Outline for January 19, 1999

1. Greetings and felicitations!
   a. *lassen* should be reachable from anywhere in the department. I'm working on having it recognize the rest of the Internet!
2. ACM and primitive operations
   a. Go over subjects, objects (includes subjects), and state (S, O, A) where A is ACM
   b. Transitions modify ACM entries; primitive operations follow
   c. **enter** *r* **into** $A[s,o]$
   d. **delete** *r* **from** $A[s,o]$
   e. **create subject** *s'* (note $A[s',x] = A[x,s'] = \emptyset$ for all x)
   f. **create object** *o'* (note $A[x,o'] = \emptyset$ for all x)
   g. **destroy subject** *s'*
   h. **destroy object** *o'*
3. commands
   a. **command** $c(s_1, ..., s_k, o_1, ..., o_k)$
      if r_1 **in** $A[s_1, o_1]$ **and**
      r_2 **in** $A[s_2, o_2]$ **and**
      ...
      r_m **in** $A[s_m, o_m]$
      then
      op_1;
      op_2;
      ...;
      op_n;
      end.
   b. Example 1: creating a file
      command *create_file*(p, f)
      create object *f*;
      enter *Own* **into** $A[p, f]$
      enter *Read* **into** $A[p, f]$
      enter *Write* **into** $A[p, f]$
      end.
   c. Example 2:granting one process read rights to a file
      command *grant_read*(p, q, f)
      if *Own* **in** $A[p, f]$
      then
      enter *Read* **into** $A[q, f]$
      end.
4. 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?
5. Mono-operational protection systems: decidable
   a. Theorem: there is an algorithm that decides whether a given mono-operational system and initial state is safe for a given generic right.
   b. Proof: finite number of command sequences; can eliminate **delete** , **destroy** .
      Ignore more than one **create** as all others are conditioned on access rights in the matrix. (One exception: no subjects; then we need one **create subject**).
      Bound: *s* number of subjects (possibly one more than in original), *o* number of objects (same), *g* number of generic rights; number of command sequences to inspect is at most 2^{gso}.

6. General case: It is undecidable whether a given state of a given protection system is safe for a given generic right.
    a. Represent TM as ACM; reduce halting problem to it
7. Take-Grant