Intelligent Information Management, **2012, 4, 188-193**
http://dx.doi.org/10.4236/iim.2012.45028 Published Online September 2012 (http://www.SciRP.org/journal/iim)

Visualization Analysis of Multi-Domain Access Control Policy Integration Based on Tree-Maps and Semantic Substrates

Li Pan, Qian Xu
Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai, China
Email: panli@sjtu.edu.cn, delia1988@gmail.com

Received April 22, 2012; revised May 23, 2012; accepted June 1, 2012

ABSTRACT

The complexity of multi-domain access control policy integration makes it difficult to understand and manage the policy conflict information. The policy information visualization technology can express the logical relation of the complex information intuitively which can effectively improve the management ability of the multi-domain policy integration. Based on the role-based access control model, this paper proposed two policy analyzing methods on the separated domain statistical information of multi-domain policy integration conflicts and the policy element levels of inter-domain and element mapping of cross-domain respectively. In addition, the corresponding visualization tool is developed. We use the tree-maps algorithm to statistically analyze quantity and type of the policy integration conflicts. On that basis, the semantic substrates algorithm is applied to concretely analyze the policy element levels of inter-domain and role and permission mapping of cross-domain. Experimental result shows tree-maps and semantic substrates can effectively analyze the conflicts of multi-domain policy integration and have a good application value.

Keywords: Cross-Domain Information Exchange; Visualization Analysis; Tree-Maps; Semantic Substrates

1. Introduction

Along with the development of network technology, more and more network information services need to information exchange across trusted domains, such as collaborative computing, distributed storage, etc. The large set of the cross-domain access control policies makes the management a complicated task [1]. The policy information visualization [2] technology can express the logical relation of the complex information intuitively which can effectively improve the management ability of the multi-domain policy integration.

The characteristics of RBAC model, such as role hierarchy, least privilege and separate of duty, make it widely used in multi-domain environment. In the particular background of cross-domain information exchange, the administrators in different domains are different. When the administrator deletes, changes or adds something to the policy, conflicts may appear. The separated-domain statistical information can give the administrator a macro-cognition and help him obtain qualitative results. But it's still tough to sort out the relations between amounts of element mappings. In order to troubleshoot and resolve conflicts, he needs to learn more information about the intra-domain hierarchy and inter-domain mapping of RBAC model. So separated-domain statistical information on the macro-level and intra-domain hierarchy and inter-domain mapping of RBAC model on the micro-level guarantee the correctness and effectiveness.

Scholars applied the information visualization methods to the research on visualization analysis of access control policies. Prathima Rao *et al.* [3] proposed the multi-level gird-based technique for visualizing results of policy analysis. Xu *et al.* [4] proposed both semantic substrates and adjacency matrix technique for the policy query and the violations presentation of SELinux security policy. Reeder *et al.* [5] proposed expandable Grids tool for displaying and authoring policies. Ghazinour *et al.* [6] proposed a visualization model for privacy policy and applied it on the Facebook analysis. Above-mentioned works are for particular application scenarios, such as similarity analysis, SELinux, policy author or privacy policy etc. And such works are not related to visualization analysis of multi-domain information.

2. Preliminary

2.1. Symbol Definition

Definition 1. We define domain set

$$G:\quad G=\{G_i \mid i=1,2,3\cdots n\},$$

the policy set

$$P:\ P=\left\{P_{ij}\mid i=1,2\cdots n;\ j=1,2\cdots m\right\},$$

where P_{ij} is the No.j policy in Domain G_i, the rule set

$$R:\ R=\left\{R_{ijk}\mid i=1,2\cdots n;\ j=1,2\ldots m;\ k=1,2\cdots l\right\},$$

where R_{ijk} is the No.k rule of the policy P_{ij}. Assume the administrator of G_0 is analyzing the conflicts between G_0 and other Domains in this paper. S is the number of policy conflicts, $S(G_i)$ means the number of conflicts between Domain G_0 and G_i, $S(P_{ij})$ means the number of conflicts between P_{ij} and G_0, $S(R_{ijk})$ means the number of conflicts between R_{ijk} and G_0.

