Assignment 3

EECS325 Spring 2019
Yue Shu
Due: Thursday, April 11, 2019

1. Three types of switching fabric are discussed in Section 4.2. List and briefly describe each type. Which, if any, can send multiple packets across the fabric in parallel?

- **switching via memory**
  - traditional computers with switching under direct control of CPU
  - datagrams are copied to system's memory
  - switching rate is limited by memory bandwidth (2 bus crossings per datagram)
- **switching via a bus**
  - switch datagram from input port memory to output port memory via a shared bus
  - all output ports will receive the datagram, but only the correct one will keep it
  - switching rate is limited by bus bandwidth
- **switching via interconnection network (crossbar)**
  - switch datagrams via 2N buses that connect N input ports to N output ports
  - each verticle bus intersects with each horizontal bus at an intersection controlled by the fabric controller
  - fragment datagram into cells of fixed length and switch these cells through the fabric
  - datagrams to be forwarded will not be blocked by datagrams being forwarded to the other ports
  - overcome bus bandwidth limitations
- only **switching via interconnction network** can send multiple datagrams across the fabric in parallel

2. What is HOL blocking? Does it occur in input ports or output ports?

HOL (Head of the Line) blocking describes the situation when queued datagram at front of the queue prevents the following datagrams in the queue from moving forward.
HOL occurs in input ports.

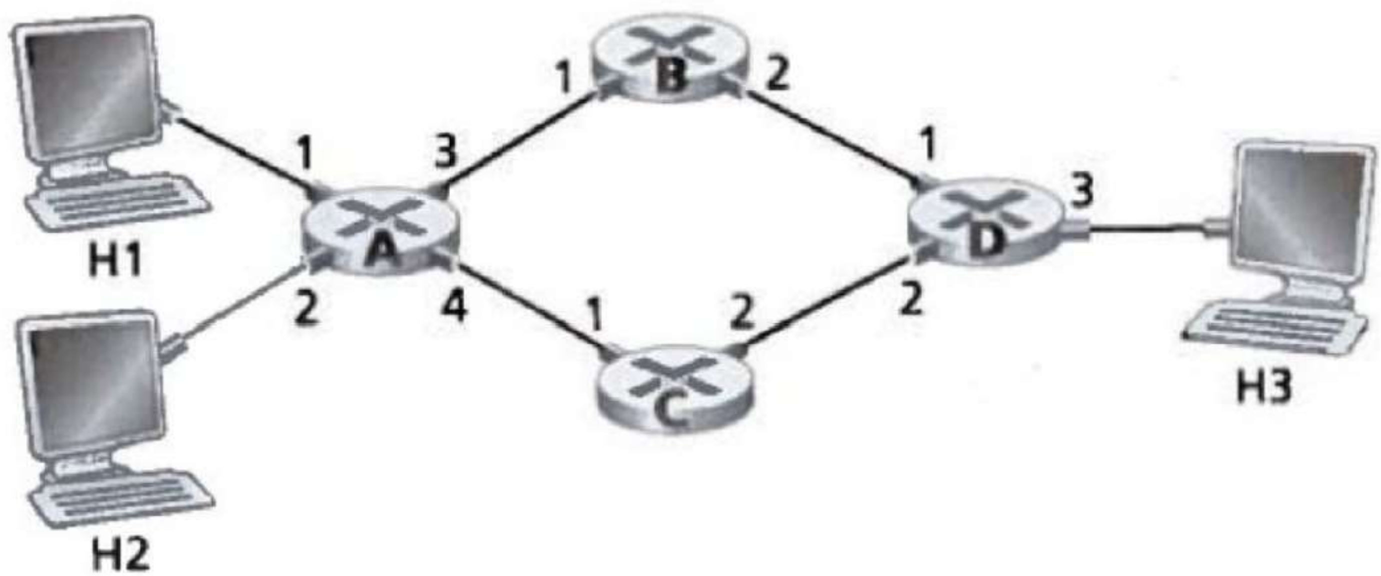
3. What fields in the IP header can be used to ensure that a packet is forwarded through no more than N routers?

The `time to live` field. It keeps track of the maximum number of remaining hops and decrements the value at each router.

4. When a large datagram is fragmented into multiple smaller datagrams, where are these smaller datagrams reassembled into a single larger datagram?

These smaller datagrams are only reassembled into a single larger datagram at their **final destination**, before it is passed to and processed by the transport layer.

5. Consider the network below:



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

Destination Address	Link Interface
H3	3

b. Can you write down a forwarding table in router A, such that all traffic from H1 destined to host H3 is forwarded through interface 3, while all traffic from H2 destined to host H3 is forwarded through interface 4?

