Mutually exclusive permissions in RBAC

Muhammad Asif Habib

FIM, Johannes Kepler University,
Altenbergerstraße 69, A-4040 Linz, Austria
E-mail: habib@fim.uni-linz.ac.at
and
Department of CS,
National Textile University,
Sheikhupura Road, Faisalabad (37610), Pakistan
E-mail: drasif@ntu.edu.pk

Qaisar Abbas

Department of CS,
National Textile University,
Sheikhupura Road, Faisalabad (37610), Pakistan
E-mail: drqaisar@ntu.edu.pk

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 easier. 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 end users who use the system. Implementing mutual exclusion on the basis of roles reduces the authority of the RBAC user for which the user is authorised. 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 a model to counter the problems.

Keywords: dynamic separation of duty; DSOD; mutually exclusive roles; MERs; conflict of interest; role inheritance; RI; role-based access control; RBAC; mutually exclusive permissions; MEPs.

Reference to this paper should be made as follows: Habib, M.A. and Abbas, Q. (2012) 'Mutually exclusive permissions in RBAC', *Int. J. Internet Technology and Secured Transactions*, Vol. 4, Nos. 2/3, pp.207–220.

Biographical notes: Muhammad Asif Habib received his PhD from FIM, Johannes Kepler University Linz Austria in 2011. Currently, he is working as an Assistant Professor at National Textile University Faisalabad, Pakistan. He is doing research in information and network security especially in role-based access control. He is also working in cloud computing.

Qaisar Abbas obtained his PhD in Computer Application Technology from HUST, China. He is doing research in various fields including networks, artificial intelligence, and bioinformatics. Currently, he is working as an Assistant Professor at the Computer Science Department, National Textile University Faisalabad, Pakistan. He is supervising various projects as well.

1 Introduction

An organisation always invest a lot to make its system secure. The information security 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 organisation 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 organisational 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 (Sandhu et al., 1996). The roles are created by the security administrator while keeping in mind the organisational structure of the organisation.

Separation of duty (SOD) is one of the important and affective constraints in RBAC. SOD is defined as static separation of duty (SSOD) and dynamic separation of duty (DSOD) in ANSI INCITS 359 (2004). SOD is used to enforce information security from internal security threats. SOD is implemented in terms of mutual exclusion of roles. The mutually exclusive roles (MER) implement the SOD in a static or dynamic way (Kuhn, 1997). The SOD 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 organisational structure. In RI, 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. RH is explained in detail in Moffett (1998). 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 SOD in terms of MERs 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 RI (Lan-Sheng et al., 2006).

This paper is divided into different sections. The research background has been given in next section. In Section 3, the discovered problems and complexities in existing RBAC models are discussed and the proposed model is given in Section 4. In Section 5, the model is explained with the help of an illustration. In Section 6, we conclude this paper. And last but not least the discussion is given in Section 7.

2 Research background

RBAC is an evolution in the field of 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 organisations can implement the RBAC as per organisational requirements. The addition of more and more constraints in RBAC upgrades the level of RBAC. There are different types of constraints in RBAC like SOD in terms of mutual exclusion of roles. The SOD can be static or dynamic. The next generation of RBAC will be dynamic activation and revocation of roles (Sandhu and Bhamidipati, 2008). The detailed mechanism of dynamic activation and revocation of sessions is given in Mühlbacher and Praher (2009). The SOD has been described in Gligor et al. (1998), and Simon and Zurko (1997). There is another constraint called least privilege principle (Lan-Sheng et al., 2006). 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 SOD 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 minimise 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 (Crampton, 2003; Sandhu, 1988). 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 SOD is implemented in terms of mutual exclusion of roles (Kuhn, 1997). When the roles are declared mutually exclusive to each other then the user who is authorised to exercise all MERs will be able to execute only one of the MERs due to mutual exclusion of roles (ANSI INCITS 359, 2004). There are different flavours of SOD found in literature (Nash and Poland, 1990; Ferraiolo et al., 1995; Habib and Praher, 2009) as static and dynamic SOD, object-based dynamic SOD, operational and history-based SOD.