Definition 2. For representation for the user of G_i, we use $G_i\_U_i$. For the role of G_i , we use $G_i\_R_i$. For the permission of G_i, we use $G_i\_PR_i$.

2.2. Problem Analysis

In this paper, we define the visualization analysis problems aiming at the analysis of the RBAC model when cross-domain information exchange oriented.

The solutions of conflicts due to different reasons are different. The administrator needs to get the common information first, then the details of RBAC model. So the key is to solve the following two problems:

1) Obtain common information: the relation between different domains, the conflict type and quantity.

2) Obtain detail information: element hierarchy of intra-domain, the element mapping of inter-domain.

2.3. Tree Structure of the Statistical Information of Conflicts

If Domain G_0 has conflicts with G_i, it will be found as G_0 conflicts with P_{ij} etc. Actually if P_{ij} contains several rules, the behavior will be the conflicts of G_0 with R_{ijk} of P_{ij}. The quantity will satisfy the following equations:

$$S(G_i)=\sum_{j=0}^{m}S(P_{ij}) \tag{1}$$

$$S(P_{ij})=\sum_{k=0}^{l}S(R_{ijk}) \tag{2}$$

It suites the typical three level tree structure, so can be expressed by tree structure.

For each tree, the root nodes represent G_i, the child-nodes of the 2nd level represent P_{ij}, and the child-nodes of the 3rd level represent R_{ijk}. A policy consists of one or more rules. If the policy has only one rule, the 2nd level node is the leaf node. If not, the leaf node is the 3rd level node.

The attributes are: 1) the size of the node is the number of conflicts; 2) different colors mean different conflict types. According to the Shafiq [7], we define red for modality conflict, yellow for multiple management conflict, blue for cyclic inheritance conflict, green for SoD conflict.

2.4. The Relationship between Elements of RBAC

According to RBAC96 [8], we define RBAC types as follows: User, Role, Permission.

When the background is multi-domain information exchange, the relationships between those types are as follows:

Intra-Domain:

1) User Assignment (UA): a many to many user-to-role assignment relation.

2) Permission Assignment (PA): a many to many permission-to-role assignment relation.

3) Role Hierarchy (RH): the relationship between roles is hierarchy.

Inter-Domain:

4) Role Mapping (RM): the purpose is making the two roles from two different domains can access the other part.

5) Permission Equality (PE): the purpose is making the role mapping possible.

3. Policy Visualization Analysis

3.1. Tree-Maps

Tree-maps [9] algorithm is an approach in which each node is a rectangle whose area is proportional to some attribute such as node size. The traditional tree structure can express the hierarchical relation of tree structure exactly. But there are two shortages: firstly, with the growth of node number, it will overwhelm the whole screen. The user cannot get complete information; secondly, it cannot contain any other attributes, such as the size of the node, the importance of the node, etc. The rectangle-filling approach can solve these two problems. **Figure 1** shows that the size of the rectangle represents the size of the node and it can also contain the other attributes. In this paper, the size of the rectangle shows the size of the conflict number; the different color shows the different conflict type; the text information of the rectangle is the specific conflict policy. The administrator can get the statistical information from the above attributes.

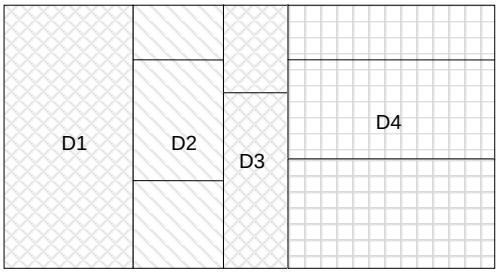


Figure 1. Tree-maps layout.

In tree-maps algorithm, the size of the node determines the size of the rectangle. The size of the root node is the sum of all the child nodes. For the 1st level child nodes, we do vertical partition according to the size proportion of each node; for the 2nd level child nodes, we do horizon partition; keep replacing the partition directions to the leaf node.

The implementation steps of tree-maps are as **Figure 2**.

