

# Applying Fuzzy Relations in Role-Based Access Control

Amir Hedayaty  
hedayat@ce.sharif.edu

Mohsen Taherian  
taherian@ce.sharif.edu

Computer Engineering Department  
Sharif University of Technology  
Tehran, Iran

Computer Engineering Department  
Sharif University of Technology  
Tehran, Iran

## Abstract

Current computer security systems are based on the premise that once a user presents valid credentials to the authentication system (e.g. valid ID and password), they are granted access permission to all resources assigned to the user that they claim to be. However, numerous studies have shown that most security breaches are done by unauthorized users impersonating as authorized users (e.g. by cracking or stealing passwords) or by circumventing the authentication system altogether (by exploiting security “holes” in the system). Once the authentication system is broken, the system and the information kept in it become wide open to unauthorized access and malicious usage.. In this paper, we plan to first investigate the applicability of fuzzy approach to RBAC by identifying access control requirements that are inherently fuzzy in organizational contexts. Then we propose to develop a Fuzzy-RBAC (F-RBAC) model by extending an existing RBAC model with fuzzy parameters to allow imprecise access control policies.

## 1. Introduction

In recent years, with the widespread use of intranets and internets, users have become more and more dependent on the services provides by networked systems where computer programs and potentially sensitive information are kept in (geographically) dispersed systems and exchanged over telecommunication facilities. Distributed systems have emerged to provide the means through which networked systems cooperate to process users' tasks in a seamless and efficient fashion. Such systems provide tremendous benefits to their users but also raise new challenges. One of those challenges is to provide robust security mechanisms to guard against unauthorized access.

Security access control mechanisms play a key role in the overall structure of any security system. They are responsible for controlling the access permissions to system resources; i.e. determining who has access to which resource and with what type of access. Access control mechanisms rely on the authentication mechanisms to identify the users and ensuring that they are actually who they claim to be. The most common authentication method used to date is the user ID and password (or PIN number) combination, though other methods, such bio-metric identification, have been used with varying degrees of success [4].

The goal of access control is to allow only authorized users to access sensitive information. Role based access control (RBAC) is emerging as a generalized approach to security and has been shown to be applicable to a wide range of security requirements of organizations and applications. Possibility of using RBAC approach to an environment with multiple policy domains further justifies the tremendous momentum seen in RBAC research in the recent years. However, with the increasing size of the problem domain represented by today's huge information systems that cross organizational boundaries, the issue of management and complexity of security and providing its assurance poses daunting challenges. A fuzzy approach to addressing the issue of security can provide a pragmatic and promising new direction in this area.

RBAC approach is based on the premise that authorizing a particular user to access organizational information and to make modifications to it is generally based on the roles and responsibilities the user has within the organization. However, precise categorization of information in order to build an access policy, particularly in today's systems, becomes futile because of the enormity of the information space.

In many cases, human decision is needed for such categorizations as well as the formulation of security policies, bringing in the fuzziness that is natural in real world. For example, a document may be categorized somewhere between being *top secret* or *secret*. Role definitions vary from organization to organization. This increases the difficulty when systems with different policies and role structures have to interact. Furthermore, the specification of organizational security policies using such roles becomes imprecise highlighting the possibility that a fussy approach may be a more pragmatic way of dealing with such a problem.

The rest of the paper is organized as follows. Section 2 gives an overview of different types of Access Control. Section 3 is an introduction to using fuzzy relations for adding a security level pointed above. Following sections 4 and 5 will describe our methods for applying fuzzy to RBAC. Finally in section 6 we conclude our work and speak about future work.

## **2. Access Control Models**

A security policy may use two types of access controls, alone or in combination. In one, access control is left to the discretion of the owner. In the other, the operating system controls access, and the owner cannot override the controls.

The first type is based on user identity and is the most widely known: If an individual user can set an access control mechanism to allow or deny access to an object, that mechanism is a discretionary access control (DAC), also called an identity-based access control (IBAC).

Discretionary access controls base access rights on the identity of the subject and the identity of the object involved. Identity is the key; the owner of the object constrains who can access it by allowing only particular subjects to have access. The owner states the constraint in terms of the identity of the subject, or the owner of the subject.

