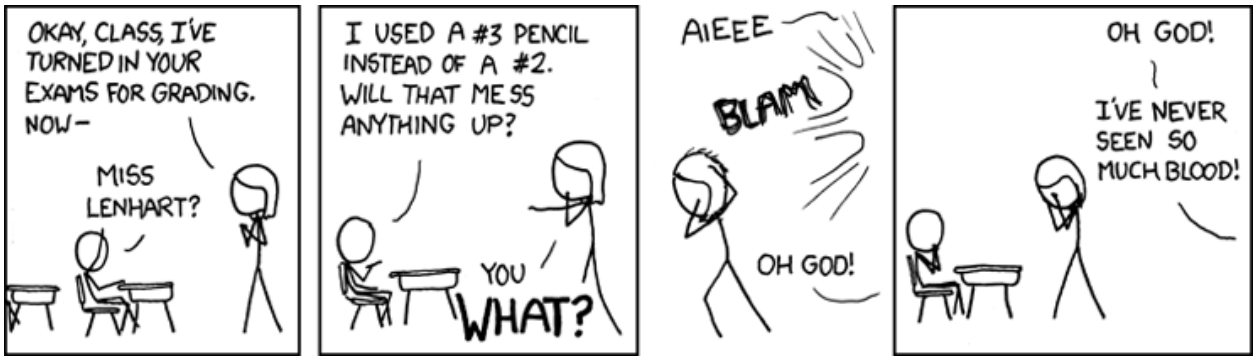
Name: _____ UNI: _____

COMS W4187: Security Architecture and Engineering
October 2014

Rules

- Remember to write your name and UNI on the blue exam book.
- **Important: also write your name on this paper**.
- You must turn in *both* the exam sheet and the blue book
- Books and notes are allowed during this examination; computers are not permitted.
- This is a time-limited test. All papers must be turned in 75 minutes after the beginning of the test.
- Most questions can be answered in just a paragraph or two; if you think you need to write several pages, you're writing too much and may be on the wrong track entirely.
- The total points add up to 70.
- Good luck, and may the Force be with you.

Question	Points	Score
1	15	
2	15	
3	20	
4	20	
Total:	70	



1. (15 points) Someone has built a JPG virus embedded in pictures (of kittens?). The picture triggers a bug in common image display software, thereby infecting the machine. The virus can only infect other JPGs—but JPGs can arrive via email (including encrypted email), social networks, other web sites, flash drives, and more. Both desktops and phones are susceptible to the virus. The virus is highly polymorphic and encrypted; it sometimes changes more rapidly than the virus definitions can be updated.

   What is the best strategy for an organization to stop this malware?

   Answer:

   No one defense will work; you need defense in depth. It enters in many different ways, so a firewall scanner won't suffice; you thus need scanners on each computer. Because it changes rapidly, you need very frequent updates of your antivirus definitions. You also need behavioral detection and scans of existing files, to catch viruses that had gotten in before the definitions were updated.

   If possible, patch the buggy software (but that requires waiting for the vendor.)

   Install a converter program that isn't vulnerable to convert the JPGs to PNG or some other such format.

   Not quite acceptable: "Use a different viewer". There are too many programs that display JPGs; quite likely, they all use the same renderer.

2. (15 points) To aid in generating random cryptographic keys, every employee of a company has a mag stripe card and a keyboard with a mag stripe *reader*. To generate a new key, the employee swipes the card; a secret value from the card is cryptographically combined with the time of day. (Assume that the combination algorithm is cryptographically correct.) Outline some possible attacks against this system. If you need to make any assumptions, state them explicitly—and make sure that they're plausible.

   Answer:

   A keystroke logger can capture the key as it's being read in. Anyone with even brief access to the card can copy it. Alternatively, the stripe could be overwritten with a new, known value.

   An employee could swipe a different, and perhaps more convenient card; it may not have enough random digits.

   IF the time is controlled by, say, NTP or other network protocol, the attacker can control the time and replay previous values.

   Note: I confess that when I wrote "time of day", I was thinking of the Unix gettimeofday() system call, which returns the amount of time, in microseconds, since the "epoch"—00:00:00 Jan 1, 1970, UTC. This means that someone doing something at the same time each day isn't really a threat. However, that didn't really affect anything in the grading.

3. (20 points) Consider the design of a distributed computer system where the components are connected by an ordinary network and the user workstations have *no* privileged operations. Effectively, each user is `root` or `Administrator` on his or her machine. Instead, there are a few privileged computers that implement the file system, etc. Give at least two of the security challenges of this design? How would you solve these problems?

Answer:

This is a message-passing system, with all that implies. You can't trust anything coming over the network, and in particular you can't believe any assertions of identity, so you have to rely on cryptographic authentication. Also, since this is all network-based, you need to be very careful about parsing input, to avoid standard network attacks.

There are also network-based threats, such as assorted ways to divert traffic or be a "man in the middle". These are defeated by bilateral authentication: the user has to authenticate the server, and vice-versa, before any sensitive information is sent. Standard cryptographic techniques can be used.

4. You're designing the authentication system for a multi-level secure system. The system will hold data at all levels from ⟨**Unclassified**⟩ to ⟨**Top Secret**⟩. (For convenience, I've included the lattice diagram from class.)

   (a) (10 points) One proposal is to use passwords; that, though, requires a password file. One group suggests labeling the password file as ⟨**High**, PW⟩; another group suggests ⟨**Low**, PW⟩. Give one advantage of each possibility.

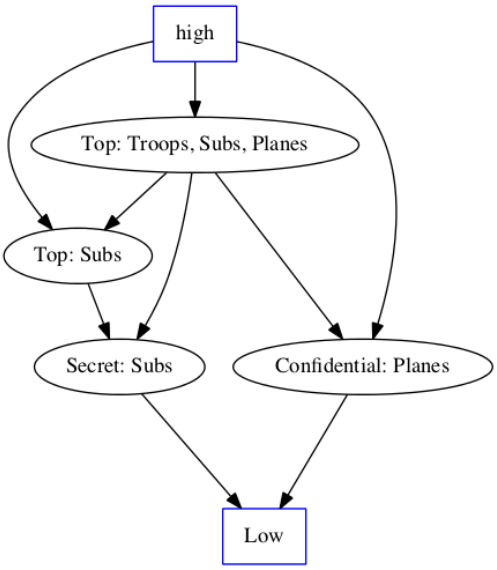
   Answer:
   The "no read up" mechanism means that no untrusted applications can read a ⟨**High**, PW⟩ file, thus protecting your password file. The "no write down" property, on the other hand, prevents any applications from writing new entries to a password file labeled ⟨**Low**, PW⟩.

   (b) (10 points) A third group suggests that passwords alone are a bad idea and that they should be supplemented with a time-based token such as a SecurID card. These cards require a database of keys, one per card. Again, give one advantage of each of the choices above.

Answer:

The answers are basically the same. However, "no read up" is more important here than for passwords, because what would be exposed is the actual keys, rather than just hashed passwords.

The replay prevention aspect of the SecurID tokens is not relevant to the question.


high
Top: Troops, Subs, Planes
Top: Subs
Secret: Subs
Confidential: Planes
Low