多域多应用环境下的访问控制研究

洪 帆 段素娟

(华中科技大学计算机科学与技术学院 武汉 430074)

摘 要 为适应多域多应用环境下的安全互操作的需求,本文通过扩展 RBAC96 模型有关概念,增加其对多域环境下多应用的刻画,通过引入全局角色、域角色和关联角色概念,提出了一种多域多应用访问控制模型 DPM。通过对 PMI 授权管理构架进行扩展,实现了 DPM 模型,为多域多应用环境下的安全互联提出了一个实际的解决方案。

关键词 PMI,访问控制,授权模型

Research on Access Control in Multi-domain & Multi-application Environment

HONG Fan DUAN Su-Juan

(College of Computer Science and Technology, HuaZhong University of Sci. & Tech., Wuhan 430074)

Abstract According to the security requirements in multi-domain & multi-application environment, an access control model called DPM (Distributed Privilege Model) is presented by adding application aspect into RBAC96 model and introducing global role, domain role and correlation role concept. Through extending PMI(Privilege Management Infrastructure), DPM model is realized. In conclusion, it proposes a practical scheme for the authorization and privilege management in multi-application and multi-domain context.

Keywords Privilege management infrastructure, Access control, Authorization model

1 研究背景

随着办公自动化的不断深入,政府部门各单位根据各自的业务需求建立了局域网并开发了各自的应用,采用基于角色的访问控制策略 RBAC[1] 进行管理。随着信息化的发展,这些局域网(可称为单域)之间实现互连和信息共享的需求越来越迫切。然而,原有自主可控的单域网络在与其它网络互连后,如何实现安全可控的开放并保持原有应用的安全,这是我们在信息化实施过程中要解决的关键问题。解决多域多应用网络中的安全互操作的主要难题在于:第一,每个域是自主的,有独立的用户群,对各自管理的服务有一定的自主度,而每个应用又都有自己的授权管理系统;第二,角色集、权限集等与具体应用紧密相关,而用户又分布在各个域中,用户角色的分配管理难度很大。

对于多域多应用的安全互操作,国内外研究并不多,文[2,3]引入域-域之间角色映射的方法,将一个域中的角色映射到另一个域中,从而实现了用户跨域访问的安全互操作,然而当域的规模和数量增大时,域间角色映射关系管理复杂;PMI 构架[4]可为大型单域网络提供统一的授权管理,不适用于多域多应用。

本研究采用集中与分散相结合的管理思想,扩展 RBAC96 模型中的有关概念,增加其对多域环境下多应用的刻画,通过引入全局角色、域角色和关联角色的概念,提出多域多应用访问控制模型 DPM,实现了统一构架下的多域多应用的授权管理。

2 多域多应用访问控制模型 DPM

本模型假设存在一个全局的组织结构,可以构建全局角色集。

2.1 定义

定义 1 D 为由单个管理机构管理的主机、用户、应用等的集合,称为域,GD 称为全局域,是所有域的集合。

定义 2 U 为全局用户集,R 为全局角色集,$UA \subseteq U \times R$ 为全局用户到全局角色的指派。

全局用户是人事部门定义的个人身份,全局角色是组织部门规定的职别、职级,如某部的部长、处长或科长。

定义 3 R_D 为 D 域的角色集,P_D 为 D 域的权限集 APP_D 为 D 域中所有应用的集合;$D.app$ 表示 D 域中的一个应用 app;$R_{D.app}$ 为 D 域中的应用 app 的角色集;$P_{D.app}$ 为 D 域中的应用 app 的权限集。

D 域的角色集 R_D 是该域中所有应用的角色集的并集,即:

$$R_D = \bigcup_{app \in APPD} R_{D.app}$$

D 域的权限集 P_D 是该域中所有应用的权限集的并集,即:

$$PD = \bigcup_{app \in APPD} P_{D.app}$$

定义 4 $PA_D \subseteq \bigcup_{app \in APPD} R_{D.app} \times P_{D.app}$ 是 D 域上的角色到权限的多对多分配关系,是各应用的角色到权限分配关系的并集。

