Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich
Spring Term 2014

Operating Systems and Networks
Assignment 6

Assigned on: **27th Mar 2014**
Due by: **4th Apr 2014**

1 CRC

Suppose we want to transmit the message 11001001 ($P(x) = x^7 + X^6 + x^3 + 1$) and protect it from errors using the CRC polynomial $C(x) = x^3 + 1$.

a) *Use polynomial long division to determine the message that should be transmitted.*

Answer: We take the message 1100 1001, append 000 to it, and divide by 1001. The quotient is 1101 0011 and the remainder is 011. Therefore, what we transmit is the original message with the remainder appended: 1100 1001 011.

b) *Suppose the leftmost bit of the message is inverted due to noise on the transmission link. What is the result of the receiver's CRC calculation? How does the receiver know that an error has occurred?*

Answer: Inverting the first bit gives 0100 1001 011; dividing by 1001 ($x^3 + 1$) gives a quotient of 0100 0001 and a remainder of 10. Since the receiver will obtain a nonzero remainder implying that an error has occurred.

c) *Give an example of the error which can not be detected by CRC. Is there a general pattern for undetectable errors?*

Answer:

One example: 10000001011.

The errors are of the general form $E(x) = (x^3 + 1) * \sum_{0 \leq i \leq k} x^i$, here, the is are different.

We know from the text book (Page 97) that "If $P(x)$ is transmitted over a link and there are no errors introduced during transmission, then the receiver should be able to divide $P(x)$ by $C(x)$ exactly, leaving a remainder of zero." Since $P(x)$ as well as $E(x)$ are divisible by $C(x)$, the errors can't be detected by the receiver:

```
  11001001011
+  1001
---------------
  10000001011      can't be detected
+        1001
---------------
  10000000010      can't be detected
+ 1001
---------------
  00010000010      can't be detected
```

2 Spanning Tree Algorithm

a) *Given the network shown in Figure* **??**, *where the letters A to J represent LANs and the circles B1 to B7 represent a switch node. Indicate which ports are not selected by the spanning tree algorithm.*

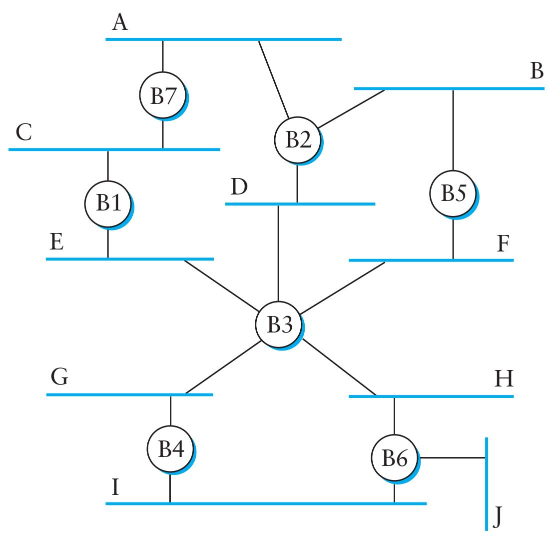

Figure 1: Network for spanning tree algorithm

Answer: We depict the collection of switches in a local area network (LAN) as a graph whose nodes are switches and LAN segments (or cables), and whose edges are the interfaces connecting the switches to the segments.

The spanning tree algorithm works as follows [1]:

- Elect a single switche, among all the switches on all the LANs, to be the *Root Bridge*.
- Calculate the distance of the shortest path from the switches to the Root Bridge.
- For each LAN, elect *Designated Bridge* from among the switches residing on that LAN. The elected switche is the one closest to the Root Bridge. The Designated Bridge will forward packets from that LAN toward the Root Bridge.
- Choose a port (known as the *root port*) that gives the best path from each Designated Bridge to the Root Bridge.
- Select ports to be included in the spanning tree. The port selected will be the root port plus any ports on which 'self' has been elected Designated Bridge.

The ports which are not selected by the spanning tree algorithm are shown as red dotted lines in Figure **??**.

b) *Given the network shown in Figure* **??**, *assume that switch B1 suffers catastrophic failure. Indicate which ports are not selected by the spanning tree algorithm after the recovery process and a new tree has been formed.*

Answer: After the recovery process, the new formed spanning tree is shown in Figure **??**. The ports which are not selected by the spanning tree algorithm are shown as red dotted lines.

[1] Radia Perlman. 1999. Interconnections (2nd Ed.): Bridges, Routers, Switches, and Internetworking Protocols. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.

