Role Based Access Control

M. Eswararao
MTECH, Department of Computer science Engineering, Sreyas Institute of Engineering and Technology, Hyderabad, Telangana, India.

Dr. V. Goutham
HOD/Professor, Department of Computer science Engineering, Sreyas Institute of Engineering and Technology, Hyderabad, Telangana, India.

Abstract

The essence of Role-Based Access Control (RBAC) is that system permissions are assigned to defined “roles” rather than to individual users. User’s ac quire these permissions by virtue of being authorized to act in a categorized manner known as a “role”. The driving motivation for RBAC is to simplify security policy administration while facilitating the definition of flexible, customized policies. Basic RBAC models have been successfully applied since the mainframe era, but emerging networked systems, which have greater numbers of users, roles, and program components, challenge the expressive power of these classical RBAC models. This is particularly true for cross-enterprise distributed networks for electronic commerce applications. The development of new modeling concepts and techniques is required to support large-scale, enterprise-wide, distributed systems.

Introduction

A study by the United States Government’s National Institute for Standards and Technology (found that Role Based Access Control (RBAC) addresses many of the systems security needs of both the commercial and government sectors. Many organizations base access control decisions on the “roles” that individual users take on as part of an organization. The preferred information system of use for RBAC would exhibit the following characteristics:

User characteristics: large number of users, few security administrators, frequent change ofjob responsibility;

Data and application characteristics: large number of data objects, sharing based on jobfunctions;

Enterprise characteristics: data owned by enterprise, controlled by security administrators,before and after the fact audit, and periodic assessment of access control policy enforcement necessary. One of RBAC’s major points is its administrative capabilities. Systems administrators control access at a level of abstraction that reflects the way enterprises typically conduct their business. This is achieved by statically and dynamically regulating users’ actions through the establishment and definition of roles, role hierarchies, relationships and constraints. This is in contrast to the conventional and less intuitive process of administering lower level access control mechanisms directly on an object-by-object basis.

RBAC is pervasive in computer systems enveloping the database, network, distributed systems, operating system, security and application communities. Concepts of RBAC have evolved more or less independently in these communities.

What is RBAC?

In Role-Based Access Control (RBAC),rightsand permissions are assigned to roles rather than to individual users. Users acquire these rights and permissions as they are assigned membership in appropriate roles. This simple idea greatly eases the administration of authorizations. The basic concepts behind RBAC have been around since the advent of multi-user computing and information systems in the late 60's and early 70's. Many such systems evolved from research and development efforts of the US Department of Defense.

These two basic types of access control mechanisms are used to protect information from unauthorized access:

DAC Because DAC places the decision of who can access information at the*discretion*ofthe creator on the information; DAC is not applicable to the majority of personal information systems.

MAC Because MAC requires all those who create, access, and maintain information tofollow rules set by administrators, MAC is the kind of access control mechanism required of personal information systems. The most commonly used MAC is the multilevel security (MLS) mechanism used by the US DoD. This mechanism associates information with such labels as *TOP SECRET, SECRET, and CONFIDENTIAL.* This type of MAC is not flexible enough forcommercial use, nor is it adequate for the needs of personal information systems.

RBAC then is a MAC, which has evolved to meet the needs of commercial information systems. Rather than labeling information, it associates *roles* with each individual who might have a need to access information. Each role defines a specific set of operations that the individual acting in that role may perform. The operations may be broad or very specific, e.g., when a diagnosis is entered into a patient record, the symptoms leading to that diagnosis must also be entered. Indeed the data "semantics" must be implemented by any RBAC scheme

Why is RBAC better than other Access Control policies?

The major benefits of RBAC are the ability to express and enforce enterprise-specific security policies and to simplify the process of security management. RBAC is a framework of policy rich mechanisms that allow per-subject (role) as well as per-object access review. Its configuration is dependent on organizational policies. This allows RBAC to be adaptable (more so than other types of access control) to any organizational structure and means of conducting business. The policies implemented under RBAC can evolve over time as enterprise and organizational structure and security needs change. RBAC has been seen as a "commercial" and cost-effective alternative to the earlier "MAC" concepts as outlined in the "Rainbow Series".

RBAC provides greater productivity on the part of security administrators, resulting in fewer errors and a greater degree of operational security. It can also be argued that RBAC/MAC actually simplifies information system management, administration and security. This is achieved by statically and dynamically controlling the actions of users by establishing and defining roles, role hierarchies, relationships and constraints. After this RBAC environment is established, the main administrative tasks tend to be adding and deleting users to and from roles.

