

# 运用 LDAP 在 Web 上实现 RBAC 的一个方案\*

## A Scheme Using LDAP to Implement RBAC on the Web

欧阳星明, 赵 颢, 程 剑

OUYANG Xing-ming, ZHAO Hao, CHENG Jian

(华中科技大学计算机科学与技术学院, 湖北 武汉 430074)

(School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan 430074, China)

**摘 要:**本文在对基于角色的访问控制模型和轻量级目录访问协议 LDAP 进行研究的基础上,提出了一个在 Web 上运用 LDAP Server 做角色服务器来实现基于角色访问控制的方案。

**Abstract:** On the basis of studying the role-based access control model and the lightweight directory access protocol, the paper proposes a scheme to implement RBAC on the Web using LDAP Server which is used as the role server.

**关键词:** 访问控制; Web; RBAC; LDAP; LDAP Server

**Key words:** access control; Web; RBAC; LDAP; LDAP Server

**中图分类号:** TP309

**文献标识码:** A

## 1 引言

Sandu 等人提出的基于角色访问控制(RBAC)模型给出了基于角色的访问控制框架。该模型引入了角色这个中介,安全管理人员根据需要定义各种角色,并设置合适的访问权限,而用户根据其责任和资历再被指派为不同的角色。这样,整个访问控制过程就分成了两部分,即访问权限与角色相关联、角色再与用户相关联,从而实现了用户与访问权限的逻辑分离。正是鉴于此,基于角色的策略极大地方便了权限管理工作。

随着计算机技术、网络技术的发展,大型分布式基于角色访问控制系统得以应用开发。系统用户或应用程序每天要从服务器检索大量信息,而且每个应用程序可能还必须访问分散在不同地点的应用服务器。怎样向用户提供快捷高质量的服务是系统迫切要解决的问题,而目录服务则是一个很好的解决技术。

本文主要研究的是将基于角色的访问控制和目录服务运用到系统中来,构建一个安全的在 Web 上的实现方案。该方案只有经过身份认证的用户才能访问系统,以及只有经过授予一定权限的用户才能使用系统资源。目录服务的运用能对用户的请求作出快速响应。

## 2 相关技术

### 2.1 RBAC 模型

RBAC 模型的核心包括用户/组、角色、许可权限、会话和约束五个组件。用户 U 是与计算机系统资源的交互者。组是满足一定特征的多个用户的集合。角色 R 是系统的一组特定的功能集,它描述了授予该角色的用户的职责和权限。角色之间的关系(即角色层次 RH)是一个偏序关系,因此角色之间可以传递、继承。许可权限是作用在一个或多个对象上的一组特定的操作。会话 S 是一个角色实例根据许可权限对系统资源的一次操作。约束 C 是一组限制规则,它包括静态约束和动态约束。用户与角色的分派关系 UA、角色与权限的分派关系 PA 均为多对多的关系。RBAC 模型的描述如图 1 所示。

### 2.2 轻量级目录访问协议(LDAP)

LDAP 是一个运行在 TCP/IP 之上的应用层协议,是目录服务的前端访问协议,以 C/S 方式工作。LDAP 是目录客户机与目录服务器之间的通讯协议。当目录客户机需要某项服务时,客户就传送一个请求到服务器,服务器负责在目录中执行该请求必要的操作。执行完这些必要操作之

\* 收稿日期:2004-09-02;修订日期:2004-10-22

作者简介:欧阳星明(1950-),女,湖南宁乡人,教授,研究方向为基于网络的计算机应用技术、计算机控制技术与接口技术等;赵颢,硕士生,研究方向为计算机网络信息管理工程。

通讯地址:430074 湖北省武汉市华中科技大学计算机科学与技术学院;Tel:(027)87543343;E-mail:ouyangxingming@163.com  
Address: School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, Hubei 430074, P. R. China

