ASSESSING ROLE MINING ALGORITHMS USING RMINER TOOL IN IDENTIFYING THE BEST MODEL FOR ACCESS CONTROL

[1,3]NAZIRAH ABD HAMID, [2]ALA ALAROOD, [2]AZIZAH ABDUL MANAF, [1]RABIAH AHMAD

[1]Faculty of Information & Communication Technology, Universiti Teknikal Malaysia Melaka (UTeM), Melaka, Malaysia

[2]Faculty of Computing and Information Technology, University of Jeddah (UJ), Saudi Arabia

[3]Faculty of Informatics and Computing, Universiti Sultan Zainal Abidin (UniSZA), Besut, Terengganu, Malaysia

E-mail: [1,3]owenira@gmail.com, [2]alaa.alarood@gmail.com, [2]azizahmanaf18@gmail.com, [1]rabiah@utem.edu.my

ABSTRACT

Role-based access control model has been accepted by many organizations because of the security feature that the model offers. Generally, role-based access control executes based on a role which includes set of permissions that built in a matrix form and the advantage of this model is the flexibility to modify or reassign the roles of a user if the permissions are changed. However, one of the main challenges in designing and executing the role-based access control model is to define a complete and correct set of roles and assigning them with permissions and therefore to solve this problem, the researchers have proposed role mining algorithms to produce such role set. Moreover, there are only a few comprehensive studies have been carried out to compare these algorithms and the evaluations are usually using different dataset and evaluation criteria. In order to give the researcher some insights into their performances, this paper presents an analysis on some of the well-known role mining algorithms by using RMiner tool that can be applied for role mining process and deliberate some of the dataset that can be used in role mining environment.

Keywords: *Access Control, RBAC, Review, Role Mining, Trends*

1. INTRODUCTION

An access control model (ACM) is one of the method that can be applied for ensuring data security. Traditional ACM for example, Mandatory Access Control (MAC) and Discretionary Access Control (DAC) may not be adequate to support a large number of resources and users such as in cloud computing environment [1]. Thus, Role Based Access Control (RBAC) is one of the recent ACM that could accommodate the dynamically structure of cloud computing.

These studies by [2] and similarly by [3] have revealed that RBAC advantages include the hierarchy element that make an access control model easy to manage and versatile especially if the organization involve a lot of users. Apart from that, RBAC also can manage the sensitive information from leaking out by implementing least privilege and separation of roles elements. These features enable to make RBAC more powerful, robust and secure model.

Nonetheless, the management of role can be a big issue in RBAC. Occasionally it is difficult to identify which permissions that related to the user which has been associated with a particular role. Job roles are assigned based on the least privilege but still if some modification has been done to the role, a user might have some misunderstanding when considering the permissions of each user associated with that role [4].

This role management process can be identified as role engineering and can be categorized into top-down approach and bottom-up approach. According to [5] the top-down approach required the interference from human expert to understand the business processes and extract relevant roles from the analysis. In contrast, the aforementioned author explains that the bottom-up approach or often called as role mining exploit the existing user-permission assignment (UPA) to define roles. Role mining has the advantage of being automated by applying certain algorithms as has been discussed by many authors such as [6], [7] and [8].

However, there are only a few comprehensive study have been done to compare these algorithms and usually these role mining algorithms have been evaluated using different dataset and evaluation measurement. Additionally, to effectively implementing the role mining process, an efficient role mining tool that can deliver cost-effective assistance is needed in delivering a comprehensive RBAC system [9].

The objective of this paper is to present an analysis on some distinguished role mining algorithms by using RMiner tool that can be applied for role mining process and deliberate some of the dataset that can be used in role mining environment and these experiments were conducted on the running platform of OS X El Capitan (Version 10.11.6).

The remainder of the paper is structured as follows. Section 2 presents a background study while Section 3 discusses RMiner tool. The methodology is shown in Section 4 while Section 5 gives the evaluation results and lastly Section 6 concludes the article.

2. RELATED WORKS

According to [7], the difficulty of determining an optimal set of roles from the UPA is denoted as the Role Mining Problem (RMP) and the objective of RMP is to minimize either the number of roles, the total of UA and PA and other metrics.

Minimizing the number of roles has been discussed by many authors and in this paper, the authors have discussed genetic algorithm as a method to find minimum chromatic number from a set of graphs to deploy certain constraints. The experiments have been conducted using generated or random dataset to evaluate the min user count and the running time [10]. Furthermore, the authors have examined the greedy algorithms to resolve the problem of large candidate roles pool and the established dataset have been utilized to calculate three different metrices namely executed time, similarity and size [11].