3.2. Semantic Substrates

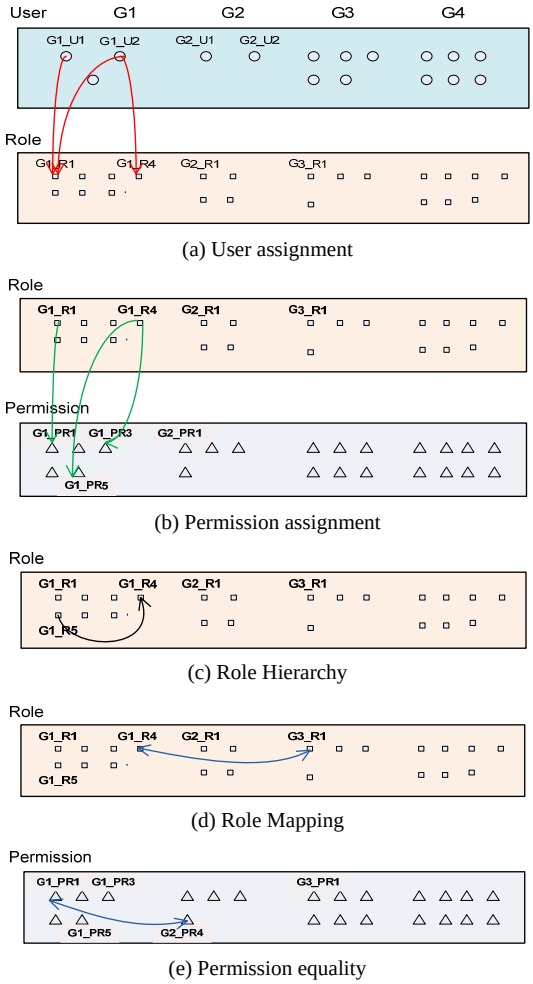
Semantic substrates [10] is a spatial template for a network, where nodes are grouped into regions and laid out within each region according to one or more node attributes. It's applicable to demonstrate the data structure which has following two features: 1) the data can be grouped according to their attributes and regions do not overlap; 2) the data of each region is the network relation, and the links between different regions have different semantics. It can solve two problems: 1) the cross of the multiple links; 2) the different semantics of the links between different regions. The complexities of analyzing the multi-domain policy based RBAC are: 1) the cross of the links due to the multiple inheritances and distribution mapping; 2) different semantics due to the five types of relations. So, semantic substrates can exactly resolve these two problems.

Two steps to organize nodes: 1) nodes are grouped into rectangular regions according to the three types: user, role and permission; 2) nodes are placed in each region according to their domain, as **Figure 3**.

The round represents user, the rectangle represents role, and the triangle represents permission.

The arrows connecting the elements, according to their different colors and different directions, show different semantic. The one-way arrow means the entities are one-way relations. The two-way arrow means the entities are two-way relations.

The steps of semantic substrates are as **Figure 4.**



(a) User assignment

(b) Permission assignment

(c) Role Hierarchy

(d) Role Mapping

(e) Permission equality

Figure 3. Example for query results.

1. **for** root Node, size(root)=sum of Size(rootNode)//calculate the size of root
2. set $O(x_1,y_1)$,$Q(x_2,y_2)$,the upper left and lower right coordinate
3. draw the rectangle
4. **for** I = 1 to num_children of 1st level, do step 5~6//for the 1st level child node
5. $x_{1i} = x_1 + \left(\sum_{j=1}^{i} \text{size}(\text{child}_i(\text{root})) / \text{size}(\text{root}) \right)^{*} (x_2 - x_1)$ //coordinate of 1st partition
6. at each (x_{1i},y_1),draw vertical line down to (x_{1i}, y_2)
7. **for** each node of the 2nd level, do step 8~11//for the 2nd level child node
8. **for** I = 1 to num_Children of 1st level
9. **for** j = 1 to num_Children n
10. $y_{2j} = y_2 + \left(\sum_{k=j}^{n} size(\text{child}_j(\text{child}_{1i})) / \text{size}(\text{child}_{1i}) \right)^{*} (y_1 - y_2)$ // coordinate of 2nd panel
11. at each(x_{1i},y_{2j}), draw horizontal line to(x_{1i},y_{2j})
12. **if** the third level exists
13. set root = $\text{child}_j(\text{child}_{1i})$, do step 5~6//according to the method of 1st level

Figure 2. The Tree-maps algorithm.

