CSE3214 Computer Network Protocols and Applications

Chapter 2
Examples and Homework Problems

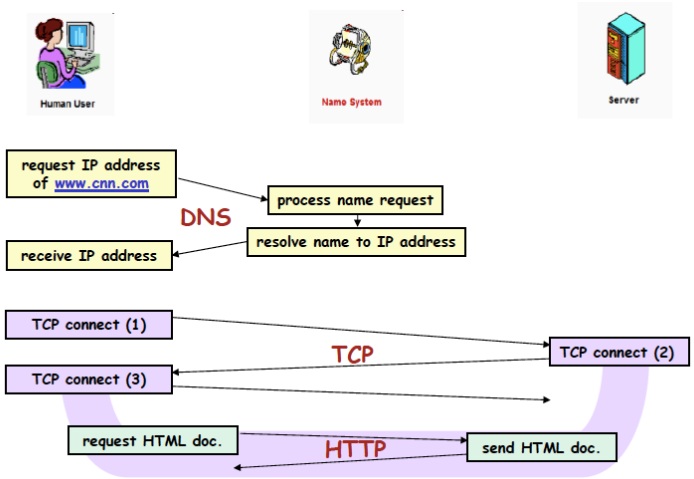
Example 1 (P3)

Consider an HTTP client that wants to retrieve a Web document at a given URL. The IP address of the HTTP server is initially unknown. What transport and application layer protocols besides HTTP are needed in this scenario?

Solution to Example 1

Application layer protocols: DNS and HTTP

Transport layer protocols: UDP for DNS; TCP for HTTP



Example 2 (P7)

Suppose within your Web browser you click on a link to obtain a Web page. The IP address for the associated URL is not cached in your local host, so a DNS lookup is necessary to obtain the IP address. Suppose that n DNS servers are visited before your host receives the IP address from DNS, the successive visits incur an RTT of $RTT_1 \ldots RTT_n$. Further suppose that the Web page associated with the link contains exactly one object, consisting of a small amount of HTML text. Let RTT_0 denote the RTT between the local host and the server containing the object. Assuming zero transmission time of the object, how much time elapses from when the client clicks on the link until the client receives the object?

Solution to Example 2

The total amount of time to get the IP address is:
$RTT_1 + RTT_2 + \ldots + RTT_n$

Once the IP address is known, elapses to set up the TCP connection and another elapses to request and receive the small object.
The total response time is:
$2RTT_0 + RTT_1 + RTT_2 + \ldots + RTT_n$

Example 3 (R10)

Consider a short, 10m link, over which a sender can transmit at a rate of 150 bits/sec in both directions. Suppose that packets containing data are 100,000 bits long, and packets containing only control (e.g. ACK or hand-shaking) are 200 bits long. Assume that N parallel connections each get $1/N$, of the link bandwidth. Now consider the HTTP protocol, and suppose that each downloaded object is 100 Kbits long, and that the initial downloaded object contains 10 referenced objects from the same sender. Would parallel downloads via parallel instance of non-persistent HTTP make sense in this case? Now consider persistent HTTP. Do you expect significant gains over the non-persistent case? Justify and explain your answer.

Solution to Example 3

Note that each downloaded object can be completely put into one data packet. Let Tp denote the one-way propagation delay between the client and the server.

First consider parallel downloads using non-persistent connections. Parallel downloads would allow 10 connections to share the 150 bits/sec bandwidth, giving each just 15 bits/sec. Thus, the total time needed to receive all objects is given by:

$(200/150+Tp + 200/150 +Tp + 200/150+Tp + 100{,}000/150+ Tp)+ (200/(150/10)+Tp + 200/(150/10) +Tp + 200/(150/10)+Tp + 100{,}000/(150/10)+ Tp)$

$= 7377 + 8*Tp$ (seconds)

Solution to Example 3 (con't)

Now consider a persistent HTTP connection. The total time needed is given by:

$(200/150+Tp + 200/150 +Tp + 200/150+Tp + 100{,}000/150+ Tp)+ 10*(200/150+Tp + 100{,}000/150+ Tp)$

$=7351 + 24*Tp$ (seconds)

Assuming the speed of light is $300*10^6$ m/sec, then Tp=$10/(300*10^6)$=0.03 microsec. Tp is therefore negligible compared with transmission delay.

Thus, we see that persistent HTTP is not significantly faster (less than 1 percent) than the non-persistent case with parallel download.

Homework Problems

Solutions will be available in week 4

Problem 1: True or False? (P1)

1. A user requests a Web page that contains of some text and three images. For this page, the client will send one request message and receive four response messages.
2. Two distinct Web pages (for example, www.mit.edu/research.html and www.mit.edu/students.html) can be sent over the same persistent connection.
3. With non-persistent connections between browser and origin server, it is possible for a single TCP segment to carry two distinct HTTP request messages.

Problem 1: True or False? (con't)

4. The Date: header in the HTTP response message indicates when the object in the response was last modified.
5. HTTP response messages never have an empty message body.

Problem 2 (P14, part)

a. HTTP and FTP are both file transfer protocols and have many common characters; for example, they both run on top of TCP. What are the main differences between HTTP and FTP?

b. How does SMTP mark the end of a message body?

Problem 3 (P8, part)

Referring to Example 2 (P7), suppose the HTML file references eight very small objects on the same server. Neglecting transmission times, how much time elapses with

a. Non-persistent HTTP?
b. Persistent HTTP?

Problem 4 (P19)

In this problem, we use the useful *dig* tool available on Unix and Linux hosts to explore the hierarchy of DNS servers. Recall a DNS server higher in the DNS hierarchy delegates a DNS query to a DNS server lower in the hierarchy, by sending back to the DNS client the name of that lower-level DNS server. First read the man (manual) page for *dig,* and then answer the following questions.

a. Starting with a root DNS server (from one of the root server [a-m].root-servers.net), initiate a sequence of queries for the IP address for www.cse.yorku.ca by using *dig*. Show the list of the names of DNS servers in the delegation chain in answering your query.
b. Repeat part a) for www.google.com, www.yahoo.com.

Useful info for Problem 4

- List below are Web based pages related to *dig*. These links are useful if you do not have access to a Linux or Unix machine.
  - Man page for *dig*: http://linux.die.net/man/1/dig
  - Online *dig* tool: http://networking.ringofsaturn.com/Tools/dig.php

Problem 5 (P21)

Suppose that your department has a local DNS server for all computers in the department. You are an ordinary user (i.e. not network/system administrator). Can you determine if an external Web site was likely accessed from a computer in your department a couple of seconds ago? Explain.