Outline: Lecture 8

Date: April 21, 2011
Topic: Secure Stuff: What To Look For

1. Homework discussion
2. What does "secure" mean?
3. What do you look for?
4. Basic requirements
   a. Paranoia
   b. Defending against stupidity
   c. Showing only that which the user needs to see
   d. Assume anything can happen, and guard against undesirable things
5. What does the program depend on?
   a. Network access—what happens if it can't connect to the network?
   b. User settings—are these easy to do?
   c. Files—what files (intermediate, input, output, does it use?
   d. How does it handle contradictory settings?
   e. Other dependencies, especially on what the user/system/administration does not control?
6. Does the program do what you expect?
   a. Is it clear what the program is to do under all circumstances?
   b. What happens when you give it lots of input or use it on large data sets?
   c. What happens if you give it *no* input when it expects some?
   d. What happens if you try to exceed some limit?
7. What happens if you give it strange input?
   a. Does it handle "meta-characters" properly?
   b. Does it check for and handle bad characters, or does it check for good characters?
   c. What happens if the input is malformed?
8. Does it interact with other programs?
   a. What happens if the other program is not present?
   b. What happens if it malfunctions?
   c. Will the programs deadlock?
   d. Does the result depend on the order in which the programs interact—and if so, is that ordering enforced?
9. What does it do if something "impossible" happens?
   a. A system database returns an unexpected value (or no value)
   b. A network connection is broken before it shuts down
   c. A configuration file or database is corrupted
10. Tools for analysis
    a. Static analysis
    b. Dynamic analysis (testing)
    c. Penetration testing