RI in RBAC specifies the organisational structure with reference to roles. The implementation of RI facilitates the security administrators in the administration of security policy (Moffett, 1998). The implementation of RI produces ultimately the implementation of least privilege principle.

3 Discrepancies in existing RBAC models

There are so many discrepancies in existing RBAC models. We are going to discuss two main drawbacks of implementing existing RBAC models. First is regarding the decrement in the authority domain of RBAC users as a result of implementing mutual exclusion on the basis of roles and second is about the security threat for bigger roles which come in to being due to the RI. These are discussed in detail in below subsections.

3.1 Drawbacks of implementing mutual exclusion on the basis of roles

In RBAC, the roles are of main interest and centric. The security administrator creates roles according to the organisational structure and organisational requirements. Various users are assigned to the roles and a set of permissions is assigned to the role by the security administrator. In RBAC, the security administrator declares certain roles to be mutually exclusive to some specific roles. This is done because some of the permissions of the roles create conflict of interest with some of the permissions of other roles. Therefore, the roles are declared as MERs only on the basis of mutually exclusive permissions (MEPs). So, a role is comprised of two parts one is mutually exclusive part and other is that part which is not mutually exclusive. This means that a role can have two types of permissions; one is MEP and other which is not mutually exclusive. The mutual exclusion is used to implement SOD which is one of the constraints used to enforce information security.

So, a role is a combination of MEPs and the permissions which are not mutually exclusive to other permissions. A user who is authorised to have two MERs can activate only one role due to mutual exclusion of roles. If the user has already activated one of two MERs then the user will not be allowed to activate other MER. In this way, the user cannot activate that part of other MER which does not have MEPs. Suppose if two roles having hundred permissions each and both roles are made MERs due to five different permissions which cause the conflict of interest with three permissions of other role then as both roles have some permissions which are mutually exclusive that is why they are made as MERs.

If a role will have hundred permissions and only one permission is declared as MEP out of hundred then on this basis the role will be made as MER which is a great drawback of having mutual exclusion. This is one the big drawback of implementing mutual exclusion on the basis of roles. Ultimately, the users have to suffer due to this problem. Generally, if a user has been authorised for any number of permissions under some role and that role becomes mutually exclusive due to certain number of permissions then the user should be able to activate those permissions which are not mutually exclusive at minimum. Thus, due to the implementation of SOD 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 authorised to activate that par of the role. This happens as a result of implementing the mutual exclusion on the basis of roles. Ultimately, the users will have to sacrifice a part of their authority domain to implement SOD.

3.2 RI creates bigger roles

RI is used to specify organisational structure in terms of roles. The permissions of roles are inherited to other roles. If the RH is not implemented then the administration of security policy will be more complicated as compared to the policy where RH is implemented. The RH has been explained in Lan-Sheng et al. (2006). There is a need to implement RH in a complete and proper manner. In RH, 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 RH the role size expands in terms of number of permissions and as we will move down in RH 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 RH. Normally, in RH, 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 RH 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 model

There are two main problems which are discussed above. First problem deals with the decrement in the authority domain of RBAC users due to the implementation of mutual exclusion on the basis of roles and second problem affects directly the least privilege principle and second problem deals with the increment in the size of roles due to RH which is a security threat itself. The remedy to first problem where the users can not activate that part of MER which has not MEPs is given below. A model to counter the problem of decrement in the authority of RBAC users is also given in Habib (2010).

The security administrator creates roles as per structure and organisational requirements and assigns permissions to the roles. The roles are divided into two parts, one part will be comprised of all MEPs and other part will consist of all those permissions which are not mutually exclusive. At the time of permissions assignment, the security administrator will 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 MEPs and other sub role consists of permissions which are not mutually exclusive.

The user who is authorised to exercise MERs will be authorised to activate all those parts of the MER consisting of permissions which are not mutually exclusive. In this way, the user will be practically authorised to activate more and more number of roles which the user has already authorised to do so. The roles will be divided into two parts which will result in the reduction of the size of the roles also. This will help to enforce least privilege principle. This model will have one drawback in the sense that the security administrator will have to specify mutual exclusion on the basis of permissions explicitly.