Moreover some authors have introduced a task-constrained RBAC model which comprises of four different constraints that could restrain the permission that can flexibly control the permission and role activations that finally could subsequently find the minimal number of permissions. inheritance and role activations [12]. Additionally Pan et al. have suggested three pointers to express the problem of hierarchy in RBAC reconfiguration with minimal weight structure complexity (WSC) and perturbation [13].

3. RMINER TOOL

According to [14] RMiner is a tool that offers assistant to researchers or security administrators to perform role engineering activities. This tool provides a platform to do comparison experiments on the chosen algorithms. RMiner is designed and developed in the Java-based platform on WEKA environment. Generally, RMiner begins with pre-processing stage that specifically works to remove any noises and the clean data acts as the input data to the chosen role mining algorithm and lastly the output of the role mining can be epitomized in graphically manner.

This tool delivers a user-friendly interface that can be utilized to implement role mining algorithms such as CompleteMiner and FastMiner [15], HierarchicalMiner [16], ORCA [17], Graph Optimisation [18], HP Edge Minimization and HP Role Minimization [19] and lastly FeatureMiner and CCRMimplement [20] and each algorithm could produce output as in Table 1 [14].

4. METHODOLOGY

The general framework of RMiner can be described in Figure 1 that includes main stages namely, data pre-processing, role mining and role assignment [14]. This framework is quite similar with role mining model that has been designed by [21]. The RMiner stages are described in the next sub-section.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 1: Role Mining Algorithms

Role Mining Algorithm	Criteria
CompleteMiner (CM)	Can output role set
FastMiner (FM)	
HP Role Minimization (HPRM)	
ORCA	Can output role set and role hierarchies
Graph Optimisation (GO)	
HierarchicalMiner (HM)	
HP Edge Minimization (HPEM)	
FeatureMiner (FeM)	
CCRMimplement (CCRM)	Consider constraints

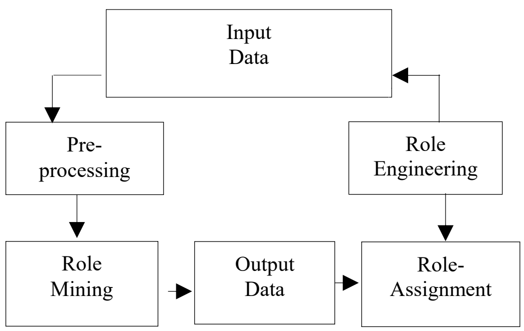

Figure 1: RMiner Framework

4.1 Input Data

According to [21], in most of the role mining algorithms only user-permission assignment (UPA) matrix can be considered as input data and this is consistent with [14] which has shown that their tool can only consider UPA as the input data. Accordingly, an active research should be done to explore the possibility of other types of data to be used in role mining process [21].

4.2 Pre-processing

Many researchers have emphasized on the importance of this stage particularly to generate a clean and quality data. In RMiner, pre-processing stage involve the process to clean the noises that might affect the results. The data could be selected between predefined real data sets and an artificial data generation tool however for RMiner, this stage should be performed manually [14].

4.3 Role Mining

In this stage, the user could use the clean dataset as the input for the role mining algorithms. In RMiner, the users could determine to use their own algorithm or to select from 13 existing algorithms. In this process, firstly, the users could elect the desired role mining algorithm, configure the algorithm parameters, and activate the algorithm. Then, RMiner would execute the selected algorithm by recording the running time, memory space usage and other evaluation information. Eight algorithms have been proposed by other researchers while Feature Miner, Weighted Role Mining, CCRM implement, Thesis Algorithm Implement and YeHRMiner have been designed by the authors [14].

4.4 Role Assignment

Role assignment process in RMiner offers control to the users to manually edit and update the role state in RMiner so that the interpretability and generalization ability of the output meet the security standard. This menu indirectly able to provide knowledge on the role hierarchy and the weights of roles to the users. After the updating activity, the users then could assign the roles to users according to the original UPA and according to the authors this procedure is known as a set covering problem. As not many research have been done on this subject, RMiner only recommends greedy algorithm to solve the role assignment problem [14].

