# Mutual Exclusion and Role Inheritance affecting Least Privilege in RBAC

Muhammad Asif Habib
*FIM, Johannes Kepler University, Austria*
*habib@fim.uni-linz.ac.at*

## Abstract

*Role based access control (RBAC) always provides tight security of information and ease of management to security policy. There are certain constraints which make the information security tight. Separation of duty (SOD) in terms of mutual exclusion and role inheritance (RI) are some of those constraints which provide security of information and make the management of security policy easy. On one side after implementing separation of duty, we may able to get tight security but on the other side it can create complexity for the security administrator and the user who uses the system. In this paper we describe the complexities and complications which can be faced after implementing separation of duty in terms of mutually exclusive roles (MER). We also describe the problems which can be faced If either the role inheritance is not implemented or implemented in an incomplete manner. We also propose the solutions to the given problems and propose a model against all the problems discussed.*

## 1. Introduction

Information security in any organization always remains a complex and challenging task. There are so many security policies exist where the information security is made sure but they create so many problems for the users and security administrators that they are not considered worth implementing. So, there should be a balance between the security and ease of management of the security policy. Therefore information security in any organization should be made sure and also there should be an ease of management of security policy both for users and security administrators. Role based access control (RBAC) is an evolution in access control for information security and ease of management. RBAC offers different modules to implement as per organizational requirements. There are different constraints in RBAC which make sure the security of information and ease of management. In RBAC users and permissions are assigned to the roles where permissions are the privileges associated with objects [6]. The roles are created by the security administrator while keeping in mind the organizational structure of the organization.

Separation of duty (SOD) is one of the important and affective constraints in RBAC. Separation of duty is defined as static separation of duty (SSOD) and dynamic separation of duty (DSOD) in [15]. Separation of duty is used to enforce information security from internal security threats. Separation of duty is implemented in terms of mutual exclusion of roles. The mutually exclusive roles (MER) implement the separation of duty in a static or dynamic way [7]. The separation of duty constraint binds a user from having one man control which is also an old strategy for making sure the security of information. Role inheritance (RI) is a mechanism or methodology used to define roles in hierarchy as per organizational structure. In role inheritance permissions are inherited from senior roles to junior roles. In role hierarchy (RH) the senior roles have more authority in terms of number of permissions as compared to the junior roles who have lesser authority in terms of lesser number of permissions. Role hierarchy is explained in detail in [2]. The least privilege is a principle used to enhance security of the information with a distinct way. It states that the user should be given the discretion or authority of exercising or activating only the required roles or permission which are necessary to execute the required tasks. The users should not be given extra liberty or discretion which is not necessary to execute the business tasks. The proper implementation of separation of duty in terms of mutually exclusive roles (MER) can enforce the implementation of least privilege principle and on the other side the principle of least privilege can be violated if the SOD is not implemented in a proper way. The principle of least privilege has been described in detail in the framework of role inheritance [8].

The paper is divided into different sections. The research background has been given in next section. In section 3 the discovered problems and complexities are discussed and the proposed model is given in section 4. The analysis of findings is given in section 5. In section 6 we conclude the paper. At last but not least the discussion is given in section 7.

## 2. Research Background

Role based access control (RBAC) is known as tight information security and ease of management to security policy. One of the benefits the RBAC claims is that it provides the implementation of RBAC at different levels. The organizations can implement the RBAC as per organizational requirements. The addition of more and more constraints in RBAC upgrades the level of RBAC. There are different types of constraints in RBAC like separation of duty in terms of mutual exclusion of roles. The separation of duty can be static or dynamic. The next generation of RBAC will be dynamic activation and revocation of roles [5]. The detailed mechanism of dynamic activation and revocation of sessions is given in [10]. The separation of duty has been described in [3] [9]. There is another constraint called least privilege principle [8]. The least privilege principle demands that only the required authority should be given to the user which is necessary to execute the business processes.

The separation of duty is implemented to avoid one man control. One business process is divided into multiple small processes and those processes are assigned to more than one role and at last those roles are assigned to more than one user. In this way we will be able to minimize the chances of committing fraud because a lock which requires more than one key is more secure than a lock which requires only one key to unlock. The SOD is implemented to make information more secure. SOD is used to implement against internal security threats [4] [13]. Any business process which can be executed by only one user produces maximum chances of committing fraud as compared to the business process where more than one user are required to execute one business process. The separation of duty is implemented in terms of mutual exclusion of roles [7]. When the roles are declared mutually exclusive to each other then the user who is authorized to exercise all mutually exclusive roles will be able to execute only one of the mutually exclusive roles due to mutual exclusion of roles [15]. There are different flavors of separation of duty found in literature [12] [1] [11] as static and dynamic SOD, object based dynamic SOD, operational and history based SOD.