Thus, the roles will be separated in two different sub heads. Any change in the role structure can also cause the security administrators to do some extra efforts again in this respect. But ultimately, the authorised 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 in case of implementing mutual exclusion users were deprived of their authorised authority to exercise. So, 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 will be considered theoretical authority and practical authority means that if a user qualifies for an authority and that user would be able to exercise that authority at any time.

The other drawback was the big size of the senior roles as a result of implementing RH in terms of number of permissions. The size of the role in terms of number of permissions increases as we move upward in RH. On the same way, the size of the role in terms of number of permissions decreases as we move downward in RH.

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 large number of permissions. Once if any unauthorised user will be able to get control of a senior role then that unauthorised user can damage the security of information to a maximum level because the intruder will get the full control of the role with all the permissions.

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

4.1 Check access

In the light of the proposed model and with the help of proposed model flow chart, the check access to the request of the users is given below step by step:

1 When the system will receive the user's request to activate a particular permission of a role then first of all the system will verify the authority of the user. The system will confirm that whether the user has been authorised to activate the requested role or not.

2 If the user will not be authorised for the requested role then the user will be given a message of access denied.

3 But if the user will be verified with positive authority then the system will verify that whether the role for which the user is making request to activate is mutually exclusive or not. A role will be a MER if that role will have minimum one permission as MEP otherwise the role without any MEP the role will be considered as non-MER.

4 If the role will not be mutually exclusive then the user will get access to activate the role otherwise, the system will check whether the permission for which the user is requesting is mutually exclusive or not.

5 If the requested permission will not be mutually exclusive to any other permission then the user will get access to activate the permission instantly.

6 But if the system finds that the requested permission is mutually exclusive to some other permission then the system will have two alternatives here which depend on the nature of the implementation of the model.

7 One way of doing this will be that the system will find a set of all MEPs which are mutually exclusive to the requested permission. Then the system will compare find whether the user has already activated any one of the MEPs from the set. If the user

has already activated any one of MEPs present in the set then the user will get the message of access denied. But if the user has not activated any of the MEPs in the set then the user will get instant access to activate the permission.

8 The other way of doing this is that the system will find all MEPs which are mutually exclusive to any permission in the requested role. If the user has already activated any one of the MEP then the system will deny the access to the user otherwise the user will get access to activate the permission to activate.

So, in this, the access request from the user will be decided as given in check access step by step.

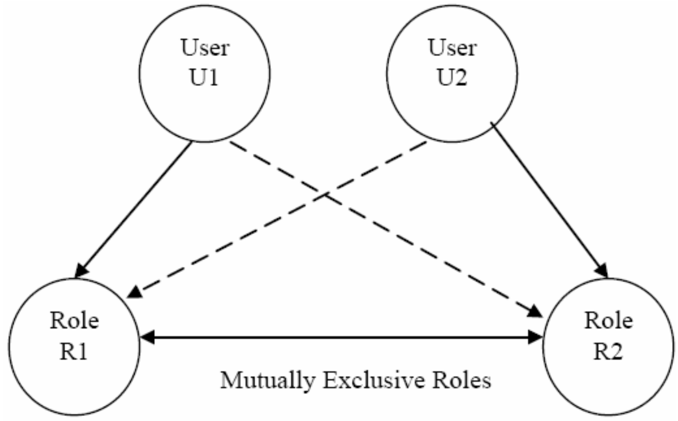
5 Illustrating the proposed model

The proposed model is elaborated in more detail with the help of following examples. The first example will be about MERs and second example will be about RI.

5.1 Partitioning of role

Suppose we have two users U1 and U2 authorised to activate two MERs R1and R2. If any user will activate one of two MERs 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.

Figure 1 Mutually exclusive roles



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 MEPs and P1 to P6 permissions which are not mutually exclusive. One the same pattern role R2 has been assigned with 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 MEPs. 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 MEPs. Now, any user U1 or U2 can activate roles R11 and R21 because both roles are comprised of permissions which are not mutually exclusive.

