CS-736 Midterm Exam II: Taking It Home

On this exam, each question is labeled with its value. There are four (4) normal questions, for a total of 100 points.

You have until *Thursday, December 14 at noon* to finish. When you are finished, please turn in the exam to Remzi in his office. If he is not there for some reason, slipping your exam under the door will do.

You may use whatever materials you need in order to answer the questions herein. Thus, it is an open-book, open-note exam. However, **do not talk to anyone about any aspect of this exam except the instructor**, as this exam is meant to test your skills and yours alone. You are of course free to send Remzi email about the exam, if you have clarifying questions, scalding complaints, witty remarks, or other concerns.

Answer clearly and concisely, use your own extra paper as necessary, and good luck to all!

Please put your name on every page that you turn in!

1. **To Send Or To Receive - That Is The First Question. (20 points)** Networking forms the core of much of modern computer systems. This question examines two networking layers, the bare-bones minimalism of U-net, and the much higher-level package provided by Xerox RPC.

   a.1: U-net purports to be a networking system that simply provides the most basic thing a networking system should do – safe, protected multiplexing of the network interface. Thus, like exokernel, U-net tries to remove all policy from itself, and leave that to higher-level protocols. Unfortunately, this is not always possible. Describe and discuss in detail **two** policy decisions that must be made in a real production U-net system.

   a.2: For each policy decision that you list, what would a reasonable policy be? (justify)

   b.1: The RPC system makes several design decisions that impact performance of the system. Name and discuss **two** such design decisions.

   b.2: Quantify their impact of each of the two design decisions on system performance. Make whatever assumptions you need to about the network or other aspects of the system.

2. **My Consistency Is Weaker Than Yours. (30 points)** In this question, we discuss one of the mobile systems that we examined in class, Coda.

   a.1: In Coda, during the hoarding phase, there is a tension between collecting data in the local disk cache in preparation for disconnection ("disconnected data") versus data that is actively being used ("active data"). If the cache favors disconnected data too strongly, what happens? (give a simple example)

   a.2: What happens if the cache favors active data too strongly? (give a simple example)

   a.3: Which of your two problems do you think is more problematic for the user? (justify)

   b: To balance this tension, the authors of Coda developed a priority-based caching algorithm. This algorithm takes information about both the user-assigned priority as well as current usage patterns into account when replacing items in the cache. Assume that a file has user-assigned priority P, the time that the file was last accessed is T_a, and the current time is T_c. We want to write a policy function $f(P, T_a, T_c)$ such that $f()$ has the following properties: (1) as the user-assigned priority increases, $f()$ should increase linearly, (2) if the file has been accessed within the last minute but has no user-assigned priority, $f()$ should be equal to 10, (3) if the file is 20 minutes old or older, its age should not affect $f()$ at all, (4) the effect of age on $f()$ between 1 minute and 20 minutes should decrease linearly, and finally, (5) a user-assigned priority of 30 should have the same impact on $f()$ that being accessed within the last minute does.

   Write $f()$ down in closed form or write code that will compute $f()$.

   c.1: Unfortunately in Coda, sometimes the cache could become out of balance, because hoard walks were only performed infrequently. Sketch out pseudo-code of an algorithm that performs a hoard walk, including both the data structures that must be traversed, and the actions that must be taken to restore balance to the cache.

   c.2: How much file and network activity does the restoration described in (c) part 1 above cause to occur? (again make any assumptions that you need to)

3. **Which comes first, A or N? (20 points)** NFS and AFS are two competing distributed file systems, which both provide Unix-like file service to clients, albeit in an entirely different manner internally. In this problem, we explore their differences.

   a.1: Write a single short Unix program that would perform much better on AFS than it would on NFS. That is, assume you have two identical server machines, one running AFS, the other NFS, and both with identical file contents. When you run your program on an AFS client, it should run much faster than the same exact program run on an NFS client. Of course, also assume that such clients are identical, and that network connectivity is identical. In other words, your program should stress aspects of the AFS and NFS *protocols* to achieve the difference.

   For simplicity, assume that you can skip things like header files and such, and just use primitives such as open(), read(), write(), lseek(), and close() to access the file system.

   a.2: How much faster would your program run on AFS than NFS? Show your results analytically, making whatever assumptions you need to in order to complete the question.

   b.1: Now do the same, but write a program that will run much faster on NFS than AFS, with the same assumptions listed above.

   b.2: Again, analyze the difference analytically.

4. **Insecurity (30 points).** Security is hard, but this question shouldn't be.

a.1: You are a network security administrator, and there has been an intruder on your system! This intruder has stolen valuable documents from the company, and it is your job to figure out who did it. Fortunately, you have been tracking many aspects of your system, and you find out that the way in which the data was stolen was a *covert channel.* Somebody has set-up a web server that serves an empty page when you request data from it, but seemingly takes different amounts of time to do so. Thus, when any old user accesses the page, it does not seem to be revealing corporate secrets, but you know better. Describe how you could use this web server and its covert channel to send out the string "hello world" to the Internet (just in case it's not obvious, you are not allowed to simply post a page that says "hello world" on it).

a.2: What makes this covert channel hard to use?

a.3: How can you limit the effectiveness of this covert channel while still allowing access to the web server (or indeed, any network facility) from the outside world?

b.1: After your work in part a), you've discovered that your boss is the thief! In order to get your boss fired (and yourself promoted), you have to tell your boss's boss (the president of the company) about what's happened. Unfortunately, all such communications must be given a stamp of approval by your boss, due to corporate standards and arcane Florida elections law.

Fortunately for you, though, you understand encryption, know how to use a public-key encryption system, and know that your boss will often pass along email with approval even though the boss can't read the contents. Describe how you would use a public-key encryption system to send a message to your boss's boss, subject to the following constraints: your boss must first get and approve the message and your boss must not be able to see the contents of the message, and your boss's boss must know the message is from you and also that your boss has received and approved the message (though not its contents). Assume that the condemning message is labeled M, you are known as E (for Employee) and your keys as E_{SK} and E_{PK} (the first is your secret key and second is your public key), your boss is B, and your boss's boss is P (for president), with their keys known as B_{SK}, B_{PK}, P_{SK}, and P_{PK}, respectively.

b.2: Assuming that no keys have been cached, how much total communication must occur in the above transaction, including all communication with an authentication server that must take place?