基于 RBAC 的扩展权限管理模型①

金　鑫[1,2]，王　晶[1,2]，李　炜[1,2]

[1](北京邮电大学网络与交换技术国家重点实验室，北京 100876)
[2](东信北邮信息技术有限公司，北京 100191)

摘　要：本文对基于角色的访问控制（Role-Based Access Control）模型进行了研究，并针对 Web 系统的特点和安全性等问题，提出了相应的设计原则。最后，对应具体项目，采用 FleaPHP 作为框架，设计并实现了一种专用的扩展权限管理模型。

关键词：RBAC；安全；权限管理；FleaPHP

Extended Privilege Management Model Based On RBAC

JIN Xin[1,2], WANG Jing[1,2], LI Wei[1,2]

[1](State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China)
[2](EBUPT Information Technology Co. Ltd., Beijing 100191, China)

Abstract: The article researches models based on RBAC, and then introduces several design principles, aiming to solve the security problems of a web system. Finally, the article designs an extended privilege management model in a real project using the structure of FleaPHP.

Key words: RBAC; Security; privilege management; FleaPHP

随着互联网的不断发展，网络安全问题显得日益重要。而基于 Web 的系统，由于资源的开放性和共享性，其系统安全也一直是急需解决的问题。经过人们长期的深入研究，提出了许多加密、认证机制。但随着系统规模和问题复杂度的不断增加，人们越来越意识到，任何单一的安全手段都存在一定的局限性，要想真正的解决问题，必须依靠多种手段的综合。

传统的访问控制策略包括：自主访问控制（DAC，Discretionary Access Control）、强制访问控制（MAC，Mandatory Access Control）等。但是它们都存在一些缺陷，如 DAC 安全性能较弱，缺乏防止“隐密通道”的能力，MAC 对于访问控制严格，实现工作量大，不适用于主体或者客体经常更新的应用环境。

在此基础上，20 世纪 90 年代 David Ferraiolo 和 Rick Kuhn 提出了首个 RBAC（Role-Based Access Control）模型[1]。在此之后，Ravi Sandhu 等人又提出了具有代表性和规范性的 RBAC96 模型[2]，由于该模型实现了用户和权限的分离，使得权限控制更加灵活，因而得到了广泛的应用。

本文设计了一种专用的扩展 RBAC 权限管理模型，并将其应用到实际中。该模型符合基本 RBAC 模型的规范标准，并采用基于用户行为[3]的方法对用户访问进行控制。这一过程中利用的设计原则和方法，对于其他 Web 系统，具有一定的借鉴意义。

1　RBAC模型介绍

1.1 基本思想

RBAC 的基本思想是在用户和权限之间引入角色的概念，将用户和角色联系起来，并通过对角色的授权来控制用户对资源的访问。RBAC 对访问权限的授

① 基金项目:国家自然科学基金(61072057,60902051);国家 973 计划项目(2012CB315802);中央高校基本科研业务费专项资金(BUPT2009RC0505);
国家科技重大专项(2011ZX03002-001-01.移动互联网总体架构研究)
收稿时间:2011-09-23;收到修改稿时间:2011-11-13

予由管理员统一管理，并授予给角色，而用户不直接与权限关联，系统将根据用户在组织内所拥有的角色来做出访问控制。

由于 RBAC 将用户与访问权限进行逻辑分离的特性，因此极大的方便了对于权限的管理。在实际工作环境中，由于角色与用户的对应关系一经确定就相对固定，而且指派工作不需要过多的技术，这一任务可由对业务流程较为熟悉的行政或管理人员担当。而将权限授予角色的工作较为复杂，需要一定技术能力，可由专门的开发人员进行分配，但开发人员无法将权限授予给具体用户，这与实际工作情况也是一致的。

在 NIST(The National Institute of Standards and Technology，美国国家标准与技术研究院)标准 RBAC 模型中，主要由四个部件模型组成，包括基本模型 RBAC0(Core RBAC)、角色继承模型 RBAC1(Hierarchal RBAC)、角色限制模型 RBAC2（Constraint RBAC）和统一组合模型 RBAC3（Combined RBAC）。RBAC1 和 RBAC2 都是建立在 RBAC0 模型基础之上的，而 RBAC3 是 RBAC1 和 RBAC2 的组合，RBAC3 模型对核心模型引入了角色继承和约束限制，是一个相对完整的模型。RBAC 模型之间的关系如图 1 所示：

