Outline for March 8, 1999

1. Greetings and felicitations!
   a. Should have homework 1 and 2 back, along with project parts 1 and 2 (if you handed them in)?
2. Penetration Exercises
   a. MTS
   b. Burroughs
3. Vulnerabilities Databases
   a. Federated or centralized?
   b. What should they contain?
   c. Use: basis forFlaw Hypothesis step of FHM
   d. Use: basis for testing classification schemes
4. Exploit Databases
   a. Federated or centralized?
   b. Difference between them and vulnerabilities databases
   c. Use: analysis of detritus
5. Avoiding Vulnerabilities
   a. Good programming design (eight rules follow; Saltzer and Schroeder)
   b. Good implementation practise (more next week)
6. Principles of Secure Design
   a. Refer to both designing secure systems and securing existing systems
   b. Speaks to limiting damage
7. 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.)
8. Principle of Fail-Safe Defaults
   a. Default is to deny
   b. Example of violation: *su* program
9. Principle of Economy of Mechanism
   a. KISS principle
   b. Enables quick, easy verification
   c. Example of complexity: *sendmail*
10. Principle of Complete Mediation
    a. All accesses must be checked
    b. Forces system-wide view of controls
    c. Sources of requests must be identified correatly
    d. Source of problems: caching (because it may not reflect the state of the system correctly); examples are race conditions, DNS poisoning
11. Principle of Open Design
    a. Designs are open so everyone can examine them and know the limits of the security provided
    b. Does ***not*** apply to cryptographic keys
    c. Acceptance of reality: they can get this info anyway
12. Principle of Separation of Privilege
    a. Require multiple conditions to be satisfied before granting permission/access/*etc*.
    b. Advantage: 2 accidents/errors/*etc*. must happen together to trigger failure

13. Principle of Least Common Mechanism
    a. Minimize sharing
    b. New service: in kernel or as a library routine? Latter is better, as each user gets their own copy
14. Principle of Psychological Acceptability
    a. Willingness to use the mechanisms
    b. Understanding model
    c. Matching user's goal
15. Auditing
    a. Goals: reconstruction or deduction?
    b. Relationship to security policy
    c. Application logs
    d. System logs
16. Example analysis technique
    a. GOAL methodology
    b. Do it on local file accesses
17. Problems
    a. Log size
    b. Impact on system services
    c. Correllation of disparate logs