The second type of access control is based on fiat, and identity is irrelevant: When a system mechanism controls access to an object and an individual user cannot alter that access, the control is a mandatory access control (MAC), occasionally called a rule-based access control.

The operating system enforces mandatory access controls. Neither the subject nor the owner of the object can determine whether access is granted. Typically, the system mechanism will check information associated with both the subject and the object to determine whether the subject should access the object. Rules describe the conditions under which access is allowed.

## 2.1. Role Based Access Control (RBAC)

The ability, or need, to access information may depend on one's job functions. This suggests associating access with the particular job of the user. For this reason a special type of access control is introduced called role-based access control (RBAC). We note here some definitions and some rules of this access control type to have a background in next sections.

A *role* is a collection of job functions. Each role  $r$  is authorized to perform one or more transactions (actions in support of a job function). The set of authorized transactions for  $r$  is written  $trans(r)$ . The active role of a subject  $s$ , written  $actr(s)$ , is the role that  $s$  is currently performing. The authorized roles of a subject  $s$ , written  $authr(s)$ , is the set of roles that  $s$  is authorized to assume. The predicate  $canexec(s, t)$  is true if and only if the subject  $s$  can execute the transaction  $t$  at the current time.

Three rules reflect the ability of a subject to execute a transaction.

**Rule 1:** Let  $S$  be the set of subjects and  $T$  the set of transactions. The rule of *role assignment* is  $(\forall s \in S)(\forall t \in T)[canexec(s, t) \rightarrow actr(s) \neq \emptyset]$ . This axiom simply says that if a subject can execute any transaction, then that subject has an active role. This binds the notion of execution of a transaction to the role rather than to the user.

**Rule 2:** Let  $S$  be the set of subjects. Then the rule of *role authorization* is  $(\forall s \in S)[actr(s) \subseteq authr(s)]$ . This rule means that the subject must be authorized to assume its active role. It cannot assume an unauthorized role. Without this axiom, any subject could assume any role, and hence execute any transaction.

**Rule 3:** Let  $S$  be the set of subjects and  $T$  the set of transactions. The rule of *transaction authorization* is  $(\forall s \in S)(\forall t \in T)[canexec(s,t) \rightarrow t \in trans(actr(s))]$ . This rule says that a subject cannot execute a transaction for which its current role is not authorized.

The forms of these axioms restrict the transactions that can be performed. They do not ensure that the allowed transactions can be executed. This suggests that role-based access control (RBAC) is a form of mandatory access control. The axioms state rules that must be satisfied before a transaction can be executed. Discretionary access control mechanisms may further restrict transactions.

Capturing the notion of mutual exclusion requires a new predicate. Let  $r$  be a role, and let  $s$  be a subject such that  $r \in auth(s)$ . Then the predicate  $meauth(r)$  (for mutually exclusive authorizations) is the set of roles that  $s$  cannot assume because of the separation of duty requirement.

### 3. Applying Fuzzy Logic to Role Based Access Control

Recently distributed systems are becoming more and more critical. Designing a distributed system now requires much more responsibility. Access control systems play a key role in security systems. Classic access control systems sometimes do not satisfy the system needs. Some other parameters such as user hostility and amount of damage system can tolerate can be used to implement access control systems. We define  $\tau$  the amount of damage that system can tolerate by serving any user. The main goal of fuzzy access control method is to protect the systems from damages more than  $\tau$ .

Previous works on the subject [1] have provided a fuzzy access control implementation in distributed systems. Also they have provided a similar method for threat analysis in distributed system [2]. In this paper we will present two methods to implement fuzzy role based access control systems. One of the concerns with role based access control is determining active role set for a user from  $Auth(x_i)$ . In most of the role based system the active role set is always the same as  $Auth(x_i)$ , but in some cases this

may threaten the system security. In the next section we will present a method to guarantee the damage posed by hostile users to be less than  $\tau$ .

Also fuzzy method can be applied to evaluate the probability of user being hostile  $Ph_i$ . We will present a method to evaluate  $Ph_i$  so that limitation of a role based systems are applied to fuzzy decision making system. This can be used in combination with other fuzzy values for  $Ph_i$ .

