ANDREW S. TANENBAUM

COMPUTER NETWORKS

FOURTH EDITION

PROBLEM SOLUTIONS

soulingm

第 1 章 概述

1. 答:狗能携带 21 千兆字节或者 168 千兆位的数据。18 公里/小时的速度等于 0.005 公里/秒，走过 x 公里的时间为 x / 0.005 = 200x 秒， 产生的数据传输速度为 168/200x Gbps 或者 840 /x Mbps。因此，与通信线路相比较，若 x<5.6 公里，狗有更高的速度。

2. 使用局域网模型可以容易地增加节点。如果局域网只是一条长的电缆，且不会因个别的失效而崩溃(例如采用镜像服务器)的情况下，使用局域网模型会更便宜。使用局域网可提供更多的计算能力和更好交互式接口。

3. 答：横贯大陆的光纤连接可以有很多千兆位/秒带宽， 但是由于光速度传送要越过数千公里，时延将也高。相反，使用 56 kbps 调制解调器呼叫在同一大楼内的计算机则有低带宽和较低的时延。

4. 声音的传输需要相应的固定时间，因此网络时隙数量是很重要的。传输时间可以用标准偏差方式表示。 实际上，短延迟但是大变化性比更长的延迟和低变化性更糟。

5. 答：不，传送.速度为 200,000 公里/秒或 200 米/ 微秒。信号在 10 微秒中传送了 2 千米，每个交换机相当于增加额外的 2 公里电缆。 如果客户和服务器之间的距离为 5000 公里，平均通过 50 个交换机给那些总道路只增加 100 公里，只是 2%。 因此，交换延迟不是这些情形中的主要因素。

6. 答：由于请求和应答都必须通过卫星，因此传输总路径长度为 160,000 千米。在空气和真空中的光速为 300，000 公里/秒， 因此最佳的传播延迟为 160,000/300，000 秒，约 533 msec。

7. 显而易见，在这里没有正确的独立的答案。但下列问题好像相关：目前的系统有它的很多惯性(检测和平衡)。 当新的团体掌握权力的时候，这惯性可保持法律、经济和社会制度的稳定。 此外，很多人对社会问题没有真的知道事情的真相，但却具有很强烈的、引起争论的意见。 将不允许讲道理的观点写进法律也许不合适。还必须考虑某些专业组织有影响的宣传活动。另一主要问题是安全。黑客可能侵入系统和伪造结果。

8. 答：将路由器称为A，B，C，D 和E.：则有 10 条可能的线路；*AB*, *AC*, *AD*, *AE*, *BC*, *BD*, *BE*, *CD*, *CE*,和*DE*。 每条线路有 4 种可能性(3 速度或者不是线路)，这样，拓扑的总数为 4^{10} = 1,048,576。

检查每个拓扑需要 100 ms，全部检查总共需要 104,857. 6 秒，或者稍微超过 29 个小时。

9. 答：

这意味着，从路由器到路由器的路径长度相当于路由器到根的两倍。 若在树中，根深度为 1，深度为 n，从根到第 n 层需要 n-1 跳，在该层的路由器为 0.50。

从根到n-1 层的路径有router的 0.25 和n^{-}-2 跳步。 因此，路径长度*l*为：

$$l = 0.5 \times (n-1) + 0.25 \times (n-2) + 0.125 \times (n-3) + \cdots$$

或

$$l = \sum_{i=1}^{\infty} n\,(0.5)^i - \sum_{i=1}^{\infty} i(0.5)^i$$

This expression reduces to $l = n - 2$, The mean router-router 路径为 2n-4。

10. 区分*n*-2 事件。 事件 1 到n由主机成功地、没有冲突地使用这条信道的事件组成。

这些可能性的事件的概率为$p(1-p)^{n-1}$。事件$n+1$是一个空闲的信道，其概率为$(1-p)^n$。事件$n+2$是一个冲突。由于事件$n+2$互斥，它们可能发生的事件必须统一合计。 冲突的可能性等于那些小部分的槽的浪费，只是

$$1-np(1-p)^{n-1}-(1-p)^n$$

11. 答：通过协议分层可以把设计问题划分成较小的易于处理的片段。分层意味着某一层的协议的改变不会影响高层或低层的协议。

12. 答：不.，在 ISO 协议模型中，物理通讯只在最低的层里进行，不在每个层里。

13. 无连接通信和面向连接通信的最主要区别是什么？

答：主要的区别有两条。

其一：面向连接通信分为三个阶段，第一是建立连接，在此阶段，发出一个建立连接的请求。只有在连接成功建立之后，才能开始数据传输，这是第二阶段。接着，当数据传输完毕，必须释放连接。而无连接通信没有这么多阶段，它直接进行数据传输。

其二：面向连接的通信具有数据的保序性， 而无连接的通信不能保证接收数据的顺序与发送数据的顺序一致。

14. 答：不相同。在报文流中，网络保持对报文边界的跟踪；而在字节流中，网络不做这样的跟踪。例如，一个进程向一条连接写了 1024 字节，稍后又写了另外 1024 字节。那么接收方共读了 2048 字节。对于报文流，接受方将得到两个报文。每个报文 1024 字节。 而对于字节流，报文边界不被识别。接收方把全部的 2048 个字节当作一个整体，在此已经体现不出原先有两个报文的事实。

15. 答：协商就是要让双方就在通信期间将使用的某些参数或数值达成一致。最大分组长度就是一个例子。

16. 服务是由 k 层向 k＋1 层提供的。服务必须由下层 k 提供，即，对层 k 的服务是由 k- 1 层提供的。

17. The probability, Pk, of a frame requiring exactly k transmissions is the probability of the first k-1 attempts failing, p^{k-1}, times the probability of the k-th transmission succeeding, $(1-p)$. The mean number of transmission is then just

$$\sum_{k=1}^{\infty} kP_k = \sum_{k=1}^{\infty} k(1-p)p^{k-1} = \frac{1}{1-p}$$

18. OSI 的哪一层分别处理以下问题？

把传输的比特流划分为帧——数据链路层

决定使用哪条路径通过子网——网络层.

19. **答：**帧封装包。 当一个包到达数据链路层时，整个数据包，包括包头、数据及全部内容，都用作帧的数据区。或者说，将整个包放进一个信封(帧)里面，(如果能装入的话)。

20. 一个有 n 层协议的系统，应用层生成长度为 m 字节的报文，在每层都加上 h 字节报头，那么网络带宽中有多大百分比是在传输各层报头？

hn/(hn+m)*100%

[注意：题中已说明每层都要附加报头，不要考虑实际的 OSI 或者 TCP/IP 协议]

21. 相似点：都是独立的协议栈的概念；层的功能也大体相似。

不同点：OSI 更好的区分了服务、接口和协议的概念，因此比 TCP/IP 具有更好的隐藏性，能够比较容易的进行替换；OSI 是先有的模型的概念，然后再进行协议的实现，而 TCP/IP 是先有协议，然后建立描述该协议的模型；层次数量有差别；TCP/IP 没有会话层和表示层，OSI 不支持网络互连。OSI 在网络层支持无连接和面向连接的通信，而在传输层仅有面向连接的通信，而 TCP/IP 在网络层仅有一种通信模式（无连接），但在传输层支持两种模式。

22. TCP 是面向连接的，而 UDP 是一种数据报服务。

23. 如果 3 枚炸弹炸毁与右上角那 2 个节点连接的 3 个节点，可将那 2 个节点与其余的节点拆开。系统能禁得住任何两个节点的损失。

24. Doubling every 18 months means a factor of four gain in 3 years. In 9 years, the gain is then 43 or 64, leading to 6.4 billion hosts. My intuition says that is much too conservative, since by then probably every television in the world and possibly billions of other appliances will be on home LANs connected to the Internet. The average person in the developed world may have dozens of Internet hosts by then.

25. 如果网络容易丢失分组，那么对每一个分组逐一进行确认较好，此时仅重传丢失的分组。而在另一方面，如果网络高度可靠，那么在不发差错的情况下，仅在整个文件传送的结尾发送一次确认，从而减少了确认的次数，节省了带宽；不过，即使有单个分组丢失，也需要重传整个文件。

26. Small, fixed-length cells can be routed through switches quickly, and completely in hardware. Small, fixed-size cells also make it easier to build hardware that handles many cells in parallel. Also, they do not block transmission lines for very long, making it easier to provide quality-of-service guarantees.

27. The speed of light in coax is about 200,000 km/sec, which is 200 meters/ sec. At 10 Mbps, it takes 0.1 sec to transmit a bit. Thus, the bit lasts 0.1 sec in time, during which it propagates 20 meters. Thus, a bit is 20 meters long here.

28. The image is 1024 768 3 bytes or 2,359,296 bytes. This is 18,874,368 bits. At 56,000 bits/sec, it takes about 337.042 sec. At 1,000,000 bits/sec, it takes about 18.874 sec. At 10,000,000 bits/sec, it takes about 1.887 sec. At 100,000,000 bits/sec, it takes about 0.189 sec.

29. Think about the hidden terminal problem. Imagine a wireless network of five stations, *A* through *E*, such that each one is in range of only its immediate neighbors. Then *A* can talk to *B* at the same time *D* is talking to *E*. Wireless networks have potential parallelism, and in this way differ from Ethernet.

30. One disadvantage is security. Every random delivery man who happens to be in the building can listen in on the network. Another disadvantage is reliability. Wireless networks make lots of errors. A third potential problem is battery life, since most wireless devices tend to be mobile.

31. 优点 1：如果每个人都使用标准，那么每个人都可以与其他任何人交流；优点 2：广泛使用标准将导致规模经济，比如生产大规模集成电路芯片。缺点 1：为了取得标准化所

需要的政治妥协经常会导致差的标准；缺点 2：一旦标准被广泛采用了，要对它再做改变就会非常困难，即使发现了新的更好的技术或方法，也难以替换。

32. 具有国际标准的系统的例子包括 CD 播放器和 CD 盘片，随声听和录音磁带，照相机和 35mm 胶卷等。缺乏国际标准的领域包括合适录像机和录像带（美国是 NTSC VHS，欧洲是 PAL），手提电话，电灯和灯泡（不同的国家使用不同的电压），影印机和纸（美国为 8.5*11 英寸，其他地方为 A4）等。

第 2 章 物理层

1.

答：本题是求周期性函数的傅立叶系数。而题面中所给出的为信号在一个周期内的解析式。

$$g(t)=\frac{1}{2}c+\sum_{n=1}^{\infty}a_n\sin(2\pi nft)+\sum_{n=1}^{\infty}b_n\cos(2\pi nft)$$

$$f(t)=t \qquad T=1 \qquad f=1/T=1$$

$$a_n=\frac{2}{T}\int_0^1 f(t)\sin(2\pi nft)dt=\frac{-1}{\pi n}$$

$$b_n=\frac{2}{T}\int_0^1 f(t)\cos(2\pi nft)dt=0$$

$$c=\frac{2}{T}\int_0^1 f(t)dt=1$$

即；$a_n=\frac{-1}{\pi n},\ b_n=0,\ c=1.$

2. 答：无噪声信道最大数据传输率公式：最大数据传输率=2Hlog$_2$V b/s。因此最大数据传输率决定于每次采样所产生的比特数，如果每次采样产生 16bits，那么数据传输率可达 128kbps；如果每次采样产生 1024bits，那么可达 8.2Mbps。注意这是对无噪声信道而言的，实际信道总是有噪声的，其最大数据传输率由香农定律给出。

3. 答：采样频率 12MHz，每次采样 2bit，总的数据率为 24Mbps。

4. 答：信噪比为 20 dB 即 $S/N=$ 100.由于 log2101≈6.658，由香农定理，该信道的信道容量为 3log$_2$(1 +100) =19.98kbps。

又根据乃奎斯特定理，发送二进制信号的 3kHz 信道的最大数据传输速率为

2*3 log$_2$2=6 kbps。

所以可以取得的最大数据传输速率为 6kbps。

5. 答：为发送 T1 信号，我们需要

$$H\log_2(1+\frac{S}{N})=1.544\times10H^6$$

$$H=50000$$

$$\frac{S}{N}=2^{31}-1$$

$$10\log_{10}(2^{31}-1)=93\ \text{dB}$$

所以，在 50kHz 线路上使用 T1 载波需要 93dB 的信噪比。

6. 答：无源星没有电子器件，来自一条光纤的光照亮若干其他光纤。有源中继器把光信号转换成电信号以作进一步的处理。

7. 答：

$$f=\frac{c}{\lambda} \qquad \frac{df}{d\lambda}=\frac{c}{\lambda^2}$$

$$df=\frac{c}{\lambda^2}d\lambda \qquad \Delta f=\frac{c}{\lambda^2}\Delta\lambda$$

$c = 3\times10^8 \qquad \lambda = 10^{-6}m$

$\Delta\lambda = 0.1\times10^{-6} = 10^{-7}m$

$\Delta f = \frac{3\times10^8}{(10^{-6})^2}\times10^{-7} = 30\times10^{12}Hz = 30THz$

因此，在 0.1μm的频段中可以有 30THz。

8. 答：数据速率为 480× 640×24× 60bps，即 442Mbps。

$\Delta f = 4.42\times10^8$

$f = \frac{c}{\lambda} \quad \frac{df}{d\lambda} = -\frac{c}{\lambda^2}$

$|\Delta f| = \frac{\lambda^2\Delta f}{c} = \frac{(1.3\times10^{-6})^2\times4.42\times10^8}{3\times10^8} = 2.5\times10^{-12}m = 2.5\times10^{-6}\mu m$

需要 442Mbps 的带宽，对应的波长范围是$2.5\times10^{-6}\mu m$。

9. 答：奈奎斯特定理是一个数学性质，不涉及技术处理。该定理说，如果你有一个函数，它的傅立叶频谱不包含高于 f 的正弦和余弦，那么以 2 f 的频率采样该函数，那么你就可以获取该函数所包含的全部信息。因此奈奎斯特定理适用于所有介质。

10. 答：3 个波段的频率范围大约相等，根据公式

$\Delta f = \frac{c}{\lambda^2}\Delta f$

λ小的波段$\Delta\lambda$ 也小，才能保持Δf 大约相等。

顺便指出，3 个带宽大致相同的事实是所使用的硅的种类的一个碰巧的特性反映。

11. 答：

$f = \frac{c}{\lambda}$

当λ为 1cm 时，f 为 30GHz。
当λ为 5m 时，f 为 60MHz。

12. 答：1GHz 微波的波长是 30cm。如果一个波比另一个波多行进 15cm，那么它们到达时将 180 异相。显然，答案与链路长度是 50km 的事实无关。

13. 答：

$A = arctg\frac{0.001}{100} = 0.00057^o$

If the beam is off by 1 mm at the end, it misses the detector. This amounts to a triangle with base 100 m and height 0.001 m. The angle is one whose tangent is thus 0.00001. This angle is about 0.00057 degrees.

14. With 66/6 or 11 satellites per necklace, every 90 minutes 11 satellites pass overhead. This means there is a transit every 491 seconds. Thus, there will be a handoff about every 8 minutes and 11 seconds.

15. The satellite moves from being directly overhead toward the southern horizon, with a maximum excursion from the vertical of 2 . It takes 24 hours to go from directly overhead to maximum excursion and then back.

16. The number of area codes was 8× 2× 10, which is 160. The number of prefixes was 8× 8 ×10, or 640. Thus, the number of end offices was limited to 102,400. This limit is not a problem.

17. With a 10-digit telephone number, there could be 1010 numbers, although many of the area codes are illegal, such as 000. However, a much tighter limit is given by the number of end offices. There are 22,000 end offices, each with a maximum of 10,000 lines. This gives a maximum of 220 million telephones. There is simply no place to connect more of them. This could never be achieved in practice because some end offices are not full. An end office in a small town in Wyoming may not have 10,000 customers near it, so those lines are wasted.

