ECE 461– Internetworking
Fall 2011

Quiz 2 - Solutions

Instructions (read carefully):

- The time for this quiz is 50 minutes.
- This is a closed book and closed notes in-class exam.
- No calculators may be used.
- Write your answers on the pages provided, using front and back if needed. Use extra sheets if needed.
- Do not give aid or receive aid from other students.
- Show all steps of your solutions.
- Make sure your answers are legible. If we cannot read an answer, we will not grade it.

	Max Points	
1	10	
2	5	
3	5	
4	10	
5	10	
6	10	
Total	55	

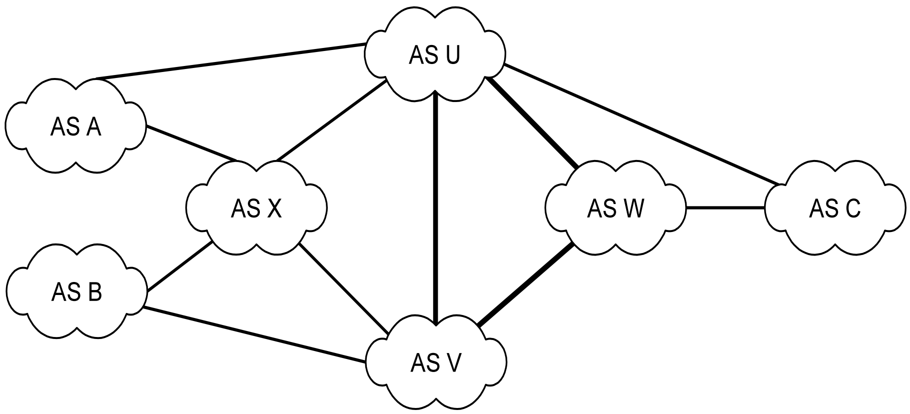
STAPLE EXTRA SHEETS WITH YOUR SOLUTIONS TO THIS SHEET BEFORE TURNING IN !

Name: ____________________

Email: ___________________

Problem 1. *(10 Points)*
Consider the network of autonomous systems in the figure, where AS A is a customer of AS X and AS U, and AS B is a customer of AS X and AS V. AS X, in turn, has two provider ASes U and V, while AS C is a customer AS of AS W and AS U. The three ASes U, V and W have peering relationships among themselves.

- AS C owns the prefix 128.1.0.0/16.
- AS A owns two network prefixes 64.1.10.0/24, and 128.101.34.0/24,



The inter-domain routing protocol BGP is used among the ASes to exchange routing information.

a. *(5 points)* For a packet from a host in AS A with the destination IP address 128.1.34.35, which ASes would this packet most likely traverse to reach its final destination? Explain your answer.

b. *(5 points)* Suppose AS A wants the traffic to its prefix 128.101.34.0/24 to come from AS U and the traffic to its prefix 64.1.10.0/24 to come from AS X. Which routes should AS A announce to AS U, and what routes should AS A announce to AS X? For each announcement, provide the prefix and the AS-PATH attribute. Explain your answers.

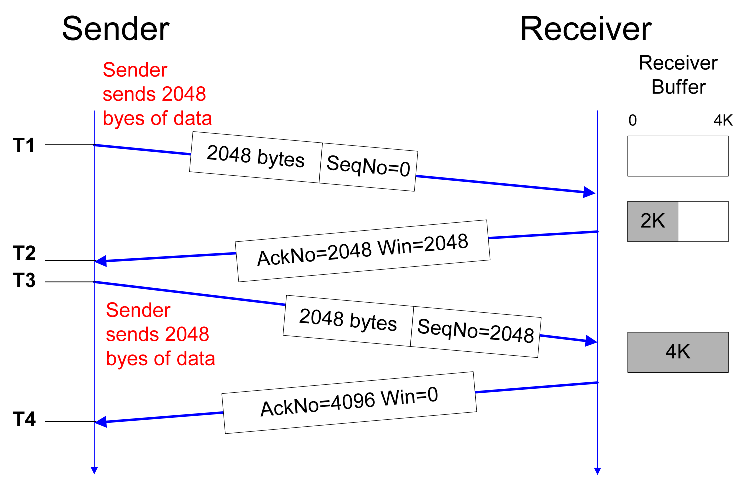
a) The IP address is in AS C.
Based on the customer/provider relationships both ASX and ASU will advertise the prefix to 128.1.0.0/16. Since the path A → U → C is shorter than A → X → U → C, the shorter path is selected.

