Designing of RBAC Model For Secure Multiple Domain Environments

Cungang Yang
Department of Electrical and Computer Engineering
Ryerson University
Toronto, Ontario, M5B 2K3
cungang@ee.ryerson.ca

Chang N. Zhang
Department of Computer Science
University of Regina, TRLabs
Regina, Saskatchewan, S4S 0A2
zhang@cs.uregina.ca

Bobby Ma
Department of Electrical and Computer Engineering
Ryerson University
Toronto, Ontario, M5B 2K3
bma@ee.ryerson.ca

Leon Pan
Department of Computer Science
University of Regina, TRLabs
Regina, Saskatchewan, S4S 0A2
panli111@cs.uregina.ca

Abstract:

In the secure domain computing environments, it is important to keep resources and information integrity from unauthorized access. Therefore, there is a strong demand on the access control for shared resources. In the past few years, Role-Based-Access-Control (RBAC) has been introduced and offered a powerful means of specifying access control decisions. In this paper, we propose an RBAC model for multiple enterprise domains that efficiently represents the real world. It supports multiple secure domains access control for E-business carried out by multiple enterprises over the Internet.

Keywords: Secure domain, RBAC, Least Privilege, Digital Credential.

1. Introduction

An important element of access control is *secure domain* which consists of a set of users and objects managed by a common security policy and defined by a single authority. With the increasing of shared resources, unauthorized access to information by illegal users also increases, it is necessary to secure data through user authentication and access control policies.

Nowadays, there are three basic access control techniques: mandatory access control (MAC), discretionary access control (DAC) and role-based access control (RBAC) [1][2][3]. MAC enforces access control on the basis of information security labels attached to users and objects. MAC can determine all kinds of access control between subjects and objects consistently. Security labels are granted to all subjects and objects by the system supervisor and the modification of the security labels only in accordance with the content of the object. As MAC policy is not flexible, it is not suitable to be used in commercial areas. In case of DAC, each object has an access control list. However, in a large information system there are millions of objects, and each of which is assigned to thousands of subjects. The access control lists will be enormous in size and their maintenance will be much difficult and costly.

Compared with DAC and MAC, the central notion of RBAC is that users do not directly get access to enterprise objects; instead, access privileges of the objects are associated with roles, each user is assigned to one or multiple members of appropriate roles. As a result, an organization can not only preserve access control policy appropriate to its characteristics consistently, but it can also maintain access control relationships between users and objects independently. Users can be assigned to members of roles as determined by their responsibilities and qualifications, on the other hand, they can be easily reassigned without modifying the underlying access structure. RBAC greatly simplifies the management of authorizations while providing an appropriate method for great flexibility in specifying and enforcing enterprise-specific protection policies and reducing the management costs.

In the last few years, the fundamentals of RBAC policies have been identified [1] and a number of RBAC models have been proposed to satisfy security requirements in different areas [2][3][4], but they are all concept models and have not been efficiently represented the real world.

In this paper, we propose a variation of RBAC model for multiple enterprise domains. In addition,

an implementation method for multiple secure domains is presented.

2. RBAC Model for Multiple Secure Domain Environment

A number of different viewpoints about RBAC have been discussed [5][6][7]. The abstract model defined in this section intends to capture the essential features of RBAC and fully realizes the original RBAC functions for a multiple secure domain environment. The class diagram of the model described by United Modeling Language (UML) [8] is shown in Fig. 1.

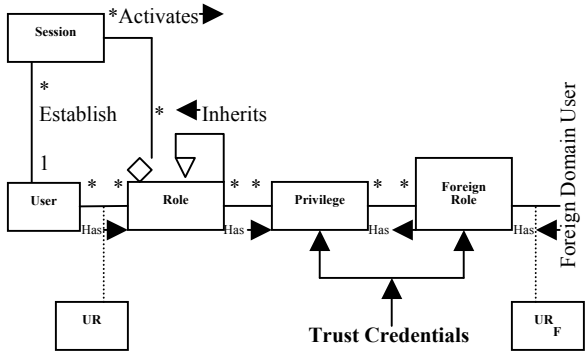

Fig.1. Class Diagram of the RBAC Model For Multiple Secure Domain Environment

In the model, a *user* is a human being or an autonomous agent, a *role* is a collection of privileges to perform a certain task, a *privilege* is an access mode that can be exercised on objects. A user can be assigned to a number of different roles, and a role can have multiple users. A role may have multiple privileges, and the same privilege can be associated to different roles. Moreover, a *role hierarchy* is introduced to reflect inheritance of authority and responsibility among the roles and it is defined by: If $r_i \leftarrow r_j$, then role r_i inherits the privileges of role r_j, role r_i is a *direct parent role* of role r_j and role r_j is a *direct child role* of role r_i. In the model, each user is assigned a role set according to his/her responsibility and authority, we call it *assigned role set*.

In the model, *session* is a mapping of a user to possible many roles [9], after the roles are authorized to a user, they are called *authorized roles* for the user. A user may have one or more than one sessions activated at the same time and each session may have a different combination of authorized roles.

