 缓存中。

为了做一个更完整的研究，如下：

Craig E. Wills, Mikhail Mikhailov, Hao Shang
"Inferring Relative Popularity of Internet Applications by Actively Querying DNS Caches", in IMC'03, October 27-29, 2003, Miami Beach, Florida, USA

P21. Suppose that your department has a local DNS server for all computers in the department. You are an ordinary user (i.e., not a network/system administrator). Can you determine if an external Web site was likely accessed from a computer in your department a couple of seconds ago? Explain.

答：在 Unix 或者 Linux 环境下，我们可以用 dig 程序在本地 DNS 服务器中去查询 Web 站点。比如，"dig baidu.com"将会返回对"baidu.com"站点的查询时间，如果 baidu.com 在几秒前就被访问过，在本地 DNS 缓存中，baidu.com 的记录被缓存，因此查询时间会为 0。否则，查询时间会很漫长。

P22. Consider distributing a file of F = 15 Gbits to N peers. The server has an upload rate of us = 30 Mbps, and each peer has a download rate of di = 2 Mbps and an upload rate of u. For N = 10, 100, and 1,000 and u = 300 Kbps, 700 Kbps, and 2 Mbps, prepare a chart giving the minimum distribution time for each of the combinations of N and u for both client-server distribution and P2P distribution.

答：在客户机/服务器分发模式下，为了计算最小分发时间，我们用以下公式：

$$D_{cs} = max\left\{\frac{NF}{u_s}, \frac{F}{d_{min}}\right\}$$

在 P2P 分发模式下，为了计算最小分发时间，用以下公式：

$$\mathrm{D}_{P2P} = max\left\{\frac{F}{u_s}, \frac{F}{d_{min}}, \frac{NF}{u_s + \sum_{i=1}^{N} u_i}\right\}$$

在这里，$\mathrm{F} = 15\mathrm{Gbits} = 15 \times 1024\mathrm{Mbits}$，$u_s = 30Mbps$，$d_{min} = d_i = 2Mbps$，注意，$300\mathrm{Kbps} = 300/1024\mathrm{Mbps}$，则：

客户机服务器模式：

	N		
	10	**100**	**1000**
300 Kbps	7680	51200	512000
700 Kbps	7680	51200	512000
2 Mbps	7680	51200	512000

P2P 模式：

	N		
	10	**100**	**1000**
300 Kbps	7680	25904	47559
700 Kbps	7680	15616	21525
2 Mbps	7680	7680	7680

P23. Consider distributing a file of F bits to N peers using a client-server architecture. Assume a fluid model where the server can simultaneously transmit to multiple peers, transmitting to each peer at different rates, as long as the combined rate does not exceed us.

a. Suppose that us/N≤dmin. Specify a distribution scheme that has a distri- bution time of NF/us.
b. Suppose that us/N≥dmin. Specify a distribution scheme that has a distri- bution time of F/ dmin.
c. Conclude that the minimum distribution time is in general given by max{NF/us, F/ dmin}.
答：a. 向 N 个对等方发送文件，令向每个对等方发送的速率为u_s/N，因为$u_s/N \le d_{min}$，所以每个接收方接收完整个文件的时间为$\frac{F}{u_{s/N}} = \frac{FN}{u_s}$，向每个对等方的发送时间一样，所以总分发时间仍等于$\frac{FN}{u_s}$。

b. 因为$u_s/N \ge d_{min}$，所以取向每个对等方发送的速率为F/d_{min}，所以每个接收方接收完整文件的时间为F/d_{min}。向每个对等方发送的时间一样，所以总分发时间等于F/d_{min}。

c. 由 2.6 节可知，$D_{CS} \ge max\left\{\frac{FN}{u_s}, \frac{F}{d_{min}}\right\}$。若$(u_s/N) \le d_{min}$，则$D_{CS} \ge \frac{FN}{u_s}$，又由 a 可知在其他分配速率 F，$D_{CS} \le \frac{FN}{u_s}$，所以$D_{CS} = \frac{FN}{u_s}$；若$(u_s/N) \ge d_{min}$，则$D_{CS} \ge \frac{F}{d_{min}}$又由 b 可知在其他分配速率 F，$D_{CS} \le \frac{F}{d_{min}}$，所以$D_{CS} = \frac{F}{d_{min}}$所以，综上所述$D_{CS} = max\left\{\frac{FN}{u_s}, \frac{F}{d_{min}}\right\}$。

P24. Consider distributing a file of F bits to N peers using a P2P architecture. Assume a fluid model. For simplicity assume that dmin is very large, so that peer download bandwidth is never a bottleneck.
a. Suppose that us ≤ (us + u1 + ... + uN)/N. Specify a distribution scheme that has a distribution time of F/us.
b. Suppose that us ≥ (us + u1 + ... + uN)/N. Specify a distribution scheme that has a distribution time of NF/(us + u1 + ... + uN).
c. Conclude that the minimum distribution time is in general given by max{F/us, NF/(us + u1 + ... + uN)}.
答：a. 定义$u = u_1 + u_2 + \cdots + u_n$。我们假定：
$u_s \le (u_s + u)/N$　　等式 1
将这个文件分成N份，第i份大小为$\frac{u_i}{u}F$。服务器将第i份传给对等方i的速率为$r_i = \frac{u_i}{u} \times u_s$。由$r_1 + r_2 + \cdots r_N = u_s$，因此服务器对 N 个对等方的传输速率之和不会超过不会超过服务器的链路传输速率。并且每一个对等方 i 以速率r_i从$N-1$个对等方接收 bits 再进行转发。对于每一个对等方 i 而言总转发速率为：$(N-1)r_i$。我们有：

$$(N-1)r_i = (N-1)(u_s u_i)/u \le u_i$$

在这里不等号服从等式 1。对等方 i 的总转发速率低于链路传输速率u_i。
在这个分发方案里，对等方 i 接收 bits 的总速率为：

$$r_i + \sum_{j \ne i} r_j = u_s$$

因此每个对等方接收文件的速率为：F/u_s。
b. 再一次定义$u = u_1 + u_2 + \cdots + u_n$。我们假定：
$u_s \ge (u_s + u)/N$　　等式 2
令$r_i = u_i/(N-1)$并且$r_{N+1} = \left(u_s - u/(N-1)\right)/N$，
在这个分配方案里，这个文件被分成$N+1$份，服务器以速率r_i将第 i 个部分的 bits 发往第 i 个对等方$(i = 1, \cdots, N)$。每一个对等方 i 以速率r_i向其它 N-1 个对等方转发 bits。此外，服务

器从第$\mathrm{N}+1$部分开始，以速率r_{N+1}向其它 N 个对等方发送 bits。这些对等方不会转发来自第$\mathrm{N}+1$部分的 bits。

服务器的总发送速率为：

$$r_1+r_2+\cdots r_N+Nr_{N+1}=\frac{\mathrm{u}}{(N-1)}+u_s-\frac{\mathrm{u}}{(N-1)}=u_s$$

因此发送方的发送速率不会超过它的链路连接速率。对等方 i 的总发送速率为：

$$(\mathrm{N}-1)r_i=u_i$$

因此每个对等方的发送速率不会超过它的链路连接速率。

在这个分发方案里，对等方 i 接收 bits 的总速率为：

$$r_i+r_{N+1}+\sum_{j\neq i}r_j=\frac{u}{(N-1)}+\frac{\left(u_s-\frac{u}{N-1}\right)}{N}=(u_s+u)/N$$

因此每个对等方接收文件的速率为：$\mathrm{NF}/(u_s+u)$。

(简而言之，我们忽略了详细说明这个文件第$\mathrm{i}=1,\cdots,\mathrm{N}+1$部分的大小。我们在这里提供说明。令$\Delta=(u_s+u)/N$为分发时间。对于$\mathrm{i}=1,\cdots,\mathrm{N}+1$而言，文件第 i 部分大小是$F_i=\mathrm{r}_i\Delta bits$。第 N+1 部分文件大小为$F_{N+1}=\mathrm{r}_{N+1}\Delta bits$。这个直观显示了$F_1+\cdots+F_{N+1}=F$。)

c. 这个问题的解决办法和 17(c)类似，我们从 2.6 节可知：

$$D_{P2P}\geq max\left\{\frac{F}{u_s},\frac{NF}{(u_s+u)}\right\}$$

结合 a, b 我们可以得到这个答案。

P25. Consider an overlay network with N active peers, with each pair of peers hav- ing an active TCP connection. Additionally, suppose that the TCP connections pass through a total of M routers. How many nodes and edges are there in the corresponding overlay network?

答：在覆盖的网络中有 N 个节点，对应有$\frac{\mathrm{N}(N-1)}{2}$条边。

P26. Suppose Bob joins a BitTorrent torrent, but he does not want to upload any data to any other peers (so called free-riding).

a. Bob claims that he can receive a complete copy of the file that is shared by the swarm. Is Bob’s claim possible? Why or why not?

b. Bob further claims that he can further make his “free-riding” more efficient by using a collection of multiple computers (with distinct IP addresses) in the computer lab in his department. How can he do that?

答：a. 他的声明是可行的，只要在长时间内有足够多的对等方在比特洪流内。Bob 可以从其他对等方通过乐观的疏通的链路接收数据。

b.他的第二次声明也是可行的。他可以在每台主机上运行一个客户端，让每一个客户端“free-riding”，收集来自不同主机的大量数据，并将他们结合成一个单一的文件。他甚至可以写一分校的目录程序，对于这份文件的不同的数据块对不同的主机进行请求。这是 P2P 网络典型的一种 Sybil 攻击。

P27. In the circular DHT example in Section 2.6.2, suppose that peer 3 learns that peer 5 has left. How does peer 3 update its successor state information? Which peer is now its first successor? Its second successor?

答：因为对等方 3 已经了解到对等方 5 离开系统，所以对等方 3 会向它的第一继任对等方(对

等方 4)请求请求它的立即继任对等方(对等方 8)的标示符。对等方 3 然后将对等方 8 作为第二继任对等方。

P28. In the circular DHT example in Section 2.6.2, suppose that a new peer 6 wants to join the DHT and peer 6 initially only knows peer 15's IP address. What steps are taken?

答：对等方 6 首先向对等方 15 发送一个报文，说"对等方 6 的前驱和后继对等方是什么?"，这个报文通过 DHT 转发直到它到达对等方 5 和 8，5 意识到它是 6 的前驱对等方，8 意识到它是 6 的后继对等方。然后，对等方 5 发送这个前驱后继对等方信息返回给对等方 6。对等方 6 现在可以加入 DHT 表，让对等方 8 成为它的后继对等方并且通知对等方 5 它的直接后继对等方变为 6。

P29. Because an integer in $[0,\cdots,2^{n-1}]$ can be expressed as an n-bit binary number in a DHT, each key can be expressed as k = (k0, k1, . . . , kn–1), and each peer identifier can be expressed p = (p0, p1, . . . , pn–1). Let's now define the XOR dis- tance between a key k and peer p as

$$\mathrm{d}(k,p)=\sum_{j=0}^{n=1}\left|k_j-p_j\right|2^j$$

Describe how this metric can be used to assign (key, value) pairs to peers. (To learn about how to build an efficient DHT using this natural metric, see [Maymounkov 2002] in which the Kademlia DHT is described.)

答：对于每一个 key，我们第一步计算出它本身到所有对等方的距离，用$\mathrm{d}(k,p)$，然后在和 key 最近的对等方里面存储 key(用最小距离值)。

P30. As DHTs are overlay networks, they may not necessarily match the underlay physical network well in the sense that two neighboring peers might be physically very far away; for example, one peer could be in Asia and its neighbor could be in North America. If we randomly and uniformly assign identifiers to newly joined peers, would this assignment scheme cause such a mismatch? Explain. And how would such a mismatch affect the DHT's performance?

答：是的，向对等方随机分配 keys，根本不考虑底层网络，因此它很可能导致配错(错误匹配)。

这样的错误匹配可能会降低查询性能。比如，考虑一个逻辑路径 p1(仅仅考虑两条链路)：A→B→C，A 和 B 是相邻对等方，B 和 C 是相邻对等方。假定有另一条逻辑链路 p2，是从 A 到 C 的(考虑有三条逻辑链路)：A→D→E→C。

可能导致 A 和 B 在物理上来说非常遥远(通过许多路由器隔开)，并且 B 和 C 在物理上来说非常遥远(通过许多路由器隔开)。但是也可能出现以下情形：A，D，E 三者和 C 在物理上十分接近(通过很少的路由器隔开)。换言之，一个非常短的的逻辑链路对应一个非常长的物理链路。

P31. Install and compile the Python programs TCPClient and UDPClient on one host and TCPServer and UDPServer on another host.

a. Suppose you run TCPClient before you run TCPServer. What happens? Why?

b. Suppose you run UDPClient before you run UDPServer. What happens? Why?

c. What happens if you use different port numbers for the client and server sides?

答：a. 如果先运行 TCPClient，客户端将会尝试和一个并不存在服务器进程建立连接，不会建立 TCP 连接。

b. UDPClient 不会和服务器建立 TCP 连接。因此，当你首次运行 UDPClient 时一切正常，然后运行 UDPServer，然后可以打字输入。

c. 如果使用不同端口号，然后客户机会尝试和错误的或者不存在的进程建立连接。会出现错误。

P32. Suppose that in UDPClient.py, after we create the socket, we add the line:

clientSocket.bind(('', 5432))

Will it become necessary to change UDPServer.py? What are the port numbers for the sockets in UDPClient and UDPServer? What were they before making this change?

答：在原始的程序中，UDPClient 在创建套接字时并没有明确指出端口号。在这种情形下，代码会让底层操作系统选择一个端口号。在有了这个额外的一行后，在 UDPClient 被执行时，一个带有端口号 5432 的套接字 UDP 套接字被创建。

UDPServer 需要知道客户机的端口号以便它可以将分组发送给正确的客户机套接字。通过在 UDPServer 一看，我们了解到客户机端口号不是和服务器代码硬性相连的；而是，UDPServer 通过解析来自客户机的数据报，来决定客户机端口号。UDP Server 将会使用任何客户机端口号工作，包括 5432。UDPServer 不需要被修改。

在此之前，需要执行：

Client socket = x (chosen by OS)

Server socket = 9876

在之后需要执行：

Client socket = 5432

P33. Can you configure your browser to open multiple simultaneous connections to a Web site? What are the advantages and disadvantages of having a large number of simultaneous TCP connections?

答：是的，你可以配置多个浏览器来对一个网站打开多重同步连接。优点是可能下载文件的速度更快。缺点是你可能会占用带宽。由此会显著减慢共享物理链路的其他用户的下载速度。

P34. We have seen that Internet TCP sockets treat the data being sent as a byte stream but UDP sockets recognize message boundaries. What are one advantage and one disadvantage of byte-oriented API versus having the API explicitly recognize and preserve application-defined message boundaries?

答：对于一个运用，比如远程登录(telnet 和 ssh)，一个字节流定向协议(byte-stream oriented protocol)是非常自然的，我们仅仅先考虑 TCP 连接。

在其他应用程序中，我们可能会发送一些列报文段，这些报文段间有识别报文边界。比如，当一个 SMTP 邮件服务器发送和另一个邮件服务器之间彼此发送一些邮件报文。由于 TCP 没有机制来指出报文边界，这个应用必须给它本身添加信息，因此应用的接收方可以识别当前报文和下一个报文。如果没一个报文段被放入有区别的 UDP 报文段里面，接收方能够识别各种各样的报文段不用发送方的应用程序额外提供报文边界信息。

P35. What is the Apache Web server? How much does it cost? What functionality does it currently have?

You may want to look at Wikipedia to answer this question.

答：为了创建一个一个网站服务器，我们需要在一台主机上运行网站服务器软件。许多供应商提供网站服务器软件。然而，当今最受欢迎的网站服务器软件是 Apache，它是开源的免费的。多年以来，它通过开源社区已经高度最优化。

P36. Many BitTorrent clients use DHTs to create a distributed tracker. For these DHTs, what is the “key” and what is the “value”?

答：关键字“key”在哈希表“infohash”中，值“value”是 IP 地址，在哈希表中当前有文件指定了这个 IP 地址。

Chapter 3 Transport Layer

P1. Suppose Client A initiates a Telnet session with Server S. At about the same time, Client B also initiates a Telnet session with Server S. Provide possible source and destination port numbers for

a. The segments sent from A to S.

b. The segments sent from B to S.

c. The segments sent from S to A.

d. The segments sent from S to B.

e. If A and B are different hosts, is it possible that the source port number in the segments from A to S is the same as that from B to S?

f. How about if they are the same host?

答：假定 A 的源端口号是 467，B 是 513，由 Telnet 会话目的端口为 23 可得：

		source port number	destination port number
a	A→S	467	23
b	B→S	513	23
c	S→A	23	467
d	S→B	23	513

e. 可以相同，服务器是根据 IP 区分不同主机的。

f. 不能，若同一台主机，IP 相同，端口号也相同，服务器无法区分。

P2. Consider Figure 3.5. What are the source and destination port values in the seg- ments flowing from the server back to the clients' processes? What are the IP addresses in the network-layer datagrams carrying the transport-layer segments?

答：假设三台主机 IP 为 ABC，三个发送的数据分组为 1,2,3。则：

Packet	Source Port	Destination Port	Source IP Address	Destination IP Address
1	80	26145	B	C
2	80	7532	B	C
3	80	26145	B	A

P3. UDP and TCP use 1s complement for their checksums. Suppose you have the following three 8-bit bytes: 01010011, 01100110, 01110100. What is the 1s complement of the sum of these 8-bit bytes? (Note that although UDP and TCP use 16-bit words in computing the checksum, for this problem you are being asked to consider 8-bit sums.) Show all work. Why is it that UDP takes the 1s complement of the sum; that is, why not just use the sum? With the 1s complement scheme, how does the receiver detect errors? Is it possible that a 1-bit error will go undetected? How about a 2-bit error?

答：如果溢出，采取环绕式技术处理。

```
    0 1 0 1 0 0 1 1
  + 0 1 1 0 0 1 1 0
  -----------------
    1 0 1 1 1 0 0 1
```

```
    1 0 1 1 1 0 0 1
  + 0 1 1 1 0 1 0 0
  -----------------
    0 0 1 0 1 1 1 0
```

三个字节和的补码为 11010001。为了检测错误，接收方增加了四个字节(原始的三个字节以及检验和)。如果和里面有 0，接收方会知道有错误。所有的一比特的错误都能被检测出来，但是二比特的错误不能被检测出来(比如，第一个数字的最后一个数字变成 0 的同时第二个字节的最后一个数字变成 1)。

P4. a. Suppose you have the following 2 bytes: 01011100 and 01100101. What is the 1s complement of the sum of these 2 bytes?

b. Suppose you have the following 2 bytes: 11011010 and 01100101. What is the 1s complement of the sum of these 2 bytes?

c. For the bytes in part (a), give an example where one bit is flipped in each of the 2 bytes and yet the 1s complement doesn't change.

答：a. 这两个字节的和为：11000001，其反码为：00111110。

b. 这两个字节的和为：01000000，其反码为：10111111。

c. 第一个字节为：01010100，第二个字节为：01101101。

P5. Suppose that the UDP receiver computes the Internet checksum for the received UDP segment and finds that it matches the value carried in the checksum field. Can the receiver be absolutely certain that no bit errors have occurred? Explain.

答：不，接收方不能绝对的确认没有比特错误发生。这是因为数据分组的检验和也在计算之中。如果分组中两个 16bits 的字节对应的比特(加起来的)是 0 和 1，然后他们各自变成了 1 或者 0，和保持不变。因此接收方计算得到的以 1 开头即负数的补码保持不变。这意味着甚至有传输错误时检验和被判断正确。

P6. Consider our motivation for correcting protocol rdt2.1. Show that the receiver, shown in Figure 3.57, when operating with the sender shown in Figure 3.11, can lead the sender and receiver to enter into a deadlock state, where each is waiting for an event that will never occur.

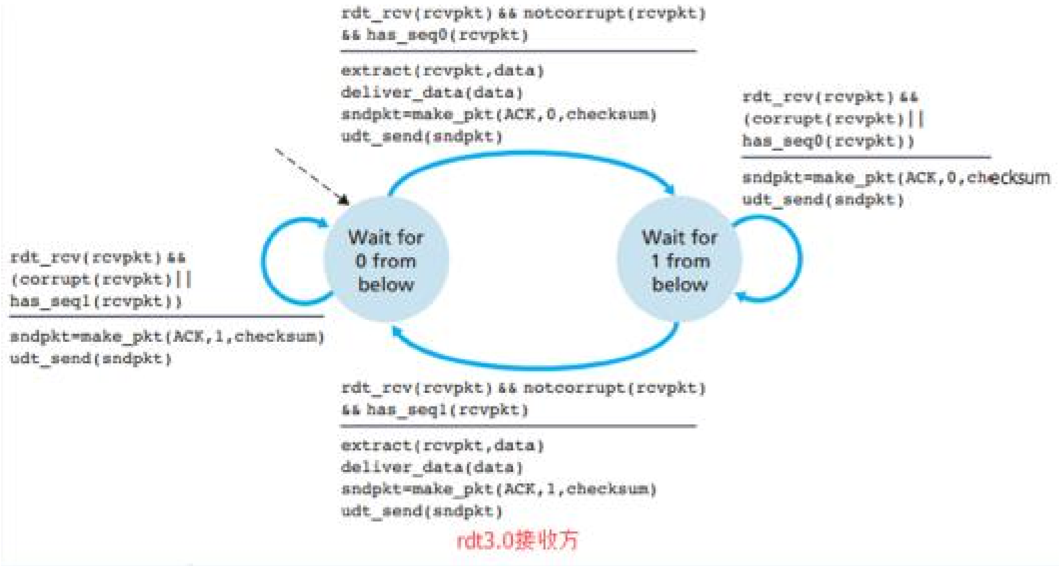
答：假定发送方处于状态"Wait for call 1 from above"并且接收方处于状态"Wait for 1 from below."发送方发送一个序号为 1 的分组，并且状态转变为"Wait for ACK or NAK 1"，等待一个 ACK 或者 NAK。假定现在接收方正确收到这一个序号为 1 的分组，发送一个 ACK 为 1，状态转变为"Wait for 0 from below"，等待一个序号为 0 的数据分组。然而，这个 ACK 是错误的。当 rdt2.1 发送方收到这个错误的 ACK，它重发一个序号为 1 的分组。然而，接收方等待序号为 0 的分组，并且总是发送一个 NAK，如果它没有收到一个序号为 0 的分组。因此发送方将一直发送一个序号为 1 的分组，并且接受方对这个分组一直回复 NAK。两者的状态都不会发生改变，进入死锁状态。

P7. In protocol rdt3.0, the ACK packets flowing from the receiver to the sender do not have sequence numbers (although they do have an ACK field that contains the sequence number of the packet they are acknowledging). Why is it that our ACK packets do not require sequence numbers?

答：为了最准确回答这个问题，首先考虑为什么我们需要序号。我们了解到发送方需要序号，以便接收方告知一个数据分组是否是一个已成功接收分组的副本。ACK 在这种情况下，发送方不需要这个信息(也就是，一个 ACK 的序号)来告知检测到一个重复的 ACK。一个重复的 ACK 对于 rdt3.0 接收方而言是显而易见的，自从它接收到一个最初的 ACK 以后就转变成下一个状态。这个重复的 ACK 不是发送方需要的 ACK 因此它会被 rdt3.0 发送方忽略掉。

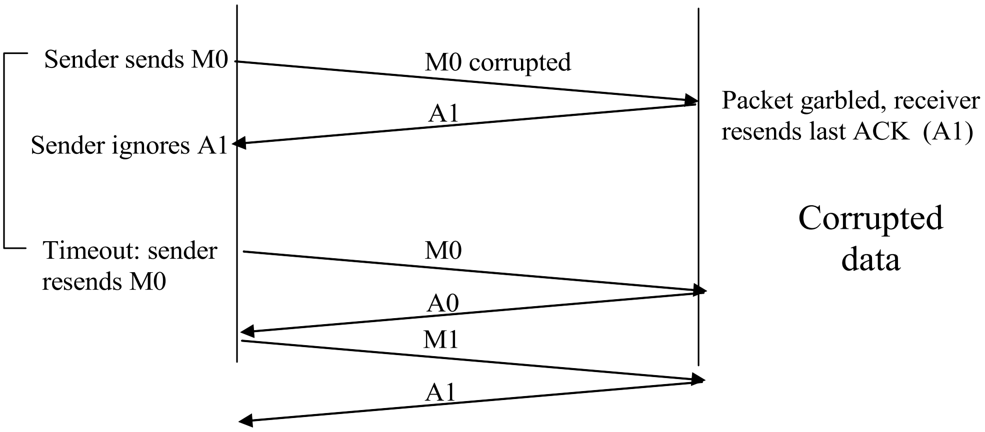
P8. Draw the FSM for the receiver side of protocol rdt3.0.

答：rdt3.0 的发送方与 rdt2.2 的发送方区别之处在于：增加了超时。我们已经了解到超时的引入会将重复分组的概率增加到-发送方接收方数据流上。然而，rdt2.2 的接收方已经能处理重复的分组(在 rdt2.2 接收方分组重复将会发生由于以下原因：接收方发送的 ACK 已丢失，发送方重传旧的数据)。因此 rdt2.2 接收方亦可作为 rdt3.0 的接收方。如下图：



P9. Give a trace of the operation of protocol rdt3.0 when data packets and acknowledgment packets are garbled. Your trace should be similar to that used in Figure 3.16.

答：假定这个协议已运行一段时间。发送方处于状态“Wait for call from above”(左上角)并且接收方的状态为：“Wait for 0 from below”。这个关于错误数据和错误 ACK 的情形如图 1：



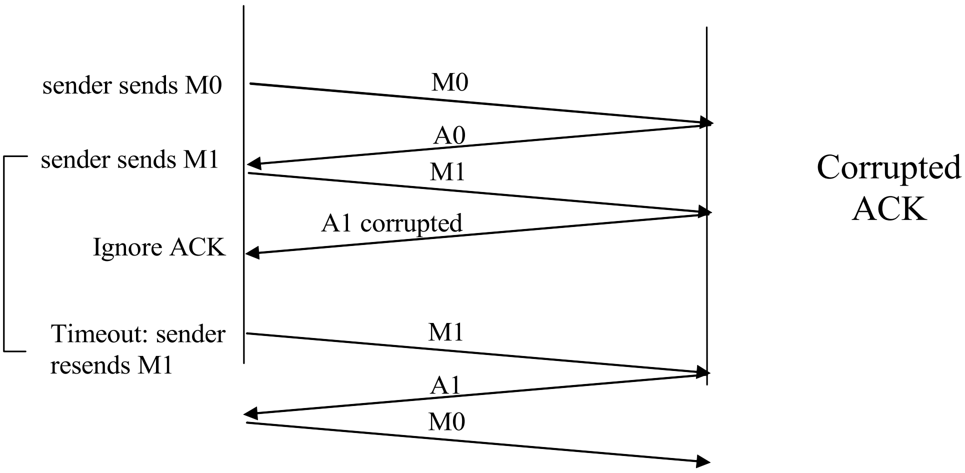


图 1：rdt3.0 情景：错误的数据，错误的 ACK

P10. Consider a channel that can lose packets but has a maximum delay that is known. Modify protocol rdt2.1 to include sender timeout and retransmit. Informally argue why your protocol can communicate correctly over this channel.

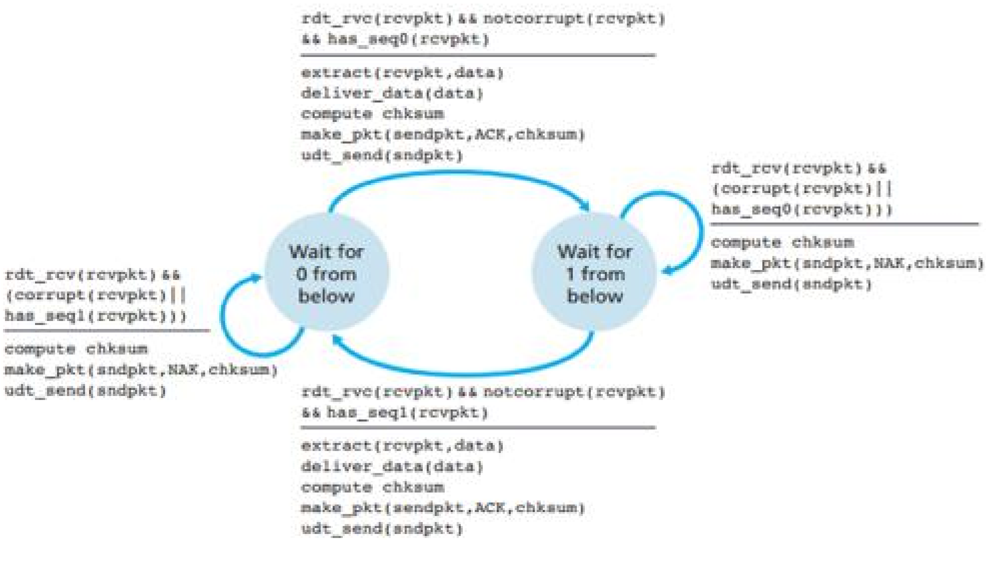


Figure 3.57 ♦ An incorrect receiver for protocol rdt 2.1

答：在这里，我们增加一个计时器，它的值大于已知的往返传播时延。对于状态“Wait for ACK or NAK0” 和“Wait for ACK or NAK1”我们增加一个超时事件。如果这个超时事件发生，最近传输的分组将被重传。让我们看看为什么这个协议依然运行着 rdt2.1 接收方协议。

a. 假定超时是由丢失数据分组引起的，即一个分组在发送方到接收方的信道中。在这种情况下，接收方永远不会接收到前一个传输的分组，并且从接收方的观点来看，如果超时重传被接收，看起来完全像最初传输的数据被接收到。

b. 现在假定 ACK 丢失。接收方在一个超时事件后最终将重传这个分组。但是，如果一个 ACK 被“扭曲”了(即超时未传达)，重传它几乎是完全一样的行为。因此发送方对一个丢失的 ACK 和一个“扭曲”的 ACK 会采取一样的行为。rdt2.1 的接收方已经能够处理“扭曲”ACK 的情形。

P11. Consider the rdt2.2 receiver in Figure 3.14, and the creation of a new packet in the self-transition (i.e., the transition from the state back to itself) in the Wait- for-0-from-below and the Wait-for-1-from-below states: sndpkt=make_pkt(ACK,0,checksum) and sndpkt=make_pkt(ACK,0, checksum). Would the protocol work correctly if this action were removed from the self-transition in the Wait-for-1-from-below state? Justify your answer. What if this event were removed from the self-transition in the Wait- for-0-from-below state? [Hint: In this latter case, consider what would happen if the first sender-to-receiver packet were corrupted.]

答：如果这个报文的发送被移除，发送方和接收方将会死锁，它们等待的事件将永不会出现。如下是情景：

- 发送方发送 pkt0，进入状态“Wait for ACK0 state”，等待接收方返回分组。
- 接收方处于“Wait for 0 from below”状态，并且从发送方收到一个错误的分组。假定它不发送任何东西给发送方只是重新进入“Wait for 0 from below”状态。

现在，发送方等待一些来自接收方的一些 ACK，接收方等待来自发送方的数据分组—陷入死锁。

P12. The sender side of rdt3.0 simply ignores (that is, takes no action on) all received packets that are either in error or have the wrong value in the acknum field of an acknowledgment packet. Suppose that in such circum- stances, rdt3.0 were simply to retransmit the current data packet. Would the protocol still work? (Hint: Consider what would happen if there were only bit errors; there are no packet losses but premature timeouts can occur. Consider how many times the nth packet is sent, in the limit as n approaches infinity.)

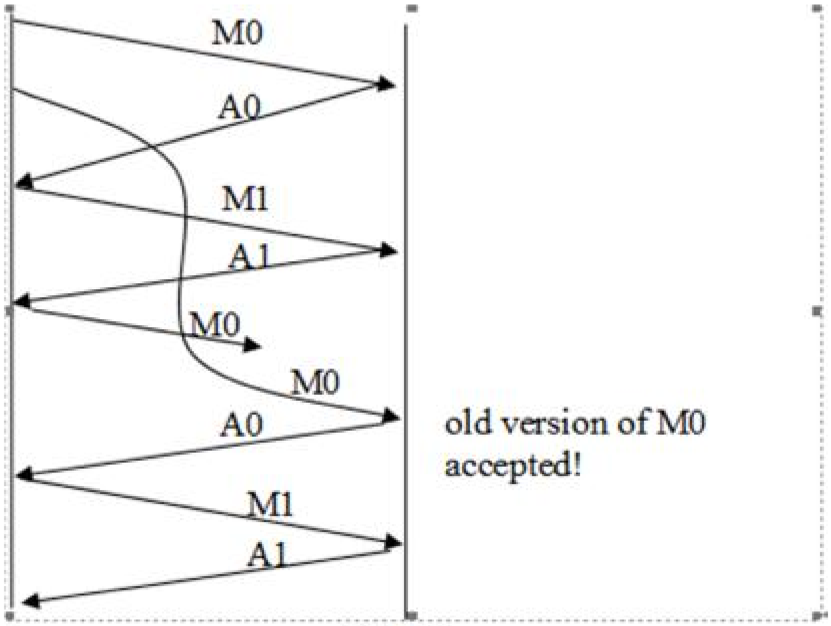
答：这个协议仍然在运行，如果这个收到的出现差错的分组实际上丢失的话，重传就会发生(并且从接收方观点来看，它从不知道这些两个事件已发生，或者即将发生)。

要对这个问题进行更进一步的讨论,就必须考虑到定时器超时过早发生的情况。在这种情况下，如果每个超大分组被确认，并且每个接收的超大分组确认信息导致另一个超大分组被发送，当 n 趋近于无穷时，分组 n 被发送的次数将无限增加。

P13. Consider the rdt 3.0 protocol. Draw a diagram showing that if the net- work connection between the sender and receiver can reorder messages (that is, that two messages propagating in the medium between the sender and receiver can be reordered), then the alternating-bit protocol will not work correctly (make sure you clearly identify the sense in which it will not work correctly).

Your diagram should have the sender on the left and the receiver on the right, with the time axis running down the page, showing data (D) and acknowledgment (A) message exchange. Make sure you indicate the sequence number associated with any data or acknowledgment segment.

答：如下图：



第13题

P14. Consider a reliable data transfer protocol that uses only negative acknowledgments. Suppose the sender sends data only infrequently. Would a NAK-only protocol be preferable to a protocol that uses ACKs? Why? Now suppose the sender has a lot of data to send and the end-to-end connection experiences few losses. In this second case, would a NAK-only protocol be preferable to a protocol that uses ACKs? Why?

答：在仅使用 NAK 的协议中，接收方只有当在接收到分组 *x+1* 时才能检测到分组 x 的丢失。也就是说接收方接收到 *x-1* 然后接收到 *x+1*，只有当接收方接收到 *x+1* 时才发现 x 的丢失。如果在传输 *x* 和传输 *x+1* 之间有很长时间的延时，那么在只有 NAK 的协议中，x 的修复要花费很长的时间。

另一方面，如果数据被发送，那么在只用 NAK 的协议中修复的速度将很快。并且，如果错误很少，那么 NAK 只是偶尔发送，并且从不发送 ACK，与只有 ACK 的情况相比，只有 NAK 的情况将明显减少反馈时间。

P15. Consider the cross-country example shown in Figure 3.17. How big would the window size have to be for the channel utilization to be greater than 98 percent? Suppose that the size of a packet is 1,500 bytes, including both header fields and data.

答：将一个分组发送到链路中需要12μs(或者0.012ms)，由$\frac{1500bytes}{1Gbps} = \frac{(1500\times8)b}{10^9bps} = 12\mu s$。为了使发送方信道利用率超过98%，即它至少在98%的时间处于忙碌状态。我们需满足：

$$\frac{0.012n}{30.012} \geq 0.98$$

解得 n 最小值为 2451，所以窗口大小至少为 2451 时可满足条件。

P16. Suppose an application uses rdt 3.0 as its transport layer protocol. As the stop-and-wait protocol has very low channel utilization (shown in the cross- country example), the designers of this application let the receiver keep sending back a number (more than two) of alternating ACK 0 and ACK 1 even if the corresponding data have not arrived at the receiver. Would this application design increase the channel utilization? Why? Are there any potential problems with this approach? Explain.

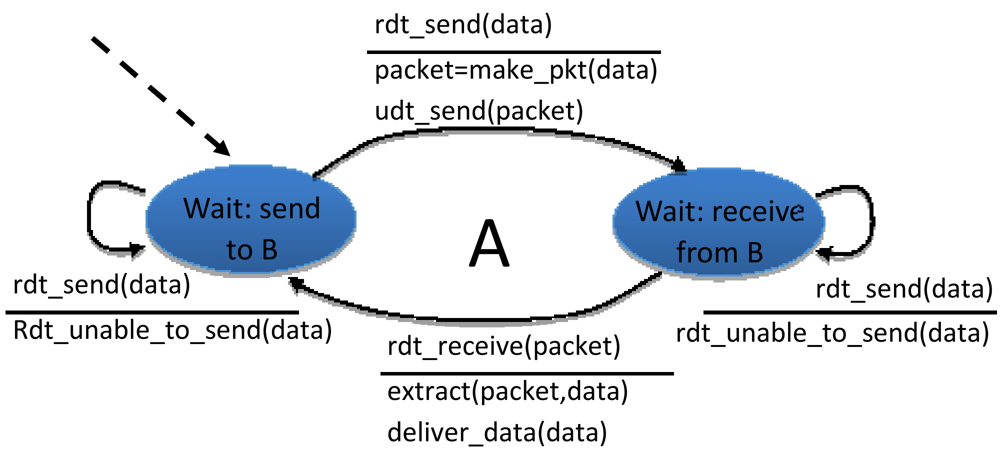
答：是的，这个确实导致发送方发送大量流状的数据(pipelined data)到信道中。

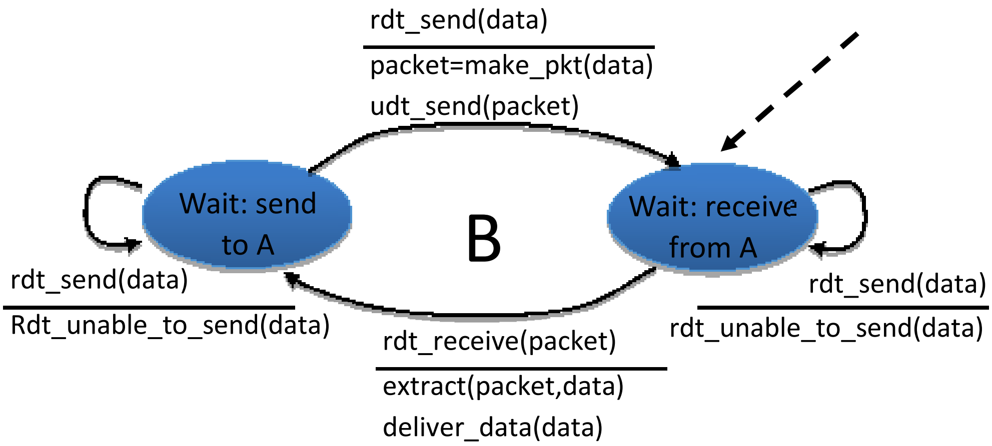
是的，存在一个潜在的问题。如果数据报文段在信道中丢失，rdt3.0 发送方不会重传那些丢失的数据，除非在应用程序中有一些额外的机制来恢复丢失的分组。

P17. Consider two network entities, A and B, which are connected by a perfect bidirectional channel(i.e., any message sent will be received correctly;the channel will not corrupt, lose, or re-order packets). A and B are to deliver data messages to each other in an alternating manner: First, A must deliver a message to B, then B must deliver a message to A, then A must deliver a message to B and so on. If an entity is in a state where it should not attempt to deliver a message to the other side, and there is an event like rdt_send(data) call from above that attempts to pass data down for transmission to the other side, this call from above can simply be ignored with a call to rdt_unable_to_send(data) , which informs the higher layer that it is currently not able to send data. [Note: This simplifying assumption is made so you don't have to worry about buffering data.]

Draw a FSM specification for this protocol (one FSM for A, and one FSM for B!). Note that you do not have to worry about a reliability mechanism here; the main point of this question is to create a FSM specification that reflects the synchronized behavior of the two entities. You should use the following events and actions that have the same meaning as protocol rdt1.0 in Figure 3.9: rdt_send(data) , packet = make_pkt(data) ,udt_send(packet) , rdt_rcv(packet) , extract(packet,data) , deliver_data(data) . Make sure your protocol reflects the strict alternation of sending between A and B. Also, make sure to indicate the initial states for A and B in your FSM descriptions.

答：





P18. In the generic SR protocol that we studied in Section 3.4.4, the sender transmits a message as soon as it is available (if it is in the window) without waiting for an acknowledgment. Suppose now that we want an SR protocol that sends messages two at a time. That is, the sender will send a pair of messages and will send the next pair of messages only when it knows that both messages in the first pair have been received correctly. Suppose that the channel may lose messages but will not corrupt or reorder messages. Design an error-control protocol for the unidirectional reliable transfer of messages. Give an FSM description of the sender and receiver. Describe the format of the packets sent between sender and receiver, and vice versa. If you use any procedure calls other than those in Section 3.4 (for example, udt_send() , start_timer() , rdt_rcv() , and so on), clearly state their actions. Give an example (a timeline trace of sender and receiver) showing how your protocol recovers from a lost packet.