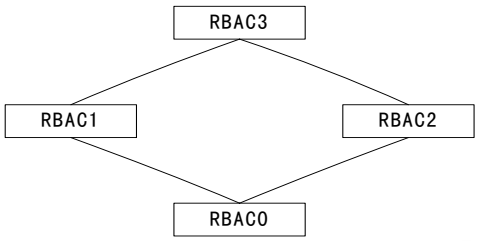

图 1 RBAC 模型家族关系图

1.2 模型的数据元素及关系

一个基本的 RBAC 模型如图 2 所示：

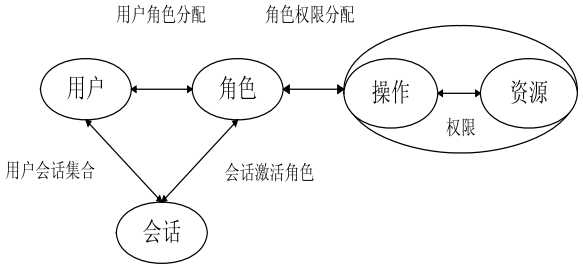

图 2 基本 RBAC 模型

可以看出，它主要由四个数据元素和两种关系组成：

四个数据元素包括：

1) 角色（Role）：一定数量的权限的集合，是粗粒度和细粒度的接口。角色是一个引入概念，是用户自身职能或者对资源操作能力的一个实体。角色可以是一个根据需要进行抽象的概念，也可以是具有某种实际含义的对应系统中某种功能的语义体。

2) 用户（User）：用户是访问或操作计算机信息系统中数据、界面、Web 页面菜单等资源实体的主体。用户可以是人、机器设备或者计算机软件程序服务等。

3) 权限[4]（Permission）：资源（Resource）是系统中我们需要访问控制的实体，如菜单、页面等；而操作（Operation）是主体实施在客体资源上的访问方式，如读写、修改、浏览等。权限就是操作资源实体上的许可，是执行操作的权利。权限由操作和资源组成，它是一个二元组（操作，资源）。用 Op 表示操作，用 Re 表示资源，用 P 表示权限，则 P=（Op，Re）。

4) 会话[4]（Session）：会话是每当用户进入系统时就会获得相应的角色集合，从而就建立了一次会话。在 RBAC 模型中用户是一个静态概念，而会话是动态的，每次会话就是由用户发起的。用户 U 与会话 S 之间是一对多的关系，即访问系统时，一个用户可能建立一个或多个会话，而一个会话只能关联到一个用户。而会话结束，相应的角色立即钝化，处于未激活状态。

另外，当用户数目较多时，为每位用户分别指派角色变成了非常繁琐的事情。此时可以引进用户组（Group）的概念，将角色指派给用户组，这样，组内的所有用户便具有用户组的所有角色权限，实现了批处理的功能。而且在实际生产环境中，用户组的概念可以与公司的组织结构相对应，具有很高的现实意义。

两种关系包括：

1) 角色权限分配 PA（Permission Assignment）：根据角色的职责为角色分配相应的权限。用户通过登录系统获取角色集间接地获取到系统中的访问权限。在分配权限时，通常遵循最小特权原则，这样既能保证在允许范围内充分行使特权又不会超出其本身应有的权限范围。

2) 用户角色分配 UA（User Assignment）：指在系统中为用户添加相应的角色，用户与角色之间通常是多对多的关系。用户角色分配通常根据用户在系统中

的实际职责来分配相应的角色。

2 扩展权限管理模型的设计

2.1 设计原则

依据基本 RBAC 模型，我们可知模型设计包括对其基本数据元素的设计，即角色设计、用户设计、权限设计和会话设计。对于每一类设计，都需要遵循一定的设计原则。

1) 角色设计原则：

角色分层原则：如果一个客体被授权访问一个角色，且该角色包含另一个角色，则该客体也被允许访问被包含的角色。

角色授权原则：角色具有互斥关系，具有互斥关系的角色不能被赋予给同一个用户。例如在银行中，一个人不能既赋予会计角色又赋予出纳角色。另外，一个客体在未经授权的情况下不能拥有某一个活跃的角色。

角色继承原则：角色是可以继承的，但是具有互斥关系的角色不能被同一角色继承。角色拥有被继承角色的所有权限，另外还可能包含一些被继承角色所不具有的权限。角色继承时，互斥关系也会被继承。

