一种基于 RBAC 的 MSTRBAC 角色授权模型①

An MSTRBAC Role Authorization Model Based on RBAC

陈正铭　(韶关学院 计算机科学学院 广东 韶关 512026)

摘　要： 通过对 RBAC 模型与其扩展方向的研究，发现 RBAC 模型不能解决授权的时间和空间依赖，提出了解决此问题的 MSTRBAC 模型。此模型在 RBAC 模型中引入了强制性、时间约束、空间约束的管理方法，具有安全性完备，授权灵活，实现简单的特点。最后给出了 MSTRBAC 模型的其中一种实现。

关键词： 权限 角色 授权 RBAC MSTRBAC

1　RBAC模型概述

基于角色的访问控制（RBAC）引入了角色（Role）的概念，目的是为了隔离用户与权限。所有的授权应该给予角色而不是直接给用户。角色-权限是多对多的关系，易于利用关系数据库实现。基于角色的访问控制方法（RBAC）的显著的两大特征是：①由于角色/权限之间的变化比角色/用户关系之间的变化相对要慢得多，减小了授权管理的复杂性，降低管理开销。②灵活地支持系统的复杂的安全策略，有很大的伸缩性。

定义 1．符号定义

$P=\{a|\forall a_i\in P\wedge\forall a_j\in P\wedge i\neq j\rightarrow a_i\cap a_j=\Phi\}$:系统权限集。

$R=\{r|r\subset P\}$:系统角色集。

$U=\{u_i|i=1,2,\ldots\}$:用户集。

$\mathrm{UR}=\{(u,r)|u\in U,r\in R\}$:用户角色集。

定义 2．操作定义

$r/u_i|\rightarrow u_j(i\neq j\wedge(u_i,r)\in UR)$:用户 u_i 申请将角色 r 授予用户 u_j，前提是用户具有角色 r 且不自我授权。

$r/u_i|\leftrightarrow u_j((u_j,r)\notin UR)\wedge(i\neq j\wedge(u_i,r)\in UR))$:用户 u_i 将角色 r 授予用户 u_j 操作成功，前提是用户 u_i 可以授权 u_j 角色 r，且目前还未授予 u_j 角色 r。

$r/u_i|\leftarrow u_j(r/u_i|\leftrightarrow u_j\wedge(u_i,r)\in UR)$:用户 u_i 回收授予用户 u_j 的角色 r，前提是 u_j 具有角色 r，且 u_i 是用户 u_j 的角色 r 主授方。

2　扩展RBAC的研究

2.1 RBAC 层次扩展 NIST 模型概述

NIST 模型就是通过对 RBAC 的一种实际应用的扩展，NIST 模型分为 4 个层次：Flat RBAC，Hierarchical RBAC，Constrained RBAC 和 Symmetric RBAC[1-3]。

(1)Flat RBAC：角色赋给用户，权限赋给角色，用户通过获得角色从而拥有相应的权限，用户和角色之间和角色和权限之间都是多对多的关系。为了安全考虑，授予操作要求提供一种检查机制。

(2) Hierarchical RBAC：这一层要求对角色分等级的支持，这种等级是用数学上偏序关系定义，等级高的角色可以获得等级低的角色的权限。在这一方面又可分为：General Hierarchical RBAC 和 Restricted Hierarchical RBAC。

(3) Constrained RBAC：这一层要求加强权力、职责的分离，要求角色权限最小化，通过把权限分给多个不同的角色，然后再把这些角色分给不同的人，这样就能阻止一个人拥有过多的角色，就避免因一人的失误造成的不良影响。

(4)Symmetric RBAC：要求在第一层对用户、角色分配进行安全检查基础上，还需要对角色、权限分配进行安全检查，而且还要遵守上面第三层提到的权限最小化分配原则。

2.2 带时间约束的扩展 RBAC

在实际的很多应用中，用户、角色、权限之间同时间的关系非常密切，例如，规定在特定每天时间段

① 基金项目:韶关市技术创新项目(韶科(教)2007-03);韶关学院科研项目(韶学院[2007]220-6)
收稿时间:2008-08-14

内用户才能进行某些操作。国内外关于带时间约束的RBAC的研究文章很多[4,5]，TRBAC (Temporal Role Based Access Control)[5]就是其中一个典型的对RBAC的时间约束的扩展。在TRBAC中，时间约束一般可分为如下几种：①激活时间范围约束；②激活时间长度约束；③时间范围内激活时间长度限制。

2.3 基于工作流的扩展RBAC

WRPTAC(Weighted Role and Periodic TimeAccess)[6]是RBAC在工作流中的具体应用。工作流是任务和任务之间约束关系的集合，任务是工作流的最小可执行单元，任务之间可能存在复杂的依赖关系。RBAC在工作流环境下的主要构成要素是任务、角色、用户、权限、会话和约束．角色是一个组织概念，它可以表示职务、岗位和职责等．任务既分配给角色又分配给用户，授权执行任务的用户如果获得了执行任务所需要的角色，则其具有执行相应任务的资格，只有当会话激活任务和执行该任务所需要的角色和用户时，用户才能使用该角色包含的权限执行任务。