Multiple S*ecure domains* are consist of a set of users and objects that are managed by different security policies and defined by multiple authorities. In the multiple secure environment, users may want to access objects located outside their administration domains. On the other hand, with the development of the Internet and e-business, there are many requirements that users in a secure domain would like to share objects with users in other secure domains. The large distributed system, like Internet, is quickly becoming the largest marketplace, allowing commerce and business between parties who are physically distant and do not know each other, such applications involve every aspect of e-business. In the model, we have extended the access control model from a single secure domain environment to multiple secure domain environments and we assume that all those different secure domains are based on the model and interconnected on the Internet. In each single secure domain, some privileges can be accessed by the users from other secure domains, we called them *foreign privileges* and those users from other secure domains are called *foreign domain users.* To establish the relationships between users in different domains, some trustworthy need to be established. The traditional approach of using authentication to differentiate between classes of clients is no longer sufficient, as knowledge of a user's identity will often not sufficient to determine whether a user is authorized a privilege, credentials make it feasible to manage trust establishment efficiently, our goal is therefore to explore the use of digital credentials [10][11][12] to solve this problem. Digital credentials are the online counterparts of paper credentials that people use in their daily life, such as a driver's license, ACM membership card, VISA card, etc. They are signed by authorized parties and can be made verifiable and unforgeable.

Assume we have three domains, domain1, 2, 3 and domain 1 provides medical information service for its secure domain users, also, it still provides the service for users from other domains such as domain 2 or domain 3 on the condition that the constraints of trust credentials should be satisfied. T*rust credentials* contains one or more than one digital credentials that should be provided by foreign domain users if they want to get some foreign privileges, such as the certificate of doctor or the certificate of nurse, for instance, if some services in domain 1 only can be accessed by the foreign domain user who is a certificated nurse, the certificate of nurse should be provided, if some service only can be accessed by the foreign domain user who is a certificated doctor, the certificate of doctor should be provided.

Based on trust credentials, we divided the privileges of a secure domain into three groups:

(1) Privileges that can be accessed by foreign domain users without the constraints of trust credentials

(2) Privileges that cannot be access by foreign domain users.
(3) Privileges that can be accessed by foreign domain users under different constraints of trust credentials.

Only privileges in group (1) and group (3) are foreign privileges, in order to simplify the administration of those foreign privileges, foreign role is introduced. *Foreign role* represents a group of foreign domain users who are authorized to get access to foreign privileges, foreign domain users do not directly get access to foreign privileges; instead, foreign privileges are associated with foreign roles, each foreign domain user is assigned to one or multiple members of appropriate foreign roles and each foreign role is assigned to one or multiple foreign domain users. Each foreign role can be assigned multiple foreign privileges and each foreign privilege can be assigned to multiple foreign roles. This idea greatly simplifies the management of authorizations while providing an appropriate method for great flexibility in specifying and enforcing multiple secure domain access control and reduces the management costs.

There are two kinds of trust credentials, *authentication credentials* and *authorization credentials*. Authentication credentials are the digital

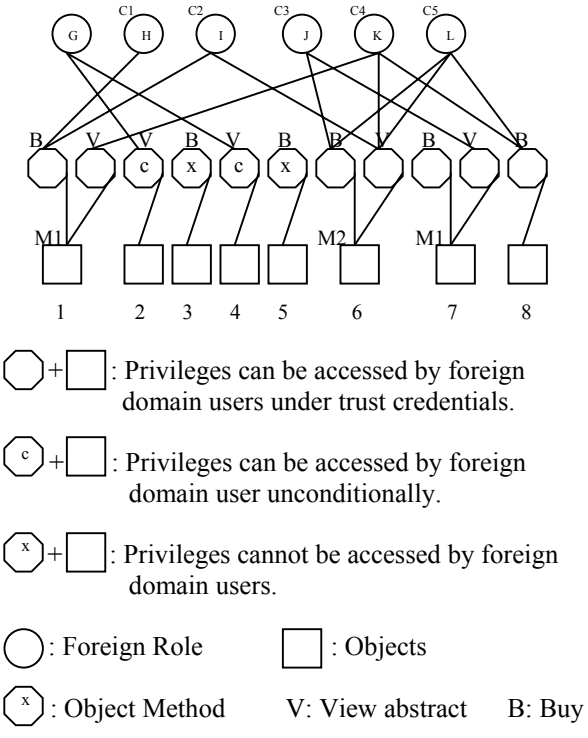

Fig 2. An example of ORBAC object diagram in multiple secure domains

credentials which constraint foreign domain users to get foreign roles. For example, if a user wants to view a web page in which only some authentication credential holder can access, it is sufficient to prove that he/she is actually an authentication credential holder, foreign domain users can be assigned to members of foreign roles as determined by authentication credentials, for instance, in Fig.2, foreign role H is authorized to the group of foreign domain users who owns the C1 authentication credential and foreign role I is authorized to the group of foreign domain users who owns the C2 authentication credential.

