Role Engineering:

The Cornerstone of RBAC

By Srinivasan Vanamali, CISA, CISSP

Role-based access control (RBAC) is becoming the norm for managing entitlements within commercial systems and applications. RBAC plays a significant role in establishing a model for enforcing security within organizations. RBAC is also one of the critical components of an identity management (IdM) framework.[1] It essentially simplifies entitlement management by using roles (as opposed to users) as authorization subjects.

RBAC should not be treated as the panacea for all ills related to access control, but it has proven to be cost-effective[2] for organizations—reducing entitlement management costs and complexity. It also reduces the risk of users having inappropriate access privileges and aggregating entitlements as they change job functions within the organization. As the users change their job function, they are assigned new roles and old roles are removed from their profile. This results in users' entitlements matching their job functions.

Evolution of Entitlement Management

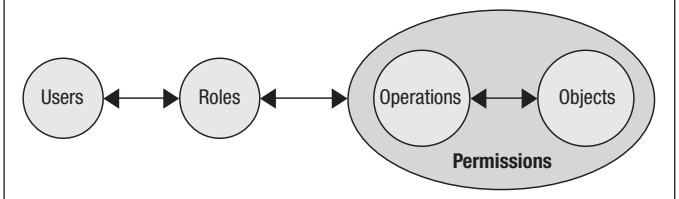
Traditionally, legacy systems and applications managed permissions by groups. Under this model, permissions are assigned to groups—users inherit permissions by being a member of a group. The ability to assign permissions to a group and determine who can inherit the permission is considered discretionary, as these determinations are made by the application and system owners. However, the authority to assign members to a group is deemed nondiscretionary and usually is performed by the security organization. This construct has evolved in recent times with the adaptation of RBAC in IdM solutions. Assigning permission to a role and determining membership of roles are supposed to be nondiscretionary. Users inherit sets of entitlements as their "birthright," as they are enrolled into the organization as part of the on-boarding process.

Conventionally, managing entitlements has been considered technical, as entitlements are related to applications and are managed in silos without much business input. With the emergence of various regulatory requirements, such as the US Sarbanes-Oxley Act, US Health Insurance Portability and Accountability Act (HIPAA), Gramm-Leach-Bliley Act (GLBA) and EU Privacy Protection Directive 2002/58/EC, it is increasingly important to streamline the entitlement management process with business oversight, as it has become a security governance and compliance issue.

RBAC 101

The fundamental concept of RBAC is that roles aggregate privileges. Users are assigned roles and inherit predefined permissions by being members of roles. They are given entitlements—no more than what is required to perform their job function—based on the "least privilege access" principle. **Figure 1** depicts the key building blocks of the core RBAC reference model per the American National Standard for Information (ANSI)/InterNational Committee for Information Technology Standards (INCITS) 359-2004 standard.

Figure 1—Basic RBAC



The key elements of RBAC (see **figure 1**) are:

- **Users**—By definition, users are individuals who perform a job function within an organization. Users traditionally have been designed to perform individual functions within an organization.
- **Roles**—In a business context, roles represent job functions and related responsibilities. Responsibilities represent users' implicit or explicit authority to execute their job function. In a technology context, roles represent a collection of entitlements that a person inherits from an application perspective to perform a job function.
- **Permissions**—In a technology context, permission is the provision of authority to someone to perform an operation against an RBAC-controlled object within an application or system.

Role Engineering

As organizations start deploying IdM solutions, it is becoming increasingly important to devise a common set of roles that can be reused over and over again, as opposed to defining roles every time an IdM component is deployed. One of the challenges often faced is that, if defined incorrectly, roles are ineffective and fail to meet the organization's requirements.[3]

Roles can be defined at an abstract level from a business perspective or can be context-specific to an application or system from a technology perspective. At an abstract level, a role can be a simple label that defines the job function with a set of responsibilities and the authority that goes with it. For example, a bank teller's job function can be a role defined as

"teller," with the responsibility to perform financial transactions with certain limits (authority). At an abstract level, there is no enforcement capability. The role "teller" in an application has specific entitlements that enable a user to execute transactions with certain limits. How this is configured within the application and how it is enforced are specific to the individual application's capability.

Whether an organization looks at defining roles as either abstract or specific to a context, the requirement to define roles is important and role definition is a critical step in deploying any RBAC systems.

Role engineering is the process of defining roles and related information, such as permissions, constraints and role hierarchies, as they pertain to the users' functional use of systems, applications and business processes. It is essentially one of the critical steps in deploying RBAC-oriented IdM systems. Organizations often implement IdM systems based on a role-based paradigm without much consideration for roles.[4] To minimize the deployment effort or to avoid project scope creep, since role engineering often is not considered part of the initial scope, organizations frequently do not invest enough time to define roles; rather, they tend to define high-level roles that do not reflect the organizational job functions. Permissions mapped to these high-level roles are usually generic in nature. The result of this haphazard process is that additional efforts are required to manage job-function-specific permissions manually, outside the IdM system capability. This often results in IdM systems being ineffective and not delivering the expected business value, for example, adherence to compliance and reduced entitlement management costs. The process of defining roles should be based on a complete analysis of how an organization functions and should include input from a wide spectrum of users, including business line managers and human resources.

Role definition and management require alignment between business and IT. They require a strong commitment and cooperation among the business units, as a role-engineering initiative could transcend across the enterprise.