定义 5 $RR \subseteq R \times R_D$ 称为全局角色到 D 域角色的关联角色集,也称全局角色到局域角色的映射。若对于 $r \in R$,$r_D \in R_D$,有 $(r, r_D) \in RR$,则全局角色 r 能以 D 域中角色 r_D 的权限访问 D 域中的应用。

定义 6 $RH \subseteq R \times R$ 称为全局域 GD 的角色层次关系,记为 $\geqslant$。

对于任意 $r1, r2 \in R$,$r1 \geqslant r2$,当且仅当:

a. 对任意 $r_D \in R_D$,若 $(r2, r_D) \in RR$,则 $(r1, r_D) \in RR$;

b. 对任意 $u \in U$,若 $(u, r1) \in UA$,则 $(u, r2) \in UA$。

角色层次关系体现了权限的向上继承和用户成员的向下

洪 帆 教授,博导,研究方向为安全模型、密码学和网络安全;段素娟 博士,研究方向为多域安全互操作、安全模型。

继承关系。

2.2 授权规则

约定 建立一个全局的身份认证机构 Cert_A 和全局的角色分配机构 Role_A，假定其合法性得到全局各域的认可。在各域建立一个授权代理，设 Di 域的代理为 $Agent_{Di}$。

规则 1 全局用户身份认证规则。由身份认证机构 Cert_A 建立全局用户集 U，并按如下规则判决全局用户 u 的身份。

若用户 $u \in U$，则身份认证约束条件 IS_valid_user(u)为 True，否则为 False。

规则 2 全局角色分配规则。由角色分配机构 Role_A 建立全局角色集 R，并按下列规则和步骤为全局用户 u 分配全局角色。

假设 Role_GD(u)为用户 u 所分配的全局角色的集合。

a. 对于任意用户 $u \in U$，令 Role_GD(u)=Φ；

b. 若 $r \in R$ 是组织部门为用户 u 规定的组织角色，即(u, r)$\in UA$，则 Role_GD(u)= Role_GD(u)$\cup\{r\}$；

c. 对于 $r' \in R$，若 $r \geqslant r'$，Role_GD(u)= Role_GD(u) $\cup\{r'\}$。

规则 3 域内应用角色授权规则。Di 域中的各应用的权限管理由授权代理 $Agent_{Di}$ 集中管理，负责为该域中各应用角色分配相应的权限。

令 $r_{Di.app} \in R_{Di.app}$ 则，

Set_Priv($r_{Di.app}$)=$\{p_{Di.app} \mid \exists p_{Di.app}(r_{Di.app}, p_{Di.app}) \in PA_{Di}\}$是 Di 域中 app 应用角色 $r_{Di.app}$ 所获得的权限集合。

规则 4 全局域角色到某个局部 Di 域的角色映射规则。

对于任意 $r \in R$，用 $GDrole\_to\_RDrole(r)$ 表示全局角色 r 到 Di 域角色集 R_{Di} 的映射集合，形成过程如下：

a. 对于任意全局角色 $r \in R$，令 GDrole_to_RDrole(r)=Φ；

b. 授权代理 $Agent_{Di}$ 根据域内的安全策略，对于任意 $r_{Di} \in R_{Di}$，若允许全局角色 r 可以角色 r_{Di} 的权限访问 Di 域中的应用，则

GDrole_to_RDrole(r) = GDrole_to_RDrole(r) $\cup\{r_{Di}\}$；

c. 对于 $r'_{Di} \in R$，若 $r_{Di} \geqslant r_{Di}'$，GDrole_to_RDrole(r)= GDrole_to_RDrole(r)$\cup\{r'_{Di}\}$。

2.3 授权步骤

假设 Dj 域中的用户 u 欲访问 Di 域中的应用 app 的资源，$Agent_{Di}$ 通过以下步骤实现 u 在 Di 域中的授权访问。

授权步 1 $Agent_{Di}$ 向全局的身份认证机构 Cert_A 提交 u，按照规则 1，Cert_A 请求执行身份认证约束条件 IS_valid_user(u)，若为 True，即 u 为全局的合法用户，继续；否则中止授权。