1. **Set** 3 rectangles from top to bottom, represent User, Role and permission
2. **for** each region G_i
3. **do** proportion segmentation from left to right
4. **for** each node
5. **if** (node ∈ User & &node ∈ G_i) //nodes grouped into different region
6. **do** drawRound$(x_j, y_j, \text{Black}), ((x_i, y_i) \in \text{Area}(\text{User}, G_i))$
7. **else if** (node ∈ Role&&node ∈ G_i)
8. **do** drawRectangle$(x_j, y_j, \text{Black}), ((x_j, y_j) \in \text{Area}(\text{Role}, G_i))$
9. **else** (node ∈ Permission&&node ∈ G_i)
10. **do** drawTriangle$(x_j, y_j, \text{Black}), ((x_j, y_j) \in \text{Area}(\text{Permission}, G_i))$
11. **if** $(UA(G_i\_U_j, G_i\_R_k) == 1)$ //if intra-domain exists UA
12. **do** drawOneWayArrow$(G_i_U_j, G_i_R_k, \text{Red})$
13. **if** $(UA(G_i\_R_j, G_i\_PR_k) == 1)$ // if intra-domain exists PA
14. **do** drawOneWayArrow$(G_i_R_j, G_i_PR_k, \text{Green})$
15. **if** $(UA(G_i\_R_j, G_i\_R_k) == 1)$ // if intra-domain exists RH
16. **do** drawOneWayArrow$(G_i_R_j, G_i_R_k, \text{Black})$

Figure 4. Semantic substrates algorithm.

Example:

1) Intra-domain UA. In G_1, the relation from user to role is UA, red one-way arrow. **Figure 3(a)** is the results of query "the user assignment of domain G_1".

2) Intra-domain PA. In G_1, the relation from role to the permission is PA, green one-way arrow. **Figure 3(b)** is the result of query "the permission assignment of domain G_1".

3) Intra-domain RH. In G_1, the relation between roles is RH, black one-way arrow. **Figure 3(c)** is the result of query "the Role hierarchy of domain G_1".

4) Inter-domain RM. The relation between roles in G_1 and roles in G_3 is RM, blue two-way arrow. **Figure 3(d)** is the result of query "the Role mapping from roles in G_1 to roles in G_3".

5) Inter-domain PE. The relation between permissions in G_1 and permissions in G_3 is PE, blue two-way arrow. **Figure 3(e)** is the result of query "the permission equal from roles in G_1 to roles in G_3".

4. The Visualization Implementation

We achieved the interactive visualization interface using eclipse standard 3.4.1 based on Java which assured users analyzing according to their own needs.

4.1. Tree-Maps

Figure 5 is the screenshot of the visualization analysis results, the application example is "the administrator of G_0 analyzing the conflict information with G_1, G_2, G_3, G_4". **Figure 5(a)** is the query result of "the quantity of the conflicts with each domain". **Figure 5(b)** is the query result of "conflict type statistical information".

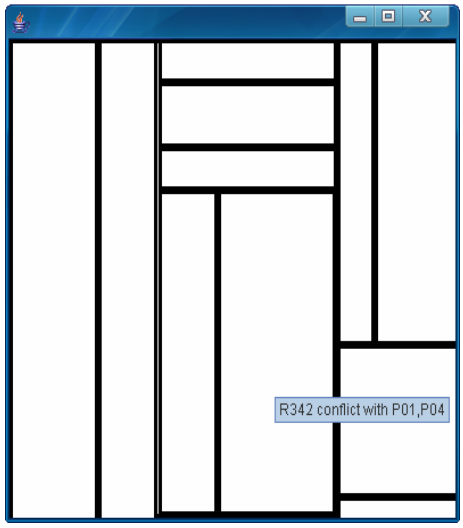
From **Figure 5(a)**, just with one look he can get G_3 has most conflicts with G_0. From the second time partition size, he knows there are 4 policies in G_3 having conflicts with G_0, and he can also get the quantity information from the size of the rectangle. From the rectangle size of the third time partition, he sees the quantity of the conflicts with each rule of each policy. He can also get the text information by moving the mouse to the related area. e.g., in **Figure 5(a)**, he can get the information "R_{312} conflict with P_{01}, P_{04}" by moving the mouse to the R_{312} area.

