Final Exam **COS 432, Fall 2004**

All questions have equal weight. You may use your own lecture notes, copies of other people's lecture notes, and any summaries you have made of those notes. You may not use any other materials.

Some problems may have more than one correct answer, so don't be too worried if you can think of good alternative answers. You must give a single answer. You'll get full credit if your answer is one of the correct ones.

If you can't figure out a complete or perfect answer to a question, tell us what you have figured out. If your answer is flawed, you'll get more credit by showing us that you know the limitations of your answer, rather than pretending not to notice what's wrong with it.

Please do your work in an exam book.

Undergraduates: Before turning in your exam, please hand-write and sign the Honor Code pledge, "I pledge my honor that I have not violated the Honor Code during this examination."

(Graduate students: No need to sign anything. You are bound automatically by university regulations.)

---

1. A group of criminals was arrested recently for bar-code fraud at Wal-Mart stores. The criminals printed stickers with product bar-codes on them. They went into Wal-Mart stores and pasted their stickers over the bar-codes on products. Then they purchased the doctored products, paying the prices associated with the false bar-codes, which were much lower than the real prices. This fraud lasted for a long time and cost Wal-Mart millions of dollars.

   What, if anything, can Wal-Mart do to mitigate this vulnerability? Come up with the best practical suggestion you can, bearing in mind that your goal is to minimize Wal-Mart's total cost (cost of fraud, plus cost of anti-fraud measures).

2. Nowadays many people filter their incoming email to remove (or quarantine) spam. This can be done by individual users on their own computers, or it can be done centrally by their employer or ISP, or both.
   a. What are the advantages of doing it centrally?
   b. What are the advantages of doing it on end-user computers?
   c. Does it make sense to do it in both places? Explain.

3. The U.S. government maintains a "watchlist" containing the names of people who are believed to pose a risk to aviation. Airport security procedures are supposed to ensure that people on the watchlist are subjected to a more intensive "secondary search" at the airport security checkpoint, but unfortunately these procedures are flawed, so that watchlisted people can avoid the secondary search.

   The currently used procedures work as follows. Each passenger gets a boarding pass from an airline check-in desk, an automated check-in kiosk, or on a home computer via the Internet. (Internet boarding passes are generated from a web page, and printed out on the passenger's home computer.) The boarding pass contains the passenger's name and flight information, and a code saying whether the person has been selected for secondary search. All of this information is encoded on the boarding pass in two forms: as English text and as a machine-readable barcode.

   At the airport security checkpoint, the passenger presents an ID and boarding pass to an official who is standing alone with a clipboard. This official verifies that the name printed on the boarding pass matches the name on the ID, and that the ID really belongs to the passenger. (For this problem, ignore the possibility of fake IDs. Assume instead that all IDs are genuine, and the official can always tell whether the ID belongs to the person who is present.) If the printed code on the boarding pass indicates that the passenger should get a secondary search, then that search is done at the checkpoint. The passenger is given back his boarding pass and continues to the gate.

   At the gate, the passenger is again asked to present his boarding pass, and the boarding pass is scanned with a barcode reader, which verifies the name and flight information in the pass's barcode against the airline's ticket database, to ensure that a passenger with that name owns a ticket to fly on that flight.

   This system can be defeated by several attacks. Here is one. Suppose the attacker's real name is Bob Badguy. He buys an airline ticket in the name of "Ivan Innocent". He uses his home computer to make a boarding pass for Ivan. Because Ivan isn't on the watchlist, this boarding pass doesn't contain the code for a secondary search. He prints out this boarding pass, but he also uses a program like Photoshop to make a second, doctored boarding pass, in the name of Bob Badguy. (Since most of it is copied from the Ivan boarding pass, the doctored boarding pass doesn't have the secondary-search code on it.) At the checkpoint, he presents the (fake) Bob Badguy boarding pass, along with his ID. Because the name on the ID matches the name printed on the (fake) boarding pass, the official is satisfied, and the attacker gets through the checkpoint without a secondary search. At the gate, the attacker presents the Ivan Innocent boarding pass, which matches the airline's database. The result is that the attacker gets onto the plane without being subjected to a secondary search.

   Describe how you would change the airport security procedures to eliminate this vulnerability, and other related vulnerabilities, in the watchlist/secondary search procedure described above. The goal is to have a system that (under the assumptions about IDs stated above) ensures that people on the watchlist cannot fly without being subjected to a secondary search. Your design should be as practical as possible, so you'll want to consider the difficulty of switching from the current procedures to the ones you propose.

4. A well-known security principle says that we should strive to minimize the size of the Trusted Computing Base (TCB), which is the software that has to work correctly in order for the system to be trustworthy. For example, in a standard operating system design, the OS kernel, which runs in privileged mode, is all in the TCB, since its misbehavior can defeat any security requirements; but most of the unprivileged code is outside the TCB.

   Some people argue that the TCB must also include the code for the compiler that compiled the code in the TCB, since the compiler can put "back doors" or other security holes into the TCB's executable code. Similarly, the argument goes, the operating system on which the TCB's code was developed should also be included, since it can corrupt the source code or the compiler.

   The same argument can be applied, recursively, to conclude that a long string of compilers (the code that compiled the TCB, the code that compiled the code that compiled the TCB, the code that compiled the code that compiled the code that compiled the TCB, and so on) should be included; and a similar argument would include a long string of operating systems.

   How far should this go? Does it make sense to think of all of this "ancestor" code as part of the TCB? Or does some other way of understanding the role of the ancestor code make more sense? Justify your answer.

   [Note: Even the elder statesmen of computer security disagree on this issue, so when we grade this question we'll be looking more for a well-argued position than for the "correct" conclusion.]

5. Some scholars have proposed an Alternative Compensation System (ACS) for online use of copyrighted music. Under an ACS, a $5 per month tax would be placed on each person's Internet connection. Revenue from the tax would be distributed among musicians, songwriters, and record companies, with each party's revenue being proportional to the number of times its music has been listened to on computers. In exchange for paying the tax, users would get legal permission to store, use, and distribute music online with no restrictions at all.

   One of the key issues in an ACS is how to keep track of how many times each song is listened to. ACS supporters propose putting a program on each user's computer, to monitor which songs are played on that computer. The monitoring program would be approved by some standards body to ensure that it was programmed to accurately record music listening but did not gather any other information about the user's habits.

   a. Why might an end user want to mislead the system about which songs he listens to?
   b. What attacks might an end user use to cause the monitoring program on his computer to report false results?
   c. What countermeasures, if any, should the authorities deploy to prevent end users from carrying out these attacks?
   d. In the "arms race" between users who want to mislead, and the authorities who want to prevent users from misleading the system, which side is likely to have the advantage?