Figure 2 MER structure

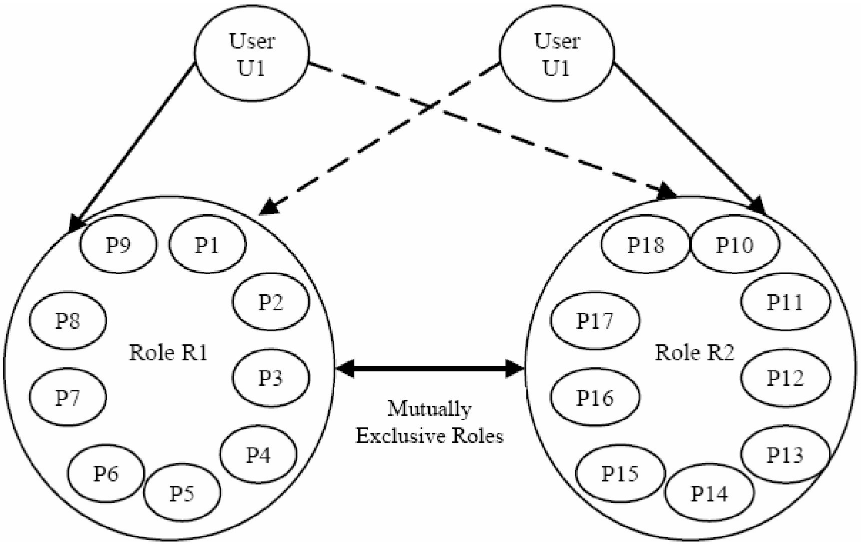


Figure 3 MER partitioning

