ECS 235B Module 37
Generalized Noninterference

Policies Changing Over Time

- Problem: previous analysis assumes static system
    - In real life, ACM changes as system commands issued
- Example: $w \in C^*$ leads to current state
    - *cando*(w, s, z) holds if s can execute z in current state
    - Condition noninterference on *cando*
    - If $\neg$*cando*(w, Lara, "write f"), Lara can't interfere with any other user by writing file f

Generalize Noninterference

- $G \subseteq S$ set of subjects, $A \subseteq Z$ set of commands, p predicate over elements of C^*
- $c_s = (c_1, \dots, c_n) \in C^*$
- $\pi''(\nu) = \nu$
- $\pi''((c_1, \dots, c_n)) = (c_1', \dots, c_n')$, where
  - $c_i' = \nu$ if $p(c_1', \dots, c_{i-1}')$ and $c_i = (s, z)$ with $s \in G$ and $z \in A$
  - $c_i' = c_i$ otherwise

Intuition

- $\pi''(c_s) = c_s$
- But if p holds, and element of c_s involves both command in A and subject in G, replace corresponding element of c_s with empty command ν
  - Just like deleting entries from c_s as $\pi_{A,G}$ does earlier

Noninterference

- G, $G' \subseteq S$ sets of subjects, $A \subseteq Z$ set of commands, p predicate over C^*
- Users in G executing commands in A are *noninterfering with users in* G' under condition p iff, for all $c_s \in C^*$ and for all $s \in G'$, $proj(s, c_s, \sigma_i) = proj(s, \pi''(c_s), \sigma_i)$
  - Written $A,G :| \; G'$ **if** p

Example

- From earlier one, simple security policy based on noninterference:

$$\forall(s \in S)\ \forall(z \in Z)\ [\ \{z\}, \{s\} :|\ S\ \textbf{if}\ \neg \mathit{cando}(w, s, z)\ ]$$

- If subject can’t execute command (the $\neg$*cando* part) in any state, subject can’t use that command to interfere with another subject

Another Example

- Consider system in which rights can be passed
  - *pass*(s, z) gives s right to execute z
  - $w_n = v_1, \ldots, v_n$ sequence of $v_i \in C^*$
  - *prev*(w_n) = w_{n-1}; *last*(w_n) = v_n

Policy

- No subject *s* can use *z* to interfere if, in previous state, *s* did not have right to *z*, and no subject gave it to *s*

  $\{ z \}, \{ s \} :| S$

  $$\textbf{if } [\neg cando(prev(w), s, z) \wedge [cando(prev(w), s', pass(s, z)) \Rightarrow \neg last(w) = (s', pass(s, z))]]$$

Effect

- Suppose $s_1 \in S$ can execute $pass(s_2, z)$
- For all $w \in C^*$, $cando(w, s_1, pass(s_2, z))$ holds
- Initially, $cando(\nu, s_2, z)$ false
- Let $z' \in Z$ be such that (s_3, z') noninterfering with (s_2, z)
  - So for each w_n with $v_n = (s_3, z')$, $cando(w_n, s_2, z) = cando(w_{n-1}, s_2, z)$

Effect

- Then policy says for all $s \in S$

  $proj(s, ((s_2, z), (s_1, pass(s_2, z)), (s_3, z'), (s_2, z)), \sigma_i) =$

  $proj(s, ((s_1, pass(s_2, z)), (s_3, z'), (s_2, \mathrm{z})), \sigma_i)$
- So s_2's first execution of z does not affect any subject's observation of system

Quiz

How does policies changing over time affect the definition of noninterference?

1. The definition does not change.
2. The definition adds that the system is noninterfering under some condition.
3. The definition is changed to delete commands that occur before the change.
4. The definition is changed to delete users that issue no commands before the change.