5. EVALUTION RESULTS

5.1 Discussion

While many role mining algorithms have been designed and developed throughout the years as has been investigated in Section 2, there only a handful of discussion have be done to compare these algorithms. These role-mining algorithms have been evaluated when they were proposed, but the evaluations were using different datasets and evaluation criteria. So this paper offers some analysis on some classical role mining algorithms by using RMiner tool and these experiments were conducted on the running platform of OS X El Capitan.

In this section, we test some existing algorithms through some real-world datasets as shown in Table 2 and the datasets comprise of the numbers of roles |R| and permissions |PERMS|. The apj and emea dataset have been obtained from Cisco firewall of Hewlett Packard (HP) networks

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[19]. The university dataset has been generated from Stony Brook University [22] while firewall1 and firewall2 dataset have been gained as the results from running an analysis algorithm on Checkpoint firewalls and available at http://www.hpl.hp.com/personal/Robert_Schreiber/. Lastly the healthcare dataset has been collected from the US Veteran's Administration [7]. We implement these experiments on 2.7 GHz Intel Core i5 with 8 GB 1867 MHz DDR3 to evaluate their performance. The running platform is OS X El Capitan (Version 10.11.6).

According to [7], the difficulty of determining an optimal set of roles from the UPA is denoted as the Role Mining Problem (RMP) and the objectives of RMP are to minimize either the number of roles, the total of UA and PA and other metrics. Until now, there is no common criterion in the literature on the goodness of a role set but many researchers have agreed that the goodness of identified roles could be measured by the degree of similarity with an existing RBAC configuration, or the ability to correctly describe the input UPA.

5.2 Analysis

In this paper, we divided the results of the experiments into three different categories namely number of roles, processing time and accuracy as displayed from Table 3 to Table 8. In Table 3 and Table 4 we have shown the number of roles that are generated by nine different algorithms by using RMiner tool from six different datasets. The figures show that averagely HPRM has performed the best for minimizing number of roles throughout the datasets whereas CM has performed the worst, confirming the results from [23] with all data could be computed. Our results indicate that algorithms with smaller dataset are likely to produce smaller number of roles.

For Table 5 and Table 6, on average CCRM has showed less time to compute than other algorithms throughout the datasets with all data could be computed and lastly Table 7 and Table 8 has presented the results of the accuracy of generated roles with original roles. It has illustrated that CM, FM and GO is capable of appropriately describing the original input UPA. The accuracy could be defined as the ratio of the number of generated roles to the number of original role sets.

$$\text{Accuracy} = \text{no.of generated roles} / \text{no.of original roles} \quad (1)$$

Table 2: Real-world Datasets

Dataset	|USERS|	|PERMS|
apj	2044	1164
university	493	56
emea	35	3046
firewall1	365	709
firewall2	325	590
healthcare	46	46

Table 3: Number of Roles

Role Mining Algorithm	apj	university	Emea
CM	796 roles	31 roles	778 roles
FM	781 roles	31 roles	242 roles
HPRM	452 roles	17 roles	34 roles
ORCA	2014 roles	105 roles	x (could not compute)
GO	452 roles	23 roles	48 roles
HM	x (could not compute)	21 roles	x (could not compute)
HPEM	453 roles	18 roles	34 roles
FeM	611 roles	22 roles	x (could not compute)
CCRM	456 roles	18 roles	35 roles

Table 4: Number of Roles

Role Mining Algorithm	firewall1	firewall2	healthcare
CM	315 roles	21 roles	30 roles
FM	266 roles	20 roles	29 roles
HPRM	60 roles	10 roles	13 roles
ORCA	1415 roles	1179 roles	91 roles
GO	92 roles	11 roles	18 roles
HM	85 roles	12 roles	15 roles
HPEM	130 roles	12 roles	17 roles
FeM	96 roles	14 roles	20 roles
CCRM	69 roles	11 roles	15 roles

Table 5: Elapsed Time

Role Mining Algorithm	apj	university	Emea
CM	12740.0 ms	78.0 ms	1276.0 ms
FM	13173.0 ms	60.0 ms	288.0 ms
HPRM	18899.0 ms	202.0 ms	429.0 ms
ORCA	515864.0 ms	66.0 ms	x (could not

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

			compute)
GO	8658.0 ms	202.0 ms	2419.0 ms
HM	x (could not compute)	169.0 ms	x (could not compute)
HPEM	22114.0 ms	183.0 ms	571.0 ms
FeM	52463.0 ms	77.0 ms	x (could not compute)
CCRM	517.0 ms	23.0 ms	380.0 ms

