Tony Gao
CS453
Written HW 2

1.

a) What is the local name and path of the file that is being retrieved in this GET message?
Answer:
The name of the file is “quotation3.htm”
The path to the file is “/kurose_ross/interactive/quotation3.htm”

b) What is the host name?
Answer:
“gaia.cs.umass.edu"

c) What version of HTTP is the client running?
Answer:
HTTP/1.0

d) Does the client already have a (possibly out-of-date) copy of the requested file? Explain. If so, approximately how long ago did the client receive the file, assuming the GET request was issued at 2:54 PM on Jan 3, 2018.
Answer:
The client last cached a copy of the file on Jan 3, 2018 at 1:54:01 PM. If the GET was done at 2:54, there was 59 minutes 59 seconds since the last download of the file. If the file was modified on the server since the date on the last cached copy on the client, the server will send a new copy of the file with a new “last-modified” header value. If not modified, the server would return HTTP 304 not modified.

e) Consider the following situation: the host sends a GET request and downloads a web document at 16:05:00 EDT on Sun, January 07, 2018. The web document will not change for the next 3 hours. Later, the host sends another GET request (this time, a conditional GET statement with the If- 2 Modified-Since parameter) at 18:50:30 EDT on the same day. What are the first two lines of the returned HTTP response (HTTP version = 1.1)?
Answer:

GET /path/to/file HTTP/1.1
Host: some.hostname.etc
If-Modified-Since: Sun, 07 Jan 2018 16:05:00 EDT

2.

a) Assuming non-persistent HTTP (and assuming no parallel connections are open between the browser and server). How long is the response time - the time from the when the user requests the URL to the point in time when the page and its embedded objects are displayed? Make sure you describe the various components that contribute to this delay.
Answer:

$L_1 = 500Kb$

$L_{2..5} = 200Kb$

$R = 100Mbps = 100000Kbps$

3 Way Handshake: $3d_{prop} = 3(125ms) = 375ms$

Page Transfer: $d_{prop} + \frac{500}{100000}s = 125ms + 5ms = 130ms$

Image Transfer: $4d_{prop} + 4(\frac{200}{100000})s = 4(125)ms + 8ms = 508ms$

Total Delay: $(5)375ms + 130ms + 508ms = 2513ms$

For each object, a new connection must be opened in serial after the server is finished transmitting the previous object. Total time 5 * 3 Way Handshake + Time to transfer objects.

b) Again assume non-persistent HTTP, but now assume that the browser can open as many parallel TCP connections to the server as it wants. What is the response time in this case?
Answer:

$L_1 = 500Kb$

$L_{2..5} = 200Kb$

$R = 100Mbps = 100000Kbps$

3 Way Handshake: $3d_{prop} = 3(125ms) = 375ms$

Page Transfer: $d_{prop} + \frac{500}{100000}s = 125ms + 5ms = 130ms$

Image Transfer: $d_{prop} + \frac{200}{100000}s = 125ms + 2ms = 127ms$

$\max(\text{Page,Image}) = 130ms$

Total Time: $375ms + 130ms = 505ms$

I assume the TCP connections are opened at the same time, there is no additional delay besides propagation delay between the server ACK, client ACK/request for object and server response to the request for each object in each parallel connection. The time to establish the TCP connection and request each object would just be $3d_{prop}$ as each connection would attempt handshake and request the object at the same time. The total time to transfer would be equal to the handshake time plus max(time page, time image) since all 5 connections would transfer the objects at the same time. So, total time would be $375ms + 130ms = 505ms$

c) Now assume persistent HTTP (i.e., HTTP1.1). What is the response time, assuming no parallel connections and no pipelining (a request is sent after reply to the previous request has been received)?
Answer:

$L_1 = 500Kb$

$L_{2..5} = 200Kb$

$R = 100\,Mbps = 100000\,Kbps$

Handshake: $2\mathrm{d}_{prop} = 2(125ms) = 250ms$

Page Transfer: $2d_{prop} + \frac{500}{100000}s = 250ms + 5ms = 255ms$

Image Transfer: $2d_{prop} + \frac{200}{100000}s = 250ms + 2ms = 252ms$

Total Time: $250ms + 255ms + 4(252ms) = 1513ms$

There would be a handshake to establish the TCP connection, then each file would incur a delay of 1 RTT + L/R for the request + time to receive object.

d) Now suppose persistent HTTP with parallel connections is used. What is the response time?
Answer:

$L_1 = 500Kb$

$L_{2..5} = 200Kb$

$R = 100\,Mbps = 100000\,Kbps$

3 Way Handshake: $3\mathrm{d}_{prop} = 3(125ms) = 375ms$

Page Transfer: $d_{prop} + \frac{500}{100000}s = 125ms + 5ms = 130ms$

Image Transfer: $d_{prop} + \frac{200}{100000}s = 125ms + 2ms = 127ms$

$\max(\text{Page,Image}) = 130ms$

Total Time: $375ms + 130ms = 505ms$

The 5 separate TCP connections would need to be established with a 3 Way handshake delay. When this is over, all 5 requests for objects have been sent to the server at the same time and the server sends the response to all 5 requests at the same time. The additional delay is the max of the delay for the page and image.