答：在我的解决方案中,发送方在接收到一对报文 ACK(seqnum 和 seqnum+1)后才开始发送下一对报文。分组的数据字段携带有 2 位的序列号，也就是说,有效的分组序列号是 0，1，2，3。(注：你应考虑为什么 1 位的只有 0 和 1 序号空间在下面的解决方案不能运行。) ACK 报文段携带它们正在确认的数据分组的序列号。

接收方和发送方的 FSM 如下图 2。注意发送状态记录：(Ⅰ)当前分组没有收到 ACKs；(Ⅱ)只收到序号为 seqnum 的分组的 ACK 或只收到序号为 seqnum+1 的分组的 ACK。在这个图中，我们假设 seqnum 由 0 起始，发送方已经发送第一对报文。发送者和接受者从丢包事件中恢复过来的时间序列图如下：

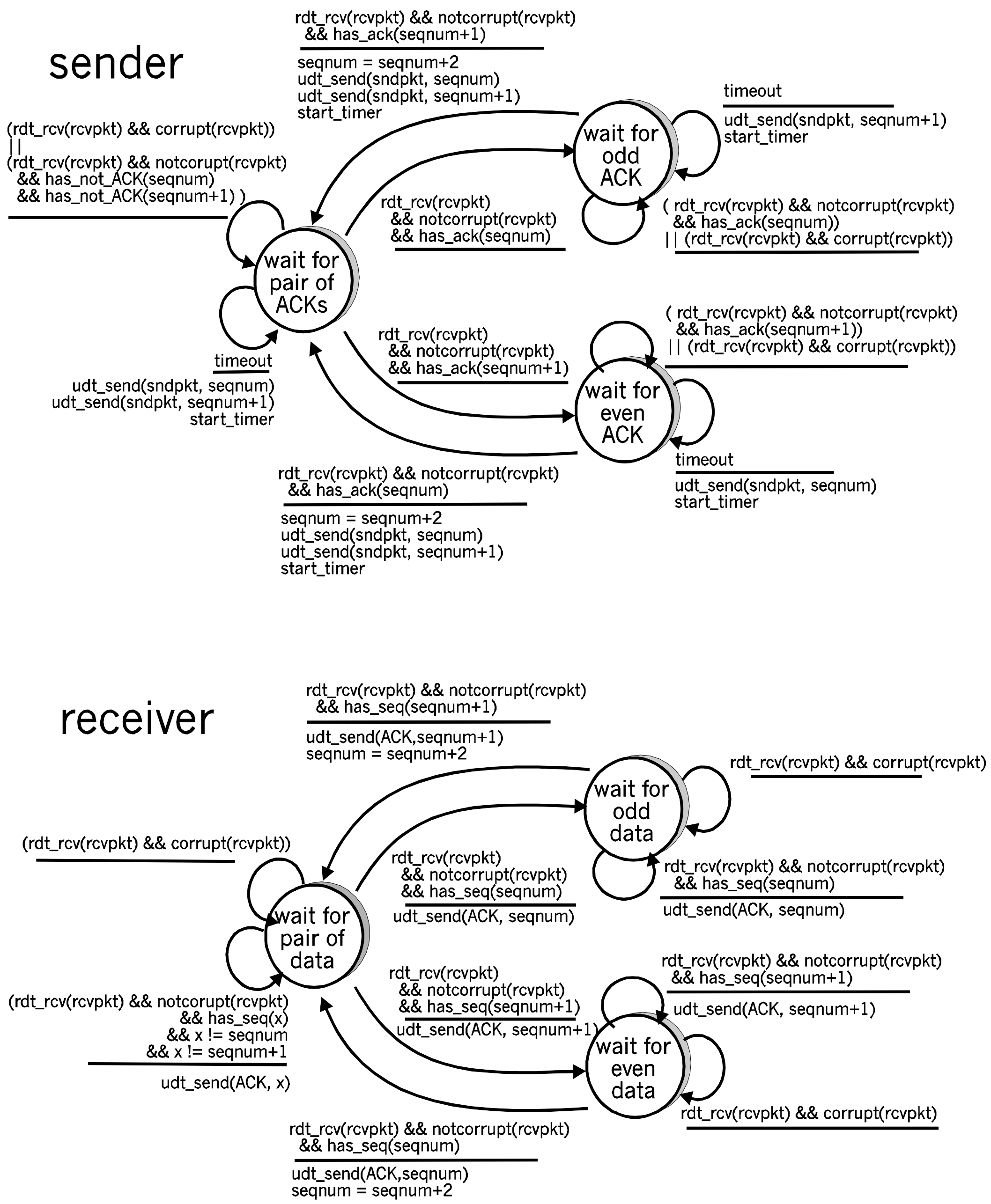


Figure 2: Sender and receiver for Problem (3.18)

Sender	Receiver
make pair (0,1)	
send packet 0	
Packet 0 drops	
send packet 1	
	receive packet 1
	buffer packet 1
	send ACK 1
receive ACK 1	
(timeout)	
resend packet 0	

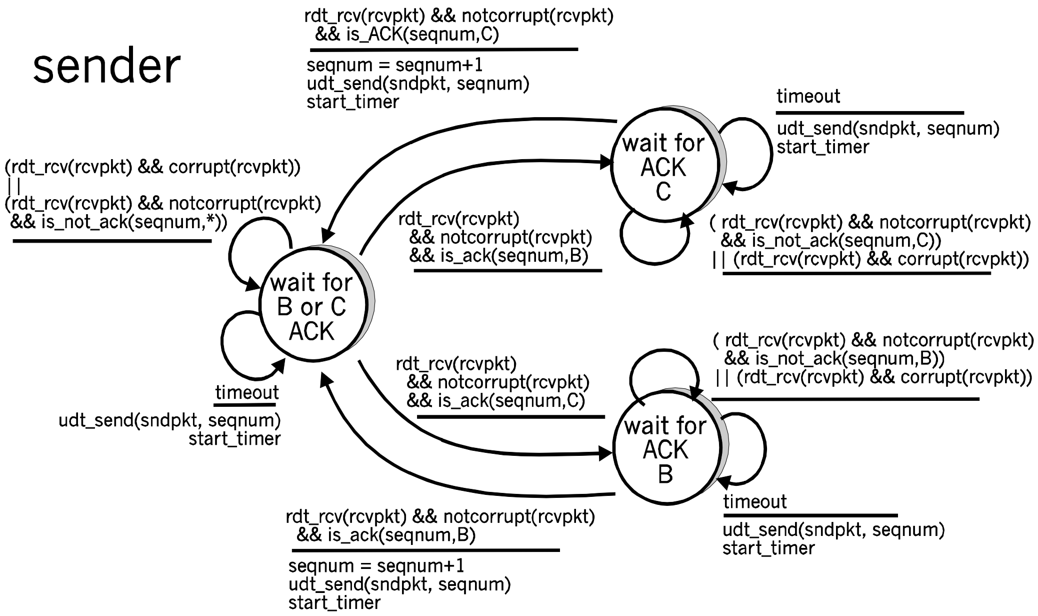
receive packet 0
deliver pair (0,1)
send ACK 0

receive ACK 0

P19. Consider a scenario in which Host A wants to simultaneously send packets to Hosts B and C. A is connected to B and C via a broadcast channel—a packet sent by A is carried by the channel to both B and C. Suppose that the broad- cast channel connecting A, B, and C can independently lose and corrupt packets (and so, for example, a packet sent from A might be correctly received by B, but not by C). Design a stop-and-wait-like error-control proto- col for reliably transferring packets from A to B and C, such that A will not get new data from the upper layer until it knows that both B and C have correctly received the current packet. Give FSM descriptions of A and C. (Hint: The FSM for B should be essentially the same as for C.) Also, give a description of the packet format(s) used.

答：这个问题是简单停等协议 rdt3.0 的变形。因为信道有可能丢失报文段，还因为在其中一个接收方已经接收到报文的情况下发送方也有可能重发报文。(要么是因为定时器超时过早发生，要么是因为另一个接收方还没有正确的接收数据)，所以序列号是必须的。就像是在 rdt3.0 中一样，在这里一个 0-bit 序列号是必须的。

发送方和接收方的 FSM 如下图 3。在这个问题中，发送方状态显示发送方是否只从 B 或 C 接收到 ACK，还是从 B 和 C 都没有接收到 ACK。接收方状态显示它等待哪一个序列号。



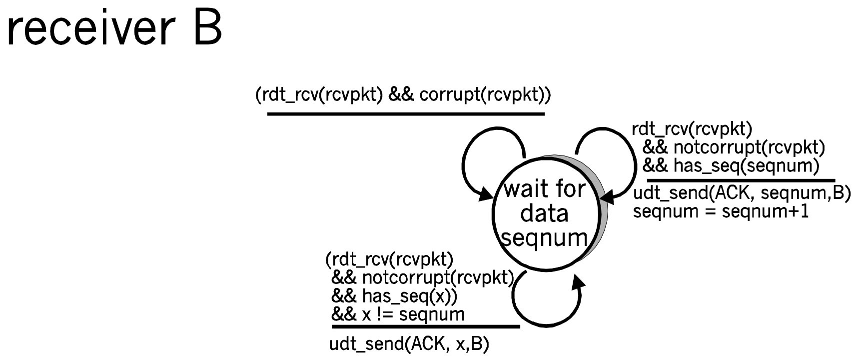


图 3　第 19 题的发送方和接收方

C 的 FSM 图中只需将 B 的 FSM 图中最下方 udt_send(ACK , x ,B)改为 udt_send(ACK , x , C)即可。

P20. Consider a scenario in which Host A and Host B want to send messages to Host C. Hosts A and C are connected by a channel that can lose and corrupt (but not reorder) messages. Hosts B and C are connected by another channel (independent of the channel connecting A and C) with the same proper- ties. The transport layer at Host C should alternate in delivering messages from A and B to the layer above (that is, it should first deliver the data from a packet from A, then the data from a packet from B, and so on). Design a stop-and-wait-like error-control protocol for reliably transferring packets from A and B to C, with alternating delivery at C as described above. Give FSM descriptions of A and C. (Hint: The FSM for B should be essentially the same as for A.) Also, give a description of the packet format(s) used.

答：发送方 FSM 和课本 3.15 一样，如下图：

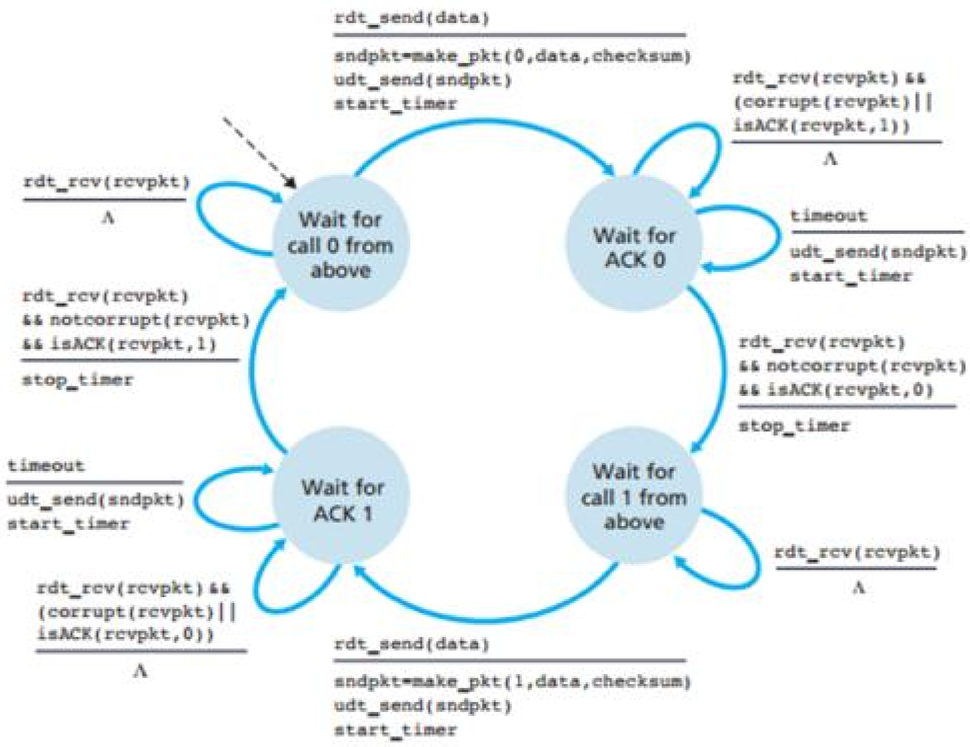


图 4　第 20 题的发送方 FSM

接收方 FSM 如下图：

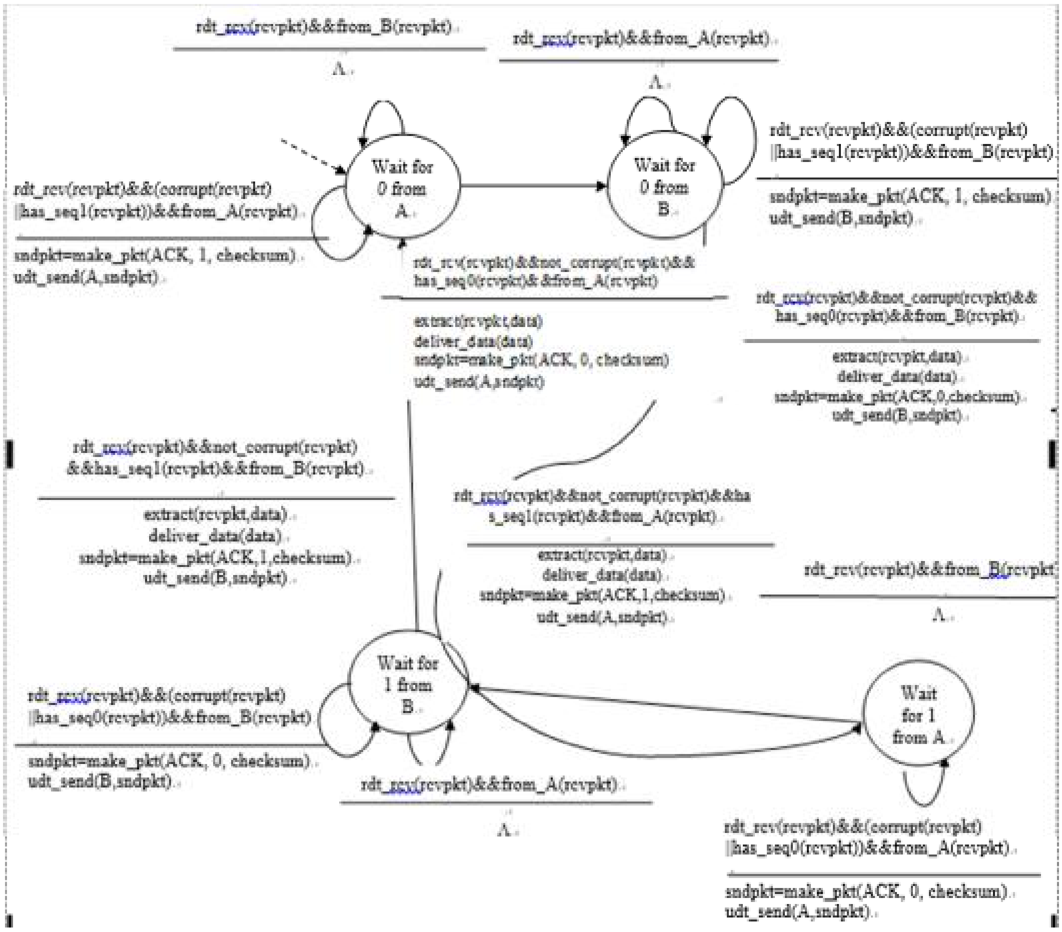


图 5　第 20 题的接收方 FSM

P21. Suppose we have two network entities, A and B. B has a supply of data messages that will be sent to A according to the following conventions. When A gets a request from the layer above to get the next data (D) message from B, A must send a request (R) message to B on the A-to-B channel. Only when B receives an R message can it send a data (D) message back to A on the B-to- A channel. A should deliver exactly one copy of each D message to the layer above. R messages can be lost (but not corrupted) in the A-to-B channel; D messages, once sent, are always delivered correctly. The delay along both channels is unknown and variable. Design (give an FSM description of) a protocol that incorporates the appropriate mechanisms to compensate for the loss-prone A-to-B channel and implements message passing to the layer above at entity A, as discussed above. Use only those mechanisms that are absolutely necessary.

答：因为 A-to-B 的信道可能丢失请求报文，A 将会检测到超时并且重传这些报文(以便能够修复丢失的分组)。因为信道的时延复杂多样并且未知，A 很可能发送重复的请求(也就是说，一个就已经被 B 接收到的报文段重发了一个请求)。为了能检测到这些重复的报文段，这个协议将实用序号。对于停等类型的请求相应协议而言，1 位的序号足以解决问题。

请求方 A 有四个状态

- “等待来自上层的调用 0” 状态。这里请求方在等待一个来自上层的调用，以便请求一个单元的数据。当它收到上层的调用时，它就发送一个请求报文 R0 到 B，启动计时器并且转变状态为“等待 D0”状态。当处于“等待来自上层的调用 0”状态时，A 忽略一切来自 B 的消息。
- “等待 D0”状态。这里请求方在等待一个来自 B 的 D0 数据报。 在这个状态时，A 的计时器一直运行。如果计时器到期，A 发送另一个 R0 报文，重启计时器并且维持在这个状态。如果 A 接收到来自 B 的 D0 报文，A 停止计时器并且将状态转变为“等待来自上层的调用 1”。处于这个状态的 A 如果收到 D1 数据报，它会忽略。
- “等待来自上层的调用 1”状态。这里请求方又一次在等待一个来自上层的调用，以便请求一个单元的数据。当它收到上层的调用时，它就发送一个请求报文 R1 到 B，启动计时器并且转变状态为“等待 D1”状态。当处于“等待来自上层的调用 1”状态时，A 忽略一切来自 B 的消息。
- “等待 D1”状态。这里请求方在等待一个来自 B 的 D1 数据报。 在这个状态时，A 的计时器一直运行。如果计时器到期，A 发送另一个 R1 报文，重启计时器并且维持在这个状态。如果 A 接收到来自 B 的 D1 报文，A 停止计时器并且将状态转变为“等待来自上层的调用 0”。如果处于这个状态的 A 如果收到 D0 数据报，它会忽略。

数据提供方 B 只有两个状态：

- “发送 0”状态。当 B 处于这个状态时，对于收到的 R0 报文它会持续通过发送 D0 作为响应，然后维持在这个状态。如果 B 收到一个 R1 报文，然后它就知道它的 D0 报文被 A 正确的接收。因此它会丢弃数据 D0(因为它已经被另一方接收到了)然后将状态转变为“发送 1”，然后在此状态下 B 将用 D1 来发送下一组请求的的数据。
- “发送 1”状态。当 B 处于这个状态时，对于收到的 R1 报文它会持续通过发送 D1 作为响应，然后维持在这个状态。如果 B 收到一个 R0 报文，然后它就知道它的 D1 报文被 A 正确的接收。因此它会将状态转变为“发送 0”。

P22. Consider the GBN protocol with a sender window size of 4 and a sequence number range of 1,024. Suppose that at time t, the next in-order packet that the receiver is expecting has a sequence number of k. Assume that the medium does not reorder messages. Answer the following questions:

a. What are the possible sets of sequence numbers inside the sender’s win- dow at time t? Justify your answer.

b. What are all possible values of the ACK field in all possible messages cur- rently propagating back to the sender at time t? Justify your answer.

答：假定接收方已收到分组$\mathrm{k}-1$，并且对$\mathrm{k}-1$以及之前的分组都发送 ACK 进行了确认，分为以下两种情况：

Ⅰ. 若发送方收到了分组$\mathrm{k}-1$的 ACK 确认，即确认号为$\mathrm{k}-1$。则新的发送窗口为$[k, k+N-1]$；

Ⅱ. 若发送方未收到分组$\mathrm{k}-1$的 ACK 确认，此时已收到的确认号为$\mathrm{k}-\mathrm{N}-1$。这时需要重传，发送窗口为$[k-N, k-1]$；

由Ⅰ和Ⅱ可得：

a. 发送窗口内可能的序号为：$[\mathrm{k}-\mathrm{N}, \mathrm{k}]$；

b. 已经收到的确认号 ACK 字段可能为$[\mathrm{k}-\mathrm{N}-1, \mathrm{k}-1]$，其中 N=3。

P23. Consider the GBN and SR protocols. Suppose the sequence number space is of size k. What is the largest allowable sender window that will avoid the occurrence of problems such as that in Figure 3.27 for each of these protocols?

答：为了避免图 3.27 中的问题，需要满足发送窗口最高序列号端和接收窗口最低序号端不交迭。假定接收方最低序号为 m，窗口大小为 w，则接收方窗口序号空间为$[\mathrm{m}, \mathrm{m}+\mathrm{w}-1]$。接收方对分组$(\mathrm{m}-1)$以及之前$(\mathrm{w}-1)$个分组向发送方发送 ACK 进行确认，若$[\mathrm{m}-\mathrm{w}, \mathrm{m}-1]$的 ACK 在发送方没有收到，发送方会重发这些分组，发送窗口为$[\mathrm{m}-\mathrm{w}, \mathrm{m}-1]$。由分析可知发送窗口的下界即最小值为$(\mathrm{m}-\mathrm{w})$接收窗口的上界即最大值为$(\mathrm{m}+\mathrm{w}-1)$，所以序号空间范围为：

$$\mathrm{Seq.space} \geq (m+w-1)-(m-w)+1=2w$$

所以，序号空间的大小至少是发送窗口的 2 倍，换言之，最大发送窗口不得不得大于序号空间大小的一半。

P24. Answer true or false to the following questions and briefly justify your answer:

a. With the SR protocol, it is possible for the sender to receive an ACK for a packet that falls outside of its current window.

b. With GBN, it is possible for the sender to receive an ACK for a packet that falls outside of its current window.

c. The alternating-bit protocol is the same as the SR protocol with a sender and receiver window size of 1.

d. The alternating-bit protocol is the same as the GBN protocol with a sender and receiver window size of 1.

答：a. 正确，假设发送方窗口大小为 3，在 t0 时刻发送分组 1，2，3。在 t1(t1>t0)时刻接收方确认 1，2，3。在 t2(t2>t1)时刻发送方计时器超时，重发 1,2,3.在 t3 时刻接收到重复的分组并重新确认 1，2，3。在 t4 时刻发送方接收到接收方在 t1 时刻发送的 ACK，并将其窗口前移到 4，5，6。在 t5 时刻发送方接收到接收方在 t2 发送的 ACK1，2，3。这些 ACK 是在当前窗口之外的报文的 ACK.

b. 正确，本质上同 a 中是一样的。

c. 正确。当窗口尺寸为 1 时,SR 与比特交替协议在功能上相同。

d. 正确。当窗口尺寸为 1 时，GBN 与比特交替协议在功能上相同。

P25. We have said that an application may choose UDP for a transport protocol because UDP offers finer application control (than TCP) of what data is sent in a segment and when.

a. Why does an application have more control of what data is sent in a segment?

b. Why does an application have more control on when the segment is sent?

答：a. 考虑用运输层协议发送一个应用层报文。 使用 TCP 时，应用程序写入数据到发送缓冲区，TCP 连接将把数据流分割成可运输地单元，把它们编号，然后逐个发送，运输层在接收端等待属于同一个进程的所有不同单元的到达，检查并放过那些没有差错的单元并以流的方式把它们交付给接收进程；而 UDP 将在一个报文段中封装给它的任何应用程序数据，此消息将成为 UDP 有效载荷部分的，即使有差错也会传输。所以应用层的程序将对 UDP 报文段中发送什么数据有更多的控制。

b. 使用 TCP，由于流量控制和拥塞控制，从一个应用程序将数据写入发送缓冲器，直到其数据发送到网络层可能有明显的延迟，因此相对 TCP 来说，报文段的传输过程中，应用程序控制的要少。而 UDP 不会出现因流量控制和拥塞控制产生时延。

P26. Consider transferring an enormous file of L bytes from Host A to Host B. Assume an MSS of 536 bytes.

a. What is the maximum value of L such that TCP sequence numbers are not exhausted? Recall that the TCP sequence number field has 4 bytes.

b. For the L you obtain in (a), find how long it takes to transmit the file.

Assume that a total of 66 bytes of transport, network, and data-link header are added to each segment before the resulting packet is sent out over a 155 Mbps link. Ignore flow control and congestion control so A can pump out the segments back to back and continuously.

答：a. 因为 TCP 序号是没个报文段首个字节的字节流编号，L 最大值即序号空间最大值，由 TCP 的序号字段是 4 字节 32 位。能存储2^{32}个数据，则 L 的最大值为：

$$\mathrm{L}_{max} = 2^{32} = 4294967296 bytes \approx 4.29 Gbytes$$

b. 报文段的数量为：

$$\frac{2^{32}}{536} \approx 8012999 \text{ 个}$$

首部长度为 66 字节，则首部所占字节数为：

$$8012999 \times 66 = 528857934 \text{bytes}$$

要传输的总的字节数为：

$$2^{32} + \frac{2^{32}}{536} \times 66 \approx 4.824 \times 10^9 bytes$$

用 155Mbps 的链路来传输时，需要的时间为：

$$\frac{4.824 \times 10^9 bytes}{155 Mbps} = \frac{4.824 \times 10^9 \times 8 bits}{155 \times 10^6 bps} \approx 249 s$$

P27. Host A and B are communicating over a TCP connection, and Host B has already received from A all bytes up through byte 126. Suppose Host A then sends two segments to Host B back-to-back. The first and second segments contain 80 and 40 bytes of data, respectively. In the first segment, the sequence number is 127, the source port number is 302, and the destination port number is 80.

Host B sends an acknowledgment whenever it receives a segment from Host A.

a. In the second segment sent from Host A to B, what are the sequence number, source port number, and destination port number?

b. If the first segment arrives before the second segment, in the acknowledgment of the first arriving segment, what is the acknowledgment number, the source port number, and the destination port number?

c. If the second segment arrives before the first segment, in the acknowledgment of the first arriving segment, what is the acknowledgment number?

d. Suppose the two segments sent by A arrive in order at B. The first acknowledgment is lost and the second acknowledgment arrives after the first time- out interval. Draw a timing diagram, showing these segments and all other segments and acknowledgments sent. (Assume there is no additional packet loss.) For each segment in your figure, provide the sequence number and the number of bytes of data; for each acknowledgment that you add, pro- vide the acknowledgment number.

答:a. 在从主机 A 发往 B 的第二个报文段中，序号为 207，源端口号为 302，目的端口号为 80。

b. 如果第一个报文段在第二个报文段之前到达，在第一个报文段的 ACK 确认中，确认号为 207，源端口号为 80，目的端口号为 302。

c. 如果第二个报文段在第一个报文段之前到达，在第一个已经到达的报文段(即第二个报文段)的确认中，因为发送的第一个报文段还没有收到，所以接收方 B 请求重传，而确认号为期待接收的报文段的第一个序号，所以确认号为 127。

d. 如下图：

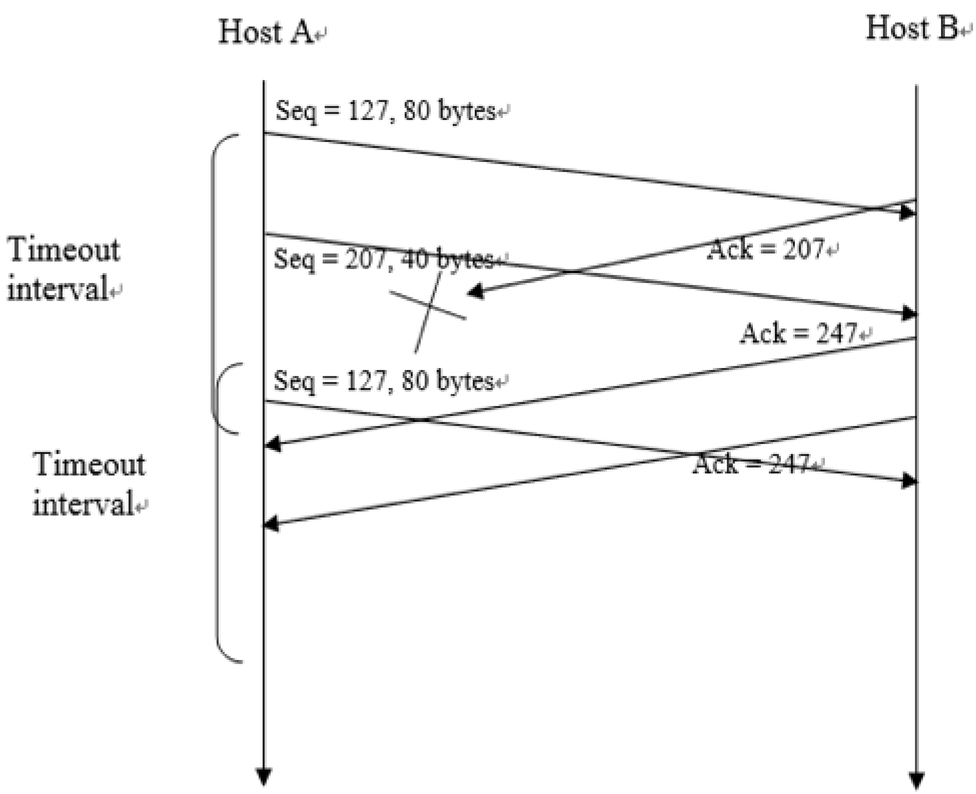


图 6 27.d 的时序图

P28. Host A and B are directly connected with a 100 Mbps link. There is one TCP connection between the two hosts, and Host A is sending to Host B an enormous file over this connection. Host A can send its application data into its TCP socket at a rate as high as 120 Mbps but Host B can read out of its TCP receive buffer at a maximum rate of 50 Mbps. Describe the effect of TCP flow control.

答：由于链路最大带宽是 100Mbps，因此主机 A 最大发送速率为 100Mbps。主机 A 向接收缓冲发送数据的速率要比主机 B 将数据从缓冲中取出的快。接收方缓存在满载时的速率大约为 40Mbps。此时，主机 B 通过将接收窗口值设置为 0，向主机 A 发送信号，让其停止发送数据。当主机 A 收到一个告知接收窗口大于 0 的 TCP 报文段时继续向主机 B 发送数据。主机 A 就是这样根据来自主机 B 发送的接收窗口的值不停的停止或发送。平均下来，主机 A 向主机 B 发送数据的速率不会超过 60Mbps。

P29. SYN cookies were discussed in Section 3.5.6.

a. Why is it necessary for the server to use a special initial sequence number in the SYNACK?

b. Suppose an attacker knows that a target host uses SYN cookies. Can the attacker create half-open or fully open connections by simply sending an ACK packet to the target? Why or why not?

c. Suppose an attacker collects a large amount of initial sequence numbers sent by the server. Can the attacker cause the server to create many fully open connections by sending ACKs with those initial sequence numbers? Why?

答：a. 服务器通过使用特殊的初始序号(从哈希表得到的源和目的 IP 地址以及源和目的端口号)以便让它本身拒绝 SUN 洪泛攻击。

b. 不能，如果攻击者仅仅通过发送 ACK 分组到靶机的方式，是不能创建半开放或者完全开放的 TCP 连接。半开放的连接是不可能出现的，因为在服务器使用的 SYN cookies 中，在连接建立之前，它没有包含连接变量和缓存。对于建立完全开放的连接，攻击者应当了解和(欺骗的)IP 地址相对应的初始序号。这个序号要求每一个服务器使用“secret”编号。由于攻击者不知道这个“secret”编号，他无法猜测到初始序号。

c. 不能，服务器只能增加一个计算初始序号的时间戳并选择为那些序号选择一个生存期，并且丢弃过期的初始序号即便攻击者重新使用它们。

P30. Consider the network shown in Scenario 2 in Section 3.6.1. Suppose both sending hosts A and B have some fixed timeout values.

a. Argue that increasing the size of the finite buffer of the router might possibly decrease the throughput (λout).

b. Now suppose both hosts dynamically adjust their timeout values (like what TCP does) based on the buffering delay at the router. Would increasing the buffer size help to increase the throughput? Why?

答：a. 如果超时的值被固定，那么发送方可能提前超时。因此，一些分组即便它们没有丢失也会重传。

b. 如果超时的值是被估算的(像 TCP 做的一样)，那么增加缓存的大小毫无疑问可以增加路由器的吞吐率。但是可能存在一个潜在的问题，排队时延可能非常延长和情景 1 显示的类似。

P31. Suppose that the five measured SampleRTT values (see Section 3.5.3) are 106 ms, 120 ms, 140 ms, 90 ms, and 115 ms. Compute the EstimatedRTT after each of these SampleRTT values is obtained, using a value of $\alpha = 0.125$ and assuming that the value of EstimatedRTT was 100 ms just before the first of these five samples were obtained. Compute also the DevRTT after each sample is obtained, assuming a value of = 0.25 and assuming the value of DevRTT was 5 ms just before the first of these five samples was obtained. Last, compute the TCP TimeoutInterval after each of these samples is obtained.

答：如下：

$$EstimatedRTT = xSampleRTT + (1-x)EstimatedRTT$$

$$DevRTT = y\left|SampleRTT - EstimatedRTT\right| + (1-y)DevRTT$$

$$TimeoutInterval = EstimatedRTT + 4 * DevRTT$$

After obtaining first sampleRTT is:

$$EstimatedRTT = 0.125*106 + 0.875*100 = 100.75ms$$

$$DevRTT = 0.25*\left|106 - 100.75\right| + 0.75*5 = 5.06ms$$

$$TimeoutInterval = 100.75 + 4*5.06 = 120.99ms$$

After obtaining second sampleRTT = 120ms:

$$EstimatedRTT = 0.125*120 + 0.875*100.75 = 103.15ms$$

$$DevRTT = 0.25*\left|120 - 103.15\right| + 0.75*5.06 = 8ms$$

$$TimeoutInterval = 103.15 + 4*8 = 135.15ms$$

After obtaining Third sampleRTT = 140ms:

$$EstimatedRTT = 0.125*140 + 0.875*103.15 = 107.76ms$$

$$DevRTT = 0.25*\left|140 - 107.76\right| + 0.75*8 = 14.06ms$$

$$TimeoutInterval = 107.76 + 4*14.06 = 164ms$$

After obtaining fourth sampleRTT = 90ms:

$$EstimatedRTT = 0.125*90 + 0.875*107.76 = 105.54ms$$

$$DevRTT = 0.25*\left|90 - 105.54\right| + 0.75*14.06 = 14.42ms$$

$$TimeoutInterval = 105.54 + 4*14.42 = 163.22ms$$

After obtaining fifth sampleRTT = 115ms:

$$EstimatedRTT = 0.125*115 + 0.875*105.54 = 106.71ms$$

$$DevRTT = 0.25*\left|115 - 106.71\right| + 0.75*14.42 = 12.88ms$$

$$TimeoutInterval = 106.71 + 4*12.88 = 158.23ms$$

P32. Consider the TCP procedure for estimating RTT. Suppose that a = 0.1. Let SampleRTT1 be the most recent sample RTT, let SampleRTT2 be the next most recent sample RTT, and so on.

a. For a given TCP connection, suppose four acknowledgments have been returned with corresponding sample RTTs: SampleRTT4, SampleRTT3, SampleRTT2, and SampleRTT1. Express EstimatedRTT in terms of the four sample RTTs.

b. Generalize your formula for n sample RTTs.

c. For the formula in part (b) let n approach infinity. Comment on why this averaging procedure is called an exponential moving average.

答：a. 用$\text{EstimatedRTT}^{(n)}$表示得到第 n 个样本后的估测。则：

$$\text{EatimatedRTT}^{(4)} = xSampleRTT_1 + (1-x)[xSampleRTT_2 + (1-x)[xSampleRTT_3 + (1-x)SampleRTT_4]]$$

$$= xSampleRTT_1 + (1-x)xSampleRTT_2 + (1-x)^2 xSampleRTT_3 + (1-x)^3 xSampleRTT_4$$

b.如下：

$$\text{EstimatedRTT}^{(n)} = [\mathrm{x}\sum_{j=1}^{n-1}(1-x)^{j-1}SampleRTT_j] + (1-x)^{n-1}SampleRTT_n$$

c. 如下：

$$\text{EstimatedRTT}^{(\infty)} = \frac{\mathrm{x}}{1-x}\sum_{j=1}^{\infty}(1-x)^j SampleRTT_j$$

$$= \frac{1}{9}\sum_{j=1}^{\infty}0.9^j SampleRTT_j$$

所以这个平均过程被称为指数移动平均。

P33. In Section 3.5.3, we discussed TCP’s estimation of RTT. Why do you think TCP avoids measuring the SampleRTT for retransmitted segments?

答：让我们研究一下如果 TCP 为重传报文的计算 SampleRTT 会出现什么情况。假设源主机发送分组 P1，P1 的计时器超时，源接着发送同一个分组的一个新的拷贝 P2。进一步假设源主机为 P2 计算 SampleRTT(重传的分组)。最后假设在源主机将 P2 传输出去很短时间内收到了 P1 的确认号。源主机将会错误的将这个确认号作为 P2 的确认号并且会将 SampleRTT 计算得出错误的值。

P34. What is the relationship between the variable SendBase in Section 3.5.4 and the variable LastByteRcvd in Section 3.5.5?

答：SendBase：是最近未被确认的字节的序号，SendBase-1 是接收方已正确按序接收到数据的最后一个字节的序号。LastByteRcvd：从网络中到达的并且已经放入主机 B 接收缓存中的数据流最后一个字节的编号。在任意给定时刻 t，SendBase-1 是发送方知道的已经被接收方正确的按序接收的最后一个比特的序列号，在 t 时刻被接收方(正确的和按序的)接收到的真正的最后一个 byte 要比在链路上传输的 ACK 要大，即：

$$(\text{SendBase} - 1) \le \text{LastByteRcvd}$$

P35. What is the relationship between the variable LastByteRcvd in Section 3.5.5 and the variable y in

Section 3.5.4?

答：y 是指发送方接收到的最新 ACK 的值在 t 时刻，发送方接收到的 ACK 的值为 y，据此发送方可以确认接收方已经接收了序号到 y-1 的数据。如果 $y \leq \text{SendBase}$ 或在线路上有其他的 ACK，在 t 时刻接收方(正确的和按序的)接收到的真正的最后一个 byte 要比 $y-1$大。所以 $y-1 \leq \text{LastByteRvcd}$。

P36. In Section 3.5.4, we saw that TCP waits until it has received three duplicate ACKs before performing a fast retransmit. Why do you think the TCP designers chose not to perform a fast retransmit after the first duplicate ACK for a segment is received?

答：是为了降低重传的概率。假如有 1,2,3 分组，当 1 号分组成功接收后，接着 2,3 颠倒，3 号分组被传送，此时则会产生一个 1 号分组冗余 ACK，若根据“收到一个冗余 ACK 重传”的规则，则会重传 1 号分组，但若根据“收到三个相同 ACK 信息重传”的规则，则在第三次收到 2 号分组时避免了重传，如此一来，大大降低了重传概率。

P37. Compare GBN, SR, and TCP (no delayed ACK). Assume that the timeout values for all three protocols are sufficiently long such that 5 consecutive data segments and their corresponding ACKs can be received (if not lost in the channel) by the receiving host (Host B) and the sending host (Host A) respectively. Suppose Host A sends 5 data segments to Host B, and the 2nd segment (sent from A) is lost. In the end, all 5 data segments have been correctly received by Host B.

a. How many segments has Host A sent in total and how many ACKs has Host B sent in total? What are their sequence numbers? Answer this question for all three protocols.

b. If the timeout values for all three protocol are much longer than 5 RTT, then which protocol successfully delivers all five data segments in shortest time interval?

答：a. 回退 N 步协议(GoBackN)：

A 一共发送了 9 个报文段。初始的报文段是 1，2，3，4，5，由于报文段 2 丢失要回到第二个报文段重新发送，所以后来的重传报文段是 2，3，4，5；

B 一共发送了 8 个 ACK。它对序号 1 进行了 4 次确认，对序号 2，3，4，5 各发送了一次一共 4 次 ACK 确认。

选择重传协议(Selective Repeat)：

A 一共发送了 6 个报文段。他们的初始报文段是 1，2，3，4，5 以及后来重传的报文段 2；

B 发送了 5 个 ACK。先是序号为 1，3，4，5 报文段的 4 个 ACK 以及后来重传报文段 2 的 ACK。

TCP 协议：

A 一共发送了 6 个报文段。他们的初始报文段是 1，2，3，4，5 以及后来重传的报文段 2；

B 发送了 5 个 ACK。由于报文段 2 丢失首先被 B 成功接收的 1，2，3，5 的 4 个 ACK 都是 2，它表示 B 期待下一个接收到 2。当报文段 2 被正确接收以后，B 发送的 ACK 为 6，表示 B 期待 A 下一次发送 6 号报文段，所以一共发送了 5 个 ACK。注意，TCP 的 ACK 若为 N，表示他之前 N-1 个报文段已接收到，期待 A 下一次发送报文段 N。

b. TCP 传输最快，因为 TCP 传输最快，不需要等待超时。

P38. In our description of TCP in Figure 3.53, the value of the threshold, ssthresh, is set as ssthresh=cwnd/2 in several places and ssthresh value is referred to as being set to half the window size when a loss event occurred. Must the rate at which the sender is sending when the loss event occurred be approximately equal to cwnd segments per RTT? Explain your answer. If your answer is no, can you suggest a different manner in which ssthresh should be set?

答：是的，发送速率总是约为cwnd/RTT。

P39. Consider Figure 3.46(b). If λ'in increases beyond R/2, canλout increase beyond R/3? Explain. Now consider Figure 3.46(c). Ifλ'in increases beyond R/2, canλout increase beyond R/4 under the assumption that a packet will be forwarded twice on average from the router to the receiver? Explain.