b) The announcements should be:
A → U: 128.101.34.0/24, AS-PATH{AS A}
A → X: 64.1.10.0/24, AS-PATH{AS A}

Problem 2. *(5 Points)* How are the maximum frame size (at the Data Link Layer), the Maximum Transmission Unit in IP (MTU), and the Maximum Segment Size in TCP (MSS) related?

MSS = MTU – size of TCP header – size of IP header
MTU = Maximum frame size – size of MAC header

Problem 3. *(5 Points)*



Consider the transmission scenario of a TCP connection as shown in the figure. Initially, the advertised window (determined by the receiver buffer) is given by 4096 bytes. For simplicity we assume that the initial sequence number (ISN) of the sender is given by 0. The figure depicts the following scenario:

(a) **time T1:** sender transmits a segment with 2048 bytes of data, starting at sequence number 0
(b) **time T2**: ACK segment from receiver arrives at sender, which acknowledges receipt of 2048, but reduces the allowed window size to 2048 bytes.
(c) **time T3**: Sender transmits a segment with 2048 bytes of data, starting at sequence number 2048.
(d) **time T4:** ACK segment from receiver arrives at sender, acknowledging receipt of the 2048 bytes sent at time T3, and reducing the allowed window size to 0 bytes.

After time T4, the sender cannot send any new data, yet, the receiver has acknowledged all transmissions. So, neither the sender nor the receiver will transmit any segments. Describe rules that can be added to the TCP protocol to prevent sender from stalling indefinitely in this situation. Explain how the added rules resolve the problem.

Possible solutions:

- *Whenever the application reads data and buffer space becomes available in the receiver buffer, receiver must send an ACK where it advertises the window.*
  This method has the disadvantage that the receiver sends an ACK whenever data is passed from the TCP layer to the application.

- *The sender is allowed to transmit an empty header or a small segment when it has been blocked for a long time*
  This is the actual solution of TCP: The sender is allowed to send a 1-byte packet to enforce an ACK from the receiver which informs it the sender of the current window size.

Problem 4. *(10 points)*
Assume you modify the TCP implementation on your computer as follows:

Modification 1. After a retransmission timeout and subsequent retransmission, do not modify the values of the congestion window (*cwnd*) or the slow-start threshold value (*ssthresh*).

Modification 2. After a retransmission timeout and subsequent retransmission, do not apply Karn's algorithm.

Modification 3. Disable the round-trip time (RTT) measurements, i.e., always use the initial values of RTO.

Modification 4. Do not use Fast Retransmit after a third duplicate acknowledgement (instead only retransmit using timeouts).

Modification 5. Never leave the Slow-Start phase.

This question may be a bit difficult to grade, since there are several sensible answers. Below is a more extensive discussion than what is expected from students.

(a) *(5 points)* Describe the impact of each of the modifications on the maximum data rate at which your computer can transmit on a TCP connection. (Consider each modification individually).

Modification 1: Since the congestion window is never reduced, the congestion window will reach maximum size and stay there. As a result the data rate will increase.

Modification 2: Since the time between retransmissions is not increased, the modification leads to lower RTO values in case of retransmissions, and thus, more frequent RTO timeouts.

- Since for each RTO timeout, we set cwnd=1 and enter slow-start, we could expect that more frequent retransmissions reduce the data rate. (However, the impact should be small since this affects only repeated retransmissions).
- On the other hand, since earlier retransmissions lead to an earlier recovery of lost data, the sender can start earlier with sending new data (thus increase the data rate).

Modification 3: The initial RTO value is quite large. Large RTO means it takes a long time to do a retransmission. There are 2 effects:

- Since the sender enters the slow-start phase later (when the timeout occurs), the sender can transmit more data.
- Since the receiver does not acknowledge new data (until it receives the retransmission), the "available window" at the sender will eventually close, thus shutting off the receiver.

