Secure software development requires you know how to design, build, test and document with security in mind. Secure software is able to withstand the most malicious attacks. Secure software is by nature robust software.

Chapter 1: The Need for Secure Systems

1. All software is vulnerable to attack, regardless of the purpose for which it was developed.
    a. Never assume your app will be run in only a few (or one) environment
    b. Never assume your app will not persist
2. Honeypot: a computer set up to attract hackers to see how the hackers operate
    a. http://project.honeynet.org
    b. Catches script kiddies, especially
    c. Side discussion: proxy – what is it and why is it important
    d. Great for testing concepts but be careful not to disclose too much of the real thing
3. Security should be a top priority, not an afterthought based on an attack
    a. If your software is attacked and is vulnerable and this knowledge goes public, what are the consequences for your company?
    b. Security after the fact is expensive
        i. Coordination of the fix
        ii. Developers must find vulnerable code (think how difficult this could be with a large product)
        iii. Code must be fixed
        iv. Code must be tested (and regression tested)
        v. Setup of fix must be tested (patch must be tested)
        vi. International versions may be necessary
        vii. Code must be digitally signed for authenticity
        viii. Code must be posted to website
        ix. Documentation for fix must be created
        x. Public relations
        xi. Bandwidth costs for download
        xii. Loss of productivity
        xiii. Customers must apply fix
            1. aside: patches, those who apply and those who don't and the ramifications
        xiv. Lost revenue due to bad press
4. Secure products are quality products
    a. Well tested
    b. Robust
    c. Fail securely
5. Why should a discovered bug always be fixed ASAP?
    a. If it is serious, entire program and/or system on which it runs can be compromised

   b. Costs associated with security after the fact!
   c. Some products are shipped with known bugs due to time-to-market pressure
6. A secure code developer should do the following
   a. Stay abreast of security issues in industry
      i. http://www.securityfocus.com
      ii. http://slashdot.org
      iii. http://www.alw.nih.gov/Security/security.html
   b. Make others aware of security issues
   c. Work with others to determine severity of security bugs and offer advice on fixes
   d. Have the ability to think like (and even act like) an attacker
   e. Be able to apply security theory in appropriate ways to practically mitigate security threats
   f. Attend conferences that discuss latest security trends/ issues
7. Attacker's Advantage / Defender's Dilemma
   a. Defender must defend all points; the attacker can choose the weakest point
      i. Aside: how to know what points must be defended (STRIDE, DREAD)
   b. Defender can defend against only know attacks (maybe); attacker can probe for unknown vulnerabilities
   c. Defender must be constantly vigilant; attacker can strike at will
   d. Defender must play by the rules (for the most part); attacker can play dirty