No, it's not possible. Since the link interface in such forwarding table is only determined by the destination address, it's impossible to defferentiate traffic from H1 to H3 with traffic from H2 to H3.

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

Destination Address Range	Link Interface
11100000 00000000 00000000 00000000 through 11100000 00111111 11111111 11111111	0
11100000 01000000 00000000 00000000 through 11100000 01000000 11111111 11111111	1
11100000 01000001 00000000 00000000 through 11100001 01111111 11111111 11111111	2
otherwise	3

a. Provide a forwarding table that has five entries, uses longest prefix matching, and forwards packets to the correct link interfaces.

Destination Address Range	Link Interface
11100000 00****** ******** ********	0
11100000 01000000 ******** ********	1
1110000* ******** ******** ********	2
11100001 1******* ******** ********	3
otherwise	3

b. Describe how your forwarding table determines the appropriate link interface for datagrams with destination addresses:

11001000 10010001 01010001 01010101
11100001 01000000 11000011 00111100
11100001 10000000 00010001 01110111

1. The longest prefix of `11001000 10010001 01010001 01010101` matches none of the first four prefix entries from the forwarding table above, therefore should be matched to the `otherwise` entry and forwarded to link interface `3`, which confirms with the destination provided by the original forwarding table.
2. The longest prefix of `11100001 01000000 11000011 00111100` matches the `1110000` prefix entry from the forwarding table above, therefore should be forwarded to link interface 2, which confirms with the destination provided by the original forwarding table.
3. The longest prefix of `11100001 10000000 00010001 01110111` matches the `11100001 1` prefix entry from the forwarding table above, therefore should be forwarded to link interface 3, which confirms with the destination provided by the original forwarding table.

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

According to the problem, it should be sufficient to provide 64 addresses for Subnet 1, 128 addresses for Subnet 2, and 16 addresses for Subnet 3. Since all of the interfaces in each of these three subnets are required to have the prefix `223.1.17/24`, the addresses should be ranging from `223.1.17.0` to `223.1.17.255`.

To make our life easier, we shall start with the subnet that requires the most addresses, so as to have addresses with longest prefix accordingly for each subnet.

- **Subnet 2**:
  - we need 128 addresses, least significant number of which ranging from 0 to 127
  - the range is satisfied by `223.1.17.0` (`223.1.17.00000000`) to `223.1.17.127` (`223.1.17.01111111`)
  - the address can be represented as `223.1.17.0/25`
- **Subenet 1**:
  - we need 64 addresses, least significant number of which ranging from 0 to 63
  - the range is satisfied by `223.1.17.128` (`223.1.17.10000000`) to `223.1.17.191` (`223.1.17.10111111`)
  - the address can be represented as `223.1.17.128/26`
- **Subnet 3**
  - we need 16 addresses, least significant number of which ranging from 0 to 15
  - the range is satisfied by `223.1.17.192` (`223.1.17.11000000`) to `223.1.17.207` (`223.1.17.11001111`)
  - the address can be represented as `223.1.17.192/28`

In conclusion, we have network address `223.1.17.128/26` for Subnet 1, `223.1.17.0/25` for Subnet 2, and `223.1.17.192/28` for Subnet 3.

