角色访问控制模型的研究及应用

张晓群,董丽丽
(西安建筑科技大学,陕西 西安 710055)

摘 要:针对大型企业信息系统在访问控制和安全管理方面的复杂性,传统的访问控制策略不适应大型企业信息系统在安全方面的要求。对角色的访问控制(RBAC)模型进行详细的分析,针对 RBAC 的不足提出改进的 IRBAC 模型,并将它应用到企业信息系统的设计中,建立企业的安全访问控制策略。采用 IRBAC 模型的访问控制策略简化了角色层次结构,方便了角色授权。

关键词:角色;角色层次;基于角色访问控制
中图分类号:TP393.08 **文献标识码**:A **文章编号**:1673-629X(2007)02-0042-04

Study and Application on Model of Role Based Access Control

ZHANG Xiao-qun, DONG Li-li
(Xi'an University of Architecture and Technology, Xi'an 710055, China)

Abstract: Aiming at the complication of access control and information management on enterprise information system, the traditional access control policy doesn't adapt secure requirements of it. On analyzing the role based access control (RBAC) model, a new improved role based access control (IRBAC) is proposed and is applied in designing of enterprise information system. The policy of access control is built for enterprise information system. The application of IRBAC model makes the structure of role hierarchical to be simplified. It is convenient for role authorization.

Key words: role; role hierarchical; role based access control

0 引 言

伴随着企业信息化的不断发展,为员工、合作伙伴和客户提供应用程序、信息和业务流程的集成视图成为企业日益迫切的需求。但为了保证信息和资源的访问的安全,对其加以控制是必不可少的。访问控制机制能够保证网络资源受控、合法地使用,使用户只能根据自己的权限大小来访问系统资源。

1 访问控制策略比较

目前,权限管理大致可以分为两类:一种是系统级的安全管理,如操作系统级的安全管理、数据库级的安全管理等;另一种是应用级的安全管理。应用级的安全访问控制策略主要有以下三种:自主访问控制(Mandatory Access Control, MAC)、强制访问控制(Discretionary Access Control, DAC)和基于角色的访问控制(Role-Based Access Control, RBAC)。

自主型的访问控制是在确认主体身份及所属组的基础上,根据访问者的身份和授权来决定访问模式,对访问进行限定的一种控制策略。主体访问者对访问的控制有一定权利。但是,正是由于这种权利使得信息在移动过程中,其访问权限关系会被改变。如用户 1 可以将其对客体目标 A 的访问权限传递给用户 2,从而使不具备对 A 访问权限的用户 2 也可以访问 A,这样做的结果是易于产生安全漏洞,所以自主访问控制的安全级别很低。

强制访问控制是指系统强制主体服从事先制定的访问控制策略。它是将主体和客体分级,预先定义用户的可信任级别及信息的敏感程度(安全级别),如可以分为绝密级、机密极、秘密级、无密级等,然后根据主体和客体的级别标记来决定访问模式。当用户提出访问请求时,系统对两者进行比较以确定访问是否合法。其缺点在于主体访问级别和客体安全级别的划分与现实要求无法一致,在同级别间缺乏控制机制,管理不便。

因此,自主访问控制和强制访问控制这种传统的

收稿日期:2006-05-10
基金项目:陕西省自然科学基金(2001X30)
作者简介:张晓群(1965-),女,陕西西安人,工程师,研究方向为分布式系统及计算机网络应用;董丽丽,副教授,研究方向为分布式系统及计算机网络应用。

访问控制策略,已不能满足基于 Intranet 的信息访问安全控制的要求。

基于角色的访问控制是美国 NIST(National Institute of Standards and Technology)于 20 世纪 90 年代初提出的一种新的访问控制技术[1]。该技术主要研究将用户划分成与其在组织结构相一致的角色,以减少授权管理的复杂性,降低管理开销和为管理员提供一个比较好的实现复杂安全策略的环境[2]。在基于角色的访问控制中,访问者的权限在访问过程中是变化的。有一组用户集和权限集,在特定的环境里,某一用户被分派一定的权限来访问网络资源;在另外一种环境里,这个用户又可以被分派不同的权限来访问另外的网络资源。这种方式更便于授权管理、角色划分、职责分担、目标分级和赋予最小特权,也是访问控制发展的趋势。

该文通过对 RBAC 模型的分析和研究,并针对 RBAC 模型的不足提出改进的 IRBAC 模型。结合实际应用系统的需求,提出将 IRBAC 思想应用到基于 Intranet 企业信息系统的权限控制中,实现整个系统对信息和资源的访问是按照授予用户的角色权限来进行的权限管理策略。

