

3

with the currently defined role, and responsibility, if applicable, assigned to a user, is made available at run-time to the respective external application requesting access authorization for a user who seeks access to said protected resource via said application. In the preferred system, the central repository runs on and is maintained by the same system that

Advantages

The data model, which is object of the invention, proposes a novel organization of the privileges held by system users, thus simplifying the maintenance of authorizations. Under the premise that the privileges of a user are bundled in roles, the purpose of the data model is to ease the role management by dramatically reducing the number of roles necessary in a system. The present invention will enable user management to evolve from proprietary, application-specific solutions that contain only high-level role information to central generic authorization repositories that offer not only role data, but also detailed, ready-to-use authorization information. Applications will be able to use information supplied by the central user repositories without additional processing or checking, eliminating the need for sophisticated user management functions within individual business applications, thus relieving the administrator from maintaining the access rights with every participating system. The needed authorization information is managed, maintained by and kept at the appropriate system that hosts the user base, e.g., human resources.

The details of one or more embodiments of the invention are set forth in the accompanying drawings and the description below. Other features, objects, and advantages of the invention will be apparent from the description and drawings, and from the claims.

DESCRIPTION OF DRAWINGS

FIG. 1 is a flow chart illustrating the basic structure for authorization management, according to the typical usage.

FIG. 2 is a data model for authorization management data used with the authorization management system of FIG. 1.

FIG. 3 is a flow chart illustrating a process for completion of authorization at runtime.

FIG. 4 is a wild-card based authorization data model.

FIG. 5 is the flow chart of FIG. 1 shaded to show the portion that is replaced by the wild card based model of the invention.

FIG. 6 is the data model of FIG. 2 associated with the flow chart of FIG. 5.

FIG. 7 is a flow chart illustrating in more detail the process for completion of authorization at runtime.

FIG. 8 is the wild-card based authorization data model of FIG. 4 associated with the flow chart of FIG. 7.

FIG. 9 is a block diagram of a dynamic authorization system using a centralized authorization repository.

Like reference symbols in the various drawings indicate like elements.

DETAILED DESCRIPTION

Introduction

Maintaining separate roles for each distinct position within the organization results in a proliferation of specific

4

roles, which often differ only by virtue of the individual's location geographically or within the organization chart. For example, the positions of sales manager for Brazil and sales manager for Portugal would typically differ critically not in function but in the geographical territory of responsibility, cost center, reporting lines, etc.

The data model of the present invention relies on the separation between roles in the functional sense and responsibilities. In other words, the actions that the user is allowed to make, in terms of access to services or processing steps, are distinguished in the data model from the resources, usually, data, on which these actions are performed. As a consequence, different responsibilities of users having the same function do not require different roles to be maintained in the system. Conversely, the area of responsibility may be the same but different users within the same area can have different functional roles. In the end, the privileges of a given user are defined by the combination of both parts, the functional roles and the areas of responsibilities.

The data model which is object of the invention takes advantage of this new way of looking at roles by taking into account the divisibility and fungibility of roles and responsibilities and using this logic to create a novel organization of the privileges held by system users, thus simplifying the maintenance of authorizations. Under the premise that the privileges of a user are bundled in roles, the purpose of the data model is to ease the role management by dramatically reducing the number of roles necessary in a system.

Instead of treating roles as a flat enumeration, the invention is based upon a quadratic model, i.e., one in which each position within the enterprise is defined as the product of two types of variables: (1) a generic role (which may be comprised of a unique set of sub roles) and (2) a specific set of responsibilities to go with the roles.

Conventional roles, i.e., "specific roles," more in the nature of position descriptions, can be separated from their associated responsibilities and thus factored into generic roles and subroles on the one hand and responsibilities and sub-responsibilities on the other hand. Generic roles can be redefined to represent exclusively the functions of the individual's position reflected in the actions he or she is allowed to perform without consideration of the objects upon which those actions are executed. The functions at the atomic level can be aggregated to form generic roles associated commonly with similar positions among the company's staff. So all the functions associated with a basic sales position would be assigned to a generic sales role. Separately, the responsibilities consist of the objects upon which the actions are executed, e.g., data that an action modifies. For sales roles, the corresponding set of responsibility parameters would include, e.g., responsibility for a particular sales territory, or range of products, or customers within a certain size range, business or other category, etc. Taken alone, these generic roles and responsibilities are an incomplete description. But together the intersection of a generic role and a specific set of responsibilities fully specifies the individual's position, or conventionally, his specific role in the organization.

This approach imparts object-oriented programming principles to role administration in the sense that the generic roles or responsibilities are treated as objects that have predefined characteristics and inheritance properties. Thus when new positions are added in the company the same modular objects (i.e., generic roles) can be re-used in most cases without generating new roles.