Outline for April 1, 2004

1. Basic components
    a. Confidentiality
    b. Integrity
    c. Availability
2. Threats
    a. Snooping
    b. Modification
    c. Masquerading; contrast with delegation
    d. Repudiation of origin
    e. Denial of receipt
    f. Delay
    g. Denial of service
3. Role of policy
    a. Example of student copying files from another
    b. Emphasize: policy *defines* security
    c. Distinguish between policy and mechanism
4. Goals of security
    a. Prevention
    b. Detection
    c. Recovery
5. Trust
    a. Hammer this home: all security rests on trust
    b. First problem: security mechanisms correctly implement security policy; walk through example of a program that logs you in; point out what is trusted
    c. Second problem: policy does what you want; define secure, precise
6. Operational issues; change over time
    a. Cost-benefit analysis
    b. Risk analysis (comes into play in cost-benefit too)
    c. Laws and customs
7. Human Factors
    a. Organizational problems
    b. People problems (include social engineering)
8. Principles of Secure Design
    a. Refer to both designing secure systems and securing existing systems
    b. Speaks to limiting damage
9. Principle of Least Privilege
    a. Give process only those privileges it needs
    b. Discuss use of roles; examples of systems which violate this (vanilla UNIX) and which maintain this (Secure Xenix)
    c. Examples in programming (making things setuid to root unnecessarily, limiting protection domain; modularity, robust programming)
    d. Example attacks (misuse of privileges, etc.)
10. Principle of Fail-Safe Defaults
    a. Default is to deny

b. Example of violation: *su* program