3 基于RBAC的MSTRBAC模型

RBAC是一种适合信息系统中使用的访问控制模型，其授权简单、灵活，且易于建立在关系数据库系统中。但由于现实工作通常还依赖于时间和地点，授权时还需要进行安全检查，因此基本RBAC系统不能完全解决各种授权问题，参考其他上述2中其他研究者对RBAC的一些扩展思路，笔者提出一种引入了强制性概念、时间约束和空间约束的角色访问控制模型：MSTRBAC（Mandatory Spatio-Temporal Role Based Access Control，强制性的基于时空约束与角色的访问控制）模型。此模型不仅具有基本RBAC中的授权灵活和实现简单的优点，而且可以强制授权的安全检查和防止非法用户在非法时间非法区域访问系统资源，实用性强。

3.1 MSTRBAC模型基本概念

MSTRBAC的权限授权实际上是God、Who、When、Where、What、How的问题。即“God决定Who在When时间Where地点对What进行How的操作”。

God:最高权限管理员，只做规则集的初始化工作；

Who:权限的拥用者或主体(如User、Group等)；

When:权限的开放(停用)时间约束(参考2.2节所述，如9:00～17:00等)；

Where:权限的开放（停用）空间约束（如某IP地址段可得到授权访问权限）；

What:权限针对的(某类)对象或资源(Resource、Class，如数据库中的某个基本表)；

How:具体的权限（Privilege，如数据库中的增、删、改等操作或一个工作流对应的操作集）；

Operator:操作，在When时间Where地点对What进行How的操作，也就是Time+Space+Privilege+Resource；

Role:角色，一定数量的权限的集合，权限分配的单位与载体,可以继承，目的是隔离User与Privilege的逻辑关系；

规则:实现事件分析、推理的基础。授权管理的规则主要有3种：确定用户权限；解决授权冲突；解决权限支配问题[7]。

从上述概念可知，MSTRBAC不仅具有RBAC安全模型的特点，而且把授权规则、权限时间约束和空间约束均考虑进授权活动中，具有授权活动安全性更完备的特点。

3.2 MSTRBAC模型定义

由于MSTRBAC模型是在基本RBAC模型扩展而来，因此下面给出其补充与扩展的定义：

定义3．补充符号定义

T={(ts,te) | ts,te为系统有效时刻,ts≤te}:系统时间约束集。

S={s |有效空间地址,可采用IP地址或硬码,0为任意地址}:系统空间约束集。

$RTS=\{(r,t,s)|r\in R\wedge t\in T\wedge s\in S\}$:带时空约束的角色集。

$\mathrm{URTS}=\{(u,(r,t,s))|u\in U,(r,t,s)\in RTS\}$:带有时空约束的用户角色集。

定义4．扩展操作定义

$(r,t,s)/u_i|\rightarrow u_j(i\neq j\wedge(u_i,(r,t,s))\in URTS)$:用户ui申请将具有时空约束的角色r授予用户$u_j$，前提是用户具有角色r且不自我授权。

$(r,t,s)/u_i|\leftrightarrow u_j((u_j,(r,t,s))\notin URTS)\wedge(i\neq j\wedge(u_i,(r,t,s))\in URTS))$：用

户 u_i 将具有时空约束的角色 r 授予用户 u_j 操作成功，前提是用户 u_i 可以授权 u_j 角色 r，且目前还未授予 u_j 角色 r。

$(r,s,t)/u_i|\leftarrow u_j((r,s,t)/u_i|\leftrightarrow u_j \wedge (u_i,(r,s,t)\in URTS)$：用户 u_i 回收授予用户 u_j 的具有时空约束的角色 r，前提是 u_j 具有角色 r，且 u_i 是用户 u_j 的角色 r 主授方。

从上述定义可知，MSTRBAC 的授权操作与 RBAC 的授权一样，实现简单，授权灵活。

3.3 MSTRBAC 模型实现

由系统开发人员在开发系统时设定策略与规则集(如上述各定义)，作为整个系统的约束条件，这是强制性的，不能由用户更改的。规定具体的使用对象或资源，如某些数据库基本表、视图、设备。作为规则集的维护接口只提供给系统开发人员，不提供给用户。另外还需建立一个具备完整权限的角色和授予该角色的超级用户，并由该超级用户依据实际情况创建若干其他用户、时间约束条件、空间约束条件、角色等，完成访问控制的授权任务。如图 1 所示：

