DELEGATION MODEL FOR CSCW BASED ON RBAC POLICIES AND VISUAL MODELING *

ZHANG Zhiyong , PU Jiexin

Electron.Inf.Eng.Coll., Henan Univ.of Sci. & Technol., Road Xiyuan No.48 Luoyang, Henan 471003, China
E-mail:zhangzy@mail.haust.edu.cn

Abstract: Traditional access control of CSCW system is lack of delegation mechanism. For improving it and achieving resource share in the heterogeneous distributed environment, Role-Activity Based Delegation Model for CSCW (abbr. RABDM for CSCW) was presented based on RBAC policies and the important properties of activity, object-oriented visual modeling was also further represented using Unified Modeling Language. The model with time and constraint characters solves the issue of authorization centralization in the actual application of Graphics and Images Collaboration Processing System Based on CSCW, and realizes distributed and dynamic characters. Visual modeling also shortens the gap between the abstract theory model and implementation of the applied system, and benefits designing and development of systems based on the self-contained model.

Keyword: access control; computer support cooperative work; role; delegation; constraint rule.

1 Introduction

The goal of CSCW system is realizing cooperative work and resource sharing among users on the heterogeneous distributed network platform, and access control ensures efficient and secure collaboration and sharing. Thus far, the research of access control has concentrated on centralized authorization based on RBAC policies and models. However, with large numbers of users and objects increasing, traditional authorization is not suited to distributed CSCW environment, and centralized sever also will not endure due to burdensome authorization.

As the essential component of authorization and key technology of fulfilling distributed characters, delegation solves the above mentioned questions, but now there is lack of further research and application. Some issues and methods of role-based collaboration were represented by Haibin Zhu, such as role assignment and migration, role coordination, role collision and so on [1]. Anand Tripathi realized role model and domain policy that meet dynamic security, but a formal security model was not specified [2]. A role-based access control model for CSCW was formally defined, and it focused on basic component and general authorization rules, not dealing with delegation mechanism [3, 4]. This paper specifies a delegation model with time characters and constraint rules, called RBADM for CSCW, and represents access control architecture of CSCW system and object-oriented visual modeling in the basis of on the model.

2 Delegation Model and Related Features

The basic idea of delegation is that active entity (user, process, agent, et. al.,) in application could grant some own permissions or roles to others, which can carry out some privileges and functions on behave of the former. For example, in the enterprise organization, somebody could delegate some permissions to other staffers and share privileges with them because of being absent or needing to collaborate with others. At the same time, he could also revoke these delegated-rights in need. The concepts related to delegation have delegator, delegated role or permission and delegatee.

Delegation has some important features as follows:
1. Delegation Granularity: The unit of Delegation has three kinds as follow: Permission-based thin granularity [5], role-based medium granularity [6, 7], permission and role -based fat granularity proposed by ZHANG Zhiyong [8]. Thin granularity means that user could delegate the partial permissions of a role to delegtee, not just whole role. So granularity is depressed, and it meets the principle of least privilege, but brings about some non-integrity role led to authorization complexity .For

* This work is partially supported by National Nature Science Foundation of China (Grant No.60475021), Henan Province Science Fund for Distinguished Young Scholars (Grant No.0412000400).

medium granularity, delegtor only could delegate role as a whole, thus delegtee would acquires entire permissions of role. Apparently, it sacrifices the principle of least privilege at some degrees to eliminating non-integrity role. Fat granularity allow user to delegate own permission or role discretionarily, which is flexible compared with above two granularities besides of complex realization. With regard to delegation unit, it should be chosen according to applied system.

2. Delegation Step: It is subdivided into single-step delegation and multi-steps delegation. The former is that delegatee could not delegate role or permission to others further; the latter allows delegatee to grant further, but in the condition revoke is more complicated.

3. Delegation Revoking: The contrary operation of delegation is revoking which means that delegated roles or permissions are called off. Revoking mainly includes the following features, such as cascading revoke, non-cascading revoke, grant-independent revoke, grant-dependent revoke, system automatic revoke and user revoke.