RBAC Characteristics and Policies

RBAC policies are described in terms of users, subjects, roles, role hierarchies, operations and protected objects. To perform an operation on a RBAC controlled object, a user must be active in some role. This assumes that the user is an authorized member of that role. RBAC enables administrators to place constraints on role authorization, role activation and operation execution. Constraints could include cardinality and mutual exclusivity rules that can be applied on a role-by-role basis. Constraints can also be placed on the authorization of an operation to a role and on operations being performed on objects, e.g. time and location constraints.

The following itemized list outlines many of the characteristics of RBAC:

- within the RBAC framework, a **user** is a person, a **role** is a collection of job functions, and an **operation** represents a particular mode of access to a set of one or more protected RBAC objects;

- the type of operations and the objects that RBAC controls is dependent on the type of system in which it will be implemented, e.g. within a transaction management system, operations would take the form of and exhibit all the properties of a transaction;

- Roles can have overlapping responsibilities and privileges, i.e. users belonging to different roles may need to perform common operations. RBAC therefore supports the concept of role hierarchies;

- A**role hierarchy** defines roles that have unique attributes and that may contain other roles, i.e. that one role may implicitly include the operations, constraints, and objects that are associated with another role;

- **role authorization**(association of user with a role) can be subject to the following:
  The user can be given no more privilege than is necessary to perform his/her job **(principle of least privilege)**. the role in which the user is gaining membership is not mutually exclusive with another role for which the user already possesses membership **(static separation of duty);**the numerical limitation that exists for role membership cannot be exceeded (**cardinality property**);

- **Role activation** involves the mapping of a user to one or possibly many roles. A userinitiates a session during which the user is associated with a subset of roles for which that user has membership. A particular role for a user can be activated if:
  The user is authorized for the role being proposed for activation;
  The activation of the proposed role is not mutually exclusive with any other active role(s) of the user;
  The proposed operation is authorized for the role that is being proposed for activation.
  &
  The operation being proposed is consistent within a mandatory sequence of operations.

What is the difference between roles and groups?

A major difference between most implementations of groups and the concept of roles is that groups are treated as a collection of users and not as a collection of permissions and users are seen as being people. A role is both a collection of users and a collection of permissions. The role serves as an intermediary to bring these two collections together.

A well-known operating system that uses the group concept is Unix. In Unix group membership is defined in the files/etc. /password and /etc./group, from which it is easy to determine the groups a user belongs to or which users belong to a particular group. On the other hand determining the permissions of a group is not so straightforward. To do this will generally require a traversal of the file system as permissions are granted to groups on the basis of permission bits associated with individual files and directories. Additionally, the assignment of permissions to groups is highly decentralized (the owner of any Unixfile system sub-tree can assign permissions for that sub-tree to a group).

Do any standards incorporate RBAC?

A number of organizations are including provisions for RBAC in open consensus specifications and in the standards area.

RBAC is an integral part of the security model and architecture for the Secure European System for Applications in a Multi-vendor Environment (**SESAME**) security scheme for distributed computer network systems. In addition, the Object Management Group's (OMG) Common Object Request Broker Architecture (**CORBA**) security specification uses RBAC as an example of an access control mechanism that can be used with the Distributed Object Technology defined by the OMG. At the OMG Security SIG meeting in 1995, John Barkley from the US National Institute of Standards and Technology (NIST), gave a presentation on RBAC, and how it might be applied in a CORBA based environment.

Role Based Access Control Models and Architectures

Role based access control has been proclaimed as being capable of representing many kinds of security policies, models and architectures. Basic RBAC models have been successfully applied since the mainframe era, but emerging systems, which have greater number of users, roles, and program components, are challenging the expressive power of these traditional models. Current research is focusing on the evaluation of the effectiveness of RBAC models and the development of newer, better modeling concepts and techniques.

As stated previously, RBAC does not enforce any one protection policy. Coinciding with the First ACM Workshop on Role-Based Access Control in December 1995,and since then, a number of researchers have formulated or described models and architectures for RBAC. Below is a selection of these.

A new model suitable for both conceptual and logical database security design is described in the model attempts to address the semantic mismatch between the currently accepted notion of role-based access control and the real-world significance of the role concept. Additionally, a role algebra has been defined which constitutes the basis for the development of automated tools for database security design. Interestingly, to build a complete database design methodology the author suggests that the new model may be used together with a conceptual database model, such as the Entity-Relationship data model.

In this security architecture, the concept of a role is used to define access control related to a position within an organization although the role framework described caters for the specification of both authorization and obligation policies. In this framework roles and role relationships represent enterprise security requirements. Roles are expressed in terms of policies that refer to domains to provide a very flexible basis for specifying RBAC that caters for large scale, inter-organizational distributed systems. The advantage of using roles for specifying enterprise policies is that individuals can be assigned or withdrawn from the role positions without having to specify the policies applying to the role. An object oriented approach to specifying roles permits multiple instances of a basic role to be instantiated, e.g. all branch managers of a bank.

