Outline for May 27, 2004

1. Life cycle: Waterfall Model
   a. Requirements definition and analysis
   b. System and software design (system design, program design)
   c. Implementation and unit testing
   d. Integration and system testing
   e. Operation and maintenance
2. Other life cycle models
   a. Exploratory programming
   b. Prototyping
   c. Formal transformation
   d. System assembly from reusable components
   e. Extreme programming
3. Auditing
   a. Definitions
4. Architecture
   a. Logger
   b. Analyzer
   c. Notifier
5. *A Priori* Design
   a. Goal: determine what information indicates violation of security policy
   b. Implementation considerations
   c. Syntactic issues
   d. Sanitizing the logs
   e. System and application logging
6. *A Posteriori* Design
   a. Goal: detect any violations of policy, or actions known to be part of such an attempt
   b. Detect violations of a known policy
      i. State-based auditing
      ii. Transition-based auditing
   c. Detect known violations of a policy