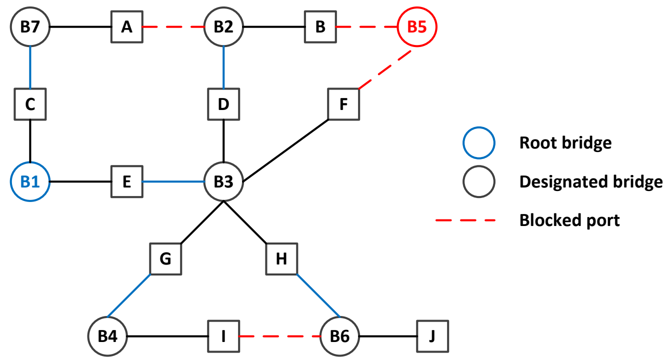

Figure 2: Spanning tree a)

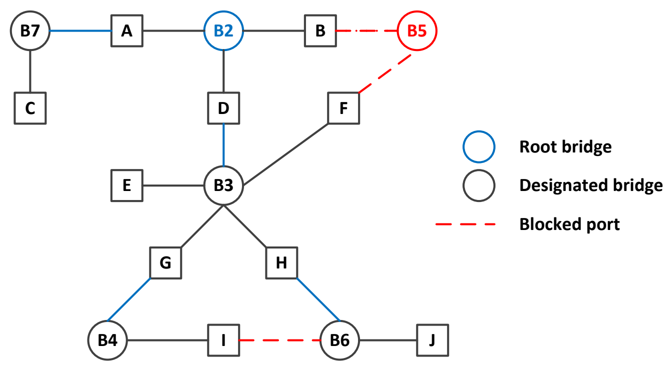

Figure 3: Spanning tree b)

3 Switches

a) *Store-and-forward switches have an advantage over cut-through switches with respect to damaged frames. Explain what it is.*

Answer: 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.

b) *What is the big benefit of using switches to connect hosts?*

Answer: A switch is basically a multi-port equipment. Every host has a private, non-shared link to the switch ("network segment" with one single host connected to it). Therefore there are no collisions at all on these links which is a big benefit over shared media.

4 IP Checksum

Here is a IP header from an IP packet received at destination :

4500 003c 1c46 4000 4006 b1e6 ac10 0a63 ac10 0a0c

Please compute the checksum for the header.

Answer:

We convert all these values in binary :

4500 -¿ 0100010100000000 003c -¿ 0000000000111100 1c46 -¿ 0001110001000110 4000 -¿ 0100000000000000 4006 -¿ 0100000000000110 0000 -¿ 0000000000000000 // Note that the checksum is set to zero since we are computing checksum at destination end ac10 -¿ 1010110000010000 0a63 -¿ 0000101001100011 ac10 -¿ 1010110000010000 0a0c -¿ 0000101000001100

Now we add these binary values one by one :

4500 -¿ 0100010100000000 003c -¿ 0000000000111100 453C -¿ 0100010100111100 /// First result

453C -¿ 0100010100111100 // First result plus next 16-bit word. 1c46 -¿ 0001110001000110 6182 -¿ 0110000110000010 // Second result.

6182 -¿ 0110000110000010 // Second result plus next 16-bit word. 4000 -¿ 0100000000000000 A182 -¿ 1010000110000010 // Third result.

A182 -¿ 1010000110000010 // Third result plus next 16-bit word. 4006 -¿ 0100000000000110 E188 -¿ 1110000110001000 // Fourth result.

E188 -¿ 1110000110001000 // Fourth result plus next 16-bit word. AC10 -¿ 1010110000010000 18D98 -¿ 11000110110011000 // One odd bit (carry), add that odd bit to the result as we need to keep the checksum in 16 bits.

18D98 -¿ 11000110110011000 8D99 -¿ 1000110110011001 // Fifth result

8D99 -¿ 1000110110011001 // Fifth result plus next 16-bit word. 0A63 -¿ 0000101001100011 97FC -¿ 1001011111111100 // Sixth result

97FC -¿ 1001011111111100 // Sixth result plus next 16-bit word. AC10 -¿ 1010110000010000 1440C -¿ 10100010000001100 // Again a carry, so we add it (as done before)

1440C -¿ 10100010000001100 440D -¿ 0100010000001101 // This is seventh result

440D -¿ 0100010000001101 //Seventh result plus next 16-bit word 0A0C -¿ 0000101000001100 4E19 -¿ 0100111000011001 // Final result.

So now 0100111000011001 is our final result of summing up all the 16 bit words in the header. As a last step we just need to do a ones compliment of it to obtain the checksum.

4E19 -¿ 0100111000011001 B1E6 -¿1011000111100110 // CHECKSUM

5 Network Tool

5.1 Traceroute

a) Run a traceroute to www.netsec.ethz.ch. Record the output. Describe what is strange about the observed output, and why traceroute gives you such an output. Refer to the traceroute man page for useful hints.

b) Explain how traceroute discovers a path to a remote host. The man page might be useful in answering this question.