RBAC-MAC-DAC

Prof. Ravi Sandhu

RBAC96

ROLE HIERARCHIES

USER-ROLE ASSIGNMENT

PERMISSIONS-ROLE ASSIGNMENT

USERS

ROLES

PERMISSIONS

SESSIONS

CONSTRAINTS
WHAT IS THE POLICY IN RBAC?

- RBAC is policy neutral
  - Role hierarchies facilitate security management
  - Constraints facilitate non-discretionary policies
RBAC96: LIBERAL *-PROPERTY

- user $\in xR$, user has clearance $x$
  user $\in LW$, independent of clearance
- Need constraints
  - session $\in xR$ iff session $\in xW$
  - read can be assigned only to $xR$ roles
  - write can be assigned only to $xW$ roles
  - $(O,\text{read})$ assigned to $xR$ iff
    $(O,\text{write})$ assigned to $xW$

LBAC: STRICT *-PROPERTY

- Diagram with roles M1, M2, and clearance levels H, L, and access levels Read, Write.

© Ravi Sandhu 2001
RBAC96: STRICT *-PROPERTY

Variations of DAC

- Strict DAC
- Liberal DAC
Strict DAC

- Only owner has a discretionary authority to grant access to an object.
- Example:
  - Alice has created an object (she is owner) and grants access to Bob. Now Bob cannot grant propagate the access to another user.

Liberal DAC

- Owner can delegate discretionary authority for granting access to other users.
  - One Level grant
  - Two Level Grant
  - Multilevel Grant
One Level Grant

- Owner can delegate authority to another user but they cannot further delegate this power.

Alice ➔ Bob ➔ Charles

Two Level Grant

- In addition a one level grant the owner can allow some users to delegate grant authority to other users.

Alice ➔ Bob ➔ Charles ➔ Dorothy
Revocation

- Grant-Independent Revocation.
- Grant-Dependent Revocation.

Common Aspects

- Creation of an object in the system requires the simultaneous creation of
  - three administrative roles
    - OWN_O, PARENT_O, PARENTwithGRANT_O
  - One regular role
    - READ_O
We require simultaneous creation of Eight Permissions
- canRead_O
- destroyObjet_O
- addReadUser_O, deleteReadUser_O
- addParent_O, deleteParent_O
- addParentWithGrant_O,
  deleteParentWithGrant_O
Roles and associated Permissions

- OWN_O
  - destroyObject_O, addParentWithGrant_O, deleteParentWithGrant_O
- PARENTwithGRANT_O
  - addParent_O, deleteParent_O
- PARENT_O
  - addReadUser_O, deleteReadUser_O
- READ_O
  - canRead_O

Common Aspects III

- Destroying an object O requires deletion of four roles and eight permissions in addition of destroying the object O.
Strict DAC in RBAC96

- Cardinality constraints as:
  - Role OWN_O = 1
  - Role PARENTwithGRANT_O = 0
  - Role PARENT_O = 0

One level DAC in RBAC96

- Cardinality constraints as:
  - Role OWN_O = 1
  - Role PARENTwithGRANT_O = 0
Two Level DAC in RBAC96

- **Cardinality constraints as:**
  - Role $\text{OWN}_O = 1$

![Diagram showing relationships between roles](image)

- **READ\_O role associated with members of PARENT\_O**