EECS 325/425

HW 2

Due Oct. 5, 2016 before 11:59pm

**(27 points for 325; 37 for 425)**

**Read: Sec. 2.2.5 (Web caching) and the “BitTorrent” subsection of Sec. 2.6.1.**

**Optional read: Sec. 2.6.2 (“Distributed hash tables”)**

**Problem 1:** When you are on Case campus, you can either use the default configuration, which utilizes a DNS resolver provided by Case network, or configure your device to use a public DNS resolution service, like Google’s 8.8.8.8. Assume both Case and Google resolvers have the same response time and use “vanilla” DNS as discussed in class (i.e., without any extensions). Which configuration would likely result in better performance when you obtain content from a website that subscribes to a conventional content delivery network of the kind we studied in class? Explain. **(3 points: 1 for answer and 2 for explanation)**

**Problem 2:** Consider desktop users in an organization, which wants to decide whether or not to deploy a forward proxy to improve users’ experience. The organization has measured the time to download an average object and found it to be 2s. Benchmarking the proxy server being considered showed that for a cache miss, the proxy adds 50ms of local processing overhead. However, if the proxy already has the requested object, downloading this object from the proxy cache takes on average only 100ms. Assuming the cache hit rate of 40%, will deploying this proxy improve users’ browsing experience in this organization? Justify your answer. **(2 points)**

**Problem 3:** A Web site **foo.com** wants to use a content delivery network **cdn.net** for delivering part of its content. It wants to continue serving all URLs with hostname www.foo.com itself (i.e., from its web server with IP address, say, 160.50.1.1) but use the CDN to deliver content with URLs having hostname movies.foo.com.

1. Describe the DNS records that the foo.com’s authoritative DNS server would need to maintain to allow the above functionality (i.e., to serve URLs with hostname www.foo.com from host 160.50.1.1 and URLs with hostname movies.foo.com from CDN cdn.com). **(2 points)**
2. Describe the message exchange **at the application level (no TCP)** that occurs when a user clicks on the URL <http://foo.com/home.html> (assume no DNS records are cached anywhere). Make sure to indicate clearly all the entities exchanging the messages. **(2 points)**
3. Describe the application-level message exchange that occurs when a user clicks on the URL <http://movies.foo.com/topten.html> (assume no DNS records are cached anywhere). **(2 points)**

**Problem 4:** Explain how HTTP cookies can be used to accomplish the following functions. Give specific examples in each case, including necessary URLs, cookies, HTTP header fields, etc. Non-specific answers will not receive full credit.

1. Support language preferences of users of a bilingual Web site without retaining any information about the users or their computers **(2 point)**
2. Allow an authorized user to revisit the site without repeated log-in procedure **(2 point)**
3. Track user access patterns within a Web site **(2 point)**
4. Allow a company to track user accesses across several partner sites **(2 point)**

**Problem 5**: Consider a server that must distribute a 5GB movie to 100 users. Assume all users requested the movie at the same time. Let the server’s Internet connection be 10Mbps, and each user’s connection 1Mbps for uploads and 1Mbps for downloads. Assume infinite bandwidth in the Internet core. For simplicity, assume 1 byte = 10 bits and 1Gbit = 1000MBits.

1. Assuming the client-server paradigm, propose a scheduling for the server to distribute the file to the users that minimizes the total time for all users to receive the file (hint: the server cannot process all client requests sequentially one by one – this would keep the server connection underutilized.) How long it would take for all users to receive this file, assuming this scheduling? **(2 points)**
2. Assuming the peer-to-peer paradigm, propose a scheduling (i.e., which users should be getting the file from which hosts) for the file distribution from the server to users and among the users that minimizes the total distribution time. What is the resulting time distribution? **(2 points)**

**Problem 6:** Textbook, 6th edition. Chapter 2, Problem 26. **(4 points)**

**EECS 425 only: (10 points; 10 pts bonus for 325 students):**

You will examine the use of DNS by popular web sites. Please do the following (a tip: Make sure you close any applications running on your laptop while you are performing this lab. Close any extra browser windows, etc. Modern web pages often issue HTTP requests behind the scene, and these will pollute your trace):

1. Download the list of top 1000 websites from http://stuffgate.com/stuff/website/top-1000-sites. Select a block of web sites you will be working with: if your network ID ends with 0 or has no numbers, you will work with 9 sites ranked 1, 101, 201, …, 901. If your network ID ends with “1”, work with the sites ranked 2, 102, …, 902. And so on, if your ID ends with “9”, work with the sites ranked 10, 110, …, 910.
2. Using wireshark, collect traces of accesses to your sites. You can write a simple python script or Java program that downloads your 10 pages, or you can just manually cut and paste your ten URLs to your browser one by one *(Warning: in the latter case, you may stumble on offensive content -- When you do research on the Internet, you face this issue rather often. Just try to avert your eyes… And make sure your browser is in the private browsing mode while you are doing this so you don’t pollute your computer. In this particular case, you can avoid these issues by downloading your pages programmatically from a script*).Now you have your packet trace to work with. **Note:** do not forget to stop wireshark after accessing your ten sites! **(1 pts)**
3. Write a program or programs (using python or java) to answer the following questions:
   1. How many total DNS resolutions your host had to perform for these 10 accesses? What’s the average number of DNS resolutions per site? **(3 pts)**
   2. How many of the examined sites used load balancing among multiple Web servers? Is this the exact number, or a conservative estimate? Justify your answer. (**Note:** in answering this question, only consider DNS resolutions for the top names on our web sites. E.g., for yahoo.com, just consider the response query for “yahoo.com”, and not for any additional hostnames that might be used in embedded URLs.). Does the number of servers used by a web site for load balancing correlate with the site’s rank? Substantiate your answer (do not just say “yes” or “no”. Compute an appropriate statistical metric, or use a scatter graph to visualize the dependency.) Were all your page downloads preceded by a DNS query? If some did not, how would you explain that? **(3 pts)**
   3. What’s the fraction of HTTP vs. HTTPS connections in your trace? (Hint: HTTP uses port 80 and HTTPS port 443). **(3 pts)**

**Notes:**

1. wireshark writes the trace in a binary format. To read this trace record by record, the easiest thing to do is to read this file using tcpdump utility with “-r” option and write it to a file – this will convert the trace into an ASCII file. I.e., on linux, if your trace file is named “trace\_file”, you could say “tcpdump -r -A test-trace > result” and then use the “result” file as input to your program.
2. You may need to log in as root to run tcpdump.

Deliverables:

1. The trace file in ASCII format (i.e., after processing by tcpdump).
2. Your program(s), properly commented (readability will be considered in grading). Do not prepare a separate documentation (do not bother with class documentations produced with, e.g., Dr. Java etc.) Just insert sufficient comments within your source code.
3. A lab report.