CS433/533: Computer Networks

Exam 1: Application, Transport, and Network

03/28/2005

6:30-8:00 pm

- This exam is closed book and closed notes. However, you may refer to a sheet of 8.5"x11" paper of your own design.
- ***Keep your answer concise.***
- Show your reasoning clearly. If your reasoning is correct, but your final answer is wrong, you will receive most of the credit. If you just show the answer without reasoning, and your answer is wrong, you may receive no points at all.

Name: ________________________________

Applications (25 points)	Transport (27 points)	Routing (23 points)	Total (75 points)

1. [25 points] **Applications**
In this problem, we will go over the major steps when an email client (e.g., pine or outlook) sends an email to xyz@cs.yale.edu.

a) [4 points] How does an email client discover the IP address of an email server for cs.yale.edu? Please include the protocol and the query parameters.

`DNS, type = MX, query = cs.yale.edu`

3pts – DNS protocol
1pt - parameters

b) [4 points] An alternative is to use a Napster-style directory server for address resolution. Please list at least one major disadvantage and one major advantage of this Napster-style system over that of a).

`Single point of failure; centralized administration (assignment) of domain name`

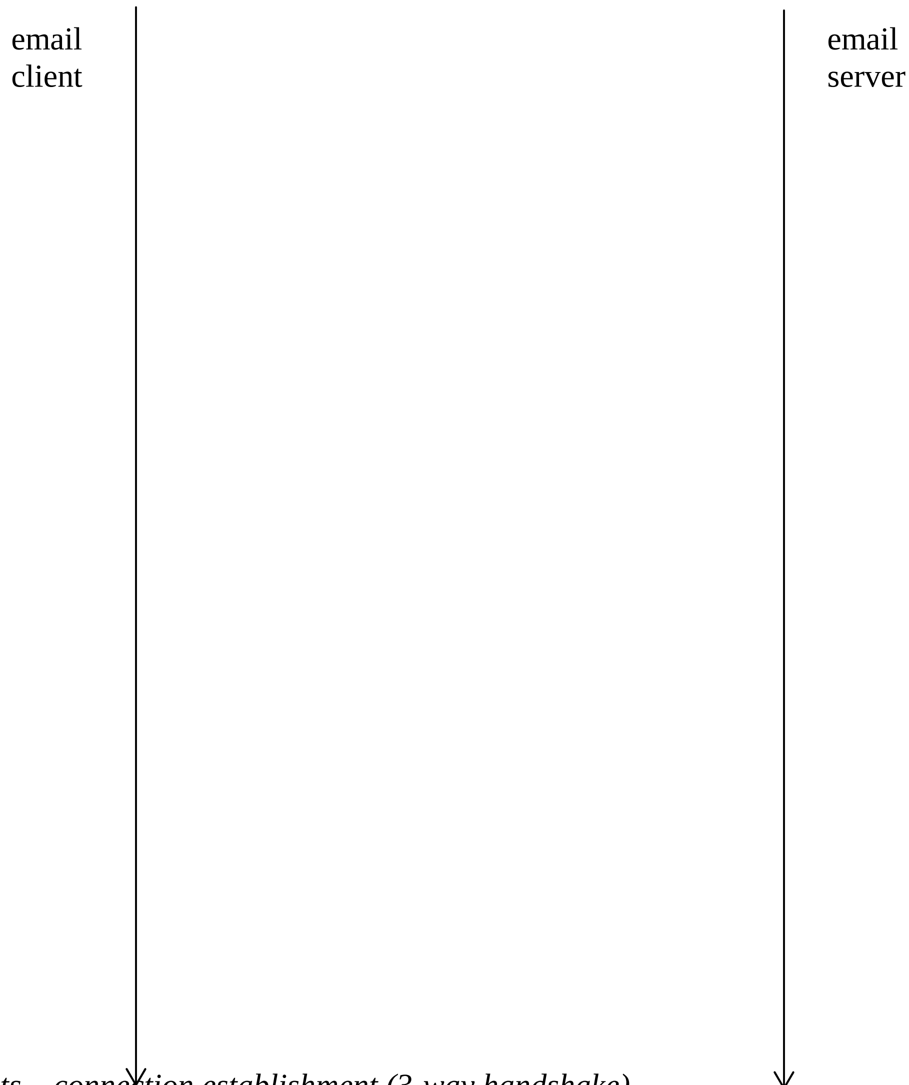
2pts – advantages
2pts - disadvantages

c) [4 points] Is it a good idea to build such a name resolution system using Freenet? If yes, please state the major benefits; if no, please state at least one major reason.

`No. Freenet search time is bad; also it cannot guarantee that the item can be found.`

4pts – convincing argument for/against the idea of using Freenet

d) [9 points] After obtaining the IP address of an email server, the email client and the email server use TCP to exchange email. Please illustrate the main messages exchanged, including TCP connection management messages.