2) 用户设计原则：

一个用户只能对应一个执行者，因为如果一个执行者能够拥有多个用户身份，那么基于角色的权限管理就失去了意义[5]。

3) 权限设计原则：

由于权限是资源与操作的集合，因此对于不同的应用系统，权限的分配与设计也应有所区别。但就一般性而言，权限设计仍具有一定约束性可言：

权限具有前置关系，即权限 P1 可能需要具有权限 P2 才有效。

权限不具有包含关系，即对于任意权限 P1、P2，不应具有关系 P1 包含于 P2。

权限也不具有互斥关系。

在此约束关系的基础上，可以认为，权限的粒度取决于资源的粒度。例如在本系统中，资源的粒度包括“终端”，那么权限的粒度应该是针对“终端”展开的，如“终端管理”、“终端监控”等权限。

4) 会话设计原则：

在 RBAC 中引入会话机制，是为了实现最小权限原则。会话允许只激活赋予给用户的部分角色，如果没有会话，所有角色将处于活动状态，这可能会违反最小权限原则。考虑到通用性，以及为了满足角色的动态职责分离，会话应该设计成可以激活用户的一个或多个角色[6]。

2.2 总体架构

营业厅联播系统，简称 EBMS（Electronic Broadcast Management System），它定位于借助网络与系统平台，通过数字传输和视频播放等信息化技术，将营业厅电视、电子海报屏幕等各类电子显示设备进行联网，以远程管理的方式实现对各类电子屏幕统一内容、统一播放、统一监控的集中化管理，提升营业厅一体化、无纸化宣传管理水平和效果。

按照要求，系统功能应该包括审批待办事项、终端管理、内容管理、任务管理等，每一功能模块对应于一类角色，包括审批员、终端、内容、任务操作员等。每类角色由系统分配相应权限，并在权限范围内各司其职。在某一类角色中，根据实际情况，具体用户还可以在继承的基础上拥有不同的权限。例如在终端管理的功能模块中，可能具有终端管理员和普通终端操作员两类用户，终端管理员具有该模块下的所有权限，包括终端的建立、编辑、删除等，还可以向终端发送远程命令、监控终端状态等。而对于普通终端操作员，可能只具有终端的建立、编辑权限。其他功能模块的情况与此类似。在系统部署到具体省份时，可能还会有省、市级管理员、广告主、代理商等角色。这时，结构将更为复杂。由此可知，系统的权限组织已经不同于传统网站的管理员、用户二级平行结构，需要我们对 RBAC 模型进行扩展，以实现权限分配的简单性和灵活性。

根据系统的实际需求，我们设计了数据库 ER（Entity Relationship）图，如图 3 所示。

该模型严格遵循前一章节所述的设计原则，并支持群组（Group）的功能。每一功能实体（如 usergroup、operator、powerinfo、role）均支持子类的继承。在角色表 role 中，字段 sysroleId 用于区分系统角色类型，包括管理员、广告主、代理商等。系统角色是根据功能类型的不同进行区分的一类角色，可以认为是所有其他角色的超类。系统角色拥有的权限在设计初期时进行定义，且一经确定便不能更改，其权限配置保存在后台文件中。

服务器端我们采用基于 PHP 语言的框架 FleaPHP

进行开发工作。FleaPHP 采用 Passive MVC 模式，将系统分为三个层次，即 Model（模型）层、View（视图）层和 Controller（控制器）层。其中，Model 用于封装与业务逻辑有关的代码和数据，例如对订单数据进行各种运算。在 Passive MVC 模式中，Model 完全不知道自己身处于 MVC 结构之中。换句话说，Model 就是一个普通的对象，与 MVC 模式里面的其他组成部分完全没有关联。View 用于呈现内容给用户，也就是将程序运行的结果返回给浏览器。Controller 用于接收用户输入信息，然后调用 Model 对输入数据进行处理并获得处理结果。最后将结果传递到 View，从而让用户能够看到自己操作的结果。在具体设计时，按照纵向分割的思想，我们在 Model 层下方设计了 Table 层，用来实现与关系数据库的所有操作。

