POLICY INCONSISTENCY DETECTION BASED ON RBAC MODEL IN CROSS-ORGANIZATIONAL COLLABORATION

Poh Kuang Teo[1], Hamidah Ibrahim [2], Fatimah Sidi [3] and Nur Izura udzir [4]

[1]*Universiti Putra Malaysia, Malaysia, pohkuang1985@yahoo.com.my*
[2]*Universiti Putra Malaysia, Malaysia, hamidah@fsktm.upm.edu.my*
[3]*Universiti Putra Malaysia, Malaysia, fatimahcd@fsktm.upm.edu.my*
[4]*Universiti Putra Malaysia, Malaysia, izura@fsktm.upm.edu.my*

ABSTRACT. Policy integration and conflict resolutions among various organizations still remain a major challenge. Moreover, policy inconsistency detection approach with logical reasoning techniques which considers integration requirements from collaboration parties has not been well studied. In this paper, we proposed a model to detect inconsistencies based on role-based access control (RBAC) that considers role hierarchy (RH) and temporal and spatial constraints. A model to prune and collect only the required policies based on access control requirements from different organizations is designed. Policy inconsistency detection should be enhanced with logical-based analysis in order to develop security policy integration. We believe this work could provide manner to filter a large amount of unrelated policies and only return potential collaboration policies for conflict resolution.

Keywords: policy inconsistency detection, collaborative environment, role-based access control (RBAC), role hierarchy, constraints

INTRODUCTION

Nowadays there are increasing needs for sharing data that contain personal information among various cross-organizational collaborations. Thus, there is a need for a dynamic architecture in order to share data among different cross-organizations in collaborative environments since each organization may join or leave at runtime. However, often such data sharing may contain personal sensitive and confidential information, such as family composition and DNA. It still remains a major challenge to ensure security issues for such data sharing in collaborative environment.

Security concerns with confidentiality, integrity and availability of data. Thus, cross-organizational data sharing should focus on the security access control to avoid sensitive data from being accessed by unauthorized users. Collaboration between organizations for the purpose of data sharing will involve security policy integration since each organization may independently specify its own security policies based on its own interest. Policy integration is a process to integrate security policies from the participating organizations in order to govern the data sharing throughout the collaborations. Our access control policies are mainly designed for role hierarchies and context constraints in RBAC model.

During the policy integration phase, the policies to collaborate from different organizations are compared and evaluated through logical-based analysis. Various inconsistencies between access policies from different distributed units may occur when elements conflict with each other between different policies from collaborating parties. Thus,

inconsistency detection is not only important to achieve a conflict-free collaboration environment but also the availability, confidentiality and integrity in security issues.

However, the policy inconsistency detection approach with semantic reasoning techniques which considers integration requirements from collaboration parties has not been well studied. In this paper, we proposed a model to detect inconsistencies based on the role-based access control (RBAC) model that considers role hierarchy (RH) and temporal and spatial constraints. A model to prune and collect only the required policies based on access control requirements from different organizations is designed. Policy inconsistency detection should be enhanced with logical reasoning techniques in order to develop security policy integration. We believe this work could provide a manner to filter a large amount of unrelated policies and only return potential collaboration policies for conflict resolutions.

The paper is organized as follow. The literature review is discussed in the following section. This is followed by the presentation of our proposed model for policy inconsistency detection with logic-based analysis and classification of inconsistencies. The last section will conclude our work.

LITERATURE REVIEW

There are a few previous studies that use Description Logic (DL) reasoning to prove that two policies are suitable, or not suitable, for collaboration purposes (He & Yang, 2009; Huang, Huang, & Liu, 2009). DL that is encoded in these studies can be used to determine the satisfiability of a concept. A study in He & Yang, 2009 concentrated on the analysis to show that different types of collaboration impose different ways of integration. This proposed model has limitation as only some of the policy inconsistencies have been encoded in the authorization RBAC policy model which is limited in certain case studies. Three types of inconsistencies are identified in this work; role inconsistencies, credential inconsistencies, and privilege inconsistencies. Huang, Huang, & Liu, 2009 work extended the existing eXtensible Markup Language (XACML) architecture which can support policy conflict detection based on their proposed DL method, but this study will always omit the attributes of condition. DL is a famous knowledge representation because it can express concept and relationship between concepts. However, this satisfiability (SAT) solver based analysis is unable to present an integrated view of relationships among policies. Besides that, these works do not consider context constraints and role hierarchy.

