Spring 21, EE 357: Computer Networks

Solution to Homework 2

Solution to problem 1

SMS (Short Message Service) is a technology that allows the sending and receiving of text messages between mobile phones over cellular networks. One SMS message can contain data of 140 bytes and it supports languages internationally. The maximum size of a message can be 160 7-bit characters, 140 8-bit characters, or 70 16-bit characters. SMS is realized through the Mobile Application Part (MAP) of the SS#7 protocol, and the Short Message protocol is defined by 3GPP TS 23.040 and 3GPP TS 23.041. In addition, MMS (Multimedia Messaging Service) extends the capability of original text messages, and support sending photos, longer text messages, and other content.

iMessage is an instant messenger service developed by Apple. iMessage supports texts, photos, audios or videos that we send to iOS devices and Macs over cellular data network or WiFi. Apple's iMessage is based on a proprietary, binary protocol APNs (Apple Push Notification Service).

WhatsApp Messenger is an instant messenger service that supports many mobile platforms such as iOS, Android, Mobile Phone, and Blackberry. WhatsApp users can send each other unlimited images, texts, audios, or videos over cellular data network or WiFi. WhatsApp uses the XMPP protocol (Extensible Messaging and Presence Protocol).

iMessage and WhatsApp are different than SMS because they use data plan to send messages and they work on TCP/IP networks, but SMS use the text messaging plan we purchase from our wireless carrier. Moreover, iMessage and WhatsApp support sending photos, videos, files, etc., while the original SMS can only send text message. Finally, iMessage and WhatsApp can work via WiFi, but SMS cannot.

Solution to problem 2

Application layer protocols: DNS and HTTP

Transport layer protocols: UDP for DNS; TCP for HTTP

Solution to problem 3

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 12:40:46 Greenwich Mean Time.

b) The document index.html was last modified on Saturday 10 Dec 2005 18:28:47 GMT.

c) There are 3848 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.

Solution to problem 4

The total amount of time to get the IP address is

$$RTT_1 + RTT_2 + \cdots + RTT_m$$

Once the IP address is known, RTT_0 elapses to set up the TCP connection and another RTT_0 elapses to request and receive the small object. The total response time is

$$2RTT_0 + RTT_1 + RTT_2 + \cdots + RTT_m$$

Solution to problem 5

a) $RTT_1 + \cdots + RTT_m + 2RTT_0 + 7 \times 2RTT_0 = 16RTT_0 + RTT_1 + \cdots + RTT_m$.

b) $RTT_1 + \cdots + RTT_m + 2RTT_0 + 2 \times 2RTT_0 = 6RTT_0 + RTT_1 + \cdots + RTT_m$

c) Persistent connection with pipelining. This is the default mode of HTTP:

$RTT_1 + \cdots + RTT_m + 2RTT_0 + RTT_0 = 3RTT_0 + RTT_1 + \cdots + RTT_m$.

Persistent connection without pipelining, without parallel connections:

$RTT_1 + \cdots + RTT_m + 2RTT_0 + 7RTT_0 = 9RTT_0 + RTT_1 + \cdots + RTT_m$.

Solution to problem 6

a) The time to transmit an object of size L over a link or rate R is L/R. The average time is the average size of the object divided by R:

$$\Delta = (750,000 \text{ bits})/(15,000,000 \text{ bits/sec}) = .05 \text{ sec}$$

The traffic intensity on the link is given by $\beta\Delta$ = (18 requests/sec)(.05 sec/request) = 0.9. Thus, the average access delay is (.05 sec)/(1 - .9) ≈ .5 seconds. The total average response time is therefore .5 sec + 2 sec = 2.5 sec.

b) The traffic intensity on the access link is reduced by 60% since the 60% of the requests are satisfied within the institutional network. Thus the average access delay is (.05 sec)/[1 – (.4)(.9)] = .078125 seconds. The response time is approximately zero if the request is satisfied by the cache (which happens with probability .6); the average response time is .078125 sec + 2 sec = 2.078125 sec for cache misses (which happens 40% of the time). So the average response time is (.6)(0 sec) + (.4)(2.078125 sec) = 0.83125 seconds.

Solution to problem 7 For calculating the minimum distribution time for client-server distribution, we use the following formula:

$$D_{cs} = \max\{NF/u_s, F/d_{min}\}$$

Similarly, for calculating the minimum distribution time for P2P distribution, we use the following formula:

$$D_{p2p} = \max\{F/u, F/d_{min}, NF/(u_s + \sum_{i=1}^{N} u_i)\}$$

,

where F = 15 Gbits = 15 * 1024 Mbits, u_s = 30 Mbps, $d_{min} = di$ = 2 Mbps.

Note, 300Kbps = 300/1024 Mbps. Here are the tables:

Client Server

		N		
		10	**100**	**1000**
	300 Kbps	7680	51200	512000
u	**700 Kbps**	7680	51200	512000
	2 Mbps	7680	51200	512000

Peer to Peer

		N		
		10	**100**	**1000**
	300 Kbps	7680	25904	47559
u	**700 Kbps**	7680	15616	21525
	2 Mbps	7680	7680	7680