8. Consider sending a 2100-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

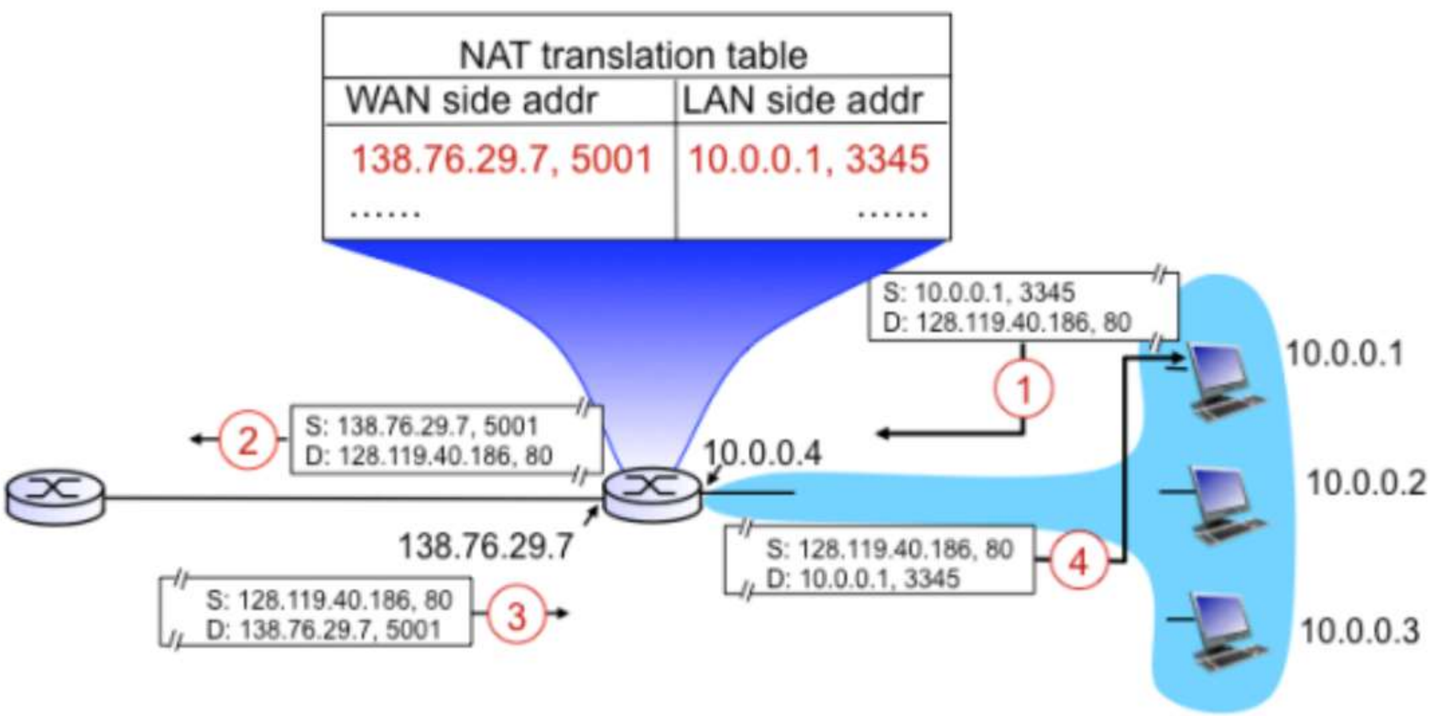
the values in the various fields in the IP datagram(s) generated related to fragmentation?

Given $MTU = 700$ byte, we may conclude that the maximum length of the data field is $700 - 20 = 680$ byte since we have 20 bytes for the IP header. Therefore, a total of $\lceil \frac{2100-20}{680} \rceil = 4$ fragments are generated.

length	16-bit identifier	flgs	fragment offset
700	422	1	0
700	422	1	85
700	422	1	170
60	422	0	255

To double check our final result, from the 4 datagrams above, we have a total of $700 \times 3 + 60 = 2160$-byte of data where $680 \times 3 + 40 = 2080$ bytes are the actual data, which confirms with the original requirement $2100 - 20 = 2080$-byte.

9. Consider the network setup below. Suppose that the ISP instead assigns the router the address 24.34.112.236 and that the network address of the home network is 192.168.1/24.



a. Assign addresses to all interfaces in the home network

Router address: `192.168.1.4`

Home addresses: `192.168.1.1`, `192.168.1.2`, `192.168.1.3`

b. Suppose each host has two ongoing TCP connections, all to port 80 at host 128.119.41.85. Provide the six corresponding entries in the NAT translation table.

WAN side addr	LAN side addr
24.34.112.236, 5001	192.168.1.1, 3345
24.34.112.236, 5002	192.168.1.1, 3346
24.34.112.236, 5003	192.168.1.2, 3345
24.34.112.236, 5004	192.168.1.2, 3346
24.34.112.236, 5005	192.168.1.3, 3345
24.34.112.236, 5006	192.168.1.3, 3346

10. What is the difference between a forward table in the destination-based forwarding routers and OpenFlow's flow table that we discussed in Section 4.4?

- **destination-based forwarding table**
  - forward based only on destination IP address
  - each entry contains only a set of destination IP address ranges and a set of corresponding link interfaces that represent where the datagram will be forwarded to
- **OpenFlow flow table**
  - generalized forwarding
    - forward based on any set of header field values
    - determined not only by the destination IP address, but also by the source IP address as well as IP prot, TCP source port, TCP destination port, etc.
  - each entry contains
    1. Rule: a set of header field values representing how the incoming datagrams should be matched
    2. Action: a set of actions to be taken when a datagram matches the rule of a flow table entry
    3. Stats: a set of counters that are updated as datagrams are matched to the flow table entries