Previous studies worked with similarity measurement to calculate the similarity scores between policies (Lin, Rao, Bertino, & Lobo, 2007; Yau & Chen, 2008). However, this work needs to be incorporated with other analysis tools in order to improve the result of conflict detection. The higher the score the more similar the two policies are. Otherwise, the request to access data is rejected by the data owner. The determination of similarity between structures in policies is computationally cheaper but less precise technique compared to matching them in full details based on logical reasoning or model-based checking (Bertino, Brodie, & Calo, 2010). The main drawback with this similarity score computation is that user intervention is required to prior match knowledge or tune parameters such as providing additional manual work to assign the weight value for emphasizing the importance of the target or condition similarity respectively. In summary from the above studies, the data owner always has the priority to do the access decision to maintain its data in the resource owner on which policy is most similar to data owner. However, each organization should have the right to claim ownership of the data. Thus, it is unreasonable to give the priority to the data owner to decide the chance for collaboration between organizations.

Mazzoleni, Crispo, Bertino, & Sivasubramanian, 2008 only discussed the semantic relationships on pair wise constraint elements among policies which always assume corresponding elements between policies are the same in value with each other. Oliva &

Saltor, 2000 presented a mandatory access control policy to endow tightly coupled federated database systems with a multilevel security system. However, this work is limited to specify the relationships between subject elements while the others elements such as constraints are always omitted in detection phases. Huang, Sun, Wang, & Si, 2009 identified the types of redundancies and inconsistencies during the policy redundancy and inconsistency checking. However, this work is limited to presenting for the inconsistency of intra-organization. We believe this method cannot scale well with a larger set of policies analysis among collaboration organizations.

Park & Lee 2007 presents an access control mechanism using temporal and spatial context information to patient information. Temporal context information classifies time into two types - doctor's regular working time and other time while spatial context information classifies location into three spaces; inner medical office, outer medical office in hospitals, and other places. In the RBAC policy application environment, context constraints play a critical role in order to maintain database security (Kumar & Newman, 2009). Thus, our work considers not only the temporal constraints, but also the spatial constraints in policy inconsistency detection.

Our study discussed the RBAC issues under context constraints and role hierarchy in a collaboration environment to further guarantee consistency policy integration in a collaboration environment. It is necessary for us to carry out a larger, yet feasible implementation that will provide the scenario required for more comprehensive policy integration.

THE PROPOSED MODEL

The Security Policy (SP) in our study is defined as follows:

SP= (*R*, *PM*, *C, E*), where *R* = Role (e.g., Physician), *PM* = Permission and is defined as a pair <*RE*, *A*>, where *RE* is Resource (e.g., patient information) and *A* is Action of data (e.g., read), *C* = Constraint, and E = Effect (e.g., permit or deny). Constraint information that is included in the policy is temporal and spatial contexts.

Each organization may specify its own security policies independently. Thus, our proposed model aims at filtering the unrelated policies and analyzing the types of inconsistencies which may occur among policies from different organizations for collaborations. The following describes our proposed model which consists of two phases, namely: policy pruning and policy inconsistency detection. Figure 1 shows our policy inconsistency detection model for security policy integration process.

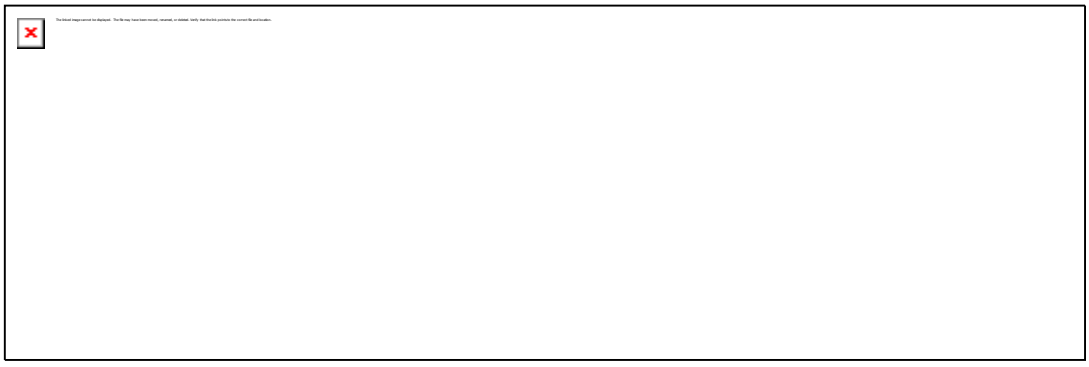

Figure 28. The Policy Inconsistency Detection Model.

Now let us explain how the policy inconsistency detection model can detect inconsistencies which always exist between collaborative organizations. For example, let us

consider three different hospitals which intend to collaborate with each other; hospital A with policies A_1, A_2, and A_3, hospital B with policies B_1, B_2, and B_3, and hospital C with policies C_1 and C_2 as specified below.