Role inheritance in RBAC specifies the organizational structure with reference to roles. The implementation of role inheritance facilitates the security administrators in the administration of security policy [2]. The implementation of role inheritance produces ultimately the implementation of least privilege principle.

## 3. Predicament as a result of Implementing RH and SOD

The security administrator creates roles according to the organizational structure and organizational requirements. Different users are assigned to the roles and a set of permissions is assigned to the role by the security administrator. While assigning permissions to the role, the security administrator declares certain permissions as mutually exclusive to already created permissions. So, the roles are made mutual exclusive on the basis of mutually exclusive permissions. Therefore the role is comprised of two parts one is mutually exclusive part and other is that part which is not mutually exclusive. A role can have two types of permissions, one is mutually exclusive permission and other which is not mutually exclusive permission. The mutual exclusion is used to implement separation of duty which is one of the constraints used to enforce information security.

So, finally a role is a combination of mutually exclusive permissions and the permissions which are not mutually exclusive to other permissions. A user who is authorized to have two mutually exclusive roles can activate only one role due to mutual exclusion of roles. If the user has already activated one of two mutually exclusive roles then the user will not be allowed to activate other mutually exclusive. In this way the user can not activate the part of other mutually exclusive role which is not mutually exclusive. This is one the problem which the user has to face. The user should be allowed to activate those parts of all mutually exclusive roles which are not mutually exclusive. Thus due to the implementation of separation of duty in RBAC in terms of mutual exclusion, the user is prohibited to use that part of the role which is not mutually exclusive even though the user is authorized to activate that par of the role. This happens as a result of implementing the mutual exclusion.

Role inheritance is used to specify organizational structure in terms of roles. The permissions of roles are inherited to other roles. If the role hierarchy is not implemented then the administration of security policy will be more complicated as compared to the policy where role hierarchy is implemented. The role hierarchy has been explained in [8]. There is a need to implement role hierarchy in a complete and proper manner. In role hierarchy the roles which are at top level have higher number of permissions as compared to the roles which are at lower level. As the level of the role is increased the size of the role is also increased in terms of number of permissions. On the same pattern as the level of the role is decreased the size of the role in terms of number of permissions is decreased. As we will move up in role hierarchy the role size expands in terms of number of permissions and as we will move down in role hierarchy the role size shrinks in terms of number of permissions.

The principle of least privilege is used to enforce the information security which states that the user should be given only so much authority which is very necessary to execute the business tasks. So, the size of the roles should be smaller instead of roles expansion as it happens in role hierarchy. Normally in role hierarchy the permissions of junior roles are added into the permissions of the senior roles. So, ultimately the roles become bigger as we move upward in role hierarchy which is against the concept of least privilege.

The proposed model is given in next section as a remedy to all above stated problems.

## 4. Proposed Solution

There are two main problems which are discussed above and both affect the least privilege principle. The remedy to first problem where the users can not activate that part of MER which has not mutually exclusive permissions is given below. The security administrator creates roles as per structure and organizational requirements and assigns permissions to the roles. The roles are divided into two parts, one part will be comprised of all mutually exclusive permissions and other part will consist of all those permissions which are not mutually exclusive. At the time of permissions assignment, the security administrator will explicitly specify whether the permission is mutually exclusive to any other permission in other role or not. So, both parts of the role will be separated. Main head of the role will remain same but in more depth, the permissions will be separated on sub heads which are mutually exclusive or not. Therefore a role is divided into two sub roles, one sub roles is comprised of all mutually exclusive permissions and other sub role consists of permissions which are not mutually exclusive.

The user who is authorized to exercise mutually exclusive roles will be authorized to activate all those parts of the MER consisting of permissions which are not mutually exclusive. In this way the user will be practically authorized to activate more and more number of roles which the user has already authorized to do so. In this way the roles will be divided into two parts which will result in the reduction of the size of the roles. This will help to enforce least privilege principle. This approach will have a drawback in the sense that the security administrator will have to do a lot in separating roles in two different sub heads. Also when there will be a change in the role structure then the security administrator will have to do a lot of work in this respect. Ultimately the authorized users will get practical authority not theoretical authority but on the other side the ease of management will be sacrifice to some extent. The theoretical authority here we mean that if a user qualifies for an authority to activate a role or a part of the role but the user will be restrained to do so and practical authority means that if a user qualifies for an authority and the user can exercise that authority.