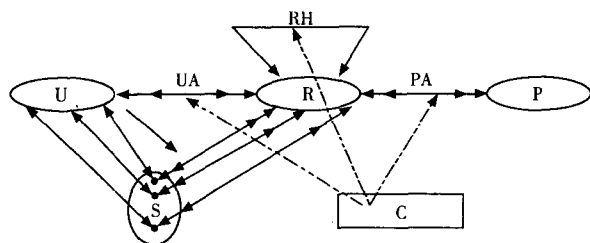


图1 RBAC模型

后,服务器向发出请求的客户返回一个响应,响应中包含了执行请求所获得的任何结果或错误。

LDAP中的数据是以目录树的形式存放的。树中的每一个节点称为目录项(Entry)。一个目录项由若干个属性构成。属性又通过一个属性类型和一个或多个属性值表示。根节点到每个目录项的路径称为DN(Distinguished Name,简称DN),它用来唯一标识树中的节点。具有相同属性的一组目录项组成一个对象类,对象类规定了目录项中的属性类型。对象类中的属性可以相互继承。LDAP中关于属性、对象类的语法定义构成模式(Schema),LDAP规范包括了一套标准的模式。用户也可以通过定义自己的schema(即模式扩展的方法)引入新的对象类和属性类型,模式扩展使得LDAP的体系结构变得灵活。

### 3 运用LDAP实现的RBAC

本文运用LDAP的模式扩展来定义用户和角色,为此必须做如下两个模式扩展。首先创建一个新的对象类并把角色定义为它的一个必须属性,然后把用户对象作为这个新的对象类的子类;其次再把角色也定义为一个新的对象类,在系统中每一个被定义的角色均为角色对象的实例。在实现上,每个用户可以由用户标识uid、用户名uname、密码password、角色role等属性组成。这样,用户/角色之间的多对多映射UA可以被建立起来。角色对象由角色标识roleid、角色名rolename及访问矩阵vmatrix等属性构成。可以用角色作为访问矩阵的行,角色对应的访问权限作为访问矩阵的列来实现角色/权限之间的分配PA,从而角色/权限之间的多对多映射关系也被构建。基于LDAP模式扩展构建的RBAC模型如图2所示。

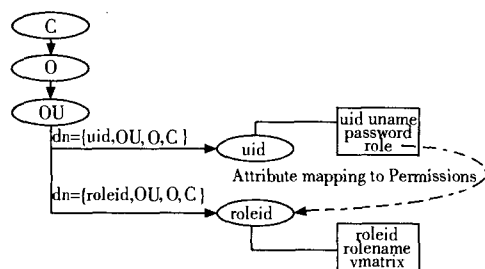


图2 基于LDAP模式扩展的RBAC

在图2所示的树形结构中,树的最高层顶点为国家C;第二层为组织O,它可以为一个单位或组织机构;第三层为组织单元OU,它可以为一个单位的部门或分部门;第四层节点在图中表示人或角色,它通过区别名dn来唯一标识。

基于LDAP数据模型构建的RBAC中,角色之间的层

次关系RH可以通过用户对象类的继承来实现,访问控制矩阵vmatrix可细粒度地控制角色对象所具有的操作权限,如对数据的创建、读、写、删除权限等。并且,这些只需安全管理员预先定义,LDAP目录服务器内嵌机制会自动对数据访问控制进行安全检查,这极大地方便了安全管理员的管理工作;同时,由于目录服务器其自身的特点,数据检索的吞吐率很高,处理时间很短,非常适合多用户、实时性要求很高的系统。

### 4 模型及访问过程

本方案用LDAP Server做角色服务器,用户通过认证服务器进行认证后再请求角色服务器获得角色后得以访问系统资源,其模型如图3所示。

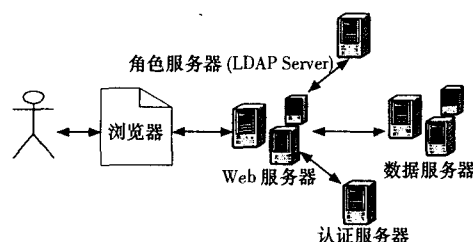


图3 系统模型

整个系统主要由以下几个部分组成:

(1) Web服务器。Web服务器用于处理用户(如工作站、移动PC、PDA等)通过浏览器发来的服务请求、认证请求、角色请求,同时也负责完成和客户端安全会话密钥的分发以及安全传输通道的建立。Web服务器是站内所有资源的唯一入口,安全管理员也是从Web服务器对系统进行管理,如用户分配角色,给角色赋予权限,管理角色之间的约束关系。

(2) 认证服务器。认证服务器有以下几个主要功能:它负责处理由各个Web服务器发来的验证用户身份的请求并验证用户身份的真实性;为系统管理员提供操作接口,这是因为新加入系统的用户必须首先注册才能使用系统提供的各种服务,因此认证服务器要为用户提供注册的服务接口。此外,认证服务器还必须提供用户注销、管理帐户资料等功能。

(3) 角色服务器。角色服务器的主要功能之一是处理Web服务器发来的角色请求,根据Web服务器发来的用户标识返回该用户的有效角色集。角色服务器也需要为安全管理员提供接口。

(4) 数据服务器。数据服务器的主要功能是处理来自Web服务器的数据请求,提供与后台数据库的交互接口,并运用一定的机制对数据进行预处理后转给Web服务器。

访问过程的UML序列图如图4所示。

用户访问系统的过程如下:

- (1) 用户发起请求,与Web服务器建立SSL会话连接。
- (2) Web服务器请求认证服务器,通过认证服务器进行用户与Web服务器身份相互认证。
- (3) Web服务器把认证结果返回给用户,若认证成功则转第(4)步,否则返回错误信息给用户。
- (4) 用户向Web服务器发送角色请求,角色请求中应包含用

(下转第19页)

```

<HTML>
<BODY>
...
<DIV><xsl:apply-templates select="通知集"/></DIV>
</BODY>
</HTML>
</xsl:template>
<xsl:template match="通知集">
<xsl:for-each select="通知">
...
</xsl:for-each>
</xsl:template>
</xsl:stylesheet>

```

### 4.3 通知与反馈的数据关联

每一条征求意见性通知都有一个相关联的反馈集合, 关联可通过设置消息属性实现。JMS 消息都有系统级 JMSMessageID 属性, 其值是唯一的, 可用于表征每一条征求意见性通知。因此, 对任何反馈消息也可以设置一个应用级属性(CWNF 中是 FeedbackSN), 让它取与之相关联的征求意见性通知的 JMSMessageID 属性值。这样, 就建立了两者的数据关联。

因此, 数据流模型“(3) 通知接收方到反馈接收方: XSL 显示(含表单)→主题(存储)”的实现流程如下: 用户在页面上选择一条征求意见性通知后, 该通知的 JMSMessageID 属性值将被传递给 FeedbackerSubServlet 组件, 该组件将使用这个属性值去匹配从主题取出的反馈消息的 FeedbackSN 属性, 从而筛选出相关联的反馈消息。那么一条征求意见性通知的 JMSMessageID 属性值又如何传递给 FeedbackerSubServlet 组件呢? 通过 ServletContext 对象只能传递可预知信息, CWNF 的做法是: 由 XSL 为每一条征求意见性通知设置一个独立的表单, 并把该通知的 JMSMessageID 属性值写在表单的 TEXTAREA 元素框内。这样, 用户在表单上选择一条征求意见性通知后, 该通知的 JMSMessageID 属性值就随表单一起提交给 FeedbackerSubServlet 组件。XSL 有关代码如下:

```

<xsl:if test="string(意见反馈)='on'">
<FORM method="post" action="http://localhost: 6888/
Feedbacker/servlet/FeedbackerSubServlet">
<BUTTON type="submit">意见反馈</BUTTON>
<TEXTAREA name="序列号" rows="1" cols="40">
<xsl:value-of select="序列号"/>
...
</xsl:if>

