ICIC Express Letters
Volume **8**, Number **6(tentative)**, June **2014**

ICIC International ©2014 ISSN 1881-803X
pp. **1–EL13-0549**

AN EFFICIENT ROLE-BASED ACCESS CONTROL APPROACH IN IRBAC 2000

Xiaopu Ma[1,*], Ning Cheng[1], Li Zhao[1] and Ruixuan Li[2]

[1]School of Computer and Information Technology
Nanyang Normal University
No. 1638, Wolong Road, Wolong District, Nanyang 473061, P. R. China
*Corresponding author: mapxiao@nynu.edu.cn

[2]School of Computer Science and Technology
Huazhong University of Science and Technology
No. 1037, Luoyu Road, Hongshan District, Wuhan 430074, P. R. China
rxli@mail.hust.edu.cn

Received May 2013; accepted August 2013

Abstract. *The Interoperable Role Based Access Control Model 2000 (IRBAC 2000) provides us with a model for the interoperability between Role-Based Access Control (R-BAC) secure domains. It translates the roles from foreign domains to local domains so that the foreign users can get the local access authorization. On the basis of this model, we present an efficient RBAC approach which is based on artificial neural network in order to calculate the permissions for the foreign users instead of searching the role-permission assignments directly. This approach considers the hierarchical relationships among roles and trains the artificial neural network using selected samples of roles as input vectors, and the according permissions as output vectors which can improve the efficiency of access control for the foreign users. Experiments on performance study prove the superiority of the method.*

Keywords: Role-based access control, IRBAC 2000, Artificial neural network

1. **Introduction.** Role-Based Access Control (RBAC) [1,2] is the most popular access control model, and widely deployed as an alternative to traditional discretionary access control (DAC) [3] and mandatory access control (MAC) [4] in enterprise security management and enterprise management products. Perhaps the most distinctive and important feature of the RBAC is the desire to specify and enforce the enterprise-specific security policies in a way that maps naturally to an organization's structure. Its emphasis on controlling who has access to operations on what objects is fundamentally different from information flow security in multi-level secure systems [5]. In this security model, permissions are no longer assigned individually to users, but as a set of permissions through roles. This change in how permissions are administrated often reduces the complexity of access control because the number of roles in an organization is significantly smaller than that of users [6]. Furthermore, it can support three well-known security principles: least privilege [7,8], separation of duties [9] and data abstraction. As a result, RBAC has been implemented successfully in a variety of commercial systems, such as insurance company and bank, and has become the norm in many applications.

Now with the rapid development of network technology and distributed applications, information interaction and cooperation in multi-domains have become increasingly frequent. In order to accomplish the interoperation problem, Kapadia et al. [10] presented the Interoperable Role-Based Access Control Model 2000 (IRBAC 2000) which provides a secure interoperability using dynamic role translation based on RBAC. In this model, the foreign users need to directly search the role-permission assignments in order to get the local access authorization. However, these techniques are time consuming and costly when there are dozens of roles, permissions and tens of thousands of users between the multi-domains. Thus, we present an RBAC approach based on neural network in order to get the local access authorization efficiently.

The remainder of this paper is organized as follows. We discuss related work in Section 2. The limitation in existing applications for getting the local access authorization drives our motivation and Section 3 proposes our notation about how to train the artificial neural network using selected sample of roles as input vectors and the according permissions as output vectors. A summary of our experimental results on simulated data is discussed in Section 4. Finally, Section 5 provides some insight into our ongoing and future work.

2. **Related Work.**

Recently, considerable attention has been paid to researching and addressing the security needs of commercial and civilian government organizations. In those types of access controls, DAC requirements have been perceived as being technically correct for commercial and civilian government security needs, as well as for single-level military systems. MAC is used for multi-level secure military systems, but its use in other applications is rare. However, each organization has unique security requirements, many of which are difficult to meet using traditional MAC and DAC controls. As a result, the RBAC model can meet with these demands. The concept of RBAC began with multi-user and multi-application on-line systems pioneered in the 1970s. The central notion of RBAC is that permissions are associated with roles, and users are assigned to appropriate roles [11]. The most distinctive and important feature of the RBAC is the desire to specify and enforce enterprise-specific security policies in a way that maps naturally to an organization's structure [12].

