Problem Set 1

CS352 Internet Technology

Instructions

Please read and follow these instructions carefully.

1. You must work on this problem set individually.
2. Please complete your answers to the questions below and upload your responses to Canvas as a single PDF file.
3. Your answers (preferably typed up rather than handwritten) must be clear, legible, and concise. If we cannot understand your answer with reasonable effort, you will get zero credit.
4. If you leave a question out or clearly state "I don't know", you will receive 25% of the points for that question.
5. We care not just about your final answer, but also how you approach the questions. In general, if you show us your reasoning for your answers, we will do our best to provide partial credit even if your final answer is incorrect. However, if you provide no reasoning, and your answer turns out to be incorrect, you will typically receive zero points. So, please explain yourself.
6. You are free to discuss the problem set on Piazza or through other means with your peers and the instructors. You may refer to the course materials, textbook, and resources on the Internet for a deeper understanding of topics. However, you cannot lift solutions from other students or from the web. Do not post these problems to question-answering services like Chegg. All written solutions must be your own. We run sophisticated software to detect plagiarism and carefully monitor student answers.
7. There is a due date/time but no "time limit" on problem sets. That is, you may take as long as you need to work on problem sets, as long as you submit them on time. Further, you may make an unlimited number of submissions on Canvas.
8. As a response to the last question of this problem set, please specify who you collaborated with, and also include all the resources you consulted to answer questions in this problem set, including URLs of pages you visited on the Internet. Also specify which question and aspect you got help with. Please be as thorough and complete as possible here. It is mandatory to answer this question.
9. If you have any questions or clarifications on the problem set, please post them on Piazza or contact the course staff. We are here to help.

Questions

1. **Layering (2 points)**. Why is Internet software/hardware arranged in layers?

Internet software & hardware is arranged in layers because it provides modularity, meaning that each layer has a well-defined function and interfaces to layers above and below it. As a result, each layer can modified and developed without affecting another layer.

2. **Modulation type (2 points)**. A digital signal contains the sequence of bits 0101100100100. What modulation is being used in the physical signal (for example, voltage plotted against time) below? Please explain how you arrived at your conclusion.

By observing the voltage vs. time plot below and the sequence of bits, I can determine that this is frequency modulation. Regions with more peaks correspond to a 1, while regions with fewer peaks correspond to a 0.

3. **Switching types (2 + 2 = 4 points).** For each question below, your answer must be one of "circuit", "message", or "packet", along with a justification.

    a. Suppose you'd like to design a network with routers that need to support application conversations that transfer a lot of data, but only between a few pairs of endpoints. Which kind of switching would you use for the network's routers? Please justify your answer (2 points).

I would use circuit switching because we only have a few endpoints and therefore it becomes acceptable to allocate paths fully for a duration of time. Additionally, circuit switching allows for the transmission of a lot of data.

b. Suppose, instead, your network must support applications that move a small number of application-level messages between a large number of pairs of endpoints. Each application-level message may be very large. Further, even if the entire message isn't received yet, it is helpful for the applications to transfer smaller portions of the messages as early as possible to the receiving endpoints. What kind of switching would you build into the network's routers? Please justify your answer (2 points).

In this situation, I would use packet switching. One of packet switching's benefits is that it can send a message in smaller portions known as "packets", which suits this situation well. Additionally, packets are rapidly transmitted

> so endpoints will receive smaller portions of messages earlier than other
> switching methods.

4. Bandwidth and delay (2 + 2 + 1 + 1 = 6 points).

a. Is it possible for a link or network path to have high bandwidth and high propagation delay? Why or why not? Justify.

> Yes, it is possible. A link could have a high bandwidth but have a long length.
> Despite being able to transmit a lot of data at once, travelling the long path
> would cause a high propagation delay.

b. Is it possible for a link or network path to have low bandwidth and low propagation delay? Why or why not? Justify.

> Yes, it is possible. If the link has a significantly short path, that would
> result in a low propagation delay. Even if the bandwidth is low, a short path
> would allow the data to be propagated quite rapidly because the time spent on
> the path would be minimal.

c. If you need to transfer a large file very quickly, which kind of link would you rather prefer, one with a higher or lower bandwidth?

> In the event of transferring a large file quickly, a link with a higher
> bandwidth would be preferred. A higher bandwidth allows for more data to be
> transmitted at once.

d. If you need to support a highly interactive (real-time) voice call, which kind of link would you rather prefer, one with a higher or lower propagation delay?

> A link with a lower propagation delay would be better for a highly interactive
> voice call. With a lower propagation delay, voices would be transmitted from
> endpoint-to-endpoint more quickly. On the contrary, a higher propagation delay
> would result in more time between speaking and the other endpoint hearing.

5. Total delay (3 + 3 = 6 points). An endpoint sends a packet of size 4000 Bytes (1 Byte = 8 bits) over a link with bandwidth 2 Mbit/s (1 M = 10^6). The link has a propagation delay of 10 milliseconds (1 milli = 10^{-3}).

a. After how long since the sender pushed the first bit of the packet into the link does the final bit of the packet arrive at the other end of the link? Explain your answer. **(3 points)**

$$4000bytes * \frac{8bits}{1byte} = 32,000bits$$

$$\frac{1s}{2Mbits} * 32,000bits = 16ms$$

$$16ms + 10ms = 26ms$$

b. At the other end of the link is a router which pushes the packet through another link that is identical to the last one (described in question above). The packet encouters a small queue with queueing delay 5 milliseconds before entering this second link. The router uses store and forward packet switching. What is the total time it takes to move the packet end to end, i.e., first bit entering the first link to last bit exiting the second link? Explain your answer. **(3 points)**

$$26ms + 26ms + 5ms = 57ms$$

6. **App-layer connection (6 points).**

a. What are the components of the 4-tuple denoting an application-layer connection?

b. Why could you not just use the 2-tuple (IPA,IPB) to denote an application-layer connection?

b. Could you think of an example where IPB and portB have the same values across many application-layer connections?