2　RBAC 模型

RBAC 的基本模型如图 1 所示。模型是由 User, Role, Permission 和 Session 四个组成部分。

2.1　RBAC 基本模型

RBAC 的基本思想是根据组织视图中的不同职能岗位划分角色,访问许可映射在角色上,用户被分配给角色,并通过会话激活角色集,能够间接访问信息资源。用户与角色以及操作许可与角色是多对多的关系,因此一个用户可以分配多个角色,一个角色可以拥有多个用户。同理,一个操作许可可以分配多个角色,一个角色可以赋予多个操作许可。角色可以划分等级,即角色的层次结构,反映企业组织的结构和人员责权的分配,并且通过继承形成新的角色,角色的继承是一种偏序关系,即满足非自反、反对称和传递性。

RBAC 模型描述如下:

—USERS, ROLES, OPS 和 OBS。

—UA⊆USERS × ROLES 是用户集合到角色集合的多对多的关系。

—assigned_users(r:ROLES) →2^{USERS}为角色 r 在用户集合上的映射。

一般形式:assigned_users(r) = {u∈USERS|(U,R)∈UA}

—PRMS = $2^{(OPS\times OBS)}$是操作许可的集合。

—PA⊆PRMS × ROLES 是操作许可集合到角色集合的多对多分配关系。

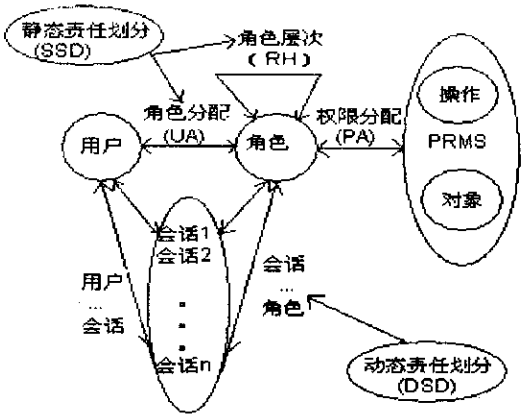

图 1　RBAC 基本模型

—assigned_permissions(r:ROLES) →2^{PRMS}是角色 r 在操作许可集合上的映射。

一般形式:assigned_permissions(r) = {p∈PRMS|(P,R)∈PA}

—Op(p:PRMS) → {bp⊆OPS}是许可 p 在操作集合上的映射,它给出了与许可 p 相关的操作集合。

—Ob(p:PRMS) → {bB⊆OBS}是许可 p 在客体集合上的映射,它给出了与许可 p 相关的客体的集合。

—SESSIONS 会话的集合。

—user_sessions(u:USERS) →$2^{SESSIONS}$是用户 u 在会话集合上的映射。

—session_roles(s:SESSIONS) →2^{ROLES}是会话 s 在角色集合上的映射。

一般形式:session_roles(s_i) ⊆ {r ∈ ROLES | (session_users(s_i),r) ∈UA}。

—avail_session_perms(s:SESSIONS)→2^{PRMS}为在一次会话中,用户的有效操作许可。

$$\bigcup_{r\in session\_roles(s)} assigned\_permissions(r)$$

2.2　RBAC 模型的角色层次(Roles Hierarchical, RH)

角色层次关系是 RBAC 模型中一个非常重要的概念,为了提高效率,避免相同角色的重复设置,RBAC 采用角色层次关系的概念。角色层次关系反映了组织关系中的上下级关系,上级角色可以继承下级角色的属性和权限,也可以增加新的属性和权限。下级角色自然拥有上级角色的限制和成员。通过角色之间的继承关系,上级角色间接地拥有下级角色所定义的权限。

"角色继承"定义了这样一些角色,它们有自己的属性,但可能还继承其它角色的属性和拥有的权限[3]。角色 r_1 继承角色 r_2,这样,管理员在定义 r_1 时,可以只定义不同于角色 r_2 的属性和拥有的权限,避免了重复定义。下面给出一般角色层次的定义:

—RH⊆ROLES×ROLES 是角色继承集合 ROLES 上的一个偏序，记为“≥”。即：

若 $r_1 \geqslant r_2$，则：authorized_permission(r_2) ⊆ authorized_permission(r_1) ∧ authorized_users(r_1) ⊆ authorized_users(r_2)