```

## 5 结束语

JMS 应用系统与数据库系统有相似性。从数据方面看, JMS 消息体的数据类型支持文本和对象, 所以 JMS 更灵活, 与 XML 集成应用的空间更大。从管理上看, JMS Provider 向管理员提供的管理功能远远低于 DBMS 提供的管理功能。因此, 在面向 Web 的应用中, JMS 宜作为中小流量、管理员参与度较低的信息系统解决方案。CWNF 教务系统经校园网实验性运行, 效果良好, 验证了面向 Web 的 JMS 应用是可行的。

### 参考文献:

- [1] Sun Microsystems, Inc. JMS Tutorial[EB/OL]. <http://java.sun.com/products/jms/tutorial/index.html>, 2004-01.

- [2] Mark Hapner, Rich Burrigde. 康博译. Java 消息服务 API 参考指南[M]. 北京: 清华大学出版社, 2002.
- [3] Sun Microsystems, Inc. Java Servlet API Specification-Version 2.1[EB/OL]. <http://java.sun.com/products/servlet/2.1/servletspec-2.1.zip>, 2004-01.
- [4] Jason Hunter, Brett McLaughlin. JDOM Beta9[EB/OL]. <http://www.jdom.org/dist/binary/jdom-b9.zip>, 2003-04.
- [5] Elliotte Rusty Harold. 刘文红, 赵伟明译. Java 语言与 XML 处理教程: SAX, DOM, JDOM 与 TrAX 指南[M]. 北京: 电子工业出版社, 2003.
- [6] 李江, 张威. 实例解析 XML/XSL/Java[M]. 北京: 希望电子出版社, 2002.

(上接第 2 页)

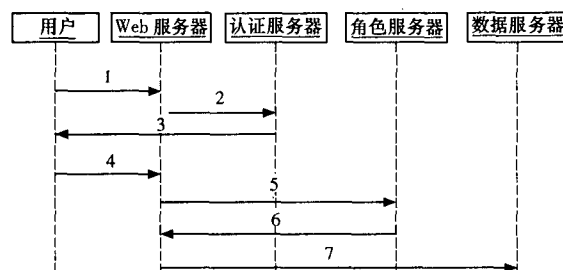


图 4 UML 访问序列图

户标识区别名 dn。

(5) Web 服务器把用户发来的角色请求转发给角色服务器 (LDAP Server)。

(6) 角色服务器根据用户标识把属于该用户的当前活动角色返回给 Web 服务器。

(7) Web 服务器根据获得的用户角色权限来比较用户的权限与访问请求资源所必需的权限。若用户的角色权限大于或等于请求资源所必需的权限, Web 服务器将用户的请求转给数据服务器处理, 否则返回“用户无权访问所需的资源”消息给用户。

## 5 结束语

随着当今分布式网络系统的研究与开发, 把基于角色的访问控制和目录服务结合起来开发基于 Web 的 MIS 系统, 必将给企业带来快捷安全的服务和高效的运作效率, 同时也必然会给企业带来很高的竞争活力。运用本文提出的模型开发的湖北清江电厂资产维护管理系统 (AMS) 已经正式投入运行, 并获得了同行的认可。

### 参考文献:

- [1] R S Sandhu, E J Coyne. Role-Based Access Control Models [J]. Computer, 1996, 29(2): 38-42.
- [2] Joon S Park, Ravi Sandhu, Gail-Jooh Ahn. Role-Based Access Control on the Web[A]. ACM Trans on Information and System Security(TISSEC)[C]. 2001.
- [3] W Yeong, T Howes, S Kille. Lightweight Directory Access Control Protocol[R]. RFC 1777, 1995
- [4] 郭军, 卢文龙, 赵明, 等. 基于 LDAPV3 目录服务系统中推荐的设计与实现[J]. 小型微型计算机系统, 2000, 21(8): 802-806.