第四章 作业

1. R17. 假定主机 A 向主机 B 发送封装在一个 IP 数据报中的 TCP 报文段. 当主机 B 接受到该数据报时, 主机 B 中的网络层怎样知道它应当将该报文段(即数据报的有效载荷)交给 TCP 而不是 UDP 或某个其他东西呢?

答： 主机 B 中的网络层通过查看数据报首部字段中的协议字段得知应该把数据部分交给哪个特定的运输层协议. 比如值为 6 表明交给 TCP, 值为 17 表明交给 UDP.

2. R18. 在 IP 首部中, 哪个字段能用来确保一个分组的转发不超过 N 台路由器.

答：寿命字段(Time-To-Live, TTL), 每当一台路由器处理该数据报时该值减 1, 当减到 0 时丢弃该数据报.

3. R19. 前面讲过因特网检验和被用于运输层报文段以及网络层数据报，现在考虑一个运输层报文段封装在一个 IP 数据报中，在报文段首部和数据报首部中的检验和要遍及 IP 数据报中的任何共同字节进行计算吗?

答：不需要. 首先搞清楚报文段检验和与数据报检验和之间的差别. - 数据报检验和只是对 IP 数据报的首部计算了检验和, 而报文段首部中的检验和是对整个报文段(包括承载的应用层数据部分)都进行计算的.

4. P2 假设两个分组在完全相同的时刻到达一台路由器的两个不同输入端口。同时假设在该路由器中没有其他分组。

a.假设这两个分组朝着两个不同的输入端口转发。当交换结构使用一条共享总线时，这两个分组可能在相同时刻通过该交换结构转发吗？

b.假设这两个分组朝着两个不同的输出端口转发。当交换结构使用经内存交换时，这两个分组可能在相同时刻通过该交换结构转发吗？

c.假设这两个分组朝着相同的输出端口转发。当交换结构使用纵横式时，这两个分组可能在相同时刻通过该交换结构转发吗？

答：

a. 不可能。当交换结构使用一条共享总线时，由于总线是共享的，同一时刻只能传送一个分组，因此两个分组不能同时进行转发。
b. 不可能。使用内存交换时，类似于 I/O 设备进行中断，这种方式也是一次只能处理一个服务，需要共享总线来传输数据，因此也不能同时进行转发。
c. 不可能。送往相同端口的两个分组共享同一个输出总线，不可能同时被转发。

注释：

（1）经内存交换：这种就像是计算机（最简单、最原始的路由器就是计算机），在输入端口与输出端口之间的交换是在 CPU（路由选择处理器）的直接控制下完成的。

（2）经总线交换：输入端口经一根共享总线将分组直接传送到输出端口，不需要选路处理器的干预。因为每个分组必须跨越单一总线（同时只能有一个分组可以跨越总线），所以路由器的交换宽带受总线速率的限制。

（3）经互联网络交换：可以克服单一、共享式总线带宽限制，可以并行转发多个分组。但如果来自两个不同输入端口的两个分组其目的地为相同的输出端口，则一个分组必须在输入端等待。综上，经互联网络交换的纵横式交换机可以跨越交换结构并行发送多个分组。

5. P8 考虑互联 3 个子网（子网 1，子网 2 和子网 3）的一台路由器：假定这 3 个子网的所有接口要求具有前缀 223.1.17/24。还假定子网 1 要求支持多达 60 个接口，子网 2 要求支持多达 90 个接口，子网 3 要求支持多达 12 个接口。提供三个满足这些限制的网络地址（形式为 a.b.e.d/x）。

答：子网 1 要求 60 个接口 2^6=64　6 位　32-6=26 位

　　子网 2 要求 90 个接口 2^7=128　7 位　32-7=25 位

　　子网 3 要求 12 个接口 2^4=16　4 位　32-4=28 位

223.1.17.0

11011111.00000001.00010001.00000000

子网 2：223.1.17.0/25　11011111.00000001.00010001.0 0000000

子网 1：223.1.17.128/26　11011111.00000001.00010001.10 000000

子网 3：223.1.17.192/28　11011111.00000001.00010001.1100 0000

提示：

此题有多种答案，都正确：

答案 1：

（1）子网 1：223.1.17.128/26

　　子网 2：223.1.17.0/25

　　子网 3：223.1.17.192/28

答案 2：

（2）子网 1：223.1.17.128/25

　　子网 2：223.1.17.0/26

　　子网 3：223.1.17.192/28

6. P12

考虑图 4-20 中显示的拓扑。（在 12:00 以顺时针开始）标记具有主机的 3 个子网为网络 A、B 和 C，标记没有主机的子网为网络 D、E 和 F。

a. 为这 6 个子网分配网络地址，要满足下列限制：所有地址必须从 214.97.254/23 起分配；子网 A 应当具有足够地址以支持 250 个接口；子网 B 应当具有足够地址以支持 120 个接口；子网 C 应具有足够地址以支持 120 个接口。当然，子网 D、E 和 F 应当支持两个接口，对于每个子网，分配采用的形式是 a.b.c.d/x 或 a.b.c.d/x-e.f.g.h/y.

