Homework 4

Suyi Liu

February 25 2017

1 Problem 1

Forwarding involves the transfer of a packet from an incoming link to an outgoing link within a single router. Routing involves all of a network's routers, whose collective interactions via routing protocols determine the paths that packets take on their trips from source to destination node.

2 Problem 2

With a shadow copy, forwarding decisions in high speed routers can be made locally, at each input port, without invoking the centralized routing processor on a per-packet basis and thus avoiding a centralized processing bottleneck

3 Problem 3

If the switch fabric is not fast enough (relative to the input line speeds) to transfer all arriving packets through the fabric without delay, then packet queuing can also occur at the input ports, as packets must join input port queues to wait their turn to be transferred through the switching fabric to the output port. If the queue gets larger and overflows the buffer, packet loss will occur.
Packet loss can be eliminated if the switching fabric is fast enough(n times as fast as the input line speed, where N is the number of input ports). Suppose all N input lines are receiving packets, and all packets are to be forwarded to the same output port, each batch of N packets (one packet per input port) can be cleared through the switch fabric before the next batch arrives.

4 Problem 4

If a single outgoing link cannot transmit packets faster than the speed N packets arrive at the outgoing port in a single amount of time, the packets will get queued. When the queue gets larger and larger and overflows the buffer, packet loss will occur.
The packet loss cannot be prevented by increasing the switch fabric speed.

5 Problem 5

IP address: 192.168.0.1
network mask: 255.255.255.0
default router: 192.168.0.1
IP address of its local DNS server: 192.168.0.1

6 Problem 6

$(20 + 20)/(90 + 20 + 20) = 30.77\%$ 30.77 percent overhead
$1 - 30.77\% = 69.23\%$ 69.23 percent application data

7 Problem 7

Nothing will be affected since port pools for TCP and UDP protocols are independent.

8 Problem 8

In a VC network:
Incoming interface: the incoming packets' link's interface number
Incoming VC number: the VC number associated with such incoming link(through which packets are coming in)
Outgoing interface: the outgoing packets' link's interface number
Outgoing VC number: the VC number associated with such outgoing link(through which packets are forwarded out)

In a datagram network:
Destination address: the address where the packet is being forwarded to
Link interface: the corresponding interface number of the link to which it can forward the packet to its destination.

9 Problem 9

range:11000000 - 11011111 interface:0 number of addresses:32
range:10000000 - 10111111 interface:1 number of addresses:64
range:11100000 - 11111111 interface:2 number of addresses:32
range:00000000 - 01111111 interface:3 number of addresses:128

10 Problem 10

One IP address that can be assigned to this network: 128.220.247.64
Prefixes for the subnets: 128.220.247.64/28, 128.220.247.80/28, 128.220.247.96/28,

128.220.247.112/28

11 Problem 11

1500 - 20 = 1480 bytes
2400 - 20 = 2380 bytes
2380 / 1480 = 1.6 So 2 fragments generated

Fragment 1:
Fragment size = 1500 bytes
ID = 299
Data size: 1480 bytes
IP header size = 20 bytes
Offset = 0
Flag = 1

