Henry Hoekel

HSHKCC 14207985

Homework 3

39) What kind of problems can arise when two hosts on the same Ethernet share the same hardware address? Describe what happens and why that behavior is a problem.

When two hosts share the same hardware address, data sent to a particular address would be sent to both nodes, when one of which is the wrong address. Another possible occurrence could be that one of the addresses sends data and the other same hardware address could see this data being sent from the same address as incoming data. This could lead to a “producer consumer,” type of collision where sent data is seen as incoming and the sent data would never go outward because it is consistently being seen as incoming data destined for a particular address.

51) Suppose that N Ethernet stations, all trying to send at the same time, require N/2 slot times to sort out who transmits next. Assuming the average packet size is 5 slot times, express the available bandwidth as a function of N.

1) Using the example network given in Figure 3.44, give the virtual circuit tables for all the switches after each of the following connections is established. Assume that the sequence of connections is cumulative; that is, the first connection is still up when the second connection is established, and so on. Also assume that the VCI assignment always picks the lowest unused VCI on each link, starting with 0, and that a VCI is consumed for both directions of a virtual circuit.

a) Host A connects to Host C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Switch | Input Port | Input VCI | Output Port | Output VCI |
| 1 | 2 | 0 | 3 | 0 |

b) Host D connects to Host B

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Switch | Input Port | Input VCI | Output Port | Output VCI |
| 1 | 0 | 0 | 1 | 0 |
| 2 | 3 | 0 | 0 | 0 |
| 3 | 0 | 0 | 3 | 0 |

c) Host D connects to Host I

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Switch | Input Port | Input VCI | Output Port | Output VCI |
| 1 | 0 | 1 | 1 | 1 |
| 2 | 3 | 1 | 0 | 1 |
| 3 | 0 | 1 | 2 | 0 |

d) Host A connects to Host B

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Switch | Input Port | Input VCI | Output Port | Output VCI |
| 1 | 2 | 1 | 1 | 2 |
| 2 | 3 | 2 | 0 | 2 |
| 3 | 0 | 2 | 3 | 1 |

e) Host F connects to Host J

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Switch | Input Port | Input VCI | Output Port | Output VCI |
| 4 | 2 | 0 | 3 | 0 |
| 2 | 1 | 0 | 0 | 3 |
| 3 | 0 | 3 | 1 | 0 |

f) Host H connects to Host A

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Switch | Input Port | Input VCI | Output Port | Output VCI |
| 4 | 0 | 0 | 3 | 1 |
| 2 | 1 | 1 | 3 | 3 |
| 1 | 1 | 3 | 2 | 2 |

**3.** For the network given in Figure 3.45, give the datagram forwarding table for each node. The links are labeled with relative costs; your tables should forward each packet via the lowest-cost path to its destination.

Node A:

|  |  |
| --- | --- |
| Destination | Next Hop |
| B | C |
| C | C |
| D | C |
| E | C |
| F | C |

Node B:

|  |  |
| --- | --- |
| Destination | Next Hop |
| A | E |
| C | E |
| D | E |
| E | E |
| F | E |

Node C:

|  |  |
| --- | --- |
| Destination | Next Hop |
| A | A |
| B | E |
| D | E |
| E | E |
| F | F |

Node D:

|  |  |
| --- | --- |
| Destination | Next Hop |
| A | E |
| B | E |
| C | E |
| E | E |
| F | E |

Node E:

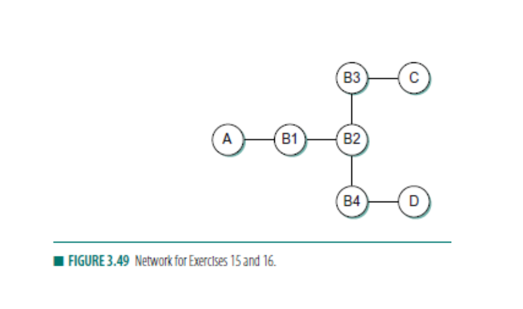
|  |  |
| --- | --- |
| Destination | Next Hop |
| A | C |
| B | B |
| C | C |
| D | D |
| F | F |

Node F:

|  |  |
| --- | --- |
| Destination | Next Hop |
| A | C |
| B | C |
| C | C |
| D | C |
| E | C |

**6.** In the source routing example of Section 3.1.3, the address received by B is not reversible and doesn’t help B know how to reach A. Propose a modification to the delivery mechanism that does allow for reversibility. Your mechanism should *not* require giving all switches globally unique names.

