Exam - Final

Due: 11:00am, May 10th, 2012

Closed Book. Maximum points: 100

NAME:

1. TCP Congestion Control [15 pts]

a. TCP Tahoe and Reno have two congestion-window increasing modes, slow start and additive increase. If a connection has a window size of w and a maximum segment size of MSS, **what is the increase in the window size *for each received acknowledgment* in each of these modes**? Explain your reasoning. [5 pts]

   i. Slow Start

   ii. Additive Increase

b. In class we saw a simplified scenario of how two flows using AIMD reach fair and efficient transmission rates (illustrated with the Chiu-Jain plots). If we have two flows, A and B, and instead of multiplying their window size by 1/2 on loss detection, they multiply their window size by 3/4, **will they still converge to the fair and efficient transmission rates? What if A uses 1/2 as a factor and B 3/4, will one of them get more bandwidth in the end? Which one?**[5 pts]

c. TCP Reno reacts differently to triple duplicate acks and to ack timeouts. **What does TCP do in each case**? **Why is the reaction different**? Explain what the sender can infer about the network and/or the receiver side.[5 pts]

2. Alternative Congestion Control [15 pts] Bittorrent recently proposed an alternative transport protocol to TCP that sill offers reliable, in-order delivery, and uses a sliding window and acknowlegments for congestion and flow control. The main difference is the congestion control algorithm. The goals of the protocol, called μTP, are to minimize the impact of large BitTorrent traffic on other applications, while fully utilizing the available bandwidth when the link is underutilized.

Here's a simplified explanation of how μTP congestion control works (assume nodes have perfectly synchronized clocks, many other details omitted):

- The sender adds a timestamp t_s to every outgoing packet.
- The receiver records a timestamp t_r for every received packet, and sends the difference $\delta = t_r - t_s$ on the acknowledgment.
- The sender keeps a running minimum of the successive δs (this is the delay with minimum queueing delay), and assumes that any extra delay in other packets is due to queueing delay.
- If the queueing delay keeps below a small threshold (say, 100ms), μTP increases the window size.
- If the queueing delay starts increasing above the threshold, μTP will decrease its sending rate in proportion to the difference.

a. One of the major impacts μTP tries to minimize is on the latency seen by other applications. Assuming you have a home DSL modem with a large outgoing queue and that the bottleneck link is the first hop between you and your ISP, **why will a very large TCP flow increase the latency of, say, a VoIP call you are making**? [5 pts]

b. In the absence of other traffic, **explain why μTP will eventually fully utilize the bottleneck link, but not go above its capacity**? [5 pts]

c. **Why will μTP effectively yield to another TCP flow**? [5 pts]

3. DNS [15 + 6 pts]

a. Google and OpenDNS offer alternative DNS servers that they claim are more reliable, better provisioned, and with better cache hit ratios than your ISP's DNS server. However, using these resolvers can result in sub-optimal interaction with CDNs like Akamai.

As an example, using my regular resolver at Brown CS, I get the following answer to looking up www.bestbuy.com:

```
www.bestbuy.com.    3552    IN  CNAME   www.bestbuy.com.edgesuite.net.
www.bestbuy.com.edgesuite.net. 5610 IN  CNAME   a1105.b.akamai.net.
a1105.b.akamai.net. 3   IN  A   198.7.236.240
```

Pinging 198.7.236.240, I get an average latency of 0.3ms, and traceroute reveals that this server is located in Oshean's network, which is Brown's network provider.

If I use Google's resolver instead, 8.8.8.8, I get the following answer:

```
www.bestbuy.com.    2169    IN  CNAME   www.bestbuy.com.edgesuite.net.
www.bestbuy.com.edgesuite.net. 19535 IN CNAME   a1105.b.akamai.net.
a1105.b.akamai.net. 11  IN  A   208.44.23.16
```

Pinging 208.44.23.16, I get an average of 26ms, and a path that has to cross at least two more ASes. **Why does Akamai give me a worse server when I use a different resolver?** [5 pts]

b. Let us examine the cost of looking up non-existing domains. Assume that www.mitsukoshi.co.jp is a domain name that exists, that there are no cached entries anywhere (other than the root servers cache), and that each part of the name is handled by a separate server. How many recursive lookups does our local resolver have to do to discover that the two following domains do not exist: [5 pts]

- ww.mitsukoshi.co.jp ?