Role Engineering Approaches

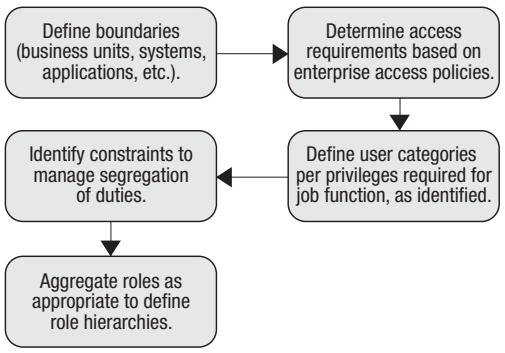
Role engineering approaches include:

- **Top-down**—This approach is primarily business-driven, and roles are defined based on the responsibilities of a given job function. For roles to be effective, there should be a strong alignment between business and IT objectives. Roles are defined by reviewing organizational business and job functions and mapping the permissions for each job function. This approach provides business oversight and alignment of roles with business functions and reusability.

  Figure 2 provides the key steps for a top-down role engineering approach:
  - For a successful role engineering project, it is pivotal to define the scope and boundaries for the project. If the organization has a large user population, it would be ideal to conduct a pilot to validate the approach and the outcome. The boundaries could be specific business units or applications that are being considered for role definition.
  - It is important to identify enterprise access policies to determine entitlements for a given job function. The objective of this exercise is to define entitlements based on the "least privilege access" principle.
  - The next step is to group users in a given business unit based on privileges corresponding to their job function. This would provide some basis for determining appropriate criteria to identify and define the user population.
  - One of the critical aspects of role definition is to avoid having mutually exclusive roles assigned to the same person. For example, a person who creates a purchase order should not be the one who approves it. This type of constraint is defined as part of segregation-of-duties policies. It is important to capture the constraints, so that rules can be established to evaluate what types of roles can be assigned to a user for a given job function.
  - Role hierarchies help simplify role definitions by aggregating roles. Role hierarchies usually follow the pattern of organizational hierarchies, where users in the higher organizational structure are able to perform the job functions of their direct and indirect reports. For example, a bank branch manager can perform the job function of a bank teller. Creating role hierarchies simplifies the number of roles assigned to a user.
- **Bottom-up**—This approach is based on performing role mining/discovery by exploring existing user permissions in current applications and systems. Once user permissions are explored, the next step is to perform role normalization and rationalization. Under this approach, roles are defined to meet specific application or system access requirements.

  One of the challenges of this approach is that it requires viable commercial tools to perform role mining. An alternate approach is to select a set of representative users and extract the entitlements that best describe the job function. If the user population is significant, it would be ideal to sample a certain percentage of the population to validate the accuracy of the outcome.

Figure 2—Top-down Role Engineering



One of the outcomes of this approach is that users often accumulate entitlements based on their previous job functions and it could become too daunting to extract the entitlements without the business' involvement. This is a key aspect of role rationalization to be considered as part of the bottom-up approach.

- **Hybrid**—This approach combines the previous two approaches. It leverages normalized roles derived from role mining and aligns them to job functions, with the involvement of the business.

Conclusion

As organizations embark on various RBAC-oriented IdM initiatives, it is becoming evident that defining high-level roles with basic entitlements does not deliver expected business benefits. It is imperative for a successful role definition to require management support, sufficient funding for the role engineering effort, business unit participation and resources committed to the project. The importance of roles should not become an afterthought, but should be considered as an integral part of any IdM initiative. Organizations also need to address requirements for roles from a compliance standpoint. Entitlement certification is becoming a critical aspect of various regulatory compliance initiatives. Having a holistic approach to role definition helps alleviate certification-related regulatory compliance challenges.

It is important for organizations to get the expected business benefits with careful consideration for how roles are going to be defined and managed on an ongoing basis. Defining roles is difficult under any circumstances, but the process could be overbearing without established limits. It is important to define boundaries for the user population, applications and platforms, and the number of business units to be covered by the project.

Role engineering, in a top-down or bottom-up approach, is a key cornerstone in the process of defining roles that meet the organizational requirements. Once the roles are defined and inventory has been published, it has to be maintained by both the business and IT, as this helps to keep the information current and available for any future IdM initiatives.

Endnotes

[1] Vanamali, Srinivasan; "Identity Management Framework: Delivering Value for Business," *Information Systems Control Journal*, vol. 4, 2004

[2] National Institute of Standards and Technology, "The Economic Impact of RBAC," USA, *http://csrc.nist.gov/rbac/*

[3] Kampman, Kevin; "The Business of Roles," *Methodologies and Best Practices*, vol. 1, 1 February 2006, Burton Group, *www.burtongroup.com*

[4] *Ibid*.

Srinivasan Vanamali, CISA, CISSP
is a global solution manager of the global security practice at CA Inc. He has more than 18 years of experience in a variety of IT management and technical roles. His expertise includes key aspects of security, including identity and access management, industry standards, deployment methodologies, compliance, and technology vision. He can be reached at *srinivasan.vanamali@ca.com*.

Information Systems Control Journal is published by ISACA. Membership in the association, a voluntary organization serving IT governance professionals, entitles one to receive an annual subscription to the *Information Systems Control Journal*.