答：如果在图 3.46(b)中到达速率超过了R/2，然后对于队列而言总的到达速率超过了队列所能允许的最大值，会导致在到达速率增大的过程中丢包率增加。当到达速率等于R/2时，离开队列的分组中有超过三分之一的分组会重传。随着丢包率的增加，离开队列的分组中会有更大比例的将要重传。对于这种情形，给定从一个队列中最大的离开队列速率为R/2；并且给定随着到达速率都增加，有至少三分之一的分组需要传输，那么成功传输数据的吞吐率不可能增加到超过 $_{out}$。接下来是相似的理由，如果离开队列的分组中有一半在传输，那么每一个周期最大输出分组率为R/2，所以 $_{out}$ 的最大值为$(\frac{R}{2})/2$即R/4。

P40. Consider Figure 3.58. Assuming TCP Reno is the protocol experiencing the behavior shown above, answer the following questions. In all cases, you should provide a short discussion justifying your answer.

a. Identify the intervals of time when TCP slow start is operating.

b. Identify the intervals of time when TCP congestion avoidance is operating.

c. After the 16th transmission round, is segment loss detected by a triple duplicate ACK or by a timeout?

d. After the 22nd transmission round, is segment loss detected by a triple duplicate ACK or by a timeout?

e. What is the initial value of ssthresh at the first transmission round?

f. What is the value of ssthresh at the 18th transmission round?

g. What is the value of ssthresh at the 24th transmission round?

h. During what transmission round is the 70th segment sent?

i. Assuming a packet loss is detected after the 26th round by the receipt of a triple duplicate ACK, what will be the values of the congestion window size and of ssthresh?

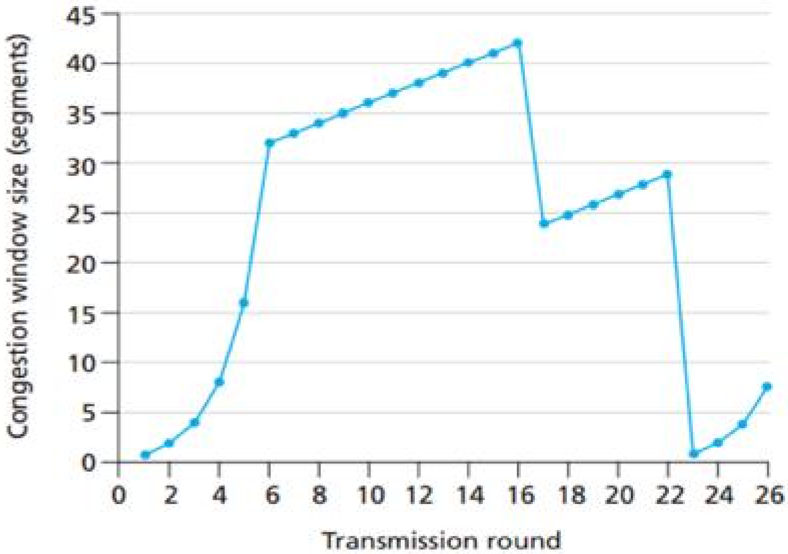


Figure 3.58 ♦ TCP window size as a function of time

j. Suppose TCP Tahoe is used (instead of TCP Reno), and assume that triple duplicate ACKs are received at the 16th round. What are the ssthresh and the congestion window size at the 19th round?

k. Again suppose TCP Tahoe is used, and there is a timeout event at 22nd round. How many packets have been sent out from 17th round till 22nd round, inclusive?

答：a. TCP 慢启动时，发送分组数量呈指数增加，其时间间隔为[1,6]和[23,26]。

b. TCP 进入拥塞避免时，发送分组数量呈线性增加，所以其时间间隔为[6,16]和[17,22]。

c. 在 16 个周期后，拥塞窗口大小减半，由此得知是根据三个重复确认检测到分组丢失。

d. 在 22 个周期后，拥塞窗口大小减为 1，由此得知是根据超时检测出分组丢失。

e. 在第一个传输周期，Threshold(阈值)的初始值为 32，表示慢启动阶段的结束和拥塞避免阶段的开始。

f. 在第 16 个周期后，由于通过三个重复确认检测到报文段丢失，所以 Threshold(阈值)和窗口大小都减小为当前窗口大小 42 的一半为 21，所以在第 18 个周期时 Threshold(阈值)为 21。

g. 在第 22 个周期后，由于通过超时检测到报文段丢失，所以 Threshold(阈值)减小为当前窗口大小 26 的一半为 13，窗口大小减小为 1，都所以在第 18 个周期时 Threshold(阈值)为 13。

h. 从第 1 带第 7 个周期发送的报文段分别为：1,2,4,8,16,32,33，由

$$1+2+4+8+16+32 < 70 < 1+2+4+8+16+32+33$$

得知第 70 个分组在第 7 个周期内发送。

i. 在第 26 个周期的窗口大小为 8，由通过三个冗余的 ACK 检测到分组丢失，则 Threshold(阈值)和窗口大小都减小为当前窗口大小的一半为 4。

j. 当使用 TCP Tahoe 时无论是通过三个重复的 ACK 还是通过超时检测出分组丢失，窗口大小都减小为 1，由于是通过三个重复 ACK 检测到分组丢失，Threshold(阈值)由 42 减小为 21。

k. 由 17 周期，1 个分组；18 周期，2 个分组；19 周期，4 个分组；20 周期，8 个分组；21 周期，16 个分组；22 周期，21 分组。因此分组总的数量为 52。

P41. Refer to Figure 3.56, which illustrates the convergence of TCP’s AIMD algorithm. Suppose that instead of a multiplicative decrease, TCP decreased the window size by a constant amount. Would the resulting AIAD algorithm converge to an equal share algorithm? Justify your answer using a diagram similar to Figure 3.56.

答：参考图 5。在图 5(a)中，在分组丢失时连接 1 和连接 2 以同样的比例线性减少，正如以同样的比例线性增加：具有同一性。在这样的情况下，总的吞吐量永远不会离开 AB 这条线。在图 5(b)中，在分组丢失时连接 1 和连接 2 之间线性减少的比例为 2:1。那就是，无论何时有分组丢失，连接减小窗口大小是连接 2 的 2 倍。我们最终看到，在丢失足够多分组之后，加上随后的递增，连接 1 的吞吐量将趋近于 0，整个链路带宽将定位于连接 2，被连接 2 独用。

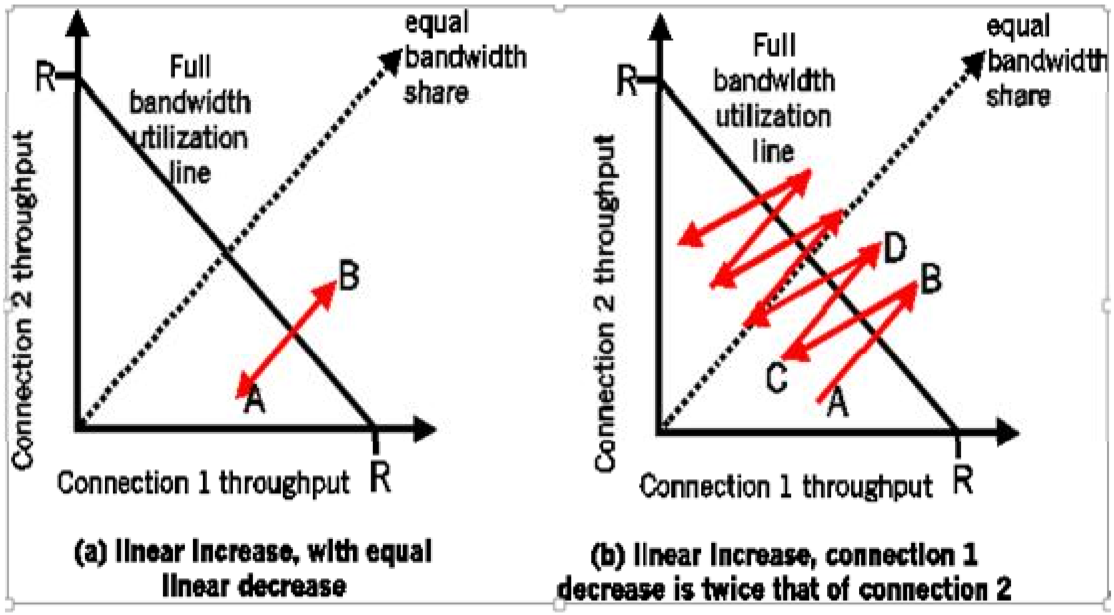


Figure 5: Lack of TCP convergence with linear increase, linear decrease

P42. In Section 3.5.4, we discussed the doubling of the timeout interval after a timeout event. This mechanism is a form of congestion control. Why does TCP need a window-based congestion-control mechanism (as studied in Section 3.7) in addition to this doubling-timeout-interval mechanism?

答：如果 TCP 是停等协议，那么加倍超时间隔将被称为拥塞控制机制。然而，TCP 使用流水线(因此它并非停等协议)，这将允许发送方有多个未完成确认的报文段(就继续传输)。加倍超时间隔不会阻止 TCP 的发送方发送大量的第一时间传输的分组到网络中，即便端到端的路径处于高度拥塞状态下。因此，基于窗口大小的拥塞控制机制是必要的，当出现网络拥塞的标志时，这个机制便阻止来自上层应用层的数据流继续发送到网络。

P43. Host A is sending an enormous file to Host B over a TCP connection. Over this connection there is never any packet loss and the timers never expire. Denote the transmission rate of the link connecting Host A to the Internet by R bps. Suppose that the process in Host A is capable of sending data into its TCP socket at a rate S bps, where S = 10 ? R. Further suppose that the TCP receive buffer is large enough to hold the entire file, and the send buffer can hold only one percent of the file. What would prevent the process in Host A from continuously passing data to its TCP socket at rate S bps? TCP flow control? TCP congestion control? Or something else? Elaborate.

答：在这个问题中，在接收方 B 没有溢出危险因为接收方的缓存可以接收整个文件。并且在计时器超时之前，没有分组丢失和确认号被返回，TCP 拥塞控制不会限制发送方。然而，主机 A 中的进程不会持续的向套接字发送数据发送方缓存很快会被填满。一旦发送方缓存被填满，进程将以速率 R<<S 来传输数据。所以需要 TCP 流量控制。

P44. Consider sending a large file from a host to another over a TCP connection that has no loss.

a. Suppose TCP uses AIMD for its congestion control without slow start. Assuming cwnd increases by 1 MSS every time a batch of ACKs is received and assuming approximately constant round-trip times, how long does it take for cwnd increase from 6 MSS to 12 MSS (assuming no loss events)?

b. What is the average throughout (in terms of MSS and RTT) for this connection up through time = 6 RTT?

答：a. CongWin 增至 6MSS 需要 1 个 RTT，此时进入拥塞避免阶段，经过 2RTT 增至 7MSS，经过 3RTT 增至 8MSS，经过 4RTT 增至 9MSS，经过 5RTT 增至 10MSS，经过 6RTT 增至 11MSS，经过 7RTT 增至 12MSS。

b. 在第 1 个 RTT 发送了 5 个 MSS 报文段，在第 2 个 RTT 发送了 6 个 MSS，在第 3 个 RTT 发送了 7 个 MSS，在第 4 个 RTT 发送了 8 个 MSS，在第 5 个 RTT 发送了 9 个 MSS，在第 6 个 RTT 发送了 10 个 MSS。因此到第 6 个 RTT 为止，一共发送并且被确认了$5+6+7+8+9+10 = 45$ MSS，所以我们可以得出在这 6 个 RTT 内平均吞吐率为：

$$\frac{45MSS}{6RTT} = 7.5MSS/\mathrm{RTT}$$

P45. Recall the macroscopic description of TCP throughput. In the period of time from when the connection's rate varies from W/(2 ? RTT) to W/RTT, only one packet is lost (at the very end of the period).

a. Show that the loss rate (fraction of packets lost) is equal to

$$\mathrm{L} = \text{loss rate} = \frac{1}{\frac{3}{8}w^2 + \frac{3}{4}w}$$

b. Use the result above to show that if a connection has loss rate L, then its average rate is approximately given by

$$\frac{1.22 \times MSS}{RTT \times \sqrt{L}}$$

答：a. 丢包率 L，是丢失的包数量与发送包数量的比率。在一个循环中，有 1 个包丢失。在这个循环中发送的包数是：

$$\frac{w}{2} + \left(\frac{w}{2}+1\right) + \cdots \mathrm{w} = \sum_{n=0}^{w/2}\left(\frac{w}{2}+n\right)$$

$$= \left(\frac{w}{2}+1\right)\frac{w}{2} + \sum_{n=0}^{w/2} n$$

$$= \left(\frac{w}{2}+1\right)\frac{w}{2} + \frac{w/2(\frac{w}{2}+1)}{2}$$

$$= \frac{w^2}{4} + \frac{w}{2} + \frac{w^2}{8} + \frac{w}{4}$$

$$= \frac{3}{8}w^2 + \frac{3}{4}w$$

所以丢包率为：

$$\mathrm{L} = \frac{1}{\frac{3}{8}w^2 + \frac{3}{4}w}$$

b. 因为 w 很大，所以$\frac{3}{8}\mathrm{w}^2 \gg \frac{3}{4}w$，则$\mathrm{L} = \frac{8}{3w^2}$，或者$\mathrm{w} = \sqrt{\frac{8}{3L}}$。由 TCP 吞吐量在两极值之间线性增长有

$$\text{TCP 连接平均吞吐量} = \frac{0.75W}{RTT}$$

代入 W 有

$$\text{平均吞吐量} = \frac{3}{4} \times \sqrt{\frac{8}{3L}} \times \frac{MSS}{RTT} \approx \frac{1.22MSS}{RTT}$$

P46. Consider that only a single TCP (Reno) connection uses one 10Mbps link which does not buffer any data. Suppose that this link is the only congested link between the sending and receiving hosts. Assume that the TCP sender has a huge file to send to the receiver, and the receiver's receive buffer is much larger than the congestion window. We also make the following assumptions: each TCP segment size is 1,500 bytes; the two-way propagation delay of this connection is 150 msec; and this TCP connection is always in congestion avoidance phase, that is, ignore slow start.

a. What is the maximum window size (in segments) that this TCP connection can achieve?

b. What is the average window size (in segments) and average throughput (in bps) of this TCP connection?

c. How long would it take for this TCP connection to reach its maximum window again after recovering from a packet loss?

答：a. 用 W 表示在报文段中测量到的最大窗口大小。那就有：

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

如果最大的传输速率超出了链路的吞吐量分组会丢失。因此，我们有：

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

解得 W 约为 125 个报文段大小。

b. 拥塞窗口大小从 W/2 变化到 W，那么平均窗口大小为 0.75W=94(93.75 的近似值)，平均吞吐量为：

$$94 \times 1500 \times \frac{8}{0.15} = 7.52\mathrm{Mbps}$$

c. $\frac{94}{2} \times 0.15 = 7.05$seconds，由于 RTT 的数量给定为 W/2。注意在每一个 RTT 中窗口大小在增加。

P47. Consider the scenario described in the previous problem. Suppose that the 10Mbps link can buffer a finite number of segments. Argue that in order for the link to always be busy sending data, we would like to choose a buffer size that is at least the product of the link speed C and the two-way propagation delay between the sender and the receiver.

答：用 W 表示最大的中窗口大小。用 S 表示缓存大小。为了方便，假定 TCP 发送方在一个又一个传输周期内发送分组，每一个周期对应一个 RTT。如果窗口的大小达到了 W，那么就会出现分组丢失。然后发送方拥塞窗口大小减半，在它再次发送数据报文段之前，发送方在等待 W/2 未完成传输分组的 ACK。为了确保链路一直在传输数据，在时间段0 到 $\mathrm{W}/(2\cdot\mathrm{C})$我们要让链路不停的传输数据(在这个时间间隔内，发送方在一直等待未完成传输的 W/2 分组的 ACK)。因此S/C一定不能超过$\mathrm{W}/(2\cdot\mathrm{C})$，即$\mathrm{S}\geq\mathrm{W}/2$。

用T_p表示在发送方和接收方的单向传播时延。当窗口大小达到最小值 W/2 之前缓存一直为空，我们需要确保链路在一直不停的传输数据，因此，我们必须保证：

$$\frac{\mathrm{W}}{2}/(2T_p)\geq\mathrm{C}$$

所以，$(\mathrm{W}/2)\geq\mathrm{C}\times2T_p$，即$\mathrm{S}\geq\mathrm{C}\times2T_p$。

P48. Repeat Problem 43, but replacing the 10 Mbps link with a 10 Gbps link. Note that in your answer to part c, you will realize that it takes a very long time for the congestion window size to reach its maximum window size after recovering from a packet loss. Sketch a solution to solve this problem.

答：a. 用 W 表示最大的窗口大小，那么有：$\mathrm{W}*\mathrm{MSS}/\mathrm{RTT}=10\mathrm{Gbps}$，如果发送速率超出了链路的吞吐量，分组将会丢失。所以我们有：

$$\mathrm{W}\times1500\times\frac{8}{0.15}=10\times10^9$$

解得$\mathrm{W}=125000$个报文段

b. 当接收窗口从 W/2 变化到 W，平均窗口大小为：0.75W=93750 个报文段，平均吞吐量为：

$93750\times1500\times\frac{8}{0.1}=7.5\mathrm{Gbps}$。

c. (93750/(2×0.15))/60=117 分钟，为了加快窗口增大速率，我们可以增加进程数量，这样我们可以得到一个更大的窗口大小值，而不是用在每一个 RTT 内窗口大小增 1 的办法。一些协议可以解决这个问题，比如 ScalableTCP 协议或者 HighSpeed TCP 协议。

P49. Let T (measured by RTT) denote the time interval that a TCP connection takes to increase its congestion window size from W/2 to W, where W is the maximum congestion window size. Argue that T is a function of TCP's aver- age throughput.

答：我们用 B 表示 TCP 连接的平均吞吐率，则：

$$\mathrm{B}=\frac{1.22\cdot MSS}{RTT\cdot\sqrt{L}}$$

由此我们得到$\mathrm{L}=(1.22\times\mathrm{MSS}/(\mathrm{B}\times\mathrm{RTT}))^2$，由于两个连续的分组丢失，由 1/L 的分组 TCP 的发送方在发送，因此$\mathrm{T}=(1/\mathrm{L})\times\mathrm{MSS}/\mathrm{B}$，这样我们可以得到$\mathrm{T}=\mathrm{B}\times RTT^2/(1.22^2\times MSS)$，由此可知 T 是一个关于吞吐量 B 的函数。

P50. Consider a simplified TCP's AIMD algorithm where the congestion window size is measured in number of segments, not in bytes. In additive increase, the congestion window size increases by one segment in each RTT. In multiplica- tive decrease, the congestion window size decreases by half (if the result is not an integer, round down to the nearest integer). Suppose that two TCP connections, C1 and C2, share a single congested link of speed 30 segments per second. Assume that both C1 and C2 are in the congestion avoidance phase. Connection C1's RTT is 50 msec and connection C2's RTT is 100 msec. Assume that when the data rate in the link exceeds the link's speed, all TCP connections experience data segment loss.

a. If both C1 and C2 at time t0 have a congestion window of 10 segments, what are their congestion window sizes after 1000 msec?

b. In the long run, will these two connections get the same share of the band- width of the congested link? Explain.

答：a. C1 和 C2 的主要区别在于 C1 的 RTT 为 C2 的一半，因此 C1 在 50ms(毫秒)后调整它的窗口大小，但 C2 在 100ms 后才开始调整窗口大小。假定无论在何时分组丢包事件发生，C1 在 50ms 后知道此情况，C2 在 100ms 后了解此情况。我们更进一步用下面简化的 TCP 模型。在每一个 RTT 之后，一个连接确定是否需要增加窗口大小。对于 C1，我们在前 50ms 计算得到平均总的发送速率，如果这个速率超出了链路最大传输速率，那么我们假定 C1 检测出分组丢失然后减小它的发送窗口大小。但对于 C2 而言，在前 100ms 我们计算得出其总的平均发送速率，如果这个速率超出了链路最大传输速率，那么我们假定 C2 检测出分组丢失然后减小它的发送窗口大小。注意在最后的 50ms 内平均发送速率超出链路传输速率这是可能发生的，但在最后的 100ms 平均发送速率是小于或者等于链路瓶颈传输速率的在这样的情况下我们假定 C1 出现分组丢失(即丢包)但是 C2 未出现。

下面的表格，基于上面的分析描述了窗口大小和发送速率的变化。

	C1		C2	
Time (msec)	Window Size (num. of segments sent in next 50msec)	Average data sending rate (segments per second, =Window/0.05)	Window Size(num. of segments sent in next 100msec)	Average data sending rate (segments per second, =Window/0.1)
0	10	200 (in [0-50]msec]	10	100 (in [0-50]msec)
50	5 (decreases window size as the avg. total sending rate to the link in **last 50msec** is 300= 200+100)	100 (in [50-100]msec]		100 (in [50-100]msec)

100	2 (decreases window size as the avg. total sending rate to the link in **last 50msec** is 200= 100+100)	40	5 (decreases window size as the avg. total sending rate to the link in **last 100msec** is 250= (200+100)/2 + (100+100)/2)	50
150	1 (decreases window size as the avg. total sending rate to the link in last 50msec is 90= (40+50)	20		50
200	1 (no further decrease, as window size is already 1)	20	2 (decreases window size as the avg. total sending rate to the link in **last 100msec** is 80= (40+20)/2 + (50+50)/2)	20
250	1 (no further decrease, as window size is already 1)	20		20

300	1 (no further decrease, as window size is already 1)	20	1 (decreases window size as the avg. total sending rate to the link in **last 100msec** is 40= (20+20)/2 + (20+20)/2)	10
350	2	40		10
400	1	20	1	10
450	2	40		10
500	1 (decreases window size as the avg. total sending rate to the link in last 50msec is 50= (40+10)	20	1	10
550	2	40		10
600	1	20	1	10
650	2	40		10
700	1	20	1	10
750	2	40		10
800	1	20	1	10
850	2	40		10
900	1	20	1	10
950	2	40		10
1000	1	20	1	10

从上面的表格可以看出，在 1000ms 以后，C1 和 C2 的窗口大小都是 1 个报文段大小。

b. 不，在长期的运行中，C1 的共享带宽是 C2 的 2 倍，因为 C1RTT 更短，因此 C1 调整窗口大小的速率是 C2 的 2 倍。我们可以从表格看出，每 200ms 是一个循环，比如：从 850ms 到 1000ms 亦包含在里面。在一个循环里面，C1 的发送速率为: $(40 + 20 + 40 + 20) = 120$，C2 为：$(10 + 10 + 10 + 10) = 40$，C1 是 C2 的 3 倍。

P51. Consider the network described in the previous problem. Now suppose that the two TCP connections, C1 and C2, have the same RTT of 100 msec. Sup- pose that at time t0, C1's congestion window size is 15 segments but C2's congestion window size is 10 segments.

a. What are their congestion window sizes after 2200msec?

b. In the long run, will these two connections get about the same share of the bandwidth of the congested link?

c. We say that two connections are synchronized, if both connections reach their maximum window sizes at the same time and reach their minimum window sizes at the same time. In the long run, will these two connections get synchronized eventually? If so, what are their maximum window sizes?

d. Will this synchronization help to improve the utilization of the shared link? Why? Sketch some idea to break this synchronization.

答：a. 和上一个问题类似，在下面的表格中我们可以通过时间计算出窗口大小，在 2200ms 以后 C1 和 C2 窗口大小都为 2。表格如下：

	C1		C2	
Time (msec)	Window Size (num. of segments sent in next 100msec)	Data sending speed (segments per second, =Window/0.1)	Window Size(num. of segments sent in next 100msec)	Data sending speed (segments per second, =Window/0.1)
0	15	150 (in [0-100]msec]	10	100 (in [0-100]msec)
100	7	70	5	50
200	3	30	2	20
300	1	10	1	10
400	2	20	2	20
500	1	10	1	10
600	2	20	2	20
700	1	10	1	10
800	2	20	2	20
900	1	10	1	10
1000	2	20	2	20
1100	1	10	1	10
1200	2	20	2	20
1300	1	10	1	10
1400	2	20	2	20
1500	1	10	1	10
1600	2	20	2	20
1700	1	10	1	10
1800	2	20	2	20
1900	1	10	1	10
2000	2	20	2	20
2100	1	10	1	10
2200	2	20	2	20

b. 是的由于的 AIMD 算法两条连接最终会有相同的 RTT。

c. 是的，这个可从上面的表格中清晰的看出来，他们的最大窗口大小为 2.

d. 不，这个同步即窗口大小相同，不会提升链路的效能，因为这两条连接在各自作为单一的连接时会在最小和最大窗口之间震荡。因此，链路不可能百分百被利用(我们假定链路缓

存为空)。一个可行的破坏同步的办法是限制链路缓存大小并且在缓存空间溢满是随机丢弃分组。这回使这两条不同连接在不同时间减小窗口大小。有很多 AMQ(Active Queue Management)技术可以实现，比如 RED(Random Early Detect)，PI (Proportional and Integral AQM)，AVQ (Adaptive Virtual Queue)，和 REM (Random Exponential Marking)等。

P52. Consider a modification to TCP's congestion control algorithm. Instead of additive increase, we can use multiplicative increase. A TCP sender increases its window size by a small positive constant a (0 < a < 1) whenever it receives a valid ACK. Find the functional relationship between loss rate L and maximum congestion window W. Argue that for this modified TCP, regardless of TCP's average throughput, a TCP connection always spends the same amount of time to increase its congestion window size from W/2 to W.

答：用 W 表示最大的窗口大小。当 TCP 窗口大小从 W/2 增大到 W 时，在时间间隔内我们可以找到发送出去的报文段的总数量为 S。由下式可得：

$$\mathrm{S} = \frac{\mathrm{W}}{2} + \frac{W}{2}(1+\alpha) + \frac{W}{2}(1+\alpha)^2 + \frac{W}{2}(1+\alpha)^3 + \cdots + \frac{W}{2}(1+\alpha)^k$$

我们解得，当$\mathrm{k} = \log_{1+\alpha} 2$时得到 S 最大值$\mathrm{S} = \mathrm{W} \times (2\alpha + 1)/2\alpha$

则丢包率 L 为：

$$\mathrm{L} = \frac{1}{S} = \frac{2\alpha}{W \times (2\alpha + 1)}$$

TCP 拥塞窗口从 W/2 增至 W 所用时间由下式可得：

$\mathrm{k} \times \mathrm{RTT} = (\log_{(1+\alpha)} 2) \times RTT$，这清楚的显示出各自 TCP 平均吞吐量。

现在，TCP 平均吞吐量由下式可得：

$$\mathrm{B} = \mathrm{MSS} \times \frac{S}{(k+1) \times RTT} = \frac{MSS}{L \times (k+1) \times RTT}$$

课本给出的平均为：

$$\mathrm{B} = \frac{1.22MSS}{RTT\sqrt{L}}$$

这与计算得到的平均吞吐率有区别，课本给出的 L 是以二次方根的形式出现。

P53. In our discussion of TCP futures in Section 3.7, we noted that to achieve a throughput of 10 Gbps, TCP could only tolerate a segment loss probability of 2×10^{-10} (or equivalently, one loss event for every 5,000,000,000 segments). Show the derivation for the values of 2×10^{-10} (1 out of 5,000,000,000) for the RTT and MSS values given in Section 3.7. If TCP needed to support a 100 Gbps connection, what would the tolerable loss be?

答：由吞吐率公式：$= \frac{1.22 \times MSS}{RTT \times \sqrt{L}}$即：

$$\frac{100\mathrm{Gb}}{\mathrm{sec}} = \frac{1.22 \times (1500 \times 8)bits}{0.1sec \times \sqrt{L}}$$

求得：

$$\sqrt{L} = \frac{1.22 \times 1500 \times 8}{0.1 \times 100 \times 10^9}$$

得到可以容忍的丢包率为$L = 2.14 \times 10^{-12}$

P54. In our discussion of TCP congestion control in Section 3.7, we implicitly assumed that the TCP sender always had data to send. Consider now the case that the TCP sender sends a large amount of data and then goes idle (since it has no more data to send) at t1. TCP remains idle for a relatively long period of time and then wants to send more data at t2. What are the advantages and dis- advantages of having TCP use the cwnd and ssthresh values from t1 when starting to send data at t2? What alternative would you recommend? Why?

答：将在 t1 时刻的 CongWin 和 Threshold 值用在 t2 时刻，其优点是 TCP 不需要经过慢启动的避免拥塞阶段即可直接跳到在 t1 时刻得到的吞吐量值。使用这些值的缺点是它们可能已经不正确了。比如，如果路径在 t1 到 t2 的时间内变得更拥塞了，发送方会将大量的窗口内的有用报文段发送到已经更加拥塞的链路上去。

P55. In this problem we investigate whether either UDP or TCP provides a degree of end-point authentication.

a. Consider a server that receives a request within a UDP packet and responds to that request within a UDP packet (for example, as done by a DNS server). If a client with IP address X spoofs its address with address Y, where will the server send its response?

b. Suppose a server receives a SYN with IP source address Y, and after responding with a SYNACK, receives an ACK with IP source address Y with the correct acknowledgment number. Assuming the server chooses a random initial sequence number and there is no “man-in-the-middle,” can the server be certain that the client is indeed at Y (and not at some other address X that is spoofing Y)?

答：a. 服务器将向 Y 发送响应报文；

b. 服务器能够确定该客户机的确是 Y。如果是其它地址 X 欺骗为 Y，那么 SYNACK 将会被发送到地址 Y，并且 X 中的 TCP 连接过程中不会向服务器发送 TCP ACK 消息，即使入侵者能够适时的发送了 TCP ACK 消息，他也不会知道服务器的初始序号(因为服务器使用随机的初始序号)，不能与服务器进行正常的通信。

P56. In this problem, we consider the delay introduced by the TCP slow-start phase. Consider a client and a Web server directly connected by one link of rate R. Suppose the client wants to retrieve an object whose size is exactly equal to 15 S, where S is the maximum segment size (MSS). Denote the round-trip time between client and server as RTT (assumed to be constant). Ignoring protocol headers, determine the time to retrieve the object (including TCP connection establishment) when

a. 4 S/R > S/R + RTT > 2S/R

b. S/R + RTT > 4 S/R

c. S/R > RTT.

答：a. 参看下图，我们可以看出，总时延为：

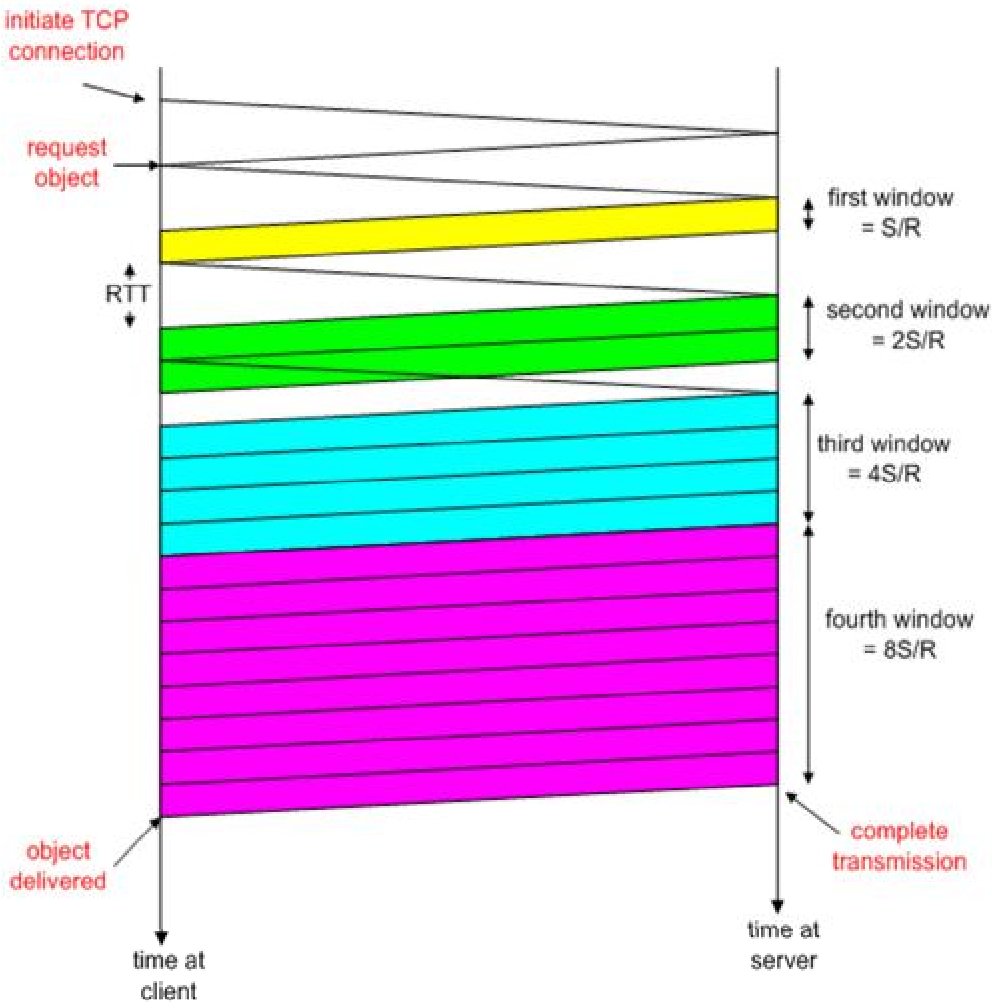
$$\mathrm{RTT} + \mathrm{RTT} + S/R + \mathrm{RTT} + S/R + \mathrm{RTT} + 12S/R = 4\mathrm{RTT} + 14\,S/R$$

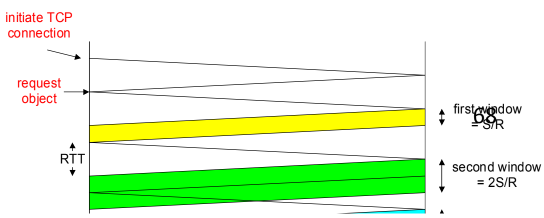
b. 类似的，此时的时延为：

$$\mathrm{RTT} + \mathrm{RTT} + S/R + \mathrm{RTT} + S/R + \mathrm{RTT} + S/R + \mathrm{RTT} + 8S/R = 5\mathrm{RTT} + 11\,S/R$$

c. 同理，在这个例子中的时延为：

$$RTT + RTT + S/R + RTT + 14\,S/R = 3\,RTT + 15\,S/R$$





Chapter 4 The Network Layer

P1. In this question, we consider some of the pros and cons of virtual-circuit and datagram networks.

a. Suppose that routers were subjected to conditions that might cause them to fail fairly often. Would this argue in favor of a VC or datagram archi- tecture? Why?

b. Suppose that a source node and a destination require that a fixed amount of capacity always be available at all routers on the path between the source and destination node, for the exclusive use of traffic flowing between this source and destination node. Would this argue in favor of a VC or datagram architecture? Why?

c. Suppose that the links and routers in the network never fail and that routing paths used between all source/destination pairs remains constant. In this scenario, does a VC or datagram architecture have more control traffic overhead? Why?

答：a. 对于一个面相连接的网络来说，路由器故障会影响到这个连接的所有路由。至少需要出现故障的路由器上游的路由器重新建立一个新的到目标节点的下行部分的路径，同时，发出一个信息，该信息包含所有与建立一个新的路径有关要求。而且，先前路径中出现故障的路由器下游的所有路由器都必须拆除这个故障连接，同时发出另一个信息，包含了与所要做的事情有关的所有要求。对于面向无连接的数据报网络来说，不会需要用于建立一个新的下行路径或者拆除一个旧的下行路径信令。然而我们知道，由于要考虑到出故障的路由器，路由表将需要更新(不论是用链路状态算法还是距离向量算法)。我们知道运用距离向量算法，我们有时可以把路由表的变化定位在出故障的路由器附近的范围内。因此，数据报网络是更可取的。

b. 为了让路由器能够确定一条输出链路的延时(或延时的界限),就需要知道通过这条链路传输的所有会话通信的特性。也就是说，路由器必须知道内部每一个会话的状态，这对于一个面相连接的网络是有可以能的，但是对于一个面向无连接的网络则不可能。此时面向连接的网络更加可取。

P2. Consider a virtual-circuit network. Suppose the VC number is an 8-bit field.

a. What is the maximum number of virtual circuits that can be carried over a link?

b. Suppose a central node determines paths and VC numbers at connection setup. Suppose the same VC number is used on each link along the VC's path. Describe how the central node might determine the VC number at connection setup. Is it possible that there are fewer VCs in progress than the maximum as determined in part (a) yet there is no common free VC number?

c. Suppose that different VC numbers are permitted in each link along a VC's path. During connection setup, after an end-to-end path is determined, describe how the links can choose their VC numbers and configure their forwarding tables in a decentralized manner, without reliance on a central node.

答：a. 链路能承载的 VCs(虚电路)最大数量为：$2^8 = 256$。

b. 中心节点可以从$\{0,1,2 \cdots, 2^8 - 1\}$中任取一个 VC 号。这种情况下,进行中的虚电路数量小于 256 而没有相同的未用 VC 号是不可能的。

c. 每一个链路可以自由的从$\{0,1,2 \cdots, 2^8 - 1\}$中分配一个 VC 号。因此，很可能一个虚电路在它的路径上的每一条链路都有各不相同的 VC 号。而虚电路路径上的每一个路由器都需要为到达的分组更换一个与输出链路有关的 VC 号。

P3. A bare-bones forwarding table in a VC network has four columns. What is the meaning of the values in each of these columns? A bare-bones forwarding table in a datagram network has two columns. What is the meaning of the values in each of these columns?

答：虚电路转发表中，分别为：入接口，入 VC 号，出接口，出 VC 号。数据报网络转发表中，分别为：目标地址(前缀匹配)，输出链路接口。

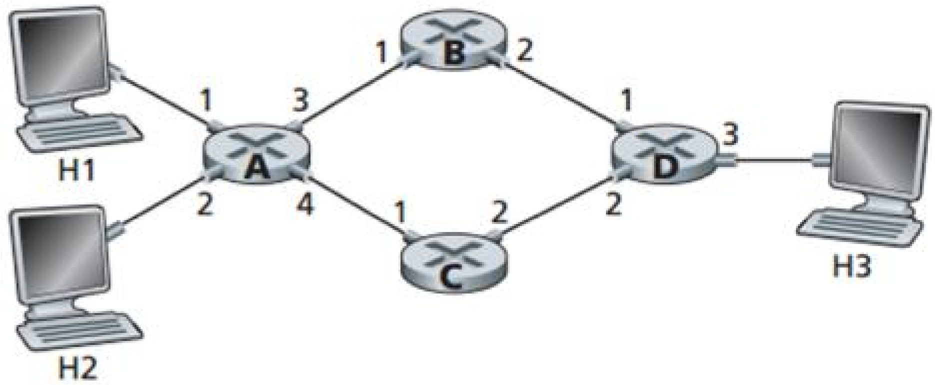
P4. Consider the network below.

a. Suppose that this network is a datagram network. Show the forwarding table in router A, such that all traffic destined to host H3 is forwarded through interface 3.

b. Suppose that this network is a datagram network. Can you write down a forwarding table in router A, such that all traffic from H1 destined to host H3 is forwarded through interface 3, while all traffic from H2 destined to host H3 is forwarded through interface 4? (Hint: this is a trick question.)

c. Now suppose that this network is a virtual circuit network and that there is one ongoing call between H1 and H3, and another ongoing call between H2 and H3. Write down a forwarding table in router A, such that all traffic from H1 destined to host H3 is forwarded through interface 3, while all traffic from H2 destined to host H3 is forwarded through interface 4.

d. Assuming the same scenario as (c), write down the forwarding tables in nodes B, C, and D.



答：a. 传向主机 H3 的数据通过接口路由器 D 的 3 转发:

目的地址	链路接口
H3	3

b. 不能，因为转发规则仅仅专注于目标地址。

c. 一个可能的配置是：

入接口	入 VC 号	出接口	出 VC 号
1	12	3	22
2	63	4	18

注意两个数据流可能有相同的 VC 号。

d. 一个可能的配置是：

路由器 B：

入接口	入 VC 号	出接口	出 VC 号
1	22	2	24

路由器 C：

入接口	入 VC 号	出接口	出 VC 号
1	18	2	50

路由器 D：

入接口	入 VC 号	出接口	出 VC 号
1	24	3	70
2	50	3	76

P5. Consider a VC network with a 2-bit field for the VC number. Suppose that the network wants to set up a virtual circuit over four links: link A, link B, link C, and link D. Suppose that each of these links is currently carrying two other virtual circuits, and the VC numbers of these other VCs are as follows:

Link A	Link B	Link C	Link D
00	01	10	11
01	10	11	00

In answering the following questions, keep in mind that each of the existing VCs may only be traversing one of the four links.

a. If each VC is required to use the same VC number on all links along its path, what VC number could be assigned to the new VC?

b. If each VC is permitted to have different VC numbers in the different links along its path (so that forwarding tables must perform VC number transla- tion), how many different combinations of four VC numbers (one for each of the four links) could be used?

答：a. 不能为新的虚电路分配 VC 号，因为由题可知 2 比特字段已经全部分配完。

b. 每个链路有 2 个可用的 VC 号，共 4 条链路，因此共有$2^4 = 16$种不同的组合，比如(10,00,00,10)为其中一种。

P6. In the text we have used the term connection-oriented service to describe a transport-layer service and connection service for a network-layer service. Why the subtle shades in terminology?

答：在一个虚电路网络中，存在一条端到端的连接，而这条路径上的每个路由器都必须保持这个连接状态，因此术语称作连接服务。在一个基于无连接网络层的面向连接运输服务中(如基于 IP 的 TCP 传输)由终端系统保持连接状态，而路由器并不清楚是怎么样连接的，因此，术语中称作面相连接服务。

P7. Suppose two packets arrive to two different input ports of a router at exactly the same time. Also suppose there are no other packets anywhere in the router.

a. Suppose the two packets are to be forwarded to two different output ports. Is it possible to forward the two packets through the switch fabric at the same time when the fabric uses a shared bus?

b. Suppose the two packets are to be forwarded to two different output ports. Is it possible to forward the two packets through the switch fabric at the same time when the fabric uses a crossbar?

c. Suppose the two packets are to be forwarded to the same output port. Is it possible to forward the two packets through the switch fabric at the same time when the fabric uses a crossbar?

