2011 International Conference on Information and Network Technology
IPCSIT vol.4 (2011) © (2011) IACSIT Press, Singapore

A Modified Conflict Detection Algorithm for Multiple Permission Assignments in Privacy Aware – RBAC Model

M Caroline Joan [1], N Jaisankar[2], A Kannan[3]

[1]carlynmartin2487@gmail.com, [2]jaisasi_win@yahoo.com, [3]kannan@annauniv.edu

[1,2]Department of Computer Science and Engineering,
MNM Jain engineering college, Chennai, Tamil Nadu, India
[3]Department of Information Science and Technology,Anna University, Chennai, Tamil Nadu, India

Abstract. Privacy is a vital requirement for all the organizations. Specifying the privacy policy is crucial problem in Role Based Access Control (RBAC) model. Conflict resolution is another important issue in permission assignment in the Privacy Aware Role Based Access Control (P-RBAC) model. It is proved that there is no conflict up to two permission assignments, but there is a conflict when three or more permission assignments are considered together. To overcome this issue and to improve the efficiency of the system, in this paper we propose a multiple conflict detection algorithm and implemented it for detecting conflicts in three or more permission assignments and redundant permission assignments are reduced in the P-RBAC model. Moreover, by considering features like purpose, condition and obligation in P-RBAC model, the system ensures that highly intricate privacy related policies are expressed properly and conflicts are handled well.

Keywords: Role Based Access Control, P-RBAC, Purpose, Obligation, Conflict Detection Algorithm.

1. Introduction

In the recent past, role-based access control (RBAC) model has received strong support from the researcher and practitioner communities. In information security, role-based access control [7] is an approach for restricting system access to authorized users. For each job function we create a role, then permissions are assigned to specific roles and they carry out the operation. Privacy is a key issue in any organization. An efficient system must be developed to decide how permissions are assigned to data and how the sensitive data are stored and maintained in the organization. Traditional access control methods do not fully meet all the aspects of privacy.

To meet the requirements of privacy the traditional RBAC model is modified and few components like purpose, condition and obligation are added to it and the family of Privacy aware RBAC (P-RBAC) [8] conceptual models is put forward.

Core P-RBAC model [8] forms the base of this entire model. The roles are assigned to the users thus giving them the permission. Conflicts detection is the main property in this model. Hierarchical P-RBAC introduces the three concepts namely Role Hierarchy (RH), Data Hierarchy (DH) and Purpose Hierarchy (PH) which gives a hierarchical structure for Core P-RBAC. Conditional P-RBAC introduces Permission Assignment Sets and it provides a condition language. Universal P-RBAC is a combination of Conditional P-RBAC and Hierarchical PRBAC.

The main components of P-RBAC model are purpose binding, conditions and obligations. Purpose binding means that data collected for one purpose should not used for another purpose, Conditions are the prerequisites that must be satisfied before any permission can be assigned and Obligations are the actions that are to be executed after a permission has been assigned and some action is executed on data objects to

make the action complete. In P-RBAC, privacy policies are expressed as permission assignments. These permissions differ from the traditional RBAC because of the presence of the additional components, representing privacy related information. However during the permission assignment conflicts may occur. We extend the existing Conflict Detection Algorithm [8] which can detect the conflict that arises in only two permission assignments. In this paper we discuss an algorithm that is capable of detecting multiple permission conflicts.

This paper is organized as follows: Related Work is presented in Section 2.Architecture is presented in Section 3. The Modified Algorithm is provided in Section 4. Result and Analysis is presented in Section 5 and the paper is concluded in Section 6.

2. Related Work

Ravi S. Sandhu et.al, [1] introduced a family of reference models for role- based access control (RBAC) in which permissions are associated with roles, and users are made members of appropriate roles. This model greatly simplifies management of permissions. Barth A. et al [2] identified four desirable properties of a privacy policy language namely: Guaranteed consistency, Guaranteed safety, Admitting local reasoning and Closure under combination. Dan Lin et.al, [3] put forward a consistent conditional model as part of core P-RBAC model.