—authorized_users(r:ROLES)→2^{USERS}，角色层次中的角色与用户集的映射关系。

—authorized_permission(r:ROLES)→2^{PRMS}，角色层次中的角色与权限集的映射关系。

在实际应用中，有些权限往往是属于角色自身的，它不能被别的角色继承。为了解决这个问题，RBAC 原模型采用了私有角色的概念。即当一个角色的某些权限不能被继承时，就为该角色设置一个私有角色，将这些权限分配给它的私有角色。这就使角色的数量迅速增长，增加了角色层次结构的复杂性[4,5]。

3 改进的 IRBAC 模型

从以上分析可知，RBAC 原模型中私有角色的设置增加了角色层次关系中角色权限分配的复杂性。文中提出将权限划分为私有权限和公有权限，以降低其复杂性。下面给出两种模型的比较。

图 2 为传统模型支持的解决私有权限问题的角色层次关系。从图中可以看出，为防止角色 P，T_1，T_3 和 T_4 的私有权限被继承，增加了角色 P″，T_1″，T_3″和 T_4″等私有角色。

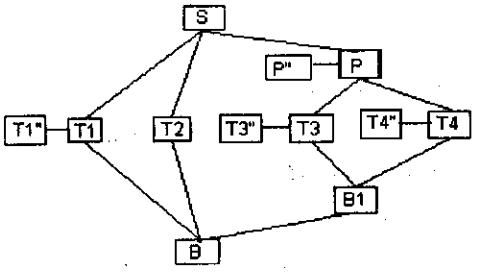

图 2 RBAC 模型的角色层次图

图 3 是完成同样功能的新模型角色层次关系。由于图 2 中的私有角色在图 3 中被私有权限所代替，所以不但新模型的角色数量减少了，结构简单了，而且与现实世界模型更贴近了。

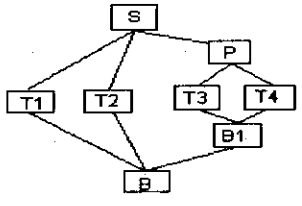

图 3 IRBAC 模型的角色层次图

现在给出 IRBAC 模型的定义。

3.1 权限划分

将授予角色的权限可分为：

(1) 私有权限(Private Role Permission，PRP)。PRP：R→2^{PRMS}，则 PRP(r)为角色 r 所具有的私有权限的集合。私有权限是一个角色自身所专用的权限，不能被别的角色继承。

(2) 公有权限(Public Role Permission，PBRP)。PBRP：R→2^{PRMS}，则 PBRP(r)为角色 r 所具有的公有权限的集合。公有权限能够被别的角色所继承，继承后，它是私有的还是公有的取决于被继承的方式。

3.2 权限继承方式

对于公有权限的继承有下列两种方式：

(1) 私有继承：公有权限经过私有继承后，转为私有权限，不能再被继承。

(2) 公有继承：公有权限经过公有继承后，仍为公有权限，可以再次被继承。

权限继承函数如下：

AddPInteritance(r_1，r_2，permission，type)；角色 r_1 从角色 r_2 处继承了权限 permission，作为自身的一个权限，其类型为 type。type∈{PRP，PBRP}

3.3 角色权限约束

(1) 权限自身没有私有性或公有性。当它被授予角色时，才确定它是私有权限还是公有权限。

(2) 同一个权限在一个角色中只能属于一种权限类别，不允许存在角色的某一种权限既是该角色的私有权限，又是该角色的公有权限，即：

$PRP(r) \cap PBRP(r) = \varnothing$

(3) 角色继承时，对于被继承角色中的同一种权限，只能采用一种方式继承。

4 基于 IRBAC 的应用系统的设计

现在用 IRBAC 的思想对某企业信息系统进行分析。该企业的组织管理分为多级管理，公司下属有几个厂部，每个厂部又分为多个不同的部门，每个部门设有不同的岗位和不同级别的用户。根据企业的组织结构图来划分应用系统的角色层次结构图。

4.1 应用系统的角色层次结构

运用 IRBAC 角色层次的概念，对企业的组织结构以及每个职能岗位的特点进行分析，划分角色，确定角色之间的继承关系、多继承或单继承关系以及角色互斥关系，生成该企业的角色层次结构树。图 4 所示为角色层次结构树中的一个分支(代表着一个部门的角色关系)。虽然企业的角色层次结构树是根据企业的组织结构而生成，但它又不完全等同于组织结构。根据具体部门的工作性质，将部门内工作的公共属性抽