3 RABDM for CSCW and Formal Definitions

In this section, RABDM for CSCW is specified including basic ideas, formal definitions of elementary components and relationships, time characters and properties, as well as constraints rules of delegation.

3.1 Basic Ideas

With regard to the above mentioned issues in Section 2, delegation granularity is role, and multi-steps delegation and grant-independent revoke is also adopted in RABDM for CSCW.

Delegation is core of RABDM. This paper proposes DRG (Delegated Role Group) that is a set of DR. In the process of delegation user could independently create DRG, not via administrator, then assign one or more DRs including explicit role and implicit inherited roles to DRG, at last delegate DRG to another user. Thus delegatee could perform privilege on behalf of delegtor. On account of DRG having time property, life-cycle of DRG is depicted in the model, thereby enriches semantic further.

Based on RBAC96 model family, RABDM introduces two concepts in the CSCW environment: task and activity. The former is a integrated workflow, divided into some activities with time sequence; the latter acquires privilege through activity-permission assignment. At the same time, Role-activity assignment and user-role assignment are the relations among user, role and activity. Besides of the one to one relations of role-activity and user-session, others are many to many relations. The model contains some basic components, such as user, regular role, delegated role, DRG, permission, task, activity, constraint and session, as Figure 1.

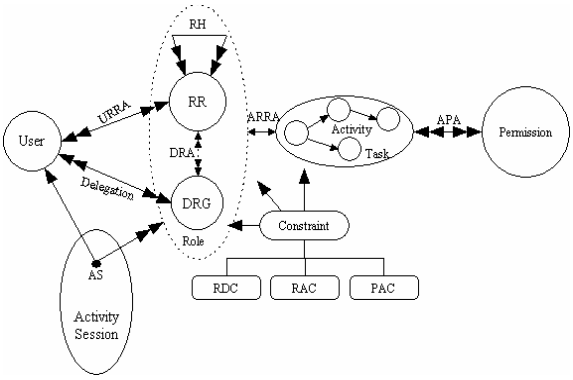

Figure 1. RABDM for CSCW

3.2 Main Components and Relationships

The following is a list of RABDM for CSCW components: U, R, AR, RR, DR, DRG, S, P, O, C, RDC, AS, T, A are respectively defined as sets of user, role, active role, regular role, delegated role, delegated role group, states, permission, object, constraint, role delegation constraint, activity session, task, activity.

Definition 3.2.1(Operation): Operation is defined as a action from user's accessing to object, which is shared resource, such as graphics and images files, blackboard, database, and so on. Operations are not only actual actions like reading, writing, executing, but also some abstract actions which are graphics and images filter, format conversion in graphics and images processing system based on CSCW.

Definition 3.2.2(Task & Activity): Roles' collaborative job is called task, the every step of task is called activity. Activity has dynamic and atom characteristic, and is basic unit of task.

$\forall\, t(t \in T) t=\{a_1, a_2 \ldots a_n \mid a_i \in A\}$

Definition 3.2.3(Activity-Permission Assignment): APA means that administrator assign some permission to activity. The relation is many to many.
$APA \subseteq A \times P$

Definition 3.2.4(Activity-Regular Role Assignment): The process of assigning activity to role is called ARRA, and the relation is one to one, depicted a tow tuple (A,RR).
$ARRA \subseteq A \times RR$

Definition 3.2.5(User-Regular Role Assignment): URRA is that user is assigned some roles according to user's duty. The relation between user and regular role is many to many.
$URRA \subseteq U \times RR$

Definition 3.2.6(Delegated Role Group): DRG is a set of delegated roles, and is also a subset RR.
$DRG=\{r_1,r_2,\ldots r_n \mid r \in RR\}$,namely $DRG \subseteq RR$.