Table 6: Elapsed Time

Role Mining Algorithm	firewall1	firewall2	healthcare
CM	1221.0 ms	137.0 ms	57.0 ms
FM	788.0 ms	106.0 ms	34.0 ms
HPRM	1305.0 ms	852.0 ms	42.0 ms
ORCA	26346.0 ms	32969.0 ms	38.0 ms
GO	2662.0 ms	2907.0 ms	75.0 ms
HM	32647.0 ms	287.0 ms	120.0 ms
HPEM	11708.0 ms	896.0 ms	37.0 ms
FeM	29428.0	136.0 ms	67.0 ms
CCRM	192.0 ms	57.0 ms	11.0 ms

Table 7: Accuracy

Role Mining Algorithm	apj	university	Emea
CM	1.0	1.0	1.0
FM	1.0	1.0	1.0
HPRM	0.932	0.838	1.0
ORCA	0.834	0.513	x (could not compute)
GO	1.0	1.0	1.0
HM	x (could not compute)	0.164	x (could not compute)
HPEM	0.888	0.821	1.0
FeM	0.601	0.0	x (could not compute)
CCRM	0.655	0.041	1.0

Table 8: Accuracy

Role Mining Algorithm	firewall1	firewall2	healthcare
CM	1.0	1.0	1.0
FM	1.0	1.0	1.0
HPRM	0.488	0.942	0.804
ORCA	0.542	0.926	0.5435
GO	1.0	1.0	1.0
HM	0.099	0.689	0.130
HPEM	0.282	0.754	0.326
FeM	0.068	0.006	0.130
CCRM	0.128	0.689	0.130

6. CONCLUSION

The objective of this paper is to present an analysis on some established role mining algorithms by using RMiner tool from three different perspectives namely number of roles, processing time and accuracy and discuss some of the dataset that can be applied for role mining environment and these experiments were conducted on the running platform of OS X El Capitan (Version 10.11.6).

In conclusion, HPRM has performed the best for minimizing number of roles throughout the datasets with all data could be computed but with elapsed time that quite high while the accuracy could be varies among the datasets.

This tool offers some interesting features such as it has user-friendly interface and could provide visualization tool to view the generated roles set easily. Nevertheless, some of the disadvantage of this tool includes, most of the large dataset could not be processed and the data clean up and redundancy process must be prepared manually.

For the future works, there is still a few interesting research that could be extended such as handling data other than UPA and role assignment methods.

ACKNOWLEDGEMENT:

This research is fully sponsored by Centre for Research and Innovation Management (CRIM), Universiti Teknikal Malaysia Melaka (UteM) and the Ministry of Education via TRGS Project coded as TRGS/1/2016/FTMK-CACT/01/D00006.

REFRENCES:

[1] N. Meghanathan, "Review of Access Control Models for Cloud Computing," *Comput. Sci. Inf. Technol. (CS IT)*, vol. 3, no. 1, pp. 77–85, Sep. 2013.

[2] M. Auxilia and K. Raja, "Dynamic Access Control Model for Cloud Computing," in *Sixth International Conference on Advanced Computing (ICoAC)*, 2014, pp. 47–56.

[3] B. S. Sunitha and A. B. Basu, "Review of Role Based Access Control Method for Securing User Space in Cloud Computing," *Int. J. Comput. Trends Technol.*, vol. 14, no. 1, pp. 22–25, 2014.

[4] Sourab, "Review Paper of Role Based Access Control Over Cloud Computing," *Int. J. Adv. Sci. Technol.*, vol. 2, no. I, pp. 31–34, 2014.

[5] J. H. Jafarian, H. Takabi, H. Touati, E. Hesamifard, and M. Shehab, "Towards a General Framework for Optimal Role Mining: A Constraint Satisfaction Approach," in *Proceedings of the 20th ACM Symposium on Access Control Models and Technologies - SACMAT '15*, 2015, pp. 211–220.

[6] M. Kunz, E. Fuchs, M. Netter, and G. Pernul, "How to Discover High-Quality Roles? A Survey and Dependency Analysis of Quality Criteria in Role Mining," in *Communications in Computer and Information Science*, vol. 576, O. Camp, E. Weippl, C. Bidan, and E. Aïmeur, Eds. Cham: Springer International Publishing, 2015, pp. 49–67.