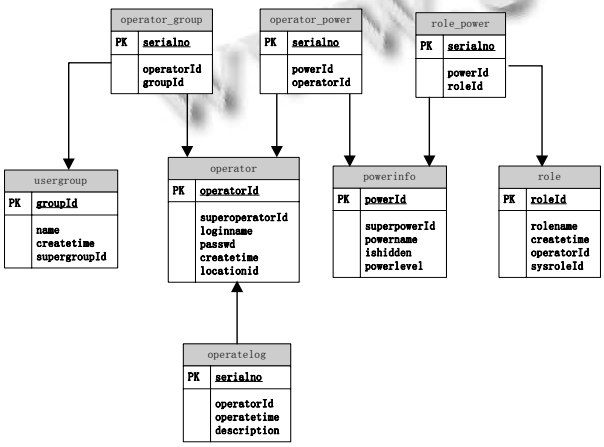

图 3 RBAC 模型对应的 ER 图

这样，借助 FleaPHP，服务器端权限访问控制的流程如图 4 所示：

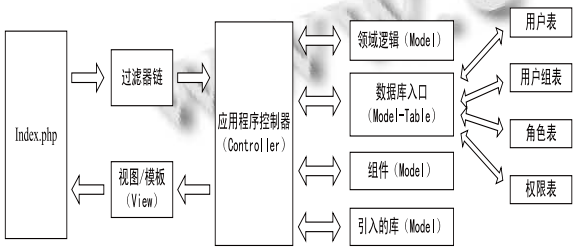

图 4 应用程序的执行流程

浏览器端，采用 ExtJS 对用户行为进行控制。ExtJS 由 JavaScript 编写，是一种用于创建前端用户界面，并与后台技术无关的前端 Ajax 框架。ExtJS 的特点在于强大的图形展现功能和对表格控件的使用，利用它构建的 Web 应用具有与桌面程序一样的标准用户界面和操作方式，并且能够横跨不同的浏览器平台。因此，ExtJS 已经成为开发具有完美用户体验的 Web 应用的首选。

3 扩展权限管理模型的实现

3.1 登录认证的实现

系统登录认证的流程图如图 5 所示：

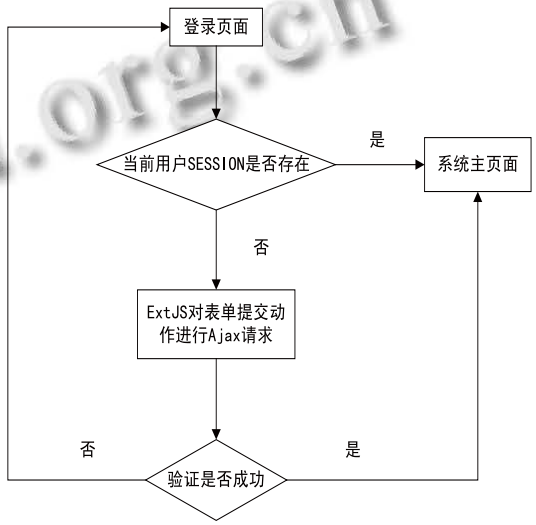

图 5 登录认证流程图

对于 ExtJS 进行 Ajax 请求的功能模块，服务器端依然采用 MVC 模式，其调用方式如图 6 所示：

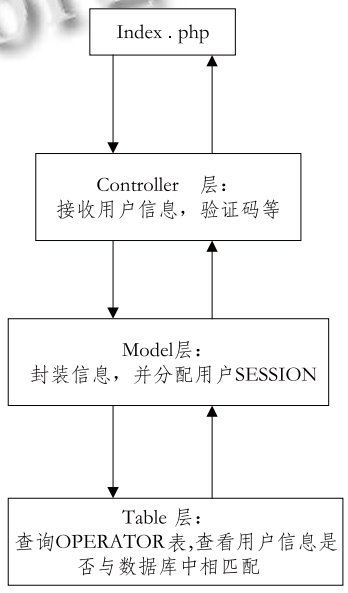

图 6 服务器端 Ajax 请求执行流程

对于图中尚未体现的 View 层，它会根据用户的具体角色信息，按照 Controller 层返回的参数，在 ExtJS 的控制下，对页面进行构造，并将最终内容返回给浏览器。例如名为“超级管理员”的用户，他拥有系统的所有权限，ExtJS 会根据 Controller 层返回的 Json 数据，将所有一、二、三级菜单和按钮呈现出来，从而实现权限的授予。而名为“Test”的用户，可能只拥有操作广告的权限，按照同样的方式，他的登录界面只能显示与广告相关的菜单和按钮。

3.2 角色管理的实现

根据 RBAC 模型的基本概念，角色与权限形成直接映射关系，因此角色管理的相关操作在 Table 层只针对角色表、权限表和角色-权限关联表。功能点包括角色列表、角色添加/编辑、权限修改、角色删除等。系统依照 MVC 的思想，自下而上，最终通过 View 层的构造将页面呈现给用户。各功能模块的调度关系如图 7 所示：