答：a. 不能，当交换结构共享总线时，在一个时刻只能传输一个分组。

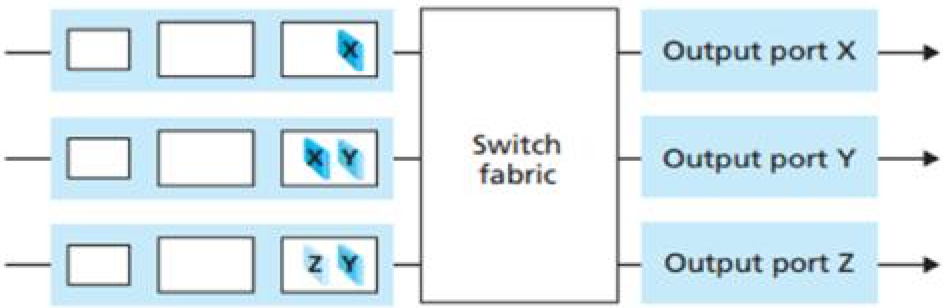
b. 可以，正如在课本讨论那样，只要两个分组使用不同的输入总线和输出总线，它们可以并行同时传输。

c. 不能，在这种情形下两个分组同时传向同一个输出总线，这是不可能的。

P8. In Section 4.3, we noted that the maximum queuing delay is (n–1)D if the switching fabric is n times faster than the input line rates. Suppose that all packets are of the same length, n packets arrive at the same time to the n input ports, and all n packets want to be forwarded to different output ports. What is the maximum delay for a packet for the (a) memory, (b) bus, and (c) crossbar switching fabrics?

答：内存，总线和纵横制三种交换结构的最大时延分别为(n-1)D，(n-1)D 和 0。

P9. Consider the switch shown below. Suppose that all datagrams have the same fixed length, that the switch operates in a slotted, synchronous manner, and that in one time slot a datagram can be transferred from an input port to an output port. The switch fabric is a crossbar so that at most one datagram can be transferred to a given output port in a time slot, but different output ports can receive datagrams from different input ports in a single time slot. What is the minimal number of time slots needed to transfer the packets shown from input ports to their output ports, assuming any input queue scheduling order you want (i.e., it need not have HOL blocking)? What is the largest number of slots needed, assuming the worst-case scheduling order you can devise, assuming that a non-empty input queue is never idle?



答：最小需要三个时隙(slot)，分配如下：

时隙 1：发送在顶端输入队列的分组 X ，发送在中间输入队列的分组 Y。

时隙 2：发送在中间输入队列的分组 X ，发送在底端输入队列的分组 Y。

时隙 3：发送在底端输入队列的分组 Z。

最大时隙数量仍然为 3.实际上，假定一个非空输入队列永不空闲，我们可以得到：在第一个时隙总是由发送在顶端输入队列的分组 X 和在中间或者底端输入队列的分组 Y 构成；在第二个时隙，我们可以发送 2 个以上的数据报，最后后的数据报可以在第三个时隙发送完成。

注意：如果第一个数据报是中间输入队列的分组 X，那么最坏的情况下需要 4 个时隙。

P10. Consider a datagram network using 32-bit host addresses. Suppose a router has four links, numbered 0 through 3, and packets are to be forwarded to the link interfaces as follows:

Destination Address Range	Link Interface
11100000 00000000 00000000 00000000 through 11100000 00111111 11111111 11111111	0
11100000 01000000 00000000 00000000 through 11100000 01000000 11111111 11111111	1
11100000 01000001 00000000 00000000	

through(到) 2
11100001 01111111 11111111 11111111

otherwise 3

a. Provide a forwarding table that has five entries, uses longest prefix match- ing, and forwards packets to the correct link interfaces.
b. Describe how your forwarding table determines the appropriate link inter- face for datagrams with destination addresses:
11001000 10010001 01010001 01010101
11100001 01000000 11000011 00111100
11100001 10000000 00010001 01110111
答：a. 如下：

前缀匹配	链路接口
11100000 00	0(entry 1)
11100000 01000000	1(entry 2)
1110000	2(entry 3)
11100001 1	3(entry 4)
其它	3(entry 5)

b. 第 1 个地址的最长前缀匹配是第 5 个条目：接口 3，
第 2 个地址的最长前缀匹配是第 3 个条目：接口 2
第 3 个地址的最长前缀匹配是第 3 个条目：接口 3

P11. Consider a datagram network using 8-bit host addresses. Suppose a router uses longest prefix matching and has the following forwarding table:

Prefix Match	Interface
00	0
010	1
011	2
10	2
11	3

For each of the four interfaces, give the associated range of destination host addresses and the number of addresses in the range.
答：如下：

目标地址范围	链路接口	地址数量
00000000 到 00111111	0	64
01000000 到 01011111	1	32
01100000 到 01111111	2	32
10000000 到 10111111	2	64
11000000 到 11111111	3	64

P12. Consider a datagram network using 8-bit host addresses. Suppose a router uses longest prefix matching and has the following forwarding table:

Prefix Match	Interface
1	0
10	1
111	2
otherwise	3

For each of the four interfaces, give the associated range of destination host addresses and the number of addresses in the range.

答：如下：

目标地址范围	链路接口	地址数量
1100 0000 到 1101 1111	0	32
1000 0000 到 1011 1111	1	64
1110 0000 到 1111 1111	2	32
0000 0000 到 0111 1111	3	128

P13. Consider a router that interconnects three subnets: Subnet 1, Subnet 2, and Subnet 3. Suppose all of the interfaces in each of these three subnets are required to have the prefix 223.1.17/24. Also suppose that Subnet 1 is required to support at least 60 interfaces, Subnet 2 is to support at least 90 interfaces, and Subnet 3 is to support at least 12 interfaces. Provide three network addresses (of the form a.b.c.d/x) that satisfy these constraints.

答：子网 1，2，3 可按如下分配：

子网 1：223.1.17.0/26；

子网 2：223.1.17.128/25；

子网 3：223.1.17.192/28。

P14. In Section 4.2.2 an example forwarding table (using longest prefix matching) is given. Rewrite this forwarding table using the a.b.c.d/x notation instead of the binary string notation.

答：如下：

目的地址	链路接口
200.23.16/21	0
200.23.24/24	1
200.23.24/21	2
其它	3

P15. In Problem P10 you are asked to provide a forwarding table (using longest prefix matching). Rewrite this forwarding table using the a.b.c.d/x notation instead of the binary string notation.

答：如下：

目的地址	链路接口
11100000(224.0/10)	0
11100000 01000000(224.64/16)	1
1110000(224/8)	2
11100001 1(225.128/9)	3
其它	3

P16. Consider a subnet with prefix 128.119.40.128/26. Give an example of one IP address (of form xxx.xxx.xxx.xxx) that can be assigned to this network. Suppose an ISP owns the block of addresses of the form 128.119.40.64/26. Suppose it wants to create four subnets from this block, with each block having the same number of IP addresses. What are the prefixes (of form a.b.c.d/x) for the four subnets?

答：IP 地址的范围是：129.119.10.128 到 129.119.10.191，可从中任选一个。四个字网前缀如下：

子网	网络地址
1	129.119.10.64/28
2	129.119.10.80/28
3	129.119.10.96/28
4	129.119.10.112/28

P17. Consider the topology shown in Figure 4.17. Denote the three subnets with hosts (starting clockwise at 12:00) as Networks A, B, and C. Denote the subnets without hosts as Networks D, E, and F.

a. Assign network addresses to each of these six subnets, with the following constraints: All addresses must be allocated from 214.97.254/23; Subnet A should have enough addresses to support 250 interfaces; Subnet B should have enough addresses to support 120 interfaces; and Subnet C should have enough addresses to support 120 interfaces. Of course, subnets D, E and F should each be able to support two interfaces. For each subnet, the assignment should take the form a.b.c.d/x or a.b.c.d/x – e.f.g.h/y.

b. Using your answer to part (a), provide the forwarding tables (using longest prefix matching) for each of the three routers.

答：a. 从 214.97.254/23 起开始分配，一个可行的分配方案如下：

子网 D：214.97.254.0/31(2 个地址)；

子网 E：214.97.254.2/31(2 个地址)；

子网 F：214.97.254.4/30(4 个地址)；

子网 B：214.97.254.0/25—214.97.254.0/29(128-8=120 个地址)；

子网 C：214.97.254.128/25(128 个地址)；

子网 A：214.97.254.255/24(256 个地址)。

b. 路由器R_1如下：

最长前缀匹配(Longest Prefix Match)	输出接口(Outgoing Interface)
11010110 01100001 11111111	A
11010110 01100001 11111110 0000000	D

11010110 01100001 11111110 000001　　F

路由器R_2如下：

最长前缀匹配(Longest Prefix Match)	输出接口(Outgoing Interface)
11010110 01100001 11111110 000001	F
11010110 01100001 11111110 0000001	E
11010110 01100001 11111110 1	C

路由器R_3如下：

最长前缀匹配(Longest Prefix Match)	输出接口(Outgoing Interface)
11010110 01100001 11111110 0000000	D
11010110 01100001 11111110 0	B
11010110 01100001 11111110 0000001	E

P18. Use the whois service at the American Registry for Internet Numbers (http://www.arin.net/whois) to determine the IP address blocks for three universities. Can the whois services be used to determine with certainty the geographical location of a specific IP address? Use www.maxmind.com to determine the locations of the Web servers at each of these universities.

答：在中国通过网站 http://www.cnnic.cn/ 进行 whois 查询得到：

陕西师范大学(202.117.144.2)的 IP 地址块为：

地址范围：202.117.144.0 - 202.117.159.255，

CIDR 为：202.117.144/20，有 12 位主机地址，约$16 \times 256 = 4096$个对外地址；

清华大学(IP 为 166.111.4.100)的 IP 地址块为：

地址范围：166.111.0.0 - 166.111.255.255，

CIDR 为：166.111/16，由 16 位主机地址，约$256 \times 256 = 65536$个对外地址；

北京大学(IP 为 111.205.231.1)的 IP 地址块为：

地址范围为：111.192.0.0 - 111.207.255.255，

CIDR 为：111.192/12，有 20 位主机地址，约$16 \times 256 \times 256 = 1048576$个对外地址。。。

不能，WHOIS 查询不能定位 IP 地址的地理位置。

通过网站 www.maxmind.com　定位 IP 地理位置得：

IP 地址	国家代码	地点	邮政编码	坐标	网络服务提供商	机构	域名	大城市代码
202.117.144.2	CN	西安, 陕西, 中国, 亚洲		34.2583, 108.9286	China Education and Research Network Center	China Education and Research Network Center		

IP地址	国家代码	地点	邮政编码	坐标	网络服务提供商	机构	域名	大城市代码
166.111.4.100	CN	中国, 亚洲		35, 105	China Education and Research Network Center	China Education and Research Network Center		

IP地址	国家代码	地点	邮政编码	坐标	网络服务提供商	机构	域名	大城市代码
111.205.231.1	CN	北京, 北京市, 中国, 亚洲		39.9289, 116.3883	China Unicom Beijing	China Unicom Beijing		

P19. Consider sending a 2400-byte datagram into a link that has an MTU of 700 bytes. Suppose the original datagram is stamped with the identification number 422. How many fragments are generated? What are the values in the various fields in the IP datagram(s) generated related to fragmentation?

答：每个 IP 数据报分片的最大尺寸为 680(20bytes IP 首部)，因此产生的分片的个数为：$\lceil(2400-20)/680\rceil=4$($\lceil\ \ \rceil$表示取整)，每个报文段都会有一个标识码为 422。除了最后一个报文段，每个报文段的大小都为 700bytes(包括 IP 首部)。最后一个报文段大小为 $2400-680*3=360$bytes(包括 20bytesIP 首部).这 4 个报文段的偏移量分别为 0，85，170，255。即偏移量跨度为 680/8=85。前 3 个报文段 flag=1，最后一个报文段 flag=0。

P20. Suppose datagrams are limited to 1,500 bytes (including header) between source Host A and destination Host B. Assuming a 20-byte IP header, how many datagrams would be required to send an MP3 consisting of 5 million bytes? Explain how you computed your answer.

答：MP3 文件大小为 500 万字节，假定这个数据是由 TCP 传输，每个 TCP 报文段有 20 字节的首部，所以每个报文段可以携带$1500-20-20=1460$字节的 MP3 文件，所以需要报文段的数量为：

$$\left\lceil\frac{5\times10^6 bytes}{1500-20-20}\right\rceil=3425$$

除了最后一个，所有的 IP 数据报都是 1500bytes 大小，最有一个数据报是

$960+40=1000$bytes。注意没有这样的分片—源主机不会创建超过 1500 字节的数据报，并且在实际中这些数据报比链路 MTU(最大传输单元)要小。

P21. Consider the network setup in Figure 4.22. Suppose that the ISP instead assigns the router the address 24.34.112.235 and that the network address of the home network is 192.168.1/24.

a. Assign addresses to all interfaces in the home network.

b. Suppose each host has two ongoing TCP connections, all to port 80 at host 128.119.40.86. Provide the six corresponding entries in the NAT translation table.

答：a. 三个家庭地址为：192.168.1.1，192.168.1.2，192.168.1.3，路由器接口为：192.168.1.4。

b. 如下：

NAT 转换表

WAN 端		LAN 端	
IP address	TCP port numbers	IP address	TCP port numbers
24.34.112.235	4000	192.118.1.1	3345
24.34.112.235	4001	192.118.1.1	3346
24.34.112.235	4002	192.118.1.2	3445
24.34.112.235	4003	192.118.1.2	3446
24.34.112.235	4004	192.118.1.3	3545
24.34.112.235	4005	192.118.1.3	3546

P22. Suppose you are interested in detecting the number of hosts behind a NAT. You observe that the IP layer stamps an identification number sequentially on each IP packet. The identification number of the first IP packet generated by a host is a random number, and the identification numbers of the subsequent IP packets are sequentially assigned. Assume all IP packets generated by hosts behind the NAT are sent to the outside world.

a. Based on this observation, and assuming you can sniff all packets sent by the NAT to the outside, can you outline a simple technique that detects the number of unique hosts behind a NAT? Justify your answer.

b. If the identification numbers are not sequentially assigned but randomly assigned, would your technique work? Justify your answer.

答：a. 由于所有的 IP 分组都被发送到 NAT 路由器外面，因此我们可以用一个分组抓包器记录所有由 NAT 路由器后面主机产生的分组。每台主机生成一系列有序的 IP 分组(他们很可能是从一个大的序号空间选择出来的)和一个明显的初始标识号码(ID)，我们可以用连续的 ID 来对 IP 分组组合成一个个簇，簇的号码也就是在 NAT 路由器后面一个个主机的号码。更实际的算法，我们参考以下书籍：

“A Technique for Counting NATted Hosts”, by Steven M. Bellovin, appeared in IMW’02, Nov. 6-8, 2002, Marseille, France.

“Exploiting the IPID field to infer network path and end-system characteristics.”

Weifeng Chen, Yong Huang, Bruno F. Ribeiro, Kyoungwon Suh, Honggang Zhang, Edmundo de Souza e Silva, Jim Kurose, and Don Towsley.

PAM'05 Workshop, March 31 - April 01, 2005. Boston, MA, USA.

b. 然而，如果这些标识号不是有序分配而是随机分配，在 a 中提到的技术将无法起作用，在抓包的数据中将无法出现簇群。

P23. In this problem we’ll explore the impact of NATs on P2P applications. Suppose a peer with username Arnold discovers through querying that a peer with username Bernard has a file it wants to download. Also suppose that Bernard and Arnold are both behind a NAT. Try to devise a technique that will allow Arnold to establish a TCP connection with Bernard without application-specific NAT configuration. If you have difficulty devising such a technique, discuss why.

答：当 Arnold 发现 Bernard 有他要下载的文件时，他会发出一个 TCP SYN 分组，带有 Bernard 的地址及一些目标端口号，当 NAT 收到这个 TCP SYN 分组时，由于没有条目指明如何从

WAN 侧建立连接，NAT 不知道应该将这个分组送到内网的哪个主机，故会丢弃这个 SYN 分组。同理，Bernard 不能成功的申请和 Arnold 建立连接，所以不可能设计出这种技术。

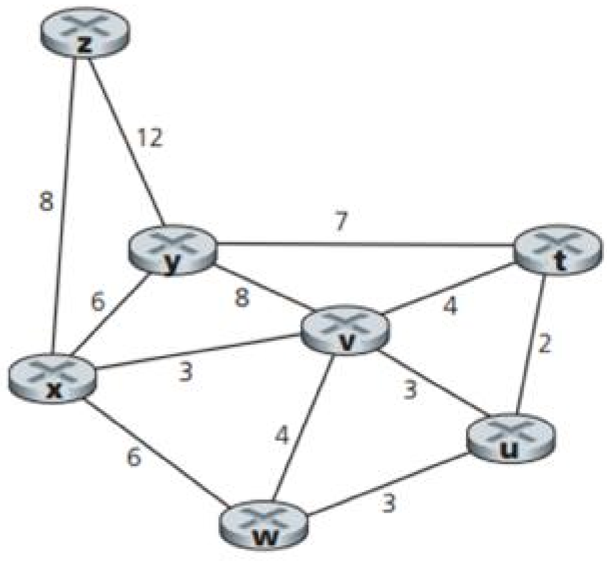
P24. Looking at Figure 4.27, enumerate the paths from y to u that do not contain any loops.

答：y-x-u，y-x-v-u，y-x-w-u，y-x-w-v-u，
y-w-u，y-w-v-u，y-w-x-u，y-w-x-v-u，y-w-v-x-u，
y-z-w-u，y-z-w-v-u，y-z-w-x-u，y-z-w-x-v-u，y-z-w-v-x-u。

P25. Repeat Problem P24 for paths from x to z, z to u, and z to w.

答：x 到 z：
x-y-z，x-y-w-z，
x-w-z，x-w-y-z，
x-v-w-z，x-v-w-y-z，
x-u-w-z，x-u-w-y-z，
x-u-v-w-z，x-u-v-w-y-z。
z 到 u：
z-w-u，z-w-v-u，z-w-x-u，z-w-v-x-u，z-w-x-v-u，z-w-y-x-u，z-w-y-x-v-u，
z-y-x-u，z-y-x-v-u，z-y-x-w-u，z-y-x-w-y-u，z-y-x-v-w-u，
z-y-w-v-u，z-y-w-x-u，z-y-w-v-x-u，z-y-w-x-v-u，z-y-w-y-x-u，z-y-w-y-x-v-u。
z 到 w：
z-w，z-y-w，z-y-x-w，z-y-x-v-w，z-y-x-u-w，z-y-x-u-v-w，z-y-x-v-u-w。

P26. Consider the following network. With the indicated link costs, use Dijkstra's shortest-path algorithm to compute the shortest path from x to all network nodes. Show how the algorithm works by computing a table similar to Table 4.3.



答：x 到所有网络节点的最短路径如下：

Step	*N'*	*D(t),p(t)*	*D(u),p(u)*	*D(v),p(v)*	*D(w),p(w)*	*D(y),p(y)*	*D(z),p(z)*

0	x	∞	∞	3,x	6,x	6,x	8,x
1	xv	7,v	6,v	3,x	6,x	6,x	8,x
2	xvu	7,v	6,v	3,x	6,x	6,x	8,x
3	xvuw	7,v	6,v	3,x	6,x	6,x	8,x
4	xvuwy	7,v	6,v	3,x	6,x	6,x	8,x
5	xvuwyt	7,v	6,v	3,x	6,x	6,x	8,x
6	xvuwytz	7,v	6,v	3,x	6,x	6,x	8,x

P27. Consider the network shown in Problem P26. Using Dijkstra's algorithm, and showing your work using a table similar to Table 4.3, do the following:

a. Compute the shortest path from t to all network nodes.

b. Compute the shortest path from u to all network nodes.

c. Compute the shortest path from v to all network nodes.

d. Compute the shortest path from w to all network nodes.

e. Compute the shortest path from y to all network nodes.

f. Compute the shortest path from z to all network nodes.

答：a. t 到所有网络节点的最短路径如下：

Step	*N'*	*D(x), p(x)*	*D(u),p(u)*	*D(v),p(v)*	*D(w),p(w)*	*D(y),p(y)*	*D(z),p(z)*
0	t	∞	2,t	4,t	∞	7,t	∞
1	tu	∞	2,t	4,t	5,u	7,t	∞
2	tuv	7,v	2,t	4,t	5,u	7,t	∞
3	tuvw	7,v	2,t	4,t	5,u	7,t	∞
4	tuvwx	7,v	2,t	4,t	5,u	7,t	15,x
5	tuvwxy	7,v	2,t	4,t	5,u	7,t	15,x
6	tuvwxyz	7,v	2,t	4,t	5,u	7,t	15,x

b. u 到所有网络节点的最短路径如下：

Step	*N'*	*D(x), p(x)*	*D(t),p(t)*	*D(v),p(v)*	*D(w),p(w)*	*D(y),p(y)*	*D(z),p(z)*
	u	∞	2,u	3,u	3,u	∞	∞
	ut	∞	2,u	3,u	3,u	9,t	∞
	utv	6,v	2,u	3,u	3,u	9,t	∞
	utvw	6,v	2,u	3,u	3,u	9,t	∞
	utvwx	6,v	2,u	3,u	3,u	9,t	14,x
	utvwxy	6,v	2,u	3,u	3,u	9,t	14,x
	utvwxyz	6,v	2,u	3,u	3,u	9,t	14,x

c. v 到所有网络节点的最短路径如下：

Step	*N’*	*D(x), p(x)*	*D(u),p(u)*	*D(t),pt)*	*D(w),p(w)*	*D(y),p(y)*	*D(z),p(z)*
	v	3,v	3,v	4,v	4,v	8,v	∞
	vx	3,v	3,v	4,v	4,v	8,v	11,x
	vxu	3,v	3,v	4,v	4,v	8,v	11,x
	vxut	3,v	3,v	4,v	4,v	8,v	11,x
	vxutw	3,v	3,v	4,v	4,v	8,v	11,x
	vxutwy	3,v	3,v	4,v	4,v	8,v	11,x
	vxutwyz	3,v	3,v	4,v	4,v	8,v	11,x

d. w 到所有网络节点的最短路径如下：

Step	*N’*	*D(x), p(x)*	*D(u),p(u)*	*D(v),p(v)*	*D(t),p(t)*	*D(y),p(y)*	*D(z),p(z)*
	w	6,w	3,w	4,w	∞	∞	∞
	wu	6,w	3,w	4,w	5,u	∞	∞
	wuv	6,w	3,w	4,w	5,u	12,v	∞
	wuvt	6,w	3,w	4,w	5,u	12,v	∞
	wuvtx	6,w	3,w	4,w	5,u	12,v	14,x
	wuvtxy	6,w	3,w	4,w	5,u	12,v	14,x
	wuvtxyz	6,w	3,w	4,w	5,u	12,v	14,x

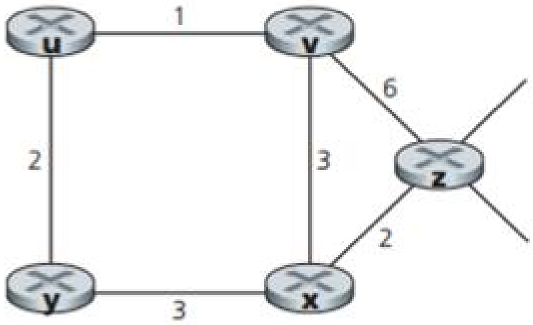
e. y 到所有网络节点的最短路径如下：

Step	*N’*	*D(x), p(x)*	*D(u),p(u)*	*D(v),p(v)*	*D(w),p(w)*	*D(t),p(t)*	*D(z),p(z)*
	y	6,y	∞	8,y	∞	7,y	12,y
	yx	6,y	∞	8,y	12,x	7,y	12,y
	yxt	6,y	9,t	8,y	12,x	7,y	12,y
	yxtv	6,y	9,t	8,y	12,x	7,y	12,y
	yxtvu	6,y	**9,t**	8,y	12,x	7,y	12,y
	yxtvuw	6,y	9,t	8,y	12,x	7,y	12,y
	yxtvuwz	6,y	9,t	8,y	12,x	7,y	12,y

f. z 到所有网络节点的最短路径如下：

Step	*N’*	*D(x), p(x)*	*D(u),p(u)*	*D(v),p(v)*	*D(w),p(w)*	*D(y),p(y)*	*D(t),p(t)*
	z	8,z	∞	∞	∞	12,z	∞
	zx	8,z	∞	11,x	14,x	12,z	∞
	zxv	8,z	14,v	**11,x**	14,x	12,z	15,v
	zxvy	8,z	14,v	11,x	14,x	**12,z**	15,v
	zxvyu	8,z	**14,v**	11,x	14,x	12,z	15,v
	zxvyuw	8,z	14,v	11,x	**14,x**	12,z	15,v
	zxvyuwt	8,z	14,v	11,x	**14,x**	12,z	15,v

P28. Consider the network shown below, and assume that each node initially knows the costs to each of its neighbors. Consider the distance-vector algorithm and show the distance table entries at node z.



答：利用距离向量算法，节点 Z 除本身临节点只有 x，z。其距离表表项如下：

From		Cost to u	V	x	y	z
	v	∞	∞	∞	∞	∞
	x	∞	∞	∞	∞	∞
	z	∞	6	2	∞	0

From		Cost to u	V	x	y	z
	v	1	0	3	∞	6
	x	∞	3	0	3	2
	z	7	5	2	5	0

From		Cost to u	V	x	y	z
	v	1	0	3	3	5
	x	4	3	0	3	2
	z	7	5	2	5	0

From		Cost to u	V	x	y	z
	v	1	0	3	3	5
	x	4	3	0	3	2
	z	6	5	2	5	0

P29. Consider a general topology (that is, not the specific network shown above) and a synchronous version of the distance-vector algorithm. Suppose that at each iteration, a node exchanges its distance vectors with its neighbors and receives their distance vectors. Assuming that the algorithm begins with each node knowing only the costs to its immediate neighbors, what is the maximum number of iterations required before the distributed algorithm converges? Justify your answer.

答：用语言表达这个问题有一些模棱两可，我们这样描述："当这个算法开始运行时从邻居迭代的次数"(即假定这些节点一开始拥有的信息是他们最近邻居的费用)。我们假设这个算法同步运行(也就是，在一个步骤里，所有节点同时计算并且改变他们的距离向量表)。

在每一次迭代中，一个节点接受来自它邻居的距离向量并且改变它的距离向量表。因此，如果有一个节点 A，其邻居节点为 B，在一次迭代之后，B 所有的邻居(距离 A 一跳或者两跳路由器)将知道一跳或者两跳到 A 的最短花费路径(即 B 告知其邻居它们到达 A 的费用)。

用 d 表示网络的"diameter(直径)"：就是在网络中任意两个节点之间没有循环的最长路径的长度。运用前面的推理，在$d-1$次迭代之后所有节点将知道 d 的最短花费路径或者到其它路由器之间更少的跳数，由于比 d 跳路由器更长的路径有循环(因此随着循环的移除比 d 跳路由器有更大的花销)，这个算法合计起来一共有$d-1$次迭代。

此外，除非特别指定了链路费用的限制，否则因为链路费用变化而调用 DV 算法，不会影响到完成迭代计算所需迭代次数的限制。

P30. Consider the network fragment shown below. x has only two attached neigh- bors, w and y. w has a minimum-cost path to destination u (not shown) of 5, and y has a minimum-cost path to u of 6. The complete paths from w and y to u (and between w and y) are not shown. All link costs in the network have strictly positive integer values.

a. Give x's distance vector for destinations w, y, and u.

b. Give a link-cost change for either c(x,w) or c(x,y) such that x will inform its neighbors of a new minimum-cost path to u as a result of executing the distance-vector algorithm.

c. Give a link-cost change for either c(x,w) or c(x,y) such that x will not inform its neighbors of a new minimum-cost path to u as a result of exe- cuting the distance-vector algorithm.

答：a. 由题意得：

$D_x(w) = \min\{c(x,y) + D_y(w), c(x,w) + D_w(w)\} = \min\{5+2, 2+0\} = 2$

$D_x(y) = \min\{c(x,y) + D_y(y), c(x,w) + D_w(y)\} = \min\{5+0, 2+2\} = 4$

$D_x(u) = \min\{c(x,y) + D_y(u), c(x,w) + D_w(u)\} = \min\{4+6, 2+5\} = 7$

b. 首先考虑如果$c(x,y)$变化将会发生什么。如果$c(x,y)$变大或者变小(只要满足：$c(x,y) \geq 1$)，从 x 到 u 的最低花费路径的最小值仍然为 7。因此$c(x,y)$的变化(只要有：$c(x,y) \geq 1$)不会导致 x 通知其邻居它的到 u 的最低花费路径有任何变化。

如果$c(x,y) = \delta < 1$，那么现在从 x 到 u 的最低花费路径将通过 y 大小为：$\delta + 6$。

接下来考虑$c(x,w)$变化。如果$c(x,w) = \varepsilon \leq 1$，那么到 u 的最低花费路径仍将通过 w 并且其花费变为$5+\varepsilon$；x 将会通知其邻居这个新的最低费用。如果$c(x,w) = \varepsilon > 6$，那么现在的最低花费路径将通过 y 大小为 11；x 将再次通知其邻居这个新的最低花费路径。

c. 在链路中任何$c(x,y)$的变化(只要$c(x,y) \geq 1$)不会导致 x 通知其邻居到 u 的一个新的最低花费路径。

P31. Consider the three-node topology shown in Figure 4.30. Rather than having the link costs shown in Figure 4.30, the link costs are c(x,y) = 3, c(y,z) = 6, c(z,x) = 4. Compute the distance tables after the initialization step and after each iteration of a synchronous version of the distance-vector algorithm (as we did in our earlier discussion of Figure 4.30).

答：如下：

节点 x 表

Cost to Cost to

		x	y	z
	x	0	3	4
From	y	∞	∞	∞
	z	∞	∞	∞

		x	y	z
	x	0	3	4
From	y	3	0	6
	z	4	6	0

节点 y 表

Cost to		x	y	z
	x	∞	∞	∞
From	y	3	0	6
	z	∞	∞	∞

Cost to		x	y	z
	x	0	3	4
From	y	3	0	6
	z	4	6	0

节点 z 表

Cost to		x	y	z
	x	∞	∞	∞
From	y	∞	∞	∞
	z	4	6	0

Cost to		x	y	z
	x	0	3	4
From	y	3	0	6
	z	4	6	0

P32. Consider the count-to-infinity problem in the distance vector routing. Will the count-to-infinity problem occur if we decrease the cost of a link? Why? How about if we connect two nodes which do not have a link?

答：不会，这是因为减少链路费用不会导致产生一个环或者回路(是由在那条链路两个节点之间的下一跳的相关联的路由器产生的)。用一条链路将两个节点连接和将这条链路的费用由无穷大变为有限大小是等效的。

P33. Argue that for the distance-vector algorithm in Figure 4.30, each value in the distance vector D(x) is non-increasing and will eventually stabilize in a finite number of steps.

答：在每一步，一个节点的距离向量的每一次更新都要基于 Bellman-Ford 方程，也就是，在距离向量表中仅仅是减少这些最低花费路径的值，不会增大，对于增大的我们忽略。如果没有更新，该节点不会发送消息出去。因此，D(x)是不会增加的。因为这些花费是有限的，所以最终的距离向量将稳定在有限的步骤。

P34. Consider Figure 4.31. Suppose there is another router w, connected to router y and z. The costs of all links are given as follows: c(x, y) = 4, c(x, z) = 50, c(y, w) = 1, c(z, w) = 1, c(y, z) = 3. Suppose that poisoned reverse is used in the distance-vector routing algorithm.

a. When the distance vector routing is stabilized, router w, y, and z inform their distances to x to each other. What distance values do they tell each other?

b. Now suppose that the link cost between x and y increases to 60. Will there be a count-to-infinity problem even if poisoned reverse is used? Why or why not? If there is a count-to-infinity problem, then how many iterations are needed for the distance-vector routing to reach a stable state again? Justify your answer.

c. How do you modify c(y,z) such that there is no count-to-infinity problem at all if c(y,x) changes from 4 to 60?

答：a. 如下：

Router z	Informs w, $D_z(x)=\infty$
	Informs y, $D_z(x)=6$
Router w	Informs y, $D_w(x)=\infty$
	Informs z, $D_w(x)=5$
Router y	Informs w, $D_y(x)=4$
	Informs z, $D_y(x)=4$

b. 是的，将会出现 count-to-infinity 问题。下面的表格显示了选路聚集过程，假定在 t_0 时刻，链路费用变化发生了；在t_1时刻 y 更新其距离向量并且通知其邻居 w 和 z，在下面的表格中，“→”表示“通知”。

time	t_0	t_1	t_2	t_3	t_4
Z	→w, $D_z(x)=\infty$ →y, $D_z(x)=6$		No change	→w, $D_z(x)=\infty$ →y, $D_z(x)=11$	
W	→y, $D_w(x)=\infty$ →z, $D_w(x)=5$		→y, $D_w(x)=\infty$ →z, $D_w(x)=10$		No change
Y	→w, $D_y(x)=4$ →z, $D_y(x)=4$	→w, $D_y(x)=9$ →z, $D_y(x)=\infty$		No change	→w, $D_y(x)=14$ →z, $D_y(x)=\infty$

我们可以看到 w，z，y 形成了一个回路在它们计算到达路由器 x 的费用过程中。如果我们持续迭代会得到下表，在时间t_{27}，z 检测到它到 x 的最低花费为 50，通过和 x 直接相连得到。在时间t_{29}，w 学习到它通过 z 到 x 的最低花费为 51。在时刻t_{30}，y 更新它的到达 x 的最低费用为 52(通过 w)。最后，在时刻t_{31}，没有更新，选路稳定了下来。表格如下：

time	t_{27}	t_{28}	t_{29}	t_{30}	t_{31}
Z	→ w, $D_z(x)=50$ → y, $D_z(x)=50$				via w, ∞ via y, 55 via z, 50
W		→ y, $D_w(x)=\infty$ → z, $D_w(x)=50$	→ y, $D_w(x)=51$ → z, $D_w(x)=\infty$		via w, ∞ via y, ∞ via z, 51
Y		→ w, $D_y(x)=53$ → z, $D_y(x)=\infty$		→ w, $D_y(x)=\infty$ → z, $D_y(x)=52$	via w, 52 via y, 60 via z, 53

c. 删除 y 和 z 之间的链路。

P35. Describe how loops in paths can be detected in BGP.

答：因为从一个自治系统到达一个 BGP 的目的地全部的路径信息是可行的，所以检测到环路非常简单—如果一个 BGP 在 AS PATH 中的对等方收到一个包含它自己的 AS 号码，然后它就会使用在环路中起作用的选路，拒绝进行通告。

P36. Will a BGP router always choose the loop-free route with the shortest AS- path length? Justify your answer.

答：选择的路径不一定必须是最短的 AS-path，回想在路由选择进程中需要考虑很多问题。由于商业原因，一个长的免费环路更宁愿是一个较短的免费环路。比如，一个 AS 更倾向发送“traffic”到一个邻居而不是到一个有更短的 AS 距离的邻居。

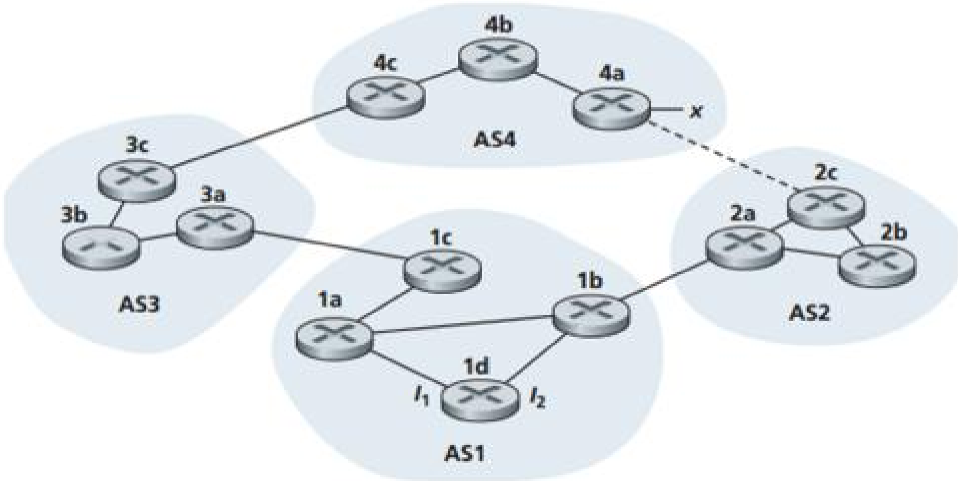
P37. Consider the network shown below. Suppose AS3 and AS2 are running OSPF for their intra-AS routing protocol. Suppose AS1 and AS4 are running RIP for their intra-AS routing protocol. Suppose eBGP and iBGP are used for the inter-AS routing protocol. Initially suppose there is no physical link between AS2 and AS4.

a. Router 3c learns about prefix x from which routing protocol: OSPF, RIP, eBGP, or iBGP?

b. Router 3a learns about x from which routing protocol?

c. Router 1c learns about x from which routing protocol?

d. Router 1d learns about x from which routing protocol?



答：a. 由于 x 在 AS4 中，3c 为 AS3 的网关路由器，所以 3c 通过 eBGP 学习到前缀 x；

b. 3a 通过 iBGP 从 3c 处学习到前缀；

c. 1c 通过 eBGP 从 3a 处学习到前缀；

d. 1d 通过 iBGP 从 1c 处学习到前缀。

P38. Referring to the previous problem, once router 1d learns about x it will put an entry (x, I) in its forwarding table.

a. Will I be equal to I1 or I2 for this entry? Explain why in one sentence.

b. Now suppose that there is a physical link between AS2 and AS4, shown by the dotted line. Suppose router 1d learns that x is accessible via AS2 as well as via AS3. Will I be set to I1 or I2? Explain why in one sentence.

c. Now suppose there is another AS, called AS5, which lies on the path between AS2 and AS4 (not shown in diagram). Suppose router 1d learns that x is accessible via AS2 AS5 AS4 as well as via AS3 AS4. Will I be set to I1 or I2? Explain why in one sentence.

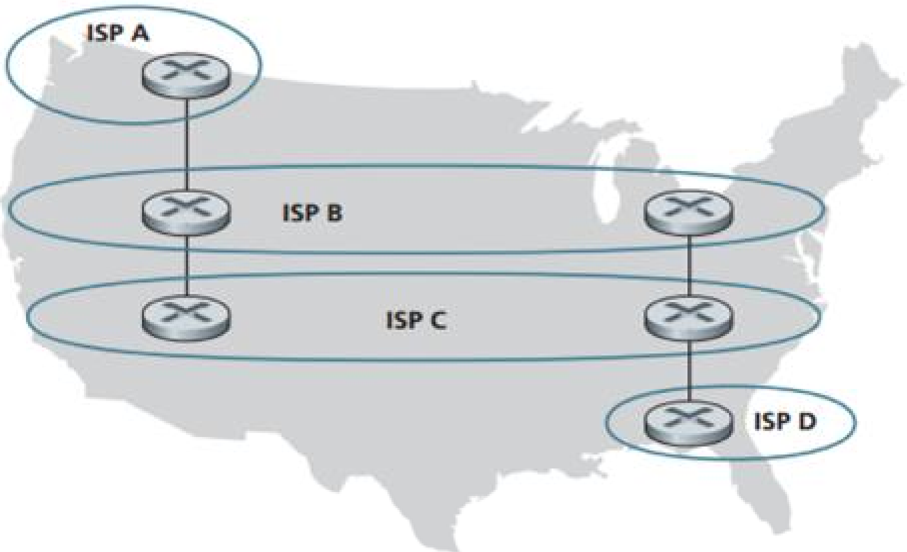
答：AS1 运行内部选路协议 RIP，表项（x，1）即为目标网络和下一跳网络，则

a. 对这个表项而言，I 应该设置为 I_1。因为这个接口开启了从内部路由器 1d 到网关路由器 1c 最低花费路径。

b. I2。通过 AS2 以及 AS3 有相同的 AS-PATH 长度。但是从 I2 开始有最近的 NEXT-TOP 路由器，即通过路由器跳数更少。

c. 选择 I1，因为其 AS-PATH 更短，比 I2 少 1。

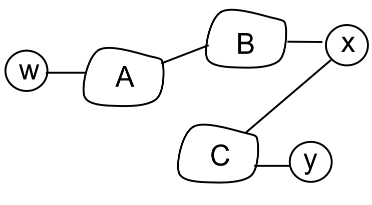
P39. Consider the following network. ISP B provides national backbone service to regional ISP A. ISP C provides national backbone service to regional ISP D. Each ISP consists of one AS. B and C peer with each other in two places using BGP. Consider traffic going from A to D. B would prefer to hand that traffic over to C on the West Coast (so that C would have to absorb the cost of carrying the traffic cross-country), while C would prefer to get the traffic via its East Coast peering point with B (so that B would have carried the traffic across the country). What BGP mechanism might C use, so that B would hand over A-to-D traffic at its East Coast peering point? To answer this question, you will need to dig into the BGP specification.



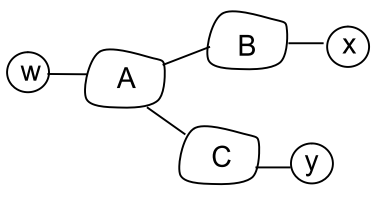
答：C 要使 B 处理所有东岸的 B 到 D 流量的一个方法是，C 只从它的东岸对等点公布到 D 的路由。

P40. In Figure 4.42, consider the path information that reaches stub networks W, X, and Y. Based on the information available at W and X, what are their respective views of the network topology? Justify your answer. The topology view at Y is shown below.

答：如下：



X 所见到的拓扑视图



W 所见到的拓扑视图

在上面的解答中，X 不知道链路 AC 因为 X 没有收到一个选路通告，它到达 w 或者 y 并且包含 AC 链路(即 X 没有收到通告，这个通告在链路上到达目的地并且包含 AS A 和 AS C)。

P41. Consider Figure 4.42. B would never forward traffic destined to Y via X based on BGP routing. But there are some very popular applications for which data packets go to X first and then flow to Y. Identify one such application, and describe how data packets follow a path not given by BGP routing.