In conditional P-RBAC, they guaranteed that there is no redundancy, indeterminism or Conflict between a new permission and a pre-existing permission assignment set. Elisa Bertino et.al, [4] presented a survey of analysis techniques that have been developed independently for the analysis of security and privacy policies, and they showed how these techniques can be synchronized to analyze the interactions between the policies.

Naikuo Yang et.al, [5] presented the division of purpose into intended purpose and access purpose corresponding to the data access. Each user is required to state his or her access purpose along with the data request. If access purpose is compliant with its intended purpose then access is allowed. Qun Ni et.al, [6] presented a novel obligation model for the Core Privacy-aware Role Based Access Control (P-RBAC). Obligations are characterized as pre-obligations, post-obligations, conditional obligations, and repeating obligations. The interaction between permissions and obligations was inspected.

John Karat et.al, [7] proposed a Policy Framework for Security and Privacy Management. They discussed three-level framework which includes Policy Specification Layer, Abstract Policy Model layer and Executable Policies layer for discussing policy within which security and privacy policy management research can be conducted. Qun Ni et.al, [8] have discussed about the P-RBAC framework covering the aspects of permission assignments and conflict detection algorithms. However, many problems were still left open, like conflict detection between three or more permission assignments. In this paper, we proposed a multiple conflict detection algorithm for providing effective access control.

3. System Architecture

The architecture of the proposed model presented in Fig.1 includes various components namely: User Session, Access Control Layer and Administrator Module. The functionalities of these components are described as follows:

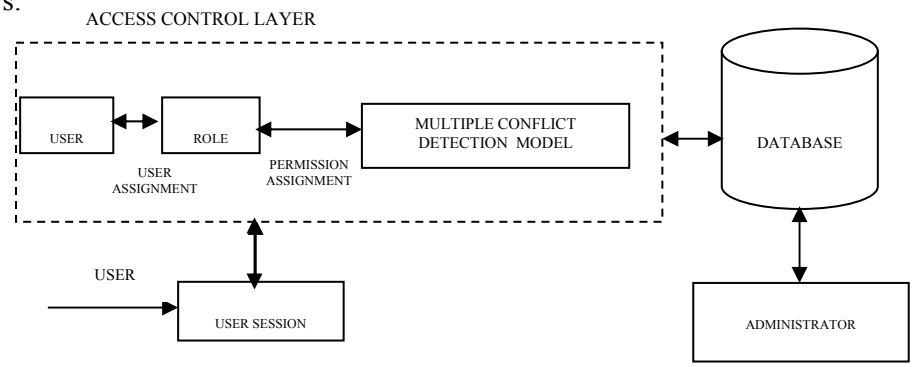


Fig. 1: System Architecture

3.1. User Session

The user could be a new user or an already existing user of the system. They can request for new permission assignments. In this model, users obtain permissions for the roles assigned to them. The permissions includes the Data that the user wants to access along with the Action like read, write and modify which are the operations to be performed on the data specified. This is followed by the three components of Purpose, Obligation and Condition. Once the permission assignment has been made or could not be made, both are intimated back to the user.

3.2. Access Control Layer

The main functionality in the Access Control Layer is Permission Assignment. The assignment is made by checking for conflicts with all the existing permissions already present in the database. The conflict and the redundancy are checked by following these steps.

3.2.1. Multiple Conflict Detection Module

The aim of this module is to detect conflicts in permission assignments, such that at the entry level itself we compare all the existing permission assignments with the new one and detect if there is a conflict. The efficiency of the model is increased by considering conflict resolution for three or more permission assignments. During permission assignment, Redundant permissions are reduced, Obligation Ambiguity is checked and Conflicting permissions assignments are validated.

3.2.1.1. Reducing Redundant Permissions

There exist permission assignments such that there is same role, same data, same action and same purpose. All the conditions that are stated must be satisfied in order to allow the access. So we can combine the permissions to became a single permission by combining the conditions with 'AND' operator.