However, with the rapid proliferation of Internet and related technologies has created tremendous possibilities for the interoperability between domains in the distributed environments [13]. Interoperability provides a means for domains to share resources and services, which enhances performance and resource utilization. Furthermore, the interoperability does not come easy as it opens the way for several security and privacy breaches. Secure interoperation between domains is a crucial technique of resource sharing and security in the distributed environment. With respect to establish interoperability between domains, Kapadia presented an IRBAC 2000 model of secure interoperability using dynamic role translation based on RBAC. However, the traditional approach is costly and time-consuming since the number of roles, users and permissions in the multi-domains may be very large.

To this aim, this research tries to use artificial neural network to provide the local access permissions for the foreign uses in order to improve the mapping efficiency in a feasible way. Our focus is on how to train the artificial neural network and how to generate the permissions based on the foreign users mapping roles. The experimental results are tested to show the effectiveness of our findings.

3. **Searching Permissions Based on Artificial Neural Network.**

3.1. **Preliminaries.**

This section gives a precise description for the IRBAC 2000. It assumes that there are four countably infinite sets: R (the set of all possible roles), U (the set of all possible users), P (the set of all possible permissions), and D (the set of all possible domains).

Definition 3.1. *(IRBAC state). An IRBAC state γ is a 4-tuple $\langle UA, PA, RH, RP\rangle$, in which the user assignment relation $UA \subset U \times R$ associates users with roles in the local domain, the permission assignment relation $PA \subset R \times P$ associates roles with permissions in the local domain, the role hierarchy relation $RH \subset R \times R$ specifies an acyclic relation among roles, and the role mapping relation between multi-domains $RP \subset RL \times RF$ reflects role mappings between local domain roles RL and foreign domain roles RF.*

An IRBAC state $\gamma = \langle UA, PA, RH, RP\rangle$ determines the set of roles of which each user is a member, and the set of permissions for which each user is authorized, and the set of role mappings for establishing a flexible dynamic role translation between different domains. Now we use an example to illustrate the concepts in this paper. This example is shown in Figure 1.