象成部门基本角色。然后,分两步生成部门内各角色。首先,部门内各角色继承部门基本角色,这是一种单继承关系。既可以通过私有化继承函数将该角色权限作为本角色的私有权限,也可以通过公有化继承函数将该角色权限作为本角色的公有权限。然后,添加各角色的自身的属性和权限,这些权限既可以作为该角色的私有权限,也可以作为该角色的公有权限,这取决于该权限是否能被上级角色继承。生成部门负责人角色的方法也是通过继承和添加两种途径来完成。但部门负责人角色是从多个下级角色处继承角色权限,这种继承是一种多继承关系。

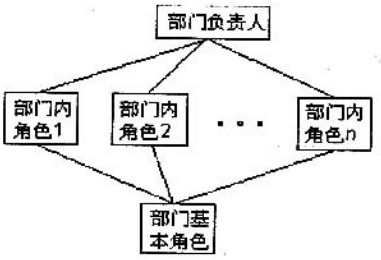

图 4　角色层次关系

部门与部门之间不存在继承与被继承关系,它们处于角色层次结构中的同一层上。整个一个部门也可以作为角色层次结构中的一个角色,能够从它的下级角色处继承属性和权限,也能够将自己的公有权限提供给它的上级角色去继承。

角色之间的存在着的上下级关系,反映了角色之间的管理和制约关系。上级角色继承下级角色的属性和限制,下级角色继承上级角色的用户成员。角色继承关系是一个偏序关系。就图 4 而言,部门负责人≥部门内角色 i≥部门内基本角色。

4.2　应用系统的系统结构

基于 IRBAC 的系统结构如图 5 所示。系统由访问控制器、IRBAC 访问请求过滤服务器、IRBAC 管理模块、用户/角色库、角色权限库等组成。系统流程及各部分的功能是:首先,用户端通过身份认证服务器获得系统访问令牌。然后,用户端通过合法的访问令牌向 IRBAC 访问请求过滤服务器提出应用服务访问请求,以获得用户的当前角色信息。IRBAC 访问控制器根据访问策略和角色权限库判断该用户是否有访问权限。如果有访问权限,将转到应用服务器,激活该角色的访问权限,并负责将应用服务器的执行结果返回给用户端。

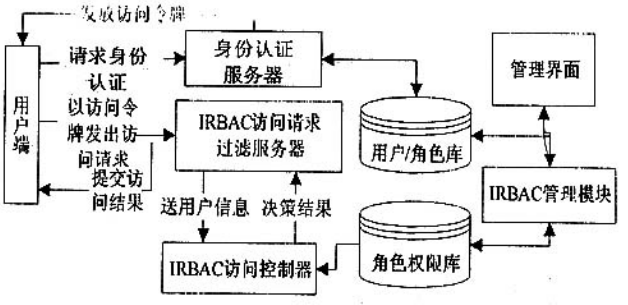

图 5　系统结构

用户/角色库存储系统管理员定义的用户集、角色集,以及用户_角色关系。角色权限库存放角色层次结构图,以及角色_权限关系。系统管理员对这两个信息库进行维护,例如,用户信息和角色信息的增加、删除、修改和查询等。当用户工作关系改变时,由管理员根据用户新的职责范围调整用户_角色关系。当企业的部门变动时,由管理员及时调整岗位角色关系,以适应岗位变动。这使得信息系统具有很强的灵活性。

5　结　论

在大型企业管理信息系统中,实行了 IRBAC 模型的安全访问控制后,使得信息系统的安全结构更能适应企业特定的安全策略,能简化用户授权,同时有效地防止非法用户的入侵以及已授权用户的越权操作,确保了大规模软件应用系统的安全性。由于权限的赋予是面向角色而不是面向用户的,所以信息系统能够适应企业内部机构的调整,具有较大的灵活性。

参考文献:

[1] Sandhu R, Coyne E J, Feinstein H L. Role-Based access control models[J]. IEEE Computers, 1996, 29(2): 38-47.

[2] Ferraiolo D F, Kuhn R. Role-Based access control[C/OL]. In Proceedings of the 15th National Computer Security Conference. Baltimore, MD, 1992: 554-563, http://hissa.ncsl.nist.gov/kuhn/.

[3] 黄益民,平玲娣,潘雪增,等.一种基于角色的访问控制扩展模型及其实现[J].计算机研究与发展,2003,40(10):1521-1527.

[4] 严　悍,张　宏,许满武,等.基于角色访问控制对象建模及实现[J].计算机学报,2000,23(10):1064-1071.

[5] 叶春晓,符云清,吴中福,等.RBAC 中权限扩展的实现[J].计算机工程,2005,31(9):141-172.