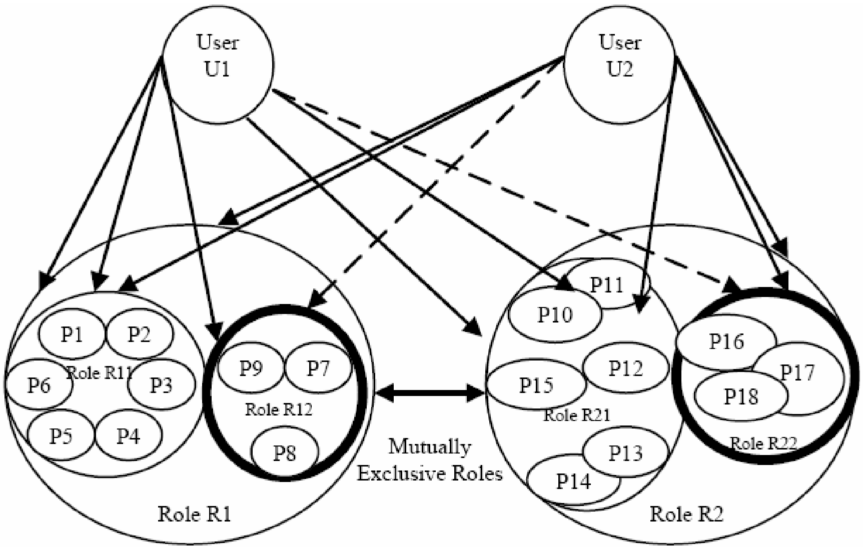
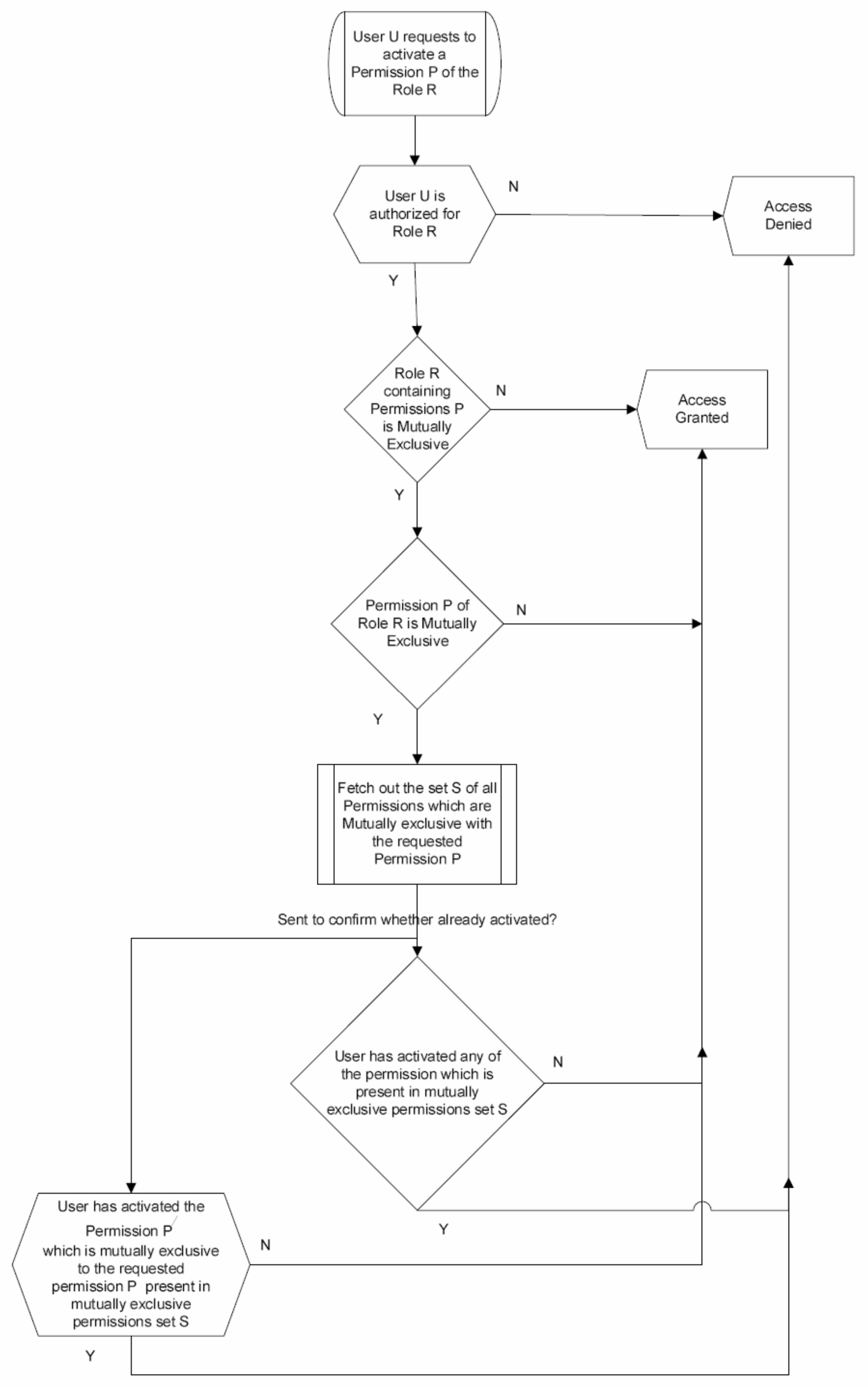