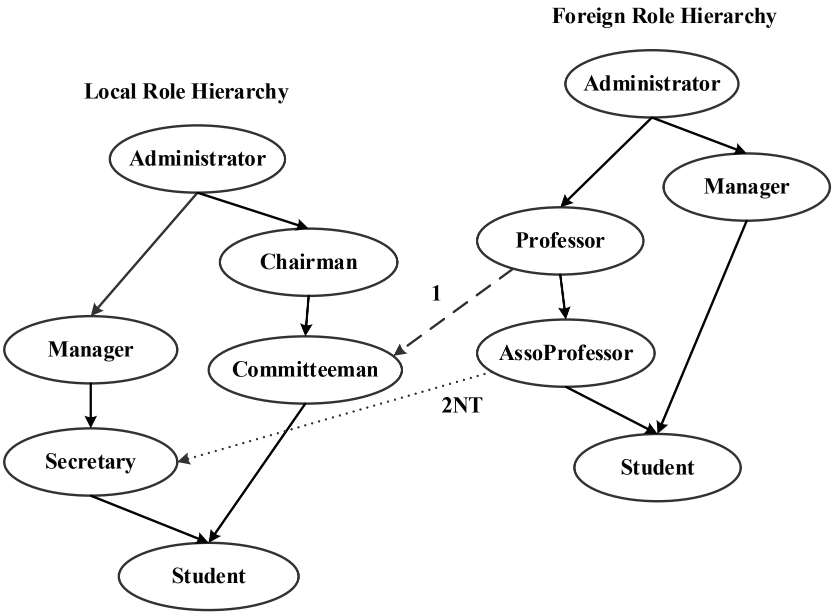

Figure 1. Associations between multi-domains

In Figure 1, there are two domains, one is the local domain, the other is the foreign domain. The interoperation between these domains is achieved by introducing role mapping between the local domain and the foreign domain. There are two types of role mapping. One is transitive associations, such as $Professor_{Foreign} \rightarrow Committeeman_{Local}$ which is labeled as 1 in Figure 1. In this situation, the role $Professor_{Foreign}$ from the foreign domain will be translated to the role of $Committeeman_{Local}$ in the local domain. This role mapping also implies that all the ancestors of role $Professor_{Foreign}$ from the foreign domain will map to the role $Committeeman_{Local}$ in the local domain. The other is no-transitive associations, such as $AssoProfessor_{Foreign} \longmapsto Secretary_{Local}$ which is labeled as 2NT in Figure 1. In this situation, the role $AssoProfessor_{Foreign}$ will be translated to $Secretary_{Local}$ and deny $Professor_{Foeign}$ and $Administrator_{Forign}$ from inheriting this association[14].

Here we can use an $m \times n$ matrix M to describe the relationships between roles and permissions where m is the number of roles and n is the number of permissions for the local domain. The element $M\{i, j\} = 1$ denotes that the ith role has the jth permission or the jth permission belongs to the ith role; otherwise, the element $M\{i, j\} = 0$ indicates that the ith role has not the jth permission. We use fu_i to indicate that the ith foreign user and $UserRoles(fu_i)$ $(i = 1, \ldots, k)$ to indicate the set of mapping roles in the local domain for the ith foreign user.

3.2. **Algorithms.** This section presents the details of the new approach using Artificial Neural Network to obtain the foreign users' access permissions. We use Back Propagation Neural Network (BPNN) [15] for training the role-permission assignments. Figure 2 shows the structures of the BPNN. It consists of three layers: input layer is applied to the inputs of the information; and signals propagate through the hidden layers to the output layer. Each link between neurons has a unique weighting value. In this figure, $x_1, x_2, \ldots, x_n$ means the input information, $w_{i,j}$ means the weight between the ith neuron in input layer

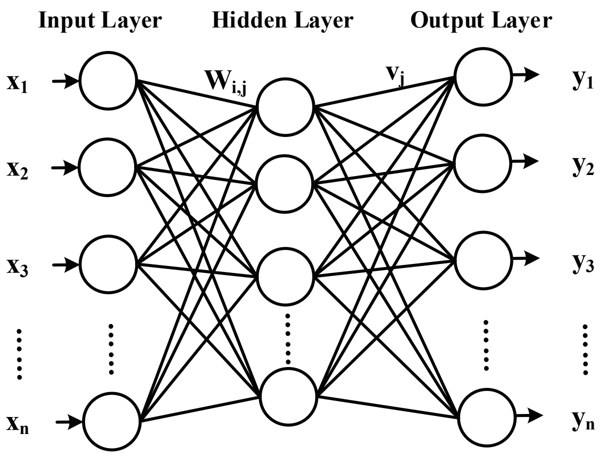

FIGURE 2. The three layers back propagation neural network

and the jth neuron in hidden layer, v_j means the weight between the jth neuron in hidden layer and the neuron in output layer, and $y_1, y_2, \ldots, y_n$ means the output information.

Based on the artificial neural network, the process of the efficient role-based access control approach in IRBAC 2000 can be decomposed into two sub-processes:

(1) Training Stage: In this stage, we analyze the relationships between roles and permissions in the local domain, especially the hierarchy among roles, then choose roles in R as the input sample for the neural network, permissions P as the outputs of the network correspondingly to train the BPNN;
(2) Operation Stage: In this stage, we achieve *UserRoles* of the foreign user, then receive the role encoding of the foreign user. Furthermore, we regard the role encoding as the input vector. Finally, we can get the output vector as the access permissions for the foreign user.