b. 使用你对（a）部分的答案，为这 3 台路由器提供转发表（使用最长前缀匹配）。

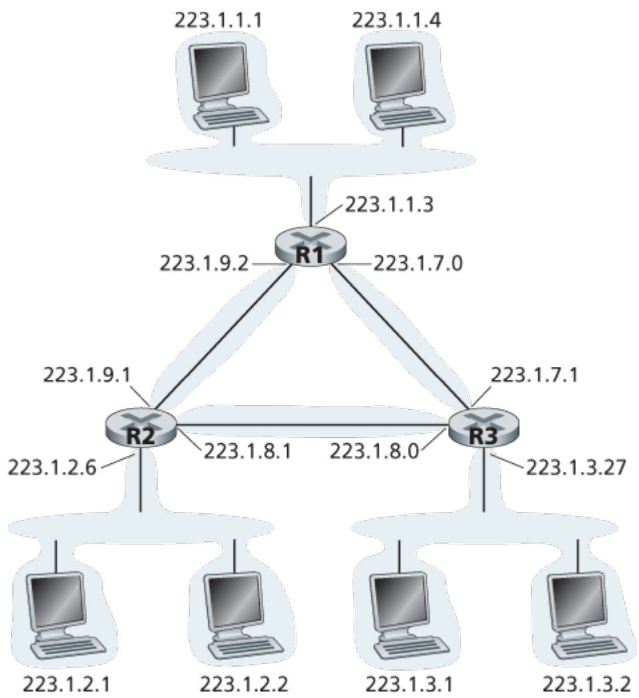


Figure 4.20 Three routers interconnecting six subnets

答：注意：此题答案不唯一，注意：前缀匹配时要唯一

可能的一种答案：

214.97.254.0/23 11010110. 11100001. 1111111 0.00000000

子网 A 要求 250 个接口 2^8=256 8 位 32-8=24 位

子网 B 要求 120 个接口 2^7=128 7 位 32-7=25 位

子网 C 要求 120 个接口 2^7=128 7 位 32-7=25 位

子网 D 要求 2 个接口 2^1=2 1 位 32-1=32 位

子网 E 要求 2 个接口 2^1=2 1 位 32-1=32 位

子网 F 要求 2 个接口 2^1=2 1 位 32-1=32 位

子网 A：214.97.254.0/24

11010110. 11100001. 11111110. 00000000

子网 B：214.97.255.0/25

11010110. 11100001. 11111111. 0 0000000

子网 C：214.97.255.128/25-214.97.255.247/29 (构造避免与 DEF 冲突的地址)

11010110. 11100001. 11111111. 1 0000000 - 11010110. 11100001. 11111111. 11110111

子网 D：214.97.255.248/31

11010110. 11100001. 1111111 1.1111100 0

子网 E：214.97.255.250/31

11010110. 11100001. 1111111 1.1111101 0

子网 F：214.97.255.252/31

11010110. 11100001. 11111111. 1111110 0

路由器 R1

11010110 01100001 11111110 子网 A

11010110 01100001 11111111 1111100　子网 D
11010110 01100001 11111111 111111　子网 F
路由器 R2
11010110 01100001 11111111 1　子网 C
11010110 01100001 11111111 1111101　子网 E
11010110 01100001 11111111 111111 子网 F
路由器 R3
11010110 01100001 11111111 0　子网 B
11010110 01100001 11111111 1111100　子网 D
11010110 01100001 11111111 1111101　子网 E

Router 3

Longest Prefix Match	Outgoing Interface
11010110 01100001 11111111 000001	Subnet F
11010110 01100001 11111110 0000001	Subnet E
11010110 01100001 11111110 1	Subnet C