The other trust credential constraint is called *authorization credential*, it constraints whether or not the foreign domain users can get access to a foreign privilege after they have held foreign roles, the authorization credentials, for example, M1 and M2 in Fig.2 are Visa or Master card credential, so if a foreign domain user wants to buy the medical web pages 1 or web page 7, a M1 or M2 authorization credential should be submitted.

Following we take an example to briefly introduce the basic elements in the model for multiple secure domains which are shown in Fig. 2.

Example 1. Consider Fig. 2 where privileges of object 3 and privileges of object 5 can not be authorized to foreign domain users, privilege (Object 2, V) and (Object 4, V) are authorized to foreign role G without authentication or authorization credentials, any foreign domain user can get access to them unconditionally. Privilege (Object 1, B) is authorized to foreign role H under the authentication credential C1; Privilege (Object 1, B) is authorized to foreign role I under the authentication credential C2; privileges (Object 6, B) is authorized to foreign role J under the authentication credential C3 and authorization credential M2, privileges (Object 6, B) is authorized to foreign role L under the authentication credential C5 and the authorization credential M2; privilege (Object 7,V) is authorized to foreign role J under the authentication credential C3 but without authorization credentials, privilege (Object 8, B) is authorized to foreign role K under the authentication credential C4 and the authorization credential M1; privilege (Object 8, B) is authorized to foreign role L under the authentication credential C5 and the authorization credential M1.

UR_F determines the mapping from the authentication credentials to foreign role and return the foreign role to the foreign domain user. The main function of UR_F is accepting applied foreign privilege from foreign domain users, for each applied foreign privilege, searches all the foreign roles which is authorized the foreign privilege using breadth-first

search method, for instance, in Fig 2, the foreign roles of the foreign privilege {Object 7, V} is J and the authentication credential for privilege {Object 7, V} is C3, if the foreign authentication credentials a user provided contain C3, the user will be authorized the foreign role J to the user, otherwise, his/her application will be refused.

3. Conclusion and Future Work:

In this paper, we have presented a RBAC model for multiple enterprises. In large enterprise-wide systems the number of roles and privileges can be in the hundreds or thousands, and user can be in the tens or hundreds of thousands. Managing these roles, privileges, users, and their interrelationship is a formidable task. How to maintain the secure status of our role hierarchy, for every basic element of RBAC, such as roles, privileges, constraints, edges, how to add them, delete them or change them without violate any constraint of the role and without leading to illegal information flow is still a problem to be solved. Suitable algorithms should be presented to deal with them.

Reference:

[1] R. Sandhu, E. J. Coyne, H.L. Feinstein, and C.E. Youman. Role based Access Control Models. IEEE Computer, 29(2), Febuary, 1996, pp 38-47.

[2] Ravi sandhu and Venkata Bhamidipati, The ARBAC97 Model for Role-Based Administration of Roles: Preliminary Description and outline, *Second ACM workshop on Role-Based-Access-Control* , Fairfax, Virginia, USA, November, 1997, pp 41-54.

[3] Trent Jaeger, Frederquegiraud, A Role-Based Access Control Model for Protection domain Derivation and Management, *Second ACM Workshop on Role-Based-Access-Control*, Fairfax, Virginia, USA, November, 1997, pp 95-108.

[4] D.Ferraiolo, J. Cugini, and D.R. Kulin. Role based access control: Features and motivacation. *In annual Computer security applications conference. IEEE Computer Society Press, 1995.*

[5] Sylvia sborn, Yuxiao Guo, Modeling users in role-based access control, *Fifth ACM workshop on Role-Based Access Control, Berlin, Germany,* July 26-27, 2000, pp31-38.

[6] Ravi sandhu, David Ferraiodo and Richard Kulin, The NIST Model for Role-Based Access Control: Towards a unified Standard, *Fifth ACM Workshop on Role-Based Access Control,* Berlin, Germany, July, 2000, pp 47-64.

[7] Integrity in Automated Information Systems. National Computer Security Center, September, 1991, pp 79-91.

[8] UnitedModeling Language, UML resource center, Http://www.rational.com/UML/

[9] Sandhu, R.S, Coyne, E.J., Feinseein, H.L., and Younman C. E., *Proceedings of the first ACM Workshop on Role-Based-Access Control* , ACM, 1996.

[10] M.Blaze, J. Feignbaum, and J.Lacy. Decentralized trust management. IEEE Symposium on Security and Privacy, Oakland, CA, MAY 1996, pp 17-28.

[11] M.Blaze, J. Feignbaum, and A.D. Keromytis, "KeyNote: Trust management of Public-key Infrastructures,' Security Protocals, 6th International Workshop, Cambridge UK, 1998, pp 59-63.

[12] M.Blaze, J. Feignbaum, J, Ioannidis, and A. Keromytis, "The KeyNote Trust Management System Version 2," Internet Draft RFC 2704, September 1999.