Organization A :

A_1: (Surgeon, Therapy Treatment $\vee$ Patient Personal Data, Read, 09:00 $\leq$ Time $\leq$ 17:00 $\wedge$ Inner Hospital, Permit)

A_2: (Specialist Physician, Treatment History, Update, Outer Office, Deny)

A_3: (Mental Nurse, Bone X-Ray $\vee$ Patient Personal Data, Read, Inner Hospital, Deny)

Organization B :

B_1: (Nurse, Bone X-Ray, Read, Inner Hospital, Permit)

B_2: (Specialist Physician, Therapy Treatment $\vee$ Patient Personal Data, Read $\vee$ Write, Outer Office, Permit)

B_3: (Laboratory Scientist, Laboratory Sample, Read $\vee$ Write $\vee$ Update, Inner Lab, Permit)

Organization C :

C_1: (Specialist Physician, Therapy Treatment, Read $\vee$ Write, (09:00 $\leq$ Time $\leq$ 12:30 $\vee$ 13.30 $\leq$ Time $\leq$ 18:00) $\wedge$ Inner Office, Permit)

C_2: (Laboratory Scientists, Blood Sample, Read $\vee$ Write $\vee$ Update $\vee$ Delete, Inner Lab, Permit)

Policy Pruning Phase

The policy pruning phase filters the policies from those organizations that are unrelated before the organizations engage in collaboration. If a pair wise policy similar with each others, then these two policies are the potential candidates for the checking in the next phase, which is the policy inconsistency detection phase, before a common set of integrated policies are generated. Assume that, $P_a = \{R_a, RE_a, A_a, C_a, E_a\}$ and $P_b = \{R_b, RE_b, A_b, C_b, E_b\}$. Five cases are identified as follows which show how the policy pruning phase works in our model.

$$R_a \neq R_b \tag{5}$$

For example, policy A_1 and policy B_1 will not have further checking since the analysis shows that these two policies are different based on the role element. Thus, referring to the case above, policy A_1 and B_1 will be pruned out in this phase.

$$if\,(R_a \leq R_b) \wedge (RE_a \cap RE_b = \square) \tag{2}$$

For example, policy B_3 specified "Laboratory Scientist" which is the junior role of "Specialist Physician" in policy C_1. However, the set of resources specified in B_3 is disjoint with the set of resources specified in policy C_1. Thus, referring to the case above, policy B_3 and C_1 will be pruned out in this phase.

$$if\,(R_a \leq R_b) \wedge [(RE_a \subseteq RE_b) \wedge (A_a \cap A_b) = \square] \tag{3}$$

For example policy A_2 specified "Specialist Physician" which is the same role specified in policy B_2. The resource, "Treatment History" which is specified in A_2 is a superset of "Therapy Treatment" which is specified in B_2. However, the set of actions specified in A_2 is disjoint with the set of actions specified in policy B_2. Thus, referring to the case above, policy A_2 and B_2 will be pruned out in this phase.

$$if\,(R_a \le R_b) \wedge [(RE_a \subseteq RE_b) \wedge (A_a \subseteq A_b)] \tag{4}$$

For example, policy A_1 specified "Surgeon" role which is the junior role in policy B_2 since this policy specified "Specialist Physician" in the role element. "Therapy Treatment" and "Patient Personal Data" in policy A_1 is the same as the resource elements in policy B_2 while the action element of policy A_1 is a subset of policy B_2. This case satisfies case *4*, policies A_1 and B_2 are considered as potential pair wise candidates for collaboration which will be submitted to the policy inconsistency detection phase for further checking.

$$\begin{aligned} &if\,(R_a \le R_b) \wedge [(RE_a \supseteq RE_b) \wedge (A_a \supseteq A_b)] \vee [(RE_a \subseteq RE_b) \wedge (A_a \supseteq A_b)] \\ &\vee [(RE_a \supseteq RE_b) \wedge (A_a \subseteq A_b)] \end{aligned} \tag{5}$$

For example, policy A_3 specified the "Mental Nurse" role, which is the junior role of "Nurse" in policy B_1. However, "Bone X-Ray" and "Patient Personal Data" in resource element of policy A_3 is a superset of "Bone X-Ray" in policy B_1. The action element of policy A_3 is the same as the policy B_1. Thus, the analysis between policies A_3 and B_1 is identified as role hierarchy inconsistencies before these two policies submit to the policy inconsistency detection phase for further checking.

It is not recommended to generate a set of policies shared by collaboration organizations based only on this phase. Those policies which are identified as potential collaborate candidates in this phase are submitted to the next phase, the policy inconsistency detection phase, for further details checking.

