ECS 235B Module 5
Attribute-Based Access Control Matrix
Attributes

• *attribute*: variable of a specific data type associated with an entity

• *att(o)*: set of attribute values associated with *o*, called the *attribute value tuple* of *o*
  - Each attribute is written *o.a_i*, with value v drawn from set *Va_i*

• *attribute predicate*: boolean expression built from attributes and constants with appropriate operation and relation symbols
  - Unary predicate: built from one attribute
  - Binary predicate: built from two attributes
  - Can have as many attributes in a predicate as needed
  - Example: *Alice.credit* ≥ $100.00
Attribute Based Access Control Matrix (ABAM)

• Change access control matrix so rows correspond to subjects and their attributes, and columns correspond to objects and their attributes

• Note access control matrix discussed previously is special case
  • Just make the attribute sets be empty
Primitive Operations

- **enter, delete** as before
- **create subject** \( s \) **with attribute tuple** \( att(s) \): create subject \( s \) with given attribute tuple; additionally, add an identity attribute with a unique value
- **create object** \( o \) **with attribute tuple** \( att(o) \): create object \( o \) with given attribute tuple; additionally, add an identity attribute with a unique value
- **destroy** as before except it also deletes the associated attribute tuple
- **update attribute** \( o.a_i \): update \( att(o) = (v_1, ..., v_i, ..., v_n) \) to \( att(o)' = (v_1, ..., v_i', ..., v_n) \), where \( v_i, v_i' \in Va_i \), and \( v_i \neq v_i' \)
Commands

• Like previous commands, except that conditions may include attribute predicates
• Let $p$ give $q r$ rights over $f$, if $p$ owns $f$ and value of $p$’s attribute jobcode is between 3 and 5 inclusive

\[
\text{command grant}\cdot\text{read}\cdot\text{file}\cdot\text{attribute}\cdot\text{3to5}(p, f, q)
\]

\[
\text{if own in } A[p, f] \text{ and } 3 \leq p.jobcode \text{ and } p.jobcode \leq 5
\]

\[
\text{then enter } r \text{ into } A[q, f];
\]

end