From **Figure 5(b)**, he can obtain the information about conflict type by the different colors of the rectangle region and also get text information by the mouse.

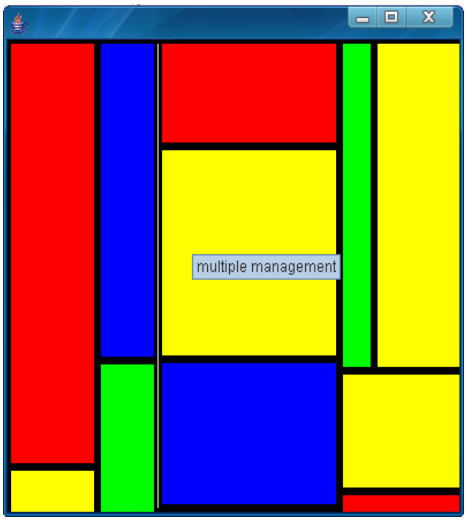
At the same time, the administrators from G_1, G_2, G_3 and G_4 can get the information about conflicts with G_0 which makes it easier for them to discuss with administrator G_0 and solve the conflicts.

4.2. Semantic Substrates

After getting the quantity and the type of the conflicts from macroscopic level, the administrator needs to check the detail information of element hierarchy when intra-domain and the mapping when inter-domain. Based on **Figure 5**, **Figure 6** is the visualization analysis result of application example: "The administrator wants to get the user assignment and permission assignment information of G_1".



(a) Conflicts quantity statistics.



(b) Conflicts types statistics.

Figure 5. Tree-maps.

He can click the relevant button to get the information. E.g., click the button "User to Role" and "Role to permission" button to achieve his aim. If he wants to cancel it, just click it again.

The user can get all the five types of information at one time by clicking all the buttons and can also just choose what they want. What's more, moving the mouse there, the user acquires the attribute. E.g. in **Figure 6**, he can move the mouse to the round of G_1_U_1 area and gain the information.

