CS 5480/6480: Computer Networks – Spring 2012 Homework 1 Solutions Due by 9:00 AM MT on January 31st 2012

Important:

- No cheating will be tolerated.
- No extension.

CS 5480 total points = 32 CS 6480 total points = 44

Question 1 (Packet Switching versus Circuit Switching) 5 points: Suppose 40 users share a 1Mbps link. Also suppose that each user alternates (independently of the other users) between periods of activity, when the user generates data at a constant rate of 100 Kbps and periods of inactivity when the user generates no data. Suppose further that the user is active (independent of the other users) only 10 percent of the time.

1. (3.5 points) When circuit switching is used to allocate resources on the shared link, 10 users can be supported. When packet switching is used, what is the probability that 11 or more users are active simultaneously? (You must show the steps for full credit.)

The probability of r users being active simultaneously, $P(r) = {}^{40}C_r * (0.1)^r * (0.9)^{40-r}$ The probability <= 10 users being active simultaneously = $P(0) + P(1) + ... + P(10) \sim 0.999$

The probability of 11 or more users being active simultaneously = 1 - 0.999 = 0.001.

2. (1.5 points) Do you expect the result of part 1 to change when all the 40 users synchronize in their use of the shared link (i.e. all the 40 users generate data or are idle exactly at the same time)? Explain with arguments. No calculations are required for this part of the question.

When 40 users are active at the same time none of the users is likely to get the desired 100 Kbps. This example shows that packet switching is useful when users are active independent of each other. However, when they are active in a correlated manner, packet switching does not help and one might need to revert to circuit switching to allow at least 10 users to get the desired bandwidth.

Question 2 (Internet Delay Components) 3 points: The figure below shows a router and two links which are part of a bigger network.



Every time a packet is received at the router it is time-stamped (T_A) . It is also timesstamped when it is about to leave link $L(T_B)$. When 5 packets of the size 500 bytes are transmitted through the link L, $(T_B - T_A)$ is measured to be 10 ms, 2.8 ms, 2.4 ms, 4 ms, and 5.5 ms. When 5 packets of the size 1000 bytes are transmitted through the link L, $(T_B - T_A)$ is measured to be 11.0 ms, 10 ms, 2.8 ms, 3.0 ms, and 5.5 ms. Assume that processing delay at the router is negligible. What is the average queuing delay experienced by the 1000 byte packets? What are the reasonable estimates of transmission and propagation delays experienced by a packet of size 600 bytes sent through the link L?

For 500 byte packets: min $(T_B - T_A) = 2.4$ ms For 1000 byte packets: min $(T_B - T_A) = 2.8$ ms

Average queuing delay experienced by the 1000 byte packets = ((11 - 2.8) + (10 - 2.8) + (3.0 - 2.8) + (5.5 - 2.8) + (2.8 - 2.8))/5 = 3.66 ms.

Let B be the bandwidth and C be the propagation delay of link L. Then, 2.4 = 500/B + C and 2.8 = 1000/B + C. Solving for B and C we get, B = 1.25 Mbytes/sec and C = 2 ms. Thus, the transmission and the propagation delays experienced by a packet of size 600 bytes sent through the link L are 600/B = 0.48 ms, and 2 ms respectively.

Question 3 (Traceroute) 6 points:

(a) (3 points) Execute the traceroute command to two destinations of your choice, at least 12 hops away, from a source. Compute the average delay (averaged over the three delay values) for each hop and plot it in a graph with x-axis showing the hop number and y-axis the average delay corresponding to that hop. Run the same experiment a few hours later and show the new results on the same graph. In all, your graph should have four curves. How many hops are common along the paths to the two destinations? Attach the traceroute outputs. (If the traceroute command does not work on your machine, try using the service at traceroute.org.)

(b) (2 points) Suppose one of the three traceroute delay values between the source and a given router hop turns out to be unusually high. What are two possible causes for this unusually high delay?

- Network congestion resulting in high queuing delay.
- Slow path queuing delay at routers.

(c) (1 point) How would you change the traceroute program to find the IP address of every third hop instead of every hop (i.e., it finds the address of the 3^{rd} hop, the 6^{th} hop, ..., you can assume that the destination is a multiple of 3 hops away from the source)?

Increment the *time-to-live* field in the IP header by 3 instead of 1.

Question 4 (Web):

(a) (2.5 points, 0.5 point for each question) Consider the following string of ASCII characters that were captured by Wireshark when the browser sent an HTTP GET message (i.e., this is the actual content of an HTTP GET message). The characters $\langle cr \rangle \langle lf \rangle$ are carriage return and line-feed characters (that is, the italized character string $\langle cr \rangle$ in the text below represents the single carriage-return character that was contained at that point in the HTTP header). Answer the following questions, indicating where in the HTTP GET message below you find the answer.