PA1: (Emp, ((Read, EmailAddr), Advertisment, OP=Yes, NA))

PA2: (Emp, ((Read, EmailAddr), Advertisment, Age=Under13 ^ Parent_conset=Yes, NA).

PA3: (Emp, ((Read, EmailAddr), Advertisment, OP= Yes ^ Age = Under13^ Parent_Conset = Yes, NA)).

By combining the permissions we should not disrupt the meaning of the permission. So we introduce the notion of splitting context variable (SCV) as shown in [8]. Such variables separate the data with which they are associated according to the values they assume. Here, employee's age and salary are SCVs, whereas consent and current time are not.

3.2.1.2. Obligation Ambiguity

An obligation is a responsibility for a subject to do some actions in order to allow certain action to be executed. A typical example is sending a notification to a data owner after each access to his sensitive data. One undesired effect of obligations is ambiguity. Ambiguous obligations are the identical procedure name but different parameters. The discretion of the administrator plays an important role to overcome such ambiguous obligations.

PA4: (Manager, ((Read, Email_Addr), Promo, Age=under13, {Notify (By_Email, Opt-ut}))
PA5: (Manager, ((Read, Email_Addr), Promo, Age=under13, {Notify (By _Email)}))

3.2.1.3. Conflicting Permissions Assignments

If there is a SCV used in the conditions but with different values those permission assignments do not conflict with each other because they actually work on different data.

PA6: (Partner, ((Read, Info), Research, Age=Teenager ^ Time=7PM-10PM, NA))
PA7: (Partner, ((Read, Info), Research, Age=Adult ^ Time=10PM-6AM, NA)).

Conflicts occurs if two permission assignments have compatible conditions i.e., the intersection of the value sets of context variable in different permission assignments is not empty.

3.3. Administrator Module

The Administrator has rights to add new users, roles, purposes, conditions, obligations. He can also assign, modify and delete the permissions, users and roles. So when the session is initiated by the user the request is forwarded and the administrator will process the request and if conflicts exists reports are generated.

4. Modified Multiple Conflict Detection Algorithm

To detect the conflicts that occur in the permission assignments we put forward four algorithms, namely: Condition-Validity-Test, Condition-Conflict-Test, Obligation-Ambiguity-Test, Multiple-PA-Conflict-Detection algorithms [8]. Here we present the modified multiple conflict detection algorithm in Table 1.

Initialization: PA -permission assignment, **CV-** Context variable, **Lcp-** List of conflicting permissions, **Lpa**- List of all the permission assignments already made, **n**- Total number of permissions.

Table. 1: Modified Conflict Detection Algorithm

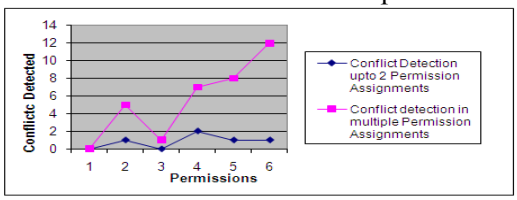
```
1: result←Condition-Validity-Test
        (PA.condition, cv1, cv2,...,cvn)
2: if result = -1 then
3: exit // invalid condition
4: end if
5: for all pa such that pa∈ Lpa do
6: for i = 1 to n do
7: result←Condition-Conflict-Test
             (PA.condition, pa[i], cv[i])
8: if its result is equal to -1 then
9: do begin
10: for j = 1 to n do
11: Lcp.add(pa[j], cv[j])//conflicting
    permission
12: end
13: exit
14: end if
15: end for
```

```
16: for i = 1 to n do
17: result← Obligation-Ambiguity- Test
     (PA.obligation,pa[i].obligation,cv[i].obligatio
     n)
18: if its result is equal to -1 then
19: do begin
20: for j = 1 to n do
21: Lcp.add(pa[j], cv[j])
22: end
23: exit
24: end if
25: end for
26: if PA.purpose to ≠ PA.purpose.intended then
27: Lcp.add(pa[i], result)
28: end if
29: if result equals to 1 then
30: assg.add(PA, CV)
31: end if
32: end for
```

