50.012 Networks (2020 Term 6)

Homework 1

Hand-out: 21 Sep

Due: 2 Oct 23:59

Name: melody Student ID: 1004489

1. Why two ISPs peer with each other? (Hint: consider the different types of peering: Regional ISP with Regional ISP, Tier 1 with Tier 1, and Regional ISP with content provider) What is the role of an IXP? How does an IXP generate revenue? (Hint: study some IXP, e.g., https://www.sgix.sg/)

ISPs Peer to:

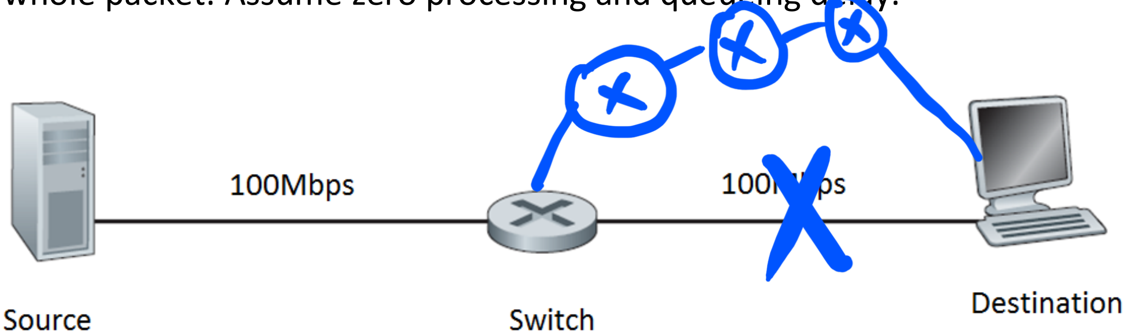
1. Reduce cost
2. Avoid using the intermediate ISP provider of a higher tier. Hence, avoid payment to that intermediate ISP provider.
3. By pass third party connection to send traffic to multiple networks

The role of an IXP is to allow networks to interconnect directly via the exchange point instead of going through upstream intermediaries and allow traffic to pass directly. IXP essentially contains a network switch to route traffic between different members networks.

Definitely, to connect to an IXP requires money. Hence, the IXP charges each internet service provider (ISP) that connects to each of them.

2. (2019's mid-term exam question): Calculate the end-to-end delay (i.e., the duration from the first bit sent by the source to the last bit received by the destination) for a packet with size of 1500 bytes (12,000 bits) for the following settings:

2.1 The source and the destination are connected via a store-and-forward switch (as in the figure below). Assume that each link in the figure has a propagation delay of 12 µs (1 µs = 10^{-6}s) and operates at 100Mbps (consider 1M = 10^6). The switch begins forwarding immediately after it finishes receiving the whole packet. Assume zero processing and queueing delay.