GET / cs453 / index.html HTTP / 1.1 < cr > <lf>Host: gaia.cs.umass.edu < cr > <lf>User-Agent: Mozilla / 5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.7.2) Gecko / 20040804 Netscape / 7.2 (ax) < cr > <lf>Accept:ext / xml, application / xml, application / xhtml + xml, text / html; q=0.9, text / plain; q=0.8, image / png, */*; q=0.5 < cr > <lf>Accept-Language: en-us, en; q=0.5 < cr > <lf>Accept-Encoding: zip, deflate < cr > <lf>Accept-Charset: ISO-8859-1, utf-8; q=0.7, *; q=0.7 < cr > <lf>Keep-Alive: 300 < cr > <lf>Connection: keep-alive < cr > <lf>

a. What is the URL of the document requested by the browser?

b. What version of HTTP is the browser running?

c. Does the browser request a non-persistent or a persistent connection?

d. What is the IP address of the host on which the browser is running?

e. What type of browser initiates this message? Why is the browser type needed in an HTTP request message?

a. The document request was http://gaia.cs.umass.edu/cs453/index.html. The Host : field indicates the server's name and /cs453/index.html indicates the file name.

b. The browser is running HTTP version 1.1, as indicated just before the first <cr><lf>pair.

c. The browser is requesting a persistent connection, as indicated by the Connection: keep-alive.

d. This is a trick question. This information is not contained in an HTTP message anywhere. So there is no way to tell this from looking at the exchange of HTTP messages alone. One would need information from the IP datagrams (that carried the TCP segment that carried the HTTP GET request) to answer this question.

e. Mozilla/5.0. The browser type information is needed by the server to send different versions of the same object to different types of browsers.

(b) (2 point, 0.5 point for each question) The text below shows the reply sent from the server in response to the HTTP GET message in the question above. Answer the following questions, indicating where in the message below you find the answer.

HTTP/1.1 200 OK<cr><lf>Date: Tue, 07 Mar 2008 12:39:45GMT<cr><lf>Server: Apache/2.0.52 (Fedora)<cr><lf>Last-Modified: Sat, 10 Dec2005 18:27:46 GMT<cr><lf>ETag: "526c3-f22-a88a4c80"<cr><lf>Accept-Ranges: bytes<cr><lf>Content-Length: 3874<cr><lf>Keep-Alive: timeout=max=100<cr><lf>Connection:Keep-Alive<cr><lf>Content-Type: text/html; charset=ISO-8859-1<cr><lf><cr><lf><ld>cr><lf><ld>cr><lf></ld></dd></dr>transitional//en"><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr><lf><cr<<lf><cr><lf><cr<<lf><cr<<ld><cr<</cr>

a. Was the server able to successfully find the document or not? What time was the document reply provided?

b. When was the document last modified?

c. *How many bytes are there in the document being returned?*

d. What are the first 5 bytes of the document being returned? Did the server agree to a persistent connection?

a. The status code of 200 and the phrase OK indicate that the server was able to locate the document successfully. The reply was provided on Tuesday, 07 Mar 2008 12:39:45 Greenwich Mean Time.

b. The document index.html was last modified on Saturday 10 Dec 2005 18:27:46 GMT.

c. There are 3874 bytes in the document being returned.

d. The first five bytes of the returned document are : <!doc. The server agreed to a persistent connection, as indicated by the Connection: Keep-Alive field

Question 5 (DNS) (3.5 points, 0.5 point for each part):

 \tilde{a} . What is a whois database?

b. Use various whois databases on the Internet to obtain the names of two DNS servers. Indicate which whois databases you used.

c. Use nslookup on your local host to send DNS queries to three DNS servers: your local DNS server and the two DNS servers you found in part (b). Try querying for Type A, NS, and MX reports. Summarize your findings.

d. Use nslookup to find a Web server that has multiple IP addresses. Does the Web server of your institution (school or company) have multiple IP addresses?

e. Use the ARIN whois database to determine the IP address range used by your university.

f. Describe how an attacker can use whois databases and the nslookup tool to perform reconnaissance on an institution before launching an attack.

g. Discuss why whois databases should be publicly available.

(a) For a given input of domain name (such as cnn.com), IP address or network administrator name, *whois* database can be used to locate the corresponding registrar, whois server, dns server, etc.

(f) An attacker can use the *whois* database and *nslookup* tool to determine the IP address ranges, DNS server addresses, etc. for the target institution.