答：BitTorrent 文件共享和 Skype P2P(网络电话)应用程序。

考虑 BitTorrent 文件共享网络有三个对等方 1，2，3 独立的在三个桩网络 W，X 和 Y 中。由于 BitTorrent 的文件共享机制，极有可能对等方 2 从对等方 1 得到数据块然后将这些数据块转发给对等方 3。这和 B 将数据最终转发到桩网络 Y 是等价的。

P42. In Figure 4.42, suppose that there is another stub network V that is a customer of ISP A. Suppose that B and C have a peering relationship, and A is a customer of both B and C. Suppose that A would like to have the traffic destined to W to come from B only, and the traffic destined to V from either B or C. How should A advertise its routes to B and C? What AS routes does C receive?

答：A 应该通告 B 两个选路，AS-paths：A-W 和 A-V；A 应该通告 C 一个选路：A-V。

C 接收的选路 AS paths 有：B-A-W，B-A-V 和 A-V。

P43. Suppose ASs X and Z are not directly connected but instead are connected by AS Y. Further suppose that X has a peering agreement with Y, and that Y has a peering agreement with Z. Finally, suppose that Z wants to transit all of Y's traffic but does not want to transit X's traffic. Does BGP allow Z to implement this policy?

答：由于 Z 想传输 Y 的流量，Z 会发送选路通告给 Y。在这种情况下，当 Y 有一个数据报并且其指定的 IP 可以通过 Z 到达，Y 就会选择通过 Z 发送这个数据报。然而，如果 Z 将选路通告发送给 Y，Y 可以将这些选路通告再发送给 X。那么在这种情况下，Z 将什么也做不了来阻止来自 X 的流量通过 Z 来传输。

P44. Consider the seven-node network (with nodes labeled t to z) in Problem P26. Show the minimal-cost tree rooted at z that includes (as end hosts) nodes u, v, w, and y. Informally argue why your tree is a minimal-cost tree.

答：最低费用树如下：z 连接到 y 通过 x 费用为：14(=8+6)；

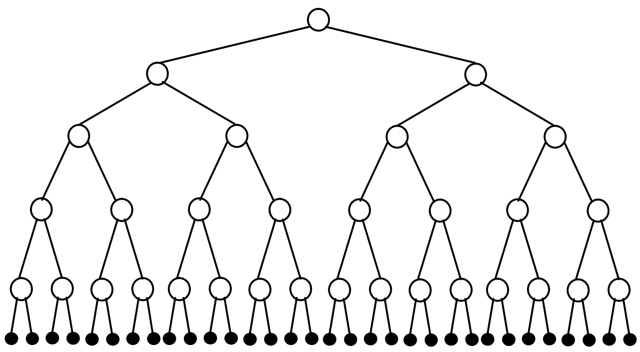
z 连接到 v 通过 x，其费用为：11(=8+3)；

z 连接到 u 通过 x，v，其费用为：14(=8+3+3)；

z 连接到 w 通过 x，v，u，其费用为：17(=8+3+3+3)。这由 Prim 算法得到可以画出最低费用树。

P45. Consider the two basic approaches identified for achieving broadcast, unicast emulation and network-layer (i.e., router-assisted) broadcast, and suppose spanning-tree broadcast is used to achive network-layer broadcast. Consider a single sender and 32 receivers. Suppose the sender is connected to the receivers by a binary tree of routers. What is the cost of sending a broadcast packet, in the cases of unicast emulation and network-layer broadcast, for this topology? Here, each time a packet (or copy of a packet) is sent over a single link, it incurs a unit of cost. What topology for interconnecting the sender, receivers, and routers will bring the cost of unicast emulation and true net- work-layer broadcast as far apart as possible? You can choose as many routers as you'd like.

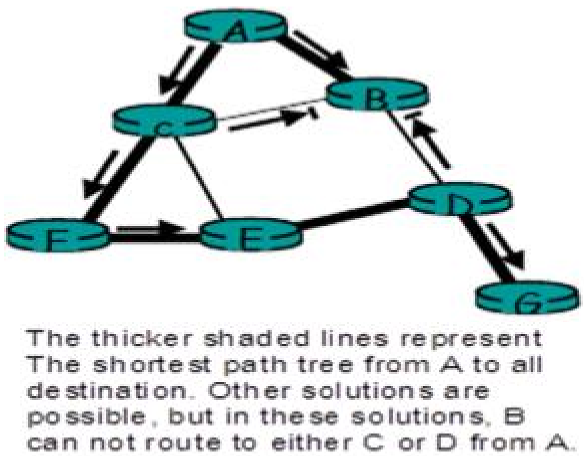
答：如图所示：



分析：32 个用户通过路由器二叉树连接到发送方。采用网络层广播的话，消息拷贝在每个链路中只传播以一次。一共有$62(2+4+8+16+32)$个链路交叉(所以费用为 64)。采用单播模拟的话，发送方将信息拷贝发送到任意一个接收方都要经过一条 5 跳的路径。一共有$160(5*32)$个链路交叉(费用 160)。所有接收方连成一线，发送方在线的一侧，这种网络拓扑使单播模拟与真正的网络层广播产生的费用相差最大。

P46. Consider the operation of the reverse path forwarding (RPF) algorithm in Figure 4.44. Using the same topology, find a set of paths from all nodes to the source node A (and indicate these paths in a graph using thicker-shaded lines as in Figure 4.44) such that if these paths were the least-cost paths, then node B would receive a copy of A's broadcast message from nodes A, C, and D under RPF.

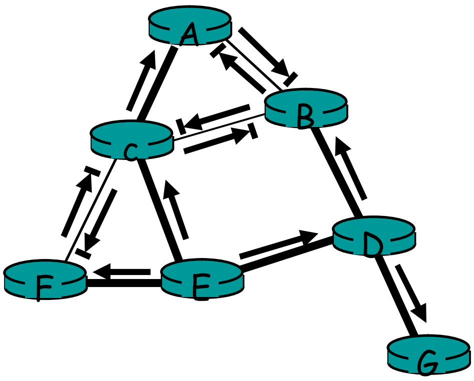
答：如下：



图中粗线指出了从A到所有目标节点的最短路径树。还可能有其它的解决方法。在这一树中，B没有到达C或D的路由。

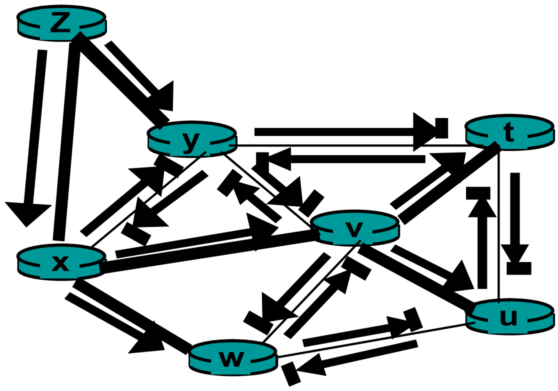
P47. Consider the topology shown in Figure 4.44. Suppose that all links have unit cost and that node E is the broadcast source. Using arrows like those shown in Figure 4.44 indicate links over which packets will be forwarded using RPF, and links over which packets will not be forwarded, given that node E is the source.

答：如下图：



P48. Repeat Problem P47 using the graph from Problem P26. Assume that z is the broadcast source, and that the link costs are as shown in Problem P26.

答：如下图：



P49. Consider the topology shown in Figure 4.46, and suppose that each link has unit cost. Suppose node C is chosen as the center in a center-based multicast routing algorithm. Assuming that each attached router uses its least-cost path to node C to send join messages to C, draw the resulting center-based routing tree. Is the resulting tree a minimum-cost tree? Justify your answer.

答：在初始的图中显示的关于拓扑的基于中心的生成树连接方式为：A 和 C 相连，B 和 C 相连，E 和 C 相连，F 和 C 相连(都是直接相连)。D 通过 E 连接到 C，G 通过 D，E 连接到 C。这个基于中心的生成树和图中显示的最低费用生成树是不同的。

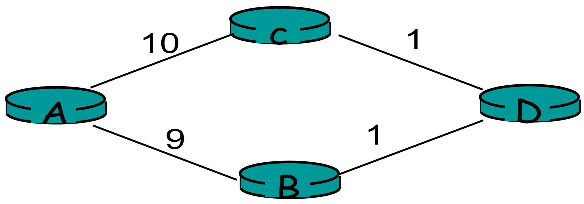
P50. Repeat Problem P49, using the graph from Problem P26. Assume that the center node is v.

答：在初始的图中显示的关于拓扑的基于中心的生成树连接方式为：t 和 v 相连，u 和 v 相连，w 和 v 相连，x 和 v 相连(都是直接相连)。z 通过 x 连接到 v。这个基于中心的生成树和图中显示的最低费用生成树是不同的。

P51. In Section 4.5.1 we studied Dijkstra's link-state routing algorithm for computing the unicast paths that are individually the least-cost paths from the source to all destinations. The union of these paths might be thought of as forming a least-unicast-cost path tree (or a shortest unicast path tree, if all link costs are identical). By constructing a counterexample, show that the least-cost path tree is not always the same as a minimum spanning tree.

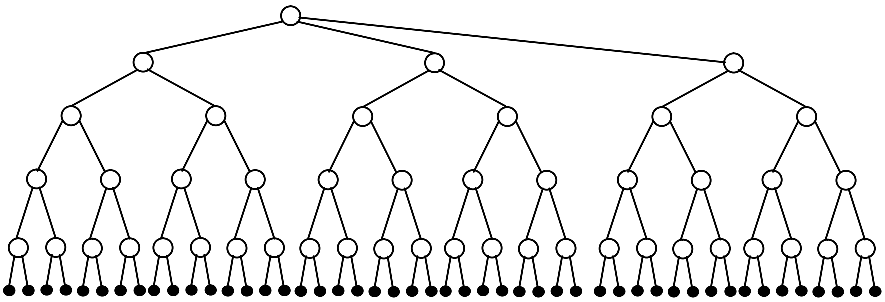
答：这个网络的 Dijkstra 链路状态选路算法如下，A 是源节点使得最低单播费用路径树的链

接为 AC，AB 和 BD 总共花费为 20，最小生成树的链路为 AB，BD 和 DC，总花费为 11。



P52. Consider a network in which all nodes are connected to three other nodes. In a single time step, a node can receive all transmitted broadcast packets from its neighbors, duplicate the packets, and send them to all of its neighbors (except to the node that sent a given packet). At the next time step, neighboring nodes can receive, duplicate, and forward these packets, and so on. Sup- pose that uncontrolled flooding is used to provide broadcast in such a network. At time step t, how many copies of the broadcast packet will be transmitted, assuming that during time step 1, a single broadcast packet is transmitted by the source node to its three neighbors?

答：在时间步 1 之后，即第二个时间段(time step)期间传输了 3 个拷贝，在第三个时间段期间传输了 6 个拷贝，在第四个时间段期间传输了 12 个拷贝，以此类推传输。在第 k 个时间段传输的分组个数为：$3 \times 2^{k-1}$。如下图：



P53. We saw in Section 4.7 that there is no network-layer protocol that can be used to identify the hosts participating in a multicast group. Given this, how can multicast applications learn the identities of the hosts that are participating in a multicast group?

答：这个协议必须建立在应用层。比如，一个应用程序可能会用应用层的报文周期性的向其他组成员多播它的身份。

P54. Design (give a pseudo-code description of) an application-level protocol that maintains the host addresses of all hosts participating in a multicast group. Specifically identify the network service (unicast or multicast) that is used by your protocol, and indicate whether your protocol is sending messages in- band or out-of-band (with respect to the application data flow among the multicast group participants) and why.

答：一个简单的允许所有组里成员去了解其它成员的身份的应用层协议，这个协议满足这个应用层程序的每一个具体实例，比如它向其他成员发送一个包含其身份的多播报文段。这个协议以 inband(同一宽带内)方式发送报文段，因为这个多播频道用于分发标识信息和数据就在这个应用程序本身上。同带信号(in-band signaling)传输的使用充分利用了多播分布式机制，引导了一个十分简易的设计。

P55. What is the size of the multicast address space? Suppose now that two multicast groups randomly choose a multicast address. What is the probability that they choose the same address? Suppose now that 1,000 multicast groups are ongoing at the same time and choose their multicast group addresses at random. What is the probability that they interfere with each other?

答：一共有$32-4=28$位可用于多播地址，因此多播的地址空间为：$\mathrm{N}=2^{28}$。

两个多播组选择相同的地址概率为：

$$\frac{1}{N}=2^{-28}=3.73\times10^{-9}$$

1000 个多播组选择相同地址的概率为：

$$\frac{N\times(N-1)\times(N-2)\times\cdots\times(N-999)}{N^{1000}}=\left(1-\frac{1}{N}\right)\left(1-\frac{2}{N}\right)\cdots\left(1-\frac{999}{N}\right)$$

因为 N 值很大，忽略叉乘 $\frac{1}{N}\times\frac{2}{N}\times\cdots\times\frac{999}{N}$，得到近似结果为：

$$1-\left(\frac{1+2+\cdots+999}{N}\right)=1-\frac{999\times1000}{2N}=0.998$$

Chapter 5 The Link Layer: Links, Access Networks, and LANs

P1. Suppose the information content of a packet is the bit pattern 1110 0110 1001 1101 and an even parity scheme is being used. What would the value of the field containing the parity bits be for the case of a two-dimensional parity scheme? Your answer should be such that a minimum-length checksum field is used.

答：如下，最后一列和最后一行均是校验比特：

1	1	1	0	1
0	1	1	0	0
1	0	0	1	0
1	1	0	1	1
1	1	0	0	0

P2. Show (give an example other than the one in Figure 5.5) that two-dimensional parity checks can correct and detect a single bit error. Show (give an example of) a double-bit error that can be detected but not corrected.

答：如下，假定最初的奇偶校验矩阵(two-dimensional parity matrix)是：

0	0	0	0
1	1	1	1
0	1	0	1
1	0	1	0

其中位于(2,3)的值由 1 变成了 0，则第二行第三列的奇偶校验码是错误的(该差错可检测可纠正)，如下：

0	0	0	0
1	1	0	1
0	1	0	1
1	0	1	0

假设第二行第二列，第二行第三列分别有一个单比特差错。第二行的奇偶校验码是正确的，但是第二列和第三列的奇偶校验码是错误的，但我们不能检测出错误发生在哪一行（可检测但不能纠正，意思是可以检测到出现错误，但是无法精确定位具体位置）！如下：

0	0	0	0
1	0	0	1
0	1	0	1
1	0	1	0

综上所述，二维奇偶校验方案，它能够检测和纠正一个分组中一比特的错误，能够检测一个分组中二比特的错误但是不能纠正。

P3. Suppose the information portion of a packet (D in Figure 5.3) contains 10 bytes consisting of the 8-bit unsigned binary ASCII representation of string “Networking.” Compute the Internet checksum for this data.

答：如下：

01001100 01101001
+01101110 01101011

10111010 11010100
+00100000 01001100

11011011 00100000
+01100001 01111001

00111100 10011010 (overflow, then wrap around)
+01100101 01110010

10100010 00001100

和的反码即互联网校验和为 01011101 11110011

P4. Consider the previous problem, but instead suppose these 10 bytes contain

a. the binary representation of the numbers 1 through 10.

b. the ASCII representation of the letters B through K (uppercase).

c. the ASCII representation of the letters b through k (lowercase). Compute the Internet checksum for this data.

答：a. 计算互联网校验和，把 16 比特的值全部加起来如下：

00000001 00000010
00000011 00000100
00000101 00000110
00000111 00001000
00001001 00001010

00011001 00011110

和的反码为：11100110 11100001。

b. 为了计算大写字母 B 到 K 互联网校验和，我们把这些值用 16 进制表示加起来：

01000010 01000011
01000100 01000101
01000110 01000111
01001000 01001001
01001010 01001011

10011111 10100100

和的反码为：01100000 01011011

c. 为了计算小写字母 b 到 k 互联网校验和，我们把这些值用 16 进制表示加起来：

01100010　01100011
01100100　01100101
01100110　01100111
01101000　01101001
01101010　01101011

00000000　00000101

和的反码为：11111111　11111010。

P5. Consider the 7-bit generator, G=10011, and suppose that D has the value 1010101010. What is the value of R?

答：我们用生成多项式(generator)G=10011 去除接收到的比特$\mathrm{D} \times 2^4 = 10101010100000$，我们得到商 1011011100，余数 R=0100。此外 G=10011 是标准的 CRC-4-ITU。

P6. Consider the previous problem, but suppose that D has the value

a. 1001010101.

b. 0101101010.

c. 1010100000.

答：a. 我们得到商 1000110000，以及余数$\mathrm{R} = 0000$；

b. 我们得到商 010101010101，以及余数$\mathrm{R} = 1111$；

c. 我们得到商 1011010111，以及余数$\mathrm{R} = 1001$。

P7. In this problem, we explore some of the properties of the CRC. For the generator G (=1001) given in Section 5.2.3, answer the following questions.

a. Why can it detect any single bit error in data D?

b. Can the above G detect any odd number of bit errors? Why?

答：a. 不失一般性，假设第 i 个比特出错(flipped)了，i 满足$0 \le \mathrm{i} \le \mathrm{d} + \mathrm{r} - 1$，并且假定最低的比特位是第 0 比特(assume that the least significant bit is 0th bit)。一个单独的比特错误意味着接受的数据是$\mathrm{K} = \mathrm{D} \times 2^r\ XOR\ R + 2i$。显而易见，如果我们用 K 除以 G，余数不会是 0。一般而言，如果 G 中至少包含两个比特位为 1，那么单比特的错误总会被检测出来(In general, if G contains at least two 1's, then a single bit error can always be detected.)。

b. The key insight(洞察力 直觉) here is that G can be divided(除) by 11 (binary number), but any number of odd-number(奇数) of 1's(?)(一个比特的错误) cannot be divided by 11. Thus, a sequence (not necessarily(必要的) contiguous(邻近的)) of odd-number bit errors(奇数比特的差错) cannot be divided by 11, thus it cannot be divided by G.

关键点是 G 能够被二进制的 11 整除，但是任意奇数个的二进制的 1 不能被 11 整除。因此，一个有奇数比特的差错(不必相邻)不能被 11 整除，因此它不能被 G 整除。

P8. In Section 5.3, we provided an outline of the derivation of the efficiency of slotted ALOHA. In this problem we'll complete the derivation.

a. Recall that when there are N active nodes, the efficiency of slotted ALOHA is $\mathrm{Np}(1 - \mathrm{p})^{\mathrm{N}-1}$. Find the value of p that maximizes this expression.

b. Using the value of p found in (a), find the efficiency of slotted ALOHA by letting N approach infinity. Hint: $(1-1/N)^N$ approaches 1/e as N approaches infinity.

答：a. 由题意时隙 ALOHA 的效率为：$E(p)=Np(1-p)^{N-1}$，对其进行求导可得最大值。

$$E'(p)=N(1-p)^{N-1}+Np\times(N-1)\times(1-p)^{N-2}\times(-1)$$
$$=N(1-p)^{N-2}[(1-p)-p(N-1)]=N(1-p)^{N-2}(1-pN)$$

由$E'(p)=0$可得$p=\frac{1}{N}$时有$E(p)_{max}$。

b. 由 a 可得当$p=\frac{1}{N}$是有：

$$E(p)_{max}=N\times\frac{1}{N}\times(1-\frac{1}{N})^{N-1}=\frac{(1-\frac{1}{N})^N}{1-\frac{1}{N}}$$

因为

$$\lim_{N\to\infty}(1-\frac{1}{N})^N=\frac{1}{e}\quad\lim_{N\to\infty}(1-\frac{1}{N})=1$$

所以

$$E(p)_{max}=\frac{1}{e}$$

P9. Show that the maximum efficiency of pure ALOHA is 1/(2e). Note: This problem is easy if you have completed the problem above!

答：纯 ALOHA 中，$E(p)=Np(1-p)^{2(N-1)}$，则

$E'(P)=N\cdot(1-p)^{2(N-1)}+Np\cdot(1-p)^{2N-3}\cdot(-1)\cdot(2N-2)$

$=N(1-p)^{2N-3}[(1-p)-(2N-2)p]$

令$E'(p)=0$，解得$p=\frac{1}{2N-1}$，带入 E'(p)有：

$$E'(p)_{max}=N\cdot\frac{1}{2N-1}\cdot\left(1-\frac{1}{2N-1}\right)^{2(N-1)}=\frac{N}{2N-2}\cdot(1-\frac{1}{2N-1})^{2N-1}$$

$$\lim_{N\to\infty}E'(p)_{max}=\lim_{N\to\infty}\frac{N}{2N-2}\cdot(1-\frac{1}{2N-1})^{2N-1}=\frac{1}{2}\cdot\frac{1}{e}=\frac{1}{2e}$$

P10. Consider two nodes, A and B, that use the slotted ALOHA protocol to contend for a channel. Suppose node A has more data to transmit than node B, and node A's retransmission probability pA is greater than node B's retransmission probability, pB .

a. Provide a formula for node A's average throughput. What is the total efficiency of the protocol with these two nodes?

b. If pA = 2pB , is node A's average throughput twice as large as that of node B?Why or why not? If not, how can you choose pA and pB to make that happen?

c. In general, suppose there are N nodes, among which node A has retrans- mission probability 2p and all other nodes have retransmission probability p. Provide expressions to compute the average throughputs of node A and of any other node.

答：a. 由题意 A 的平均吞吐率由时隙 ALOHA 公式，A 在传输，B 不在传输得 A 的平均吞

吐率为：

$$E(p_A) = p_A(1 - p_B)$$

A，B 两个节点总吞吐率为：

$$E(p) = E(p_A) + E(p_B) = p_A(1 - p_B) + p_B(1 - p_A)$$

b. 当$p_A = 2p_B$时，A 的平均吞吐率为：

$$E(p_A) = 2p_B \cdot (1 - p_B) = 2p_B - 2{p_B}^2$$

B 的平均吞吐率为：

$$E(p_B) = p_B \cdot (1 - 2p_B) = p_B - 2{p_B}^2$$

显然 A 的吞吐率不是 B 的 2 倍，为了满足：$E(p_A) = 2E(p_B)$，我们有：

$$p_A(1 - p_B) = 2p_B(1 - p_A)\text{解得}$$

$$p_A = \frac{2p_B}{p_B + 1}$$

c. A 的吞吐率为：$2p \cdot (1 - p)^{N-1}$，其它节点吞吐率为：$p \cdot (1 - p)^{N-2} \cdot (1 - 2p)$。

P11. Suppose four active nodes—nodes A, B, C and D—are competing for access to a channel using slotted ALOHA. Assume each node has an infinite number of packets to send. Each node attempts to transmit in each slot with probability p. The first slot is numbered slot 1, the second slot is numbered slot 2, and so on.

a. What is the probability that node A succeeds for the first time in slot 5?

b. What is the probability that some node (either A, B, C or D) succeeds in slot 4?

c. What is the probability that the first success occurs in slot 3?

d. What is the efficiency of this four-node system?

答：a. 由题意 a 节点在第五个时隙(slot)成功传输表示，在前四个节点 A 均不在传输并且在第五个节点只有 A 进行传输其它节点没有传输。P(A)表示在一个时隙内 A 成功传输，它需要满足此时间段 A 以其概率在进行传输，B，C，D 均不在传输，则

$$P(A) = p(1 - p)(1 - p)(1 - p) = p(1 - p)^3$$

因此所求概率为：

$$(1 - p(A))^4 \cdot P(A) = (1 - p \cdot (1 - p)^3)^4 \cdot p \cdot (1 - p)^3$$

b. 由时隙 ALOHA 的定义，A，B，C，D 四个节点成功传输的概率是一样的，为$p(1 - p)^3$则：

$$p\big(\text{A, B, C, D 任意一个节点在时隙 4 成功传输的概率}\big) = 4p(1 - p)^3$$

(因为这四个事件是互斥的)

c. 由 b 任意某个节点在一个时隙成功的概率为$4p(1 - p)^3$，其失败概率为$1 - 4p(1 - p)^3$，在第三个时隙传输成功意味着在前两个时隙传输失败，在第三个时隙传输成功则所求概率为：

$$(1 - 4p(1 - p)^3)^2 \cdot 4p(1 - p)^3$$

d. 效率 $= p\big(\text{任意某个节点在时隙内传输成功}\big) = 4p(1 - p)^3$。

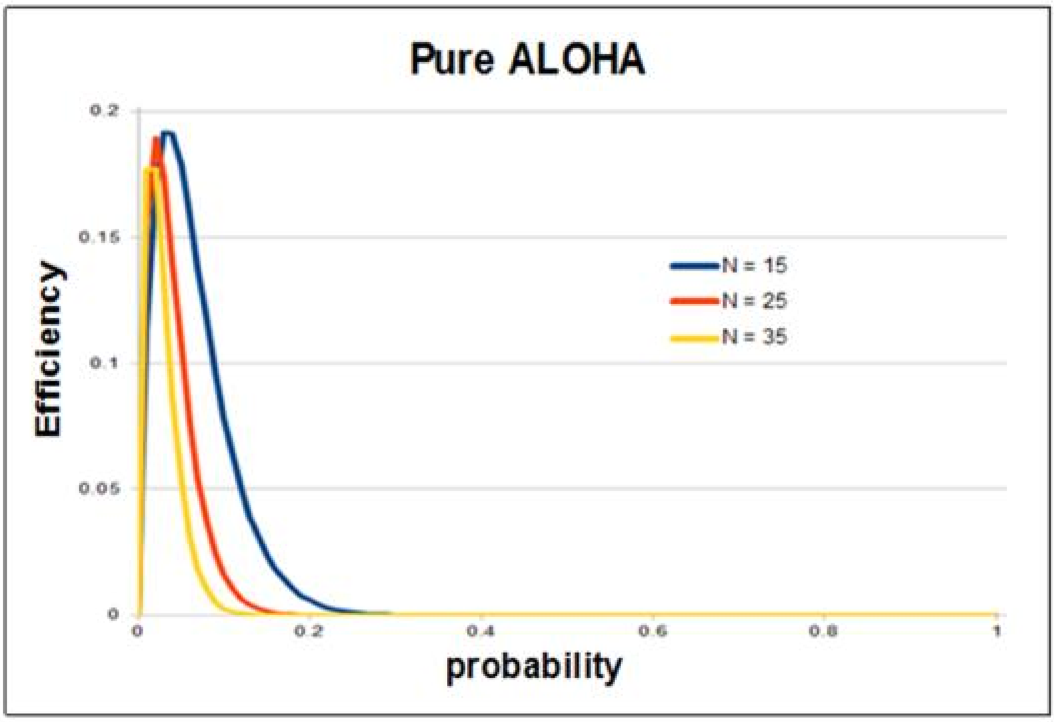
P12. Graph the efficiency of slotted ALOHA and pure ALOHA as a function of p for the following values of N:

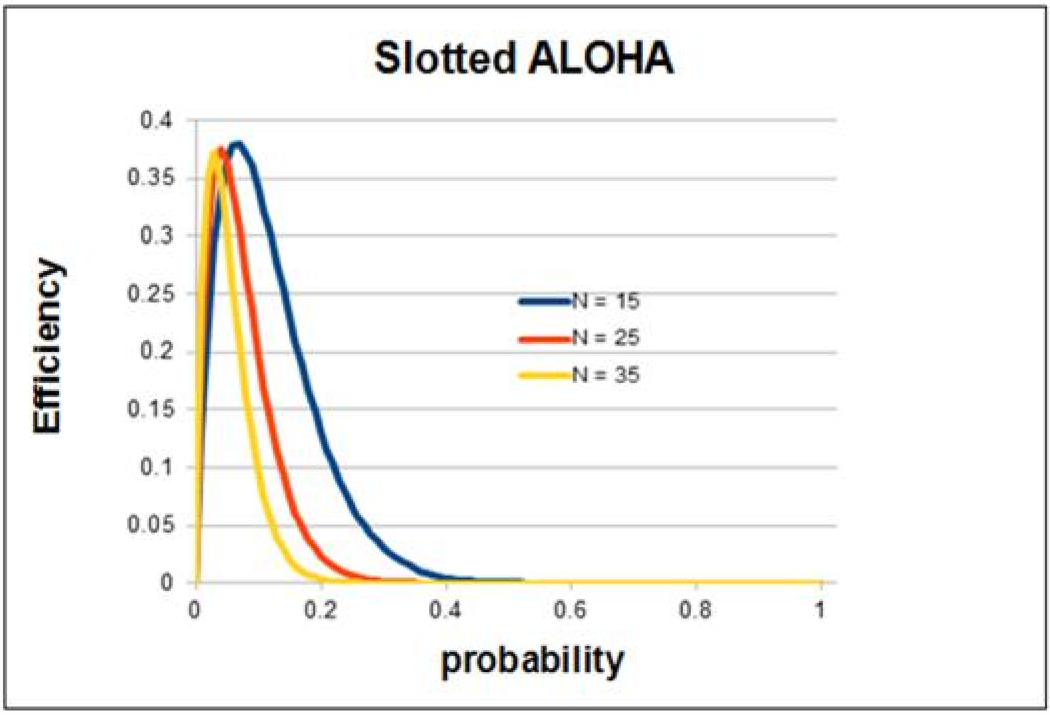
a. N=15.

b. N=25.

c. N=35.

答：如下图：





P13. Consider a broadcast channel with N nodes and a transmission rate of R bps. Suppose the broadcast channel uses polling (with an additional polling node) for multiple access. Suppose the amount of time from when a node completes transmission until the subsequent node is permitted to transmit (that is, the polling delay) is d_{poll}. Suppose that within a polling round, a given node is allowed to transmit at most Q bits. What is the maximum throughput of the broadcast channel?

答：由题意轮询周期的长度(The length of a polling round)即 N 个节点得总时延，为：

$(\text{传输时延} + \text{轮询时延}) = N \cdot (\frac{Q}{R} + d_{poll})$，在一个轮询周期传输量为 NQ 比特则该信道最大

吞吐量为：

$$\frac{NQ}{N\cdot(\frac{Q}{R}+d_{poll})}=\frac{R}{1+\frac{d_{poll}\cdot R}{Q}}$$

P14. Consider three LANs interconnected by two routers, as shown in Figure 5.33.

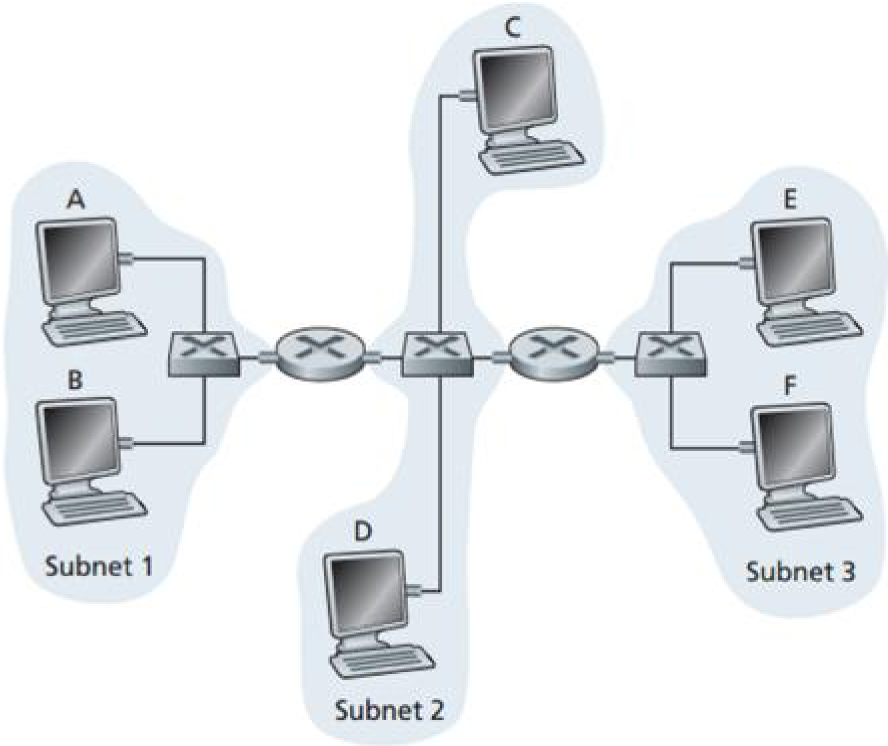


Figure 5.33 ♦ Three subnets, interconnected by routers

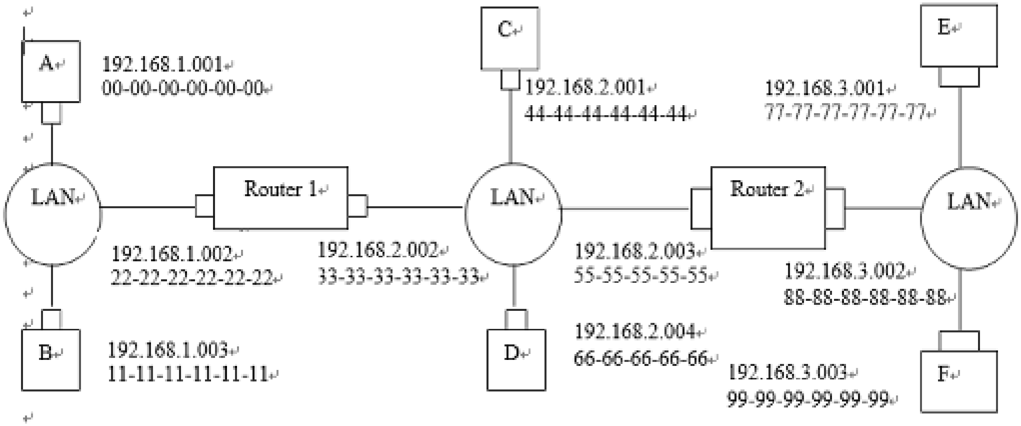
a. Assign IP addresses to all of the interfaces. For Subnet 1 use addresses of the form 192.168.1.xxx; for Subnet 2 uses addresses of the form 192.168.2.xxx; and for Subnet 3 use addresses of the form 192.168.3.xxx.

b. Assign MAC addresses to all of the adapters.

c. Consider sending an IP datagram from Host E to Host B. Suppose all of the ARP tables are up to date. Enumerate all the steps, as done for the single-router example in Section 5.4.1.

d. Repeat (c), now assuming that the ARP table in the sending host is empty (and the other tables are up to date).

答：a，b 如下：



c. 第一步：主机 E 中转发表决定这个 IP 数据报应该选路到路由器接口 192.168.3.002；

第二部：主机 E 的适配器创造一个以太帧数据包(Ethernet packet)，其内的目的 MAC 为 88-88-88-88-88-88；

第三步：路由器 2 接收到这个数据包并且提取出 IP 数据报。这个路由器的转发表决定这个数据报选路到 192.168.2.002；

第四步：然后路由器 2 通过它的 IP 为 192.168.2.003 的接口发送一个以太帧数据包，该包的源 MAC 地址为 55-55-55-55-55-55 且目的 MAC 地址为 33-33-33-33-33-33。

第五步：这个过程一直持续直到这个分组到达了主机 B。

d. 主机 E 现在必须确定接口 192.168.3.002 的 MAC 地址，因此它用一个广播的以太帧发送一个 ARP 请求分组。路由器 2 接收到这个分组并且向主机 E 发送一个 ARP 响应分组。这个 ARP 响应分组被携带在一个目的 MAC 地址为 77-77-77-77-77-77 的以太帧中。

P15. Consider Figure 5.33. Now we replace the router between subnets 1 and 2 with a switch S1, and label the router between subnets 2 and 3 as R1.

a. Consider sending an IP datagram from Host E to Host F. Will Host E ask router R1 to help forward the datagram? Why? In the Ethernet frame containing the IP datagram, what are the source and destination IP and MAC addresses?

b. Suppose E would like to send an IP datagram to B, and assume that E's ARP cache does not contain B's MAC address. Will E perform an ARP query to find B's MAC address? Why? In the Ethernet frame (containing the IP datagram destined to B) that is delivered to router R1, what are the source and destination IP and MAC addresses?

c. Suppose Host A would like to send an IP datagram to Host B, and neither A's ARP cache contains B's MAC address nor does B's ARP cache contain A's MAC address. Further suppose that the switch S1's forwarding table contains entries for Host B and router R1 only. Thus, A will broadcast an ARP request message. What actions will switch S1 perform once it receives the ARP request message? Will router R1 also receive this ARP request message? If so, will R1 forward the message to Subnet 3? Once Host B receives this ARP request message, it will send back to Host A an ARP response message. But will it send an ARP query message to ask for A's MAC address? Why? What will switch S1 do once it receives an ARP response message from Host B?

答：a. 不会。E 要检查主机 F 的 IP 地址的子网前缀，然会会发现 F 在同一个局域网(LAN)，因此 E 不会请求路由器 R1 转发这个数据报。

从 E 到 F 的以太帧包括：

源 MAC 和 IP 地址是主机 E 的 MAC 地址和 IP 地址，目的 MAC 和 IP 地址是主机 F 的 MAC 地址和 IP 地址。

b. E 不用发送 ARP 请求报文去知道 B 的 MAC 地址，因为它们不在同一个 LAN。E 只需要 B 的 IP 地址即可成功发送这个 IP 数据报。E 向 R1 发送的以太帧包括：

源 IP=E 的 IP 地址，目的 IP=B 的 IP 地址，源 MAC 地址=E 的 MAC 地址，目的 MAC 地址=路由 1 接到子网 3 的接口 MAC 地址。

c. 当交换机 S1 接收到一个目的地址是广播地址的 ARP 帧时，S1 会通过它所有的接口转发这个以太帧。并且它学习到 A 在子网 1，子网 1 连接到交换机 S1。然后 S1 会更新其转发表，表中包含一个主机 A 的条目。

是的，路由器 S2 也会接收到这个 ARP 请求报文，并且 S2 会将这个请求分组发往它的所有路由器接口。

B 不会发送 ARP 请求报文来询问 A 的 MAC 地址，因为这个地址包含在了 A 的 ARP 请求报文中。

一旦交换机 S1 接收到了 B 的响应报文，它会在其转发表中增加一个主机 B 的条目，然后丢弃这个以太帧，因为目的主机 A 和源主机 B 在同样的路由器接口(即 A 和 B 是同样的 LAN 报文段)。

P16. Consider the previous problem, but suppose now that the router between subnets 2 and 3 is replaced by a switch. Answer questions (a)–(c) in the previous problem in this new context.

答：我们将子网 2 和子网 3 之间的交换机命名为 S2，即在子网 2 和 3 之间的路由器换为交换机 S2。

a. 不用。E 会检查主机 F 的 IP 地址的前缀，并且学习到 F 在同一个 LAN 网段上。因此，E 不会向 S2 发送分组。E 到 F 的以太帧包含：

源 IP 地址=E 的 IP 地址，目的 IP 地址=F 的 IP 地址，源 MAC 地址=E 的 MAC 地址，目的 MAC 地址=F 的 MAC 地址。

b. 会的，因为 E 需要知道 B 的 MAC 地址。在这种情况下，E 会发送一个 ARP 请求分组其目的 MAC 地址是广播地址。这个请求分组会被交换机 S1 再次广播，最终被主机 B 收到。

E 发送到 S2 的以太帧包含：

源 IP 地址=E 的 IP 地址，目的 IP 地址=B 的 IP 地址，源 MAC 地址=E 的 MAC 地址，目的 MAC 地址=广播地址：FF- FF- FF- FF- FF- FF。

c. 当交换机 S1 接收到一个目的 MAC 地址为广播地址的帧时，它将通过它的所有接口转发这个以太帧。并且它学习到 A 在子网 1，子网 1 连接到交换机 S1(And it learns that A resides on Subnet 1 which is connected to S1 at the interface connecting to Subnet 1)。然后 S1 会更新其转发表，表中包含一个主机 A 的条目(表项 entry)。

是的，交换机 S2 也会收到这个 ARP 请求报文并且 S2 将会向它的所有接口广播这个报文。

B 不会发送 ARP 请求报文来询问 A 的 MAC 地址，因为它将包含在 A 的请求报文中。

一旦 S1 接收到 B 的响应报文，它会在其转发表中增加一个主机 B 的条目(表项)，然后将该以太帧丢弃，因为主机 A 和 B 是在与相同的接口相连的 LAN 网段上。

P17. Recall that with the CSMA/CD protocol, the adapter waits K • 512 bit times after a collision, where K is drawn randomly. For K = 100, how long does the adapter wait until returning to Step 2 for a 10 Mbps broadcast channel? For a 100 Mbps broadcast channel?

答：适配器(adapter)等待时间为$\mathrm{K} \cdot 512 = 100 \cdot 512\mathrm{bit}$时间。当 R=10Mbps 时，

$$\frac{5.12 \times 10^4 bits}{10 \times 10^6 bps} = 5.12msec$$

当 R=100Mbps 时，等待时间为 512μ msec。

P18. Suppose nodes A and B are on the same 10 Mbps broadcast channel, and the propagation delay between the two nodes is 325 bit times. Suppose CSMA/CD and Ethernet packets are used for this broadcast channel. Suppose node A begins transmitting a frame and, before it finishes, node B begins transmitting a frame. Can A finish transmitting before it detects that B has transmitted? Why or why not? If the answer is yes, then A incorrectly believes that its frame was successfully transmitted without a collision. Hint: Suppose at time t = 0 bits, A begins transmitting a frame. In the worst case, A transmits a minimum-sized frame of 512 + 64 bit times. So A would finish transmitting the frame at t = 512 + 64 bit times. Thus, the answer is no, if B's signal reaches A before bit time t = 512 + 64 bits. In the worst case, when does B's signal reach A?