Each of the individual components of PA can be separately accessed as role, data, action, purpose, condition and obligation. The Multiple-PA-Conflict Detection algorithm takes the requested permission as input and divides it into the atomic level. At this level the entries are checked with the already existing values in the previous permission assignments. The Condition-Validity-Test algorithm, Condition-Conflict-Test algorithm, Obligation-Ambiguity-Test algorithms are used to find if a conflict occurs. Further, the purpose is checked to see if it matches the intended purpose of the data. If at any of the stage a conflict occurs they are noted and reported, detailed report indicating where the conflict occurs. By providing such detailed reports the user and the administrator can make use of it to avoid the conflict and revise any of the existing permission assignment.

5. Result and Analysis

Graph 1 shows that the modified multiple conflict detection algorithm has the capability to check the conflicts that occur in the three components namely purpose, obligation and condition. The Conflict Detection Algorithm [8] can spot conflict only between two permissions. But the modified algorithm overcomes this and can detect conflicts in two or more permission assignments.



Graph 1: Conflicts detection accuracy analysis

6. Conclusion

In this paper we have discussed a privacy aware multiple conflict detection algorithm using n level conflicts, where the conflicts could occur in the system. The existing Conflict Detection Algorithm [8] has been modified and is extended to identify conflicts that might occur between multiple permission assignments. Moreover, an effective detection mechanism that identifies the conflict in the first stage itself so that all the sensitive data are well protected in the system is also considered in this work. Therefore, this model considers the features like purpose, condition and obligation so that the system is ensures that the privacy related information are properly expressed and conflicts are handled well.

7. References

[1] Ravi S. Sandhu, Edward J. Coynek, Hal L. Feinsteink and Charles E. Youmank. "Role-Based Access Control Models". IEEE Computer, Volume 29, Number 2, pages 38-47, 1996.

[2] Barth, A., Mitchell, J.C., and Rosenstein, J. "Conflict and combination in privacy policy languages". In WPES'04: Proceedings of the 2004 ACM workshop on Privacy in the Electronic society. ACM Press, USA, 45–46, 2004.

[3] Dan Lin, Qun Ni, Elisa Bertino and Jorge Lobo. "Conditional Privacy aware Role Based Access Control". In ESORICS '07: the Proceedings of the 12th European Symposium On Research In Computer Security, LNCS 4734, pp. 72 - 89, 2007, Dresden, Germany.

[4] Elisa Bertino, Carolyn Brodie, Seraphin Calo, Lorrie Cranor, Clare-Marie Karat, John Karat, Ninghui Li, Dan Lin, Jorge Lobo, Qun Ni, Prathima Rao and Xiping Wang. "Analysis of Privacy and Security Policies". IBM Journal of Research and Development, Volume 53, Number 2, 2009.

[5] Naikuo Yang, Barringer H., Ning Zhang. "A Purpose – Based Access Control Model". Journal of Information Assurance and Security 51-58. 2008

[6] Qun Ni, Elisa Bertino and Jorge Lobo. "An Obligation Model Bridging Access Control Policies and Privacy Policies", In SACMAT '08, the Proceedings of the 13th ACM symposium on Access control models and technologies, Estes Park, CO 80517, USA, 2008.

[7] John Karat, Clare-Marie Karat, Elisa Bertino, Ninghui Li, Qun Ni, Carolyn Brodie, Jorge Lobo, Seraphin Calo, Lorrie Cranor, Ponnurangam Kumaraguru and Robert Reeder. "A Policy Framework for Security and Privacy Management " IBM Journal of Research and Development Volume 53 Issue:2 page : 4:1 - 4:14. 2009

[8] Qun Ni, Elisa Bertino, Carolyn Brodie, Clare-Marie Karat, John Karat, Jorge Lobo and Alberto Trombetta. "Privacy aware Role Based Access Control", ACM Transactions on Information and and System Security (TISSEC). Volume 13 Issue 3, July 2010.