- www.mitsukoshi.co.jap ?

c. Now assume you look up www.mitsukoshi.co.jp after looking up each one of the two domains above. **How many recursive lookups does your local resolver do after looking up each one**? (Further assume that the caches are cleared after each pair of lookups). [5 pts]

d. **(Bonus)** Suppose that Rhode Island wants to become an independent country, and wants to start by registering its own top-level .ri domain. What are the necessary steps so that: [6 pts]

i. This works for all Internet users, without requiring any changes?

ii. This works for only for users who change their DNS settings?

4. Caching [15 pts]

a. **Why is caching so effective in the DNS system?** [3 pts]

b. Both IP and DNS have the concept of TTL, or time to live. **What is the purpose of TTL in IP packet headers, and what is the purpose of TTL in DNS responses?** [3 pts]

c. In the snippets of response from Akamai, in the previous question, **why were the TTLs for the first two entries set to very high numbers of seconds, and to only a few seconds for the third entry**? [3 pts]

d. Cache poisoning can have very undesirable effects in the network. **Give two examples of cache poisoning attacks, specifying how the attacker has to proceed, and what are possible implications of a successful attack.** [6 pts]

5. Consistent Hashing [14 pts] We saw that consistent hashing is a very useful abstraction which has been used, for example, by Akamai to select caches for content objects, and is adapted to the distributed setting in the Chord algorithm. It is also used in most memcached client libraries, as well as in Amazon's Dynamo storage system.

a. **Explain how consistent hashing works.** [8 pts]

b. **What is its main advantage**? Assume you have 20 servers caching 8400 objects, and you want to add a server. **Contrast what happens, in terms of expected network traffic to rearrange the objects, if you are using consistent hashing and a naïve modulo hashing scheme.** [6 pts]

6. HTTP [11 pts] Suppose you click on a link on your web browser, and assume the DNS mapping for the host in question is already cached. The resulting web page references 10 objects. Assume that each object is small enough that all of them could be fetched together in a single RTT.

a. **How many RTTs do you have to wait until all objects are fetched, if you have:** [6 pts]

   i. HTTP 1.0 with no parallel connections

   ii. HTTP 1.1 with no persistent connections and up to 5 parallel connections allowed

   iii. HTTP 1.1 with persistent connections and no pipelining

   iv. HTTP 1.1 with persistent connections and pipelining

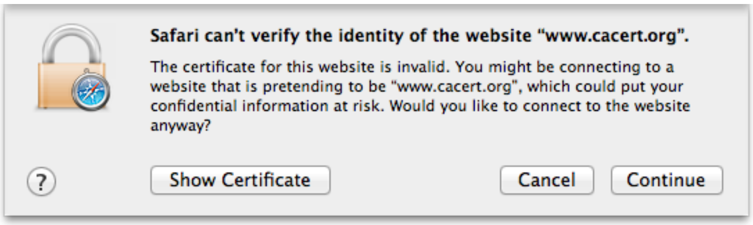
b. SPDY is a new protocol very similar to HTTP, but adding a couple of interesting features. One of them is the multiplexing of multiple request/response pair on the same TCP connection, with the ability to prioritize any substream. **Why is this better than HTTP pipelining?** [5 pts]

7. Security [15 pts]

a. In SSL/TLS, generally the server has a certificate, and a private/public key pair, but the client does not. Explain how they can still achieve bidirectional confidentiality. [6 pts]

b. What is the implication of the lack of a client-side certificate? [4 pts]

c. When using HTTPS, sometimes you may come across a warning like the following: [5 pts]



Safari can't verify the identity of the website "www.cacert.org".

The certificate for this website is invalid. You might be connecting to a website that is pretending to be "www.cacert.org", which could put your confidential information at risk. Would you like to connect to the website anyway?

i. If you choose to go on, is your connection still going to be encrypted?

ii. What security property are you giving up?

iii. Give two reasons for which a browser would not accept a server's certificate.