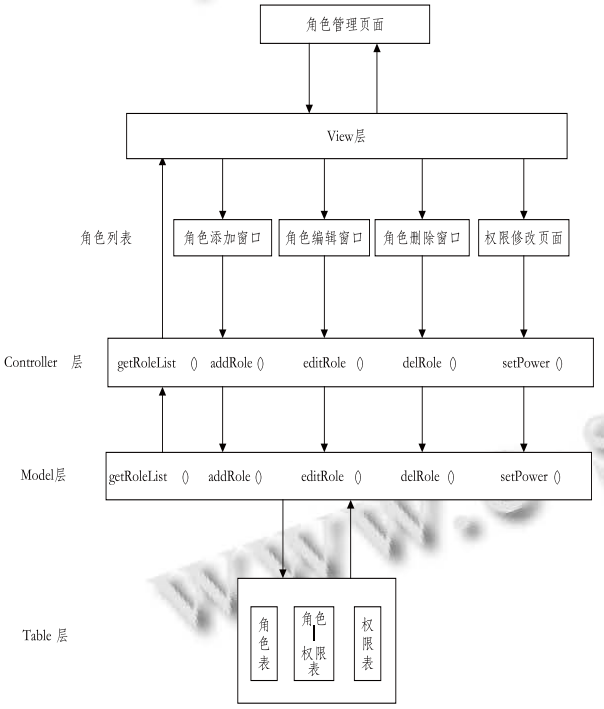

图 7　角色管理示意图

3.3 用户管理的实现

由于用户所属角色拥有的权限是该用户权限的最大集合，因此，用户权限无法超越其角色权限。这是通过在 Table 层联查用户、角色和权限表来进行控制的。属于同一角色的不同用户可以根据自身特点在权限集合内任意设置权限而不必相互一致，这大大增加了系统的灵活性，方便用户根据需求进行自定义设置。另外，用户管理功能还支持群组，便于日常管理。用户管理各功能模块的关系如图 8 所示：

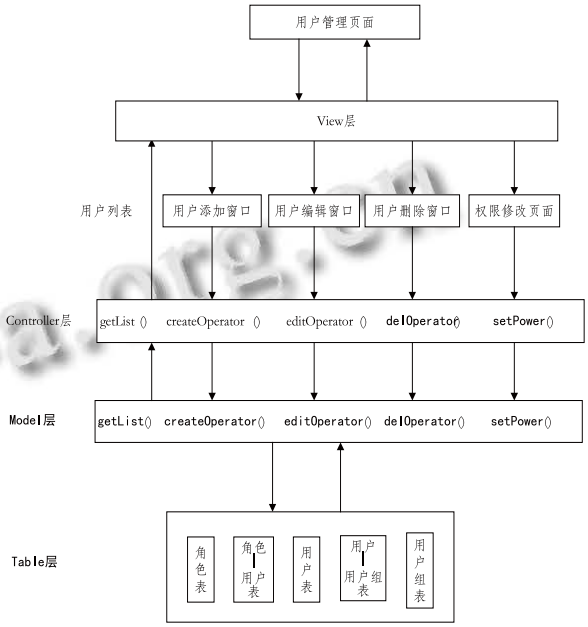

图 8　用户管理示意图

4　结语

本文介绍了 RBAC 模型的基本概念，模型数据元素及其关系，并提出了模型设计时应遵循的原则。在此基础上，根据项目的实际需要，提出了一种基于基本 RBAC 的扩展权限管理模型，并从数据库 ER 图设计、系统总体架构的角度对该模型进行了阐述。通过对该模型的设计，并借助优秀的 PHP 框架 FleaPHP，我们实现了对用户、角色和权限三者的良好控制，不但减小了授权管理的复杂性，增加了系统的安全性，而且在操作上，权限分配更加直观。基于 RBAC 的扩展权限管理模型，对于其他相似系统的使用，也具有一定借鉴价值。

参考文献

1 David F. Ferraiolo, Richard Kuhn D. Role–Based Access Controls.15th National Computer Security Conference (1992) Baltimore, Oct 13–16, 1992:554–563.

