Lecture 2 Outline

Reading: text, §2, 3.1

1. Access control matrix and entities
   a. Subject, objects (includes subjects)
   b. State is \((S, O, A)\) where \(A\) is access control matrix

2. Primitive operations
   a. \textit{enter} \(r\) into \(A[s, o]\)
   b. \textit{delete} \(r\) from \(A[s, o]\)
   c. \textit{create subject} \(s\) (note that \(\forall x \ [A[x, s'] = A[x, s] = \emptyset]\))
   d. \textit{create object} \(o\) (note that \(\forall x \ [A[x, o'] = \emptyset]\))
   e. \textit{destroy subject} \(s\)
   f. \textit{destroy object} \(o\)

3. Commands and examples
   a. Regular command: \textit{create}•\textit{file}
   b. Mono-operational command: \textit{make}•\textit{owner}
   c. Conditional command: \textit{grant}•\textit{rights}
   d. Biconditional command: \textit{grant}•\textit{read}•\textit{if}•\textit{r}•\textit{and}•\textit{c}
   e. Doing “or” of 2 conditions: \textit{grant}•\textit{read}•\textit{if}•\textit{r}•\textit{or}•\textit{c}
   f. General form

4. Miscellaneous points
   a. Copy flag and right
   b. Own as a special right
   c. Principle of attenuation of privilege

5. What is the safety question?
   a. An unauthorized state is one in which a generic right \(r\) could be leaked into an entry in the ACM that did not previously contain \(r\). An initial state is safe for \(r\) if it cannot lead to a state in which \(r\) could be leaked.
   b. Question: in a given arbitrary protection system, is safety decidable?

6. Mono-operational case: there is an algorithm that decides whether a given mono-operational system and initial state is safe for a given generic right.

7. General case: It is undecidable whether a given state of a given protection system is safe for a given generic right.
   a. Approach: represent Turing machine tape as access control matrix, transitions as commands
   b. Reduce halting problem to it