An Effective Solution of Privileges Privatization in Information Grid

Xiao-Yan Liu[1], Qing-Lun Xiao [2]

1.School of Computer Science & Technology, Henan Polytechnic University, Jiaozuo Henan 454000

2.School of Communist Youth League, Henan Polytechnic University, Jiaozuo Henan 454000

e-mail xyanliu@hpu.edu.cn

Abstract: Introduce several methods of privileges privatization. A new kind of security access control mechanism (RBAC) is put forward to solve privileges privatization of privileges management issue in information grid. Research on basic concepts and principles of RBAC mechanism and make a study of the access control description strategy and realization method playing the role as the foundation in this mechanism. Based on the mechanism, privileges management among domains cooperation is solved, thus it satisfies the basic requirements of resource informatization, vitrification and sharing.

Keywords: information grid; access control; privileges management

1 Introduction

In recent years, with improvement in the level of information technology, information systems security issues become more and more important. In the field of information systems, access control efforts of the traditional discretionary access control are too weak, and mandatory access control are too strong. With the wide use of computer in commercial and civil sectors, different departments have different security needs. Most of the needs are unable to use discretionary access control and mandatory access control to describe and control. However, role-based access control can not only have functions of discretionary access control but also have functions of mandatory access control through the role of configuration.

Role-based Access Control(RBAC) is a new access control technique proposed by National Institute of Standards and Technology(NIST) in the 20th century and early 90. On the basis of system users' behavior, role-based strategy controls the access of information. This kind of access control policy description method has great influence in the field of information grid.

A role can be obtained through the inheritance of all the privileges of another role in Multi-level RBAC matrix put forward in RBAC96. Although the significance of multi-level has been widely recognized, it has also brought a number of defects: The role of the father through the son can have all the authority of sub role by inheriting; This makes the top role have higher authority and can access all the resource of sub role. But in reality, certain rights can only be owned by sub role, namely, sub role sometimes need to own certain private rights. For instance, in a software development company, project manager's role is the father role of test engineer's role and software development engineer's role. Through the role of inheritance, project manager's role gains all the privileges of test engineer's role and software development's role. However, test engineer's role and software development's role need to reserve certain rights as private to be not inherited by project manager's role. In information grid, for the issue of role authority privatization, many scholars have conducted extensive research.

2 Studies of Authority Privation

With the role privileges privatization, Sandhu puts forward the conception of Private Role in literature [1]: add a private role r' for each role own private role, put privileges which can be inherited by father role in the r and put permissions which can't be inherited by father role in the r'. Each role in an organization may all own private rights, so we have to build private role for each role thus doubling the number of roles. Therefore, use this method will enable rapid expansion of the number of roles, so that the role of management becomes very complicated.

Some scholars have put forth a depth of identifier used

to address the issue of role privileges privatization. This method use access components to denote rights, in the form of (m,n):(s,o)|r1,r2,…ri>0|, m、n respectively denote depth of privileges along the dissemination of the subject and the object, s、o respectively denote the subject and the object of access, r1,r2, …ri denote access rules. While m>0, it shows rules are inherited by upper subject; While m=0, it shows rules spread not along subject level; While m<0, it shows rules are inherited by lower subject; While m=∞, it shows rules are inherited by upper subject until the highest level; While m=-∞, it shows rules are inherited by lower subject until the lowest level. For the object, n has the similar definition as m. The role of authorization is to grant the role to visit components. Role inheritance in components can be reckoned through following way: first, obtain subject collection S and object collection O that past by the rules of access components on the basis of m,n; then spread each (s',o')∈S×O by the rules of access components and form basic rules(s',o') |r1,r2, …ri>0|. In this way, each access component at least has m×n rules. This could easily lead to the rules of combination of the explosion.

Literature[2] puts forth another effective program of solving privatization of the role of authority.

