基于J2EE平台的单点登录模块的设计

李幼红,梁京章
(广西大学 计算机与电子信息学院,广西 南宁 530004)

摘 要:客户在信息化的过程中拥有了多套不同厂商开发的J2EE应用。这样使得软件市场中开始出现较多的应用整合的需求,而用户安全模块的整合成为这当中很关键的一个问题。文中针对用户安全模块整合中的SSO(Single Sign-On)即单点登陆问题进行了分析,并给出了一个J2EE平台上通用的单点登陆的解决方案。整个解决方案基于一些规范的技术手段实现,不依赖于任务特定的技术平台,适用于大多数J2EE架构下的B/S应用。

关键词:J2EE;单点登陆;JAAS
中图分类号:TP393 **文献标识码**:A **文章编号**:1673-629X(2006)05-0232-02

Design of SSO Module Based on J2EE

LI You-hong, LIANG Jing-zhang
(School of Computer Science and Electronics Information, Guangxi University, Nanning 530004, China)

Abstract: J2EE is a set of coordinated specifications and practices for the solution of server-centric applications. J2EE applications have been provided with many clients. Some of them use several J2EE applications from their respective company. Thus, more and more demands for application integration is coming up. Therein SSO in the integration of user's security module is the key problem. To the question, this paper presents an analysis and a solution. The solution based on some technical means of specifications does not rely on the technical platform for specially appointed task. It applies to the B/S application in majority J2EE.

Key words: J2EE; SSO; JAAS

0 引 言

在信息化的过程中,有些客户拥有了多套不同厂商开发的J2EE应用[1]。他们往往希望做到使用者一次登陆即可使用所有跟他相关的系统,也就是这里所说的单点登陆。单点登陆[2]的含义是指:在一个企业级的信息系统环境下,一个使用者一次登陆,即可使用所有他有权限访问的应用系统。人们希望能达到以下效果:登陆一次,可使用多个应用系统;统一的身份认证;可配置的身份认证;各个应用系统之间共享用户信息,以方便系统间数据流转。

这里就会存在以下2个关键问题:

(1)登陆信息如何在多个应用之间传递;

(2)各个应用如何检查用户是否已经登陆。

下面将从设计思路入手,说明如何解决上述2个问题。

1 设计思路

先来看第一个问题,如何在多个应用之间传递登陆信息。在多个应用系统之间一一建立联系是难以实现的,为此则可以考虑建立一个中心的认证服务。各个应用在检测到尚未登陆之后,转去这个统一的认证服务,确认是否已经登陆,并获取登陆用户的身份信息。

第二个问题,各个应用如何检查用户是否已经登陆。各个Servlet和JSP都是通过URL来提供访问,当用户访问的时候,如果每个Servlet和JSP都去统一认证服务进行检查,开发的开销非常大。非常幸运的是,在Servlet的规范(版本2.3以上)中提供了一种组件形式——Filter,即针对HTTP访问的过滤器[3]。通过Filter机制,对于每一个应用系统,在一个集中的点进行身份认证的检查即可。

整个设计思路如下:

* 首先提供一个集中的中心认证服务器,通过这个认证服务来进行用户的身份认证。

* 各个应用系统在原有实现的基础之上,增加一个Filter来进行是否登陆的检测。

* 当访问某个应用系统时,Filter首先检查是否已经登陆到了当前系统。如果尚未登陆,则将请求转向认证服务器,检查是否已经在中心认证服务器上登陆;如果确认在中心认证服务器登陆了,则过滤器自动完成登陆;如果确认了尚未在中心认证服务器登陆,则提示用户输入相应的信息完成登陆即可。

收稿日期:2005-12-06
作者简介:李幼红(1978-),女,湖南常德人,硕士研究生,研究方向为远程教育、网络信息系统;梁京章,教授,研究方向为远程教育、网络信息安全、网络应用技术。

* 中心认证服务器可能面对不同的实现要求。认证服务实现可能采用LDAP Server,也可能是关系数据库。这要求认证服务的接口设计提供适应各种认证服务的能力,具备可扩展的结构。

下面来看看各个环节具体的设计。

2 认证服务的设计

对于认证服务(Authentication Service,AS),要关注两个方面:一个是如何设计一个统一的认证接口;另一个则是如何做到可扩展的认证结构。

2.1 统一认证身份

对于统一认证身份,先来看接口设计:

* 首先提供一个Identity的类,封装了各个应用系统最基本的登陆信息,包括了登陆名(loginName)、密码(password)等关键的认证信息,通过认证域(authDomain)属性说明了这个认证信息对应到哪个应用系统的认证模块。同时,在其中还包含了一个unifiedId,即统一认证的标识号,各个应用系统的身份认证信息通过这个标识字段关联在一起。

* 然后提供一个接口IdentityManager,负责管理各种认证信息和统一标识号之间的关联。这里仅仅描述接口,针对不同的认证服务,例如关系数据库或者LDAP Server,可提供相应的实现类。

* 另外一个是AuthHandler,负责执行认证的动作。这里都只是提供接口,针对实际不同的认证服务需求,则可以提供相应的实现类即可,而对于外部调用者,看到的接口是统一的。

2.2 可配置的认证服务

上文提到了要达到认证模块的可配置,所提到的设计已经在接口上满足了要求,但在体系结构上还不完整。而在Java平台上,已经有了这方面的规范,即JAAS(Java Authentication Authorization Service)[4]。下面先说明一下笔者如何利用JAAS所提供的结构。

* JAAS提供了一个LoginModule这样的接口,来封装对于不同的认证服务的实现。对于数据库和LDAP Server,只需提供对应的LoginModule就可以了。

* JAAS提供了一个可配置的、插件化的架构。通过配置相关的文件即可指定具体使用哪个LoginModule,或者组合使用多个LoginModule。

基于JAAS来设计认证模块,从体系结构上保证了整个模块实现上的可扩展性和规范性。

在建立起认证服务之后,接下来要考虑的就是如何实现客户应用的过滤器。

3 客户应用过滤器的设计

按照之前的设计思路,客户应用的过滤器主要完成两个职责,一个是检查是否登陆,并与认证服务进行交互;还有就是负责完成客户应用的自动登陆。针对这两个职责,笔者设计两个独立的Filter来满足相应的要求[5]:

a. ASFilter:认证服务过滤器。通过这个过滤器来检查是否登陆,如未登陆,则与认证服务进行交互,完成整个登陆过程。

b. CustomFilter:客户应用过滤器,通过这个过滤器完成客户应用的自动登陆。

下面来看看这两个过滤器如何工作。

①ASFilter:

* 当用户访问某个受到保护的URL资源的时候,ASFilter拦截到HTTP访问请求,并调用相关的认证服务的接口进行认证的动作。如果已经认证了,则不需要进行认证动作。

* 在ASFilter确保认证的动作完成之后,则把HTTP请求继续转发给CustomFilter,由它完成更进一步的动作,最主要就是自动执行客户应用的登陆动作。

②CustomFilter:

* CustomFilter拦截到ASFilter转发过来的请求之后,首先取得认证服务的IdentityManager接口的实现类,调用其getRelationalIdentity方法,取得对应的客户应用的身份标识。

* 然后使用所取得的身份标识自动完成客户应用的登陆动作,并设置好相关的登陆信息。此后其他Servlet或者JSP都按照原有的操作方式处理登陆信息即可。

完成上面这部分的设计,整个单点登陆的整个环节中的关键问题都有了明确的解决方案。

4 总结与展望

按照前述的分析和设计,给出了在一个J2EE平台上的通用的单点登陆的解决方案。对于绝大多数J2EE架构下的B/S应用,这个解决方案都是适用的,而且整个解决方案基于一些规范的技术手段实现,不依赖于任务特定的技术平台。随着J2EE应用的进一步发展,企业级客户所面临的应用整合的需求越来越多,这方面的方案也会更进一步的丰富和完善。

参考文献:

[1] 张能立,万　歆,李　曦.J2EE平台在企业管理系统中的应用[J].微机发展,2005,15(10):147-149.

[2] Upadhyay M,Marti R. Single Sign-on Using Kerberos in Java[EB/OL]. http://java.sun.com/j2se/1.4.2/docs/guide/security/jgss/single-signon.html,2001.

[3] Java Servlet规范2.3 . Java Servlet Specification2.3.2001 [EB/OL]. http://java.sun.com/products/servlet/ download.html#specs,2001.

[4] JAVA授权认证规范.JAAS(Java Authentication Authorization Service)[EB/OL]. http://java.sun.com/products/jaas/, 2001.

[5] JSP规范1.2. JavaServer Pages Specification 1.2.2001[EB/OL]. http://java.sun.com/products/jsp/index.jsp,2001.