[7] B. Mitra, S. Sural, J. Vaidya, and V. Atluri, "A Survey of Role Mining," *ACM Comput. Surv.*, vol. 48, no. 4, pp. 1–37, 2016.

[8] N. A. Hamid, R. Ahmad, and S. S. Rahayu, "Recent Trends in Role Mining Algorithms for Role-Based Access Control: A Systematic Review," *World Appl. Sci. J.*, vol. 35, no. 7, pp. 1054–1058, 2017.

[9] I. Molloy, H. Chen, T. Li, Q. Wang, N. Li, E. Bertino, S. Calo, and J. Lobo, "Mining Roles with Multiple Objectives," *ACM Trans. Inf. Syst. Secur.*, vol. 13, no. 4, pp. 1–35, Dec. 2010.

[10] A. Roy, S. Sural, and A. K. Majumdar, "Minimum User Requirement in Role Based Access Control with Separation of Duty Constraints," in *2012 12th International Conference on Intelligent Systems Design and Applications (ISDA)*, 2012, pp. 386–391.

[11] H. Huang, F. Shang, J. Liu, and H. Du, "Handling Least Privilege Problem and Role Mining in RBAC," *J. Comb. Optim.*, vol. 30, no. 1, pp. 63–86, Jul. 2015.

[12] L. Ma, Y. Zhou, and W. Duan, "Extended RBAC Model with Task-Constraint Rules," in *Future Information Technology*, vol. 309, J. J. Park, Y. Pan, C.-S. Kim, and Y. Yang, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, pp. 245–250.

[13] N. Pan, L. Sun, L. He, and Z. Zhu, "An Approach for Hierarchical RBAC Reconfiguration with Minimal Perturbation," *IEEE Access*, vol. 2169–3536, no. c, pp. 1–1, 2017.

[14] R. Li, H. Li, W. Wang, X. Ma, and X. Gu, "RMiner: A Tool Set for Role Mining," in *Proceedings of the 18th ACM symposium on Access control models and technologies - SACMAT '13*, 2013, p. 193.

[15] J. Vaidya, V. Atluri, and J. Warner, "RoleMiner: Mining Roles using Subset Enumeration," in *Proceedings of the 13th ACM conference on Computer and communications security*, 2006, no. January, pp. 144–153.

[16] I. Molloy, H. Chen, T. Li, Q. Wang, N. Li, E. Bertino, S. Calo, and J. Lobo, "Mining Roles with Semantic Meanings," in *Proceedings of the 13th ACM Symposium on Access Control Models and Technologies - SACMAT '08*, 2008, no. May 2014, pp. 21–30.

[17] J. Schlegelmilch and U. Steffens, "Role Mining with ORCA," in *Proceedings of the tenth ACM Symposium on Access Control Models and Technologies*, 2005, pp. 168–176.

[18] D. Zhang, K. Ramamohanarao, and T. Ebringer, "Role Engineering using Graph Optimisation," in *Proceedings of the 12th ACM Symposium on Access Control Models and Technologies*, 2007, pp. 139–144.

[19] A. Ene, W. Horne, N. Milosavljevic, P. Rao, R. Schreiber, and R. E. Tarjan, "Fast Exact and Heuristic Methods for Role Minimization Problems," in *Proceedings of the 13th ACM Symposium on Access Control Models and Technologies*, 2008, pp. 1–10.

[20] R. Li, W. Wang, X. Ma, X. Gu, and K. Wen, "Mining Roles using Attributes of Permissions," *Int. J. Innov. Comput. Inf.*

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Control, vol. 8, no. 11, pp. 7909–7923, 2012.

[21] L. Fuchs and S. Meier, “The Role Mining Process Model - Underlining the Need for a Comprehensive Research Perspective,” in *2011 Sixth International Conference on Availability, Reliability and Security*, 2011, pp. 35–42.

[22] S. D. Stoller, P. Yang, C. R. Ramakrishnan, and M. I. Gofman, “Efficient policy analysis for administrative role based access control,” in *Proceedings of the 14th ACM Conference on Computer and Communications Security - CCS ’07*, 2007, p. 445.

[23] I. Molloy, N. Li, T. Li, Z. Mao, Q. Wang, and J. Lobo, “Evaluating Role Mining Algorithms,” in *Proceedings of the 14th ACM Symposium on Access Control Models and Technologies*, 2009, pp. 95–104.