Outline for February 24, 2012

Reading: §8

1. Composing deterministic, noninterference-secure systems
2. Nondeducibility
   a. Event system
   b. Deducibly secure
   c. Composing deducibly secure systems
3. Generalized noninterference
   a. Assumptions and nondeducibility
   b. Composing generalized noninterference systems
   c. Feedback-free systems
4. Restrictiveness
   a. State machine model
   b. Composing restrictive systems

Table of Notation

notation	*meaning*
S	set of subjects s
Σ	set of states σ
O	set of outputs o
Z	set of commands z
C	set of state transition commands (s, z), where subject s executes command z
C^*	set of possible sequences of commands $c_0, \ldots, c_{n_i}$
ν	empty sequence
c_s	sequence of commands
$T(c, \sigma_i)$	resulting state when command c is executed in state σ_i
$T^*(c_s, \sigma_i)$	resulting state when command sequence c_s is executed in state σ_i
$P(c, \sigma_i)$	output when command c is executed in state σ_i
$P^*(c_s, \sigma_i)$	output when command sequence c_s is executed in state σ_i
$proj(s, c_s, \sigma_i)$	set of outputs in $P^*(c_s, \sigma_i)$ that subject s is authorized to see
$\pi_{G,A}(c_s)$	subsequence of c_s with all elements (s, z), $s \in G$ and $z \in A$ deleted
$dom(c)$	protection domain in which c is executed
$\sim^{dom(c)}$	equivalence relation on system states
$\pi'_d(c_s)$	analogue to π above, but with protection domain and subject included
w_n	$v_1, ..., v_n$ where $v_i \in C^*$
w	sequence of elements of C leading up to current state
$cando(w, s, z)$	true if s can execute z in current state
$pass(s, z)$	give s right to execute z
$prev(w_n)$	w_{n-1}
$last(w_n)$	v_n