Outline for January 24, 2008

1. Expressive power
   a. HRU vs. SPM
   b. Multiparent joint creates in HRU
   c. Adding multiparent joint creates to SPM (giving ESPM)
   d. Simulation of multiparent joint creates by 2-parent joint creates
   e. Monotonic ESPM, monotonic HRU equivalent
   f. Safety question in ESPM decidable if acyclic attenuating scheme
2. Comparing Expressive Power of Models
   a. Graph representation
   b. Simulate 3-parent joint create using 2-parent joint create
   c. Correspondence between two schemes in terms of graph representation
   d. Formal definition of scheme A simulating scheme B
   e. Model expressive power
   f. Result: monotonic 1-parent models less expressive than monotonic multiparent models (so ESPM more expressive than SPM)
3. Typed Access Matrix Model
   a. Add notion of type for entities—set of types T, set of subject types TS $\subseteq$ T
   b. New create rules: specify subject/object type
   c. Safety decidable for systems with acyclic MTAM schemes
4. Security policies and mechanisms
   a. Policy vs. mechanism
   b. Secure, precise
   c. Observability postulate
   d. Theorem: for any program p and policy c, there is a secure, precise mechanism m^* such that, for all security mechanisms m associated with p and c, $m^* \approx m$
   e. Theorem: There is no effective procedure that determines a maximally precise, secure mechanism for any policy and program
5. Bell-LaPadula Model: intuitive, security classifications only
   a. Show level, categories, define clearance and classification
   b. Lattice: poset with $\leq$ relation reflexive, antisymmetric, transitive; greatest lower bound, least upper bound
   c. Apply lattice
      i. Set of classes *SC* is a partially ordered set under relation *dom* with *glb* (greatest lower bound), *lub* (least upper bound) operators
      ii. Note: *dom* is reflexive, transitive, antisymmetric
      iii. Example: (A, C) *dom* (A', C') iff $A \leq A'$ and $C \subseteq C'$; $lub((A, C), (A', C')) = (max(A, A'), C \cup C')$, $glb((A, C), (A', C')) = (min(A, A'), C \cap C')$
   d. Simple security condition (no reads up), *-property (no writes down), discretionary security property
   e. Basic Security Theorem: if it is secure and transformations follow these rules, it will remain secure
   f. Maximum, current security level