Another problem is the big size of the senior roles in role hierarchy in terms of number of permissions. The size of the role in terms of number of permissions increases as we move upward in role hierarchy. On the same way the size of the role in terms of number of permissions decreases as we move downward in role hierarchy. This creates big roles in terms of number of permissions which is against the least privilege principle. This can violate the security of information because big roles have more permission, once if any unauthorized user will be able to get control of a senior role then that unauthorized user can damage the security of information to a maximum level.

The solution to the problem of big roles with greater number of permissions is that the roles at higher level in role hierarchy will have lesser number of permission but more number of roles.

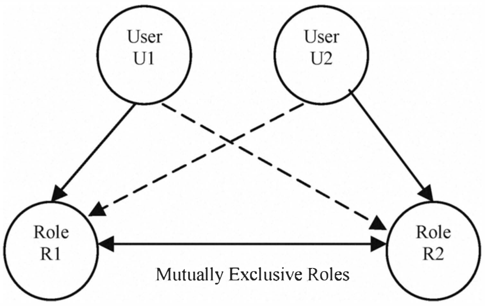

**Figure 1. Mutually exclusive roles**

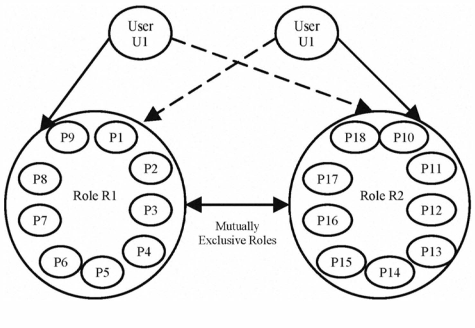

**Figure 2. Mutually exclusive role structure**

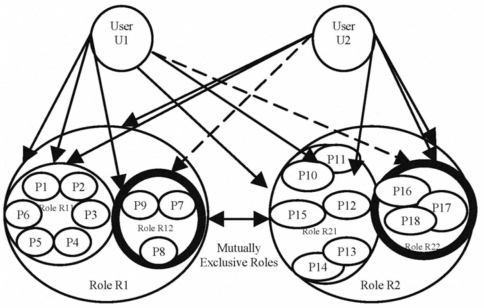

**Figure 3. Mutually exclusive role partitioning**

The roles which are assigned to the user at higher level of hierarchy should have many roles but every role should have minimum number of permissions. Instead of addition of permissions from junior roles to the senior roles, the junior roles should be assigned to the senior roles. This will be good for making sure the information security as well as it will be an ease of management for security administrators. This has been explained in detail in section 5.2.

## 5. Analysis of Findings

The proposed model is elaborated in more detail with the help of following examples. The first example will be about mutually exclusive roles and second example will be about role inheritance.

### 5.1. Partitioning of Role

Suppose we have two users U1 and U2 authorized to activate two mutually exclusive roles R1and R2. If any user will activate one of two mutually exclusive roles then that user will not be able to activate other role partially or fully. In Figure 1 user U1 activated role R1 and user U2 activated role R2. The dashed arrows show that the users can not activate those roles due to mutual exclusion.

In Figure 2, all the permissions of each role are shown. Role R1 is comprised of permissions P1 to P9 and role R2 is comprised of permissions P10 to P18. Some of the permissions of each role are mutually exclusive and some of them are not mutually exclusive. In Figure 3 two separate sub roles of each main role are formed with one role having all permissions as mutually exclusive and other role having all those permissions which are not mutually exclusive.

Role R1 has permissions P7 to P9 as mutually exclusive permissions and P1 to P6 permissions which are not mutually exclusive. One the same pattern role R2 has permissions from P10 to P15 which are not mutually exclusive and permissions from P16 to P18 which are mutually exclusive. Role R1 has been divided into two sub roles R11 and R12. The role R11 is comprised of permissions which are not mutually exclusive and R12 is comprised of mutually exclusive permissions. On the same pattern Role R2 has been divided into two sub roles R21 and R22. The role R21 consists of permissions which are not mutually exclusive and role R22 consists of mutually exclusive permissions. Now any user U1 or U2 can activate roles R11 and R21 because both roles are comprised of permissions which are not mutually exclusive. The arrows with dotted lines represent that these roles can not be activated by the same user due to mutual exclusion.