This scheme makes some modifications for RBAC: definite the form of privileges as X(x,n), n≥0 show privileges spread up only, n=∞ show privileges can be inherited by all upper roles; the form of R is defined as (rname,rpset), rpset= rpsetl∪rpset2, rpsetl denotes privileges while n=0(collection of private privileges), rpset2 denotes privileges while n>0(collection of public privileges). Public privileges can be inherited by upper roles. Through inheritance, father role can automatically own public privileges of sub role which dissemination depth need to minus 1. When its dissemination depth become 0, the privileges inherited can be added to the private privileges collection of father role, else be added to the public privileges collection.

All these methods conduct a study from different sides of privileges privation, but have shortcomings of role inflation and combinational explosion. We study an effective solution of role privileges privation which makes the privileges management of role become more convenient and easier.

3 An Effective Solution of Authority Privation

On the basis of RBAC, we redefine privilege as (P,N), P denotes privilege, N denotes spread depth of privilege, that is, privilege P along the main role of the level of succession can be up to N layers. Following are some correlative definitions:

(1) Public Privileges(PUP): spread depth is N=∞. It can be understood as open to public, must be inherited by superior role. It can be inherited by father role but has no relation with inherited levels.

(2) Private Privileges(PRP):spread depth is N=0. It can be understood as non-public, dedicated privilege. It can't be inherited by father role.

(3) Protect Privileges(POP):spread depth is ∞>N>0. It can be inherited by father role but can only be inherited in certain depth, and its spread depth decreases after each inheritance.

(4) Form of the role is(RN,RP), RN denotes the name of the role, RP denotes RN privilege collection. RP=RPUPS∪RPRPS∪RPOPS, RPUPS denotes public privilege of the role, RPRPS denotes private privilege of the role, RPOPS denotes protect privilege.

(5) Functions of obtaining role privilege:

PUPS(R): return public privileges of role R

PRPS(R): return private privileges of role R

POPS(R): return protect privileges of role R

PS(R): return all privileges of role R

(6) Function Rank(R,P): return spread depth N of privilege P in role R.

(7) Role R2 directly inherits R1 is recorded as $R2 \rightarrow R1$. Role R2 inherits R1 in n stages is recorded as $R2 \xrightarrow{n} R1$. Father role inherits sub role in n stages refers that father role obtains privileges of sub role by inheritance, the spread depth of which in father role decreases n compared to spread depth in sub role.

Role inheritance rules in traditional matrix are not good solutions to role inheritance. According to practical application and need of role privileges privation in information grid, we define a new mode of role inheritance.

(1) Public Inheritance(PUI): If R1、R2 meet $R2 \xrightarrow{n} R1$, then we call inheritance which meets following terms as public inheritance and record it as $R2 \xrightarrow[PUI]{n} R1$:

① $\forall p \in PUPS(R1)$, $p \in PUPS(R2)$;

② $\forall p \in PRPS(R1)$, $p \notin PRPS(R2)$;

③ $\forall p \in POPS(R1)$, if $n' = Rank(R1,P) - n \geq 0$, then $(p, n') \in PS(R2)$。

In public inheritance, father role can't inherit private privileges of sub role and can only inherit protect privileges and public privileges of sub role. Protect privileges are still protect privileges or private privileges after inherited and public privileges are still public privileges after inherited.

(2) Private Inheritance(PRI): If R1、R2 meet $R2 \xrightarrow{n} R1$, then we call inheritance which meet following terms as private inheritance and record it as $R2 \xrightarrow[PRI]{n} R1$:

① $\forall p \in PUPS(R1) \cup POPS(R1), p \in PRPS(R2)$;

② $\forall p \in PRPS(R1), p \notin PS(R2)$。

In private inheritance, father role inherit all non-private privileges of sub role as private privileges but can't inherit private privileges of sub role.

(3) Protected Inheritance(POI): If R1、R2 meet $R2 \xrightarrow{n} R1$, then we call inheritance which meet following terms as protected inheritance and record it as $R2 \xrightarrow[POI]{n} R1$:

① $\forall p \in PUPS(R1), (p, n) \in POPS(R2)$;

② $\forall p \in PRPS(R1), p \notin PS(R2)$;

③ $\forall p \in POPS(R1), p \in PRPS(R2)$。

In protected inheritance, protected privileges of sub role are all inherited as private privileges by father role, however, public privileges of sub role will be inherited as protected privileges with spread depth of n by father role.