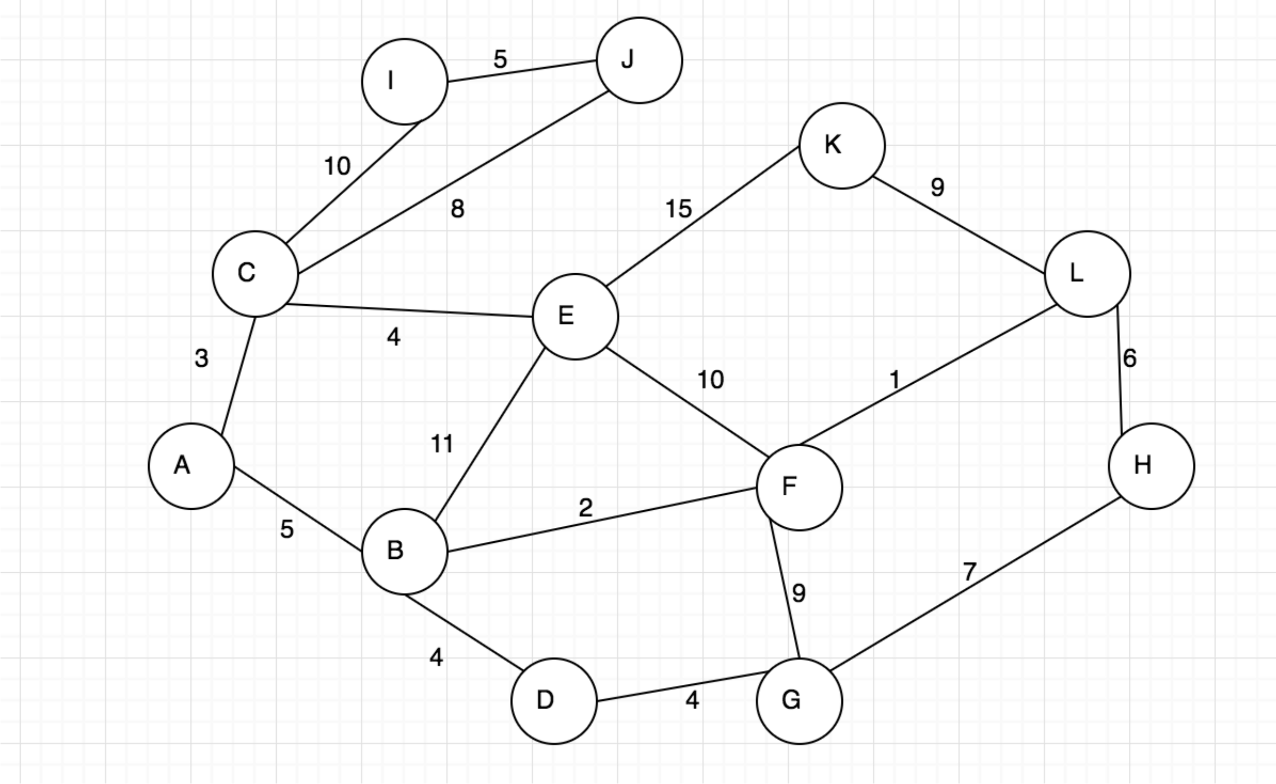
Fragment 2:
Fragment size = 920 bytes
ID = 299
Data size: 900 bytes
IP header size = 20 bytes
Offset = 185
Flag = 0