3.

a) Assume the browsers on a.com network are configured to access the internet through the local web cache. List the sequence of DNS and HTTP messages sent/received from/by m1.a.com as well as any other messages that leave/enter the a.com network that are not directly sent/received by m1.a.com from the point that the URL is entered into the browser until the file is completely received. Indicate the source and destination of each message. You can assume that every HTTP request by m1.a.com is first directed to the HTTP cache in a.com (before any DNS query) and that the cache is initially empty and that all DNS requests are iterated queries.
Answer:
I assume the iterative DNS model follows the canonical DNS query model described on

p137-138 of the book where there is a recursive query to the local DNS server and iterative queries onwards to root, TLD, and authoritative name servers.

1. The browser for m1.a.com will parse the URL, extracting the hostname "www.b.com".
2. The browser for m1.a.com sends a DNS query to the Local DNS server for a Type A record mapping the host name to an IP Address. If the Local DNS Server has this record cached from a previous query, it responds with the mapping.
3. If the Local DNS server does not have this mapping, it forwards the Type A record request to the root DNS server. The root DNS Server will respond with this mapping, if it has it cached from a previous request. If it does not have the mapping cached, it will respond with a list of Type NS and Type A records for TLD Nameservers responsible for .com, one of which will be the TLD server that has a Type NS and Type A record for the authoritative DNS server for www.b.com.
4. The Local DNS Server resends the DNS query for "www.b.com" to each TLD nameserver returned in the Type NS/Type A records from the root nameserver. One of these TLD servers will respond with the Type NS/Type A record for the authoritative DNS server for b.com. One of these servers may also have a cached mapping for "www.b.com", in which case the TLD server would respond with the Type A mapping from "www.b.com" to IP to the local DNS server.
5. The Local DNS Server resends the Type A DNS query for "www.b.com" to the authoritative DNS server for b.com. The authoritative DNS server will return the Type A record for "www.b.com" to the Local DNS Server. The local DNS server will return this record to the browser.
6. The browser for m1.a.com establishes a TCP connection with the web cache and sends an HTTP get request for www.b.com/bigfile.htm. The cache opens a TCP connection to the IP of www.b.com.
7. If the cache has copy of bigfile.htm, it sends a CONDITIONAL GET request to www.b.com with a if-modified-since header of the datetime of the recent cached copy of bigfile.htm. If www.b.com responds with 304 Not Modified, the cache responds to the GET request for bigfile.htm from m1.a.com with the cached copy of bigfile.htm. If the file was modified since the last cached file, www.b.com responds with 200 OK and the newer version of bigfile.htm to the cache. If the cache does not have a copy of bigfile.htm, it sends a GET request to www.b.com. www.b.com responds with the requested file, and a last-modified header. The file is cached in the web cache as last modified according to this header.
8. The web cache responds to m1.a.com with bigfile.htm.

b) How long does it take to accomplish the steps you outlined in your answer to 2a? Explain how you arrived at this answer. In answering this question, you can make the following assumptions:
   - The packets containing any DNS commands and HTTP commands such as GET are very small compared to the size of the file, and thus their transmission times (but not their propagation times) can be neglected.
   - The one-way propagation delay from anywhere in a.com to any other site in the Internet (including b.com) is 250 ms.
   - The throughput of the connection between R2-R1 is 1 M bit per second (Mbps)

   Answer:

I assume all DNS caches are empty at all levels of the hierarchy and the Local Web Cache is also empty. I also assume there is a propagation delay on the 1Gbps LAN intranet of m1.a.com.