#### 4. Fuzzy Role Assignment

In this section we will present a way to determine whether a role  $R_j$  can be used in active role set of a user  $x_i$  or not (Of course  $R_j \in Auth(x_i)$ ). While  $R_j$  can perform a set of operation  $o_k$  on data  $d_l$ . The term  $EW_{kl}$  is used for the maximum amount of damage that performing operation  $o_k$  on data  $d_l$  will cause. We will use the term  $MR_j$  for the maximum amount of damage that the role  $R_j$  can make.  $MR_j$  is evaluated as follow:

$$MR_j = \sup\{ EW_{kl} \mid R_j \in trans(o_k, d_l)\}$$

Given two fuzzy sets  $A$  and  $B$ , their maximizing set is the fuzzy set that contains all the supports from  $A$  and  $B$  with the degree for each support is the ratio of the support itself to the maximum support of  $A$  and  $B$ . Then we will use Jain method [3] to determine whether grant the user  $x_i$  to activate role  $R_j$  or not. We will first compute

$E = Ph \times MR$ . Where  $A \times B = \sum_{i=1}^n \sum_{j=1}^m \text{Min}\{A(x_i), B(y_j)\} / (x_i \cdot y_j)$  Then we

will compute  $E_{ij} \wedge M$  and  $\tau \wedge M$ . Where  $M$  is the maximizing set for  $E_{ij}$  and  $\tau$ . If

$GMV(E_j \wedge M) < GMV(\tau \wedge M)$ <sup>1</sup> then  $R_j$  can be active role set of the user  $x_i$ .

Probability of hostility of a user can be computed by considering several parameters and an adaptive relation  $R$ . The method for computing probability of hostility is presented in [1].

#### 5. Implementing Role Based Access control using Fuzzy method

---

GMV stand for Greatest Membership value<sup>1</sup>

In this section we will model a Role Based system using a fuzzy model. We will apply the method presented in [1] to the system, also we will evaluate the probability of user hostility according to a role based system. So that, no user can perform operations which role based system did not grant.  $Ph_i$  is evaluated as:

$$Ph'_i = \sup\{EW_{kl} \mid \forall r \in Active(x_i) : r \notin trans(o_k, d_l)\}$$

$Ph_i = Ph'_i \wedge Ph''_i$ , Where  $Ph''_i$  is the probability of user  $x_i$  to be hostile proposed by adaptive method by [1].

Now we can use  $Ph_i$  as before. Evaluate  $E = Ph \times EW$ . Then compute  $E_{ikl} \wedge M$  and  $\tau \wedge M$ . If  $GMV(E_{ikl} \wedge M) < GMV(\tau \wedge M)$  then grant user  $x_i$  to perform operation  $o_k$  on data  $d_l$ .

## 6. Conclusion & Future work

The method presented by [1] is shown to be applicable for RBAC system. This will solve the concern with role assignment problem. Another method to train system with Fuzzy parameters is also presented. The method can be applied in other Access Control fields.

## References

- [1] Alo R., Berrached A., De Korvin A., Beheshti M., (1998), *Using Fuzzy Relation Equations for Adaptive Access Control in Distributed Systems*.
- [2] Berrached A., Beheshti M., De Korvin A., Alo R., (1998), *Applying Fuzzy Relation Equations to Threat Analysis*. In proc. of 35th Hawaii International Conference on System Sciences, 2002.
- [3] Jain, R., (1977), A Procedure for Multiple Aspect Decision Making Using Fuzzy Sets, Int. J. System Science.
- [4] Krause M. and Tipton F.H., (1998), *Handbook of Information Security Management*, CRC Press LLC, Boca Raton, Florida.
- [5] Math Bishop, *Computer Security*, Addison Wesley, November 29, 2002.
- [6] Kandel, "Fuzzy Statistics and Policy Analysis", *Fuzzy Sets: Theory and Applications to Policy Analysis and Information Systems*, ed. By P. Wang and S. Chang, Plenum Press, New York, 1980.
- [7] H.-J. Zimmermann, "Fuzzy Set Theory and Its Application", By Kluwer Academic Publishers, Third Edition 1996.