Definition 3.2.7(Delegated Role Assignment)): DRA is a many to many relation between RR and DRG, it is assignment from RR to DRG.
$DRA \subseteq RR \times DRG$

Definition 3.2.8(Activity Session): AS is a mapping relation between U and R, denoted by f(u,ar).
$f(u,ar)$：$u \rightarrow ar(u \in U, ar \in AR)$

Definition 3.2.9(Delegation): Delegation relation is five tuple （ U_1,U_2,DRG,DTL,RDC ） ,where U_1 is delegator, U_2 is delegtee, TL is the limitation set of delegation time, and RDC is the conditions set of role delegation constraints. The relation is that user U_1 could delegate DRG to another user U_2, thereby U_2 acquires whole explicit and implicit permissions of DRG, where delegation must meet prerequisite conditions of RDC.

3.3 Time Characters and Properties of Delegation

Definition 3.3.1(States Set): The states set of DRG S={init, invoke, sleep, expire}, init is beginning-state, invoke is active-state, sleep is sleepy-state, expire is exiting-state.

Definition 3.3.2(Delegation Time Limit):DRG has the property of time limitation, DTL={x $\mid$ x=[τ_{bi},τ_{ei}](i=1,2…n)}, where τ_{bi} is begin-time , and τ_{ei} is end-time.

Definition 3.3.3(State Transitions):ST is system time, $\forall$ i(i ∈ N) ST ∉ [τ_{bi},τ_{ei}] ∧ ST<τ_{b1} → S=init； ∋ i (i ∈ N)ST ∈ [τ_{bi}, τ_{ei}] → S=invoke ； $\forall$ i(i ∈ N)ST ∉ [τ_{bi}, τ_{ei}] ∧ (ST>τ_{b1}) ∧ (ST<τ_{en}) → S=sleep ； $\forall$ i(i ∈ N)ST ∉ [τ_{bi}, τ_{ei}] ∧ ST>τ_{en} → S=expire.

Property 3.3.1(Activity Time-Order): The relation of activities is synchronization or concurrent. Between every two activities exits steady time order that meet partial order relation, denoted by "□".
$\forall t,a_i,a_j(t \in T,a_i,a_j \in A)(a_i \in t \wedge a_j \in t \rightarrow a_i \square a_j)$
$\forall t,a_i,a_j,a_k(t \in T,a_i,a_j,a_k \in A)(a_i \square a_j \wedge a_j \square a_k \rightarrow a_i \square a_k)$

Property 3.3.2(DR Run-Order): DR is executed in serious or concurrently, and is partial order, denoted by "$\geqq_{DR}$".
$\forall t,dr_i,dr_j,dr_k(t \in T,dr_i,dr_j,dr_k \in t)(dr_i \geqq_{DR} dr_j \wedge dr_j \geqq_{DR} dr_k \rightarrow dr_i \geqq_{DR} dr_k)$

Property 3.3.3(Activity-DR Time–Order Consistent): Activity and DR is consistent in time order owing to their one to one mapping relation.
$\forall t,a_i,a_j,dr_i,dr_j(a_i,a_j \in t)(g(a_i,dr_i) \wedge g(a_j,dr_j) \wedge a_i \square a_j \rightarrow dr_i \geqq dr_j \vee \neg(dr_i \geqq dr_j) \wedge \neg(dr_j \geqq dr_i))$

3.4 Constraints Rules of Delegation

The traditional authorization is centralized, and the actor of regular permission and role assignment is system administrator or security officer,; but in the distributed environment delegation is more decentralized, delegator is anyone of users. Delegation constraint can strengthen delegation management, preventing users' intended or involuntary illegal delegation, as well as improving the confidentiality, integrity and controllability. The important constraints rules like non-delegated roles, delegation collision, delegation depth and delegation cardinality are involved in RABDM for CSCW. The delegation policies are consisted of above the rules that are defined as follows. There is not AR constraint other than RBAC96 model family due to one to one mapping relation between U and AS.