18. 答：每部电话每小时做 0.5 次通话，每次通话 6 分钟。因此一部电话每小时占用一条电路 3 分钟，60/3=20，即 20 部电话可共享一条线路。由于只有 10%的呼叫是长途，所以 200 部电话占用一条完全时间的长途线路。局间干线复用了 1000000/4000=250 条线路，每条线路支持 200 部电话，因此，一个端局可以支持的电话部数为 200*250=50000。

19. 答：双绞线的每一条导线的截面积是$\pi \times (1/2)^2 = 0.25\pi$ mm²，每根双绞线的两条导线在 10km 长的情况下体积是$0.25\pi \times (10^{-3})^2 \times 10 \times 1000 \times 2 = 0.5\pi \times 10^{-2}$ m³，即约为 15708cm。由于铜的密度等于 9.0g/cm^3，每个本地回路的质量为 9×15708 =141372 g，约为 141kg。这样，电话公司拥有的本地回路的总质量等于 141×1000×10^4= 1.41× 10 ^{9}kg，由于每千克铜的价格是 3 美元，所以总的价值等于 3× 1.4×10 ^{9}=4.2 × 10^{9}美元。

20. Like a single railroad track, it is half duplex. Oil can flow in either direction, but not both ways at once.

21. 通常在物理层对于在线路上发送的比特不采取任何差错纠正措施。在每个调制解调器中都包括一个 CPU 使得有可能在第一层中包含错误纠正码，从而大大减少第二层所看到的错误率。由调制解调器做的错误处理可以对第二层完全透明。现在许多调制解调器都有内建的错误处理功能。

22. 每个波特有 4 个合法值，因此比特率是波特率的两倍。对应于 1200 波特，数据速率是 2400bps。

23. 相位总是 0，但使用两个振幅，因此这是直接的幅度调制。

24. If all the points are equidistant from the origin, they all have the same amplitude, so amplitude modulation is not being used. Frequency modulation is never used in constellation diagrams, so the encoding is pure phase shift keying.

25. Two, one for upstream and one for downstream. The modulation scheme itself just uses amplitude and phase. The frequency is not modulated.

26. There are 256 channels in all, minus 6 for POTS and 2 for control, leaving 248 for data. If 3/4 of these are for downstream, that gives 186 channels for downstream. ADSL modulation is at 4000 baud, so with QAM-64 (6 bits/baud) we have 24,000 bps in each of the 186 channels. The total bandwidth is then 4.464 Mbps downstream.

27. A 5-KB Web page has 40,000 bits. The download time over a 36 Mbps channel

is 1.1 msec. If the queueing delay is also 1.1 msec, the total time is 2.2 msec. Over ADSL there is no queueing delay, so the download time at 1 Mbps is 40 msec. At 56 kbps it is 714 msec.

28. There are ten 4000 Hz signals. We need nine guard bands to avoid any interference. The minimum bandwidth required is 4000× 10 400×9 =43,600 Hz.

29. 答：125μs的采样时间对应于每秒 8000 次采样。一个典型的电话通道为 4kHz。根据奈奎斯特定理，为获取一个 4kHz 的通道中的全部信息需要每秒 8000 次的采样频率。

(Actually the nominal bandwidth is somewhat less, but the cutoff is not sharp.)

30. 每一帧中，端点用户使用 193 位中的 168（7*24）位，开销占 25（=193-168）位，因此开销比例等于 25/193=13%。

31. 答：比较使用如下方案的无噪声 4kHz 信道的最大数据传输率:

(a) 每次采样 2 比特的模拟编码 ——16kbps

(b) T1 PCM 系统——56kbps

In both cases 8000 samples/sec are possible. With dibit encoding, two bits are sent per sample. With T1, 7 bits are sent per period. The respective data rates are 16 kbps and 56 kbps.

32. 答：10 个帧。

在数字通道上某些随机比特是 0101010101 模式的概率是 1/1024。察看 10 个帧，若每一帧中的第一位形成比特串 0101010101，则判断同步成功，而误判的概率为 1/1024，小于 0.001。

33. 答：有。编码器接受任意的模拟信号，并从它产生数字信号。而解调器仅仅接受调制了的正弦（或余弦）波，产生数字信号。

34. 答：a. CCITT 2.048Mbps 标准用 32 个 8 位数据样本组成一个 125μs的基本帧，30 个信道用于传信息，2 个信道用于传控制信号。在每一个 4kHz 信道上发送的数据率就是

8*8000=64kbps。

b. 差分脉码调制（DPCM）是一种压缩传输信息量的方法，它发送的不是每一次抽样的二进制编码值，而是两次抽样的差值的二进制编码。现在相对差值是 4 位，所以对应每个 4kHz 信道实际发送的比特速率为 4*8000=32bps。

c. 增量调制的基本思想是：当抽样时间间隔 $s\ t$ 很短时，模拟数据在两次抽样之间的变化很小，可以选择一个合适的量化值? 作为阶距。把两次抽样的差别近似为不是增加一个? 就是减少一个? 。这样只需用 1bit 二进制信息就可以表示一次抽样结果，而不会引入很大误差。因此，此时对应每个 4kHz 信道实际发送的数据速率为 1*8000=8kHz。

35. 答：在波的 1/4 周期内信号必须从 0 上升到 A。为了能够跟踪信号，在 T/4 的时间内（假定波的周期是 T）必须采样 8 次，即每一个全波采样 32 次，采样的时间间隔是 1/x，因此波的全周期必须足够的长，使得能包含 32 次采样，即 $T > 32/x$，或 f max =x/32。

36. 答：10^{-9} 的漂移意味着 10^{9} 秒中的 1 秒，或 1 秒中的 10^{-9} 秒。对于OC-1 速率，即 51.840Mbps，取近似值 50Mbps，大约一位持续 20ns。这就说明每隔 20 秒，时钟就要偏离 1 位。这就说明，时钟必须每隔 10 秒或更频繁地进行同步，才能保持不会偏离太大。

37. 答：基本的 SONET 帧是美 125μs产生 810 字节。由于 SONET 是同步的，因此不论是

否有数据，帧都被发送出去。每秒 8000 帧与数字电话系统中使用的 PCM 信道的采样频率完全一样。

810 字节的 SONET 帧通常用 90 列乘以 9 行的矩形来描述，每秒传送 51.84Mbps，即 8×810×8000＝51840000bps。这就是基本的 SONET 信道，它被称作同步传输信号 STS-1，所有的 SONET 干线都是由多条 STS-1 构成。

每一帧的前 3 列被留作系统管理信息使用，前 3 行包含段开销，后 6 行包含线路开销。

剩下的 87 列包含 87×9×8×8000＝50112000bps。被称作同步载荷信封的数据可以在任何位置开始。线路开销的第一行包含指向第一字节的指针。同步载荷信封（SPE）的第一列是通路开销。

通路开销不是严格的 SONET 结构，它在嵌入在载荷信封中。通路开销端到端的流过网络，因此把它与端到端的运载用户信息的 SPE 相关联是有意义的。然而，它确实从可提供给端点用户的 50.112Mbps 中又减去 1×9×8×8000＝576000bps，即 0.576Mbps，使之变成 49.536Mbps 。OC-3 相当于 3 个 OC-1 复用在一起，因此其用户数据传输速率是 49.546× 3 ＝ 148.608 Mbps。

38. VT1.5 can accommodate 8000 frames/sec ×3 columns× 9 rows× 8 bits =1.728 Mbps. It can be used to accommodate DS-1. VT2 can accommodate 8000 frames/sec × 4 columns× 9 rows × 8 bits = 2.304 Mbps. It can be used to accommodate European CEPT-1 service. VT6 can accommodate 8000 frames/sec× 12 columns× 9 rows× 8 bits = 6.912 Mbps. It can be used to accommodate DS-2 service.

39. Message switching sends data units that can be arbitrarily long. Packet switching has a maximum packet size. Any message longer than that is split up into multiple packets.

40. 答：当一条线路（例如 OC-3）没有被多路复用，而仅从一个源输入数据时，字母 c（表示 conactenation，即串联）被加到名字标识的后面，因此，OC-3 表示由 3 条单独的 OC-1 线路复用成 155.52Mbps，而 OC-3c 表示来自单个源的 155.52Mbps 的数据流。OC-3c 流中所包含的 3 个 OC-1 流按列交织编排，首先是流 1 的第 1 列，流 2 的第 1 列，流 3 的第 1 列，随后是流 1 的第 2 列，流 2 的第 2 列，……以此类推，最后形成 270 列宽 9 行高的帧。

OC-3c 流中的用户实际数据传输速率比 OC-3 流的速率略高（149.760Mbps 和 148.608Mbps），因为通路开销仅在 SPE 中出现一次，而不是当使用 3 条单独 OC-1 流时出现的 3 次。换句话说，OC-3c 中 270 列中的 260 列可用于用户数据，而在 OC-3 中仅能使用 258 列。更高层次的串联帧（如 OC-12c）也存在。

OC-12c 帧有 12*90=1080 列和 9 行。其中段开销和线路开销占 12*3=36 列，这样同步载荷信封就有 1080-36=1044 列。SPE 中仅 1 列用于通路开销，结果就是 1043 列用于用户数据。

由于每列 9 个字节，因此一个 OC-12c 帧中用户数据比特数是 8 × 9×1043＝75096。每秒 8000 帧，得到用户数据速率 75096×8000 =600768000bps，即 600.768Mbps。

所以，在一条 OC-12c 连接中可提供的用户带宽是 600.768Mbps。

41. 答：The three networks have the following properties:

星型：最好为 2，最差为 2，平均为 2；

环型：最好为 1，最差为 n/2，平均为 n/4

如果考虑 n 为奇偶数，

则 n 为奇数时，最坏为（n-1）/2，平均为（n+1）/4

n 为偶数时，最坏为 n/2，平均为$n^2/4(n$ 1)

全连接：最好为 1，最差为 1，平均为 1。

42. 对于电路交换， $t= s$ 时电路建立起来； t ＝$s+$ x / d 时报文的最后一位发送完毕； t = $s+$ $x/b+kd$ 时报文到达目的地。而对于分组交换，最后一位在 $t=x/b$ 时发送完毕。

为到达最终目的地，最后一个分组必须被中间的路由器重发 k 1 次，每次重发花时间 p/ b，所以总的延迟为

$$\frac{x}{b}+(k-1)\frac{p}{b}+kd$$

为了使分组交换比电路交换快，必须：

$$\frac{x}{b}+(k-1)\frac{p}{b}+kd<s+\frac{x}{b}+kd$$

所以：

$$s>(k-1)\frac{p}{b}$$

43. 答：所需要的分组总数是 x /p ，因此总的数据加上头信息交通量为 $(p+h)x/p$ 位。

源端发送这些位需要时间为 $(p+h$ $)x$ $/pb$

中间的路由器重传最后一个分组所花的总时间为 $(k-1)(p$ $+h$ $)/ b$

因此我们得到的总的延迟为

$$(p+h)\frac{x}{pb}+(p+h)(k-1)\frac{1}{b}$$

对该函数求 p 的导数，得到

$$\frac{p-(p+h)}{p^2}\frac{x}{b}+\frac{k-1}{b}$$

令

$$\frac{p-(p+h)}{p^2}\frac{x}{b}+\frac{k-1}{b}=0$$

得到

$$\frac{hx}{p^2}=k-1$$

因为 $p>0$，所以

$$p=\sqrt{\frac{hx}{k-1}}$$

故

$$p=\sqrt{\frac{hx}{k-1}}$$

时能使总的延迟最小。

44. Each cell has six neighbors. If the central cell uses frequency group *A*, its six neighbors can use *B*, *C*, *B*, *C*, *B*, and *C* respectively. In other words, only 3 unique cells are needed. Consequently, each cell can have 280 frequencies.

45. First, initial deployment simply placed cells in regions where there was high density of human or vehicle population. Once they were there, the operator often did not want to go to the trouble of moving them. Second, antennas are typically placed on tall buildings or mountains. Depending on the exact location of such structures, the area covered by a cell may be irregular due to obstacles near the transmitter. Third, some communities or property owners do not allow building a tower at a location where the center of a cell falls. In such cases, directional antennas are placed at a location not at the cell center.

46. If we assume that each microcell is a circle 100 m in diameter, then each cell has an area of 2500 . If we take the area of San Francisco, 1.2 108 m2 and divide it by the area of 1 microcell, we get 15,279 microcells. Of course, it is impossible to tile the plane with circles (and San Francisco is decidedly three-dimensional), but with 20,000 microcells we could probably do the job.

47. Frequencies cannot be reused in adjacent cells, so when a user moves from one cell to another, a new frequency must be allocated for the call. If a user moves into a cell, all of whose frequencies are currently in use, the user's call must be terminated.

48. It is not caused directly by the need for backward compatibility. The 30 kHz channel was indeed a requirement, but the designers of D-AMPS did not have to stuff three users into it. They could have put two users in each channel, increasing the payload before error correction from 260 ×50= 13 kbps to 260× 75 = 19.5 kbps. Thus, the quality loss was an intentional trade-off to put more users per cell and thus get away with bigger cells.

49. D-AMPS uses 832 channels (in each direction) with three users sharing a single channel. This allows D-AMPS to support up to 2496 users simultaneously per cell. GSM uses 124 channels with eight users sharing a single channel. This allows GSM to support up to 992 users simultaneously. Both systems use about the same amount of spectrum (25 MHz in each direction).

D-AMPS uses 30 KHz× 892 = 26.76 MHz. GSM uses 200 KHz × 124 =24.80 MHz. The difference can be mainly attributed to the better speech quality provided by GSM (13 Kbps per user) over D-AMPS (8 Kbps per user).

50. The result is obtained by negating each of *A*, *B*, and *C* and then adding the three chip sequences. Alternatively the three can be added and then negated.

The result is (+3 +1 +1 1 3 1 1 +1).

51. By definition $\mathbf{S}\bullet\mathbf{T} \equiv \frac{1}{m}\sum_{i=1}^{m} S_i T_i$

If *T* sends a 0 bit instead of 1 bit, its chip sequence is negated, with the *i*-th element becoming *Ti* . Thus,

$$\mathbf{S}\bullet\mathbf{T} \equiv \frac{1}{m}\sum_{i=1}^{m} S_i(-T_i) = -\frac{1}{m}\sum_{i=1}^{m} S_i T_i = 0$$

52. When two elements match, their product is +1. When they do not match, their product is 1. To make the sum 0, there must be as many matches as mismatches. Thus, two chip sequences are orthogonal if exactly half of the corresponding elements match and exactly half do not match.

53. Just compute the four normalized inner products:

(1 +1 3 +1 1 3 +1 +1) d (1 1 1 +1 +1 1 +1 +1)/8 = 1

(1 +1 3 +1 1 3 +1 +1) d (1 1 +1 1 +1 +1 +1 1)/8 = 1

(1 +1 3 +1 1 3 +1 +1) d (1 +1 1 +1 +1 +1 1 1)/8 = 0

(1 +1 3 +1 1 3 +1 +1) d (1 +1 1 1 1 1 +1 1)/8 = 1

The result is that *A* and *D* sent 1 bits, *B* sent a 0 bit, and *C* was silent.

54. 答：可以，每部电话都能够有自己到达端局的线路，但每路光纤都可以连接许多部电话。忽略语音压缩，一部数字 PCM 电话需要 64kbps 的带宽。如果以 64kbps 为单元来分割 10Gbps，我们得到每路光缆串行 156250 家。现今的有线电视系统每根电缆串行数百家。

55. 答：它既像 TDM，也像 FDM。100 个频道中的每一个都分配有自己的频带（FDM），在每个频道上又都有两个逻辑流通过 TDM 交织播放（节目和广告交替使用频道）。

This example is the same as the AM radio example given in the text, but neither is a fantastic example of TDM because the alternation is irregular.

56. A 2-Mbps downstream bandwidth guarantee to each house implies at most 50 houses per coaxial cable. Thus, the cable company will need to split up the existing cable into 100 coaxial cables and connect each of them directly to a fiber node.

57. The upstream bandwidth is 37 MHz. Using QPSK with 2 bits/Hz, we get 74 Mbps upstream. Downstream we have 200 MHz. Using QAM-64, this is 1200 Mbps. Using QAM-256, this is 1600 Mbps.

58. Even if the downstream channel works at 27 Mbps, the user interface is nearly always 10-Mbps Ethernet. There is no way to get bits to the computer any faster than 10-Mbps under these circumstances. If the connection between the PC and cable modem is fast Ethernet, then the full 27 Mbps may be available. Usually, cable operators specify 10 Mbps Ethernet because they do not want one user sucking up the entire bandwidth.

第 3 章 数据链路层

1. 答：由于每一帧有 0.8 的概率正确到达，整个信息正确到达的概率为 $p=0.8^{10}=0.107$。

为使信息完整的到达接收方，发送一次成功的概率是p ，二次成功的概率是$(1-p)p$，三次成功的概率为$(1-p)^2 p$，i 次成功的概率为$(1-p)^{i-1} p$，因此平均的发送次数等于：

$$E=\sum_{i=1}^{\infty} ip(1-p)^{i-1}=\frac{1}{p}=\frac{1}{0.107}\approx 9.3$$

2. The solution is

(a) 00000100 01000111 11100011 11100000 01111110

(b) 01111110 01000111 11100011 11100000 11100000 11100000 01111110 01111110

(c) 01111110 01000111 110100011 111000000 011111010 01111110

3. After stuffing, we get A B ESC ESC C ESC ESC ESC FLAG ESC FLAG D.

4. If you could always count on an endless stream of frames, one flag byte might be enough. But what if a frame ends (with a flag byte) and there are no new frames for 15 minutes. How will the receiver know that the next byte is actually the start of a new frame and not just noise on the line? The protocol is much simpler with starting and ending flag bytes.