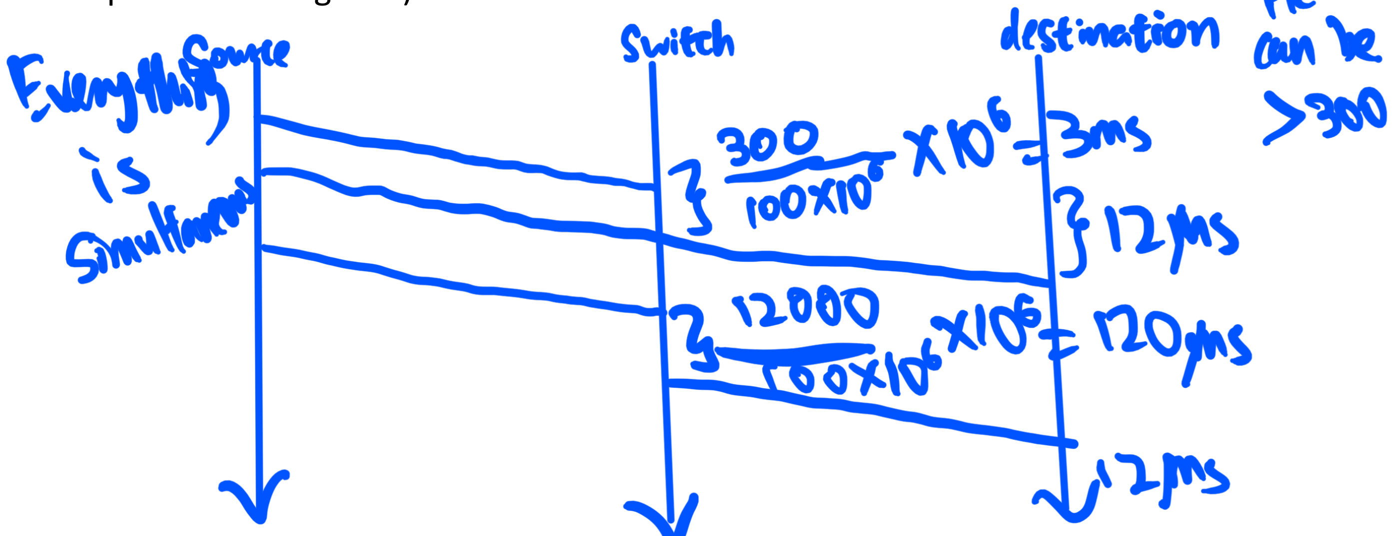
Single switch case

$$2\left(12 + \left(\frac{12000}{100\times10^6}\times10^6\right)\right) = 264\,\mu s$$

2.2 Same scenario as **2.1** above, calculate the end-to-end delay when there are four switches chained together (each switch and each link has the same setting as in scenario **2.1**) in the path between the source and the destination.

$$5\left(12 + \left(\frac{12000}{100\times10^6}\times10^6\right)\right) = 660\,\mu s$$

2.3 Same scenario as **2.1** above, i.e., there is only a single switch between the source and the destination, but assume the switch implements "cut-through" switching, i.e., the switch begins to forward the packet after the first 300 bits of the packet have been received. Calculate the end-to-end delay. (Hint: Draw a space-time diagram.)



Total delay $= 3\mu s + 12\mu s + 120\mu s + 12\mu s$
$= 147\mu s$

3. (textbook chapter 2, review problem R26): In Section 2.7 of the textbook, the UDP server described needed only one socket, whereas the TCP server needed two sockets. Why? If the TCP server were to support n simultaneous connections, each from a different client host, how many sockets would the TCP server need?

The characteristic of the UDP and TCP protocol allows. For UDP servers, these servers follow the UDP protocol which states that connection doesn't need to be accepted. Hence, there can only be one set of buffered sent and/or received packets. TCP servers have only 2 different kinds of state. One state is to keep track of the socket's port number and interface that is listened. (for one socket). For the other TCP socket, once the TCP connection is accepted, the socket has its own connection-specific state needed to be tracked separately - The client's state. So, having two TCP sockets, one is to close the socket to stop sending and receiving on the client socket while continuously listening on the other socket. With having only one socket, you close all of its state that is not saved. Hence, violating the characteristic of the TCP protocol - continuous connection. UDP server doesn't have a welcome socket. Each connection will enter server through this socket.

The TCP need n+1 sockets for n simultaneous connection. One welcome socket for all the connections to be accepted and n others to support one connection each.

(Adapted from stackoverflow.com)

Conclusion: TCP - two socket - one welcome socket to accept connections
→ another to receive subsequent connections

Note to TA: if cannot read or understanding handwriting for qn 4, I highlighted in diff colour for the where I find my ans for you.

4. (textbook problem chapter 2, problem 4) 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 <cr><lf> are carriage return and line-feed characters (that is, the italized character string <cr> 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><cr><lf>

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

b. What version of HTTP is the browser running? HTTP/1.1

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

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? Mozilla firefox.

a. http://gaia.cs.umass.edu/cs453/index.html

b. version 1.1 indicated just before <cr><lf> pair

c. Persistent. Connection: Keep-alive

d. Trick question. This information is not contained in an HTTP message anywhere.

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.

Note: Same note as qn 4.