Definition 3.4.1(Non-Delegated Role Set): NDP is a set of whole roles that are not delegated to others.
$NDR=\{r_1,r_2,\ldots r_i\}$

Constraint Rule 3.4.1(Non-Collision Delegation Constraint): The elements of NDR can not be delegated.
$\forall x(x \in NDR) \rightarrow (x \notin DRG)$
Definition 3.4.2(Delegation Collision Role): Two roles r_i and r_j are delegation collision roles, if they are not delegated to others at the same time, we denote it by $collr(r_i, r_j)$.

Constraint Rule 3.4.2(DRG Non-Collision Constraint): Every two roles do not exist delegation collision in DRG set.
$\forall r_i,r_j (r_i \in DRG, r_j \in DRG) collr(r_i, r_j)=F$

Definition 3.4.3(Delegation Depth and Cardinality): Delegation Depth d is a natural number about cascading delegation degree of role r_i .It is called single-step delegation if d=1,and is called multi-step delegation if d>1.Delegation cardinality is also a natural number about delegated users' number of role r_i.

Constraint Rule 3.4.3(Multi-Steps Delegation Constraint): Delegation depth and cardinality of DRG lie on minimum delegation depth and cardinality of all roles in DRG.
$\forall r_i (r_i \in DRG)\ d_{DRG}<d_{ri}$
$\forall r_i (r_i \in DRG)\ n_{DRG}<n_{ri}$

Property 3.4.1(Multi-Steps Delegation): DRG multi-steps delegation is a partial order relation with reflexive, antisymmetric, transitive properties.

Property 3.4.2(Cascading Delegation Revoke): When original user revokes delegation or ST exceeds DTL, the whole roles and permissions of DRG which is multi-steply delegated will be revoke cascadely.

Property 3.4.3(Grant-Independent Revoke): Every delegator in the delegation path could revoke delegation, not only original delegtor has right to revoke.

Property 3.4.4(System Imperative Revoke): When system clock exceed to DTL of DRG, CSCW system automatically revoke delegation including all explicit and implicit roles.

4 RABDM for CSCW Visual Modeling

In distributed computing environment, RABDM with distributed character resolves the issue of authorization complexity owing to centralized administration. There is necessary to system modeling by the thinking of object-oriented and UML, which supports object-oriented analysis and designing, thus shortening the gap of academic abstract model and implementation of actual applied system, and benefiting designing and development applied system based on RABDM [9].

4.1 Visual Static Modeling

With regard to static modeling of RABDM, use case diagram, entity class and class relationship diagram are mainly represented. System functions are introduced from user's angel in use case diagram. Users are basically subcategorized into four kinds, which fulfill different functions respectively. System administrators manage the assignment of users, roles and privileges; system officers take charge of authorization constraint management and delegation constraint management, carrying out the constraints of role, permission and session; system auditors mainly track operations of system administrators' authorization, and audit delegation process and data access to applied system; regular users can create and close activity session, delegate DRG to others and access shared data. The functions of RABDM system are illustrated as Figure 2.

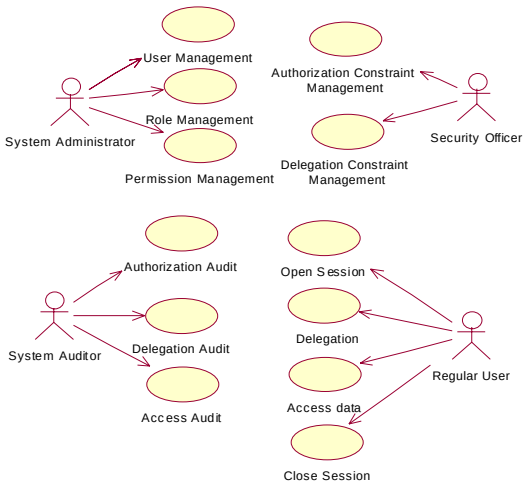

Figure 2. Use Case Diagram

Some important entity classes including attributes and methods, class relationship represented the static characters of RABDM. The relationships mainly include association, aggregation, generalization. Role class is generalized into two subclasses: regular role and DRG, which inherit the public attributes and methods of parent-class, as well as possessing private attributes and methods respectively. Constraint class is also generalized into RDC, RAC, PAC. These constraints respectively effect on delegating DRG, assigning role and permission. The relationship and multiplicity among classes are not illustrated in detail owing to length limitation of the paper.