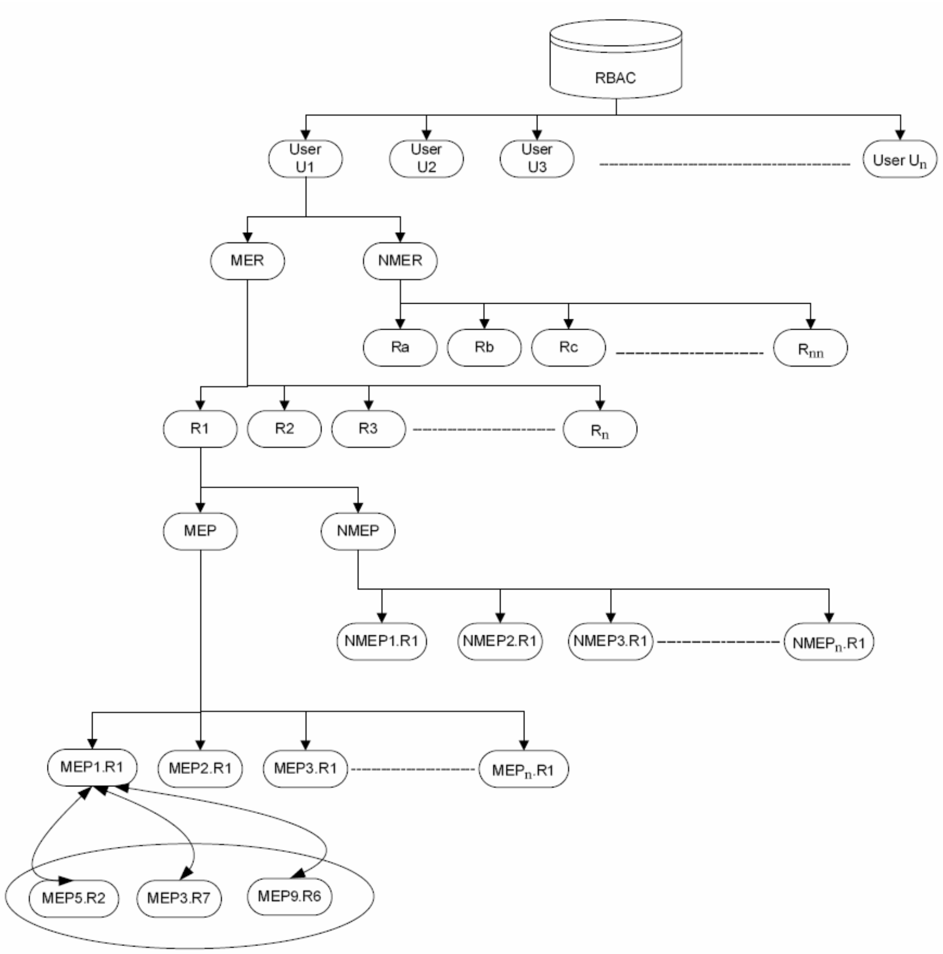
Figure 4 Proposed model flow chart



User U requests to activate a Permission P of the Role R
User U is authorized for Role R
N
Access Denied
Y
Role R containing Permissions P is Mutually Exclusive
N
Access Granted
Y
Permission P of Role R is Mutually Exclusive
N
Y
Fetch out the set S of all Permissions which are Mutually exclusive with the requested Permission P
Sent to confirm whether already activated?
User has activated any of the permission which is present in mutually exclusive permissions set S
N
Y
User has activated the Permission P′ which is mutually exclusive to the requested permission P present in mutually exclusive permissions set S
N
Y

The arrows with dotted lines represent that these roles can not be activated by the same user due to mutual exclusion in Figure 3. We 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 authorised 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. In Figure 5, the complete RBAC tree is given in the light of proposed model. The whole tree can be traversed from top to bottom starting with any user then traversing all MERs and non-mutually exclusive roles (NMER) and then at last reaches the depth of tree to the MEPs and non-mutually exclusive permissions (NMEP). With the help of the model as shown in tree, we can find any permission of any role assigned to any user whether it is mutually exclusive or not. The bold oval which is at the bottom of the Figure 5 shows the set of all MEPs.

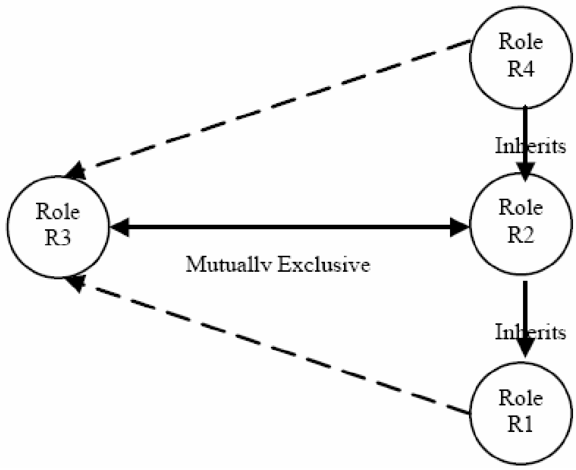
Figure 5 RBAC tree



5.2 Role inheritance

We see that the size of the roles of the users in terms of number of permissions in RH 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. The problems of violating the mutual exclusion due to RI have been given in Habib (2011).