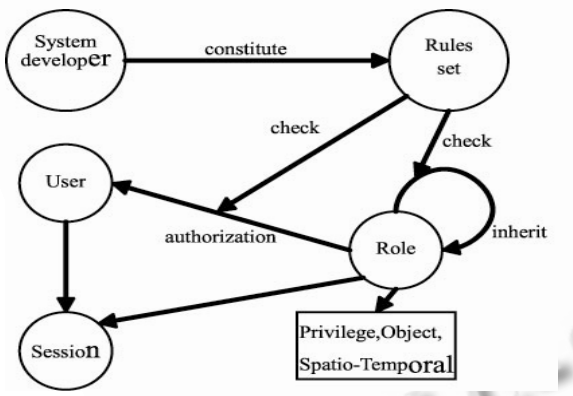

图 1　MSTRBAC 模型

4　实例验证

4.1 相关数据结构

目前正在开发的韶关市交通局电子政务网上行政审批与许可系统使用了 MSTRBAC 模型作为其安全模型。此模型需要在关系数据库系统中建立以下数据结构(简化基本表)：

(1)权限颗粒表(id，权限代号，说明，函数集)；

(2)对象资源表(id，对象代号，说明，属性集，方法集，事件集)；

(3)时间条件约束表(id，时间约束条件名，类型，时间段集)；

(4)空间条件约束表(id，空间约束条件名，类型，IP 地址段集)；

(5)操作表(id，权限 id，对象资源 id，时间条件约束 id，空间条件约束 id)；

(6)角色表(id，角色名，操作集)；

(7)用户表(id，用户名，密码，数字证书指针，角色集)；

4.2 常用操作实现示例

(1)创建”OURGROUP”角色：INSERT INTO 角色表 SET id= 1，角色名=’OURGROUP’,操作集=’’；

(2)创建”Me”用户，并授予 OURGROUP 角色：INSERT INTO 用户表 SET id= 2，用户名=’Me’，数字证书指针=NULL,角色集=’OURGROUP’；

(3)创建”签字”权限颗粒：INSERT INTO 权限颗粒表 SET id=3，权限代号=’signature’，说明=’签字同意’，函数集=’signature()’；

(4)创建”许可营运”对象资源：INSERT INTO 对象资源表 SET id=4，对象代号=’ permission’，说明=’许可营运’，函数集=’’，方法集=’’,事件集=’’；

(5)创建”上班”时间约束：INSERT INTO 时间条件约束表 SET id=5，时间约束条件名=’上班时间’，类型=’允许’，时间段集=’8:30~12:00&14:30~17:30’；

(6)创建”工作用机”空间：INSERT INTO 空间约束条件名 SET id=6，空间约束条件名=’工作用机’，类型=’允许’，IP 地址段集=’192.168.1.8~192.168.1.16’；

(7)创建签字许可操作：INSERT INTO 操作表 SET id=7，权限 id=3，对象资源 id=4，时间条件约束 id=5，空间条件约束 id=6；

(8)授予”OURGROUP”角色签字许可操作：UPDATE 角色表 SET 操作集=’7’ WHERE id=1；

(9)创建授权情况视图:CREATE VIEW Authorization AS(SELECT * FROM 权限颗粒表,对象资源表,时间条件约束表,空间条件约束表,操作表,角色表,用户表 WHERE 权限颗粒表.id=操作表. 权限 id AND 对象资源表.id=操作表.对象资源 id AND 时间条件约束表.id=操作表.时间条件约束 id AND 空间条件

(下转第 52 页)

(上接第 79 页)

约束表.id=操作表.空间条件约束 id AND 操作表.id=角色表.操作集 AND 角色表 id=用户表.角色集;

(10)验证函数伪代码:

```
Boolean verification(userid,privilegeid,resourceid,
timeid,spaceid){
    select count(*) as temp where
userid=Authorizati on.用户表.id and
privilegeid=权限颗粒表.id and resourceid=对象
资源表.id and timeid=时间条件约束表.id and
spaceid=空间条件约束表.id;
    return temp
}
```

5 结束语

经实例验证，MSTRBAC 模型在基础 RBAC 中引入了强制性、时间约束、空间约束的管理方法，完善了基础 RBAC，具有实现简单，安全性完备，授权灵活的特点。

参考文献

1 Sandhu R,Coyne E J. Role-Based Access Control Models. IEEE Computer, 1996, 29(2):38－47.

2 Sandhu R，Ferraiolo D，Kuhn R. NIST Model for Role-Based Access Control：Towards a Unified Standard. Proceedings of 5th ACM Workshop On Role Based Access Control. New York：ACM Press，2000:47－63.

3 Ferraiolo DF，Sandhu R，Kuhn R，et al. Proposed NIST Standard for Role-Based Access Control. ACM Transactions on Information and System Security, 2001,4(3):224－249.

4 Bertino E, Bonatti PA, Ferrai E. TRBAC: A temporal role-based access control model. ACM Transactions on Information and System Security, 2001,4(3):191－223.

5 黄建,卿斯汉,温红子.带时间特性的角色访问控制.软件学报,2003,14(11):1944－1954.

6 王小明,赵宗涛,郝克刚.工作流系统带权角色与周期时间访问控制模型.软件学报，2003,14(11):1841－1848.

7 苗雪兰.一种基于角色的授权管理安全模型的研究与实现.计算机工程, 2002, 28(9):96－161.