Now we can see that the user is given the practical authority to activate all those permissions and roles which are not mutually exclusive and for which the user is authorized to exercise those roles. The size of the roles is decreased in terms of number of permissions as a result of the partitioning of role which will help to enforce security of information in terms of least privilege principle.

### 5.2 Role Inheritance

We see that the size of the roles of the users in terms of number of permissions in role hierarchy increases on top levels as compared to low levels. We support the idea of assigning roles of the junior employees to the senior employees either fully or partially instead of addition of permissions of junior roles to the senior roles either fully or partially. In this way the size of the senior roles increases which creates problems for information security with respect to principle of least privilege.

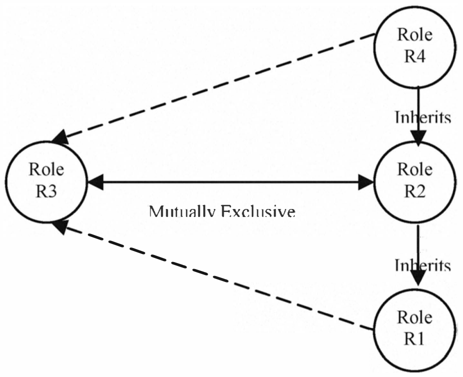

**Figure 4. Mutually exclusive role inheritance**

In Figure 4 we have four different roles R1, R2, R3 and R4 having certain permissions. Role R3 and role R2 are made mutually exclusive to each other. Role R1 is at the lowest level of role hierarchy and role R4 stands on the highest level of role hierarchy. While roles R2 and R3 are in between role R1 and role R4 in role hierarchy. Role R4 has maximum number of permissions due to highest level role as it has inherited permissions from role R1 and R2. Each role has some specific number of permissions as below.

R1 = {P5, P8, P10, P12}
R2 = {P5, P6, P7, P8}
R3 = {P1, P2, P3, P4}
R4 = {P22, P23, P5, P10, P6, P7}

We can see in this example that bigger the organization will be bigger the size of the senior roles will be in terms number of permissions. So, instead of using technique of assigning permissions of junior roles to the senior role for implementing role inheritance, another way would be recommend of implementing role inheritance. The way is to assign the junior roles or part of roles to the senior users. The user who is authorized to exercise role R4 should also be authorized to execute two more roles. One of those roles will comprise of two permissions P5 and P10 inherited from role R1 and second role will consist of two permissions P6 and P7 inherited from role R2.

In this way the higher level users in role hierarchy will have maximum number of roles and each role will have less number of permissions. This will help to implement principle of least privilege.

R1 = {P5, P8, P10, P12}
R2 = {P5, P6, P7, P8}
R3 = {P1, P2, P3, P4}

R1$'$ = {P5, P10}
R2$'$ = {P6, P7}
R4 = {P22, P23}
R4$'$ = R1$'$ + R2$'$ + R4

The role R4 being a senior most role has some certain permissions which are part of the role R1 and R2. Instead of assigning those junior level permissions to the senior level role, it is recommended to take those permissions as roles and then assign those roles to the senior role R4. Thus new role R4$'$ will consist of three new roles R1$'$, R2$'$ and R4. In role inheritance the above process shows that the senior users will have maximum number of roles with less number of permissions. In this way the size of the users will be small in terms of number of permissions.

The big roles in terms of number of permissions are at greater security risk. Suppose if an unauthorized user will be able to get control of the big role in anyway then due to big authority domain of the user the unauthorized user can damage at its maximum level. But if a role will be small in size in terms of number of permissions then the role will be at lesser security risk due to limited authority domain of the user as compared to the big roles with maximum authority domain. So, this will help to implement information security as well as an ease of management for security administrator.

## 6. Conclusion

RBAC provides security of information and ease of management due to its constraints. But if these constraints especially the separation of duty in terms of mutually exclusive roles and role inheritance will be implemented in its real spirit then it can give more benefits to information security and ease of management.

We have proposed to separate the role in two parts, first which is containing permissions which are not mutually exclusive and second which is containing mutually exclusive permissions. In this way the users can exercise their full authority which is difficult to get without role division on the basis of mutual exclusion of roles. But this has only drawback of increasing the workload of the security administrator. The proposed solution for implementing role inheritance can boost information security due to ultimate implementation of principle of least privilege.