1. 0ms
2. d_{prop} of m1.a.com to the local dns server
3. DNS Request from Local DNS to Root: 250ms
DNS Response from Root to Local DNS: 250ms
4. Assuming there is only one TLD server returned in the previous request.
DNS request from Local DNS to TLD: 250ms
DNS response from TLD to Local DNS: 250ms
5. DNS request from Local DNS to Authoritative DNS: 250ms
DNS Response from Authoritative DNS to Local DNS: 250ms
DNS Response from Local DNS to Browser: d_{prop} of m1.a.com to local DNS
6. TCP Handshake from m1.a.com to local web cache: 2 * d_{prop} of the local network from m1.a.com to the local web cache
GET Request from m1.a.com to local web cache: d_{prop} of the local network from m1.a.com to local web cache.
TCP Handshake from local web cache to www.b.com: 500ms
7. GET Request from local web cache to www.b.com: 250ms
Response to GET from www.b.com to local web cache: 250ms + 500000ms
8. Response to GET from local web cache to m1.a.com: d_{prop} from m1.a.com to local web cache + 500ms

Total time: 2 * d_{prop} of m1.a.com to local dns server + 500ms + 500ms + 500ms + 3 * d_{prop} of m1.a.com to local web cache + 500ms + 250ms + 250ms + 500000ms + d_{prop} of m1.a.com to local web cache + 500ms = 503000ms + 2 * d_{prop} of m1.a.com to local dns + 4 * d_{prop} of m1.a.com to local cache

Explanation:
Total time = Total time for DNS resolution + TCP Handshake delay from browser to cache + Propagation delay of HTTP GET from browser to cache + TCP Handshake delay from cache to www.b.com + Propagation delay of GET from m1.a.com to local cache + TCP Handshake delay from cache to www.b.com + GET request delay from cache to www.b.com + GET response delay from www.b.com to cache + Time to transfer 500Mbit from www.b.com to local web cache + Propagation delay from local web cache to m1.a.com + Time to transfer 500Mbit from local web cache to m1.a.com

c) Now assume that machine m2.a.com makes a request to exactly the same URL that m1.a.com made. List the sequence of DNS and HTTP messages sent/received from/by m2.a.com as well as any other messages that leave/enter the a.com network that are not directly sent/received by m2.a.com from the point that the URL is entered into the browser until the file is completely received. Indicate the source and destination of each message. [Hint: make sure you consider caching here]
Answer:
1. DNS Query from m2.a.com to local DNS server for hostname www.b.com. The DNS

Server has the Type A record for www.b.com cached from the last request.
2. DNS Query response from local DNS server to m2.a.com with Type A record for www.b.com.
3. TCP Handshake from m2.a.com to local web cache.
4. HTTP GET from m2.a.com to local web cache for www.b.com/bigfile.htm
5. TCP Handshake from local web cache to www.b.com
6. HTTP CONDITIONAL GET from local web cache to www.b.com
7. HTTP 304 NOT MODIFIED response from www.b.com to local web cache
8. HTTP GET response from local web cache to m2.a.com with cached copy of bigfile.htm

d) Now suppose there is no HTTP cache in network a.com. What is the maximum rate at which machines in a.com can make requests for the file www.b.com/bigfile.htm while keeping the time from when a request is made to when it is satisfied non-infinite in the long run?
Answer:

$L = 500\,Mb$

$R = 1\,Mb/s$

To solve: a = average packets per second

$$\frac{aL}{R} \leq 1$$

$$500a \leq 1$$

$a \leq \dfrac{1}{500}$ packets per second, arriving every 500 seconds

According to the golden rule for traffic intensity, the (average rate of packet arrival) * (transmission delay per packet) must be less than or equal to 1. If requests are made on average every 500 seconds, there will be on average little to no queuing delay between each request. If a > 1/500 packets per second, the queueing delay will approach infinity as more and more requests will pile up in the outbound queue of the router on b.com's institutional network.

4.

One of the features of the browser on your computer is to keep a history of the site you have visited. Suppose you accidentally erased the history. Now you want to find out if you have very recently accessed a particular web site. How would you do that?

Answer:

I would browse the browser local cache of my web browser. On Chrome, this would be done by entering chrome://cache/ into the omnibar. Or Firefox: about:cache. Then, I would search the cache for the hostname of the site that I want to know if I accessed.

5.

Suppose client A initiates a Secure Shell (SSH) session with server S. At about the same time, client B also initiates a SSH session with server S.

a. Provide possible source and destination port numbers for:

	Source Port No.	Destination Port No.
a) The segment from A to S	Randomly Assigned Port	22
b) The segment from B to S	Randomly Assigned Port	22
c) The segment from S to A	22	Randomly assigned port from the initial TCP handshake
d) The segment from S to B	22	Randomly assigned port from the initial TCP handshake

b. If A and B are different hosts, is it possible that the source port number in the segments from A to S is the same as that from B to S? (Yes or No!) Answer:
   Yes
c. How about if they are the same host? (Yes or No!). Answer:
   No

6.