授权步 2 $Agent_{Di}$ 向全局的角色分配机构 Role_A 请求获取 u 所有的全局角色集，按照规则 2，Role_A 提供 Role_GD(u)=$\{r \mid (u,r) \in UA\}$。

授权步 3 对于用户 u 的任意全局角色 $r \in$ Role_GD(u)，按照规则 4，$Agent_{Di}$ 求其在 Di 域中的关联角色集合 GDrole_to_RDrole(r)=$\{r_{Di} \mid (r, r_{Di}) \in RR\}$，若非空则继续，否则中止授权。

授权步 4 若 $r_{Di} \in R_{Di.app}$ 且 $P_{Di.app} \in Set\_Priv(r_{Di.app})$，则 Dj 中的用户 u 可访问 Di 域中的应用 app 的资源。

3 模型的实现

3.1 概述

通过建立 PKI＋扩展 PMI 架构，实现多域多应用访问控制模型 DPM。以 PKI 作为该模型中的全局身份认证机构 Cert_A，为全局用户签发用户身份证书 PKC；以 PMI 作为全局的角色分配机构 Role_A，为全局角色签发授权证书 AC；下挂代理授权服务器 PS 作为各域中的授权代理，为解决多域多应用的安全互联问题提出了一个实用的解决方案。

3.2 方案描述

扩展 PMI 构架由分级 AA(表示授权证书签发机构)和下挂的代理授权服务器 PS 组成。一个 AA 可由多个域共享，AA 下属的每个域下建立一个代理授权服务器，该服务器为本域的多个应用提供公共的授权管理服务，授权分配架构如图 1。

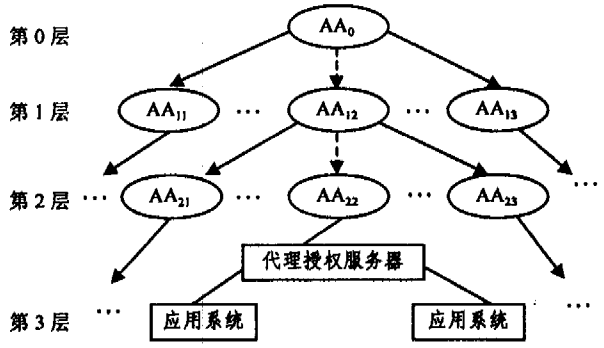

图 1 多域多应用环境下授权管理架构图

通过 PMI 下的 AA 分别为管理域范围的用户定义全局角色、可扩展的角色模板集及全局角色的授权证书(AC 证书)。代理授权服务器根据 PMI 提供的角色模板集，在角色模板的扩展项中为本域的具体应用定义域内应用角色，从而建立统一模板下具有个性特征的角色集，实现全局角色到具体应用角色的映射；同时，为本域应用定义权限集，定义域内角色-权限分配集，形成访问控制判决支持数据库，为应用系统的访问控制提供支持。当用户访问应用系统的资源时，提交 PKC 证书和全局角色 AC 证书，应用系统将其提交代理授权服务器。代理授权服务器提交 PKI 验证 PKC 的正确性，提交 PMI 验证 AC 的正确性，并根据 AC 中所描述的全局角色，在访问控制判决支持数据库中查找所映射的应用角色和相应的权限，并向应用系统给出是否允许用户访问相应资源的判决，应用系统根据判决响应用户的请求，从而实现了跨域多应用的访问控制。

图 1 的多级 AA 授权证书签发框架，旨在建立一个可信的分层式第三方公证网络结构。AA_0，$AA_{1i}(i \in N)$，$AA_{2j}(j \in N)$分别为第 0 层，第 1 层和第 2 层的授权证书签发机构。上述授权证书签发机构均已由 PKI 分配并签发了相应的公钥证书。我们仅以图中虚线箭头所指链路为例，给出各级 AA 签发的授权证书链。为了简化描述，首先定义一些符号，其中，第 i 层的某个 AA 用 AA_i 表示。约定以下描述中 $i \in [0,1,2,3]$。$PK^{(i)}$，$SK^{(i)}$，$ID^{(i)}$ 分别表示第 i 层某个 AA 的公钥、私钥和用户身份证书 ID。$PKC^{(i)}$ 表示 PKI 为 AA_i 签发的公钥证书，则 AC^i 表示 AA_{i-1} 为其从属的 AA_i 签发的授权证书，它是用 AA_{i-1} 的私钥 $SK^{(i-1)}$ 对 AA_{i-1} 的权限的签名。AC^{is} 表示 AA_{i-1} 为其从属的代理授权服务器 PS 签发的授权证书。