The authorization or access control policies used in this architecture provide a very flexible means of specifying access control permissions and including constraints (such as a time validity policy), to limit their applicability. They can also be extended to specify delegation policy, including cascaded delegation of access rights. Policies explicitly identify both subjects and targets, and since domains maintain information about the policies applying to them, it is easy to analyses the policies to determine those applying to a specific object.

Access control and authentication are implemented using security agents on a per host basis to achieve a high degree of transparency to the application level. The domain based authentication service uses symmetric cryptography and is implemented by replicated servers that maintain minimal state. More specifically, the security architecture consists of four main components:

Sample Uses of Role Based Access Control

Health Care Information Security

Within the health care industry, there are continual problems associated with how to ensure the secrecy and integrity of health information, in particular, patient information. In Australia, health care information security and privacy is addressed by the Standards Australia publication AS4400. It is generally accepted that RBAC is more suited to health care applications than other access control schemes due to the variety of people interested in the patients' medical condition. A doctor (role) would require access to all patient information, whereas a pathologist (role) would only require access to some patient information, age, sex, and clinical data. At the First ACM/NIST RBAC Workshop in 1995 it was noted that role is a fluid concept in hospitals. This strongly infers that dynamic discretionary roles are important. In other words, a mechanism is required for a user to change roles. This is an important area of RBAC for future applied research.

Security Administration Manager (SAM)

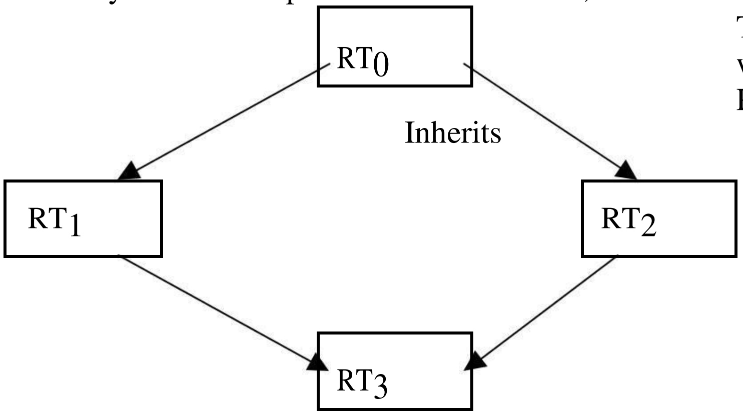
The Security Administrator Manager (SAM) of Schumann Security Software, Inc. is a tool for enterprise-wide security management that implements many RBAC concepts. It has been commercially available since 1993.

With SAM, all security systems and all application security of an enterprise can be administered from a single point of administration/control. SAM provides an architecture under which company-wide information security administration can be automated by interfacing to human resource applications, authentication servers, single sign-on services and access control systems. The SAM environment provides all features required to maintain users, their authorizations and privileges, and the authorized resources, across multiple platforms.

Implementing Role Based Access Control

Is RBAC object-oriented?

In a Role Type Inheritance Property is depicted as follows.



This seems to infer that RBAC is well suited to the object-oriented Paradigm.

In fact, under the direction of John Barkley at NIST, a distributed, object-oriented, RBAC demonstration system has been developed This Demonstration uses CORBA, ORACLE's SQL/RDA (remote data access) and RBAC. In this demonstration, a patient record data base object (PRDBO) is defined. CORBA is used as a means of implementing the PRDBO. The methods in the object implementation access the data however and wherever the data is actually stored. SQL/RDA is used within the PRDBO methods to access the data.

The PRDBO organizes patient information into groups. Access to patient information is controlled using the RBAC mechanism. To successfully log into the system, a correct combination of username/role must be supplied. After successfully logging in, screens are

Presented which are associated with the role chosen at login. These presented screens are related to the level of access that is associated with the role.

Implementing RBAC using Object Technology

The abovementioned project uses a concept of layered objects to facilitate flexible administration while minimizing impact of role changes on applications. More specifically, with the Role Based Access Control (RBAC) implemented in this project, each role is associated with a set of operations which a user in that role may perform. The power of RBAC as an improved access control mechanism is centered on the concept that an operation may theoretically be anything an applications developer desires. This is contrasted to traditional access control mechanisms where labels are associated with information blocks. These labels indicate relatively simple operations, such as, read or write, which can be performed on an information block. In direct contrast, operations in RBAC may be arbitrarily complex.