The following example demonstrates the idea of the proposed method. Assume a hypothetical local organization has 7 roles. Table 1 shows the encoding of roles and the relationships between roles and permissions. In the training stage, we use ⟨1000000⟩, ⟨0100000⟩, ⟨0010000⟩, ⟨0001000⟩, ⟨0000100⟩, ⟨0000010⟩, ⟨0000001⟩ as the input vector, ⟨11111111111⟩, ⟨11111100000⟩, ⟨11110011000⟩, ⟨00111110000⟩, ⟨00101101000⟩, ⟨0000000 0111⟩, ⟨00000101111⟩ as the output vector respectively to train the BPNN. In the operation stage, the system will obtain the permissions of the foreign user when setting the foreign user's mapping roles $UserRoles$ as the input information for the artificial neural network. Hence, the foreign user can obtain the access authorization in the local domain through the dynamic role mapping efficiently. For example, the input information is ⟨0110000⟩ when the *UserRoles*$(fu_i) = \{r_2, r_3\}$, the output information is ⟨11111111000⟩.

TABLE 1. Sample data for an example organization

	coding	p_1	p_2	p_3	p_4	p_5	p_6	p_7	p_8	p_9	p_{10}	p_{11}
r_1	1000000	1	1	1	1	1	1	1	1	1	1	1
r_2	0100000	1	1	1	1	1	1	0	0	0	0	0
r_3	0010000	1	1	1	1	0	0	1	1	0	0	0
r_4	0001000	0	0	1	1	1	1	1	0	0	0	0
r_5	0000100	0	0	1	0	1	1	0	1	0	0	0
r_6	0000010	0	0	0	0	0	0	0	0	1	1	1
r_7	0000001	0	0	0	0	0	1	0	1	1	1	1

4. **Experimental Results.** In this section, we will implement the proposed method of obtaining the access authorization based on BPNN on an Intel(R) Core(TM) i5-2400 @CPU 3.1G machine with 4GB memory to evaluate how well our method performs using different metrics.

To study the performance of our method, we generate the synthetic test data as follows. First, we use for loop to create the relationships between roles and permissions. For each role, a random number of permissions are chosen. The value of each element in the matrix is randomly chosen as 0, indicating that the role has no such permission, or 1, indicating that the role has such permission. Finally, we randomly choose the role set as $UserRoles(fu_i)$ for the ith foreign user through the dynamic role mapping.

We present the evaluation of our method (BP method) with the directly searching role-permission assignments method (DS method). We are interested in two things: the accuracy of our method, and how quickly it finds it. Table 2 shows the test parameters.

Table 2. Parameter settings for testing performance

	Number of roles	*Number of permissions*
data1	10000	100
data2	20000	100
data3	100000	1000
data4	200000	2000

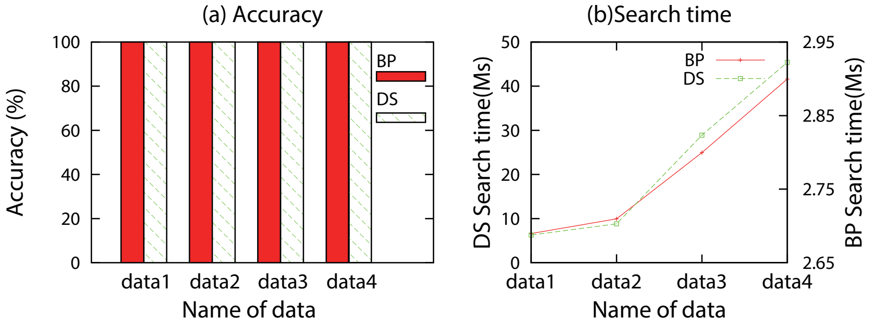

Figure 3. Performance and accuracy comparison under the different data sets

Figure 3(a) shows the accuracy of the different algorithms under the different data sets. From the figure, we can see that both of the approaches have the same accuracy. Furthermore, we can also see that the accuracy of the two approaches is quite good, with the largest number of roles and permissions getting to 100%. Figure 3(b) shows the average search time under the different number of roles and permissions. We can see that our approach costs less time. If the number of permissions and roles in the role permission assignments is larger, the advantage of our approach will be more obvious. Hence, it can decrease the workload for the foreign user to obtain the access authorization in the local domain through the dynamic role mapping.

