(4) Role Engineering

Edward J. Coyne

SETA Corporation
ecoyne@seta.com

1.0 Discussion

Implicit in role-based access control (RBAC) is the existence of a set of roles that accurately reflect the activities, functions, and responsibilities within the organization. Definition of the roles with their assigned permissions must be accomplished before all the benefits of RBAC can be realized. The definition of roles is essentially a requirements engineering process. The goal is to define a set of roles that is complete, correct, and efficient. Therefore, a methodology for establishing a valid set of roles with assigned permissions is needed. Software tools to assist in carrying out the methodology would be useful as well.

The concept of role engineering (RE) is an approach to defining roles and assigning permissions to the roles. RE must capture the organization's business rules, as these relate to access control, and reflect these rules in defining, naming, structuring, and constraining a valid set of roles. In particular, RE should seek to design all components of the RBAC3 model except for assignment of users to roles. The RBAC3 model is comprised of three sub-models that describe features of RBAC. The RBAC3 model is described in [SAND96a] and includes the following:

- Basic RBAC (RBAC0), where permissions are assigned to roles and users are assigned to roles,
- Role-hierarchy RBAC (RBAC1), where hierarchies of roles with inheritance may be defined, and
- Constrained-role RBAC (RBAC2), where constraints among roles and assignments may be defined.

That is, RBAC3 permits both role hierarchies and constraints on roles.

The components of the RBAC3 model to be defined as part of RE are the following:

- Roles
- Permissions
- Constraints
- Hierarchies

The ability to identify roles appropriate to the organization should be useful to organizations operating or using large and complex information systems. RE will make it possible to implement RBAC with an effective scheme to control access and to simplify the management of that access.

ACM RBAC Workshop, MD, USA
© 1996 ACM 0-89791-759-6/95/0011 $3.50

An added benefit will be that RE dovetails with other requirements engineering efforts to identify user roles as a basis for designing system functions and user interfaces.

Identification of roles can be performed following these steps:

- Collect activities performed by system users. These activities will be stated as verbs/object pairs (e.g., "authorize payment," "issue payment," and "search master file").
- Collect activities into clusters to be performed by a single individual.
- Name each activity cluster using a noun that describes the role being carried out via the cluster to define a candidate role.
- Write a one- or two-sentence description of each candidate role.
- Compare the candidate roles with one another and eliminate duplicate candidate roles.
- For each candidate role, identify the minimal set of permissions required to perform the role.
- Simulate the user's activities using the candidate roles and their assigned permissions.

At this point, a set of basic (candidate) roles has been defined. The next step would be to identify constraints among these roles. To do this efficiently, it is advisable to articulate the organization's security policy. If the policy includes separation of duties, two-person rules, or other operational constraints, these should be identified. Elements of the policy such as these will translate to constraints among roles, e.g., mutually exclusive roles or collaborative roles.

Given the candidate roles and their role-to-role constraints, it is appropriate to identify any hierarchies among related roles that are relevant to the organization. The hierarchies may follow organization lines or such other relationships as level of expertise or specialization. Once the hierarchical relationships have been established, it is necessary to partition the permission set according to the role at each level of the hierarchy.

It may be desirable to reflect organizational structure in the definition of roles. It is possible to do this by preparing an accurate organization chart and attaching roles to the nodes on the chart. Those roles that are unique to an organizational unit may be tagged with the organization name to indicate the line of authority associated with the role.

References

[SAND96a] Ravi S. Sandhu, Edward J. Coyne, Hal L. Feinstein, and Charles E. Youman, "Role-Based Access Control," *IEEE Computer*, 29:2, February 1996, 38-47.