

Outline for January 26, 2007

1. Greetings and Felicitations!
2. 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
3. Bell-LaPadula Model: intuitive, security classifications only
 - a. Security clearance, classification
 - b. Simple security condition (no reads up), *-property (no writes down), discretionary security property
 - c. Basic Security Theorem: if it is secure and transformations follow these rules, it will remain secure
4. Bell-LaPadula Model: full model
 - a. Show categories, refine 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
5. BLP: formally
 - a. Elements of system: s_i subjects, o_i objects
 - b. State space $V = B \times M \times F \times H$ where:
 - B set of current accesses (i.e., access modes each subject has currently to each object);
 - M access permission matrix;
 - F consists of 3 functions: f_s is security level associated with each subject, f_o security level associated with each object, and f_c current security level for each subject;
 - H hierarchy of system objects, functions $h: O \rightarrow \mathcal{P}(O)$ with two properties:
 - i. If $o_i \neq o_j$, then $h(o_i) \cap h(o_j) = \emptyset$
 - ii. There is no set $\{o_1, \dots, o_k\} \subseteq O$ such that for each i , $o_{i+1} \in h(o_i)$ and $o_{k+1} = o_1$.