5. **Conclusions and Future Work.** RBAC has become the norm in many applications. Furthermore, the IRBAC 2000 provides us with a model for the interoperability between RBAC secure domains. Hence, how to get the local access authorization for the foreign users efficiently is a considerable task before the foreign users can access the objects in the local domain. Hence, in this paper, we present an efficient RBAC approach which is based on artificial neural network in order to calculate the permissions of the foreign users instead of searching the role-permission assignments directly. This approach considers the hierarchical relationships between roles and trains the artificial neural network using roles

as input vectors, and the according permissions as output vectors which can improve the efficiency of access control. As a result, the proposed approach has superior performance to traditional methods in speed. For the future work, we will evaluate our technique with others especially the algorithms based on neural network.

Acknowledgment. This work is supported by National Natural Science Foundation of China under grants 61173170, 60873225 and 70771043, National High Technology Research and Development Program of China under grant 2007AA01Z403, Natural Science Foundation of Hubei Province under grant 2009CDB298, Innovation Fund of Huazhong University of Science and Technology under grants 2011TS135 and 2010MS068, CCF Opening Project of Chinese Information Processing, and Innovation Fund of Nanyang Normal University under grants ZX2013013. The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

REFERENCES

[1] M. Liu, X. Wang and H. Zhao, Research on illegal information flow in role-based access control model based on petri net, *ICIC Express Letters*, vol.6, no.1, pp.139-144, 2012.

[2] R. Sandhu, E. J. Coyne, H. L. Feinstein and C. E. Youman, Role-based access control models, *IEEE Computer*, vol.29, no.2, pp.38-47, 1996.

[3] B. W. Lampson, Protection, *Proc. of the 5th Princeton Symposium on Information Science and Systems*, pp.437-443, 1971.

[4] R. S. Sandhu, Role hierarchies and constraints for lattice-based access control, *Proc. of the 4th European Symposium on Research in Computer Security*, Rome, Italy, pp.25-27, 1996.

[5] D. F. Ferraiolo, J. A. Cugini and D. R. Kuhn, Role-based access control (RBAC): Features and motivations, *Proc. of the 11th Annual Computer Security Applications Conference*, pp.241-248, 1995.

[6] D. Zhang, K. Ramamohanarao, T. Ebringer and T. Yann, Permission set mining: Discovering practical and useful roles, *Proc. of the Annual Computer Security Applications Conference*, Anaheim, CA, USA, pp.247-256, 2008.

[7] R. Li, Z. Tang, Z. Lu and J. Hu, Request-driven role mapping framework for secure interoperation in multi-domain environments, *International Journal of Computer Systems Science and Engineering*, vol.23, no.2, pp.193-207, 2008.

[8] X. Ma, R. Li, Z. Lu, J. Lu and M. Dong, Specifying and enforcing the principle of least privilege in role-based access control, *Concurrency and Computation: Practice and Experience*, vol.23, no.12, pp.1313-1331, 2011.

[9] G. J. Ahn and R. S. Sandhu, Role-based authorization constraints specification, *ACM Transactions on Information and System Security*, pp.207-226, 2000.

[10] A. Kapadia, J. Al-Muhtadi, R. Campbell and D. Michunas, IRBAC2000: Secure interoperability using dynamic role translation, *Technical Report: UIUCDCS-R-2000-2162*, 2000.

[11] M. P. Gallagher, A. O'Connor and B. Kropp, The economic impact of role-based access control, *National Institute of Standards and Technology, Planning Report 02-1*, 2002.

[12] A. Schaad, J. Moffett and J. Jacob, The role-based access control system of a european bank: A case study and discussion, *Proc. of the 6th ACM Symposium on Access Control Models and Technologies*, Chantilly, VA, USA, pp.3-9, 2001.

[13] X. Wang, X. Yang, C. Huang and D. Wu, Security violation detection for RBAC based interoperation in distributed environment, *IEICE Transactions on Information and Systems*, pp.1447-1456, 2008.

[14] X. Ma, R. Li, Z. Lu and J. Lu, Global static separation of duty in multi-domains, *Proc. of the International Conference on Multimedia Information Networking and Security*, Wuhan, China, pp.506-509, 2009.

[15] T. D. Sanger, Optimal unsupervised learning in a single-layer linear feed forward neural network, *Neural Networks*, vol.12, pp.837-863, 1989.