5. (textbook problem chapter 2, problem 5) 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><!doctype html public "//w3c//dtd html 4.0 transitional//en"><lf><html><lf><head><lf> <meta http-equiv="Content-Type"content="text/html; charset=iso-8859-1"><lf> <meta name="GENERATOR" content="Mozilla/4.79 [en] (Windows NT 5.0; U) Netscape]"><lf> <title>CMPSCI 453 / 591 / NTU-ST550A Spring 2005 homepage</title><lf></head><lf> <much more document text following here (not shown)>

=> the GET status code is 200

a. Was the server able to successfully find the document or not? What time was the document reply provided? Yes. 12:39:45 GMT 07 Mar 2008

b. When was the document last modified? Sat 10 Dec 2005 18:27:46 GMT

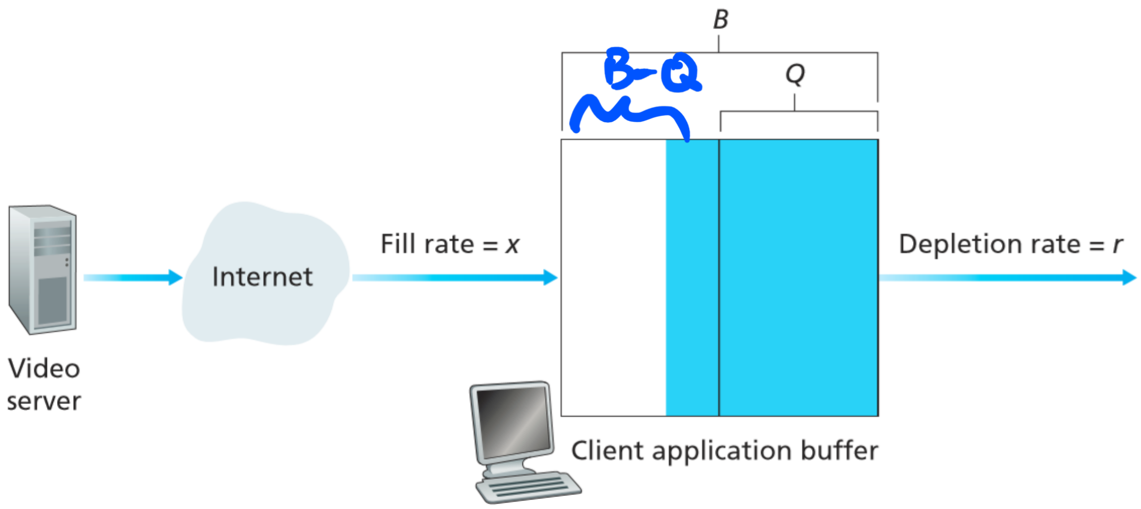
c. How many bytes are there in the document being returned?

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

c. 3874 bytes

d. <!doc. The server agreed to a persistent connection

6. Recall the simple model for HTTP streaming shown in the Figure below. Let B denote the size of the client's application buffer, and let Q denote the number of bits that must be buffered before the client application begins playout. Also, let r denote the video consumption rate. Assume that the server sends bits at a constant rate x whenever the client buffer is not full.



6.1 Suppose that x < r. In this case playout will alternate between periods of continuous playout and periods of freezing. Determine the length of each continuous playout and freezing period as a function of Q, r, and x.

6.2 Now suppose that x > r. At what time does the client application buffer become full?

6.1 Freezing Period $= \frac{Q}{x}$ s , continuous playout $= \frac{Q}{r-x}$ s

6.2 Fill the minimum buffer before playout $= \frac{Q}{x}$

Fill buffer $= \frac{B-Q}{x-r}$

$\therefore$ time taken $= \frac{B-Q}{x-r} + \frac{Q}{x}$ s

7. (2019's midterm exam question) Consider distributing a file of F = 6 x 10^9 bits to N=100 peers. The server has an upload rate of u_s = 30 Mbps, and each peer has a download rate of d_i = 2 Mbps and an upload rate of u_i=1 Mbps. Assume 1M = 10^6. Calculate the minimum distribution time (i.e., to let every peer have a copy of the file) for:

7.1 the client-server distribution mode, and

7.2 the P2P distribution mode.

7.1 $D_{c-s} \geq \max(NF/u_s,\ F/d_{min})$

$= \max\left(\frac{6\times10^9\times100}{30\times10^6}, \frac{6\times10^9}{2\times10^6}\right)$

$= 20000s$

7.2 $D_{P2P} \geq \max\left(F/u_s,\ F/d_{min},\ \frac{NF}{(u_s+\sum_{i=1}^{n} u_i)}\right)$

$= \max\left(\frac{6\times10^9}{30\times10^6}, \frac{6\times10^9}{2\times10^6}, \frac{100\times6\times10^9}{(30\times10^6)+(100\times10^6)}\right)$

$= \max\left(200, 3000, \frac{60000}{13}\right)$

$= \frac{60000}{13}s$ #

=== END ===