(g) If under an attack a victim can analyze the source address of packets, the victim can then use whois to obtain information about domain from which attack is coming and possibly inform the administrators of the origin domain.

Question 6 (P2P) :

(a) (4 points) Consider distributing a file of F bits to N peers using a client-server architecture. Assume a fluid model where the server can simultaneously transmit to multiple peers, transmitting to each peer at different rates, as long as the combined rate does not exceed u_s .

a. Suppose that $u_s/N \le d_{\min}$. Specify a distribution scheme that has a distribution time of NF/u_s .

b. Suppose that $u_s/N \ge d_{min}$. Specify a distribution scheme that has a distribution time of F/d_{min} .

 d_{min} . c. Conclude that the minimum distribution time is in general given by max{NF/u_s, F/ d_{min} }.

a. Consider a distribution scheme in which the server sends the file to each client, in parallel, at a rate of a rate of u_s/N . Note that this rate is less than each of the client's download rate, since by assumption $u_s/N \le d_{\min}$. Thus each client can also receive at rate u_s/N . Since each client receives at rate u_s/N , the time for each client to receive the entire file is $F/(u_s/N) = NF/u_s$. Since all the clients receive the file in NF/u_s , the overall distribution time is also NF/u_s .

b. Consider a distribution scheme in which the server sends the file to each client, in parallel, at a rate of d_{\min} . Note that the aggregate rate, $N d_{\min}$, is less than the server's link rate u_s , since by assumption $u_s/N \ge d_{\min}$. Since each client receives at rate d_{\min} , the time for each client to receive the entire file is F/ d_{\min} . Since all the clients receive the file in this time, the overall distribution time is also F/ d_{\min} .

c. From Section 2.6 we know that

 $D_{CS} \ge \max \{ NF/u_s, F/d_{\min} \}$ (Equation 1)

Suppose that $u_s/N \le d_{\min}$. Then from Equation 1 we have $D_{CS} \ge NF/u_s$. But from (a) we have $D_{CS} \le NF/u_s$. Combining these two gives:

 $D_{CS} = NF/u_s$ when $u_s/N \le d_{\min}$. (Equation 2)

We can similarly show that:

 $D_{CS} = F/d_{\min}$ when $u_s/N \ge d_{\min}$ (Equation 3).

Combining Equation 2 and Equation 3 gives the desired result.

(b) (4 points) Consider distributing a file of F bits to N peers using a P2P architecture. Assume a fluid model. For simplicity, assume that d_{min} is very large, so that peer download bandwidth is never a bottleneck.

a. Suppose that $u_s \leq (u_s + u_1 + ... + u_N)/N$. Specify a distribution scheme that has a distribution time of F/u_s .

b. Suppose that $u_s \ge (u_s + u_1 + ... + u_N)/N$. Specify a distribution scheme that has a distribution time of $NF/(u_s + u_1 + ... + u_N)$. c. Conclude that the minimum distribution time is in general given by max{ F/u_s , $NF/(u_s + u_1 + ... + u_N)$ }

a. Define $u = u_1 + u_2 + \dots + u_N$. By assumption

$$u_s \le (u_s + u)/N$$
 Equation 1

Divide the file into N parts, with the ith part having size $(u_i/u)F$. The server transmits the ith part to peer i at rate $r_i = (u_i/u)u_s$. Note that $r_1 + r_2 + \ldots + r_N = u_s$, so that the aggregate server rate does not exceed the link rate of the server. Also have each peer i forward the bits it receives to each of the N-1 peers at rate r_i . The aggregate forwarding rate by peer i is $(N-1)r_i$. We have

$$(N-1)r_i = (N-1)(u_s u_i)/u \le u_i,$$

where the last inequality follows from Equation 1. Thus the aggregate forwarding rate of peer i is less than its link rate u_i .

In this distribution scheme, peer i receives bits at an aggregate rate of

$$r_i + \sum_{j <> i} r_j = u_s$$

Thus each peer receives the file in F/u_s .

b. Again define $u = u_1 + u_2 + \dots + u_N$. By assumption

 $u_s \ge (u_s + u)/N$ Equation 2

Let $r_i = u_i/(N-1)$ and $r_{N+1} = (u_s - u/(N-1))/N$

In this distribution scheme, the file is broken into N+1 parts. The server sends bits from the ith part to the ith peer (i = 1, ..., N) at rate r_i . Each peer i forwards the bits arriving at rate r_i to each of the other N-1 peers. Additionally, the server sends bits from the (N+1)st part at rate r_{N+1} to each of the N peers. The peers do not forward the bits from the (N+1)st part.