Consider the following statements and determine if they are either true/false. If a statement is false, state your reasons.

1. FTP and HTTP are similar, but the biggest difference between the two is that HTTP sends its control information in-band while FTP does so out-of-band. Answer:
   False. There are too many differences between FTP and HTTP to consider them similar. They are both file transfer protocols that run on TCP, but with big differences. FTP is a stateful protocol that uses two TCP connections, transferring control information on a persistent TCP connection out of band with data. HTTP is a stateless protocol that uses either a persistent or non-persistent TCP connection depending on the specific implementation. HTTP transfers control information through the same TCP connection as data, in band. All the information to serve an HTTP request can be recovered from the initial request, whereas FTP servers must remember certain parameters about the user associated to the particular control channel. HTTP is a pull protocol. FTP is both a push and pull protocol depending on if the client implementation uses active or passive FTP.
2. SMTP is primarily a pull protocol. Answer:
   False, SMTP is a push protocol. The mail server that is sending the mail is the server that initiates the connection.
3. SMTP and POP3 are mail access protocols. Answer:
   False, SMTP is a mail transfer protocol with primary purpose to send mail. POP3 is a mail access protocol that supports downloading mail from a mail server.
4. The primary task of DNS servers is to translate IP addresses to hostnames. Answer:
   False, the primary task of DNS servers is to translate hostnames to IP addresses.

7.

1. What is the purpose of DNS resource records?
   Answer:
   Hostname to IP mapping (Type A), Hostname to Canonical Hostname Aliasing (Type C), Domain to Nameserver mapping (Type NS), Mail hostname to canonical hostname aliasing (Type MX)
2. What is the primary advantage of a conditional GET request? Answer:
   Performance increase/decrease of transmission delay. The conditional GET involves no transfer of object data if the requested object has not changed since the last cached copy.
3. What is the difference between a control connection versus a data connection? Answer:
   Commands meant to control or invoke certain functions defined by a protocol are sent over the control connection. Data that is required as a response to or as a part of a command are sent over the data connection.
4. State the 3 primary components of the internet mail system. Answer:
   Mail server, Mail User Agent, SMTP
5. What is Tit-for-Tat in BitTorrent protocol? Answer:
   Tit-for-Tat is the system of incentivizing the top 4 peers that are currently uploading data to the current peer. If a peer is among those top 4 peers, the current peer will begin to upload to those peers and prioritize the upload bandwidth for that peer.

8.

Supposed that you are working in a company that is going to build a bundle of software to sell in early next year, and you are assigned to choose a suitable application layer protocol and transport protocol (TCP, UDP) for each software before the developers get started. Please give a reason to support your answer

Application Description	Application Layer Protocol	Transport Layer Protocol
1. Server-based file sharing application that will be used in big corporate.	SFTP. Security is paramount to file sharing in a corporate setting, so secure FTP should be used.	TCP. FTP uses TCP sockets to communicate.
2. Peer-to-peer file sharing application for distributing files.	Bittorrent. It is an effective, fast peer to peer file sharing protocol.	TCP. Bittorrent uses TCP sockets to communicate.
3. E-mail client application that user can send/receive email and also manage all email into many folders	SMTP and IMAP. SMTP allows for sending/receiving mail and is crucial for mail transfer. IMAP allows for organizing mail on the mail server.	TCP. Both SMTP and IMAP use TCP sockets.
4. Web-based E-mail client that have the same features as the E-mail client application	HTTP. HTTP allows for browsers to communicate to mail servers assuming the mail server has support for HTTP.	TCP. HTTP communicates over TCP sockets.

Application Description	Application Layer Protocol	Transport Layer Protocol
5. Video on demand website that can adjust video quality based on user bandwidth.	DASH. DASH was designed to provide variable streaming rates over HTTP	TCP. HTTP requires TCP.
6. Hostname to IP converter application that can convert hostname to IP address	DNS. This is one of the main reasons DNS was created.	UDP. DNS uses UDP if the transmission size is under 512B. A hostname : ip mapping is sure to be smaller than 512B
7. Credit card payment website for online shopping customer	HTTPS. The customers would not want their credit card numbers stolen, so HTTP over SSL/TLS should be used	TCP. HTTP requires TCP.
8. Video call application	SIP and RTP. SIP to identify the IP address of the callee and establish the call. RTP to transfer the encoded video and voice bits over the established call.	UDP. SIP can operate over both TCP and UDP. RTP typically runs on UDP, so for simplicity and adherence to the transport layer protocol norm with RTP, just UDP should be used.