5. The output is 011110111110011111010.

6. 答：可能。假定原来的正文包含位序列 01111110 作为数据。位填充之后，这个序列将变成 01111010。如果由于传输错误第二个 0 丢失了，收到的位串又变成 01111110，被接收方看成是帧尾。然后接收方在该串的前面寻找检验和，并对它进行验证。如果检验和是 16 位，那么被错误的看成是检验和的 16 位的内容碰巧经验证后仍然正确的概率是 $1/2^{16}$。如果这种概率的条件成立了，就会导致不正确的帧被接收。显然，检验和段越长，传输错误不被发现的概率会越低，但该概率永远不等于零。

7. 答：如果传播延迟很长，例如在探测火星或金星的情况下，需要采用前向纠错的方法。还有在某些军事环境中，接收方不想暴露自己的地理位置，所以不宜发送反馈信号。如果错误率足够的低，纠错码的冗余位串不是很长，又能够纠正所有的错误，前向纠错协议也可能是比较合理和简单的。

8. Making one change to any valid character cannot generate another valid character due to the nature of parity bits. Making two changes to even bits or two changes to odd bits will give another valid character, 所以 Hamming 距离为 2

9. Parity bits are needed at positions 1, 2, 4, 8, and 16, so messages that do not extend beyond bit 31 (including the parity bits) fit. Thus, five parity bits are sufficient. The bit pattern transmitted is 01101011001100111010 1

10. The encoded value is 101001001111.

11. If we number the bits from left to right starting at bit 1, in this example,

bit 2 (a parity bit) is incorrect. The 12-bit value transmitted (after Hamming encoding) was 0xA4F. The original 8-bit data value was 0xAF.

12. 答：单个错误将引起水平和垂直奇偶检查都出错。两个错误，无论是否同行或者同列，也容易被检测到。对于有三位错误的情况，就有可能无法检测了。for example, if some bit is inverted along with its row and column parity bits. Even the corner bit will not catch this.

13. 答：用n 行k 列的矩阵来描述错误图案，在该矩阵中，正确的位用0 表示，不正确的位用1 表示。由于总共有4 位传输错误，每个可能的错误矩阵中都恰有4 个1。则错误矩阵的个数总共有C_{nk}^{4}个。而在错误矩阵中，当4 个1 正好构成一个矩形的4 个顶点的时候，这样的错误是检测不出来的。则检测不出来的错误矩阵的个数为$C_n^2 \quad C_k^2$

所以，错误不能检测的概率是：

$$\frac{C_n^2 \cdot C_k^2}{C_{nk}^4} = \frac{\frac{n(n-1)}{2} \cdot \frac{k(k-1)}{2}}{nk(nk-1)(nk-2)(nk-3)/(1 \cdot 2 \cdot 3 \cdot 4)} = \frac{6(n-1)(k-1)}{(nk-1)(nk-2)(nk-3)}$$

即；

$$\frac{\sum_{p=0}^{k-2n-2} \sum_{q=0} (k-p-1)(n-q-1)}{nk(nk-1)(nk-2)(nk-3)}$$

14. 答：如所列的除式，所得的余数为x^2+x+1。

```
      10110
1001|10100001
    1001
    ----
     1100
     1001
     ----
      1010
      1001
      ----
       111.
```

15. The frame is 10011101. The generator is 1001. The message after appending three zeros is 10011101000. The remainder on dividing 10011101000 by 1001 is 100. So, the actual bit string transmitted is 10011101100. The received bit stream with an error in the third bit from the left is 10111101100.

Dividing this by 1001 produces a remainder 100, which is different from zero.

Thus, the receiver detects the error and can ask for a retransmission.

16. 答：CRC 是在发送期间进行计算的。一旦把最后一位数据送上外出线路，就立即把CRC 编码附加在输出流的后面发出。如果把CRC 放在帧的头部，那么就要在发送之前把整个帧先检查一遍来计算CRC。这样每个字节都要处理两遍，第一遍是为了计算检验码，第二遍是为了发送。把CRC 放在尾部就可以把处理时间减半。

17. 答：当发送一帧的时间等于信道的传播延迟的2 倍时，信道的利用率为50%。或者说，当发送一帧的时间等于来回路程的传播延迟时，效率将是50%。而在帧长满足发送时间大于延迟的两倍时，效率将会高于50%。

现在发送速率为4Mb/s，发送一位需要0.25μs。

$(20\times10^{-3}\times2)\div(0.25\times10^{-6})=160000$ bit

只有在帧长不小于160kb 时，停等协议的效率才会至少达到50%。

18. 答：为了有效运行，序列空间（实际上就是发送窗口大小）必须足够的大，以允许发送方在收到第一个确认应答之前可以不断发送。信号在线路上的传播时间为

6×3000= 18000 μs，即18ms。

在T1 速率，发送64 字节的数据帧需花的时间：$64\times8\div(1.536\times10^{6})$ = 0.33μs。

所以，发送的第一帧从开始发送起，18.33ms 后完全到达接收方。确认应答又花了很少的发送时间（忽略不计）和回程的18ms。这样，加在一起的时间是36.33ms。发送方应该有足够大的窗口，从而能够连续发送36.33ms。

36. 33/0.33=110

也就是说，为充满线路管道，需要至少110 帧，因此序列号为7 位。

19. It can happen. Suppose that the sender transmits a frame and a garbled acknowledgement comes back quickly. The main loop will be executed a second time and a frame will be sent while the timer is still running.

20. Let the sender's window be (SI, Su) and the receiver's be (RI, Ru). Let the window size be W. The relations that must hold are:

$0 \leqslant S_u - S_i + 1 \leqslant W_1$

$R_u \quad R_1 \quad 1 \quad W$

$S_1 \leqslant R_1 \leqslant S_u \quad 1$

21. 答：改变检查条件后，协议将变得不正确。假定使用3 位序列号，考虑下列协议运行过程：

A 站刚发出7 号帧；B 站接收到这个帧，并发出捎带应答ack。A 站收到ack，并发送0~6 号帧。假定所有这些帧都在传输过程中丢失了。B 站超时，重发它的当前帧，此时捎带的确认号是7。考察A 站在r.rack=7 到达时的情况，关键变量是ack_expected=0，r.rack=7，next_frame_to_send_=7。修改后的检查条件将被置成“真”，不会报告已发现的丢失帧错误，而误认为丢失了的帧已被确认。另一方面，如果采用原先的检查条件，就能够报告丢失帧的错误。所以结论是：为保证协议的正确性，已接收的确认应答号应该小于下一个要发送的序列号。

22. 答：可能导致死锁。假定有一组帧正确到达，并被接收。然后，接收方会向前移动窗口。

现在假定所有的确认帧都丢失了，发送方最终会产生超时事件，并且再次发送第一帧，接收方将发送一个NAK。然后NONAK 被置成伪。假定NAK 也丢失了。那么从这个时候开始，发送方会不断发送已经被接收方接受了的帧。接收方只是忽略这些帧，但由于NONAK 为伪，所以不会再发送NAK，从而产生死锁。如果设置辅助计数器（实现“else”子句），超时后重发NAK，终究会使双方重新获得同步。

23. 答：删除这一段程序会影响协议的正确性，导致死锁。因为这一段程序负责处理接收到的确认帧，没有这一段程序，发送方会一直保持超时条件，从而使得协议的运行不能向前进展。

24. It would defeat the purpose of having NAKs, so we would have to fall back

to timeouts. Although the performance would degrade, the correctness would not be affected. The NAKs are not essential.

25. 答：这里要求 r.rack+1<next_frame_to_send。考虑下列操作细节：

A 站发送 0 号帧给 B 站。B 站收到此帧，并发送 ACK 帧，但 ACK 丢失了。A 站发生超时，重发 0 号帧。但 B 站现在期待接收 1 号帧，应此发送 NAK，否定收到的 0 号帧。显然，现在 A 站最好不重发 0 号帧。由于条件 r.rack+1<next_frame_to_send 不成立，所以用不着选择性重传 0 号帧，可以继续向前推进传送 1 号帧。这个例子就说明了这段程序中的另一个条件，即 r.rack+1<next_frame_to_send 也是重要的。

26. 答：不可以。最大接收窗口的大小就是 1。现在假定该接收窗口值变为 2。开始时发送方发送 0 至 6 号帧，所有 7 个帧都被收到，并作了确认，但确认被丢失。现在接收方准备接收 7 号和 0 号帧，当重发的 0 号帧到达接收方时，它将会被缓存保留，接收方确认 6 号帧。当 7 号帧到来的时候，接收方将把 7 号帧和缓存的 0 号帧传递给主机，导致协议错误。因此，能够安全使用的最大窗口值为 1。

27. 答：发送 1 位用时间 1μs，发送 1000bit 的最长帧花时间 1ms。由于超时间隔是 10ms，而 1s 才能产生一个新的数据帧，所以超时是不可避免的。假定 A 站向 B 站发送一个帧，正确到达接收方，但较长时间无反向交通。不久，A 站发生超时事件，导致重发已发过的一帧。

B 站发现收到的帧的序列号错误，因为该序列号小于所期待接收的序列号。因此 B 站将发送一个 NAK，该 NAK 会携带一个确认号，导致不再重发该帧。结果每个帧都被发送两次。

28. 答：不能，协议的运行将会失败。当 MaxSeq=4，序列号的模数=4+1=5，窗口大小将等于：NrBufs<=5/2=2.5，即得到，NrBufs=2。因此在该协议中，偶数序号使用缓冲区 1。这种映射意味着帧 4 和 0 将使用同一缓冲区。假定 0 至 3 号帧都正确收到了，并且都确认应答了，并且都确认应答了。如果随后的 4 号帧丢失，且下一个 0 号帧收到了，新的 0 号帧将被放到缓冲区 0 中，变量 arrived[0]被置成“真”。这样，一个失序帧将被投递给主机。事实上，采用选择性重传的滑动窗口协议需要 MaxSeq 是奇数才能正确的工作。然而其他的滑动窗口协议的实现并不具有这一性质。

29. 答：对应三种协议的窗口大小值分别是 1、7 和 4。

使用卫星信道端到端的典型传输延迟是 270ms，以 1Mb/s 发送，1000bit 长的帧的发送时间为 1ms。我们用 t=0 表示传输开始的时间，那么在 t=1ms 时，第一帧发送完毕；t=271ms 时，第一帧完全到达接收方；t=272ms，对第一帧的确认帧发送完毕；t=542ms，带有确认的帧完全到达发送方。因此一个发送周期为 542ms。如果在 542ms 内可以发送 k 个帧，由于每一个帧的发送时间为 1ms，则信道利用率为 k/542，因此：

（a） k=1，最大信道利用率=1/542=0.18%

（b） k=7，最大信道利用率=7/542=1.29%

（c） k=4，最大信道利用率=4/542=0.74%

30. 答：使用选择性重传滑动窗口协议，序列号长度是 8 位。窗口大小为 128。卫星信道端到端的传输延迟是 270ms。以 50kb/s 发送，4000bit（3960+40）长的数据帧的发送时间是 0.02*4000=80ms。我们用 t=0 表示传输开始时间，那么，t=80ms，第一帧发送完毕；

t=270+80=350ms，第一帧完全到达接收方；t=350+80=430ms，对第一帧作捎带确认的反向数据帧可能发送完毕；t=430+270=700ms，带有确认的反向数据帧完全到达发送方。因此，

周期为 700ms，发送 128 帧时间 80*128=10240ms，这意味着传输管道总是充满的。每个帧重传的概率为 0.01，对于 3960 个数据位，头开销为 40 位，平均重传的位数为 4000*0.01=40 位，传送 NAK 的平均位数为 40*1/100=0.40 位，所以每 3960 个数据位的总开销为 80.4 位。

因此，开销所占的带宽比例等于 80.4/(3960+80.4)=1.99%。

31. 答：使用卫星信道端到端的传输延迟为 270ms，以 64kb/s 发送，周期等于 604ms。发送一帧的时间为 64ms，我们需要 604/64=9 个帧才能保持通道不空。

对于窗口值 1，每 604ms 发送 4096 位，吞吐率为 4096/0.604=6.8kb/s。

对于窗口值 7，每 604ms 发送 4096*7 位，吞吐率为 4096*7/0.604=47.5kb/s。

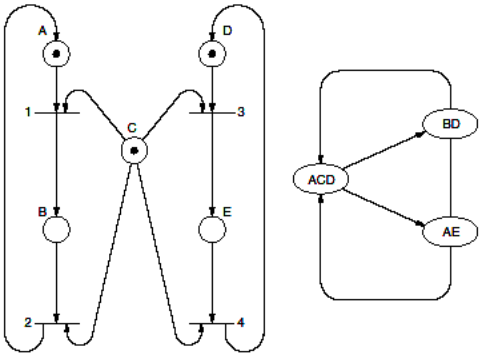
对于窗口值超过 9（包括 15、127），吞吐率达到最大值，即 64kb/s。

32. 答：在该电缆中的传播速度是每秒钟 200 000km，即每毫秒 200km，因此 100km 的电缆将会在 0.5ms 内填满。T1 速率 125μs传送一个 193 位的帧，0.5ms 可以传送 4 个 T1 帧，即 193*4=772bit。

33. Each machine has two key variables: *next*3*frame*3*to*3*send* and *frame*3*expected*, each of which can take on the values 0 or 1. Thus, each machine can be in one of four possible states. A message on the channel contains the sequence number of the frame being sent and the sequence number of the frame being ACKed. Thus, four types of messages exist. The channel may contain 0 or 1 message in either direction. So, the number of states the channel can be in is 1 with zero messages on it, 8 with one message on it, and 16 with two messages on it (one message in each direction). In total there are 1 + 8 + 16 = 25 possible channel states. This implies 4 4 25 400 possible states for the complete system.

34. The firing sequence is 10, 6, 2, 8. It corresponds to acceptance of an even frame, loss of the acknowledgement, timeout by the sender, and regeneration of the acknowledgement by the receiver.

35. The Petri net and state graph are as follows:



The system modeled is mutual exclusion. *B* and *E* are critical sections that may not be active simultaneously, i.e., state *BE* is not permitted. Place *C* represents a semaphore that can be seized by either *A* or *D* but not by both together.

36. PPP was clearly designed to be implemented in software, not in hardware as HDLC nearly always is. With a software implementation, working entirely with bytes

is much simpler than working with individual bits. In addition, PPP was designed to be used with modems, and modems accept and transmit data in units of 1 byte, not 1 bit.

37. At its smallest, each frame has two flag bytes, one protocol byte, and two checksum bytes, for a total of five overhead bytes per frame.

第 4 章 介质访问子层

1. The formula is the standard formula for Markov queueing given in section 4.1.1, namely, $T = 1/(\mu C - \lambda)$. Here $C = 10^8$ and $\mu = 10^{-4}$, so $T = 1/(10000 - lambda)$ sec. For the three arrival rates, we get (a) 0.1 msec, (b) 0.11 msec, (c) 1 msec. For case (c) we are operating a queueing system with $\rho = \lambda/\mu C = 0.9$, which gives the 10×delay.

