Foundations of Computer Security

Lecture 26: Role-Based Access Control

Dr. Bill Young
Department of Computer Sciences
University of Texas at Austin

Role-Based Access Control

Role-based access control (RBAC) is a widely used security framework claimed to be especially appropriate for commercial settings.

Unlike access control policies that assign permissions to subjects, RBAC associates permissions with functions/jobs/roles within an organization.

A *role* is a collection of job functions. Roles within a bank might include: `president`, `manager`, `trainer`, `teller`, `auditor`, `janitor`, etc.

Roles and Transactions

An individual has:

- a set of *authorized roles*, which it is allowed to fill at various times;
- a set of *active roles*, which it currently occupies.

Roles have an associated set of *transactions*, which are the activities that someone in that role is permitted to carry out.

The set of transactions can be organization specific: open an account, cash a check, transfer funds, etc.

Primary Rules

The following are the three primary RBAC rules:

- **Role assignment:** A subject can execute a transaction only if the subject has an active role.
- **Role authorization:** A subject's active role must be an authorized role for that subject.
- **Transaction authorization:** A subject can execute a transaction only if the transaction is authorized for one of the subject's active roles.

Note that a subject can have multiple roles. For example, in a pinch a bank president might also act as a teller.

Subsumption and Separation of Function

One role may *subsume* another, meaning that anyone having role r_j can do at least the functions of r_i.

Example: a `trainer` can perform all of the actions of a `trainee`, as well as some others.

RBAC can also model *separation of function* (one individual cannot assume both roles r_1 and r_2).

Example: if `teller` is among S's authorized roles, `auditor` cannot be.

On the video, I mistakenly called this separation of duty.

RBAC Advantages

RBAC is generally more flexible than standard access control policies:

- RBAC is easy to administer. Everyone in role `teller` has the same permissions.
- Permissions are appropriate to the organization—"open an account" rather than "read a file."
- RBAC recognizes that a subject often has various functions within the organization.
- RBAC allows a subject to transition between roles without having to change identities.

Lessons

- RBAC associates access permissions with a job/function/role rather than with individual subjects.
- This provides a flexible approach to modeling the dynamism of commercial organizations.

Next lecture: Storing the ACM