2 Ravi S. Sandhu, Edward J. Coynek, Hal L. Feinsteink, Charles E. Youmank. Role–Based Access Control Models.

(下转第 105 页)

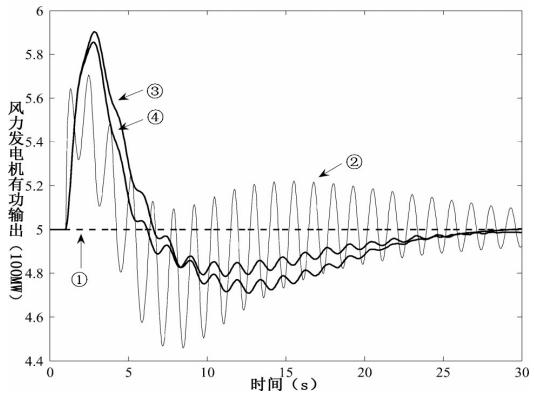

图 8 WECS 输出有功功率曲线

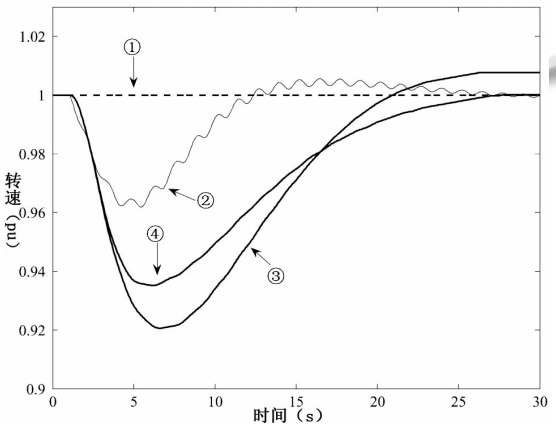

图 9 风力发电机转速曲线

4 结论

将传统惯性控制策略的直接用于 WECS 在参与频率调节时 WECS 出力很小，不能最大发挥地 WECS 频率调节能力。根据 WECS 的快速改变输出功率特性，提出改进策略，WECS 在频率波动开始时起到主要频率调节功能，通过协调信号使得传统机组尽快出力支持频率，弥补了 WECS 只能短时出力这一缺点。仿真研究表明，这一扬长避短的改进，使得 WECS 在频率暂态过程中表现很好，在暂态开始避免了惯性控制对 WECS 出力限制的缺点，能最大实现对频率的支持，通过协调控制使传统机组更早的参与调频，加速频率暂态结束，并且风电机组转速恢复时间、波动大小合理。相较于惯性控制改进策略大大减小了恢复过程中系统频率、WECS 有功功率输出和转速的震荡，对于电力系统和 WECS 运行都是很好的改进。

参考文献

1 Lalor G, Ritchie J, Rourke S, Flynn D, O'Malley M.J. Dynamic frequency control with increasing wind generation. USA: IEEE Power Eng. Soc. General Meeting, Jun. 6–10, 2004,2:1715–1720.

2 European Wind Energy Association (EWEA). Large Scale Integration of Wind Energy in the European Power Supply: Analysis, Issues and Recommendations. USA: EWEA, 2005.

3 Almeida RG, Lopes JA. Participation of doubly fed induction wind generators in system frequency regulation. USA:IEEE Trans. Power Syst. 2007,22(3):944–950.

4 Morren, S de Haan, Kling WL, Ferreira J. Wind turbines emulating inertia and supporting primary frequency control. IEEE Trans. Power Syst. 2006,21(1):433–437.

5 Lalor G., Mullane A, O'Malley M. Frequency control and wind turbine technolog-ies. IEEE Trans. Power Syst. 2005, 20(4):1905–1913.

6 Bevrani H., Ghosh A., Ledwich G. Renewable energy sources and frequency regulation: survey and new perspectives. IET Renewable Power Generation, 2010,438–457.

7 Almeida R, Castronuovo E, Lopes J. Optimum generation control in wind parks when carrying out system operator requests. IEEE Trans. Power Syst, 2006,21(2):718–725.

(上接第 24 页)

IEEE Computer,1996,29(2):38–47.

3 Ping Ni, Jianxin Liao, Chun Wang, Keyan Ren. Web information recommendation based on user behaviors. 2009 WRI World Congress on Computer Science and Information Engineering. 2009.3.31-2009:426–430.

4 汤象峰.基于 RBAC 的动态工作流系统的研究与应用[硕士学位论文],武汉:武汉理工大学, 2010.

5 Ferraiolo DF, Barkley J, Kuhn DR. A Role Based Access Control Model and Reference Implementation within a Corporate Intranet. ACM Trans. on Information Sys- tems Security, February 1999,1(2).

6 周志烽,王晶.基于 RBAC 的安全管理模块的设计与实现. 电信工程技术与标准化, 2010,(10):84–88.