2. 答：对于纯的 ALOHA，可用的带宽是 0.184×56 Kb/s =10.304 *Kb/ s*。

每个站需要的带宽为 1000/100=10b/s。而 N=10304/10≈1030

所以，最多可以有 1030 个站，即 N 的最大值为 1030。

3. 答：对于纯的 ALOHA，发送可以立即开始。对于分隙的 ALOHA，它必须等待下一个时隙。这样，平均会引入半个时隙的延迟。因此，纯 ALOHA 的延迟比较小。

4. 每个终端每 200（=3600/18）秒做一次请求，总共有 10 000 个终端，因此，总的负载是 200 秒做 10000 次请求。平均每秒钟 50 次请求。每秒钟 8000 个时隙，所以平均每个时隙的发送次数为 50/8000=1/160。

5. 答：

（a）在任一帧时间内生成 k 帧的概率服从泊松分布

$$\Pr[k] = \frac{G^k e^{-G}}{k!}$$

生成 0 帧的概率为 e^{-G}

对于纯的ALOHA，发送一帧的冲突危险区为两个帧时，在两帧内无其他帧发送的概率是 $e^{-G} \times e^{-G} = e^{-2G}$

对于分隙的ALOHA，由于冲突危险区减少为原来的一半，任一帧时内无其他帧发送的概率是 e^{-G} 。

现在时隙长度为 40ms，即每秒 25 个时隙，产生 50 次请求，所以每个时隙产生两个请求，G=2。因此，首次尝试的成功率是：$e^{-2} = 1/ e^2$

（b） $(1-e^{-G})^k e^{-G} = (1-e^2)^k e^{-2} = 0.135 \times (1-0.135)^k = 0.135 \times 0.865^k$

（c）尝试 k 次才能发送成功的概率（即前 k-1 次冲突，第 k 次才成功）为：

$$p_k = e^{-G}(1-e^{-G})^{k-1}$$

那么每帧传送次数的数学期望为

$$E = \sum_{k=1}^{\infty} k p_k = \sum_{k=1}^{\infty} k e^{-G}(1-e^{-G})^{k-1} = e^G = e^2 = 7.4$$

6. 答：

（a）从泊松定律得到 $p_0 = e^{-G}$，因此 $G = -\ln p_0 = -\ln 0.1 = 2.3$

（b） $S = G e^{-G}$， G =2.3，e^{-G}=0.1

S=2.3×0.1=0.23

（c）因为每当 G>1 时，信道总是过载的，因此在这里信道是过载的。

7. 答：每帧传送次数的数学期望为：

$$E = \sum_{k=1}^{\infty} kp_k = \sum_{k=1}^{\infty} ke^{-G}(1-e^{-G})^{k-1} = e^{G}$$

E 个事件为E-1 个长度等于 4 个时隙的间隔时间所分隔。因此一个帧从第一次发送开始时间到最后一次尝试成功的发送开始时间之间的长度即延迟是 $4(e^{G}-1)$，吞吐率 $S=Ge^{-G}$。

对于每一个G 值，都可以计算出对应的延迟值 $D=4(e^{G}-1)$，以及吞吐率值 $S=Ge^{-G}$ 。

按此方法即可画出时延对吞吐率的曲线。

8. (a) The worst case is: all stations want to send and *s* is the lowest numbered station. Wait time *N* bit contention period + (N-1) *d* bit for transmission of frames. The total is N+(N-1) *d* bit times. (b) The worst case is: all stations have frames to transmit and *s* has the lowest virtual station number.

Consequently, *s* will get its turn to transmit after the other N-1 stations have transmitted one frame each, and *N* contention periods of size log2 *N* each.

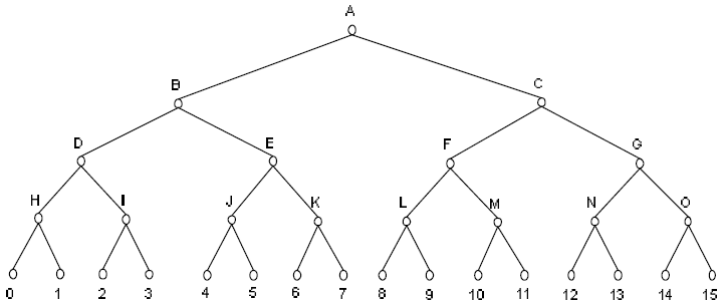
Wait time is thus (N+1)× *d*+*N*×log2 bits.

9. 答：在解答这一问题之前，首先要了解什么是 Mok 和 Ward 版本的二进制倒计数法。在二进制倒计数法中，每个想要使用信道的站点首先将其地址以二进制位串的形式按照由高到低的顺序进行广播，并且假定所有地址的长度相同。为了避免冲突，必须进行仲裁：如果某站发现其地址中原本为 0 的高位被置换为 1，那么它便放弃发送。对于次高位进行同样的信道竞争操作，直到最后只有一个站赢得信道为止。一个站点在赢得信道竞争后便可发送一帧，然后另一个信道竞争周期又将开始。Mok 和 Ward 提出了二进制倒计数法的一个变种。该方法采用了并行接口而不是串行接口：还使用虚拟站号，在每次传输之后对站重新编号，从 0 开始，已成功传送的站被排在最后。如果总共有 N 个站，那么最大的虚拟站号是 N-1。

在本题中，当 4 站发送时，它的号码变为 0，而 0、1、2 和 3 号站的号码都增 1，10 个站点的虚站号变为 8，3，0，5，2，7，4，6，9，1 当 3 站发送时，它的号码变为 0，而 0、1 和 2 站的号码都增 1，10 个站点的虚站号变为：8，0，1，5，3，7，4，6，9，2

最后，当 9 站发送时，它变成 0，所有其他站都增 1，结果是：9，1，2，6，4，8，5，7，0，3。

10. 答：在自适应树遍历协议中，可以把站点组织成二叉树（见图）的形式。在一次成功的传输之后，在第一个竞争时隙中，全部站都可以试图获得信道，如果仅其中之一需用信道，则发送冲突，则第二时隙内只有那些位于节点 B 以下的站（0 到 7）可以参加竞争。如其中之一获得信道，本帧后的时隙留给站点 C 以下的站；如果 B 点下面有两个或更多的站希望发送，在第二时隙内会发生冲突，于是第三时隙内由 D 节点以下各站来竞争信道。



本题中，站 2、3、5、7、11 和 13 要发送，需要 13 个时隙，每个时隙内参加竞争的站的列表如下：

第一时隙：2、3、5、7、11、13

第二时隙：2、3、5、7

第三时隙：2、3

第四时隙：空闲

第五时隙：2、3

第六时隙：2

第七时隙：3

第八时隙：5、7

第九时隙：5

第十时隙：7

第十一时隙：11、13

第十二时隙：11

第十三时隙：13

11. 答： 2^n个站点对应n+1 级，其中 0 级有 1 个节点，1 级有 2 个节点， n 级有 2^n 个节点。在i 级的每个节点下面所包括的站的个数等于总站数的 $1/2^i$。本题中所需要的时隙数取决于为了到达准备好发送的两个站的共同先辈点必须往回走多少级。先计算这两个站具有共同的父节点的概率p_1。在 2^n个站中，要发送的两个站共享一个指定的父节点的概率是

$$\frac{1}{C_{2^n}^2}=\frac{1}{2^{n-1}(2^n-1)}$$

总共 2^{n-1}个父节点，所以，

$$p_1=\frac{1}{2^{n-1}(2^n-1)}\cdot 2^{n-1}=\frac{1}{2^n-1}$$

因为 $2^n >> 1$

所以$p_1 \approx 2^{-n}$

在共享父节点的条件下遍历树，从第二级开始每一级访问两个节点，这样遍历树所走过的节点总数$n_1 = 1+ 2+\cdots+2+2=1=2n$，接下来，我们考察两个发送站共享祖父节点的概率$p_2$和遍历树所走过的节点总数$n_2$。此时在每个父节点下面仅可能有一个站发送。两个发送站共享一个指定的祖父节点的概率是 $1/ C_2^{2^{n-1}}$。

共有 2^{n-2}个祖父节点

$$p_2 = \frac{2^{n-2}}{C_{2^{n-1}}^2} = \frac{1}{2^{n-1}-1} \approx \frac{1}{2^{n-1}} = 2^{-n+1}$$

遍历树比 1 n 减少两个节点，即

$$n_2 = 1 + 2n - 2 = 2n - 1$$

通过类似的分析和计算，可以得到，两个发送站共享曾祖父节点（属n-3 级祖先节点）的概率是 $p3= 2^{-n+2}$

遍历树所经过的节点总数比n_2又少两个节点，

$$n_3 = 2n - 1 - 2 = 2n - 3$$

$$\vdots$$

$$p_{i+1} = 2^{-(n-i)}$$

$$n_{i+1} = 2n + 1 - 2i$$

因此，最坏的情形是 2n+1 个时隙（共享父节点），对应于 i=0；

最好的情形是 3 个时隙，对应于 i=n-1 （两个发送站分别位于左半树和右半树），所以平均时隙数等于

$$m = \sum_{i=0}^{n-1} 2^{-(n-i)}(2n + 1 - 2i)$$

该表达式可以简化为

$$m = (1 - 2^{-n})(2n + 1) - 2^{-(n-1)} \sum_{i=0}^{n-1} i2^i$$

12. 答：如果所有站的发射有效范围都很大，以至于任一站都可以收到所有其他站发送的信号，那么任一站都可以与其他站以广播方式通信。在这样的条件下，CSMA/CD 可以工作的很好。

13. 答：WDMA（wave length division multiple access）是一个波分多路访问协议。每个站点分配 2 个信道；其中窄信道是控制信道，接收其他站发给该站的控制信号；宽信道用作该站点输出数据帧的信道。每个信道被划分成许多个时隙组。时隙 0 用某种特殊的方式标记，以便于后继时隙的识别。所有的信道均用同一个全局时钟来同步。每个站点都有 2 个发送端和 2 个接收端，它们分别是：

（1）一个波长固定不变的接收端，它用来侦听本站点的控制信道。

（2）一个波长可调的发送端，它用于向其他站点的控制信道发送帧。

（3）一个波长固定不变的发送端，它用于输出数据帧。

（4）一个波长可调的接收端，它用来选择要侦听的数据发送端。

也就是说，每个站点都侦听自己的控制信道，看是否有请求产生，并将接收端的波长调为发送端的波长，从而得到数据。

GSM（Global system for mobile communication）是一种数字蜂窝无线电系统信道分配方案。

系统中每个蜂窝最多可拥有 200 多个全双工信道，每个信道包括下行链路频率（从基站到可移动站）和上行链路频率（从可移动站到基站），每个频段宽 200kHz。每一个信道均可采用时分复用技术，支持多个独立的连接。

两种协议都使用 FDM 和 TDM 结合的方法，它们都可以提供专用的频道（波长），并且都划分时隙，实现 TDM。

14. Yes. Imagine that they are in a straight line and that each station can reach only its nearest neighbors. Then *A* can send to *B* while *E* is sending to *F*.

15. (a) Number the floors 1-7. In the star configuration, the router is in the middle of floor 4. Cables are needed to each of the 7 15 1 104 sites. The total length of these cables is

$$4\sum_{i=1}^{7}\sum_{j=1}^{15}\sqrt{(i-4)^2+(j-8)^2}$$

The total length is about 1832 meters.

(b) For 802.3, 7 horizontal cables 56 m long are needed, plus one vertical cable 24 m long, for a total of 416 m.

16. 答：以太网使用曼彻斯特编码，这就意味着发送的每一位都有两个信号周期。标准以太网的数据率为 10Mb/s，因此波特率是数据率的两倍，即 20MBaud。

17. The signal is a square wave with two values, high (H) and low (L). The pattern is LHLHLHHLHLHLLHHLLHHL.

18. The pattern this time is HLHLHLLHHLLHLHHLHLLH.

19. The round-trip propagation time of the cable is 10 sec. A complete transmission has six phases:

transmitter seizes cable (10 sec)

transmit data (25.6 sec)

Delay for last bit to get to the end (5.0 sec)

receiver seizes cable (10 sec)

acknowledgement sent (3.2 sec)

Delay for last bit to get to the end (5.0 sec)

The sum of these is 58.8 sec. In this period, 224 data bits are sent, for a rate

of about 3.8 Mbps.

20. 答：把获得通道的尝试从 1 开始编号。第i 次尝试分布在 2^{i-1} 个时隙中。因此，i 次尝试碰撞的概率是 $2^{-(i-1)}$，开头k-1 次尝试失败，紧接着第k 次尝试成功的概率是：

$$p_k = (1-2^{-(k-1)})[2^{-0}\cdot 2^{-1}\cdot 2^{-2}\cdots 2^{-(k-2)}] = (1-2^{-(k-2)})2^{-(k-1)(k-2)/2}$$

即：

$$P_k = (1-2^{-(k-1)})\prod_{i=1}^{k-1}2^{-(i-1)}$$

上式可简化为：

$$P_k = (1-2^{-(k-1)})\,2^{-(k-1)(k-2)/2}$$

所以每个竞争周期的平均竞争次数是Σ kp_k

21. 答：对于 1km 电缆，单程传播时间为 1/200000 =5×10^{-6} s，即 5μs，来回路程传播时间为 2t =10μs。为了能够按照CSMA/CD 工作，最小帧的发射时间不能小于 10μs。以 1Gb/s 速率工作，10μs可以发送的比特数等于：

$$\frac{10\times10^{-6}}{1\times10^{-9}}=10000$$

因此，最小帧是 10 000 bit 或 1250 字节长。

22. The minimum Ethernet frame is 64 bytes, including both addresses in the Ethernet frame header, the type/length field, and the checksum. Since the header fields occupy 18 bytes and the packet is 60 bytes, the total frame size is 78 bytes, which exceeds the 64-byte minimum. Therefore, no padding is used.

23. The maximum wire length in fast Ethernet is 1/10 as long as in Ethernet.

24. The payload is 1500 bytes, but when the destination address, source address, type/length, and checksum fields are counted too, the total is indeed 1518.

25. The encoding is only 80% efficient. It takes 10 bits of transmitted data to represent 8 bits of actual data. In one second, 1250 megabits are transmitted, which means 125 million codewords. Each codeword represents 8 data bits, so the true data rate is indeed 1000 megabits/sec.

26. The smallest Ethernet frame is 512 bits, so at 1 Gbps we get 1,953,125 or almost 2 million frames/sec. However, this only works when frame bursting is operating. Without frame bursting, short frames are padded to 4096 bits, in which case the maximum number is 244,140. For the largest frame (12,144 bits), there can be as many as 82,345 frames/sec.

27. Gigabit Ethernet has it and so does 802.16. It is useful for bandwidth efficiency (one preamble, etc.) but also when there is a lower limit on frame size.

28. Station *C* is the closest to *A* since it heard the RTS and responded to it by asserting its NAV signal. *D* did not respond so it must be outside *A*'s radio range.

29. A frame contains 512 bits. The bit error rate is *p* 10 7. The probability of all 512 of them surviving correctly is (1 *p*)512, which is about 0.9999488. The fraction damaged is thus about 5 10 5. The number of frames/sec is 11 106 /512 or about 21,484. Multiplying these two numbers together, we get about 1 damaged frame per second.

30. It depends how far away the subscriber is. If the subscriber is close in, QAM-64 is used for 120 Mbps. For medium distances, QAM-16 is used for 80 Mbps. For distant stations, QPSK is used for 40 Mbps.

31. Uncompressed video has a constant bit rate. Each frame has the same number of pixels as the previous frame. Thus, it is possible to compute very accurately how much bandwidth will be needed and when. Consequently, constant bit rate service is the best choice.

32. One reason is the need for real-time quality of service. If an error is discovered, there is no time to get a retransmission. The show must go on. Forward error correction can be used here. Another reason is that on very low quality lines (e.g., wireless channels), the error rate can be so high that practically all frames would have to be retransmitted, and the retransmission would probably damaged as well. To avoid this, forward error correction is used to increase the fraction of frames that arrive correctly.

33. It is impossible for a device to be master in two piconets at the same time. There are two problems. First, only 3 address bits are available in the header while as many as seven slaves could be in each piconet. Thus, there would be no way to uniquely address each slave. Second, the access code at the start of the frame is derived from the master's identity. This is how slaves tell which message belongs to which piconet. If two overlapping piconets used the same access code, there would be no way to tell which frame belonged to which piconet. In effect, the two piconets would be merged into one big piconet instead of two separate ones.

34. Bluetooth uses FHSS, just as 802.11 does. The biggest difference is that Bluetooth hops at a rate of 1600 hops/sec, far faster than 802.11.

35. An ACL channel is asynchronous, with frames arriving irregularly as data are produced. An SCO channel is synchronous, with frames arriving periodically at a well-defined rate.

36. They do not. The dwell time in 802.11 is not standardized, so it has to be announced to new stations that arrive. In Bluetooth this is always 625 sec.

There is no need to announce this. All Bluetooth devices have this hardwired into the chip. Bluetooth was designed to be cheap, and fixing the hop rate and dwell time leads to a simpler chip.

37. The first frame will be forwarded by every bridge. After this transmission, each bridge will have an entry for destination *a* with appropriate port in its hash table. For example, *D*'s hash table will now have an entry to forward frames destined to *a* on LAN 2. The second message will be seen by bridges *B*, *D*, and *A*. These bridges will append a new entry in their hash table for frames destined for *c*. For example bridge *D*'s hash table will now have another entry to forward frames destined to *c* on LAN 2. The third message will be seen by bridges *H*, *D*, *A*, and *B*. These bridges will append a new entry in their hash table for frames destined for *d*. The fifth message will be seen by bridges *E*, *C*, *B*, *D*, and *A*. Bridges *E* and *C* will append a new entry in their hash table for frames destined for *d*, while bridges *D*, *B*, and *A* will update their hash table entry for destination *d*.

38. Bridges *G*, *I* and *J* are not used for forwarding any frames. The main reason for having loops in an extended LAN is to increase reliability. If any bridge in the current spanning tree fails, the (dynamic) spanning tree algorithm reconfigures the

spanning tree into a new one that may include one or more of these bridges that were not a part of the previous spanning tree.