The second effect dominates. Thus, the overall data rate will be reduced.

Modification 4: This will cause longer waits for a retransmission, thus reducing the data rate. (If fast recovery is also disabled, then the sender will enter the slow start phase more often, which further reduces the data rate).

Modification 5: Never leaving the Slow-Start phase leads to a faster increase of cwnd (i.e., cwnd is doubled in each transmission of a full window size). This leads to an increase of the data rate

(b) *(5 points)* Describe the impact of each of the modification when all computers on the Internet are modified. (Again, consider each modification individually).

Modification 1: TCP senders no longer reduce their rate when the network is congested. This will lead to congestion in the Internet.

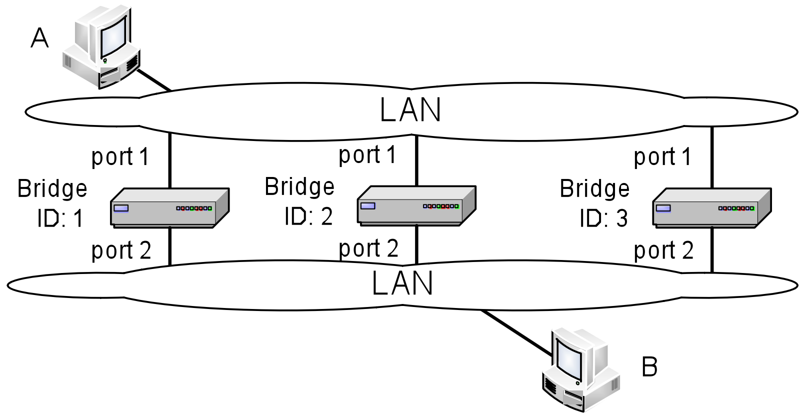
Modification 2: The Internet will see more retransmissions. (But the impact should be small since each sender reduces the congestion window for each retransmission).

Modification 3: Since the retransmission timeout no longer adapts to the actual dealys between sender and receiver (and RTO remains at a high value), many users of the Internet will experience long delays if a packet is lost. It will appear that TCP traffic will become very slow as soon as there are retransmissions.

Modification 4: If there are single packet errors (which are recovered by Fast Retransmit), TCP connections take longer to retransmit, since the sender must wait for a retransmission timeout.

Modification 5: When TCP connections ramp up their data rate quickly (as is the case in Slow Start), the Internet will experience more congestion. Since congestion leads to packet losses, and packet losses set cwnd=1, the data rate of TCP connections will oscillate between very low and very high data rates.

Problem 5. *(10 points)*
The figure below depicts a scenario where three bridges are connected to two LANs, and one host is connected to each LAN.



a) *(4 points)* Suppose that the SPT protocol is not enabled. Describe problems that may occur when host A transmits a frame to Host B.

Solution:
The described scenario should describe (1) a loop and (2) duplication of frames in each loop.

Suppose all bridges are started with empty tables. When A transmits a frame to B, each bridge will note that host A is reachable via its respective Port 1, and then queues the frame for transmission on the LAN of host B (LAN B). One of the bridges (say Bridge 2) will be the first to transmit the frame to LAN B. This frame also arrives on Ports 2 of Bridge 1 and Bridge 3. Both bridges receive the frame, not in their tables that Host A is now located on LAN 2, and queue the packet for forwarding on the LAN of Host A (LAN A).

Let us say that Bridge 1 is the next bridge that sends the frame on LAN B. Here, Bridges 2 and 3 see the frame, and note that Host A is still on LAN B. Both will then queue the frame for transmission on LAN A.

Let us assume that, next, Bridge 1 transmits the frame that is queued for LAN 1. When Bridge 2 and 3 see the frame they change their tables and associate Host A with LAN A. **…. And so forth**

So, there is a looping of packets. There is also a multiplication: each successful transmission results in an additional two packets being transmitted.

b) *(3 points)* Now suppose the SPT protocol is enabled. If the SPT protocol is activated at all bridges at the same time, what are the initial BPDUs transmitted by the bridges?