> Accessing a remote computer via ssh/sftp. If the user is connected via an
> application like MobaXTerm, the application would use ssh and sftp, typically on
> port 22 with an unchanging IP address.

7. **Client and server (2 points).** Explain one difference between a "client" and a "server" in an application.

> A server may handle many incoming connections, whereas a client will connect to
> a single server.

8. **Domain Name System (2 * 6 = 12 points).****
   a. Why do we need the domain name system (DNS)? What problem does it solve?

> IP addresses are difficult to remember for the average human and the DNS allows
> for human-readable addresses to online destinations. The DNS solves the problem
> of delivering a client to the correct IP address based on a client-specified
> domain name.

b. Explain how hierarchy helps DNS meet the scaling needs of serving Internet users.

> The DNS hierarchy has many benefits that work to serve a large sum of internet
> users. Having multiple DNS servers on different levels prevents any single
> server from becoming oversaturated and slowing down. Traffic directed to servers
> higher in the hierarchy is reduced.

c. Explain how replication helps DNS meet the scaling needs of serving Internet users.

> The redundancy created by replication prevents any single server on a level from
> becoming over-saturated. Having many servers on a level allows for multiple
> servers to split the traffic so that no single server begins to run too slowly.
> Additionally, this also increases the bandwidth at that level of the hierarchy.

d. Explain how caching helps DNS meet the scaling needs of serving Internet users.

> Caching helps DNS meet the scaling needs by reducing the number of DNS requests
> made at every level it is implemented. By locally storing the resolved IP from
> the DNS request, an origin server / client will not need to resolve that domain
> name again. This reduces traffic to DNS servers.

e. What is the purpose of the identification field on a DNS protocol message?

> The identification field on a DNS protocol message enables bidirectional
> communication between the DNS and the client. The DNS uses the identificatoin
> field to respond to the any requests from the client.

f. What is an authoritative DNS server?

> An authoritative DNS server is a DNS server that handles all the DNS requests
> for a region. It responds to requests with information for that region.

9. **Finding authority (3 points)**. Can you list a set of authoritative DNS servers for cs.princeton.edu? Explain how you arrived at the answer.

```
dns1.cs.princeton.edu.
ns6.dnsmadeeasy.com.
dns2.cs.princeton.edu.
ns5.dnsmadeeasy.com.
ns7.dnsmadeeasy.com.
```

I used the following command to get this answer.

```
dig +short ns cs.princeton.edu
```

10. **Partitioning, caching, and DNS query load (7 points)**. Suppose there are N domain names available in the Internet, equally subdivided over T top-level domains (such as .com). Suppose there are C local DNS resolvers, each issuing N $*$k queries, i.e., k queries to each of the N domains, where k >1. Assume that each of the N domains has a unique authoritative name server with an entry in the corresponding top-level DNS server. Further assume that queries are iterative from the point of view of the local DNS resolver. Now answer the questions below. Explain your reasoning clearly to receive partial credit.

a. Suppose there is no caching on any of the DNS servers. How many queries does the root DNS server service? **(1 point)**

$$C * N * k$$

> N$*$k queries are made by each local DNS resolver, and there are C local DNS
> resolvers. Therefore, there are C$*$N$*$k requests made to the root DNS server.

b. Suppose there is no caching on any of the DNS servers. How many queries does each top-level DNS server service? **(2 points)**

$$\frac{C * N * k}{T}$$

c. Suppose all local DNS resolvers cache the DNS responses they receive indefinitely. Now how many queries does the root DNS server service? **(2 points)**

$$C * N$$

Given C local DNS resolvers that make N*k requests, we know that the DNS
resolver will cache the result after the first of k requests for N. This reduces
the amount of requests sent by each individual DNS resolver to N. Therefore,
given C local DNS resolvers, we have C*N queries received by the root DNS.

d. Suppose all local DNS resolvers cache the DNS responses they receive indefinitely. How many queries does each top-level DNS server service? **(2 points)**

$$\frac{C * N}{T}$$

Given C local DNS resolvers that make N*k requests, we know that the DNS
resolver will cache the result after the first of k requests for N. This reduces
the amount of requests sent by each individual DNS resolver to N. Therefore,
given C local DNS resolvers, we have C*N queries sent in total. Then, we
distribute the C*N queries evenly across the T top-level servers, to get the
answer (C*N)/T.

11. **Collaboration and References (mandatory)**. Who did you collaborate with on this problem set? What resources and references did you consult? Please also specify on what questions and aspects of the problem set you got help on. If you did not consult any resources other than the lecture slides and textbook, just say "no collaboration".

I worked my classmate Akash Shah and consulted
- my lecture notes
- stack overflow
- TA

I used dig -h so I could find the correct arguments to help me with problem 9.

#dns
#modulation
#bandwidth
#partitioning
#caching