39. The simplest choice is to do nothing special. Every incoming frame is put onto the backplane and sent to the destination card, which might be the source card. In this case, intracard traffic goes over the switch backplane. The other choice is to recognize this case and treat it specially, sending the frame out directly and not going over the backplane.

40. The worst case is an endless stream of 64-byte (512-bit) frames. If the backplane can handle 109 bps, the number of frames it can handle is 109 ⁄512. This is 1,953,125 frames/sec.

41. The port on *B1* to LAN 3 would need to be relabeled as GW.

42. A store-and-forward switch stores each incoming frame in its entirety, then examines it and forwards it. A cut-through switch starts to forward incoming frames before they have arrived completely. As soon as the destination address is in, the forwarding can begin.

43. Store-and-forward switches store entire frames before forwarding them. After a frame comes in, the checksum can be verified. If the frame is damaged, it is discarded immediately. With cut=through, damaged frames cannot be discarded by the switch because by the time the error is detected, the frame is already gone. Trying to deal with the problem is like locking the barn door after the horse has escaped.

44. No. Hubs just connect all the incoming lines together electrically. There is nothing to configure. No routing is done in a hub. Every frame coming into the hub goes out on all the other lines.

45. It would work. Frames entering the core domain would all be legacy frames, so it would be up to the first core switch to tag them. It could do this by using MAC addresses or IP addresses. Similarly, on the way out, that switch would have to untag outgoing frames.

第 5 章 网络层

1. **答：**文件传送、远程登录和视频点播需要面向连接的服务。另一方面，信用卡验证和其他的销售点终端、电子资金转移，以及许多形式的远程数据库访问生来具有无连接的性质，在一个方向上传送查询，在另一个方向上返回应答。

2. **答：**有。中断信号应该跳过在它前面的数据，进行不遵从顺序的投递。典型的例子是当一个终端用户键入退出（或 kill）健时。由退出信号产生的分组应该立即发送，并且应该跳过当前队列中排在前面等待程序处理的任何数据（即已经键入但尚未被程序读取的数据）。

3. **答：**不对。为了从任意源到任意目的地，为连接建立的分组选择路由，虚电路网络肯定需要这一能力。

4. **答：**在连接建立的时候可能要协商窗口的大小、最大分组尺寸和超时值。

5. **答：**虚电路实现需要在 1000 秒内固定分配 5*8=40 字节的存储器。数据报实现需要比虚电路实现多传送的头信息的容量等于(15-3 ） ×4×200＝9600 字节-跳段。现在的问题就变成了 40000 字节-秒的存储器对比 9600 字节-跳段的电路容量。如果存储器的使用期为两年，即 3600×8×5×52×2= 1.7×10^{7}秒，一个字节-秒的代价为 1/(1.5×10^{7}) = 6.7×10^{-8} 分，那么 40000 字节-秒的代价为 2.7 毫分。另一方面，1 个字节-跳段代价是 10^{-6} 分，9600 个字节-跳段的代价为 10^{-6} × 9600=9.6×10^{-3}分，即 9.6 毫分，即在这 1000 秒内的时间内便宜大约 6.9 毫分。

6. **答：**有可能。大的突发噪声可能破坏分组。使用k 位的检验和，差错仍然有 2^{-k}的概率被漏检。如果分组的目的地段或虚电路号码被改变，分组将会被投递到错误的目的地，并可能被接收为正确的分组。换句话说，偶然的突发噪声可能把送往一个目的地的完全合法的分组改变成送往另一个目的地的也是完全合法的分组。

7. It will follow all of the following routes: ABCD, ABCF, ABEF, ABEG, AGHD, AGHF, AGEB, and AGEF. The number of hops used is 24.

8. **答：**使用最短通路搜索算法选择一条路径，然后，删除刚找到的路径中的使用的所有的弧（对应各条链路）。接着，再运行一次最短通路搜索算法。这个第 2 条路径在第 1 条路径中有线路失效的情况下，可以作为替代路径启用；反之亦然。

9. **答：**通过B 给出（11，6，14，18，12，8）

通过D 给出（19，15，9，3，12，13）

通过E 给出（12，11，8，14，5，9）

取到达每一目的地的最小值（C 除外）得到：（11，6，0，3，5，8）

输出线路是：（B，B，-，D，E，B）

10. **答：**路由表的长度等于 8*50=400bit。该表每秒钟在每条线路上发送 2 次，因此 400*2=800b/s，即在每条线路的每个方向上消耗的带宽都是 800 bps。

11. **答：**这个结论总是成立的。如果一个分组从某条线路上到达，必须确认包的到达。如果线路上没有分组到达，它就是在发送确认。情况 00 (没有分组到达并且不发送确认)和 11 (到达和返回)逻辑上错误，因此不存在。

12. 所谓分级路由，就是将路由器按区（REGION）进行划分，每个路由器只须知道在自己的区内如何为分组选择路由到达目的地的细节，而不用知道其他区的内部结构。对于大的网络，也许两级结构是不够的，还可以把区组合成簇（CLUSTER），把簇再组合成域（ZONE），……对于等级式路由，在路由表中对应所有的本地路由器都有一个登录项，所有其他的区（本簇内）、簇（本域内）和域都缩减为单个路由器，因此减少了路由表的尺寸。

在本题中，4800=15*16*20。当选择 15 个簇、16 个区，每个区 20 个路由器时（或等效形式，例如 20 个簇、16 个区，每个区 15 个路由器），路由表尺寸最小，此时的路由表尺寸为 15+16+20=51。

The minimum occurs at 15 clusters, each with 16 regions, each region having 20 routers, or one of the equivalent forms, e.g., 20 clusters of 16 regions of 15 routers. In all cases the table size is 15 + 16 + 20 = 51.

13. Conceivably it might go into promiscuous mode, reading all frames dropped onto the LAN, but this is very inefficient. Instead, what is normally done is that the home agent tricks the router into thinking it is the mobile host by responding to ARP requests. When the router gets an IP packet destined for the mobile host, it broadcasts an ARP query asking for the 802.3 MAC-level address of the machine with that IP address. When the mobile host is not around, the home agent responds to the ARP, so the router associates the mobile user's IP address with the home agent's 802.3 MAC-level address.

14. **答：**在一个子网中，从所有的源到一个指定的目的地的最佳路由的集合形成一棵以该目的地为根的树。这样的树就称作汇集树。汇集树不必是唯一的，其他具有相同通路长度的树可能存在。所有路由选择算法的目标都是要为所有的路由器寻找和使用汇集树。在广播形式的应用中，源主机需要向所有其他的主机发送报文。在称为反向通路转发的广播路由选择中，当广播分组到达路由器时，路由器对此分组进行检查，查看该分组是否来自于通常用于发送分组到广播源的线路，如果是，则此广播分组本身非常有可能是从源路由器来的第一个拷贝。

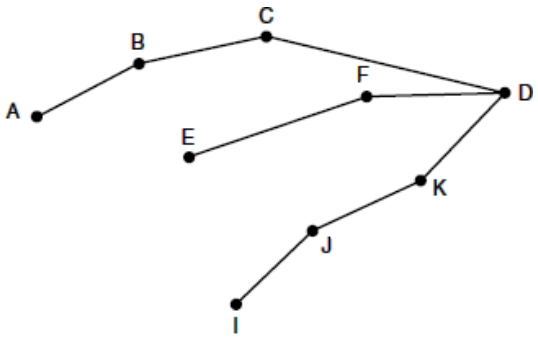
在这种情况下，路由器将此分组复制转发到进入线路以外的所有线路。然而，如果广播分组到来的线路不是到达源端的线路，那么分组就被当作副本而扔掉。

（1）反向通路转发算法，算法进行到 5 个跳段后结束，总共产生 28 个分组。

（2）使用汇集树算法，需要 4 个跳段，总共产生 14 个分组。

15. Node F currently has two descendants, A and D. It now acquires a third one, G, not circled because the packet that follows IFG is not on the sink tree. Node G acquires a second descendant, in addition to D, labeled F. This, too, is not circled as it does not come in on the sink tree.

16. Multiple spanning trees are possible. One of them is:



17. When H gets the packet, it broadcasts it. However, I knows how to get to I, so it does not broadcast.

18. Node H is three hops from B, so it takes three rounds to find the route.

19. It can do it approximately, but not exactly. Suppose that there are 1024 node identifiers. If node 300 is looking for node 800, it is probably better to go clockwise, but it could happen that there are 20 actual nodes between 300 and 800 going clockwise and only 16 actual nodes between them going counterclockwise.

The purpose of the cryptographic hashing function SHA-1 is to produce a very smooth distribution so that the node density is about the same all along the circle. But there will always be statistical fluctuations, so the straightforward choice may be wrong.

20. The node in entry 3 switches from 12 to 10.

21. **答：**对时间以 T 秒为单位分时隙。在时隙中，源路由器发送第一个分组。在时隙 2 的开始，第 2 个路由器收到了分组，但不能应答。在时隙 3 的开始，第 3 个路由器收到了分组，但也不能应答。这样，此后所有的路由器都不会应答。仅当目的地主机从目的地路由器取得分组时才会发送第 1 个应答。现在确认应答开始往回传播。在源路由器可以发送第 2 个分组之前，需要两次穿行该子网，需要花费的时间等于 2(n-1)T 秒/分组，显然，这种协议的效率是很低的。

22. **答：**（1）由源主机发送的每个分组可能行走 1 个跳段、2 个跳段或 3 个跳段。走 1 个跳段的概率为p，走 2 个跳段的概率为（1- p）p ，走 3 个跳段的概率为（1- p）^{2} p。那么，一个分组平均通路长度的期望值为：

$$L = 1\cdot p + 2\cdot(1-p)p + 3\cdot(1-p)^2 = p^2 - 3p + 3$$

即每次发送一个分组的平均跳段数是 p^2-3 p +3。

（2）一次发送成功（走完整个通路）的概率为(1- p)^2，令a = (1- p)^2，两次发射成功的概率等于 (1- a) a，三次发射成功的概率等于 (1- a)^2 a ，…，因此一个分组平均发送次数为：

$$T = \sum_{n=1}^{\infty} n\alpha(1-\alpha)^{n-1} = \frac{1}{\alpha} = \frac{1}{(1-p)^2}$$

即一个分组平均要发送 1/(1- p)^{2}次。

（3）最后，每一个接收到的分组行走的平均跳段数等于

$$H = L \times T = (p^2 - 3p + 3)/(1-p)^2$$

23. First, the warning bit method explicitly sends a congestion notification to the source by setting a bit, whereas RED implicitly notifies the source by simply dropping one of its packets. Second, the warning bit method drops a packet only when there is no buffer space left, whereas RED drops packets before all the buffer are exhausted.

24. **答：**通常计算机能够以很高的速率产生数据，网络也可以用同样的速率运行。然而，路由器却只能在短时间内以同样高的速率处理数据。对于排在队列中的一个分组，不管它有多大，路由器必须做大约相同分量的工作。显然，处理 10 个 100 字节长的分组所作的工作比处理 1 个 1000 字节长的分组要做的工作多得多。

25. **答：**不可以发送任何大于 1024 字节的分组。

26. **答：**每 5μs产生一个令牌，1 秒中可以发送 200 000 个信元。每个信元含有 48 个数据字节，即 8×48= 384bit。

$384 \times 2 \times 10^5 = 76.8 \times 10^6$b/s

所以，最大的可持续的净数据速率为 76.8Mb/s。

27. **答：**本题乍看起来，似乎以 6Mb/s 速率发送用 4/3 秒的时间可以发送完桶内 8Mb 的数据，使漏桶变空。然而，这样回答是错误的，因为在这期间，已有更多的令牌到达。正确的答案应该使用公式 $S = C/(M-P)$，这里的 S 表示以秒计量的突发时间长度，M 表示以每秒字节计量的最大输出速率，C 表示以字节计的桶的容量，P 表示以每秒字节计量的令牌到达速率。则：

$$S = \frac{(8 \times 10^6)/8}{(6 \times 10^6)/8 - (1 \times 10^6)/8} = 1.6\,\text{s}$$

因此，计算机可以用完全速率 6Mb/s 发送 1.6 s 的时间。

28. **答：**令最大突发时间长度为? t 秒，在极端情况下，漏桶在突发期间的开始是充满的（1MB），在突发期间另有 10? t MB 进入桶内。在传输突发期间的输出包含 50? t MB。由等式 1+10? t=50? t，得到? t=1/40s，即 25ms。因此，以最大速率突发传送可维持 25ms 的时间。

29. The bandwidths in MB/sec are as follows: A: 2, B: 0, C: 1, E: 3, H: 3, J: 3, K:2, and L: 1.

30. Here μ is 2 million and λ is 1.5 million, so $\rho = \lambda/\mu$ is 0.75, and from queueing theory, each packet experiences a delay four times what it would in an idle system. The time in an idle system is 500 nsec, here it is 2μsec. With 10 routers along a path, the queueing plus service time is 20 μsec.

31. There is no guarantee. If too many packets are expedited, their channel may have even worse performance than the regular channel.

32. **答：**在这两种情况下都需要分割功能。即使在一个串接的虚电路网络中，沿通路的某些网络可能接受 1024 字节分组，而另一些网络可能仅接受 48 字节分组，分割功能仍然是需要的。

33. **答：**可以。只需把分组封装在属于所经过的子网的数据报的载荷段中，并进行发送。

34. The initial IP datagram will be fragmented into two IP datagrams at I1. No other fragmentation will occur.

Link A-R1:

Length = 940; ID = x; DF = 0; MF = 0; Offset = 0

Link R1-R2:

(1) Length = 500; ID = x; DF = 0; MF = 1; Offset = 0

(2) Length = 460; ID = x; DF = 0; MF = 0; Offset = 60

Link R2-B:

(1) Length = 500; ID = x; DF = 0; MF = 1; Offset = 0

(2) Length = 460; ID = x; DF = 0; MF = 0; Offset = 60

35. If the bit rate of the line is b, the number of packets/sec that the router can emit is b/8192, so the number of seconds it takes to emit a packet is 8192/b.

To put out 65,536 packets takes 229 /b sec. Equating this to the maximum packet lifetime, we get 229 /b 10. Then, b is about 53,687,091 bps.

36. **答：**因为为每一个分割的片段选择路由都需要该选项信息，因此该选项必须出现在每一个片段中。

37. **答：**除去 2 位作为前缀，将剩下 18 位表示网络。概念上，网络数目可以有 18^2 或 262144 个。然而，全 0 和全 1 是特别地址，所以只有 262142 个可供分配。

38. **答：**The address is 194.47.21.130.

39. **答：**对于一个 B 类网络，高端 16 位形成网络号，低端 16 位是子网或主机域。在子网掩码的低端 16 位中，最高有效 4 位为 1111，因此剩下 12 位用于主机号。因此，存在 4096 个主机地址。但由于全 0 和全 1 是特别地址，因此最大的主机数目为 4094。

40. To start with, all the requests are rounded up to a power of two. The starting address, ending address, and mask are as follows: A: 198.16.0.0 –198.16.15.255 written as 198.16.0.0/20

B: 198.16.16.0 – 198.23.15.255 written as 198.16.16.0/21

C: 198.16.32.0 – 198.47.15.255 written as 198.16.32.0/20

D: 198.16.64.0 – 198.95.15.255 written as 198.16.64.0/19

41. They can be aggregated to 57.6.96/19.

42. It is sufficient to add one new table entry: 29.18.0.0/22 for the new block. If an incoming packet matches both 29.18.0.0/17 and 29.18.0.0./22, the longest one wins. This rule makes it possible to assign a large block to one outgoing line but make an exception for one or more small blocks within its range.

43. The packets are routed as follows:

(a) Interface 1

(b) Interface 0

(c) Router 2

(d) Router 1

(e) Router 2

44. After NAT is installed, it is crucial that all the packets pertaining to a single connection pass in and out of the company via the same router, since that is where the mapping is kept. If each router has its own IP address and all traffic belonging to a given connection can be sent to the same router, the mapping can be done correctly and multihoming with NAT can be made to work.

45. **答：**不对。ARP 不是向网络层提供服务，它本身就是网络层的一部分，帮助向传输层提供服务。在数据链路层不存在 IP 地址的问题。数据链路层协议是像 HDLC 和 PPP 这样的协议，它们把比特串从线路的一端传送到另一端。

46. **答：**在 RARP 的实现中有一个 RARP 服务器负责回答查询请求。在 ARP 的实现中没有这样的服务器，主机自己回答 ARP 查询。

47. In the general case, the problem is nontrivial. Fragments may arrive out of order and some may be missing. On a retransmission, the datagram may be fragmented in different-sized chunks. Furthermore, the total size is not known until the last fragment arrives. Probably the only way to handle reassembly is to buffer all the pieces until the last fragment arrives and the size is known. Then build a buffer of the right size, and put the fragments into the buffer, maintaining a bit map with 1 bit per 8 bytes to keep track of which bytes are present in the buffer. When all the bits in the bit map are 1, the datagram is complete.