Role Based Access Control Research

The Fourth ACM Workshop on Role-Based Access Control was scheduled for October 28-29, 1999. The theme for this workshop will be the discussion of the RBAC applications (Web, CORBA), various implementations of RBAC and products and tools based on the RBAC model.

The previous three workshops have examined the theoretical foundations of RBAC and application of RBAC models to a variety of systems. The inaugural workshop, held in November 1995, was successful in its modest goal of taking a first step towards a consensus reference model for RBAC. The second workshop, held in November 1997, saw applications of RBAC models in operating systems, databases, distributed applications, and the administration of RBAC policies themselves. In addition, several new modeling concepts were introduced in the second workshop that may enable complex systems to be built and administered more easily. The third workshop, held in October 1998 focused discussion on the application of RBAC models and concepts to both traditional and emerging systems. In particular, researchers looked at the evaluation of the effectiveness of current RBAC models and the development of new modeling concepts to address the needs of future applications based on the RBAC paradigm. Proceedings of the first three workshops are available from ACM.

Probably the ideal way to portray what current research is being done in Role Based Access Control is to present an overview of the technical program for the Fourth ACM Workshop on RBAC: (1999)

- **Session 1: Applications**
  RBAC on the WEB by Smart Certificates
  Role-Based Access Control on the Web Using JAVA
  A Framework for Implementing Role-Based Access Control using CORBA Security Service

- **Session 2: Constraints**
  On the Increasing Importance of Constraints
  The RSL99 Language for Role-Based Separation of Duty constraints
  Supporting Relationships in Access Control Using Role Based Access Control

- **Session 3: Implementations**
  Migrating to Role-Based Access Control
  Secure Flow: A Secure Web-enabled Workflow Management System RBAC in UNIX Administration

- **Session 4: Models and Model Extensions**

  Managing Trust between Collaborating Companies Using Outsourced Role Based Access Control

  Dynamic Rights: Safe Extensible Access Control

  Attribute Certification: An Enabling Technology for Delegation and Role-Based Controls in Distributed Environments

- **Session 6: Tools**

  Towards a UML Based Approach to Role Engineering Napoleon Network Application Policy Environment

  The Uses of Role Hierarchies in Access Control

Conclusions

Role Based Access Control is increasingly being incorporated into the security features found in many operating systems and database management systems With RBAC, permissions are assigned to roles rather than individual users. There are a number of advantages to using RBAC. Firstly, roles are an enterprise or an organizational concept. This allows the modeling of security from an enterprise perspective as security modeling can be aligned to the roles and responsibilities in an enterprise. Secondly, RBAC is more saleable than user-based security mechanisms since security can be administered as a whole for all users belonging to a role.

Current research in RBAC is focusing on the evaluation of the effectiveness of current RBAC models and the development of new modeling concepts to address the needs of future applications based on the RBAC paradigm. In particular, there is the need for a sophisticated role language mechanism that can be utilized with a RBAC model to deal with the dynamic nature of roles and real-time changes in authorization.

References

1. Y.Y.Al-Salqan et al, “Security and Confidentiality in Health Care Informatics”, Proceedings of the First ACM Workshop on Role-Based Access Control, December 1995

2. Roland Awischus, “Role Based Access Control with the Security Administration Manager (SAM)”, Proceedings of the Secon d ACM Workshop on Role-Based Access Control, November 1997

3. John Barkley, "Application Engineering in Health Care", Computer Systems Laboratories NIST, May 1995

4. J.Barkley, D.Kuhn, L.Rosenthal, M.Skall, A.Cincotta, "Role-Based Access Control for the Web", CALS Expo International & 21st Century Commerce 1998: Global Business Solutions for the New Millennium.

5. John Barkley, "Implementing Role-Based Access Control using Object Technology", Proceedings of the First ACM Workshop on Role-Based Access Control, December 1995

6. John Barkley, "Comparing Simple Role Based Access Control Models and Access Control Lists", Proceedings of the Second ACM Workshop on Role-Based Access Control, November 1997

7. Hans H. Bruggemann, “Report of discussion sessions following presentations, RBAC and Next-Generation Security Models?”, Proceedings of the First ACM Workshop

on Role-Based Access Control, December 1995

8. E.Bertino, E.Ferrari, V.Atluri, "A Flexible Model Supporting the Specification and Enforcement of Role-based Authorizations in Workflow Management Systems", Proceedings of the Second ACM Workshop on Role-Based Access Control, November 1997
9. R.Thomas, B.Hartman, “RBAC and Distributed Object-Based Enterprise Computing”, December 1995