答：在$t = 0$时刻 A 开始传输，在$t = 576$时刻 A 完成传输。在最坏的情况下，B 在$t = 324$时刻开始传输，这个时间刚好是 A 的第一个比特交付给 B 的前一个时刻。在时间$t = 324 + 325 = 649$，B 的第一个比特到达主机 A。因为$649 > 576$，A 在检测到 B 在传输之前它已经完成了传输。因此 A 会错误的认为它在传输的过程中没有发生碰撞，它成功的完成了传输。

P19. Suppose nodes A and B are on the same 10 Mbps broadcast channel, and the propagation delay between the two nodes is 245 bit times. Suppose A and B send Ethernet frames at the same time, the frames collide, and then A and B choose different values of K in the CSMA/CD algorithm. Assuming no other nodes are active, can the retransmissions from A and B collide? For our purposes, it suffices to work out the following example. Suppose A and B begin transmission at t = 0 bit times. They both detect collisions at t = 245 bit times. Suppose $K_A = 0$ and $K_B = 1$. At what time does B schedule its retransmission? At what time does A begin transmission? (Note: The nodes must wait for an idle channel after returning to Step 2—see protocol.) At what time does A's signal reach B? Does B refrain from transmitting at its scheduled time?

答：如下表：

时间，t	事件
0	A 和 B 开始传输
245	A 和 B 检测到碰撞
293	A 和 B 完成了阻塞信号的传输
293+245=538	B 的最后一个比特到达 A，A 侦听到信道空闲
538+96=634	A 开始传输
293+512=805	B 会退到第二步，它必须等待 96 比特时间并且侦听到信道空闲才能进行传输
634+245=879	A 的传输成功到达 B

P20. In this problem, you will derive the efficiency of a CSMA/CD-like multiple access protocol. In this protocol, time is slotted and all adapters are synchronized to the slots. Unlike slotted ALOHA, however, the length of a slot (in seconds) is much less than a frame time (the time to transmit a frame). Let S be the length of a slot. Suppose all frames are of constant length L = kRS, where R is the transmission rate of the channel and k is a large integer. Suppose there are N nodes, each with an infinite number of frames to send. We also assume that dprop < S, so that all nodes can detect a collision before the end of a slot time. The protocol is as follows:

• If, for a given slot, no node has possession of the channel, all nodes contend for the channel; in particular, each node transmits in the slot with probability p. If exactly one node transmits in the slot, that node takes possession of the channel for the subsequent k – 1 slots and transmits its entire frame.

• If some node has possession of the channel, all other nodes refrain from transmitting until the node that possesses the channel has finished trans- mitting its frame. Once this node has transmitted its frame, all nodes contend for the channel.

Note that the channel alternates between two states: the productive state, which lasts exactly k slots, and the nonproductive state, which lasts for a random number of slots. Clearly, the channel efficiency is the ratio of k/(k + x), where x is the expected number of consecutive unproductive slots.

a. For fixed N and p, determine the efficiency of this protocol.

b. For fixed N, determine the p that maximizes the efficiency.

c. Using the p (which is a function of N) found in (b), determine the efficiency as N approaches infinity.

d. Show that this efficiency approaches 1 as the frame length becomes large.

答：a. 设 Y 为一个随机变量，表示成功传输需要的时隙，β表示某个时隙节点在传输的概率，则：

$$\mathrm{P(Y = m) = \beta(1-\beta)}^{m-1}$$

这是一个几何分布，均值为$1/\beta$，浪费的时隙数目为：$\mathrm{X = Y - 1}$，则：

$$\mathrm{x = E[X] = E[Y-1]} = \frac{1-\beta}{\beta}$$

$$\mathrm{\beta = Np(1-p)}^{N-1}$$

$$所以\ \mathrm{x} = \frac{1-\mathrm{Np(1-p)}^{N-1}}{\mathrm{Np(1-p)}^{N-1}}$$

$$效率 = \frac{k}{k+x} = \frac{k}{k+\frac{1-\mathrm{Np(1-p)}^{N-1}}{\mathrm{Np(1-p)}^{N-1}}}$$

b. 求最大效率等价于求 x 的最小值，也即求β的最大值。从前面的练习中我们知道β取得最大值的条件是：$\mathrm{p = 1/N}$。

c. 所求为：

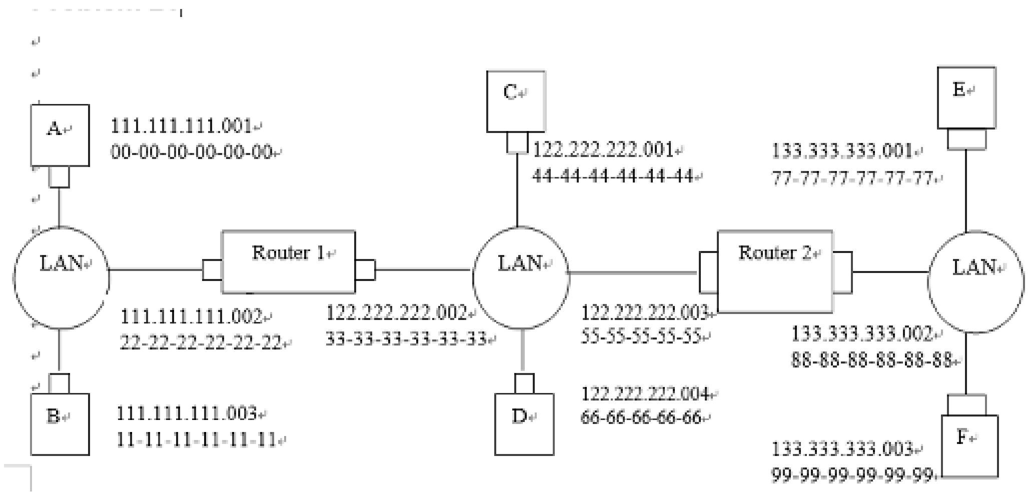
$$效率 = \frac{k}{k+\frac{1-(1-\frac{1}{N})^{N-1}}{(1-\frac{1}{N})^{N-1}}}$$

$$\lim_{N\to\infty} 效率 = \frac{\mathrm{k}}{k+\frac{1-1/e}{1/e}} = \frac{k}{k+e-1}$$

d. 显然，当$\mathrm{k}\to\infty$时，$\frac{k}{k+e-1}$趋近于 1。

P21. Consider Figure 5.33 in problem P14. Provide MAC addresses and IP addresses for the interfaces at Host A, both routers, and Host F. Suppose Host A sends a datagram to Host F. Give the source and destination MAC addresses in the frame encapsulating this IP datagram as the frame is transmitted (i) from A to the left router, (ii) from the left router to the right router, (iii) from the right router to F. Also give the source and destination IP addresses in the IP datagram encapsulated within the frame at each of these points in time.

答：如下图：



(i). 从 A 到左边路由器：

源 MAC 地址：00-00-00-00-00-00，目的 MAC 地址：22-22-22-22-22-22；源 IP 地址：111.111.111.001，目的 IP 地址：111.111.111.002。

(ii). 从左边路由器到右边路由器：

源 MAC 地址：33-33-33-33-33-33，目的 MAC 地址：55-55-55-55-55-55；源 IP 地址：122.222.222.002，目的 IP 地址：122.222.222.003。

(iii). 从右边路由器到 F：

源 MAC 地址：88-88-88-88-88-88，目的 MAC 地址：99-99-99-99-99-99；源 IP 地址：133.333.333.002，目的 IP 地址：133.333.333.003。

P22. Suppose now that the leftmost router in Figure 5.33 is replaced by a switch. Hosts A, B, C, and D and the right router are all star-connected into this switch. Give the source and destination MAC addresses in the frame encapsulating this IP datagram as the frame is transmitted (i) from A to the switch, (ii) from the switch to the right router, (iii) from the right router to F. Also give the source and destination IP addresses in the IP datagram encapsulated within the frame at each of these points in time.

答：(i) 从 A 到交换机：

Source MAC address: 00-00-00-00-00-00，

Destination MAC address: 55-55-55-55-55-55；

Source IP: 111.111.111.001，

Destination IP: 133.333.333.003。

(ii)从交换机到右边路由器：

Source MAC address: 00-00-00-00-00-00，

Destination MAC address: 55-55-55-55-55-55；

Source IP: 111.111.111.001，

Destination IP: 133.333.333.003。

(iii)从右边路由器到 F：

Source MAC address: 88-88-88-88-88-88，
Destination MAC address: 99-99-99-99-99-99；
Source IP: 111.111.111.001，
Destination IP: 133.333.333.003。

P23. Consider Figure 5.15. Suppose that all links are 100 Mbps. What is the maximum total aggregate throughput that can be achieved among the 9 hosts and 2 servers in this network? You can assume that any host or server can send to any other host or server. Why?

答：如果 9+2=11 个节点以最大传输率 100Mbps 将数据发送出去，总的聚合吞吐量为：$11 \times 100 = 1100\text{Mbps}$ 是可能的。

P24. Suppose the three departmental switches in Figure 5.15 are replaced by hubs. All links are 100 Mbps. Now answer the questions posed in problem P23.

答：每个部门集线器(hub)是一个单独的冲突域(collision domain)，有着最大吞吐量 100Mbps。这些连接到 web 服务器和邮件服务器的链路有着 100Mbps 的最大吞吐量。因此，如果三个冲突域，web 服务器和邮件服务器以他们最大的传输速率来将数据发送出去，在这 11 个端系统中最大总体聚合吞吐量为 500Mbps。

P25. Suppose that all the switches in Figure 5.15 are replaced by hubs. All links are 100 Mbps. Now answer the questions posed in problem P23.

答：如果所有的路由器都换成集线器，那么这 11 个端系统将位于同一个冲区域，这样在这 11 个端系统中最大总体聚合吞吐量为 100Mbps。

P26. Let's consider the operation of a learning switch in the context of a network in which 6 nodes labeled A through F are star connected into an Ethernet switch. Suppose that (i) B sends a frame to E, (ii) E replies with a frame to B, (iii) A sends a frame to B, (iv) B replies with a frame to A. The switch table is initially empty. Show the state of the switch table before and after each of these events. For each of these events, identify the link(s) on which the transmitted frame will be forwarded, and briefly justify your answers.

答：如下表：

事件	交换机表状态	链路分组转发到	解释说明
B 向 E 发送一个帧	交换机识别到 B 的 MAC 地址并记录它对应的接口	A，C，D，E 和 F	因为交换机表为空，不知道 E 的 MAC 地址对应的接口
E 向 B 回复一个帧	交换机识别到 D 的 MAC 地址并记录它对应的接口	B	因为交换机已经知道了 B 的 MAC 地址对应的接口
A 向 B 发送一个帧	交换机识别到 A 的 MAC 地址并记录它对应的接口	B	因为交换机已经知道了 B 的 MAC 地址对应的接口
B 向 A 回复一个帧	交换机状态不变	A	因为交换机已经知道了 A 的 MAC 地址对应的接口

P27. In this problem, we explore the use of small packets for Voice-over-IP applications. One of the drawbacks of a small packet size is that a large fraction of link bandwidth is consumed by overhead bytes. To this end, suppose that the packet consists of P bytes and 5 bytes of header.

a. Consider sending a digitally encoded voice source directly. Suppose the source is encoded at a constant rate of 128 kbps. Assume each packet is entirely filled before the source sends the packet into the network. The time required to fill a packet is the packetization delay. In terms of L, determine the packetization delay in milliseconds.

b. Packetization delays greater than 20 msec can cause a noticeable and unpleasant echo. Determine the packetization delay for L = 1,500 bytes (roughly corresponding to a maximum-sized Ethernet packet) and for L = 50 (corresponding to an ATM packet).

c. Calculate the store-and-forward delay at a single switch for a link rate of
R = 622 Mbps for L = 1,500 bytes, and for L = 50 bytes.

d. Comment on the advantages of using a small packet size.

答：a. 填充一个L · 8bits 信元需要的时间为：

$$\frac{L \times 8}{128 \times 10^3}\sec = \frac{L}{16}msec$$

b. 当 L=1500 时，分组化时延为

$$\frac{1500}{16}\text{msec} = 93.75\text{msec}$$

当 L=50 时，分组化时延为

$$\frac{50}{16}msec = 3.125msec$$

c.

$$存储转发时延 = \frac{L \times 8 + 5 \times 8}{R} = \frac{L \cdot 8 + 40}{R}$$

当 L=1500 时，存储转发时延为：

$$\frac{1500 \times 8 + 40}{622 \times 10^6}\sec \approx 19.4\mu \sec$$

当 L=50 时，存储转发时延为：

$$\frac{50 \times 8 + 40}{622 \times 10^6} sec \approx 0.7\mu\, sec < 1\mu\, sec$$

d. 对于流行的 ATM 链路速率而言，小信元长度和大信元长度的存储转发时延都很小，然而 L=1500 字节时，对于实时话音应用而言，其分组化时延太大，影响通话质量。

P28. Consider the single switch VLAN in Figure 5.25, and assume an external router is connected to switch port 1. Assign IP addresses to the EE and CS hosts and router interface. Trace the steps taken at both the network layer and the link layer to transfer an IP datagram from an EE host to a CS host (Hint: reread the discussion of Figure 5.19 in the text).

答：在 EE 部门这三台电脑从左到右 IP 地址为：111.111.1.1，111.111.1.2，111.111.1.3。网络地址为：111.111.1/24。

在 CS 部门这三台电脑从左到右 IP 地址为：111.111.2.1，111.111.2.2，111.111.2.3。网络地址为：111.111.2/24。

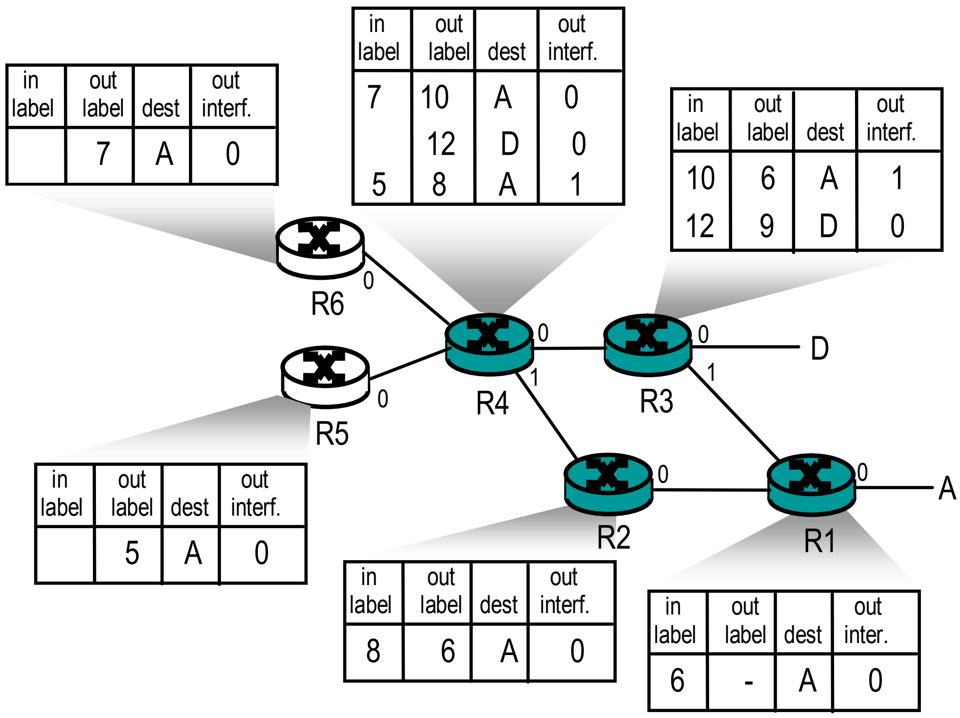
连接到端口 1 的路由器接口卡可以被分配两个子接口 IP 地址：111.111.1.0 和 111.111.2.0。第一个用于 EE 部门的子网，第二个用于 CS 部门的子网。每个 IP 地址和一个 VLAN ID 相

关联。假定 111.111.1.0 和 VLAN 11 相关，111.111.2.0 和 VLAN 12 相关。这意味着来自子网 111.111.1/24 的每个帧将被添加一个 802.1q 的标签 VLAN 11，来自子网 111.111.2/24 的每个帧将被添加一个 802.1q 的标签 VLAN 12。

假定主机 A 在 EE 部门且 IP 为 111.111.1.1，主机 B 在 CS 部门且 IP 为 111.111.2.1，A 想向 B 发送一个数据报。主机 A 首先将 IP 数据报(目的地为 111.111.2.1)封装成一个帧，帧的目的 MAC 地址等于路由器接口卡的 MAC 地址，该接口卡连接到交换机端口 1。一旦路由器接收到这个帧，路由器会上传到它的 IP 网络层，该层决定这个数据报应该通过子接口 111.111.2.0 转发到子网 111.111.2/24。然后路由器将这个 IP 数据报封装成帧并且发送到端口 1。注意到这个帧已经有一个 802.1q 的标签 VLAN 12，一旦交换机在端口 1 接收到这个帧，交换机将知道这个帧目的地为 VLAN 12，因此交换机将这个帧发送到 CS 部门的主机 B。一旦主机 B 接收到这个帧，B 会移除 802.1q 标签。

P29. Consider the MPLS network shown in Figure 5.29, and suppose that routers R5 and R6 are now MPLS enabled. Suppose that we want to perform traffic engineering so that packets from R6 destined for A are switched to A via R6-R4-R3-R1, and packets from R5 destined for A are switched via R5-R4-R2-R1. Show the MPLS tables in R5 and R6, as well as the modified table in R4, that would make this possible.

答：如下表：



R6:

in label	out label	dest	out interf.
	7	A	0

R4:

in label	out label	dest	out interf.
7	10	A	0
	12	D	0
5	8	A	1

R3:

in label	out label	dest	out interf.
10	6	A	1
12	9	D	0

R5:

in label	out label	dest	out interf.
	5	A	0

R2:

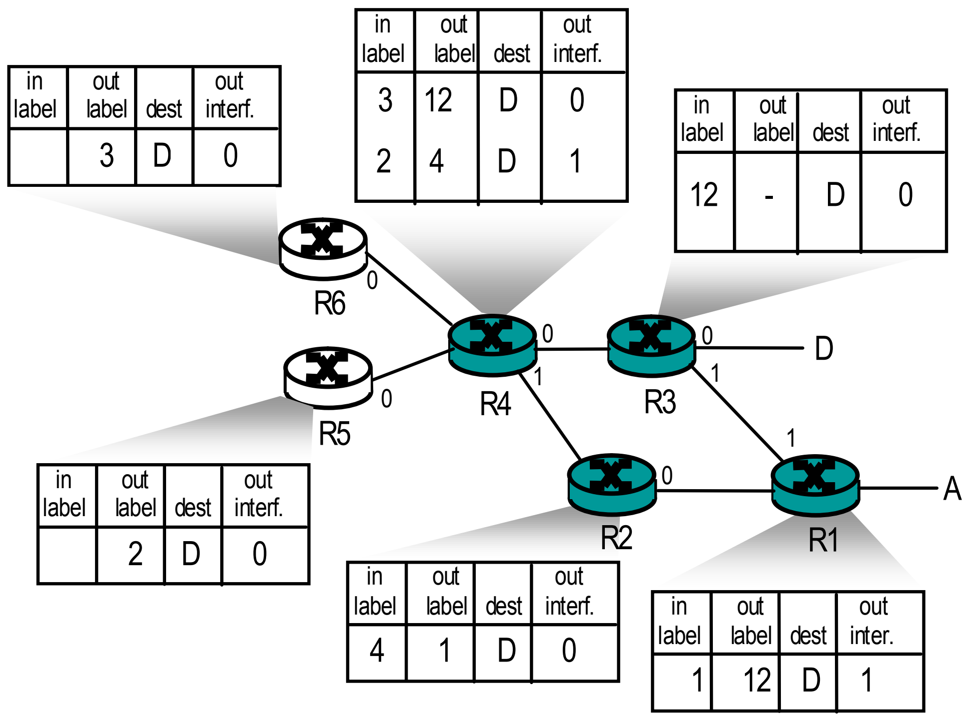
in label	out label	dest	out interf.
8	6	A	0

R1:

in label	out label	dest	out inter.
6	-	A	0

P30. Consider again the same scenario as in the previous problem, but suppose that packets from R6 destined for D are switched via R6-R4-R3, while pack- ets from R5 destined to D are switched via R4-R2-R1-R3. Show the MPLS tables in all routers that would make this possible.

答：如下图：



P31. In this problem, you will put together much of what you have learned about Internet protocols. Suppose you walk into a room, connect to Ethernet, and want to download a Web page. What are all the protocol steps that take place, starting from powering on your PC to getting the Web page? Assume there is nothing in our DNS or browser caches when you power on your PC. (Hint: the steps include the use of Ethernet, DHCP, ARP, DNS, TCP, and HTTP protocols.) Explicitly indicate in your steps how you obtain the IP and MAC addresses of a gateway router.

答：第一步：DHCP。DHCP 是一个应用层协议，DHCP 报文封装在运输层 UDP 中，UDP 封装在网络层 IP 中，进而封装在以太帧中。你的电脑首先生成一个 DHCP discover 报文，封装成的以太帧在 LAN 广播，被运行 DHCP 的服务器上被收到，分配给你的电脑一个 IP，并且提供了第一跳路由器地址，和本地 DNS 服务器的名称以及地址。

由于你的电脑 ARP 缓存初始为空，因而它用 ARP 协议获得第一跳路由器和本地 DNS 服务器的 MAC 地址。

第二部：DNS。在访问网页，如：www.google.com 时需要知道其 IP 地址，这用到 DNS。DNS 请求报文被创造，封装到 UDP，UDP 封装到 IP，IP 封装到以太帧中，在第一步已经通过 ARP 知道了第一跳路由器的 MAC 地址以及 DNS 服务器的名称以及地址，因此 DNS 报文通过本地局域网 LAN 转发，从客户机转发到到第一跳路由器，再选路到 DNS 服务器。通过各级 DNS 服务器查询获得网页的 IP 地址。

其中选路包括自治系统 AS 内部选路(如 RIP 和 OSPF 等)和 AS 之间选路(如 IS-IS 和 BGP 等)。

第三步：TCP 和 HTTP。在得到网页 IP 地址后，你的电脑向其发送 HTTP 请求报文，为了完成此步，需要建立 TCP 连接。首先，客户机对服务器创建了 TCP 套接字进程，然后 TCP 通过 SYN，SYNACK，ACK 三次握手建立连接。此过程又用到多种选路算法以及协议。

然后是 HTTP 请求报文封装到 TCP 中，TCP 封装到 IP 中，IP 封装到以太帧中。含有 HTTP 请求的 IP 数据报发送到网站，网站发送一个响应报文，该数据报通过选路返回到你的电脑。至此，可以浏览网页了。

P32. Consider the data center network with hierarchical topology in Figure 5.30.

Suppose now there are 80 pairs of flows, with ten flows between the first and ninth rack, ten flows between the second and tenth rack, and so on. Further suppose that all links in the network are 10 Gbps, except for the links between hosts and TOR switches, which are 1 Gbps.

a. Each flow has the same data rate; determine the maximum rate of a flow.

b. For the same traffic pattern, determine the maximum rate of a flow for the highly interconnected topology in Figure 5.31.

c. Now suppose there is a similar traffic pattern, but involving 20 hosts on each hosts and 160 pairs of flows. Determine the maximum flow rates for the two topologies.

答：a. 所有通过那条链路的流量均享链路容量，然后这穿过交换机 B 到达 10Gbps 访问路由器(access-router)的 80 个数据流量，每一个仅仅收到$10\text{Gbps}/80 = 125\text{Mbps}$的链路速率。

b. 在图 5.31 的拓扑中，有 4 条明显的路径存在于第一个和第三个 tier-2 路由器之间，从 racks 1-4 和 racks 9-12 为通信(traffic)总共提供 40Gbps。类似的，有 4 条链路存在于第二个和第四个 tier-2 路由器之间，从 racks 5-8 和 racks 13-16 为通信(traffic)总共提供 40Gbps。因此，总的聚合带宽是 80Gbps，平局每个流量的带宽为 1Gbps。

c. 在 TOR 交换机对之间 20 个数据流量想要共享 1Gbps，因此主机到主机答比特速率为 0.5Gbps。

P33. Consider the hierarchical network in Figure 5.30 and suppose that the data center needs to support email and video distribution among other applications. Suppose four racks of servers are reserved for email and four racks are reserved for video. For each of the applications, all four racks must lie below a single tier-2 switch since the tier-2 to tier-1 links do not have sufficient bandwidth to support the intra-application traffic. For the email application, suppose that for 99.9 percent of the time only three racks are used, and that the video application has identical usage patterns.

a. For what fraction of time does the email application need to use a fourth rack? How about for the video application?

b. Assuming email usage and video usage are independent, for what fraction of time do (equivalently, what is the probability that) both applications need their fourth rack?

c. Suppose that it is acceptable for an application to have a shortage of servers for 0.001 percent of time or less (causing rare periods of perform- ance degradation for users). Discuss how the topology in Figure 5.31 can be used so that only seven racks are collectively assigned to the two applications (assuming that the topology can support all the traffic).

答：a. 邮件和视频应用程序每百分之一时间都使用第四个 rack。

b. 两个应用程序都需要第四个 rack 的概率为：$0.001 \times 0.001 = 10^{-6}$。

c. 假定前三个 racks 用于视频，后三个 racks 用于视屏和邮件，再后面的三个 racks 用于邮件。我们假设第四个 rack 用全部的数据和软件，以用于邮件和视频应用程序。由图 5.31 的拓扑，两个应用程序有足够的内部带宽并且不会同时使用第四个 rack。从 b 部分，两个应用程序同时使用第四个 rack 的时间不超过 0.00001 %，这在 0.0001%的要求范围内。

Chapter 6 Wireless and Mobile Networks

P1. Consider the single-sender CDMA example in Figure 6.5. What would be the sender's output (for the 2 data bits shown) if the sender's CDMA code were (1, –1, 1, –1, 1, –1, 1, –1)?

答：对应数据比特 d_1 的输出为：d1 = [-1,1,-1,1,-1,1,-1,1]，对应数据比特 d_0 的输出为：d0 = [1,-1,1,-1,1,-1,1,-1]。

P2. Consider sender 2 in Figure 6.6. What is the sender's output to the channel(before it is added to the signal from sender 1), $Z_{i,m}^2$?

答：发送方 2 的输出为：$= [1,-1,1,1,1,-1,1,1]$; $[1,-1,1,1,1,-1,1,1]$。

P3. Suppose that the receiver in Figure 6.6 wanted to receive the data being sent by sender 2. Show (by calculation) that the receiver is indeed able to recover sender 2's data from the aggregate channel signal by using sender 2's code.

答：如下：

$$\mathrm{d}_2^1 = \frac{1\times1+(-1)\times(-1)+1\times1+1\times1+1\times1+(-1)\times(-1)+1\times1+1\times1}{8} = 1$$

$$\mathrm{d}_2^2 = \frac{1\times1+(-1)\times(-1)+1\times1+1\times1+1\times1+(-1)\times(-1)+1\times1+1\times1}{8} = 1$$

P4. For the two-sender, two-receiver example, give an example of two CDMA codes containing 1 and —1 values that do not allow the two receivers to extract the original transmitted bits from the two CDMA senders.

答：发送方 1：(1, 1, 1, -1, 1, -1, -1, -1)，发送方 2：(1, -1, 1, 1, 1, 1, 1, 1)。

P5. Suppose there are two ISPs providing WiFi access in a particular café, with each ISP operating its own AP and having its own IP address block.

a. Further suppose that by accident, each ISP has configured its AP to operate over channel 11. Will the 802.11 protocol completely break down in this situation? Discuss what happens when two stations, each associated with a different ISP, attempt to transmit at the same time.

b. Now suppose that one AP operates over channel 1 and the other over channel 11. How do your answers change?

答：a. 两个 AP 有不同的 SSID 和 MAC 地址。一个到达咖啡馆的无线站点(或者说终端)将和其中一个 SSID 相关联(即和其中一个 AP 关联)。关联之后，在新的站点和 AP 之间有一条虚拟链路。将接入点 AP 标记为 AP1 和 AP2，假定新的站点和 AP1 关联。当新的站点发送一个帧，它将被定位到 AP1。尽管 AP2 也接收到了这个帧，它不会对帧进行处理因为这个帧不是定位发送给到它的。因此，两个 ISP 在同样的信道(channel)能够并行(in parallel)工作。然而，两个 ISP 也将共享同一个无线带宽。如果在不同的网路提供商 ISPs 同时传输数据，会发生碰撞。对于 802.11b，两个 ISP 的最大聚合传输率是 11Mbps。

b. 现在，如果两个在不同 ISP(并且是不同的信道)的两个无线站点同时传输，不会发生碰撞，因此，应用 802.11b 的两个 ISP 的最大聚合传输率是 22Mbps。

P6. In step 4 of the CSMA/CA protocol, a station that successfully transmits a frame begins the CSMA/CA protocol for a second frame at step 2, rather than at step 1. What rationale might the designers of CSMA/CA have had in mind by having such a station not transmit the second frame immediately (if the channel is sensed idle)?

答：假定无线站点 H1 有 1000 个帧要传输(H1 可能是一个将 MP3 转发到其他站的 AP)。假定初始的 H1 是仅仅想要转发的站点，但是在它第一个帧传输到一半的时候，H2 想要传输一个帧。为了简单起见，假定每个站点可以接收到其他站点的信号(即没有隐藏终端)。在传输之前，H2 会侦听到信道繁忙，并且选择一个随即回退时间。

现在假定传输完第一个帧后，H1 回到第一步；即它想等待一段时间(DIFS)然后启动传输第二个帧。当 H2 处于回退时间等待信道空闲时，H1 将传输第二个帧。因此，在 H2 有机会接入信道传输之前，H1 应该能传输完 1000 个帧。另一方面，如果 H1 传输完一个帧后要回到第二步，它也会选择一个随机回退时间值，因此这样讲公平的给 H2 机会去传输。因此 CSMA/CA 是基于公平性的原理。

P7. Suppose an 802.11b station is configured to always reserve the channel with the RTS/CTS sequence. Suppose this station suddenly wants to transmit 1,000 bytes of data, and all other stations are idle at this time. As a function of SIFS and DIFS, and ignoring propagation delay and assuming no bit errors, calculate the time required to transmit the frame and receive the acknowledgment.

答：不含数据的帧长度为 32 字节，假定链路传输速率为 11Mbps，传输一个控制帧(比如 RTS 帧，CTS 帧或者 ACK 帧)时间为：

$\frac{(32\times8)\text{bits}}{11\text{Mbps}}=23\text{usec}$。传输数据帧的时间为：$\frac{(1000+32)\times8bits}{11Mbps}=751usec$，所以总时间为：

$$\text{DIFS} + \text{RTS} + \text{SIFS} + \text{CTS} + \text{SIFS} + \text{FRAME} + \text{SIFS} + \text{ACK}$$
$$= \text{DIFS} + 3\text{SIFS} + (3\times23 + 751)\ \text{usec} = \text{DIFS} + 3\text{SIFS} + 820\ \text{usec}$$

P8. Consider the scenario shown in Figure 6.33, in which there are four wireless nodes, A, B, C, and D. The radio coverage of the four nodes is shown via the shaded ovals; all nodes share the same frequency. When A transmits, it can only be heard/received by B; when B transmits, both A and C can hear/receive from B; when C transmits, both B and D can hear/receive from C; when D transmits, only C can hear/receive from D.

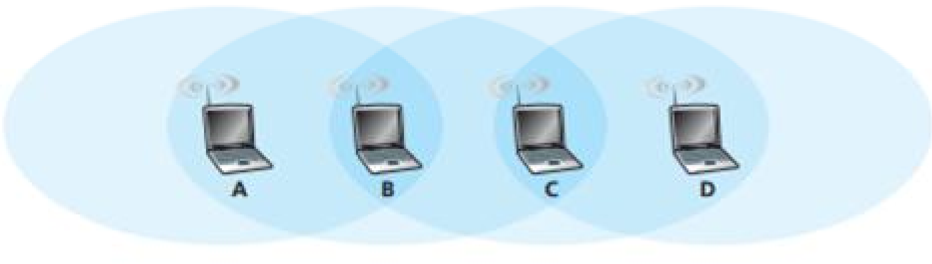


Figure 6.33 ♦ Scenario for problem P8

Suppose now that each node has an infinite supply of messages that it wants to send to each of the other nodes. If a message's destination is not an imme- diate neighbor, then the message must be relayed. For example, if A wants to send to D, a message from A must first be sent to B, which then sends the message to C, which then sends the message to D. Time is slotted, with a message transmission time taking exactly one time slot, e.g., as in slotted Aloha. During a slot, a node can do one of the following: (i) send a message; (ii) receive a message (if exactly one message is being sent to it), (iii) remain silent. As always, if a node hears two or more simultaneous transmissions, a collision occurs and none of the transmitted messages are received success- fully. You can assume here that there are no bit-level errors, and thus if exactly one message is sent, it will be received correctly by those within the transmission radius of the sender.

a. Suppose now that an omniscient controller (i.e., a controller that knows the state of every node in the network) can command each node to do whatever it (the omniscient controller) wishes, i.e., to send a message, to receive a message, or to remain silent. Given this omniscient controller, what is the maximum rate at which a data message can be transferred from C to A, given that there are no other messages between any other source/destination pairs?

b. Suppose now that A sends messages to B, and D sends messages to C. What is the combined maximum rate at which data messages can flow from A to B and from D to C?

c. Suppose now that A sends messages to B, and C sends messages to D. What is the combined maximum rate at which data messages can flow from A to B and from C to D?

d. Suppose now that the wireless links are replaced by wired links. Repeat questions (a) through (c) again in this wired scenario.

e. Now suppose we are again in the wireless scenario, and that for every data message sent from source to destination, the destination will send an ACK message back to the source (e.g., as in TCP). Also suppose that each ACK message takes up one slot. Repeat questions (a) – (c) above for this scenario.

答：a. 最大速率为 1 报文/2 时隙(message/ 2 slots)。

b. 最大速率为 2 报文/时隙(2 messages/slot)。

c. 最大速率为 1 报文/时隙(1 message/slot)。

d. (i) 1 message/slot；(ii) 2 messages/slot；(iii) 2 messages/slot；

e. (i) 1 message/4 slots

(ii) slot 1: Message A→ B, message D→ C

slot 2: Ack B→ A

slot 3: Ack C→ D

= 2 messages/ 3 slots

(iii)

slot 1: Message C→ D

slot 2: Ack D→C, message A→ B

slot 3: Ack B→ A

Repeat

= 2 messages/3 slots

P9. Describe the format of the 802.15.1 Bluetooth frame. You will have to do some reading outside of the text to find this information. Is there anything in the frame format that inherently limits the number of active nodes in an 802.15.1 network to eight active nodes? Explain.

答：略。

P10. Consider the following idealized LTE scenario. The downstream channel (see Figure 6.20) is slotted in time, across F frequencies. There are four nodes, A, B, C, and D, reachable from the base station at rates of 10 Mbps, 5 Mbps, 2.5 Mbps, and 1 Mbps, respectively, on the downstream channel. These rates assume that the base station utilizes all time slots available on all F frequencies to send to just one station. The base station has an infinite amount of data to send to each of the nodes, and can send to any one of these four nodes using any of the F frequencies during any time slot in the downstream sub-frame.

a. What is the maximum rate at which the base station can send to the nodes, assuming it can send to any node it chooses during each time slot? Is your solution fair? Explain and define what you mean by "fair."

b. If there is a fairness requirement that each node must receive an equal amount of data during each one second interval, what is the average transmission rate by the base station (to all nodes) during the downstream sub-frame? Explain how you arrived at your answer.

c. Suppose that the fairness criterion is that any node can receive at most twice as much data as any other node during the sub-frame. What is the average transmission rate by the base station (to all nodes) during the sub- frame? Explain how you arrived at your answer.

答：a. 如果基站仅仅发送到节点 A，这个解决方案不公平因为仅仅 A 获得服务。"公平"的含义是这四个节点的每一个应该被分配相同数量的时隙。

b. 为了达到公平，在每个下游子帧期间，因而每个节点要接收等量的数据，用 n1，n2，n3，n4 分别表示 A，B，C，D 得到的的时隙的数量。然后，

在一个时隙内传输到 A 的数据=10t Mbits(假定每个时隙大小为 t)。因此，传输到 A 的数据总量(在 n1 个时隙内)=10t n1。类似的，传输到 B，C，D 的数据总量分别为 5t n2，2.5t n3 和 1t n4。因此有，n2=2n1，n3=4n1，n4=10n1。现在假定总的时隙数目为 N，则：

n1+ n2+ n3+ n4 = N，即 n1+ 2 n1 + 4 n1 + 10 n1 = N，即 n1 = N/17，由此可得：n2 = 2N/17，n3 = 4N/17，n4 = 10N/17，所以平均传输速率为：

$$(10\mathrm{t}\,\mathrm{n}1 + 5\mathrm{t}\,\mathrm{n}2 + 2.5\,\mathrm{n}3 + \mathrm{t}\,\mathrm{n}4)/\mathrm{tN}$$

$$= \left(\frac{10\mathrm{N}}{17} + 5 \times \frac{2\mathrm{N}}{17} + 2.5 \times \frac{4\mathrm{N}}{17} + 1 \times \frac{10\mathrm{N}}{17}\right)$$

$$= \frac{40}{17} = 2.35\mathrm{Mbps}$$

c. 让 A 在下游子帧期间接收到的数据为 B，C，D 的两倍，因此有：

$10\mathrm{tn}1 = 2 \times 5\mathrm{tn}2 = 2 \times 2.5\mathrm{tn}3 = 2 \times \mathrm{tn}4$，即：$\mathrm{n}2 = \mathrm{n}1$，$\mathrm{n}3 = 2\mathrm{n}1$，$\mathrm{n}4 = 5\mathrm{n}1$

再次，n1+ n2+ n3+ n4 = N，即 n1+ n1 + 2n1 + 5n1 = N，即 n1 = N/9，此刻平均传播速率为：

$$\frac{10\mathrm{t}\,\mathrm{n}1 + 5\mathrm{t}\,\mathrm{n}2 + \ 2.5\mathrm{t}\,\mathrm{n}3 + \mathrm{t}\,\mathrm{n}4}{\mathrm{tN}} = \frac{25}{9} = 2.78\mathrm{Mbps}$$

类似的，考虑节点 B，C 和 D 接收到的数据为其它三个节点的 2 倍，能算出不同的平均传播速率值。

P11. In Section 6.5, one proposed solution that allowed mobile users to maintain their IP addresses as they moved among foreign networks was to have a for- eign network advertise a highly specific route to the mobile user and use the existing routing infrastructure to propagate this information throughout the network. We identified scalability as one concern. Suppose that when a mobile user moves from one network to another, the new foreign network advertises a specific route to the mobile user, and the old foreign network withdraws its route. Consider how routing information propagates in a distance-vector algorithm (particularly for the case of interdomain routing among networks that span the globe).

a. Will other routers be able to route datagrams immediately to the new for- eign network as soon as the foreign network begins advertising its route?

b. Is it possible for different routers to believe that different foreign networks contain the mobile user?

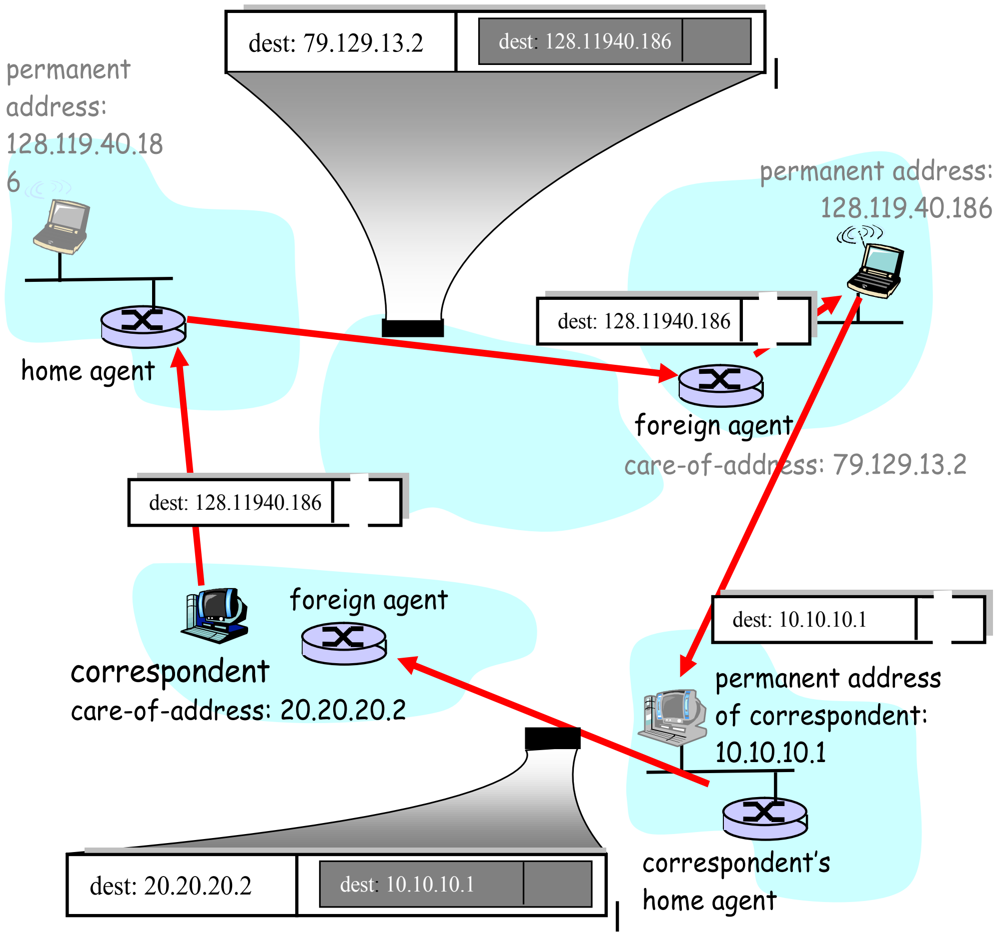
c. Discuss the timescale over which other routers in the network will eventually learn the path to the mobile users.