48. **答：**对接收方而言，这是一个新的 IP 数据报的一部分，该数据报的其他部分还不得而知，收到的这个片段被放在队列中，等待其余片段的到来，显然，在其余的片段不可能到达的情况下，这个片段最终也会因为超时而被丢弃。

49. **答：**在头中的错误比在数据中的错误更严重。例如，一个坏的地址可能导致分组被投递到错误的主机。许多主机并不检查投递给它们的分组是否确实是要投递给它们的。它们假定网络从来不会把本来是要前往另一个主机的分组邮递给它们，有的时候数据不参与检验和的计算，因为这样做代价大，上层协议通常也做这种检验工作，从而引起重复和多余。

50. **答：**在回答这一问题之前，我们需要搞清楚移动 IP 的概念，允许其用户漫游的每个场点都必须建立一个本地代理。允许外界访问的每个场点都要建立一个外部代理。当一个移动主机抵达一个外部场点时，它与那里的外部代理主机联系，并进行登记。然后，该外部代理主机与移动用户的原居住地的本地代理联系，并给它一个转交地址，通常就是该外部代理的 IP 地址。

当一个分组到达用户的本地 LAN 时，它进入连接到该 LAN 的某个路由器。路由器然后尝试以通常的方式寻找主机的位置。它广播一个 ARP 分组，询问（例如）“160.80.40.20”的以太网地址是什么？“本地代理通过给出自己的以太网地址来应答这个询问。路由器把前往 160.80.40.20 的分组发送给本地代理。本地代理又以隧道通信的方式把分组发送给转交地址，即前往外部代理。外部代理再取出 IP 分组，并投递到移动主机的数据链路地址。此外，原居住地的本地代理把转交地址提供给发送方，使得随后的分组可直接地隧道发往外部代理。

现在回到本题的解答。答案是仍然需要通过上述的本地代理和外部代理的一整套过程。实际上，明尼阿波利斯的局域网是无线网的事实并不会使得波士顿发给该用户的分组会突然

的跳到明尼阿波利斯。在波士顿的本地代理必须把分组以隧道方式传给明尼阿波利斯的无线 LAN 上的外部代理。看待这一问题的最好方法是用户必须接入明尼阿波利斯的 LAN，并且是以与在明尼阿波利斯的其他用户一样的方式接入。连接是使用无线方式还是有线没有关系。

51*注意根据英文版，本题中应为每 1ps 分配 100 万地址，而不是 12ps。

答：使用 16 个字节，总的地址数为 2^{128}或 3.4×10^{38}。如果我们以每皮秒 10^{6}，即每秒 10^{18} 的速率分配它们，这些地址将会持续 3.4×10^{20} s，大约 10^{13}年。这个数字是宇宙年龄的 1000 倍。

当然，地址空间不是扁平的，因此它们的分配不会是线性的。但这个计算结果表明，这么大的地址空间，几乎是永远也用不完的。

52. **答：**设置协议段的目的是要告诉目的地主机把 IP 分组交给那一个协议处理程序。中途的路由器并不需要这一信息，因此不必把它放在主头中。实际上，这个信息存在于头中，但被伪装了。最后一个（扩展）头的下一个字段就用于这一目的。

53. **答：**从概念上讲，不需要改变。在技术上，由于被请求的 IP 地址现在变大了，因此需要比较大的域。

第 6 章 传输层

1. 答：不是。事实上，LISTEN 调用可以表明建立新连接的意愿，但不封锁。当有了建立连接的尝试时，调用程序可以被提供一个信号。然后，它执行，比如说，OK 或 REJECT 来接受或拒绝连接。然而，在原先的封锁性方案中，就缺乏这种灵活性。

2. 答：从“被动连接建立在进行中”到“已建立”的虚线不再依确认的传输情况而定。该变迁可立即发生。实质上，“被动连接建立在进行中”状态已经消失，因为它们什么时候都不可见。

3. If the client sends a packet to *SERVER3PORT* and the server is not listening to that port, the packet will not be delivered to the server.

4. 答：在具体解答这个问题之前，需要先熟悉一下时钟驱动方案的内容。首先我们引入参数 T，假定在发送出一个分组之后等待长度等于 T 的时间，我们就可以肯定，所有关于该分组的踪迹都已消失，不管是该分组本身，还是对于它的确认都不会再以外的出现。我们还假定，每个主机都配有一个表示一天的时间的时钟，不同主机上的时钟不必同步。每个时钟都采用二进制计数器的形式，并且以长度一致的间隔时间递增。而且，计数器的比特数必须等于或超过序列号所使用的比特数。最后一点，时钟被假定是连续运行，即使主机关闭时也不间断。

时钟驱动方案的基本思想是同一时间不会有两个活动的 TPDUs 使用相同的序列号。在一条连接建立的时候，时钟的低端 k 个比特被用作初始序列号（也是 k 位）。因此，每条连接可以从不同的序列号开始为 TPDU 编号。序列号空间应该足够大，使得当编号循环一周时，具有相同号码的旧的 TPDU 已经不复存在。

当主机系统崩溃时会产生一些问题。在重新启动后，主机的传输层实体不知道它曾经处在序列号空间的什么位置。一种解决方法是要求传输实体在恢复后的 T 秒内处于空闲状态，让所有老的 TPDUs 都消失。然而，在一个复杂的互联网上，T 值可能很大，所以这不是一个好的解决方法。

为了避免从崩溃恢复后的 T 秒不工作状态，需要对序列号的使用施加新的限制。在一些编号可能被用作初始序列号之前，必须在长度为 T 的时间内禁止使用这些编号。在任何连接上发送 TPDU 之前，传输层实体必须读一次时钟，检查该 TPDU 的编号是否在禁止区内。

显然，在任何连接上的最大数据率是每个时钟滴答发送一个 TPDU。在系统崩溃后重启动时，在打开一条新的连接之前，传输实体必须等待到下一个时钟滴答，以避免同样的号码重复使用。如果数据速率低于始终速率，实际使用的序列号对于时间的曲线将最终从左边进入禁止区。如果这样的情况发生了，要么延迟 TPDU 达 T 长度时间，或者重新同步序列号。

作为例子，如果在坐标起点发 1 号 TPDU，到接近时钟大循环编码的末尾才发送第 2 个 TPDU，此时为避免在下一大循环开始重复使用序列号，就需要在大循环接近末尾处重新同步，使用大的初始序列号，以避免使用禁止区号码。

（a） 时钟大循环周期是 2^{15}，即 32768 滴答，每滴答 100ms，即 0.1 秒，所以大循环周期是 3276.8s 。假定数据产生速率非常低（接近零），那么发送方在 3276.8-60=3271.8 秒时进入禁止区，需要进行一次重新同步。

（b） 每分钟使用 240 个序列号，即每秒使用 4 个号码，如果时间以 t 表示（以秒为单位），那么实际的序列号是 4t。当接近大循环的末尾时以及在下一大循环的开始阶段，4t 有一定的大小，位于禁止区的上方，现在由于每秒钟 10 个滴答，禁止区的左边是 10(t-3216.8)。令 $4t=10(t-3216.8)$，得 $t=5316.3$ 秒。即当 $t=5316.3$ 时，开始进入禁止区，因此当 $t=5316.3$ 时需要进行一次重新同步。

5. **答：**首先看三次握手过程是如何解决延迟的重复到达的分组所引起的问题的。

正常情况下，当主机 1 发出连接请求时，主机 1 选择一个序号 x，并向主机 2 发送一个包含该序号的请求 TPDU；接着，主机 2 回应一个接受连接的 TPDU，确认 x，并声明自己所选用的初始序列号 y；最后，主机 1 在其发送的第一个数据 TPDU 中确认主机 2 所选择的初始序列号。

当出现延迟的重复的控制 TPDU 时，一个 TPDU 是来自于一个已经释放的连接的延迟重复的连接请求（ CONNECTION REQUEST），该 TPDU 在主机 1 毫不知情的情况下到达主机 2。

主机 2 通过向主机 1 发送一个接受连接的 TPDU（CONNECTION ACCEPTED）来响应该 TPDU，而该接受连接的 TPDU 的真正目的是证实主机 1 确实试图建立一个新的连接。在这一点上，关键在于主机 2 建议使用 y 作为从主机 2 到主机 1 交通的初始序列号，从而说明已经不存在包含序列号为 y 的 TPDU，也不存在对 y 的应答分组。当第二个延迟的 TPDU 到达主机 2 时，z 被确认而不是 y 被确认的事实告诉主机 2 这是一个旧的重复的 TPDU，因此废止该连接过程。在这里。三次握手协议是成功的。

最坏的情况是延迟的“连接请求”和对“连接被接收”的确认应答都在网络上存活。可以设想，当第 2 个重复分组到达时，如果在网上还存在一个老的对序列号为 y 的分组的确认应答，显然会破坏三次握手协议的正常工作，故障性的产生一条没有人真正需要的连接，从而导致灾难性的后果。

6. **答：**我们知道，3 次握手完成两个重要功能，既要双方做好发送数据的准备工作（双方都知道彼此已准备好），也要允许双方就初始序列号进行协商，这个序列号在握手过程中被发送与确认。

现在把三次握手改成仅需要两次握手，死锁是可能发生的。作为例子。考虑计算机 A 和 B 之间的通信。假定 B 给 A 发送一个连接请求分组，A 收到了这个分组，并发送了确认应答分组。按照两次握手的协定，A 认为连接已经成功的建立了，可以开始发送数据分组。

可是，B 在 A 的应答分组在传输中被丢失的情况下，将不知道 A 是否已经准备好，不知道 A 建议什么样的序列号用于 A 到 B 的交通，也不知道 A 是否同意 A 所建议的用于 B 到 A 交通的初始序列号，B 甚至怀疑 A 是否收到自己的连接请求分组。在这种情况下，B 认为连接还未建立成功，将忽略 A 发来的任何数据分组，只等待接收连接确认应答分组。而 A 在发出的分组超时后，重复发送同样的分组。这样就形成了死锁。

7. **答：**（a）参见教材。

（b）不存在。对于多于两支部队的情况，问题在实质上是同样的。

8. **答：**在解答本题前，让我们先考察主机从崩溃恢复所带来的问题。我们总是希望，在服务器崩溃随后又很快重新引导的情况下，客户机能够继续工作。为了说明这一问题的难度，我们假定一个客户主机发送一个长文件给另一个服务器主机，并且使用简单的停-等协议。在服务器上的传输层只是简单的把接收到的 TPDU 一个一个的递交给传输用户。假定在文件

传输的过程中，服务器崩溃了。当服务器恢复的时候，它的表被重新初始化，因此再也不知道崩溃前文件传送到什么地方了。

在试图恢复先前状态的过程中，服务器可能发送一个广播到所有其他主机，宣布自己刚刚发生了一次崩溃，请求客户告知所有打开的连接的状态。此时。每个客户机都可能处于二中选一的状态：有一个悬而未决的 TPDU 的 S1 状态，或者没有未确认应到的 TPDU 的 S0 状态。

可以想到的一种解决方案是基于这一状态信息，客户机决定是否要重复发最近的一个 TPDU。

乍看起来，这一解决方案似乎能解决问题，可是深入仔细的分析一下，困难仍然很大。作为示例，假定服务器的传输层实体先发送 ACK，在 ACK 被发出之后，再执行把收到的 TPDU 写到应用进程的操作。把 TPDU 写到输出设备和发送 ACK 是两个不同的事件，不能同时进行。如果服务器主机的崩溃刚好发生在应答被发送之后，并且是在写操作之前，那么客户机将接收到确认应答，当崩溃恢复到达时会处于状态 S0。因此，客户机不会重传 TPDU，错误的认为服务器成功的接收到并存放好了它最后一次发送的 TPDU。实际的情况并非如此，从而结果是丢失了最后一个 TPDU。

到此，你也许认为：“这个问题容易解决，只要你重新编写程序，让传输实体先执行写操作然后再发送 ACK 就可以了。”可是，写操作尽管成功了，但崩溃可能发生在发送出 ACK 之前。此时客户机将会处于状态 S1，因而重新发送，导致对服务器的应用进程的输出中产生未检测到的重复 TPDU。

如图 6-18 所示，服务器可以选择两种方式中的一种：先确认应答，或者先执行写操作。客户机可以选择 4 种方式中的一种：总是重传最后一个 TPDU，永不重传最后一个 TPDU，仅在 S0 状态时重传，或者仅在 S1 状态时重传。这样就存在 8 种可能的组合，但可以看出，对于每一种组合，都有一些事件会使协议的运行失败。

在服务器方可能发生 3 种事件：发送一个 ACK（A），对输出进程的写操作（W）和系统崩溃（C）。3 种事件可能以 6 种不同的次序发生：AC（W），AWC，C（AW），C（WA），WAC 和 WC（A），这里的圆括号表示，在系统崩溃 C 后，A 和 W 事件就不可能了。图 6-18 示出了客户机和服务器的策略的所有 8 种组合，以及对于每一种组合的有效事件序列。值得注意的是，对于每一种策略都存在某些事件会引起协议失败。例如，如果客户机选择总是重发送，AWC 事件将产生检测不出来的收到重复分组的错误。尽管对于 C（AW）和 C（WA）该协议都工作的很好。

现在回到本题的答案。如果 AW 或 WA 间隔时间很短，事件 AC（W）和 W（CA）就不太可能发生。此时的最好发送方策略是，如果崩溃恢复时处于状态 S1，应该重传最后一个 TPDU，接收方采用顺序 AW 或 WA 则无关紧要。

9. 答：该传输实体有可能死锁。当双方同时执行 RECEIVE 时就会进入死锁状态。

10. 答：有，$n_2 + n_3 + n_6 + n_7 = 1$

因为状态listening（n_2）、waiting（n_3）、sending（n_6）和receiving（n_7）都意味着用户被封锁，因此当处在其中的一个状态时，就不可能是在另一个状态。

11. 答：长度为零的报文被另一边接收。这种报文的发送可以被用来表示文件结束的信号。

12. 答：因为文件处于封锁状态，所有的传输层原语都不可能执行。因此，仅分组到达事件是可能的，而且还不是所有的到达事件。事实上，仅仅跟呼叫请求、清除请求、数据分组和信用量分组这几个分组到达有关的事件是合法的。

13. 答：滑动窗口协议比较简单，仅需要管理窗口边缘一组参数，而且，对于到达顺序有错的 TPDU 不会引起窗口增加和减少方面的问题。然而，信用量方案比较灵活，允许独立于确认，动态的管理缓冲区。

14. 答：仅仅使用 IP 分组还不够。IP 分组包含 IP 地址，该地址指定一个目的地机器。一旦这样的分组到达了目的地机器，网络控制程序如何知道该把它交给哪个进程呢？UDP 分组包含一个目的地端口，这一信息是必须的，因为有了它，分组才能够被投递给正确的进程。

15. It is possible that a client may get the wrong file. Suppose client *A* sends a request for file *f1* and then crashes. Another client *B* then uses the same protocol to request another file *f2*. Suppose client *B*, running on the same machine as *A* (with same IP address), binds its UDP socket to the same port that *A* was using earlier. Furthermore, suppose *B*'s request is lost. When the server's reply (to *A*'s request) arrives, client *B* will receive it and assume that it is a reply its own request.

16. 答：128 字节等于 1024 位，在 1Gb/s 的线路上发送 1000 位需要 1μs的时间。光在光导纤维中的传播速度是 200km/ms，请求到达服务器需要传输 0.5ms 的时间，应答返回又需要 0.5ms 的传输时间。总的看来，1000 位在 1ms 的时间内传输完成。这等效于 1Mb/s，即线路效率是 0.1%。

17. 答：在 1Gb/s，响应时间由光的速度决定。可以取得的最好情况是 1ms。在 1Mb/s，发射1024 位需要大约1ms 的时间，再经过0.5ms 最后一位到达服务器，还需要另外0.5ms 应答才能返回，这是最好的情况。因此，最好的 RPC 时间是 2ms。结论是，线路速度改善到 1000 倍，性能仅改善到 2 倍。对于这种应用，除非千兆位线路特别便宜，否则是不值得拥有的。

18. Here are three reasons. First, process IDs are OS-specific. Using process Ids would have made these protocols OS-dependent. Second, a single process may establish multiple channels of communications. A single process ID (per process) as the destination identifier cannot be used to distinguish between these channels. Third, having processes listen on well-known ports is easy, but well-known process IDs are impossible.

