Security Enhanced Administrative Role Based Access Control Models

Rajkumar P V and Ravi Sandhu

System Security Administration

Security Abstractions in RBAC
- ✓ Role Based Access Control (RBAC) provides application level security abstractions: *roles, permissions*, and *users*.
- ✓ Naturally fits in administration of organizational information systems.
- ✓ Practically implemented in Operating Systems, Database Management Systems, Enterprise Resources Planning Applications, and Cloud Services.

Accountability in RBAC Administration
- ✓ Accountability of system administrators is an important factor in system security.
- ✓ Most of the existing administrative models for RBAC does not have sufficient *monitoring* features.
- ✓ Enforcing accountability of RBAC administrators requires additional mechanisms.

Administrative Obligations

Preliminary Idea
- ✓ Systematic integration of obligations into RBAC administration would help in enforcing accountability of RBAC administrators and enhancing overall system security.

An Example

Policy
- ✓ Whenever an administrator adds a new user to *"Backup_and_Recovery"* role he must report to his co-admins

SE-ARBAC

assign = < au,u, *"Backup_and_Recovery"*, *Report(au,u x Backup_and_Recovery,* $\forall au': au' \in \{SA - au\}$}>
- ✓ *Report* is an administrative obligation
- ✓ *au, u, SA* are administrative user, application user, and set of System Administrators respectively.
- ✓ *"Backup_and_Recovery"* is a role name.
- ✓ Segregates having administrative rights over roles and executing them.
- ✓ In a decentralized administration of RBAC, Co-admins may be from different organizations.

Importance and Future Directions

- ✓ Co-admin accountability checking gains importance as the cloud and mobile services increase their presence in practical business applications.
- ✓ Further study is required on designing precise administrative obligations for different application contexts.
- ✓ Development of supportive tools for accountability checking.