(w,x,y,z) – root is w, x is root path cost, y is sending bridge, z is sending port.

Bridge 1:
(1, 0, 1, 1) on LAN A
(1, 0, 1, 2) on LAN B

Bridge 2:
(1, 0, 2, 1) on LAN A
(1, 0, 2, 2) on LAN B
Bridge 3:

(1, 0, 3, 1) on LAN A
(1, 0, 3, 2) on LAN B

c) *(3 points)* After convergence of the SPT protocol, what are the BPDU's transmitted by the bridges? Indicate root ports, designated ports and blocked ports of all bridges.

Only bridge 1 transmits:
(1, 0, 1, 1) on LAN A
(1, 0, 1, 2) on LAN B

Bridge 1:
- *Port 1- designated*
- *Port 2- designated*

Bridge 2:
- *Port 1- root port*
- *Port 2- blocked*

Bridge 3:
- *Port 1- root port*
- *Port 2- blocked*

Problem 6. *(10 points)* **LAN switching**
The spanning tree algorithm for bridges (LAN switches) is known to adapt only slowly after a failure of a link or a bridge. The following is an improvement to the spanning tree protocol to accelerate the convergence of the STP protocol after a link or bridge failure:

Backbone Fast: Consider a Bridge A with Bridge R as root bridge. When Bridge A receives a BPDU from a designated bridge (say, Bridge X), which specifies Bridge S as root bridge, and S > R, Bridge A sends Root Link Query (RLQ) request messages to determine if there still exists a path from Bridge A to Bridge R. Bridge A sends an RLQ request to all ports, except the port leading to Bridge X and except all ports where Bridge A is the designated bridge. The purpose of sending RLQ requests is to determine if there still exists a path from Bridge A to Bridge R. The RLQ request includes the ID of root R. Any bridge that receives a RLQ request immediately answers with a *RLQ response* if (1) it knows it has lost connection to root R (i.e., it has a different root bridge), or if it is the root (i.e., it is Bridge R). If a bridge does not send a RLQ response, it forwards the RLQ request to its root port. The sender of the RLQ response puts the ID of its root bridge in the PDU. RLQ responses are flooded on designated ports. Once Bridge A receives RLQ responses for each transmitted RLQ request and no response specifies Bridge R as root bridge, Bridge A initiates a re-computation of the spanning tree.

The following questions relate to the Backbone Fast improvement:

a. *(3 Points)* Describe the valid conclusions that Bridge A can draw if it receives a BPDU from Bridge X, which specifies an inferior root bridge.

b. *(3 Points)* Explain why Bridge A does not send RLQ requests to ports leading to Bridge X and to ports where Bridge A is the designated bridge.

c. *(4 Points)* Describe the valid conclusions that Bridge A can draw if all RLQ responses specify a different root bridge. Describe how the re-computation of the spanning tree by Bridge A results in an improvement over the standard STP algorithm.

Overall:

- Description of Backbone Fast: is found at **http://www.cisco.com/en/US/tech/tk389/tk621/technologies_tech_note09186a00800c2548.shtml**

- It may be convenient to think about the RLQ as a "ping for the root".

a) Obtaining an inferior BPDU from the designated bridge X, means that X either has lost the root, or that X's cost of the path to the root has increased. (Note that the text above says that S>R. If this is the case, then X has lost the root.)

b) A does not send a RLQ requests on ports when no new information can be learned from a RLQ response from that port:

  - A does not send to X because A already knows that X has lost the root R (This information was contained in the RLQ request)

- A does not send a RLQ request to ports where it is designated bridge, since A sends its BPDU on these ports. Since A has R as root, all bridges that receive A's BPDU also have R as root. (So, A does not need to ask these bridges what their root bridge is).

c) – If no RLQ response specifies bridge R as the root, then A can conclude that there is no path left that leads to root R.
- The improvement brought by RLQ request/reply is that the bridge A realizes much faster that no root to R exists, and starts quickly with searching for a new bridge. Without RLQ request/reply, A would just reject messages from X that advertise S>R, until the stored value for root R (on the port to X) times out.