答：a. 所有的路由器不可能为数据报立刻选路。这是因为距离向量算法(Distance Vector algorithm)(外加自制系统之间的 BGP 协议)是分散的并且花费时间才能完成。因此，在算法运行期间，由于收到来自新的外部网络的通告，一些路由器不可能为源自移动设备的数据报完成选路。

b. 是的。当一个移动设备刚离开外部网络又加入一个一个新的外部网络。这样，当这个新网络的条目在被传播时，旧的外部网络的选路条目不可能已经完全撤回。

c. 路由器知道移动用户的路径的时间取决于：在这个路由器和节点的外部网络的边界路由器两者之间的跳数。

P12. Suppose the correspondent in Figure 6.22 were mobile. Sketch the additional network-layer infrastructure that would be needed to route the datagram from the original mobile user to the (now mobile) correspondent. Show the structure of the datagram(s) between the original mobile



user and the (now mobile) correspondent, as in Figure 6.23.

答：如下图：

如果通信者(correspondent)是移动的，任何到达通信者的数据报将会通过通信者内部代理(correspondent’s home agent)。网络中被访问的外部代理也将涉及，因为外部代理将通知内部代理通信者的位置。通信者内部代理收到的数据报在通信者内部代理和外部代理之间会被重复封装/传入信道，参看图 6.23。

P13. In mobile IP, what effect will mobility have on end-to-end delays of data- grams between the source and destination?

答：因为来自移动设备的数据报必须先转发给 home agent，因而时延一半比直接通过选路要长。然而，从通信者到移动设备(即数据报不通过内部代理 home agent)的直接时延可能实际上比从通信者到内部代理(home agent)再到移动设备的总时延要短，这是可能发生的。这取决于在各种路径片段上的时延。注意，间接选路也需要添加一个内部代理处理(如封装)时延。

P14. Consider the chaining example discussed at the end of Section 6.7.2. Suppose a mobile user visits foreign networks A, B, and C, and that a correspondent begins a connection to the mobile user when it is resident in foreign network A. List the sequence of messages between foreign agents, and between foreign agents and the home agent as the mobile user moves from network A to network B to network C. Next, suppose chaining is not performed, and the correspondent (as well as the home agent) must be explicitly notified of the changes in the mobile user’s care-of address. List the sequence of messages that would need to be exchanged in this second scenario.

答：首先，我们在课本 6.5 的末尾讨论了链(chaining)。在通过内部代理(home agent)使用间接选路的链的情形下，将发生下列事件：

- 移动节点到达 A，A 通知内部代理(home agent)移动设备在访问 A 到移动设备的数据报现在应该转发给 A 中特定的转交地址 care-of-address(COA)。
- 移动节点移动到 B。B 处的外部代理必须通告 A 的外部代理这个移动节点不在 A 了，但实际上在 B 并且在 B 中有特定的 COA。此后，A 的外部代理将转发它收到一些数据报，这些数据报源于 A 中移动设备的 COA，目的地为 B 中移动设备的 COA。
- 移动节点移动到 C。C 处的外部代理必须通告 B 的外部代理这个移动节点不在 B 了，但实际上在 C 并且在 C 中有特定的 COA。此后，B 的外部代理将转发它收到一些数据报(起初它们源于 A 中的外部代理)，这些数据报源于 B 中移动设备的 COA，目的地为 C 中移动设备的 COA。

注意，当移动设备脱机 offline(即没有地址)或者回到它的内部网络，通过 A，B 和 C 中的外部代理维持的“数据报-转发”状态必须移除，这个必须发信号报文来实现。注意，内部代理并不知道 A 以外移动设备的移动变化，并且通信者完全不知道移动设备的移动变化。

在不使用链的情况下，将发生以下事件：

- 移动节点到达 A，A 通知内部代理(home agent)移动设备在访问 A 到移动设备的数据报现在应该转发给 A 中特定的转交地址 care-of-address(COA)。
- 移动节点移动到 B。B 处的外部代理必须通告内部代理和 A 的外部代理这个移动节点不在 A 了，但实际上在 B 并且在 B 中有特定的 COA。A 中的外部代理可以移除这个移动节点的状态，因为它不在 A 中了。此后，内部代理将转发它收到一些数据报，其目的地为 B 中移动设备的 COA。
- 移动节点移动到 C。C 处的外部代理必须通告内部代理和 B 的外部代理这个移动节点不在 B 了，但实际上在 C 并且在 C 中有特定的 COA。B 中的外部代理可以移除这个移动节点的状态，因为它不在 B 中了。此后，内部代理将转发它收到一些数据报，其目的地为 C 中移动设备的 COA。

当移动设备脱机 offline(即没有地址)或者回到它的内部网络，C 的外部代理维持的“数据报-转发”状态必须移除，这个必须发信号报文(signaling messages)来实现。注意，内部代理总是知道移动设备的当前外部网络。然而，通信者仍然不知道移动设备的移动变化。

P15. Consider two mobile nodes in a foreign network having a foreign agent. Is it possible for the two mobile nodes to use the same care-of address in mobile IP? Explain your answer.

答：两个移动设备在访问同一个网络时当然可以用一样的转交地址 care-of-address(COA)。实际上，如果 COA 是外部代理的地址，那么这个地址会是一样的。一旦外部代理将隧道数据报解封装并且决定移动设备的地址，那么在，各自的地址将被用来独自发送数据报到不同的被访问网络中的目的地(移动设备)。

P16. In our discussion of how the VLR updated the HLR with information about the mobile's current location, what are the advantages and disadvantages of providing the MSRN as opposed to the address of the VLR to the HLR?

答：如果将 MSRN(mobile station roaming number 移动站点漫游号码)提供给 HLR(home location register 归属位置注册器)，无论何时只要 MSRN 发生变化，在 HLR 中的 MSRN 的值必须更新(即有一个脱机使得 MSRN 发生变化)。在 HLR 中存在 MSRN 的优势是其数值可以快速被提供，没有去查询 VLR。通过提供 VLR(visitor location register 访问者注册器)的地址(而非 MSRN)，没有必要刷新 HLR 中的 MSRN。

Chapter 7 Multimedia Networking

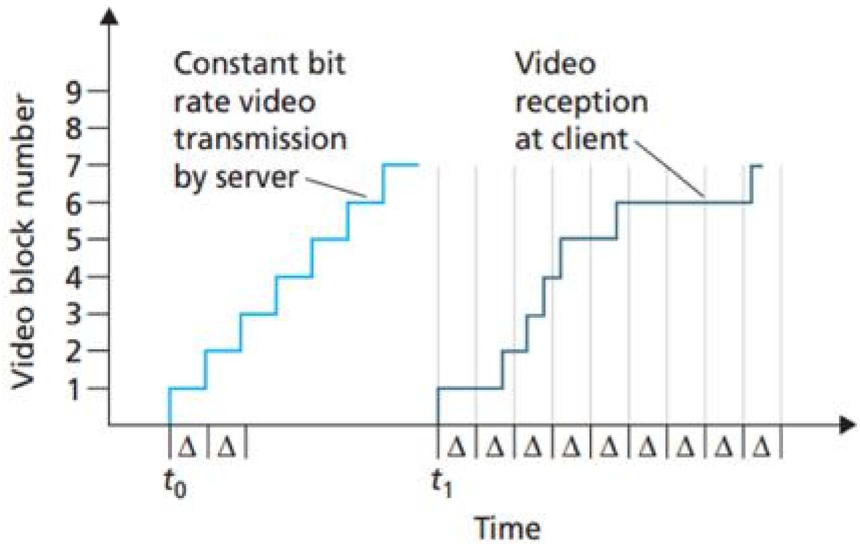
P1. Consider the figure below. Similar to our discussion of Figure 7.1, suppose that video is encoded at a fixed bit rate, and thus each video block contains video frames that are to be played out over the same fixed amount of time, Δ. The server transmits the first video block at t0, the second block at t0 +Δ, the third block at t0+2Δ, and so on. Once the client begins playout, each block should be played out Δ time units after the previous block.

a. Suppose that the client begins playout as soon as the first block arrives at t1. In the figure below, how many blocks of video (including the first block) will have arrived at the client in time for their playout? Explain how you arrived at your answer.

b. Suppose that the client begins playout now at t1 + 6. How many blocks of video (including the first block) will have arrived at the client in time for their playout? Explain how you arrived at your answer.

c. In the same scenario at (b) above, what is the largest number of blocks that is ever stored in the client buffer, awaiting playout? Explain how you arrived at your answer.

d. What is the smallest playout delay at the client, such that every video block has arrived in time for its playout? Explain how you arrived at your answer.



答：a. 只要第一个时钟到达 t_1 客户机就开始播放，并且视频时钟在确定的时间 d 内将会被播放完。因此第二个时钟应该在时间 t_1+d 之前到达以便在正确的时间播放，第三个时钟在 t_1+2d 等。我们从图中可以看到仅仅第 1，4，5，6 个视频时钟在他们时间播放完之前到达了接收方。

b. 客户机开始在 t_1+d 时刻播放，并且视频时钟在确定的时间 d 内将会被播放完。因此第二个时钟应该在时间 t_1+2d 之前到达以便在正确的时间播放，第三个时钟在 t_1+3d 等。我们从图中可以看到从第 1 到第 6 个除了第七个视频时钟在他们时间播放完之前到达了接收方。

c. 两个视频时钟的最大值存储于客户机的缓存中。第 3，4 个视频时钟 t_1+3d 之前和 t_1+2d 之后到达，因此这两个时钟存储在客户机的接收方。第 5 个视频时钟 t_1+4d 之前和 t_1+3d 之后到达，它存储在客户机的缓存中和已经存储的第 4 个视频时钟在一起。

d. 客户机最小的播放时间应该为 t_1+3d 以便确保每个时钟按时到达。

P2. Recall the simple model for HTTP streaming shown in Figure 7.3. Recall that B denotes the size of the client's application buffer, and Q denotes the number of bits that must be buffered before the client application begins playout. Also r denotes the video consumption rate. Assume that the server sends bits at a constant rate x whenever the client buffer is not full.

a. Suppose that x < r. As discussed in the text, in this case playout will alternate between periods of continuous playout and periods of freezing. Determine the length of each continuous playout and freezing period as a function of Q, r, and x.

b. Now suppose that x > r. At what time $t = t_f$ does the client application buffer become full?

答：a. 在一段播放期间(playout period)，缓存以 Q 比特开始且下降速率为 r-x。因此，在 $Q/(r-x)$ 秒后开始重播直到缓冲区为空。一旦缓冲区变空，又以速率 x 填充 Q/x 秒，此时有 Q 比特并且又开始重播。因此，freezing period 时间是 Q/x 秒。

b. 缓冲区到达 Q 比特时间是 Q/x 秒，增加额外的 B-Q 比特时间为$(B-Q)/(x-r)$，因此直到应用程序缓冲区为满时间为$\frac{Q}{x}+\frac{B-Q}{x-r}$秒。

P3. Recall the simple model for HTTP streaming shown in Figure 7.3. Suppose the buffer size is infinite but the server sends bits at variable rate x(t). Specifically, suppose x(t) has the following saw-tooth shape. The rate is initially zero at time t = 0 and linearly climbs to H at time t = T. It then repeats this pattern again and again, as shown in the figure below.

a. What is the server's average send rate?

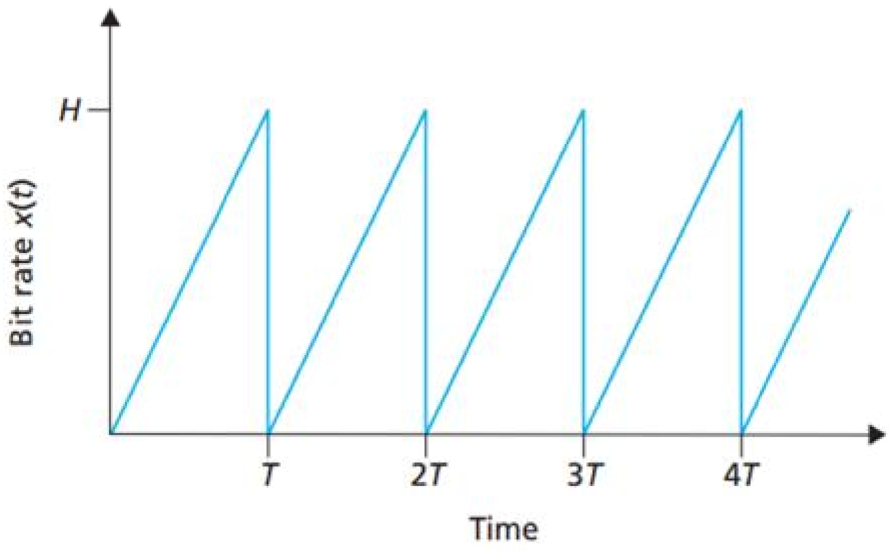
b. Suppose that Q = 0, so that the client starts playback as soon as it receives a video frame. What will happen?

c. Now suppose Q > 0. Determine as a function of Q, H, and T the time at which playback first begins.

d. Suppose H > 2r and Q = HT/2. Prove there will be no freezing after the initial playout delay.

e. Suppose H > 2r. Find the smallest value of Q such that there will be no freezing after the initial playback delay.

f. Now suppose that the buffer size B is finite. Suppose H > 2r. As a function of Q, B, T, and H, determine the time $t = t_f$ when the client application buffer first becomes full.



答：a. 服务器平均发送速率为 H/2。

b. b 是一个 odd question 将在下一版本讨论。在第一帧播放完后，由于 x(t) < r，下一个帧将会在它的计划播放完时间之后到达。因此在展示完第一帧之后重播将会停止(Thus playback will freeze after displaying the first frame.)。

c. 用 q(t)表示在时刻 t 缓冲区的比特数量，当 q(t) = Q 时 playout 开始。我们假定这个问题的吞吐率满足$HT/2 \geq Q$，因此在 x(t)的第一轮循环结束之前有 q(t) = Q。我们有：

$$q(t) = \int_0^t \frac{H}{T} sds = Ht^2/2T$$

因此，当$t = \sqrt{2QT/H} = tp$时，q(t) = Q。

d. 在 t = T 时，q (t) = HT/2 = Q，因此 playout 开始。如果随后没有 freezing，对于所有的$t \geq T$，我们需要满足 q(t + T) > 0，因此有

$$q(t+T) = \frac{HT}{2} - rt + \int_0^t x(s)\,ds$$
$$> \frac{H}{2}(T-t) + \int_0^t x(s)\,ds$$

并且有 t = nT + Δ，0 < Δ < T，我们可以从上面得到

$$q(t+T) > \frac{H}{2}(T - nT - \Delta) + \frac{nHT}{2} + \frac{H\Delta^2}{2T}$$
$$= \frac{H}{2}(T - \Delta + \frac{\Delta^2}{T})$$

显而易见对于所有的 0 < Δ < T 都是成立的。

e. 首先考虑[0, T]，我们有

$$q(t) = \frac{H}{2T}t^2 - r(t - tp) \ \ \text{For } tp \leq t \leq T$$

当 t = rT/H 时，q(t)有最小值。当且仅当$tp \geq rT/2H$时有$q\ (rT/H) \geq 0$，进一步如果 $tp = rT/2H$ 。可以扩展证明对于所有的 t ≥ T，有 q (t) > 0。因此，$tp < rT/2H$ 并且

$$Q = r2T/8H$$

f. 这是一个非常有挑战性的问题。假定在时间 T 之前 B 缓冲满，那么有：

$$\frac{H}{2T}t^2 - r\left(t - \sqrt{\frac{2QT}{H}}\right) = B$$

P4. Recall the simple model for HTTP streaming shown in Figure 7.3. Suppose the client application buffer is infinite, the server sends at the constant rate x, and the video consumption rate is r with r < x. Also suppose playback begins immediately. Suppose that the user terminates the video early at time t = E. At the time of termination, the server stops sending bits (if it hasn't already sent all the bits in the video).

a. Suppose the video is infinitely long. How many bits are wasted (that is, sent but not viewed)?

b. Suppose the video is T seconds long with T > E. How many bits are wasted (that is, sent but not viewed)?

答：a. 缓冲区增长速率为 x-r，在 E 时刻，缓冲区有$(x - r) \times E$比特并且被浪费。

b. 用 S 表示服务器传输完成整个视频的时间。如果 S>E，直到时刻 E，缓冲区增长速率为 x-r，因此又一次浪费$(x - r) \times E$；如果 S<E，那么在 E 时刻，仍然有 T-E 秒的视频在缓冲区播放，因此，浪费的比特为：$r \times (T - E)$。

P5. Consider a DASH system for which there are N video versions (at N different rates and qualities) and N audio versions (at N different rates and versions). Suppose we want to allow the player to choose at any time any of the N video versions and any of the N audio versions.

a. If we create files so that the audio is mixed in with the video, so server sends only one media stream at given time, how many files will the server need to store (each a different URL)?

b. If the server instead sends the audio and video streams separately and has the client synchronize the streams, how many files will the server need to store?

答：a. 服务器需要存储的文件数量为：$N \times N = N^2$。

b. 服务器需要存储的文件数量为：$N + N = 2N$。

P6. In the VoIP example in Section 7.3, let h be the total number of header bytes added to each chunk, including UDP and IP header.

a. Assuming an IP datagram is emitted every 20 msecs, find the transmission rate in bits per second for the datagrams generated by one side of this application.

b. What is a typical value of h when RTP is used?

答：a. 每 20 秒发送 160+h 字节，因此传输速率为：

$$\frac{(160 + h) \times 8}{20}Kbps = (64 + 0.4)Kbps$$

b. IP 首部：20 字节；UDP 首部：8 字节；RTP 首部：12 字节。h=40 字节

P7. Consider the procedure described in Section 7.3 for estimating average delay di. Suppose that u = 0.1. Let r1 – t1 be the most recent sample delay, let r2 – t2 be the next most recent sample delay, and so on.

a. For a given audio application suppose four packets have arrived at the receiver with sample delays r4 – t4, r3 – t3, r2 – t2, and r1 – t1. Express the estimate of delay d in terms of the four samples.

b. Generalize your formula for n sample delays.

c. For the formula in Part b, let n approach infinity and give the resulting formula. Comment on why this averaging procedure is called an exponential moving average.

答：a. 用$\mathrm{d}^{(n)}$表示在 n 个样本后的估计，则

$d^{(1)} = r_4 - t_4$

$d^{(2)} = u(r_3 - t_3) + (1-u)(r_4 - t_4)$

$d^{(3)} = u(r_2 - t_2) + (1-u)[u(r_3 - t_3) + (1-u)(r_4 - t_4)]$

$\quad = u(r_2 - t_2) + (1-u)u(r_3 - t_3) + (1-u^2)(r_4 - t_4)$

$d^{(4)} = u(r_1 - t_1) + (1-u)u(r_2 - t_2) + (1-u^2)(r_3 - t_3) + (1-u^3)(r_4 - t_4)$

b. n 个样本的时延为：

$$d^{(\mathrm{n})} = u\sum_{j=1}^{n-1}(1-u)^j\left(r_j - t_j\right) + (1-u)^n(r_n - t_n)$$

c. 如下：

$$d^{(\infty)} = \frac{u}{1-u}\sum_{j=1}^{\infty}(1-u)^j\left(r_j - t_j\right) = \frac{1}{9}\sum_{j=1}^{\infty}0.9\left(r_j - t_j\right)$$

给定的权值相对于上一个样本以指数方式衰减(The weight given to past samples decays exponentially.)。

P8. Repeat Parts a and b in Question P7 for the estimate of average delay deviation.

a. 用$v^{(n)}$表示在 n 个样本的平均时延偏差的估计，用$\Delta_j = \left(r_j - t_j\right)$，则：

$v^{(1)} = \left|\Delta_4 - d^{(1)}\right| (= 0)$

$v^{(2)} = u\left|\Delta_3 - d^{(2)}\right| + (1-u)\left|\Delta_4 - d^{(1)}\right|$

$v^{(3)} = u\left|\Delta_2 - d^{(3)}\right| + (1-u)v^{(2)}$

$\quad = u\left|\Delta_2 - d^{(3)}\right| + u(1-u)\left|\Delta_3 - d^{(2)}\right| + (1-u)^2\left|\Delta_4 - d^{(1)}\right|$

$v^{(4)} = u\left|\Delta_1 - d^{(4)}\right| + (1-u)v^{(3)}$

$\quad = u\left|\Delta_1 - d^{(4)}\right| + u(1-u)\left|\Delta_2 - d^{(3)}\right| + u(1-u)^2\left|\Delta_3 - d^{(2)}\right| + (1-u)^3\left|\Delta_4 - d^{(1)}\right|$

$\quad = u\left[\left|\Delta_1 - d^{(4)}\right| + (1-u)\left|\Delta_2 - d^{(3)}\right| + (1-u^2)\left|\Delta_3 - d^{(2)}\right|\right] + (1-u)^2\left|\Delta_4 - d^{(1)}\right|$

b. 当有 n 个样本时，

$$v^{(n)} = u\sum_{j=1}^{n-1}(1-u)^{j-1}\left|\Delta_j - d^{(n-j+1)}\right| + (1-u)^n\left|\Delta_n - d^{(1)}\right|$$

P9. For the VoIP example in Section 7.3, we introduced an online procedure (exponential moving average) for estimating delay. In this problem we will examine an alternative procedure. Let t_i be the timestamp of the ith packet received; let r_i be the time at which the ith packet is received. Let d_n be our estimate of average delay after receiving the nth packet. After the first packet is received, we set the delay estimate equal to $d_1 = r_1 - t_1$.

a. Suppose that we would like $d_n = (r_1 - t_1 + r_2 - t_2 + \ldots + r_n - t_n)/n$ for all n. Give a recursive formula for d_n in terms of d_{n-1}, r_n, and t_n.

b. Describe why for Internet telephony, the delay estimate described in Section 7.3 is more appropriate than the delay estimate outlined in Part a.

答：a. 由于 $r_1 - t_1 + r_2 - t_2 + \ldots + r_{n-1} - t_{n-1} = (n-1)d_{n-1}$，代入 d_n 表达式有：

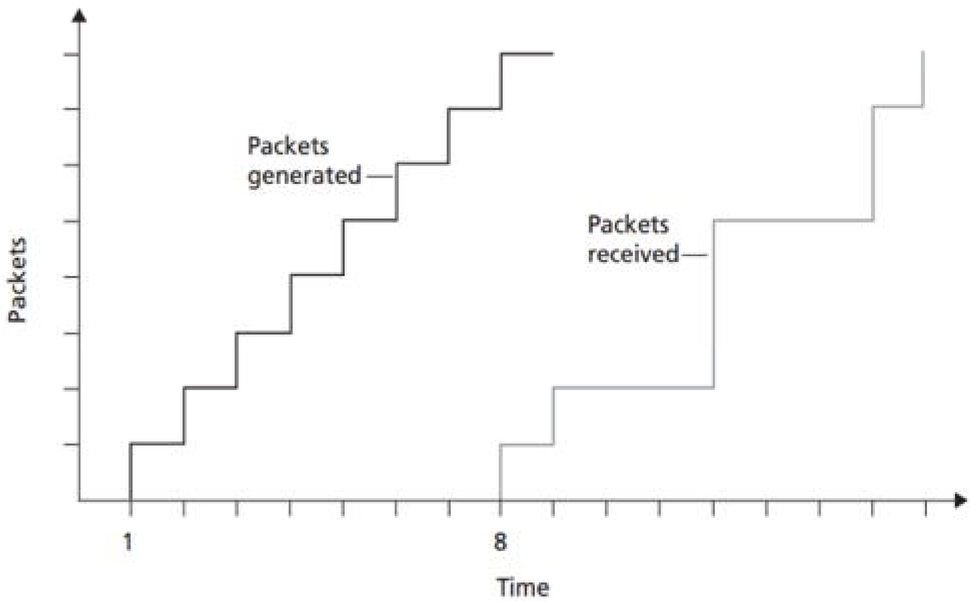
$$d_n = \frac{n-1}{n} d_{n-1} + \frac{r_n - t_n}{n}$$

b. a 中的时延预测(delay estimate)是一个平均值，对于最近的时延和老的时延，它给出了相等的权值。在 7.3 节的时延预测中给了最近的时延更大的权值，很久以前的时延对时延估测影响相对较小。

P10. Compare the procedure described in Section 7.3 for estimating average delay with the procedure in Section 3.5 for estimating round-trip time. What do the procedures have in common? How are they different?

答：这两个过程非常相似，他们使用相同的公式，从而以指数方式降低过去采样的权重。一个不同之处是对于平均 RTT 的估计，数据被发送的时间和确认被收到的时间都记录在同一个机器上的。关于时延预测的两个值记录在两个不同机器上。因此样本时延实际上可能是负的(negative)。

P11. Consider the figure below (which is similar to Figure 7.7). A sender begins sending packetized audio periodically at t = 1. The first packet arrives at the receiver at t = 8.



a. What are the delays (from sender to receiver, ignoring any playout delays) of packets 2 through 8? Note that each vertical and horizontal line segment in the figure has a length of 1, 2, or 3 time units.

b. If audio playout begins as soon as the first packet arrives at the receiver at t = 8, which of the first eight packets sent will not arrive in time for playout?

c. If audio playout begins at t = 9, which of the first eight packets sent will not arrive in time for playout?

d. What is the minimum playout delay at the receiver that results in all of the first eight packets arriving in time for their playout?

答：a. 第 2 到第 7 个分组的时延分别为 7，9，8，7，9，8 个时隙(slot)，第 8 个分组的时延大于 8 个时隙。

b. 如果在 t=8 时刻开始播放，不能按时到达进行播放的分组为第 3，4，6，7，8 个分组。

c. 如果在 t=9 时刻开始播放，不能按时到达进行播放的分组为第 3，6 个分组。

d. 使得所有前 8 个分组按时到达进行播放的最小播放时延为 t=10。

P12. Consider again the figure in P11, showing packet audio transmission and reception times.

a. Compute the estimated delay for packets 2 through 8, using the formula for di from Section 7.3.2. Use a value of u = 0.1.

b. Compute the estimated deviation of the delay from the estimated average for packets 2 through 8, using the formula for vi from Section 7.3.2. Use a value of u = 0.1.

答：a 和 b 的答案如下表：

分组序号	$r_i - t_i$	d_i	v_i
1	7	7	0
2	8	7.10	0.09
3	8	7.19	0.162
4	7	7.17	0.163
5	9	7.35	0.311
6	9	7.52	0.428
7	8	7.57	0.429
8	8	7.61	0.425

P13. Recall the two FEC schemes for VoIP described in Section 7.3. Suppose the first scheme generates a redundant chunk for every four original chunks. Suppose the second scheme uses a low-bit rate encoding whose transmission rate is 25 percent of the transmission rate of the nominal stream.

a. How much additional bandwidth does each scheme require? How much playback delay does each scheme add?

b. How do the two schemes perform if the first packet is lost in every group of five packets? Which scheme will have better audio quality?

c. How do the two schemes perform if the first packet is lost in every group of two packets? Which scheme will have better audio quality?

答：a. 两个方案都需要 25%的额外带宽，第一个方案增加到 5 个分组的被播放时延，第二个方案增加到 2 个分组的播放时延。

b. 第一种方案能够重构原始高质量的音频编码，第二个方案将使用丢失分组对应的低音频质量的数据包来代替，毫无疑问第一种方案会有更好的音频质量。

c. 第一种方案将无法重构丢失的分组，第二个方案让然用丢失的分组对应的低音频质量的数据包来代替，第二种方案会有更好的音频质量。

P14. a. Consider an audio conference call in Skype with N > 2 participants. Suppose each participant generates a constant stream of rate r bps. How many bits per second will the call initiator need to send? How many bits per second will each of the other N – 1 participants need to send? What is the total send rate, aggregated over all participants?

b. Repeat part (a) for a Skype video conference call using a central server.

c. Repeat part (b), but now for when each peer sends a copy of its video stream to each of the N – 1 other peers.

答：a. 在其他 N-1 个参与者中，每一个都以速率 r bps 发送单个音频流到发起者(initiator)。发起者再将这个收到的流和它自己的输出流组合以速率 r 创造一个流。然后它将组合流的一个 copy 发送到其它 N-1 个参与者。会话发起者的发送总速率为(N-1)r bps，再加上参与者总聚合的速率为 2(N-1)r bps。

b. 和前面一样，在其他 N-1 个参与者中，每一个都以速率 r bps 发送单个视频流到发起者(initiator)。但是由于现在是视频流，发起者不能再将它们组合成单个流。发起者反而必须将每个它收到的流发送到 N-2 个参与者。会话发起者的发送总速率为$(N-1) \times (N-1)r$ bps，再加上参与者总聚合的速率为$(N-1)r + (N-1) \times (N-1)r = N(N-1)r$ bps。

c. 所求为$N(N-1)r$ bps。

P15. a. Suppose we send into the Internet two IP datagrams, each carrying a different UDP segment. The first datagram has source IP address A1, destination IP address B, source port P1, and destination port T. The second datagram has source IP address A2, destination IP address B, source port P2, and destination port T. Suppose that A1 is different from A2 and that P1 is different from P2. Assuming that both datagrams reach their final destination, will the two UDP datagrams be received by the same socket? Why or why not?

b. Suppose Alice, Bob, and Claire want to have an audio conference call using SIP and RTP. For Alice to send and receive RTP packets to and from Bob and Claire, is only one UDP socket sufficient (in addition to the socket needed for the SIP messages)? If yes, then how does Alice's SIP client distinguish between the RTP packets received from Bob and Claire?

答：a. 我 3 们在第二章讨论过，UDP 套接字由目的 IP 地址和目的端口号组成的二元组(two-tuple)来确定，因此这两个分组实际上会通过同一个套接字。

b. 是的，Alice 只需要一个 UDP 套结字足够。Bob 和 Claire 会选择不同的 SSRC，所以 Alice 能够区分这两个流。还有一个问题就是 Alice 的 SIP 客户机怎么知道哪个流(即 SSRC)属于 Bob，哪个流属于 Claire 呢？事实上，在发送发讲话时，Alice 的 SIP 客户机会显示发送方的姓名。Alice 的 SIP 客户机会根据 RTCP 接受报告把 SSRC 映射到姓名上，由此 Alice 的 SIP 客户机就以区分 RTP 分组是来自 Bob 还是来自 Claire。

P16. True or false:

a. If stored video is streamed directly from a Web server to a media player, then the application is using TCP as the underlying transport protocol.

b. When using RTP, it is possible for a sender to change encoding in the mid- dle of a session.

c. All applications that use RTP must use port 87.

d. If an RTP session has a separate audio and video stream for each sender, then the audio and video streams use the same SSRC.

e. In differentiated services, while per-hop behavior defines differences in performance among classes, it does not mandate any particular mechanism for achieving these performances.

f. Suppose Alice wants to establish an SIP session with Bob. In her INVITE message she includes the line: m=audio 48753 RTP/AVP 3 (AVP 3 denotes GSM audio). Alice has therefore indicated in this message that she wishes to send GSM audio.

g. Referring to the preceding statement, Alice has indicated in her INVITE message that she will send audio to port 48753.

h. SIP messages are typically sent between SIP entities using a default SIP port number.

i. In order to maintain registration, SIP clients must periodically send REGISTER messages.

j. SIP mandates that all SIP clients support G.711 audio encoding.

答：a. 正确。

b. 正确。

c. 不，RTP 流可以发送/接收以任何端口号，详情请参见课本 7.4.3 节。

d. 不，它们会被分配不同的 SSRC 值。

e. 正确。

f. 错误，Alice 指示的是她希望接收到的 GSM 音频。

g. 错误，Alice 指示的是她希望接收到的来自 48753 端口的音频。

h. 正确，5060 是 SIP 的默认端口号，它既是源端口号也是目的端口号。

i. 正确。

j. 错误，这是 H.323 的规范而非 SIP 的。

P17. Suppose that the WFQ scheduling policy is applied to a buffer that supports three classes, and suppose the weights are 0.5, 0.25, and 0.25 for the three classes.

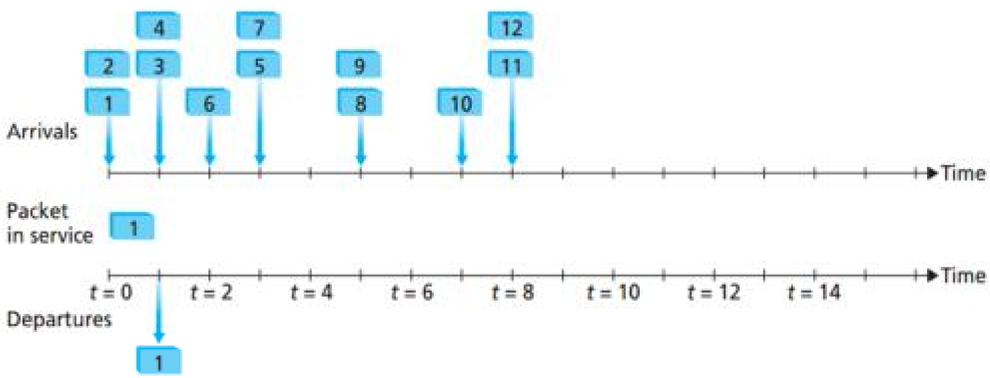
a. Suppose that each class has a large number of packets in the buffer. In what sequence might the three classes be served in order to achieve the WFQ weights? (For round robin scheduling, a natural sequence is 123123123 . . .).

b. Suppose that classes 1 and 2 have a large number of packets in the buffer, and there are no class 3 packets in the buffer. In what sequence might the three classes be served in to achieve the WFQ weights?

答：a. 一个可能的顺序为：1 2 1 3 1 2 1 3 1 2 1 3 …；另一个可能的顺序为：1 1 2 1 1 3 1 1 2 1 1 3 1 1 2 1 1 3 …。

b. 可能的顺序为：1 1 3 1 1 3 1 1 3 1 1 3 …。

P18. Consider the figure below. Answer the following questions:



a. Assuming FIFO service, indicate the time at which packets 2 through 12 each leave the queue. For each packet, what is the delay between its arrival and the beginning of the slot in which it is transmitted? What is the average of this delay over all 12 packets?

b. Now assume a priority service, and assume that odd-numbered packets are high priority, and even-numbered packets are low priority. Indicate the time at which packets 2 through 12 each leave the queue. For each packet, what is the delay between its arrival and the beginning of the slot in which it is transmitted? What is the average of this delay over all 12 packets?

c. Now assume round robin service. Assume that packets 1, 2, 3, 6, 11, and 12 are from class 1, and packets 4, 5, 7, 8, 9, and 10 are from class 2. Indi- cate the time at which packets 2 through 12 each leave the queue. For each packet, what is the delay between its arrival and its departure? What is the average delay over all 12 packets?

d. Now assume weighted fair queueing (WFQ) service. Assume that odd- numbered packets are from class 1, and even-numbered packets are from class 2. Class 1 has a WFQ weight of 2, while class 2 has a WFQ weight of 1. Note that it may not be possible to achieve an idealized WFQ schedule as described in the text, so indicate why you have chosen the particular packet to go into service at each time slot. For each packet what is the delay between its arrival and its departure? What is the average delay over all 12 packets?

e. What do you notice about the average delay in all four cases (FIFO, RR, priority, and WFQ)?

答：a. 如下表：

Packet	**Time leaving the queue**	**Delay**
1	0	0
2	1	1
3	2	1
4	3	2
5	5	2
6	4	2
7	6	3
8	7	2
9	8	3
10	9	2
11	10	2
12	11	3
Average Delay		**1.91**

b. 如下表：

Packet	**Time leaving the queue**	**Delay**
1	0	0
2	2	2
3	1	0
4	6	5
5	4	1
6	7	5
7	3	0
8	9	4
9	5	0
10	10	3
11	8	0
12	11	3
Average Delay		**1.91**

c. 如下表：

Packet	Time leaving the queue	Delay
1	0	0
2	2	2
3	4	3
4	1	0
5	3	0
6	6	4
7	5	2
8	7	2
9	9	4
10	11	4
11	8	0
12	10	2
Average Delay		**1.91**

d. 如下表：

Packet	Time leaving the queue	Delay	Note
1	0	0	WFQ
2	2	2	WFQ
3	1	0	WFQ
4	5	4	WFQ
5	3	0	WFQ
6	7	5	Idealized WFQ scheduling
7	4	1	WFQ
8	9	4	WFQ
9	6	1	Idealized WFQ scheduling
10	10	3	WFQ
11	8	0	WFQ
12	11	3	WFQ
Average Delay		**1.91**	

e. 由上可以得出四个例子平均时延是一样的为 1.91 秒

P19. Consider again the figure for P18.

a. Assume a priority service, with packets 1, 4, 5, 6, and 11 being high- priority packets. The remaining packets are low priority. Indicate the slots in which packets 2 through 12 each leave the queue.

b. Now suppose that round robin service is used, with packets 1, 4, 5, 6, and 11 belonging to one class of traffic, and the remaining packets belonging to the second class of traffic. Indicate the slots in which packets 2 through 12 each leave the queue.

c. Now suppose that WFQ service is used, with packets 1, 4, 5, 6, and 11 belonging to one class of traffic, and the remaining packets belonging to the second class of traffic. Class 1 has a WFQ weight of 1, while class 2 has a WFQ weight of 2 (note that these weights are different than in the previous question). Indicate the slots in which packets 2 through 12 each leave the queue. See also the caveat in the question above regarding WFQ service.

答：a. 如下表：

Packet	**Time leaving the queue**	**Delay**
1	0	0
2	4	4
3	5	4
4	1	0
5	3	0
6	2	0
7	6	3
8	9	4
9	7	2
10	10	3
11	8	0
12	11	3
Average Delay		**1.91**

b. 如下表：

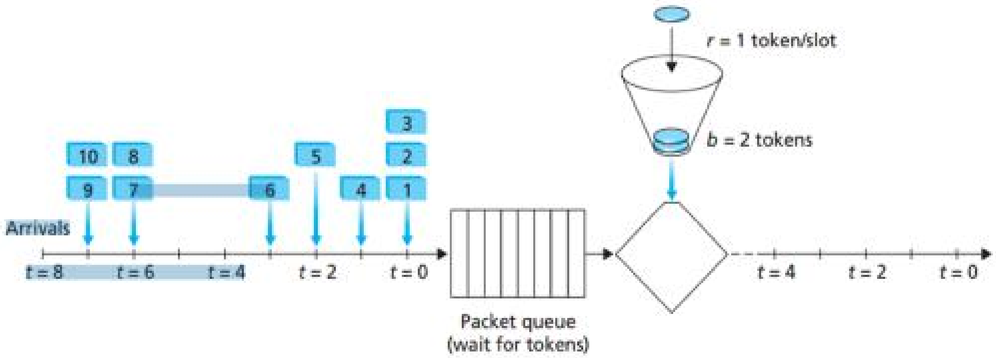
Packet	**Time leaving the queue**	**Delay**
1	0	0
2	1	1
3	3	2
4	2	1
5	6	3
6	4	2
7	5	2
8	9	4
9	7	2
10	10	3
11	8	0
12	11	3
Average Delay		**1.91**

c. 如下表：

Packet	**Time leaving the queue**	**Delay**
1	0	0
2	1	1
3	2	1
4	3	2

5	9	6
6	6	4
7	4	1
8	7	2
9	5	0
10	8	1
11	11	3
12	10	2
Average Delay		**1.91**

P20. Consider the figure below, which shows a leaky bucket policer being fed by a stream of packets. The token buffer can hold at most two tokens, and is initially full at t = 0. New tokens arrive at a rate of one token per slot. The out- put link speed is such that if two packets obtain tokens at the beginning of a time slot, they can both go to the output link in the same slot. The timing details of the system are as follows:



1. Packets (if any) arrive at the beginning of the slot. Thus in the figure, packets 1, 2, and 3 arrive in slot 0. If there are already packets in the queue, then the arriving packets join the end of the queue. Packets proceed towards the front of the queue in a FIFO manner.

2. After the arrivals have been added to the queue, if there are any queued packets, one or two of those packets (depending on the number of avail- able tokens) will each remove a token from the token buffer and go to the output link during that slot. Thus, packets 1 and 2 each remove a token from the buffer (since there are initially two tokens) and go to the output link during slot 0.

3. A new token is added to the token buffer if it is not full, since the token generation rate is r = 1 token/slot.

4. Time then advances to the next time slot, and these steps repeat. Answer the following questions:

a. For each time slot, identify the packets that are in the queue and the num- ber of tokens in the bucket, immediately after the arrivals have been processed (step 1 above) but before any of the packets have passed through the queue and removed a token. Thus, for the t = 0 time slot in the example above, packets 1, 2 and 3 are in the queue, and there are two tokens in the buffer.

b. For each time slot indicate which packets appear on the output after the token(s) have been removed from the queue. Thus, for the t = 0 time slot in the example above, packets 1 and 2 appear on the output link from the leaky buffer during slot 0.

答：a 和 b 如下表格：

Time Slot	Packets in the queue	Number of tokens in bucket
0	1, 2, 3	2
1	3, 4	1
2	4,5	1
3	5,6	1
4	6	1
5	-	1
6	7, 8	2
7	9, 10	1
8	10	1

Time Slot	Packets in output buffer
0	1, 2
1	3
2	4
3	5
4	6
5	-
6	7, 8
7	9
8	10

P21. Repeat P20 but assume that r = 2. Assume again that the bucket is initially full.

答：如下表：

Time Slot	Packets in the queue	Number of tokens in bucket
0	1, 2, 3	2
1	3, 4	2
2	5	2
3	6	2
4	-	2
5	-	2
6	7, 8	2
7	9, 10	2
8	-	2

Time Slot	Packets in output buffer
0	1, 2
1	3, 4
2	5
3	6
4	-
5	-
6	7, 8

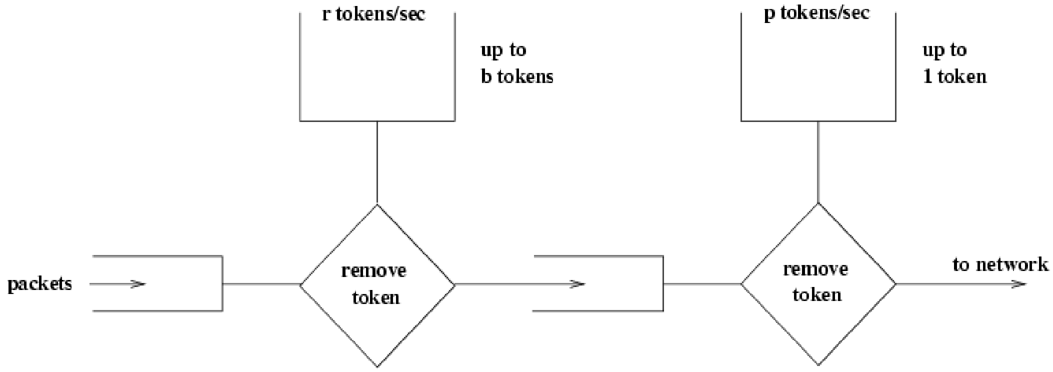
7	9, 10
8	-

P22. Consider P21 and suppose now that r = 3, and that b = 2 as before. Will your answer to the question above change?