The aggregate send rate of the server is

 $r_1 + \ldots + r_N + N r_{N+1} = u/(N-1) + u_s - u/(N-1) = u_s$

Thus, the server's send rate does not exceed its link rate. The aggregate send rate of peer i is

 $(N-1)r_i = u_i$

Thus, each peer's send rate does not exceed its link rate. In this distribution scheme, peer i receives bits at an aggregate rate of

$$r_i + r_{N+1} + \sum_{j < i} r_j = u / (N-1) + (u_s - u / (N-1)) / N = (u_s + u) / N$$

Thus each peer receives the file in $NF/(u_s+u)$.

(For simplicity, we neglected to specify the size of the file part for i = 1, ..., N+1. We now provide that here. Let $\Delta = (u_s+u)/N$ be the distribution time. For i = 1, ..., N, the ith file part is $F_i = r_i \Delta$ bits. The $(N+1)^{st}$ file part is $F_{N+1} = r_{N+1} \Delta$ bits. It is straightforward to show that $F_1 + ..., + F_{N+1} = F$.)

c. We know from section 2.6 that

$$D_{P2P} \ge max\{F/u, NF/(u+u)\}$$

Combining this with (a) and (b) gives the desired result.

(c) (2 points) Consider an overlay network with N active peers, with each pair of peers having an active TCP connection. Additionally, suppose that the TCP connections pass through a total of M routers. How many nodes and edges are there in the corresponding overlay network?

There are *N* nodes in the overlay network. There are N(N-1)/2 edges.

Question 7 (required for CS6480, extra credit for CS5480) 12 points: Read the following paper – "Internet Indirection Infrastructure" by I. Stoica, D. Adkins, S. Zhuang, S. Shenker and S. Surana, ACM Sigcomm Conference, August 2002. (http://eng.utah.edu/~cs5480/readings/i3-sigcomm.pdf)

Answer the following questions that are based on this paper:

1. (3 points) Consider a model in which data sent from two senders is added together before being sent to a receiver. Show and describe how this could be implemented using the i3 architecture.



The combination server will set a timer T. Once T expires, it combines the data it has received from two senders via (Data_Combine, Combination) together and send it to the receiver via (ID,R). In the figure above, both senders use the id stack and send their data to the id pair (Data_Combine, Combination). The combination server gets its data from (Data_Combine, Combination). And after it processes the data, it sends the data to the tuple pair (ID, R). Receiver R receives all the data sent to (ID, R).

2. (3 points) Consider a model in which multimedia MPEG data sent from a sender must be converted to the JPEG format before it is received at a receiver. The conversion operation is specified by the sender. The receiver wants to ensure that all the data it receives comes through a firewall. Show and describe how this could be implemented using the i3 architecture.

Solution (provided by Siddharth Ramesh):

- The sender sends all packets with ((id_MPEG-JPG , id_data) , data)
- i3 would then forward this packet to the conversion server (which serves the identifier, id_MPEG-JPG).
- The conversion server does the necessary conversion, pops off id_MPEG-JPG from the stack. Now the packet contains just the identifier id_data. It is then forwarded by i3 to the server responsible for serving 'id_data'

There are 2 ways in which a receiver could enforce a firewall on the converted packet.

Method 1: If the firewall is a separate machine

- Receiver sends out a trigger (id_data, (id_FIREWALL, recv_addr)) which resides in the server responsible for serving id_data
- When a packet with identifier 'id_data' sent by the conversion server matches the above trigger, the new identifier (id_FIREWALL, recv_addr) is inserted into the packet and forwarded.

• This reaches the firewall (which serves the identifier id_FIREWALL). The firewall (after screening) inserts the identifier recv_addr and sends it to the receiver through IP.

<u>Method 2:</u> If the firewall is running as a process on the same machine as the receiver but on a different port

- The receiver inserts the (trigger id_data, (recv_firewall_addr, recv_addr) where
- recv_firewall_addr consists of (recv_IP_addr, firewall_port_no) and recv_addr consists of (recv_IP_addr, recv_port_no) (NOTE that port numbers can be specified as a part of 'addr' in i3.)
- The conversion server sends a packet with 'id_data' which goes to the i3 server serving the identifier 'id_data'. The above trigger is already residing there.
- So, the packet is now forwarded (using IP) to 'recv_firewall_addr', so it actually goes to the firewall process.

The firewall process gets id_recv from the packet, and sends it to the recv_port on the same machine.

- 3. (3 points) Explain how mobility can be supported in the i3 architecture.
- 4. (3 points) Can you use the i3 architecture for preventing Denial-of-service attacks on servers? Discuss.