7. P14 Consider sending a 2400-byte datagram into a link that has an MTU of 700 bytes. Suppose the original datagram is stamped with the identification number 422. How many fragments are generated? What are the values in the various fields in the IP datagram(s) generated related to fragmentation? P15. Suppose datagrams are limited to 1,500 bytes (including header) between source Host A and destination Host B. Assuming a 20-byte IP header, how many datagrams would be required to send an MP3 consisting of 5 million bytes? Explain how you computed your answer.
P14 提醒：偏移量以 8 个字节为一个单位。
考虑向具有 700 字节 MTU 的一条链路发送一个 2400 字节的数据报。假定初始数据报标有标识号 422。将会生成多少个分片？在生成相关分片的数据报中各个字段的值是多少？

答：

IP 数据报首部为 20B，因此每个片的数据段为 700-20=680B

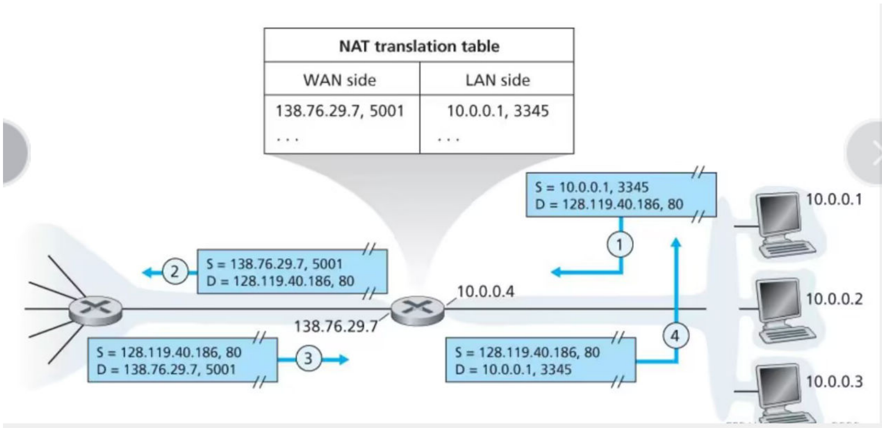
（2400-20）/680 ≈ 4 （向上取整） 所以会生成 4 个分片
680/8=85（每一片偏移 85 个单位）
（2400-20）- 680*3 = 340 （最后一片数据大小 340B）

第一片：
Length=700 ID=422 fragflag=1 offset=0 IP 数据报的数据字段中的 680B
第二片：
Length=700 ID=422 fragflag=1 offset=85 680B 数据
第三片：
Length=700 ID=422 fragflag=1 offset=170 680B 数据
第四片：
Length=360 ID=422 fragflag=0 offset=255 340B 数据
P15：

MP3 file size = 5 million bytes. Assume the data is carried in TCP segments, with each TCP segment also having 20 bytes of header. Then each datagram can carry 1500-40=1460 bytes of the MP3 file

Number of datagrams required $= \left\lceil \frac{5\times10^6}{1460} \right\rceil = 3425$. All but the last datagram will be 1,500 bytes; the last datagram will be 960+40 = 1000 bytes. Note that here there is no fragmentation – the source host does not create datagrams larger than 1500 bytes, and these datagrams are smaller than the MTUs of the links.

8. 8. P16. Consider the network setup in Figure 4.25 . Suppose that the ISP instead assigns the router the address 24.34.112.235 and that the network address of the home network is 192.168.1/24. a. Assign addresses to all interfaces in the home network. b. Suppose each host has two ongoing TCP connections, all to port 80 at host 128.119.40.86. Provide the six corresponding entries in the NAT translation table.

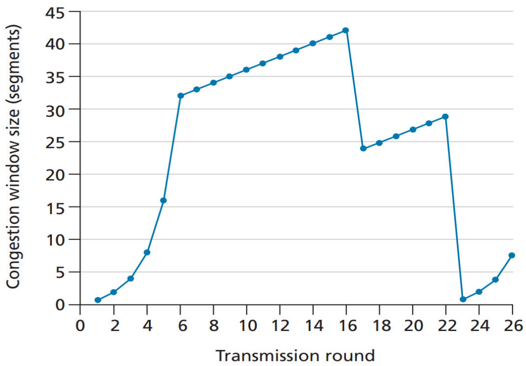


答案：Home addresses: 192.168.1.1, 192.168.1.2, 192.168.1.3 with the router interface being 192.168.1.4
b)
NAT Translation Table
WAN Side LAN Side
（下面端口号只要能区分开即可，答案不唯一）
24.34.112.235, 4000 192.168.1.1, 3345

24.34.112.235, 4001	192.168.1.1, 3346
24.34.112.235, 4002	192.168.1.2, 3445
24.34.112.235, 4003	192.168.1.2, 3446
24.34.112.235, 4004	192.168.1.3, 3545
24.34.112.235, 4005	192.168.1.3, 3546