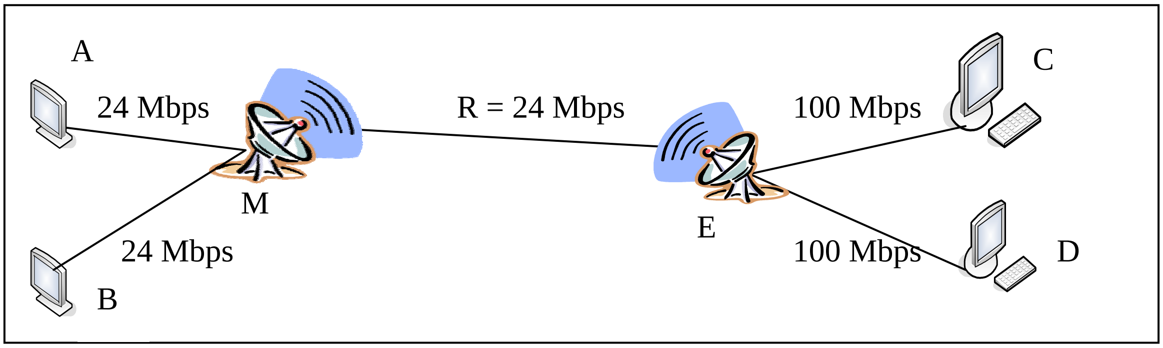
2pts – connection establishment (3-way handshake)
5pts – SMTP messages (HELO, MAIL FROM, RCPT TO, …)
2pts – connection tear-down

e) [4 points] The general practice is that it is the email client which performs close() first. Why is this so?

Client knows when it is done. Also, it releases the server of maintaining state after close.

4pts – convincing argument

2. [27 points] **Inter-planetary Reliable Transport**
In 2010, a communication link between Earth and Mars is established. The figure below shows the setup of the communication network. The only wireless link in the network is the one between M (on Mars) and E (on Earth). The other links are wire-line links. All links in the network are bi-directional, and the bandwidth of each direction of each link is labeled in the figure. Assume packet size of the network is 10,000 *bytes*. The speed of light is 300,000 km/sec. Note that we use Mbps as 10^6 *bits* per second.



Your job, as a network designer, is to design a sliding window protocol to provide reliable, efficient transport utilizing this link.

a) [3 points] We first consider the case that a single session (e.g., A transmits to C) is using the link. The session happens on March 28, 2010, when the distance between Earth and Mars is 150,000,000 km. What is the window size W of the session to efficiently utilize the link?

```
W = bandwidth * RTT = 3 M * 2 * 150,000,000 / 300,000
= 3 * 2 * 500 = 3 G bytes = 300,000 packets
```

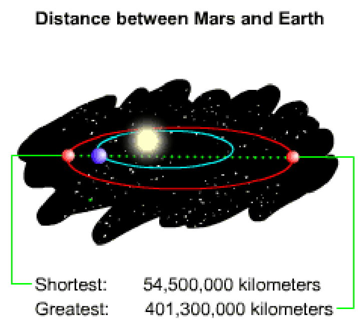
1pt – RTT computation
2pts – Using `W = bandwidth * RTT to compute W`

b) [4 points] Suppose the network can only corrupt and drop packets. As in TCP, we assign seq# to each *byte*. If you are using go-back-n, what should the number of bits be for seq#? How about selective repeat?

```
go-back-n M > 3 G + 1 -> 32 bits
select repeat : M >= 2 * 3 G = 33 bits
```

*3pts – relations between number of bits and window size (2^n > W, 2^n >= 2*W)*
1pt – compute number of bits

c) [4 points] Even when just one session is using the link, the window size W still should not be a constant across all time. Why?



4pts – convincing argument, e.g., varying distance, flow control.

d) [4 points] Since TCP congestion control can automatically discover the correct window size, one option is to use current TCP as the transport protocol. One argument against using TCP is that the bandwidth startup phase could be slow. Is this a valid argument? Please justify your answer quantitatively.

```
TCP slow start is too slow. Increase from 1 packet per
RTT to 300,000 (3 G / 10,000) packets per RTT. Need
about 19 RTT. Each RTT is 1000 seconds, so 19,000
seconds, or about 5 hours 17 minutes.
```

3pts – slow start and RTT
1pt - compute the time that slow start takes

e) [4 points] Another argument against TCP is that TCP considers packet losses as congestion indication. Estimation is that packet loss rate due to universe background radiation is 0.01%. Can TCP fully utilize the link, say in the scenario of (a)?

```
TCP window size (in packets) due to loss is 1.4 /
\sqrt(0.01%) = 1.4 * 100 = 140 packets per RTT. But
(a) requires 300,000 packets.
```