We can create a modification that adds the return address in the packet header. Every time a packet hops to a switch, the switch adds the input port number to the return address list to allow for reversibility. Since every hop has an input and a return address, once the packet reaches its destination, the reverse could be calculated by following the return addresses.

**15.** Consider the arrangement of learning bridges shown in Figure 3.49. Assuming all are initially empty, give the forwarding tables for each of the bridges B1 to B4 after the following transmissions:

* A sends to C.
* C sends to A.
* D sends to C.

Identify ports with the unique neighbor reached directly from that port; that is, the ports for B1 are to be labeled “A” and “B2.”

After A sends to C and C sends to A, A and C learn the locations of each other, however B4 doesn’t learn where C or A is. When D sends to C, B1 also doesn’t learn where D is.

B1: A- Interface: A

B2-Interface: C (NOT D)

B2: B1-Interface: A

B3-Interface: C

B4-Interface: D

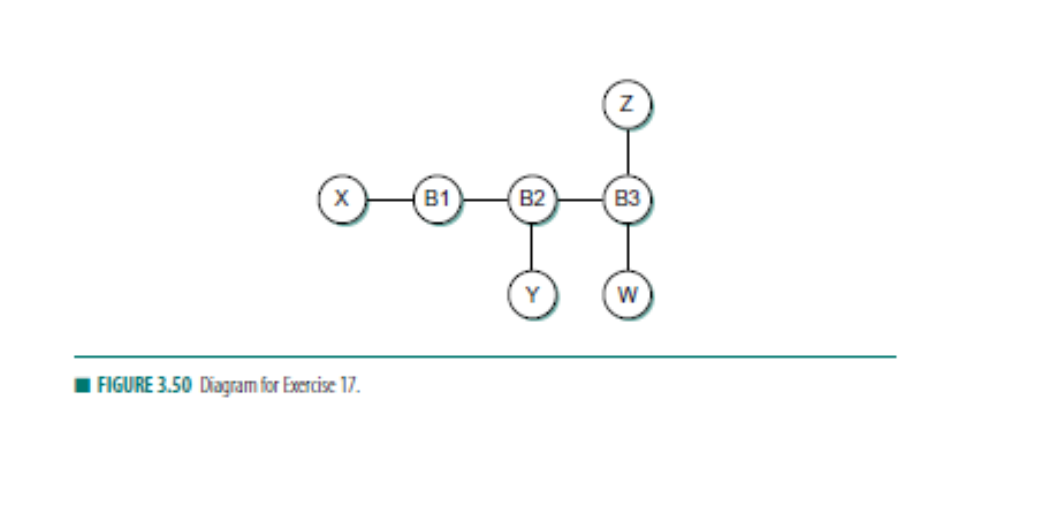
B3: B2-Interface: A, D

C-Interface: C

B4: B2-Interface: A (NOT C)

D-Interface: D

* **17.** Consider hosts X, Y, Z,W and learning bridges B1, B2, B3, with initially empty forwarding tables, as in Figure 3.50.

****

* 1. **(a) Suppose X sends to W. Which bridges learn where X is? Does Y’s network interface see this packet?**

All bridges learn where X is because all links see the packet being forwarded. B1, B2, and B3 all see packets from X to W. Y’s Interface would see this packet because B2 saw the packet

* 1. **(b) Suppose Z now sends to X. Which bridges learn where Z is? Does Y’s network interface see this packet?**

All bridges know where X is because all links see the packet. The packet traverses B3 to B2 to B1 and all links know where X is. Since all of these links see the packet, they also know where Z is. Y’s interface would not see the packet because B2 only sees B1.

* 1. **(c) Suppose Y now sends to X. Which bridges learn where Y is? Does Z’s network interface see this packet?**

Y sends to X through B2 to B1. B2 and B1 know where Y and X are, however B3 and Z never see the packet. Since Z’s interface link B3 doesn’t see the packet, Z would not be able to see it as well.

* 1. **(d) Finally, suppose W sends to Y. Which bridges learn where W is? Does Z’s network interface see this packet?**

W sends to Y through B3 to B2. B3 doesn’t know where Y is, so it would have to retransmit the message throughout all the links. Z’s network would see the packet because of this. When the message is resent, it arrives at B2 and it would then be forwarded to Y. B1 doesn’t see this message. B3 and B2 know where W is, but B1 does not.