4.2 Visual Dynamic Modeling

About dynamic modeling of RABDM for CSCW, interaction diagram and object behavior diagram are described. All dynamic characters are not introduced because of length of the paper, and the following is a role delegation sequence diagram (a kind of interaction diagram). The whole process of DRG delegation and sequence of every step are detailedly specified in Figure 3.

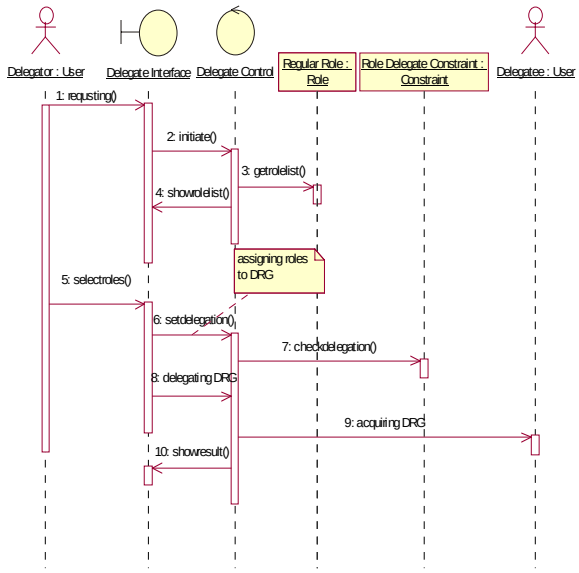

Figure 3. Delegation Sequence Diagram

5 Conclusion and Future Work

RABDM for CSCW is a self-contained model that realizes delegation in distributed CSCW system, and improves access control in GICPS. It also solves the limitation issue of lacking delegation in CSCW system. The future research focuses on the context constraints of delegation and enriching the model semantic further.

References

[1] Haibin Zhu, "Some issues of role-based collaboration," IEEE CCECE2003- CCGEI2003, Montreal, May, 2003.

[2] Anand R.Tripathi, Tanvir Ahmed, and Richa Kumar, "Specification of secure distributed collaboration system," IEEE Proceedings of the sixth International Symposium on Autonomous Decentralized Systems, 2003.

[3] LI Cheng-kai, ZHAN Yong-zhao, and XIE Li, "A role-based access control model for CSCW systems," Journal of Software, vol.11, pp. 931- 937, July 2000.

[4] Xiao Daoju, Liu Chao, and Chen Xiaosu, "The security model of CSCW system based on RBAC," J. Huazhong Univ.of Sci.&Tech.(Nature Science Edition), vol.32, pp. 56-58, May,2004.

[5] Xinwen Zhang, Sejong Oh, and Ravi Sandhu, "PBDM:a flexible delegation model in RBAC," SACMAT'03,Como, Italy, June2-3,2003.

[6] Ezedin Barka, and Ravi Sandhu, "A role-based delegation model and some extensions," The 16th Annual Computer, Sheraton New Orleans, 2000.

[7] SangYeob Na, and SuhHyun Cheon, "Role delegation in role-based access control," ACM RBAC2000, 2000.

[8] ZHANG Zhiyong, and PU Jiexin., "Permission-role based delegation model and object-oriented modeling", National Open Distributed and Parallel Computing Symposium2004, Beijing, China, Oct. ,2004.

[9] ZHANG Zhiyong, "Role-based access control and object-oriented modeling," Computer Engineering and Design, vol.25, pp.1367-1369,1374, Aug., 2004.