2pts – TCP cannot fully utilize the link
2pts – compute either the window size or the throughput to justify the answer

f) [4 points] One approach to improving TCP performance under wireless packet losses is to modify the routers M and E (see page 6) so that they snoop passing DATA and ACK packets and retransmit automatically and transparently. For example, when such a router observes a new DATA packet, it caches the packet. When it observes triple-duplicate ACK, it automatically retransmits, on behalf of the sender. Do you think this is a good idea, or a bad idea? If you think this is a good idea, please state the key step to make it effective. If you think this is a bad idea, please state your major reason.

```
It can be a good idea. There are two important points.
1) end to end semantics should be observed---the
router should never generate ACK by itself. 2) M or E
suppresses some ACKs to avoid triple duplicate ACKs
because otherwise, the performance can still be bad.
```

4pts – convincing argument for/against this idea.

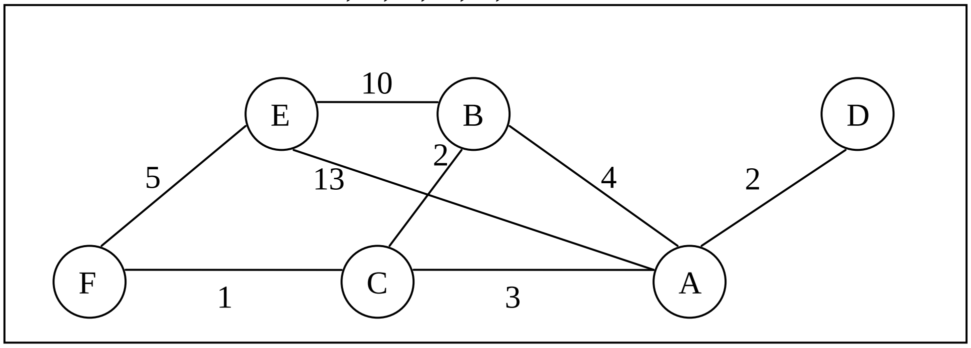
g) [4 points] Another approach is to use the idea of TCP/Vegas by observing queueing delay instead of packet loss. Suppose we can guarantee that the link bandwidth is fully utilized by TCP/Vegas. Also, since TCP/Vegas style allocation implements Nash Bargaining Solution, its rate allocation scheme might be more convincing in allocating such a scarce resource. Suppose there are three concurrent long-living sessions: 1): A to C; 2) B to D; 3) A to B. Let x1, x2, and x3 be the rates allocated to the three sessions. What are the values of x1, x2, and x3? Hint: consider the optimization framework formulation; write down the objective function and a constraint for each involved link.

```
    This is the example of the class, with 1 replaced by
24 Mbps.
```

3pts – formulation
1pt – compute values for x1, x2, and x3

3. [23 points] **Routing**

Consider a network with 6 nodes: A, B, C, D, E, and F. We consider D as the destination.



a) [5 points] Using the *synchronous* Bellman-Ford algorithm (SBF), please compute the routes to D. The initial distance estimation of all nodes except D is ∞.

round	A	B	C	E	F	D
0	∞	∞	∞	∞	∞	
1	2	X	X	X	X	
2	2	6	5	15	X	
3	2	6	5	15	6	
4	2	6	5	11	6	
5	2	6	5	11	6	

5pts - 1pt for each line

b) [4 points] Few networks run SBF; instead they run asynchronous Bellman-Ford algorithm. Why not synchronous routing algorithms?

`See slides`

c) [4 points] Synchronous or not, a serious problem of distance vector algorithms is the count-to-infinite problem. Can count-to-infinite happen to the network in Figure 4? If so, please indicate the event (i.e., link status change); if no, please state why.

Yes. Cut AD link.

d) [5 points] The reason for count-to-infinite is state inconsistency, i.e., the next-hop relation of the nodes form a loop. We call such a loop next-hop state loop. Suppose the routing of a network has converged. Suppose we can only *decrease* the weights of some links (reconnecting a link is equivalent to decreasing the link weight from infinite to the finite weight). Can a next-hop state loop happen? If so, please give one example; if no, please state why?

NO. This is the case of DSDV propagation stage. (Note: DSDV is not covered this year---2006)

3pts – no next-hop state loop
2pts – convincing argument(s), a very short proof is preferred

e) [5 points] Now suppose the routing protocol of the network is not distance vector, but BGP. The default route selection policy of each node is that a route with a shorter distance (as computed by distance vector in (a)) is more preferred. However, B has an exception in its route selection policy: its most preferred route is BEAD; otherwise, it is default. Node C also has an exception as well: it prefers CBAD the most; otherwise, it is default. Does the network have a stable BGP route selection? If so, please list the routes; otherwise, state why. Hint: list the first few highest ranked routes of each node.

```
The problem is BAD GADGET.
```

3pts – no stable BGP route selection
2pts – construction of a dispute wheel, or a convincing argument showing the instability