19. The default segment is 536 bytes. TCP adds 20 bytes and so does IP, making the default 576 bytes in total.

20. 答：尽管到达的每个数据报都是完整的，但可能到达的数据报的顺序是错误的，因此，TCP 必须准备适当的重组报文的各个部分。

21. Each sample occupies 4 bytes. This gives a total of 256 samples per packet. There are 44,100 samples/sec, so with 256 samples/packet, it takes 44100/256 or 172 packets to transmit one second's worth of music.

22. Sure. The caller would have to provide all the needed information, but there is no reason RTP could not be in the kernel, just as UDP is.

23. 答：不可以。一条连接仅仅用它的套接口标识。因此，（1，p）-（2，q）是在这两个端口之间唯一可能的连接。

24. The *ACK* bit is used to tell whether the 32-bit field is used. But if it were not there, the 32-bit field would always have to be used, if necessary acknowledging a byte that had already acknowledged. In short, it is not absolutely essential for normal data traffic. However, it plays a crucial role during connection establishment, where it is used in the second and third messages of the threeway handshake.

25. 答：整个 TCP 报文段必须适配 IP 分组 65,515 字节的载荷段。因为 TCP 头最少 20 个字节，所以仅剩下 65,495 字节用于 TCP 数据。

26. 答：一条途径是从 LISTEN 开始。如果收到一个 SYN，那么协议进入 SYN RECD 状态。

另一条途径是一个进程试图做一个主动打开操作，并发送一个 SYN。如果另一方也做打开操作，并收到一个 SYN，那么也将进入 SYN RECD 状态。

27. 答：Nagle 算法建议，当数据一次一个字节的来到发送方时，只发送第一个字节，并且缓冲所有其他内容，直到所发出的字节被确认为止。然后在一个 TCP 报文段中发送所有缓冲的字符。接着又开始缓冲，直到前一个报文段中的所有字节又被确认。这样，如果用户键入的速度足够快，而网络比较慢的话，那么在每个报文段中都可以有相当数量的字符。该算法还允许输入足够的数据以填满半个窗口或一个最大报文段的情况下发送一个新的分组。在这种运行方式下，尽管用户是以均匀的速度键入，而字符却是以突发的方式回印。用户可能敲击了好几个键，而屏面上什么都没有显示，然后突然的在屏幕上显示出所有已键入的字符。人们可能对此感到恼火。

28. 答：按照慢启动算法，经过 10、20、30、40ms 后拥塞窗口大小分别为 4、8、16、32，所以在 40ms 后将按照 min{24，32}=24KB 发送数据。

29. 答：由于发生了超时，下一次传输将是 1 个最大报文段，然后是 2 个、4 个、8 个最大报文段，所以在 4 次突发量传输后，拥塞窗口将是 8K 字节。

30. 答：对于每一条连接，TCP 都维持一个变量 RTT，它是当前到达目的地的最佳估计值。当发送一个报文段的时候，启动计时器，察看应答要花多长时间，如果时间太长，就要重发报文段。如果应答在超时前返回，TCP 就测量应答花了多长时间，比如说是 M，然后用下列公式更新 RTT 值：

$$RTT = \alpha RTT + (1-\alpha)M$$

现在，　α=0.9，RTT=30ms，M1=26，M2=32，M3=24

所以，

$$RTT1 = 0.9 \times 30 + (1-0.9) \times 26 = 29.6$$

$$RTT1 = 0.9 \times 30 + (1-0.9) \times 32 = 29.84$$

$$RTT1 = 0.9 \times 30 + (1-0.9) \times 24 = 29.256$$

因此，新的 RTT 估算值分别是 29.6ms、29.84ms、29.256ms。

31. 答：10ms*2=20ms

每 20ms 可以发送一个窗口大小的交通量，因此每秒 50 个窗口。

65536×8×50=26.2　　Mb/s

26.2/1000=2.6 %

所以，最大的数据吞吐率为 26.2Mb/s，线路效率为 2.6%。

32. The goal is to send 232 bytes in 120 sec or 35,791,394 payload bytes/sec. This is 23,860 1500-byte frames/sec. The TCP overhead is 20 bytes. The IP overhead is 20 bytes. The Ethernet overhead is 26 bytes. This means that for 1500 bytes of payload, 1566 bytes must be sent. If we are to send 23,860 frames of 1566 bytes every second, we need a line of 299 Mbps. With anything faster than this we run the risk of two different TCP segments having the same sequence number at the same time.

33. 答：具有相同编号的 TPDU 不应该同时在网络中传输，必须保证，当序列号循环回来重复使用的时候，具有相同序列号的 TPDU 已经从网络中消失。现在存活时间是 30 秒，那么在 30 秒的时间内发送方发送的 TPDU 的数目不能多于 255 个。

255×128×8/30=　8738b/s

所以，每条连接的最大数据速率是 8738b/s。

34. 答：计算平均值：

$$(270000\times 0+730000\times 1)\div(270000+730000)=0.73\,\text{ms}$$

因此，接收一个 TPDU 花 730 微秒的时间。

35. 答：拷贝 64 比特，即 8 个字节要用 2*6=12 条指令。12 条指令花 120ns，因此每个字节需要 15ns 的 CPU 时间。1000/15=66.67MB/s，即系统的处理能力是 66.67MB/s，也就是约 533Mb/s，这远远小于 1Gb/s 的处理需求，所以，这个系统不能够处理 1Gb/s 的线路。

36. 答：顺序号空间的大小是 2^{64}个字节，约为 2×10^{19}字节。75/8=9.375，即 75Tb/s 的发送器每秒钟消耗 9.375×10^{12}个顺序号。 $(2\times10^{19})/(9.375\times10^{12})\approx2\times10^{6}$，所以顺序号循环一周所花的时间为 2×10^{6} s，约为 23 天。因此，最长的分组生命周期小于 3 个星期可以避免顺序号循环重复的问题。

37. RPC over UDP takes only two packets instead of three. However, RPC has a problem if the reply does not fit in one packet.

38. Yes. Packet 6 acknowledges both the request and the FIN. If each one were acknowledged separately, we would have 10 packets in the sequence. Alternatively, Packet 9, which acknowledges the reply, and the FIN could also be split into two separate packets. Thus, the fact that there are nine packets is just due to good luck.

39. 答：

$$\frac{1}{10^{9}/(128\times8)}\approx10^{-6}$$

1μs可以处理完一个分组。考虑一半的 CPU 时间，要求 0.5μs处理一个分组。在 0.5μs内 100MIPS 的计算机可以执行 50 条指令。

With a packet 11.72 times smaller, you get 11.72 times as many per second, so each packet only gets 6250/11.72 or 533 instructions.

40. 答：光在光纤和铜导线中的速度大约为每毫秒 200km。对于一条 20km 的线路，单

向延迟是 100μs，往返延迟是 200μs。1K 字节就是 8192 位。如果发送 8192 位的时间为 200μs，那么发送延迟就等于传播延迟。设 W 是发送 1 位的时间，那么

$$W = \frac{200 \times 10^{-6}}{8192}$$

则 1 /W=8192/(2×10^{-4}) =40 × 10^{6}。所以，数据传输速率为 40Mb/s。

41. The answer are: (1) 18.75 KB, (2) 125 KB, (3) 562.5 KB, (4) 1.937 MB. A 16-bit window size means a sender can send at most 64 KB before having to wait for an acknowledgement. This means that a sender cannot transmit continuously using TCP and keep the pipe full if the network technology used is Ethernet, T3, or STS-3.

42. The round-trip delay is about 540 msec, so with a 50 Mbps channel the bandwidth-product delay is 27 megabits or 3,375,000 bytes. With packets of 1500 bytes, it takes 2250 packets to fill the pipe, so the window should be at least 2250 packets.

第 7 章 应用层

1. They are the DNS name, the IP address, and the Ethernet address.

2. Its IP address starts with 130, so it is on a class B network. See Chap. 5 for the IP address mapping.

3. It is not an absolute name, but relative to *.cs.vu.nl*. It is really just a shorthand notation for rowboat.cs.vu.nl.

4. It means: my lips are sealed. It is used in response to a request to keep a secret.

5. DNS is idempotent. Operations can be repeated without harm. When a process makes a DNS request, it starts a timer. If the timer expires, it just makes the request again. No harm is done.

6. The problem does not occur. DNS names *must* be shorter than 256 bytes. The standard requires this. Thus, all DNS names fit in a single minimumlength packet.

7. Yes. In fact, in Fig. 7-3 we see an example of a duplicate IP address. Remember that an IP address consists of a network number and a host number. If a machine has two Ethernet cards, it can be on two separate networks, and if so, it needs two IP addresses.

8. It is possible. www.large-bank.com and www.large-bank.ny.us could have the same IP address. Thus, an entry under *com* and under one of the country domains is certainly possible (and common).

9. There are obviously many approaches. One is to turn the top-level server into a server farm. Another is to have 26 separate servers, one for names beginning with *a*, one for *b*, and so on. For some period of time (say, 3 years) after introducing the new servers, the old one could continue to operate to give people a chance to adapt their software.

10. It belongs to the envelope because the delivery system needs to know its value to handle e-mail that cannot be delivered.

11. This is much more complicated than you might think. To start with, about half the world writes the given names first, followed by the family name, and the other half (e.g., China and Japan) do it the other way. A naming system would have to distinguish an arbitrary number of given names, plus a family name, although the latter might have several parts, as in John von Neumann.

Then there are people who have a middle initial, but no middle name. Various titles, such as Mr., Miss, Mrs., Ms., Dr., Prof., or Lord, can prefix the name. People come in generations, so Jr., Sr., III, IV, and so on have to be included. Some people use their academic titles in their names, so we need B.A., B.Sc., M.A., M.Sc., Ph.D., and other degrees. Finally, there are people who include certain awards and honors

in their name. A Fellow of the Royal Society in England might append FRS, for example. By now we should be able to please even the learned:

Prof. Dr. Abigail Barbara Cynthia Doris E. de Vries III, Ph.D., FRS

12. It is doable and relatively simple. When incoming e-mail arrives, the SMTP daemon that accepts it has to look up the login name in the *RCPT TO* message.

There is certainly a file or database where these names are located.

That file could be extended to have aliases of the form ‘‘Ellen.Johnson’’ that point to the person’s mailbox. Then e-mail can always be sent using the person’s actual name.

13. The base 64 encoding will break the message into 1024 units of 3 bytes each.

Each of these will be encoded as 4 bytes, for a total of 4096 bytes. If these are then broken up into lines of 80 bytes, 52 such lines will be needed, adding 52 CRs and 52 LFs. The total length will then be 4200 bytes.

14. If a sequence beginning with an equal sign and followed by two hexadecimal digits happens to appear in the text, e.g., =FF, this sequence will be mistakenly interpreted as an escape sequence. The solution is to encode the equal sign itself, so all equal signs always start escape sequences.

15. Some examples and possible helpers are application/msexcel(Excel), application/ppt (PowerPoint), audio/midi (MIDI sound), image/tiff (any graphics previewer), video/x-dv (QuickTime player).

16. Yes, use the *message/external-body* subtype and just send the URL of the file instead of the actual file.

17. The message sent just before logout will generate a canned reply. Its arrival will also generate a canned reply. Assuming each machine logs e-mail addresses to which it has already responded, no more canned replies will be sent.

18. First one is any sequence of one or more spaces and/or tabs. Second one is any sequence of one or more spaces and/or tabs and/or backspaces subject to the condition that the net result of applying all the backspaces still leaves at least one space or tab over.

19. The actual replies have to be done by the message transfer agent. When an SMTP connection comes in, the message transfer agent has to check whether a vacation daemon is set up to respond to the incoming e-mail, and if so, send an answer. The user transfer agent cannot do this because it will not even be invoked until the user comes back from vacation.

20. No. The POP3 program does not actually touch the remote mailbox. It sends commands to the POP3 daemon on the mail server. As long as that daemon understands the mailbox format, it can work. Thus, a mail server could change from one format to another overnight without telling its customers, as long as it simultaneously changes its POP3 daemon so it understands the new format.

21. Storing users' e-mail takes up disk space, which costs money. This factor argues for using POP3. On the other hand, the ISP could charge for disk storage above a few megabytes, thus turning e-mail into a moneymaker. The latter argues for IMAP to encourage users to keep e-mail on the server (and pay for disk space).

22. It does not use either one. But it is fairly similar in spirit to IMAP because both of them allow a remote client to examine and manage a remote mailbox. In contrast, POP3 just sends the mailbox to the client for processing there.

23. The browser has to be able to know whether the page is text, audio, video, or something else. The MIME headers provide this information.

24. If a browser receives a page with a MIME type that it cannot handle, it calls an external viewer to display the page. It finds the viewer's name in a configuration table, or it gets it from the user.

25. Yes, it is possible. Which helper is started depends on the configuration tables inside the browser, and Netscape and IE may have been configured differently.

Furthermore, IE takes the file extension more seriously than the MIME type, and the file extension may indicate a different helper than the MIME type.

26. If a module gets two requests, one will be a cache hit and one will be a cache miss on average. The total CPU time consumed is 1 msec, and the total wait time is 9 msec. This gives a 10% CPU utilization, so with 10 modules the CPU is kept busy.

27. The official RFC 1738 way to do this is *http://dns-name:port/file.*

28. DNS names may not end with a digit, so there is no ambiguity.

29. The URL is probably ftp://www.cs.stanford.edu/ftp/pub/freebies/newprog.c

30. Do it the way *toms-casino* does: just put a customer ID in the cookie and store the preferences in a database on the server indexed by customer ID. That way the size of the record is unlimited.

31. Technically, it will work but it is a terrible idea. All the customer has to do is modify the cookie to get access to someone else's bank account. Having the cookie provide the customer's identity is safe, but the customer should be required to enter a password to prove his identity.

32. If the user has turned off the automatic displaying of images or if images cannot be displayed for some other reason, then the text given in *ALT* is displayed instead of the image. Also, if the mouse hovers over the image, the text may be displayed.

33. A hyperlink consists of and . In between them is the clickable text. It is also possible to put an image here. For example:

<img src="http://www.abcd.com/im/im2">

34. It would be <a href="http://www.acm.org"> ACM <a> .

35. Here is one way to do it.

<html>

<head> <title> INTERBURGER </title> </head>
<body>
<h1> Interburger's order form </h1>
<form action="http://interburger.com/cgi-bin/burgerorder" method=POST>
 Name <input name="customer" size=46>
 Street Address <input name="address" size=40>
 City <input name="city" size=20>
Burger size Gigantic <input name="size" type=radio value="gigantic">
Immense <input name="size" type=radio value="immense">
Cheese <input name="cheese" type=checkbox>
 <input type=submit value="submit order">
</form>
</body> </html>

36. The page that displays the form looks like this:

```
<html>
<head> <title> Adder </title> </head>
<body>
<form action="action.php" method="post">
<p> Please enter first number: <input type="text" name="first"> </p>
<p> Please enter second number: <input type="text" name="second"> </p>
<input type="submit">
</form>
</body>
</html>
```

The PHP script that does the processing looks like this:

```
<html>
<head> <title> Addition </title> </head>
<body>
The sum is <?PHP echo $first + $second; ?>
</body>
</html>
```

37. (a) There are only 14 annual calendars, depending on the day of the week on which 1 January falls and whether the year is a leap year. Thus, a JavaScript program could easily contain all 14 calendars and a small database of which year gets which calendar. A PHP script could also be used, but it would be slower.

(b) This requires a large database. It must be done on the server by usingPHP.

(c) Both work, but JavaScript is faster.

38. There are obviously many possible solutions. Here is one.

```
<html>
```

```
<head> <title> JavaScript test </title> </head>
<script language="javascript" type="text/javascript">
function response(test3form) {
var n = 2;
var has3factors = 0;
var number = eval(test3form.number.value);
var limit = Math.sqrt(number);
while (n++ < limit) if (number % n == 0) has3factors = 1;
document.open();
document.writeln("<html> <body>");
if (has3factors > 0) document.writeln(number, " is not a prime");
if (has3factors == 0) document.writeln(number, " is a prime");
document.writeln("</body> </html>");
document.close();
}
</script>
</head>
<body>
<form name="myform">
Please enter a number: <input type="text" name="number">
<input type="button" value="compute primality" onclick="response(this.form)">
</form>
</body>
</html>
```

Clearly, this can be improved in various ways, but these require a bit more knowledge of JavaScript.

39. The commands sent are as follows:

```
GET /welcome.html HTTP/1.1
Host: www.info-source.com
```

Note the blank line at the end. It is mandatory.