$X \| Y$：X 与 Y 的连接，表示 X 与 Y 的二进制连接。

1. 建立授权构架各组成部分的授权证书

(1)第 $i-1$ 层 AA_{i-1} 负责向第 i 层的 AA_i 签发授权证

书：{ $PKC^{(0)} \| PKC^{(1)} \cdots \| PKC^{(i)}$，$AC^{(i)}$ }。

(2)第 $i-1$ 层 AA_{i-1} 负责向它所辖服务区域内的代理授权服务器 PS 签发授权证书：{ $PKC^{(0)} \| PKC^{(1)} \cdots \| PKC^{(i)}$，$AC_{PS}{}^{(i)}$ }。

2. 建立用户授权证书　网络中域用户 u 的初始授权证书由所在域从属的 AA_i 签发，并发布到全网各个 AA：{ $PKC^{(0)} \| PKC^{(1)} \cdots \| PKC^{(i)}$，$AC_u$ }。

用户 u 的授权证书 ACu 包含以下内容：用户名，用户的全局角色模板 $R0$ 的值(包括系统、部门、单位、级别、职务，扩展项 R 等)，其中扩展项 R 为空值。

3. 角色权限注册　应用系统向本域(设为 D 域)代理授权服务器注册角色集 $R_{D.app}$、权限集 $P_{D.app}$ 及角色-权限分配集 $PA_{D.app}$ 等。

4. 代理授权服务器建立关联角色　在扩展项 R 中设置关联的应用角色，形成关联角色后提交到从属的 AA 签发，保存在代理授权服务器的访问控制判决支持数据库中。

5. 应用授权　代理授权服务器为 D 域中的各个应用 app 进行授权，建立 app 角色集 $R_{D.app}$ 到权限集 $P_{D.app}$ 的关联，即 $PA_{D.app}$。

6. 访问控制　App 根据 PS 提交的访问请求判决应答，决定是否响应用户 u 的访问，若 PS 提供的应答为 True，则 App 响应用户 u 的请求，否则拒绝。

4　模型分析及应用情况

该模型具有以下优点：(1)通过建立统一的授权框架，使多域环境下的多个应用可以进行集中和分散相结合的授权管理，不需要再为每个应用都建立一个授权系统，简化了授权管理，实现了多域多应用的安全互操作；(2)可扩展性强；(3)易维护，应用系统不用修改，只需调用代理授权服务器的功能；(4)通过对角色、权限等进行形式化描述，授权访问控制操作实现速度快。

该方案具有很强的实用性，目前已在全国性大型网络中得到实际应用，取得了很好的应用效果。

参 考 文 献

1　Sandhu R. Role based access control models. IEEE Computer，1996，29(2)：38～47

2　段素娟，洪帆，骆婷. 多域应用安全互操作的授权模型. 华中科技大学学报，2003，11

3　洪帆，黎成兵. 多域结盟环境下基于角色的访问控制. 计算机工程与科学，2004，11

4　Chadwick D，Otenko A. The PERMIS X. 509 Role Based Privilege Management Infrastructure. In：Proc. of SACMAT Conf. ACM Press，2002. 135～140

(上接第 235 页)

时延 Petri 网是对传统 Petri 网中的每个变迁 t 都关联了一对非负有理数 a_t 和 b_t，分别为变迁 t 的发生持续时间的最小值和最大值。当一个变迁使能时，立即发生，但其发生要占用一定的时间。