Problem 40（复习题）

Consider Figure 3.61. Assuming TCP Reno is the protocol experiencing the behavior shown above, answer the following questions. In all cases, you should provide a short discussion justifying your answer.



a) TCP slowstart is operating in the intervals [1,6] and [23,26]

b) TCP congestion avoidance is operating in the intervals [7,16]（此处如果写为[6,16]也算对

c) After the 16th transmission round, packet loss is recognized by a triple duplicate ACK. If there was a timeout, the congestion window size would have dropped to 1.

d) After the 22nd transmission round, segment loss is detected due to timeout, and hence the congestion window size is set to 1.

e) The threshold is initially 32(这里简单地认为是 32，实际上也可能是小于 32 大于 16 的一个值), since it is at this window size that slow start stops and congestion avoidance begins.

f) The threshold is set to half the value of the congestion window when packet loss is detected. When loss is detected during transmission round 16, the congestion windows size is 42. Hence the threshold is 21 during the 18th transmission round.

g) The threshold is set to half the value of the congestion window when packet loss is detected. When loss is detected during transmission round 22, the congestion windows size is 29. Hence the threshold is 14 (taking lower floor of 14.5) during the 24th transmission round.（注意：只要曲线往下落，ssthresh 都减半）

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

During the 1st transmission round, packet 1 is sent; packet 2-3 are sent in the 2nd transmission round; packets 4-7 are sent in the 3rd transmission round; packets 8-15 are sent in the 4th transmission round; packets 16-31 are sent in the 5th transmission round; packets 32-63 are sent in the 6th transmission round; packets 64 – 96 are sent in the 7th transmission round. Thus packet 70 is sent in the 7th transmission round.

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

The threshold will be set to half the current value of the congestion window (8) when the loss occurred and congestion window will be set to the new threshold value + 3 MSS . Thus the new values of the threshold and window will be 4 and 7 respectively.

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

threshold is 21, and the 17th round congestion window size is 1， the19th round congestion window size is 4.

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

如果是用 Tahoe 算法，在 22nd 轮次发生 timeout，那么不存在快速恢复状态（注意：Tahoe 没有快速恢复机制，在丢包后，它不仅重发了一些已经成功传输的数据，而且在恢复期间吞吐量也不高），之前就应该是慢启动状态，那么

round 17:1 packet; round 18: 2 packets; round 19:4 packets; round 20:8 packets; round 21:16 packets; round 22: 21 packets(为什么是 21 呢？因为这时 ssthresh 是 21，第 22 轮次，本应该把窗口设置为 32，但是超出了 threshhold，所以设置为 21). So, the total number is 52.（所以，到 22nd 为止，总共发出来 52 个 packets）

注意：还有一种答案，有同学写 round 22：cwnd 已经增到 32，这个答案不太妥当，因为按照算法：

```
cwnd = cwnd + MSS,
    If (cwnd > ssthresh)
        set state to  "Congestion  Avoidance"
```

每收到一个 ack，就会增加一个 MSS，接着就判断是否大于 ssthresh，大于就进入 CA 状态。当然，具体拥塞控制算法是如何实现的，还依赖于代码，目前看国际上知名大学对此题目的解法确实有两种，而且书的作者给出的题目中也有超出 ssthreshold 的例子。综合看来，个人认为既然是 threshold，就不要让 cwnd 的实际值超出了。总之，同学们理解就可以了。

附：慢启动描述：

This process results in a doubling of the sending rate every RTT. Thus, the TCP send rate starts slow but grows exponentially during the slow start phase.

But when should this exponential growth end? Slow start provides several answers to this question.

First, if there is a loss event (i.e., congestion) indicated by a timeout, the TCP sender sets the value of cwnd to 1 and begins the slow start process anew. It also sets the value of a second state variable, ssthresh (shorthand for "slow start threshold") to cwnd/2 — half of the value of the congestion window value when congestion was detected.

The second way in which slow start may end is directly tied to the value of ssthresh. Since ssthresh is half the value of cwnd when congestion was last detected, it might be a bit reckless to keep doubling cwnd when it reaches or surpasses the value of ssthresh. Thus, when the value of cwnd equals ssthresh, slow start ends and TCP transitions into congestion avoidance mode. TCP increases cwnd more cautiously when in congestion-avoidance mode.

The final way in which slow start can end is if three duplicate ACKs are detected, in which case

TCP performs a fast retransmit and enters the fast recovery state.