Policy Inconsistency Detection Phase

After the pruning phase is performed the unrelated policies are filtered while the related policies are further analysed. In this phase, the condition and the effect of the elements are checked. Two cases are identified to indicate that two policies are similar as follows:

$$if\;\; E_a \equiv E_b \wedge [(C_a \subseteq C_b) \vee (C_a \supseteq C_b) \vee (C_a \cap C_b) \vee (C_a \not\subset C_b)] \tag{6}$$

For example, policies A_1 and B_2 are considered similar to each other after the pruning phase. We found that policy A_1 is more restricted than policy B_2 with the respect of temporal and spatial conditions. In this case, it is infeasible to make decision without analysing the contents of the conditions in details since the further negotiation is needed in the inconsistency resolution. We classified these possible pair wise policies as constraints inconsistencies.

$$if\;\; E_a \neq E_b \wedge [(C_a \subseteq C_b) \vee (C_a \supseteq C_b) \vee (C_a \cap C_b)] \tag{7}$$

For example, policies A_3 and B_1 are considered similar to each other after the pruning phase. In the condition element, we found that the spatial information identified in policy A_3 is "Inner Hospital" with denied access decision which is the same as the spatial condition element in policy B_1 with permit access decision. We cannot simply make decision without analysing the details in this kind of cases. Thus, we classified it as authorization inconsistencies where further negotiation is needed in the inconsistency resolution.

CONCLUSION

To briefly conclude, our work is to present a policy inconsistency detection model, based on a role-based access control that considers both the role hierarchy and the temporal and spatial constraints in policy inconsistency detection. The policy inconsistency detection model presented is to identify various types of inconsistencies among policies. The policy

inconsistency detection solution is not limited to inter-organizational collaboration in the Healthcare domain, but is also applicable to other domains as well. The proposed policy inconsistency detection model can be considered as a generic model. We believe this work could provide a manner to filter a large amount of unrelated policies and only return potential collaboration policies for conflict resolutions. Thus, in the future we intend to extend this study to investigate the policy inconsistency resolution for each type of inconsistencies.

REFERENCES

Bertino, E., Brodie, C., & Calo, S. (2010). Analysis of Privacy and Security Policies. *Journal of Research and Development, IBM Corp. Riverton, 53(2)*, 1-18. doi: 10.1147/JRD.2009.5429045

He, D. D. & Yang, J. (2009). Authorization Control in Collaborative Healthcare Systems. *Journal of Theoretical and Applied Electronic Commerce Research, 4(2)*, 88-109. doi: 10.4067/S0718-18762009000200008

Huang, C., Sun, J., Wang, X., & Si, Y. (2009). Security Policy Management for Systems employing Role Based Access Control Model. *Journal of Information Technology*, *Asian Network for Scientific Information, Pakistan, 8(5)*, 726-734. doi: 10.3923/itj.2009.726.734

Huang, F., Huang, Z., & Liu, L. (2009). A DL-based Method for Access Control Policy Conflict Detecting. *Proceedings of the First Asia-Pacific Symposium on Internetware.* doi: 10.1145/1640206.1640222

Kumar, M. & Newman, R.E. (2010). STRBAC - An Approach Towards Spatio-Temporal Role-based Access Control. *Proceedings of the 3rd IASTED International Conference on Communication*, *Network*, *and Information Security*, 150-155.

Lin, D., Rao, P., Bertino, E., & Lobo, J. (2007). An Approach to Evaluate Policy Similarity. *Proceedings of the 12th ACM Symposium on Access Control Models and Technologies*, 1-10. doi:10.1.1.89.2793

Mazzoleni, P., Crispo, B., Bertino, E., & Sivasubramanian, S. (2008). XACML Policy Integration Algorithms. *Journal ACM Transaction on Information and System Security* (*TISSEC*), *ACM New York, 11(1)*. doi: 10.1145/1330295.1330299.

Oliva, M. & Saltor, F. (2001). Integrating Multilevel Security Policies in Multilevel Federated Database Systems. *Proceedings of the 14th IFIP 11.3 Working Conference in Database and Applications Security*, 193-208. doi:10.1.1.76.2739.

Park, J. H. & Lee, D. G. (2007). PIS-CC RBAC: Patient Information Service based on CC_RBAC in Next Generation Hospital Considering Ubiquitous Intelligent Environment. *Proceedings of the International Conference on Multimedia and Ubiquitous Engineering* (*MUE'07*), *IEEE CS*, 196-200. doi: http://doi.ieeecomputersociety.org/10.1109/MUE.2007.171

Yau, S.S. & Chen, Z. (2008). Security Policy Integration and Conflict Reconciliation for Collaboration among Organizations in Ubiquitous Computing Environments. *Lecture Notes in Computer Science*: *Vol.* 5061/2008. *Ubiquitous Intelligence and Computing* (pp. 3-19). Berlin, Germany: Springer-Verlag. doi: 10.1007/978-3-540-69293-5_3