We have tried to show that if SOD and RI constraints are not implemented in the light of proposed model then it can create problems and complications for the users and as well as for the security administrators of the organization. The implementation of SOD and role inheritance can be used for more effective information security.

## 7. Discussion

The above proposed model after implementation gives benefits of security of information and ease of management which the RBAC promise for. Separation of duty in terms of mutual exclusion of roles is important to make information more secure against internal security threats. We believe that after implementing the given model another security constraint will be implemented ultimately which is principle of least privilege. Also in role hierarchy if the roles will be assigned to the users instead of addition of permissions to the senior roles then it will again enforce the concept of least privilege.

There is a long discussion regarding the different possibilities in implementation of SOD in terms of mutual exclusion. There is a need to give detailed specification about different possibilities where the permissions which are mutually exclusive may be common in more than one role. Also there is a need to go in depth while implementing role inheritance. There can be many more complications in role inheritance when there would be common permissions in different roles which are mutually exclusive to the lower or higher level roles at the same time.

## 8. References

[1] Ferraiolo, D., Cugini, J., Kuhn, D. R. “Role-Based Access Control (RBAC): Features and Motivations” Proc. 1995 Computer Security Applications Conference, 241-248, December 1995.

[2] Jonathan D. Moffett. 1998 Control Principles and Role Hierarchies in 3rd ACM Workshop on Role Based Access Control (RBAC), 22-23 October 1998, George Mason University, Fairfax, VA

[3] Gligor, V., Gavrila, S., and Ferraiolo, D. On the formal definition of separation-of-duty policies and their composition. In Proceedings of 1998 IEEE Symposium on Research in Security and Privacy (Oakland, California, 1998), pp. 172–183.

[4] Crampton, J. 2003. Specifying and enforcing constraints in role-based access control. In Proceedings of the Eighth ACM Symposium on Access Control Models and Technologies (Como, Italy, June 02 - 03, 2003). SACMAT '03. ACM, New York, NY, 43-50. DOI= http://doi.acm.org/10.1145/775412.775419.

[5] Ravi Sandhu and Venkata Bhamidipati, The ASCAA Principles for Next-Generation Role-Based Access Control. Proc. 3rd International Conference on Availability, Reliability and Security (ARES), Barcelona, Spain, March 4-7, 2008, pages xxvii-xxxii. Presentation Keynote Lecture.

[6] R. S. Sandhu, E.J. Coyne, H.L. Feinstein, C.E. Youman (1996), "Role-Based Access Control Models", IEEE Computer 29(2): 38-47, IEEE Press, 1996.

[7] Kuhn, D. R. 1997. Mutual exclusion of roles as a means of implementing separation of duty in role-based access control systems. In Proceedings of the Second ACM Workshop on Role-Based Access Control (Fairfax, Virginia, United States, November 06 - 07, 1997). RBAC '97. ACM, New York, NY, 23-30. DOI= http://doi.acm.org/10.1145/266741.266749.

[8] HAN LAN-SHENG, HONG FAN, ASIEDU BAFFOUR KOIO. Least Privileges and Role’s Inheritance of RBAC [J]. Wuhan University Journal of Natural Sciences, 2006(11).

[9] Simon, R., and Zurko, M. Separation of duty in role-based environments. In Proceedings of 10th IEEE Computer Security Foundations Workshop (Rockport, Massachusetts, 1997), pp. 183–194.

[10] Jörg R. Mühlbacher, Christian Praher - DS RBAC - Dynamic Sessions in Role Based Access Control; in: Journal of Universal Computer Science, Vol. 15, Issue 3, pp. 538-554, ISSN 0948-695x (2009).

[11] Habib, Muhammad Asif; Praher, Christian, "Object based dynamic separation of duty in RBAC," Internet Technology and Secured Transactions (ICITST), pp.512-516, 9-12 Nov. 2009.

[12] M.J. Nash and K.R. Poland. Some conundrums concerning separation of duty. In Proceedings of the Symposium on Security and Privacy, pages 201–207, 1990.

[13] Sandhu, R. Transaction control expressions for separation of duties. In Proceedings of 4th Aerospace Computer Security Conference (Orlando, Florida, 1988), pp. 282–286.

[14] Luigi Giuri, Pietro Iglio, "A Formal Model for Role-Based Access Control with Constraints," csfw, p. 136, Ninth IEEE Computer Security Foundations Workshop, 1996.

[15] American National Standard for Information Technology – Role Based Access Control, ANSI INCITS 359-2004.