因为在时延 Petri 网中，变迁不再有瞬时语义，即其发生不再是瞬时的，为了同遵循瞬时语义的变迁相区别，在画图时，一般用实心矩形来表示这种变迁。

有关时延 Petri 网的其它定义和有关性质可参阅相关文献，笔者下面给出时延 Petri 网到 TPN 的转换方法。

4.2　时延 Petri 网到 TPN 的转换方法

按照图 3 给出的方法，转换对原网中的每个变迁都增加两个库所和一个立即变迁(即使能后立即发生的变迁)，因此，时延 Petri 网可以认为是只允许立即变迁为冲突变迁(如果有的话，如图 3 b)所示 t_1 和 t_2)的 TPN 子集。

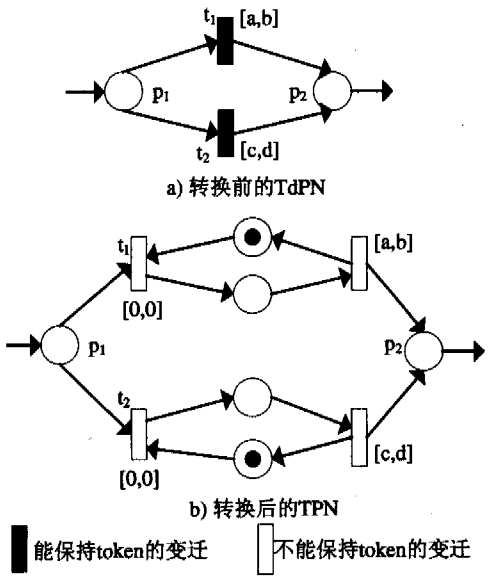

图 3　时延 Petri 网向 TPN 的转换方法

当然，把时延 Petri 网转换成时间 Petri 网后，网模型的规模会有一定的增长，为克服“状态爆炸”，这时可以利用一些时间 Petri 网的化简规则进行化简。有关 Petri 网的化简已经有很多成熟的理论和工具，这里不再详述，可参考有关文献。

结束语　模型的模拟能力一直是系统建模方面的一个重要研究课题。时间 Petri 网虽然结构较为简单，但本文首次证明了时间 Petri 网与图灵机具有同样的模拟能力。从时延 Petri 网到时间 Petri 网的转换方法为研究时延 Petri 网提供了另外一种途径，即通过结构较为简单的 TPN 来研究。时间 Petri 网的相关性质(如有界性、活性、可达性等)以及其它各种高级 Petri 网(如带抑制弧的 Petri 网、优先级 Petri 网等)向时间 Petri 网的转换方法是笔者准备进一步研究的课题内容。

参 考 文 献

1　Murata T. Petri nets – properties，analysis，and applications. Proceedings of IEEE，1989，77(11)：541～580

2　袁崇义. Petri 网原理. 北京：电子工业出版社，1998

3　Peterson J 著，吴哲辉译. Petri 网理论与系统模拟. 北京：中国矿业大学出版社，1989

4　Merlin P M，Farber D J. Recoverability of communication protocols – implications of a theoretical study. IEEE Transaction on Communications，1976，24(9)：1036～1049

5　Berthomieu B，Diaz M. Modeling and Verification of Time Dependent Systems Using Time Petri Nets. IEEE Transaction on Software Engineering，1991，17(3)：259～273

6　Berthomieu B，Menasche M. An Enumerative Approach for Analyzing Time Petri Nets. IEEE Transaction on Software Engineering，1983，17(3)：41～67

7　Popova L. On Time Petri Nets. J. Inform. Process. Cybern. 1991，27(4)：227～244

8　Hack M. Petri net languages，Technique Report，Computation structures Group，MIT，Project MAC，Memo 124，June 1975

9　Shepardson J，Sturgis H. Computability of Recursive Functions. Journal of the ACM，1963，10(2)：217～255

10　Ramchandani C. Analysis of asynchronous concurrent systems by timed Petri nets：[Technique Report，MAC-TR-120]. MIT，Cambridge MA，1974

11　Leveson N G. Safety Analysis using Petri Nets. IEEE Transaction on Software Engineering，1987，SE-13(3)：386～397