答：没有变化，回答和 21 题是一样的。

P23. Consider the leaky-bucket policer that polices the average rate and burst size of a packet flow. We now want to police the peak rate, p, as well. Show how the output of this leaky-bucket policer can be fed into a second leaky bucket policer so that the two leaky buckets in series police the average rate, peak rate, and burst size. Be sure to give the bucket size and token generation rate for the second policer.

答：如下图，对于第二个漏桶，r=p，b=1。



具体解答过程参加 26 题。

P24. A packet flow is said to conform to a leaky-bucket specification (r,b) with burst size b and average rate r if the number of packets that arrive to the leaky bucket is less than rt + b packets in every interval of time of length t for all t. Will a packet flow that conforms to a leaky-bucket specification (r,b) ever have to wait at a leaky bucket policer with parameters r and b? Justify your answer.

答：不是，当 r>=b 时，一个符合规范（r,b）的分组流可以在参数为 r 在[r,b]之间，参数为 r 和 b 的漏桶监管器不必那里等待。

P25. Show that as long as $r_1 < Rw_1/(\sum w_j)$, then d_{max} is indeed the maximum delay that any packet in flow 1 will ever experience in the WFQ queue.

答：令τ为流 1 开始进入队列的时间，我们称之为流-1(flow-1 traffic)的繁忙时间。令$t > \tau$是流 1 同一个繁忙时间的另外一个时间。令$T_1(\tau, t)$是流 1 在[τ, t]时间内传输的分组数，显然有：

$$T_1(\tau, t) \geq \frac{w_1}{\sum w_j} R(t-\tau)$$

用$Q_1(t)$表示流-1 在队列中时刻 t 时的分组大小，显然有：

$$Q_1(t) = b_1 + r_1(t-\tau) - T_1(\tau, t)$$

$$\leq b_1 + r_1(t-\tau) + \frac{w_1}{\sum w_j} R(t-\tau)$$

$$= b_1 + (t-\tau)[r_1 - \frac{w_1}{\sum w_j} R]$$

由$r_1 < \frac{W_1}{\sum W_j}R$，得$Q_1(t) \leq b_1$，因此流-1 在在队列中最大值为$b_1$，这个流量最低传输速率为：

$\frac{W_1}{\sum W_j}R$，因此流 1 中任何分组最大时延为：

$$\frac{b_1}{\frac{W_1}{\sum W_j}R} = d_{max}$$

Chapter 8 Security in Computer Networks

P1. Using the monoalphabetic cipher in Figure 8.3, encode the message “This is an easy problem.” Decode the message “rmij’u uamu xyj.”

答：报文“This is an easy problem.”加密后为：“uasi si my cmiw lokngch”。报文“rmij’u uamu xyj.”解密后为“wasn't that fun”。

P2. Show that Trudy’s known-plaintext attack, in which she knows the (ciphertext, plaintext) translation pairs for seven letters, reduces the number of possible substitutions to be checked in the example in Section 8.2.1 by approximately 10^9.

答：假设 trudy 知道了“bob” 和“alice”会出现在报文中，然后她会知道 b，o，a，l，i，c，e 这七个字母的替换规则(因为“bob”是报文中唯一的回文，并且“alice”仅仅是 5 个字母的单词)。如果 Trudy 知道 7 个单词的密文，那么只剩下 19 个字母的替换规则未知，所以她只需要尝试 $19!$次，而非$26!$。在$19!$和$26!$和区别是：$26*25*24*\cdots 20=3315312000$，大约是 10^9。

P3. Consider the polyalphabetic system shown in Figure 8.4. Will a chosen- plaintext attack that is able to get the plaintext encoding of the message “The quick brown fox jumps over the lazy dog.” be sufficient to decode all messages? Why or why not?

答：“The quick fox jumps over the lazy brown dog.”这句话包括了所有英文字母 26 个，这个报文用选择明文攻击破解(攻击者既破解报文，又加密报文)，凯撒密码(Caesar cipher)会崩溃-入侵者会知道每个明文字母对应的密文字母。然而，Vigenere cipher 不会一直将一个给定的明文字母每次都翻译成同样的密文字母，因此 Vigenere cipher 不会立刻被这个选定的明文攻击破坏。

P4. Consider the block cipher in Figure 8.5. Suppose that each block cipher Ti simply reverses the order of the eight input bits (so that, for example, 11110000 becomes 00001111). Further suppose that the 64-bit scrambler does not modify any bits (so that the output value of the mth bit is equal to the input value of the mth bit).

(a) With n = 3 and the original 64-bit input equal to 10100000 repeated eight times, what is the value of the output?

(b) Repeat part (a) but now change the last bit of the original 64-bit input from a 0 to a 1.

(c) Repeat parts (a) and (b) but now suppose that the 64-bit scrambler inverses the order of the 64 bits.

答：a. 输出值为：00000101 重复 8 次。

b. 输出值为：00000101 重复 7 次加上 10000101。

c. 我们有(ARBRCR)R = CBA，A，B，C 是字符串，R 表示倒转操作(inverse operation)。因此：

1. 对于 a，输出值为：10100000 重复 8 次；

2. 对于 b，输出值为：10100001 + 10100000 重复 7 次。

P5. Consider the block cipher in Figure 8.5. For a given “key” Alice and Bob would need to keep eight tables, each 8 bits by 8 bits. For Alice (or Bob) to store all eight tables, how many bits of storage are necessary? How does this number compare with the number of bits required for a full-table 64-bit block cipher?

答：a. 有 8 个表格，每个表格有 28 个条目，每个条目 8bits。

表格数量 × 每个表格大小 × 每个条目大小 $= 8 \times 28 \times 8 = 214\text{bits}$

b. 有 264 个条目，每个条目 64 比特，所求为 271 比特。

P6. Consider the 3-bit block cipher in Table 8.1. Suppose the plaintext is 100100100.

(a) Initially assume that CBC is not used. What is the resulting ciphertext?

(b) Suppose Trudy sniffs the ciphertext. Assuming she knows that a 3-bit block cipher without CBC is being employed (but doesn't know the specific cipher), what can she surmise?

(c) Now suppose that CBC is used with IV = 111. What is the resulting ciphertext?

答：a. 100100100 ==> 011011011；

b. Trudy 将发现三个明文块是一样的；

c. c(i) = KS(m(i) XOR c(i-1))

c(1) = KS(100 XOR 111) = KS (011) = 100

c(2) = KS(100 XOR 100) = KS (000) = 110

c(1) = KS(100 XOR 110) = KS (010) = 101。

P7. (a) Using RSA, choose p = 3 and q = 11, and encode the word "dog" by encrypting each letter separately. Apply the decryption algorithm to the encrypted version to recover the original plaintext message.

(b) Repeat part (a) but now encrypt "dog" as one message m.

答：a. 首先，由 p = 3 且 q = 11，我们由此可得 n=33 且 q=11。选择 e=9，由 3 和$(p-1)\times(q-1)=20$互为素数。选择 d=9 则$e\times d=81$，因此$e\times d-1=80$可以被 20 整除。所以，我们可以用 n=3，3e=9，d=9 来执行 RSA 操作。

b. 如下：

letter	m	m**e	ciphertext = m**e mod 33
d	4	262144	25
o	15	38443359375	3
g	7	40353607	19

ciphertext	c**d	m = c**d mod n	letter
25	38146972265625	4	d
3	19683	15	o
19	322687697779	7	g

首先我们将每个字母考虑成 5bit 数据：00100, 01111, 00111，然后我们将每个字母连接起来得到：001000111100111 并且进行加密得到 10 进制数 m=4583。连接得到的数字 m=4583 比当前的 n=33 要大。我们需要 m<n，因此我们用 p = 43, q = 107, n = p*q = 4601, z = (p-1)(q-1) = 4452. e = 61, d = 73 则：

密文= m**e mod 4601，由

m**e=21386577601828057804089602156530567188611499869029788733808438804302864595620613956725840720949764845640956118784875246785033236197777129730258961756918400292048632806197527785447791567255101894492820972508185769802881718983

密文= m**e mod 4601=402，由

c**d= 1283813313619771634195712132539793287643533147482536209328405262793027158861012392053287249633570967493122280221453815012934 2

41370540204581459871497938723214101470322779458649981794563339059
2

密文= m**e mod 4601=4583。

P8. Consider RSA with p = 5 and q = 11.

a. What are n and z?

b. Let e be 3. Why is this an acceptable choice for e?

c. Find d such that de = 1 (mod z) and d < 160.

d. Encrypt the message m = 8 using the key (n, e). Let c denote the corre- sponding ciphertext. Show all work. Hint: To simplify the calculations, use the fact: [(a mod n)*(b mod n)] mod n = (a*b) mod n.

答：由 p = 5, q = 11

a. n = p*q = 55, z = (p-1)(q-1) = 40；

b. e=3 满足条件比 n 小并且和 z 互为素数；

c. d=27。

e. m=8，me=512。密文 c= me mod n = 17。

P9. In this problem, we explore the Diffie-Hellman (DH) public-key encryption algorithm, which allows two entities to agree on a shared key. The DH algorithm makes use of a large prime number p and another large number g less than p. Both p and g are made public (so that an attacker would know them). In DH, Alice and Bob each independently choose secret keys, SA and SB, respectively. Alice then computes her public key, TA, by raising g to SA and then taking mod p. Bob similarly computes his own public key TB by raising g to SB and then taking mod p. Alice and Bob then exchange their public keys over the Internet. Alice then calculates the shared secret key S by raising TB to SA and then taking mod p. Similarly, Bob calculates the shared key S′ by rais- ing TA to SB and then taking mod p.

a. Prove that, in general, Alice and Bob obtain the same symmetric key, that is, prove S = S′.

b. With p = 11 and g = 2, suppose Alice and Bob choose private keys SA = 5 and SB = 12, respectively. Calculate Alice’s and Bob’s public keys, TA and TB . Show all work.

c. Following up on part (b), now calculate S as the shared symmetric key. Show all work.

d. Provide a timing diagram that shows how Diffie-Hellman can be attacked by a man-in-the-middle. The timing diagram should have three vertical lines, one for Alice, one for Bob, and one for the attacker Trudy.

答：如下：

	Alice	Bob
secrect key:	S_A	S_B
public key:	$T_A = (g\hat{}S_A) \bmod p$	$T_B = (g\hat{}S_B) \bmod p$
shared key:	$S = (T_B\hat{}S_A) \bmod p$	$S' = (T_A\hat{}S_B) \bmod p$

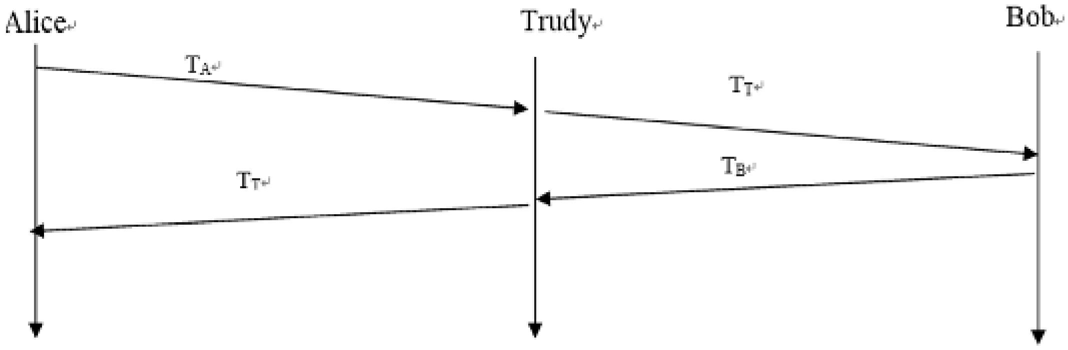
a. S = (TB^SA) mod p = ((g^SB mod p)^SA) mod p = (g^(SBSA)) mod p= ((g^SA mod p)^SB) mod p = (TA^SB) mod p = S'

b 和 c：p = 11, g = 2

	Alice	Bob
secrect key:	SA= 5	SB = 12

public key: $TA = (g^SA) \bmod p = 10$ $TB = (g^SB) \bmod p = 4$
shared key: $S = (TB^SA) \bmod p = 1$ $S' = (TA^SB) \bmod p = 1$

d.



Diffie-Hellman 公共密钥加密算法有可能被中间人攻击。

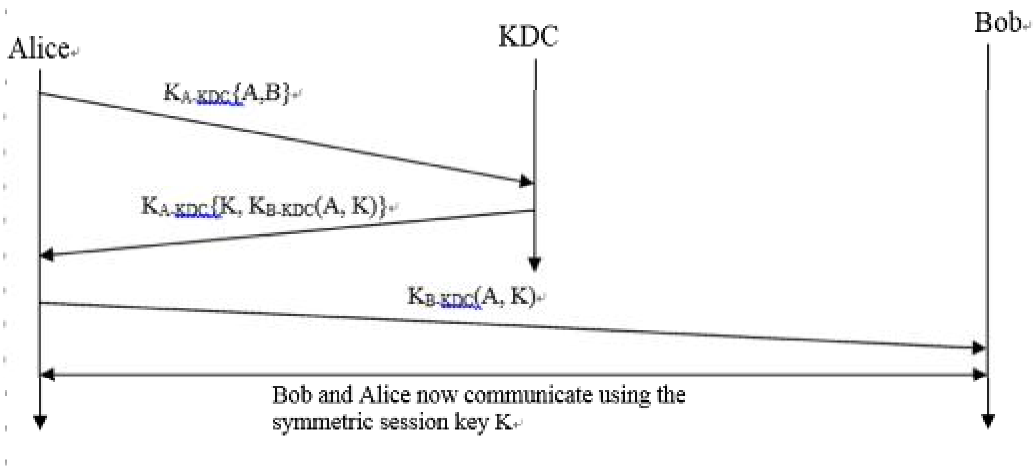
1. 在这个攻击中，Trudy 收到 Alice 发向 Bob 报文的公开值 T_A 和私密值 T_T。
2. 当 Bob 传输他的公开值 T_B，Trudy 向 Alice 发送她的公开值 T_T。
3. Trudy 和 Alice 因此共享密钥 S_{AT}，Trudy 和 Bob 共享密钥 S_{BT}。
4. 在这次交换之后，Trudy 只需要用密钥 S_{AT} 和 S_{BT} 解密 Alice 和 Bob 之间往来的报文。

P10. Suppose Alice wants to communicate with Bob using symmetric key cryptography using a session key KS. In Section 8.2, we learned how public-key cryptography can be used to distribute the session key from Alice to Bob. In this problem, we explore how the session key can be distributed—without public key cryptography—using a key distribution center (KDC). The KDC is a server that shares a unique secret symmetric key with each registered user. For Alice and Bob, denote these keys by KA-KDC and KB-KDC. Design a scheme that uses the KDC to distribute KS to Alice and Bob. Your scheme should use three messages to distribute the session key: a message from Alice to the KDC; a message from the KDC to Alice; and finally a message from Alice to Bob. The first message is KA-KDC (A, B). Using the notation, KA-KDC , KB-KDC , S, A, and B answer the following questions.

a. What is the second message?

b. What is the third message?

答：a 和 b 如下图：



P11. Compute a third message, different from the two messages in Figure 8.8, which has the same checksum as the messages in Figure 8.8.

答：报文为：

I O U 1

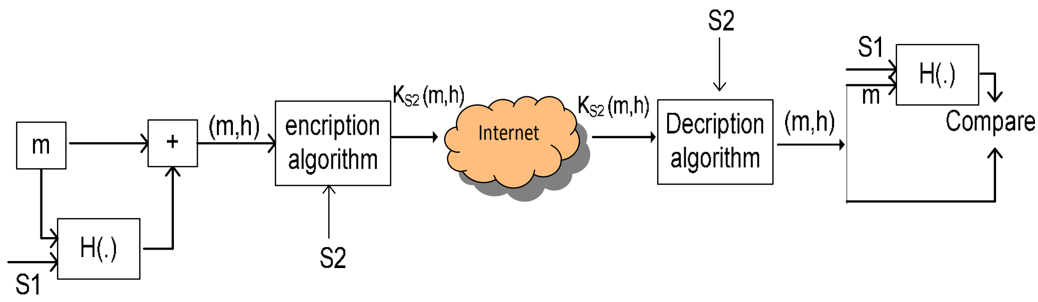
9 0 . 9

0 B O B

有一样的检验和。

P12. Suppose Alice and Bob share two secret keys: an authentication key S1 and a symmetric encryption key S2. Augment Figure 8.9 so that both integrity and confidentiality are provided.

答：如下图：



P13. In the BitTorrent P2P file distribution protocol (see Chapter 2), the seed breaks the file into blocks, and the peers redistribute the blocks to each other. Without any protection, an attacker can easily wreak havoc in a torrent by masquerading as a benevolent peer and sending bogus blocks to a small sub- set of peers in the torrent. These unsuspecting peers then redistribute the bogus blocks to other peers, which in turn redistribute the bogus blocks to even more peers. Thus, it is critical for BitTorrent to have a mechanism that allows a peer to verify the integrity of a block, so that it doesn't redistribute bogus blocks. Assume that when a peer joins a torrent, it initially gets a.torrent file from a fully trusted source. Describe a simple scheme that allows peers to verify the integrity of blocks.

答：这个文件被分成多个相同大小的块，对于每个块，计算其 hash(如 MD5 或者 SHA-1)。无论何时，当一个对等方下载块时，它计算这个块的 hash 值并且将它和 a.torrent 文件中的 hash 值作比较。如果这两个 hash 值相等，这个块是有效的。否则，这个块是伪造的，应该被丢弃。

P14. The OSPF routing protocol uses a MAC rather than digital signatures to provide message integrity. Why do you think a MAC was chosen over digital signatures?

答：数字签名需要一个基于认证中心 CA(certification authorities)提供的 PKI(Public Key Infrastructure)。对于 OSPF 算法，所有路由器在同一个局域网(或者说域 domain)，因而管理员可以很容易的为每个路由器配置对称密钥(symmetric key)，不需要 PKI。

P15. Consider our authentication protocol in Figure 8.18 in which Alice authenticates herself to Bob, which we saw works well (i.e., we found no flaws in it). Now suppose that while Alice is authenticating herself to Bob, Bob must authenticate himself to Alice. Give a scenario by which Trudy, pretending to be Alice, can now authenticate herself to Bob as Alice. (Hint: Consider that the sequence of operations of the protocol, one with Trudy initiating and one with Bob initiating, can be arbitrarily interleaved. Pay particular attention to the fact that both Bob and Alice will use a nonce, and that if care is not taken, the same nonce can be used maliciously.)

答：Bob 最初不知道他谈话的对象是 Trudy 还是 Alice。Bob 和 Alice 共享一个 Trudy 并不知道的私密密钥(secret key) K_{A-B}，Trudy 想要 Bob 将她鉴别成 Alice 。Trudy 将要让 Bob 去认证她，并且等待 Bob 开始：

1. Bob-to-Trudy: “I am Bob”。注释：Bob 开始认证他自己，Bob 对其他人证明自己的证书然后停留几步。

2. Trudy-to-Bob: “I am Alice”。注释：Trudy 开始以 Alice 的身份证明自己。

3. Bob-to-Trudy: “R”。注释：Bob 对第二步发送一个不重数 nonce 作为回应，Trudy 现在不知道 $K_{A-B}(R)$，因而她不能发出响应。

4. Trudy-to-Bob: “R”。注释：Trudy 现在可以继续回应第一步 Bob 的证明，选择 Bob 发送的不重数进行加密，和第三步 Bob 发送给她进行加密的值一样。

5. Bob-to-Trudy: “$K_{A-B}(R)$”。注释：Bob 向其他人，通过第四步他发送的不重数被加密，完成了他自己的证书鉴别。Trudy 现在知道了 $K_{A-B}(R)$。(注意，她没有也不需要 K_{A-B})。

6. Trudy-to-Bob: “$K_{A-B}(R)$”。注释：Trudy 完成了她自己的鉴定，对于第三步 Bob 发送的 R 用 $K_{A-B}(R)$进行回应。由于 Trudy 对于 Bob 第三步用合适的加密的不重数 nonce 进行响应，Bob 就会认为 Trudy 就是 Alice。(注：本题用认证，鉴定均表示 authenticate 的动词名词形式)

P16. A natural question is whether we can use a nonce and public key cryptography to solve the end-point authentication problem in Section 8.4. Consider the following natural protocol: (1) Alice sends the message “I am Alice” to Bob. (2) Bob chooses a nonce, R, and sends it to Alice. (3) Alice uses her private key to encrypt the nonce and sends the resulting value to Bob. (4) Bob applies Alice's public key to the received message. Thus, Bob computes R and authenticates Alice.

a. Diagram this protocol, using the notation for public and private keys employed in the textbook.

b. Suppose that certificates are not used. Describe how Trudy can become a “woman-in-the-middle” by intercepting Alice’s messages and then pre- tending to be Alice to Bob.

答：这个做法实际上不能解决这个问题，正如 Bob 认为(错误的)他已经在图 7.14 的前半部分对 Alice 进行了鉴定，因此 Trudy 也可以欺骗 Alice 并让她错误的认为她就是经过坚定的 Bob。这个问题的根源既不是 Bob 或者 Alice 可以鉴定出来的，而是他们使用的公开密钥被 Trudy 窃取到了。

P17. Figure 8.19 shows the operations that Alice must perform with PGP to pro- vide confidentiality, authentication, and integrity. Diagram the corresponding operations that Bob must perform on the package received from Alice.

答：如下图：

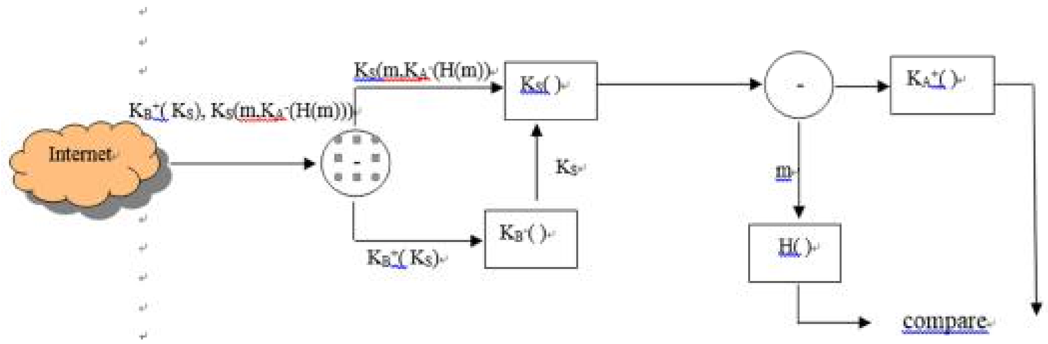


Figure: Operations performed by Bob for confidentiality, integrity, and authentication

P18. Suppose Alice wants to send an e-mail to Bob. Bob has a public-private key pair K_B^+, K_B^-, and Alice has Bob's certificate. But Alice does not have a public, private key pair. Alice and Bob (and the entire world) share the same hash function H(-).

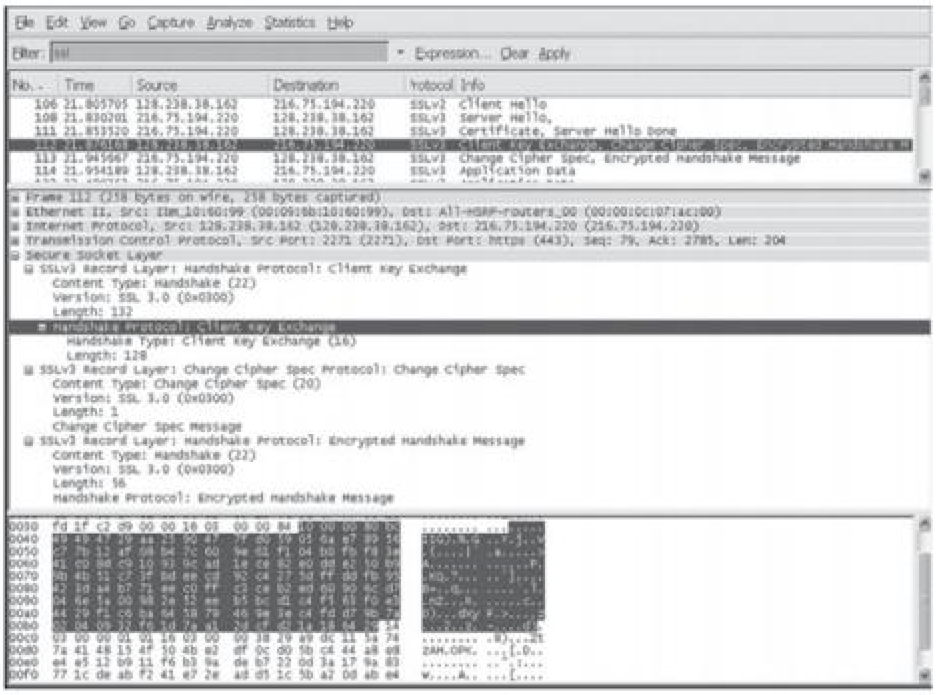
a. In this situation, is it possible to design a scheme so that Bob can verify that Alice created the message? If so, show how with a block diagram for Alice and Bob.

b. Is it possible to design a scheme that provides confidentiality for sending the message from Alice to Bob? If so, show how with a block diagram for Alice and Bob.

答：a. 不，如果没有公开-私有密钥对或者提前共享的秘密，Bob 不能核实 Alice 创造了一个报文；

b. 是的，Alice 仅仅用 Bob 提供的公开密钥对报文进行加密并且将这个加密的报文发送给 Bob。

P19. Consider the Wireshark output below for a portion of an SSL session.



(Wireshark screenshot reprinted by permission of the Wireshark Foundation.)

a. Is Wireshark packet 112 sent by the client or server?

b. What is the server's IP address and port number?

c. Assuming no loss and no retransmissions, what will be the sequence number of the next TCP segment sent by the client?

d. How many SSL records does Wireshark packet 112 contain?

e. Does packet 112 contain a Master Secret or an Encrypted Master Secret or neither?

f . Assuming that the handshake type field is 1 byte and each length field is 3 bytes, what are the values of the first and last bytes of the Master Secret (or Encrypted Master Secret)?

g. The client encrypted handshake message takes into account how many SSL records?

h. The server encrypted handshake message takes into account how many SSL records?

答：a. 是客户机 client 发送。

b. IP 地址: 216.75.194.220, 端口号 port: 443。

c. 283。

d. 3 个 SSL 记录。

e. 是的，包含的是加密的主密钥(Encrypted Master Secret)。

f. 第一个字节：bc，最后一个字节：29。

g. 6。

h. 9。

P20. In Section 8.6.1, it is shown that without sequence numbers, Trudy (a woman- in-the middle) can wreak havoc in an SSL session by interchanging TCP segments. Can Trudy do something similar by deleting a TCP segment? What does she need to do to succeed at the deletion attack? What effect will it have?

答：再一次我们假定 SSL 不提供序号，假定 Trudy，一个中间人，删除一条 TCP 报文段。因而 Bob 什么也不知道，Trudy 对于从 Alice 发送给 Bob 的随后的数据包分组，还需要调整序号，并且对 Bob 发送给 Alice 的确认号也需要调整。结果会是 Bob 没有察觉的错过了字节流中的一个分组。

P21. Suppose Alice and Bob are communicating over an SSL session. Suppose an attacker, who does not have any of the shared keys, inserts a bogus TCP segment into a packet stream with correct TCP checksum and sequence numbers (and correct IP addresses and port numbers). Will SSL at the receiving side accept the bogus packet and pass the payload to the receiving application? Why or why not?

答：不会，伪造的分组无法通过完整性检查(the integrity check)(使用一个共享的报文鉴别码 MAC 密钥)。

P22. The following True/False questions pertain to Figure 8.28.

a. When a host in 172.16.1/24 sends a datagram to an Amazon.com server, the router R1 will encrypt the datagram using IPsec.

b. When a host in 172.16.1/24 sends a datagram to a host in 172.16.2/24, the router R1 will change the source and destination address of the IP datagram.

c. Suppose a host in 172.16.1/24 initiates a TCP connection to a Web server in 172.16.2/24. As part of this connection, all datagrams sent by R1 will have protocol number 50 in the left-most IPv4 header field.

d. Consider sending a TCP segment from a host in 172.16.1/24 to a host in 172.16.2/24. Suppose the acknowledgment for this segment gets lost, so that TCP resends the segment. Because IPsec uses sequence numbers, R1 will not resend the TCP segment.

答：a. F。

b. T。

c. T。

d. F。

P23. Consider the example in Figure 8.28. Suppose Trudy is a woman-in-the- middle, who can insert datagrams into the stream of datagrams going from R1 and R2. As part of a replay attack, Trudy sends a duplicate copy of one of the datagrams sent from R1 to R2. Will R2 decrypt the duplicate datagram and forward it into the branch-office network? If not, describe in detail how R2 detects the duplicate datagram.

答：如果 Trudy 不厌其烦的去改变序号，R2 在 ESP 首部检查序号时会检测出这个复制品(the duplicate)。如果 Trudy 将序号增大，这个分组无法通过 R2 处的完整性检查(integrity check)。

P24. Consider the following pseudo-WEP protocol. The key is 4 bits and the IV is 2 bits. The IV is appended to the end of the key when generating the keystream. Suppose that the shared secret key is 1010. The keystreams for the four possible inputs are as follows:

101000: 0010101101010101001011010100100 . . .

101001: 1010011011001010110100100101101 . . .

101010: 0001101000111100010100101001111 . . .

101011: 1111101010000000101010100010111 . . .

Suppose all messages are 8-bits long. Suppose the ICV (integrity check) is 4-bits long, and is calculated by XOR-ing the first 4 bits of data with the last 4 bits of data. Suppose the pseudo-WEP packet consists of three fields: first the IV field, then the message field, and last the ICV field, with some of these fields encrypted.

a. We want to send the message m = 10100000 using the IV = 11 and using WEP. What will be the values in the three WEP fields?

b. Show that when the receiver decrypts the WEP packet, it recovers the message and the ICV.

c. Suppose Trudy intercepts a WEP packet (not necessarily with the IV = 11) and wants to modify it before forwarding it to the receiver. Suppose Trudy flips the first ICV bit. Assuming that Trudy does not know the keystreams for any of the IVs, what other bit(s) must Trudy also flip so that the received packet passes the ICV check?

d. Justify your answer by modifying the bits in the WEP packet in part (a), decrypting the resulting packet, and verifying the integrity check.

答：a. 由于初始向量 IV=11，字节流为：111110100000 ……….，给出的发送报文 m = 10100000，因此 ICV = 1010 XOR 0000 = 1010。三个 WEP(Wired Equivalent Privacy 有线等效保密)字段值为：

初始向量 IV: 11

加密报文: 10100000 XOR 11111010 = 01011010

加密 ICV: 1010 XOR 0000 = 1010。

b. 接收方提取出初始向量 IV=11 并且生成密钥流 111110100000 ……….。

将加密的报文和密钥流进行异或操作 XOR 来恢复原始的报文：

01011010 XOR 11111010 = 10100000；
将加密的 ICV 和密钥流进行异或操作 XOR 来恢复原始的 ICV：
1010 XOR 0000 = 1010。
接收方然后将恢复的报文最前面的 4 比特和最后面的 4 比特进行异或操作 XOR：
1010 XOR 0000 = 1010(等于恢复的 ICV)。
c. 由于 ICV 是由恢复的报文最前面的 4 比特和最后面的 4 比特进行异或操作 XOR 得到的，因而该报文的第 1 个比特和第 5 个比特二者中的任意一个必须被丢弃(flipped)以便让接收到的报文通过 ICV 检查。
d. 对于 a 部分，加密后报文为 01011010，丢弃第 1 个比特得到：11011010。Trudy 将这个报文和密钥流进行亦或 XOR 操作：
11011010 XOR 11111010 = 00100000
如果 Trudy 将加密的 ICV 的第一个比特舍弃(flipped)，接收方收到的 ICV 的值为 0010。接收方将这个值和密钥流进行亦或操作 XOR 得到 ICV：
0010 XOR 0000 = 0010
接收方现在可以从恢复的报文中计算 ICV：
0010 XOR 0000 = 0010(其值等于恢复的 ICV 因此接收到的分组通过了 ICV 检查)。

P25. Provide a filter table and a connection table for a stateful firewall that is as restrictive as possible but accomplishes the following:
a. Allows all internal users to establish Telnet sessions with external hosts.
b. Allows external users to surf the company Web site at 222.22.0.12.
c. But otherwise blocks all inbound and outbound traffic. The internal network is 222.22/16. In your solution, suppose that the connection table is currently caching three connections, all from inside to outside. You'll need to invent appropriate IP addresses and port numbers.
答：过滤器表如下：

Action	**Source Address**	**Dest Address**	**Protocol**	**Source port**	**Dest port**	**Flag bit**	**Check connection**
allow	222.22/16	outside of 222.22/16	TCP	> 1023	23	any	
allow	outside of 222.22/16	222.22/16	TCP	23	> 1023	ACK	x
Allow	outside of 222.22/16	222.22.0.12	TCP	>1023	80	Any	
Allow	222.22.0.12	outside of 222.22/16	TCP	80	>1023	Any	
deny	All	All	all	all	all	All	

连接表如下：

Source address	**Dest address**	**Source port**	**Dest port**

222.22.1.7	37.96.87.123	12699	23
222.22.93.2	199.1.205.23	37654	23
222.22.65.143	203.77.240.43	48712	23

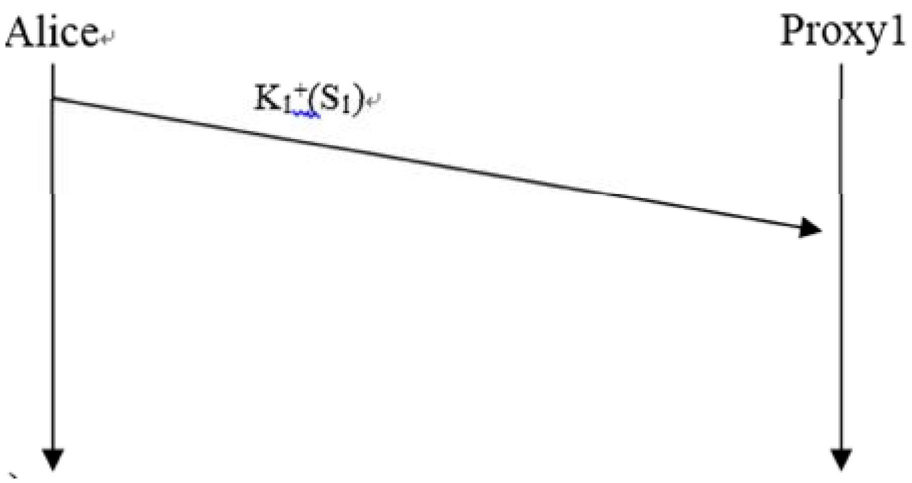
P26. Suppose Alice wants to visit the Web site activist.com using a TOR-like service. This service uses two non-colluding proxy servers, Proxy1 and Proxy2. Alice first obtains the certificates (each containing a public key) for Proxy1 and Proxy2 from some central server. Denote $K_1^+()$,$K_2^+()$and $K_2^+()$ for the encryption/decryption with public and private RSA keys.

a. Using a timing diagram, provide a protocol (as simple as possible) that enables Alice to establish a shared session key S1 with Proxy1. Denote S1(m) for encryption/decryption of data m with the shared key S1.

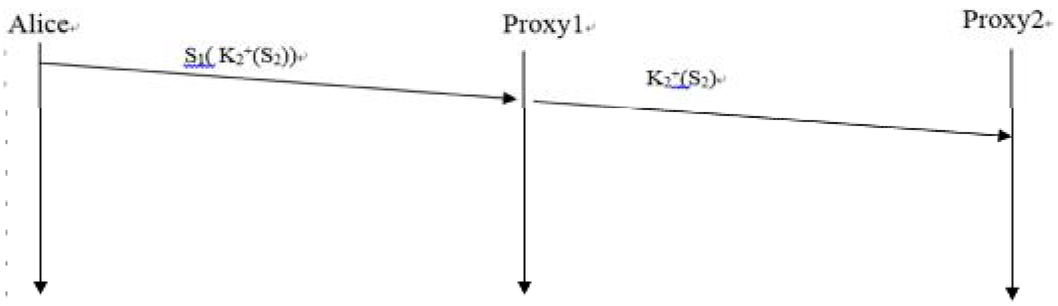
b. Using a timing diagram, provide a protocol (as simple as possible) that allows Alice to establish a shared session key S2 with Proxy2 without revealing her IP address to Proxy2.

c. Assume now that shared keys S1 and S2 are now established. Using a timing diagram, provide a protocol (as simple as possible and not using public-key cryptography) that allows Alice to request an html page from activist.com without revealing her IP address to Proxy2 and without revealing to Proxy1 which site she is visiting. Your diagram should end with an HTTP request arriving at activist.com.

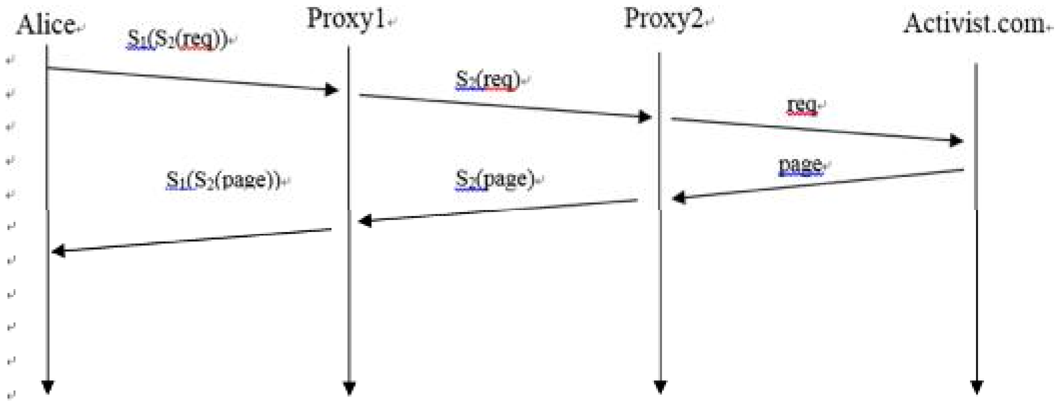
答：a. 图表如下：



b. 图表如下：



c. 图表如下：


Alice
Proxy1
Proxy2
Activist.com
S1(S2(req))
S2(req)
req
page
S2(page)
S1(S2(page))

Chapter 9 Network Management

P1. Consider the two ways in which communication occurs between a managing entity and a managed device: request-response mode and trapping. What are the pros and cons of these two approaches, in terms of (1) overhead, (1) notification time when exceptional events occur, and (3) robustness with respect to lost messages between the managing entity and the device?

答：请求响应模式一般有更多的开销(在信息交互方面的测量)，鉴于以下原因：首先，管理者收到的每一条信息要求两个报文：投票(poll)和响应报文。陷阱模式只需要生成单个报文给发送方。当一个情况发生，如果网络管理者确实需要通知时，polling 有更多的开销，因为很多 polling 报文可能会指出等待情形还没有发生。陷阱模式只有当发生情况时才生成一条报文。当一个事件发生时，陷阱模式会立即通知管理者。用 polling 时，管理者在事件发生和管理者发现(通过 poll 报文)那个事件发生之间，需要等待半个 polling 周期(平均)。

如果陷阱报(trap message)文丢失，被管理的设备不会发送另一个拷贝。如果一个 poll 报文，或者它的响应报文丢失了，管理者会知道存在一个丢失的报文(因为它的回应一直没有到达)。因此如果有必要，管理者会再次 repoll。

P2. In Section 9.3 we saw that it was preferable to transport SNMP messages in unreliable UDP datagrams. Why do you think the designers of SNMP chose UDP rather than TCP as the transport protocol of choice for SNMP?

答：经常，网络管理最被需要的时间是在网络有压力时，当网络严重拥塞和分组发生丢失时。如果用 TCP 运输 SNMP，当网络管理者需要发送 SNMP 报文时，TCP 拥塞控制会严格地使得 SNMP 回退并且停止发送报文。

P3. What is the ASN.1 object identifier for the ICMP protocol (see Figure 9.3)?

答：ICMP 的 ASN.1 对象标识符是 1.3.6.1.2.1.5。

P4. Suppose you worked for a US-based company that wanted to develop its own MIB for managing a product line. Where in the object identifier tree (Figure 9.3) would it be registered? (Hint: You'll have to do some digging through RFCs or other documents to answer this question.)

答：微软文件格式在：1.2.840.113556.4 (参见 9.3.2 节)。

P5. Recall from Section 9.3.2 that a private company (enterprise) can create its own MIB variables under the private branch 1.3.6.4. Suppose that IBM wanted to create a MIB for its Web server software. What would be the next OID qualifier after 1.3.6.1.4? (In order to answer this question, you will need to consult [IANA 2009b]). Search the Web and see if you can find out whether such a MIB exists for an IBM server.

答：详情请参见互联网。

P6. Why do you think the length precedes the value in a TLV encoding (rather than the length following the value)?

答：如果值超出长度，需要增加一个曲解长度作为值的一部分。

P7. Consider Figure 9.9. What would be the BER encoding of {weight, 165}{lastname, "Michael"}?

答：所求 BER 编码为：47Michael21 '10100101'。

P8. Consider Figure 9.9. What would be the BER encoding of {weight, 145}{lastname, “Sridhar”}?

答：所求 BER 编码为：47Sridhar 21 ‘10010001’。