Figure 6 MER 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 RH and role R4 stands on the highest level of RH. While roles R2 and R3 are in between role R1 and role R4 in RH. 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 organisation 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 RI, another way would be recommend of implementing RI. The way is to assign the junior roles or part of roles to the senior users. The user who is authorised to exercise role R4 should also be authorised 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 RH 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 RI, 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 unauthorised user will be able to get control of the big role in anyway then due to big authority domain of the user the unauthorised 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 Conclusions

This is clear that RBAC provides security of information and ease of management due to its constraints. But if these constraints especially the SOD in terms of MERs and RI will be implemented in its real spirit then it can give more benefits to information security and ease of management.

We have proposed a model to divide a role in two parts, first which is containing permissions which are not mutually exclusive and second which is containing MEPs. 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 RI 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 organisation. The implementation of SOD and RI can be used for more effective information security.

7 Discussion

The proposed model gives the benefits of security of information and ease of management which the RBAC promise for. SOD in terms of mutual exclusion of roles is important to make information more secure against internal security threats. In this way, the RBAC users will not loose their authority domain for which they are authorises. We believe that after implementing the given model another security constraint will be implemented ultimately which is principle of least privilege. Also in RH, 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 RI. There can be many more complications in RI when there would be common permissions in different roles which are mutually exclusive to the lower or higher level roles at the same time.

References

ANSI INCITS 359 (2004) 'Information technology – role based access control', American National Standards Institute (ANSI).

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, SACMAT '03*, Como, Italy, 2–3 June, ACM, New York, NY, pp.43–50, DOI = http://doi.acm.org/10.1145/775412.775419.

Ferraiolo, D., Cugini, J. and Kuhn, D.R. (1995) 'Role-based access control (RBAC): features and motivations', *Proc. 1995 Computer Security Applications Conference*, December, pp.241–248.

Gligor, V., Gavrila, S. and Ferraiolo, D. (1998) '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, pp.172–183.

Habib, M.A. (2010) 'Mutual exclusion and role inheritance affecting least privilege in RBAC', *2010 International Conference for Internet Technology and Secured Transactions (ICITST)*, pp.1–6, 8–11 November, available at http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5678530&isnumber=5678008 (accessed on 16/05/2011).

Habib, M.A. (2011) 'Role inheritance with object-based DSD', *Int. J. Internet Technology and Secured Transactions*, Vol. 3, No. 2, pp.149–160.

Habib, M.A. and Praher, C. (2009) 'Object based dynamic separation of duty in RBAC', *International Conference for Internet Technology and Secured Transactions, ICITST 2009*, 9–12 November, pp.512–516, available at http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=5402642 (accessed on 20/08/2010).

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, RBAC '97*, Fairfax, Virginia, USA, 6–7 November, ACM, New York, NY, pp.23–30, DOI = http://doi.acm.org/10.1145/266741.266749.

Lan-Sheng, H., Fan, H. and Koio, A.B. (2006) 'Least privileges and role's inheritance of RBAC', *Wuhan University Journal of Natural Sciences*, Vol. 11, No. 1, pp.185–187.

Moffett, J.D. (1998) 'Control principles and role hierarchies', *Proceedings of the Third ACM Workshop on Role-Based Access Control*, Fairfax, V., 22–23 October, pp.63–69, George Mason University, Fairfax, VA.

Mühlbacher, J.R. and Praher, C. (2009) 'DS RBAC – dynamic sessions in role based access control', *Journal of Universal Computer Science*, Vol. 15, No. 3, pp.538–554, ISSN 0948-695x.

Nash, M.J. and Poland, K.R. (1990) 'Some conundrums concerning separation of duty', in *Proceedings of the Symposium on Security and Privacy*, pp.201–207.

Sandhu, R. (1988) 'Transaction control expressions for separation of duties', in *Proceedings of 4th Aerospace Computer Security Conference*, Orlando, Florida, pp.282–286.

Sandhu, R. and Bhamidipati, V. (2008) 'The ASCAA principles for next-generation role-based access control', *Proc. 3rd International Conference on Availability, Reliability and Security (ARES)*, Barcelona, Spain, 4–7 March, pp.27–32, Presentation Keynote Lecture.

Sandhu, R.S., Coyne, E.J., Feinstein, H.L. and Youman, C.E. (1996) 'Role-based access control models', *IEEE Computer*, Vol. 29, No. 2, pp.38–47, IEEE Press.

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