1. Recap
   a. DAC
      i. Academic in origin, but messy when scaled and prone to trojan horses
   b. MAC
      i. Why “mandatory” in MAC?
         1. Military likes this word.
         2. Mostly for military.
      ii. MAC rules
         1. No reading up (simple security)
         2. No writing down (star property)
      iii. Problem with MAC?
         1. Can be too strict, especially with writing below privilege.
         2. Eg owner of a restaurant isn’t allowed to write recipes.
         3. Need something more flexible than MAC or DAC.

2. Role-based Access Control (RAC)
   a. Designed around roles
      i. Eg jobs in an organization
   b. Roles are created and assigned to users
   c. Implementation using an access matrix
      i. Following tables illustrate an example of an implementation. In the users and roles table the ‘X’ s indicate that a user has that role and in the roles and objects the entries are permissions that role has on that object.

### Users and roles

<table>
<thead>
<tr>
<th>Users and roles</th>
<th>Role(_1)</th>
<th>Role(_2)</th>
<th>(\ldots)</th>
<th>Role(_m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>User(_1)</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>User(_2)</td>
<td></td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(\ldots)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>User(_n)</td>
<td></td>
<td></td>
<td>X</td>
<td></td>
</tr>
</tbody>
</table>
Roles and objects

<table>
<thead>
<tr>
<th>Role (i)</th>
<th>Object_1</th>
<th>Object_2</th>
<th>\ldots</th>
<th>Object_m</th>
</tr>
</thead>
<tbody>
<tr>
<td>Write</td>
<td>Read</td>
<td></td>
<td></td>
<td>Execute</td>
</tr>
<tr>
<td>Role_2</td>
<td>Stop</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>\ldots</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Role_n</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- One user can have different roles in different sessions
- Roles can be assigned to other roles (hierarchy)
- Eg. System programmers are allowed to access the data center while web developers are not
- Benefits
  - Scalability
  - Flexibility
- Weaknesses
  - Hard to distinguish individuals with the same role.

3. Attribute Based Access Control (ABAC)

- Permissions are based on attributes of user
- Access Control Policy is composed of three things
  - Subject attributes
    - Eg. For an academic department, this might be things like name, advisor, education level, etc
  - Object attributes
  - Environmental conditions
    - Eg. Time of day
- Benefits
  - Very flexible
  - Expression
- Weaknesses
  - Slow due to predicate evaluation for each access
4. Open Identity Trust Framework
   a. OpenID
      i. Often used when you sign in a website via Facebook / Google account.
      ii. Happens before OAuth
   b. OpenID Foundation
      i. International, nonprofit, organization dedicated to OpenID technologies
   c. ICF: Digitize ID cards (driver’s license, etc.)
      i. In June 2008, Equifax, Google, Microsoft, Novell, Oracle Corporation, PayPal and others created the Information Card Foundation to promote the Information Card metaphor.
   d. OIX: Open Identity Exchange Corporation
      i. Neutral, international organization
   e. AXN: Attribute Exchange Network
      i. Go-between for identity providers and relying parties.

5. Database Security
   a. Databases are a common attack target since they contain so much sensitive data
   b. Relational databases
      i. Centered on "relations" (tables) that have "tuples" (rows, records) and "attributes" (columns, fields).
      ii. Each tuple is uniquely identified by a primary key.
      iii. A table can have a primary key in its attributes. This is called a foreign key.
   c. Structured Query Language (SQL)
      i. A standard language to manipulate the database.

Questions
1. What are some attributes are commonly used in ABAC scenario?
   a. Role for the subjects and function for the objects.
2. How similar will the questions on the midterm be to the questions on the quizzes?
   a. Some questions on the midterm will be similar to the quizzes. There will also be additional material
3. Is system availability an example of an environment attribute?
   a. Can be depending on the situation.
   b. For example, the access may be rejected because the NTP server is down.
4. How do roles differ from groups in Linux?
   a. More semantics are attached to roles
5. Are all websites required to follow the open identity trust framework?
   a. No
6. Wouldn't the open identity trust framework just add more points of vulnerability?
   a. Yes
7. Is the function of ABAC a bit like that of RBAC with a lot more (and highly granular) roles?
   a. Exactly!

8. Who is responsible for creating the descriptions for ABAC? Do the users create the attributes for the objects while they are creating them?
   a. It's a chicken egg problem.

9. How popular is each access control method?
   a. RBAC is primarily used, but ABAC is still considered okay. On Linux, DAC is most commonly used.