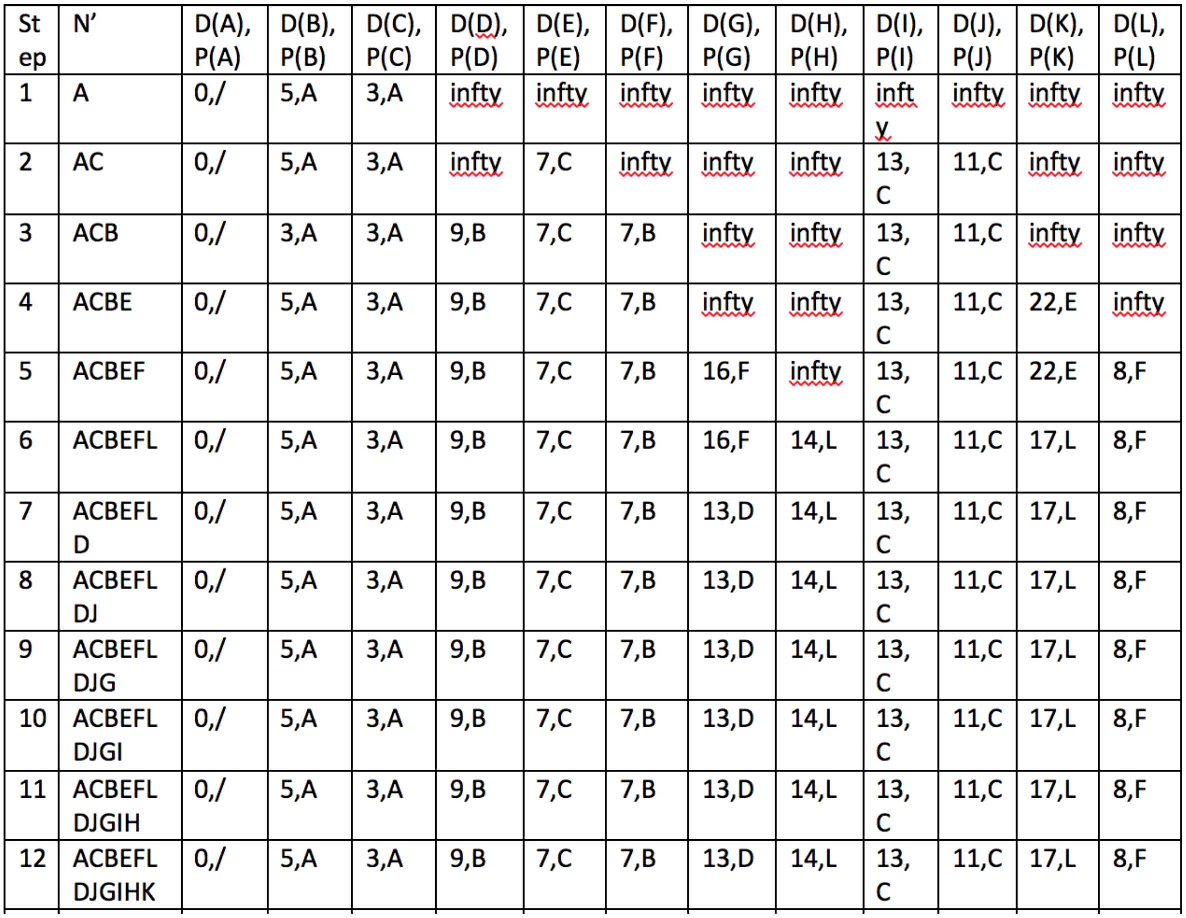
12 Problem 12

1500 - 20 = 1480 bytes of contents per datagram
300000/1480 = 202.7 = 203 datagrams

13 Problem 13

Want to find shortest path from A to L:





Step	N’	D(A), P(A)	D(B), P(B)	D(C), P(C)	D(D), P(D)	D(E), P(E)	D(F), P(F)	D(G), P(G)	D(H), P(H)	D(I), P(I)	D(J), P(J)	D(K), P(K)	D(L), P(L)
1	A	0,/	5,A	3,A	infty	infty	infty	infty	infty	infty	infty	infty	infty
2	AC	0,/	5,A	3,A	infty	7,C	infty	infty	infty	13,C	11,C	infty	infty
3	ACB	0,/	3,A	3,A	9,B	7,C	7,B	infty	infty	13,C	11,C	infty	infty
4	ACBE	0,/	5,A	3,A	9,B	7,C	7,B	infty	infty	13,C	11,C	22,E	infty
5	ACBEF	0,/	5,A	3,A	9,B	7,C	7,B	16,F	infty	13,C	11,C	22,E	8,F
6	ACBEFL	0,/	5,A	3,A	9,B	7,C	7,B	16,F	14,L	13,C	11,C	17,L	8,F
7	ACBEFLD	0,/	5,A	3,A	9,B	7,C	7,B	13,D	14,L	13,C	11,C	17,L	8,F
8	ACBEFLDJ	0,/	5,A	3,A	9,B	7,C	7,B	13,D	14,L	13,C	11,C	17,L	8,F
9	ACBEFLDJG	0,/	5,A	3,A	9,B	7,C	7,B	13,D	14,L	13,C	11,C	17,L	8,F
10	ACBEFLDJGI	0,/	5,A	3,A	9,B	7,C	7,B	13,D	14,L	13,C	11,C	17,L	8,F
11	ACBEFLDJGIH	0,/	5,A	3,A	9,B	7,C	7,B	13,D	14,L	13,C	11,C	17,L	8,F
12	ACBEFLDJGIHK	0,/	5,A	3,A	9,B	7,C	7,B	13,D	14,L	13,C	11,C	17,L	8,F

As shown in the graph and chart, the shortest path from A to L is 8, A-> B -> F -> L

14 Problem 14

- Yes. We can group IP packets whose IDs are sequential, and then count how many groups are there. The number of groups represent the number of unique hosts behind the NAT. Since identification number are randomly assigned, it is unlikely to be a consecutive number of the packet sent from another host.
- No, it won’t work since we cannot group IDs with consecutive numbers then(cannot distinguish packets from different unique hosts behind the NAT).

15 Problem 15

- We can ask ISP routers to watch if a packet is suspicious and store this flag into a separate field. It is normal if an outgoing packet is sent from internal IP address to outside world, or if an incoming packet with external ip address is sent from the outside to inside ISP. So the flag is 0 in such case. If some outgoing packet with external source IP address is sent from inside, it can be marked suspicious, set flag to 1 and get reported by ISP router. We can use this method to track down and find the host.

- We can insert the information into options field in IP header, since options field allows IP header to be extended.

- From the attacked destination host, we can use breath first search to disfunction nearby routers. If when a rounter is disfunctioned, our destination is not being attacked anymore, it means this router is along the path where the attack comes from. So we can do this recursively each time we find a "bad" router and then get to the source host.