40. Most likely HTML pages change more often than JPEG files. Lots of sites fiddle with their HTML all the time, but do not change the images much. But the effectiveness relates to not only the hit rate but also the payoff. There is not much difference between getting a 304 message and getting 500 lines of HTML. The delay is essentially the same in both cases because HTML files are so small. Image files are large, so not having to send one is a big win.

41. No. In the sports case, it is known days in advance that there will be a big crowd at the Web site and replicas can be constructed all over the place. The essence of a flash crowd is that it is unexpected. There was a big crowd at the Florida

Web site but not at the Iowa or Minnesota sites. Nobody could have predicted this in advance.

42. Sure. The ISP goes to a number of content providers and gets their permission to replicate the content on the ISP's site. The content provider might even pay for this. The disadvantage is that it is a lot of work for the ISP to contact many content providers. It is easier to let a CDN do this.

43. It is a bad idea if the content changes rapidly. Pages full of up-to-the second sports results or stock quotes are not good candidates, for example. Pages that are generated dynamically are not suitable.

44. Each Japanese kanji (word) has been assigned a number. There are about 20,000 of them in Unicode. For an all-English system, it would be possible to assign the 65,000 most common words a 16-bit code and just transmit the code. The terminal would automatically add a space between words. Words not in the list, would be spelled out in ASCII. Using this scheme, most words would take 2 bytes, far less than transmitting them character by character.

Other schemes might involve using 8-bit codes for the most common words and longer codes for less frequent codes (primitive Huffman coding).

45. Audio needs 1.4 Mbps, which is 175 KB/sec. On a 650-MB device, there is room for 3714 sec of audio, which is just over an hour. CDs are never more than an hour long, so there is no need for compression and it is not used.

46. The true values are $\sin(2 \quad i/32)$ for i from 1 to 3. Numerically, these sines are 0.195, 0.383, and 0.556. They are represented as 0.250, 0.500, and 0.500, respectively. Thus, the percent errors are 28, 31, and 10 percent, respectively.

47. In theory it could be used, but Internet telephony is real time. For music, there is no objection to spending 5 minutes to encode a 3-minute song. For real-time speech, that would not work. Psychoacoustic compression could work for telephony, but only if a chip existed that could do the compression on the fly with a delay of around 1 msec.

48. It takes 50 msec to get a pause command to the server, in which time 6250 bytes will arrive, so the low-water mark should be way above 6250, probably 50,000 to be safe. Similarly, the high-water mark should be at least 6250 bytes from the top, but, say, 50,000 would be safer.

49. It introduces extra delay. In the straightforward scheme, after 5 msec have elapsed, the first packet can be sent. In this scheme, the system has to wait until 10 msec until it can send the samples for the first 5 msec.

50. It depends. If the caller is not behind a firewall and the callee is at a regular telephone, there are no problems at all. If the caller is behind a firewall and the firewall is not picky about what leaves the site, it will also work. If the callee is behind a firewall that will not let UDP packets out, it will not work.

51. The number of bits/sec is just 800 600 40 8 or 153.6 Mbps.

52. Yes. An error in an I-frame will cause errors in the reconstruction of subsequent P-frames and B-frames. In fact, the error will continue to propagate until the next I-frame.

53. With 100,000 customers each getting two movies per month, the server outputs 200,000 movies per month or about 6600 per day. If half of these are at P.M., the server must handle about 3300 movies at once. If the server has to transmit 3300 movies at 4 Mbps each, the required bandwidth is 13.2 Gbps.

Using OC-12 connections, with a SPE capacity of 594 Mbps each, at least 23 connections will be needed. A machine serving 3300 movies simultaneously over 23 OC-12 connections is not a small machine.

54. The fraction of all references to the first r movies is given by

$C/1$ $C/2$ $C/3$ $C/4$ . . . C/r

Thus, the ratio of the first 1000 to the first 10,000 is

1/1 1/2 1/3 1/4 . . . 1/10000

1/1 1/2 1/3 1/4 . . . 1/1000

3333333333333333333333333333333333

because the Cs cancel out. Evaluating this numerically, we get 7.486/9.788.

Thus, about 0.764 of all requests will be to movies on magnetic disk.

Noteworthy is that Zipf's law implies that a substantial amount of the distribution

is in the tail, compared, say, to exponential decay.

第 8 章 网络安全

1. the time has come the walrus said to talk of many things of shoes and ships and sealing wax of cabbages and kings and why the sea is boiling hot and whether pigs have wings but wait a bit the oysters cried before we have our chat for some of us are out of breath and all of us are fat no hurry said the carpenter they thanked him much for that From Through the Looking Glass (Tweedledum and Tweedledee).

2. The plaintext is: a digital computer is a machine that can solve problems for people by carrying out instructions given to it. From Structured Computer Organization by A. S. Tanenbaum.

3. It is:

1011111 0000100 1110000 1011011 1001000 1100010 0001011 0010111 1001101 1110000 1101110

4. At 100 Gbps, a bit takes 10 11 sec to be transmitted. With the speed of light being 2 108 meters/sec, in 1 bit time, the light pulse achieves a length of 2 mm or 2000 microns. Since a photon is about 1 micron in length, the pulse is 2000 photons long. Thus, we are nowhere near one photon per bit even at 100 Gbps. Only at 200 Tbps do we achieve 1 bit per photon.

5. Half the time Trudy will guess right. All those bits will be regenerated correctly. The other half she will guess wrong and send random bits to Bob.

Half of these will be wrong. Thus, 25% of the bits she puts on the fiber will be wrong. Bob's one-time pad will thus be 75% right and 25% wrong.

6. If the intruder had infinite computing power, they would be the same, but since that is not the case, the second one is better. It forces the intruder to do a computation to see if each key tried is correct. If this computation is expensive,it will slow the intruder down.

7. Yes. A contiguous sequence of P-boxes can be replaced by a single P-box. Similarly for S-boxes.

8. For each possible 56-bit key, decrypt the first ciphertext block. If the resulting plaintext is legal, try the next block, etc. If the plaintext is illegal, try the next key.

9. The equation 2n 1015 tells us n, the number of doubling periods needed. Solving, we get n 15 log210 or n 50 doubling periods, which is 75 years.

Just building that machine is quite a way off, and Moore's law may not continue for 75 more years.

10. The equation we need to solve is 2256 10n. Taking common logarithms, we get n 256 log 2, so n 77. The number of keys is thus 1077. The number of stars in our galaxy is about 1012 and the number of galaxies is about 108, so there are

about 1020 stars in the universe. The mass of the sun, a typical star, is 2 1033 grams. The sun is made mostly of hydrogen and the number of atoms in 1 gram of hydrogen is about 6 1023 (Avogadro's number). So the number of atoms in the sun is about 1.2 1057. With 1020 stars, the number of atoms in all the stars in the universe is about 1077. Thus, the number of 256-bit AES keys is equal to the number of atoms in the whole universe (ignoring the dark matter). Conclusion: breaking AES-256 by brute force is not likely to happen any time soon.

11. DES mixes the bits pretty thoroughly, so a single bit error in block Ci will completely garble block Pi . In addition, one bit will be wrong in block Pi 1. However, all subsequent plaintext blocks will be correct. A single bit error thus only affects two plaintext blocks.

12. Unfortunately, every plaintext block starting at Pi 1 will be wrong now, since all the inputs to the XOR boxes will be wrong. A framing error is thus much more serious than an inverted bit.

13. Cipher block chaining produces 8 bytes of output per encryption. Cipher feedback mode produces 1 byte of output per encryption. Thus, cipher block chaining is eight times more efficient (i.e., with the same number of cycles you can encrypt eight times as much plaintext).

14. (a) For these parameters, z 60, so we must choose d to be relatively prime to 60. Possible values are: 7, 11, 13, 17, and 19.

(b) If e satisfies the equation 7e 1 mod 360, then 7 e must be 361, 721, 1081, 1441, etc. Dividing each of these in turn by 7 to see which is divisible by 7, we find that 721/7 = 103, hence e 103.

(c) With these parameters, e 3. To encrypt P we use the function C P3 mod 55. For P = 1 to 10, C = 1, 8, 27, 9, 15, 51, 13, 17, 14, and 10, respectively.

15. Maria should consider changing her keys. This is because it is relatively easy for Frances to figure out Maria's private key as follows. Frances knows Maria's public key is (e 1, n 1). Frances notices n 2 n 1. Frances now can guess Maria's private key(d 1, n 1) by simply enumerating different solutions of the equation d 1 e 1 1 modn 1.

16. No. The security is based on having a strong crypto algorithm and a long key. The IV is not really essential. The key is what matters.

17. The RAs from the last message may still be in RAM. If this is lost, Trudy can try to replay the most recent message to Bob, hoping that he will not see that it is a duplicate. One solution is for Bob to write the RA of every incoming message to disk before doing the work. In this case, the replay attack will not work. However, there is now a danger that if a request is written to disk followed shortly by a crash, the request is never carried out.

18. If Trudy replaces both parts, when Bob applies Alice's public key to the

signature, he will get something that is not the message digest of the plaintext.

Trudy can put in a false message and she can hash it, but she cannot sign it with Alice's private key.

19. When a customer, say, Sam, indicates that he wants to buy some pornography, gamble, or whatever, the Mafia order a diamond on Sam's credit card from a jeweler. When the jeweler sends a contract to be signed (presumably including the credit card number and a Mafia post office box as address), the Mafia forwards the hash of the jeweler's message to Sam, along with a contract signing up Sam as a pornography or gambling customer. If Sam just signs blindly without noticing that the contract and signature do not match, the Mafia forward the signature to the jeweler, who then ships them the diamond.

If Sam later claims he did not order a diamond, the jeweler will be able to produce a signed contract showing that he did.

20. With 20 students, there are (20 19)/2 190 pairs of students. The probability that the students in any pair have the same birthday is 1/365, and the probability that they have different birthdays is 364/365. The probability that all 190 pairs have different birthdays is thus (364/365)190. This number is about 0.594. If the probability that all pairs are mismatches is 0.594, then the probability that one or more pairs have the same birthday is about 0.406.

21. The secretary can pick some number (e.g., 32) spaces in the letter, and potentially replace each one by space, backspace, space. When viewed on the terminal, all variants will look alike, but all will have different message digests, so the birthday attack still works. Alternatively, adding spaces at the end of lines, and interchanging spaces and tabs can also be used.

22. It is doable. Alice encrypts a nonce with the shared key and sends it to Bob. Bob sends back a message encrypted with the shared key containing the nonce, his own nonce, and the public key. Trudy cannot forge this message, and if she sends random junk, when decrypted it will not contain Alice's nonce. To complete the protocol, Alice sends back Bob's nonce encrypted with Bob's public key.

23. Step 1 is to verify the X.509 certificate using the root CA's public key. If it is genuine, she now has Bob's public key, although she should check the CRL if there is one. But to see if it is Bob on the other end of the connection, she needs to know if Bob has the corresponding private key. She picks a nonce and sends it to him with his public key. If Bob can send it back in plaintext, she is convinced that it is Bob.

24. First Alice establishes a communication channel with X and asks X for a certificate to verify his public key. Suppose X provides a certificate signed by another CA Y. If Alice does not know Y, she repeats the above step with Y.

Alice continues to do this, until she receives a certificate verifying the public

key of a CA Z signed by A and Alice knows A's public key. Note that this may continue until a root is reached, that is, A is the root. After this Alice verifies the public keys in reverse order starting from the certificate that Z provided. In each step during verification, she also checks the CRL to make sure that the certificate provided have not been revoked. Finally, after verifying Bob's public key, Alice ensures that she is indeed to talking to Bob using the same method as in the previous problem.

25. No. AH in transport mode includes the IP header in the checksum. The NAT box changes the source address, ruining the checksum. All packets will be perceived as having errors.

26. HMACs are much faster computationally.

27. Incoming traffic might be inspected for the presence of viruses. Outgoing traffic might be inspected to see if company confidential information is leaking out. Checking for viruses might work if a good antivirus program is used.

Checking outgoing traffic, which might be encrypted, is nearly hopeless against a serious attempt to leak information.

28. If Jim does not want to reveal who he is communicating with to anyone (including his own system administrator, then Jim needs to use additional security mechanisms. Remember that VPN provides security for communication only over the Internet (outside the organization). It does not provide any security for communication inside the organization. If Jim only wants to keep his communication secure from people outside the company, a VPN is sufficient.

29. Yes. Suppose that Trudy XORs a random word with the start of the payload and then XORs the same word with the checksum. The checksum will still be correct. Thus, Trudy is able to garble messages and not have them be detected because she can manipulate the checksum through the encryption.

30. In message 2, put RB inside the encrypted message instead of outside it. In this way, Trudy will not be able to discover RB and the reflection attack will not work.

31. Bob knows that gx mod n 191. He computes 19115 mod 719 40. Alice knows that gy mod n 543. She computes 54316 mod n 40. The key is 40.

The simplest way to do the above calculations is to use the UNIX bc program.

32. There is nothing Bob knows that Trudy does not know. Any response Bob can give, Trudy can also give. Under these circumstances, it is impossible for Alice to tell if she is talking to Bob or to Trudy.

33. The KDC needs some way of telling who sent the message, hence which decryption key to apply to it.

34. No. All Trudy has to do is capture two messages from or to the same user. She can then try decrypting both of those with the same key. If the random number

field in both of them is the same, bingo, she has the right key. All this scheme does is increase her workload by a factor of two.

35. The two random numbers are used for different purposes. RA is used to convince Alice she is talking to the KDC. RA 2 is used to convince Alice she is talking to Bob later. Both are needed.

36. If AS goes down, new legitimate users will not be able to authenticate themselves, that is, get a TGS ticket. So, they will not be able to access any servers in the organization. Users that already have a TGS ticket (obtained from AS before it went down) can continue to access the servers until their TGS ticket lifetime expires. If TGS goes down, only those users that already have a server ticket (obtained from TGS before it went down) for a server S will be able to access S until their server ticket lifetime expires. In both cases, no security violation will occur.

37. It is not essential to send RB encrypted. Trudy has no way of knowing it, and it will not be used again, so it is not really secret. On the other hand, doing it this way allows a tryout of KS to make doubly sure that it is all right before sending data. Also, why give Trudy free information about Bob's random number generator? In general, the less sent in plaintext, the better, and since the cost is so low here, Alice might as well encrypt RB.

38. The bank sends a challenge (a long random number) to the merchant's computer, which then gives it to the card. The CPU on the card then transforms it in a complex way that depends on the PIN code typed directly into the card.

The result of this transformation is given to the merchant's computer for transmission to the bank. If the merchant calls up the bank again to run another transaction, the bank will send a new challenge, so full knowledge of the old one is worthless. Even if the merchant knows the algorithm used by the smart cards, he does not know the customer's PIN code, since it is typed directly into the card. The on-card display is needed to prevent the merchant from displaying: ‘‘Purchase price is 49.95’’ but telling the bank it is 499.95.

39. Compression saves bandwidth, but more important, it also wipes out the frequency information containined in the plaintext (e.g., that ‘‘e’’ is the most common letter in English text). In effect, it converts the plaintext into junk, increasing the amount of work the cryptanalyst must do to break the message. 40. No. Suppose the address was a mailing list. Each person would have his or her own public key. Encrypting the IDEA key with just one public key would not work. It would have to be encrypted with multiple public keys.

41. In step 3, the ISP asks for www.trudy-the-intruder.com and it is never supplied. It would be better to supply the IP address to be less conspicuous. The result should be marked as uncacheable so the trick can be used later if necessary.

42. The DNS code is public, so the algorithm used for ID generation is public.

If it is a random number generator, using random IDs hardly helps at all. By using the same spoofing attack as shown in the text, Trudy can learn the current (random) ID. Since random number generators are completely deterministic, if Trudy knows one ID, she can easily calculate the next one. If the random number generated by the algorithm is XORed with the time, that makes it less predictable, except that Trudy also knows the time. XORing the random number with the time and also with the number of lookups the server has done in the past minute (something Trudy does not know) and then taking the SHA-1 hash of this is much better. The trouble here is that SHA-1 takes a nontrivial amount of time and DNS has to be fast.

43. The nonces guard against replay attacks. Since each party contributes to the key, if an intruder tries to replay old messages, the new key generated will not match the old one.

44. Easy. Music is just a file. It does not matter what is in the file. There is room for 294,912 bytes in the low-order bits. MP3s require roughly 1 MB per minute, so about 18 sec of music could fit.

45. Alice could hash each message and sign it with her private key. Then she could append the signed hash and her public key to the message. People could compare verify the signature and compare the public key to the one Alice used last time. If Trudy tried to impersonate Alice and appended Alice's public key, she would not be able to get the hash right. If she used her own public key, people would see it was not the same as last time.