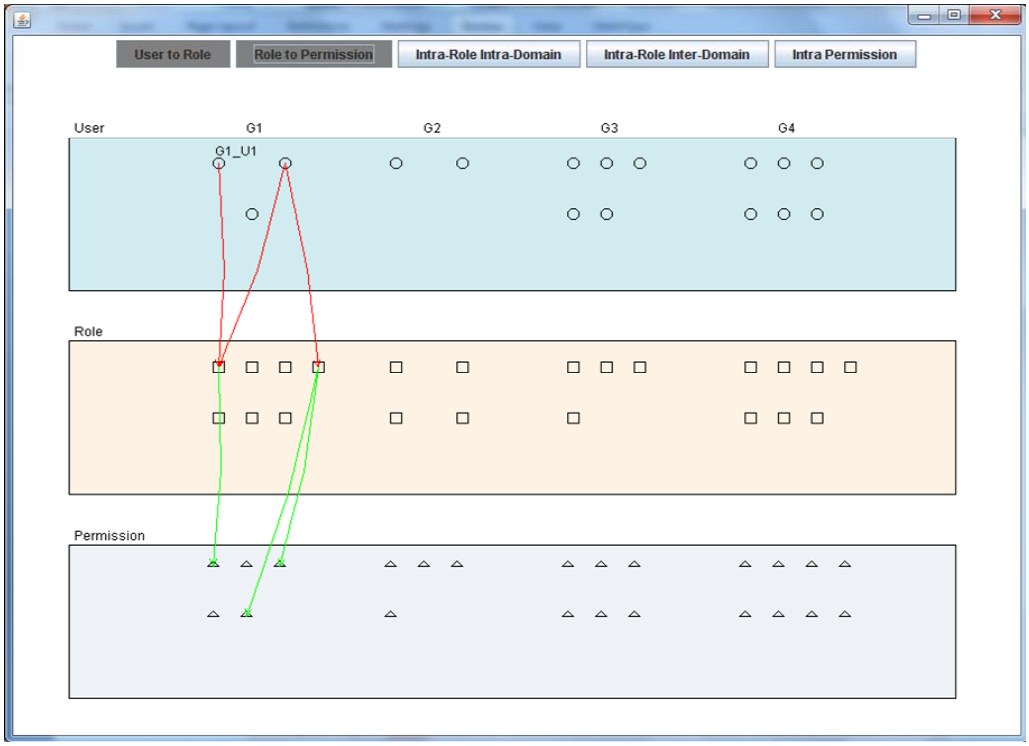


Figure 6. The screenshot of semantic substrates.

5. Conclusion

In this paper we analyzed the complexity of the policy integration when facing the cross-domain information exchange and proposed two problems which can guarantee the administrator getting proper information intuitively. Two visualization algorithms, tree-maps and semantic substrates, are applied to resolve the two problems. Furthermore, we analyzed how to use them to analyze the information, and we implemented them through Java Graphics. The current future work includes: visualization analysis contains other access control model when dealing with the multi-domain information exchange.

6. Acknowledgements

This work in the paper is supported by National Natural Science Foundation of China (Contract No. 60903191).

REFERENCES

[1] A. Schaad, J. Moffett and J. Jacob, "The Role-Based Access Control System of a European Bank: A Case Study and Discussion," *Proceedings of the Sixth ACM Symposium on Access Control Models and Technologies*, Chantilly, 3-4 May 2001, pp. 3-9. doi:10.1145/373256.373257

[2] D. Hahn, R. Shangraw, M. Keith and D. Coursey, "Does Visualization Affect Perceptions of Ethically Complex Policy Decisions: An Experimental Study," *Proceedings of the 40th Annual Hawaii International Conference on System Sciences*, Hilton Waikoloa, 3-6 January 2007 p. 96.

[3] P. Rao, G. Ghinita, E. Bertino and J. Lobo, "Visualization for Access Control Policy Analysis Results Using Multi-Level Grids," *IEEE International Symposium on Policies for Distributed Systems and Networks*, London, 20-22 July 2009.

[4] W. J. Xu, M. Shehab and G.-J. Ahn, "Visualization Based Policy Analysis: Case Study in SELinux," *Proceedings of the 13th ACM Symposium on Access Control Models and Technologies*, Estes Park, 11-13 June 2008.

[5] R. Reeder, L. Bauer, L. Cranor, M. Reiter, K. Bacon, K. How and H. Strong, "Expandable Grids for Visualizing and Authoring Computer Security Policies," *CHI'08: Proceedings of the Twenty-Sixth Annual SIGCHI Conference on Human Factors in Computing Systems*, Florence, 5-10 April 2008.

[6] K. Ghazinour, M. Majedi and K. Barker, "A Model for Privacy Policy Visualization," *Proceeding of the 4th IEEE International Workshop on Security, Trust, and Privacy for Software Application* (*STPSA* 2009), Seattle, 20-24 July 2009.

[7] B. Shafiq, J. B. D. Joshi, E. Bertino and A. Ghafoor, "Secure Interoperation in a Multi-Domain Environment Employing RBAC Policies," *IEEE Transactions on Knowledge and Data Engineering*, Vol. 17, No. 11, 2005, pp. 1557-1577.

[8] R. Sandhu, E. Coyne and H. Feinstein, "Role-Based Access Control Model," *IEEE Computer*, Vol. 29, No. 2, 1996, pp. 8-47.

[9] B. Johnson and B. Shneiderman, "Tree-Maps: A Space-

Filling Approach to the Visualization of Hierarchical Information Structures," *Proceedings of the* 2*nd Conference on IEEE Visualization*, San Diego, 22-25 October 1991, pp. 284-291.

[10] A. Aris and B. Shneiderman, "Designing Semantic Substrates for Visual Network Exploration," *Information Visualization*, Vol. 6, No. 4, 2007, pp. 281-300. doi:10.1057/palgrave.ivs.9500162