According to above three kinds of role inheritance modes, forms of sub-role's privileges in father role are as Table 1 shown.

We can obtain public privileges、private privileges and protected privileges basing on above definition of the new role and privileges inheritance mode. Suppose form of role is (RN, RP), which RN denotes role's name and RP denotes privilege set of role RN. $RP=RPUPS \cup RPRPS \cup RPOPS$, which RPUPS denotes public privilege set, RPRPS denotes private privilege set, RPOPS denotes protected privilege set. RPUPS、RPRPS、RPOPS can be obtained by following arithmetic:

(1)Obtain RPUPS、RPRPS、RPOPS of which role RN gets through direct warrant.

(2)Call the algorithm to derive all the sub role of role RN.

(3)Obtain public privileges PUPS(Ri) and protected privileges POPS(Ri) of each sub role Ri.

(4)According to Ni stages inheritance of sub-role Ri by RN and the new inheritance mode, we handle privileges of sub-role Ri $P1 \in PUPS(Ri)$ and $P2 \in POPS(Ri)$ as follows, suppose $Ni'=Rank(Ri,P2) - Ni$:

①Public inheritance: Add P1 to RPUPS of father role RN; If $Ni'>0$, then add (P2,Ni') to RPOPS of RN; If $Ni'=0$, then add (P2,Ni') to RPRPS of RN;

②Private inheritance: Add P1 to RPRPS of father role RN; If $Ni' \geqq 0$, then add (P2,Ni') to RPRPS of RN;

③Protected inheritance: Add (P1,Ni) to RPOPS of father role RN; If $Ni' \geqq 0$, then add P2 to RPRPS of Rn.

(5)Output RPUPS、RPRPS and RPOPS.

Public inheritance		Private inheritance		Protected inheritance	
Sub role	Father role	Sub role	Father role	Sub role	Father role
PUP	PUP	PUP	PRP	PUP	POP
RP	invisible	PRP	invisible	PRP	invisible
POP	PRP POP	POP	PRP	POP	PRP

Figure 1. Forms of sub-role's private privileges in father role

4 Conclusion

To be aimed at shortage of current research on privileges privatization, this paper puts forth an effective solution of privileges privatization. This method makes privileges management become more convenient and easier to achieve.

References

[1] Sandhu R, Coyne E J, Feinstein H L, etal. Role-Based Access Control models[J]. IEEE Computer, February 1996, 29(2): 38-47.

[2] Ahn GJ, Sandhu R. Role authorization constraints specification[J]. ACM Trans. On Information and System Security, 2000, 3(4): 207-226

[3] Shao Guiwei. Research on access control in information grid based on role[D]: [Master's degree thesis]. Hefei: HeFei University of Technology, 2006

[4] Song Chunlei. Research on data integration method based on information grid[D]: [Master's degree thesis]. Dalian: DaLian Maritime University, 2006

[5] Crampton J, Loizou G. Administrative Scope and Role Hierarchy Operations[C]. Proc. of the 7th ACM Symposium on Access Control Models and Technologies, 2002, 145-154

[6] Sandhu R, and Munawer Q. Configuring Role-Based Access

Control To Enforce Mandatory and Discretionary Access Control Polices[J]. ACM Trans. on Information and System Security, May 2002, 3(2): 85-106

[7] David F Ferraiolo, Ravi Sandhu, Serban Gavrila, etal.Proposed NIST Standard for Role-based Access Control[J]. ACM Trans on Information and System Security, 2001, 4(3): 224-274

[8] Moyer M J, Ahamad M. Generalized Role-Based Access Control[C]. In: Proc. of. The 2001 Intl. on Distributed Computer Systems (ICDCS), 2001, 10:391-398

[9] Sandhu R, Ranganathan K, Zhang X. Secure information sharing enabled by Trusted Computing and PEI models[C]. Proceedings of the 2006 ACM Symposium on Information, computer and communications security. 2006: 2-12

978-1-935068-01-3 © 2009 SciRes.