轻量级门户单点登录服务机制

罗 辛 吴 晶 熊 璋 蒲菊华

(北京航空航天大学 计算机学院, 北京 100191)

摘 要: 信息门户的建设过程中需要容易实施且灵活高效的整合模式. 为克服传统单点登录机制无法满足在动态松耦合环境下实现快速整合的缺陷, 结合面向服务架构思想, 提出一种轻量级门户单点登录服务机制 (LSSO-Service, Lightweight Single Sign-on Service), 可为门户整合提供结构简单、完善通用、松散耦合、快速机动的单点登录服务. LSSO-Service基于高于对象层的分布式服务集成模式进行功能划分, 可实现采用不同技术的应用系统在门户中的快速动态整合. 阐述了 LSSO-Service的设计思想和工作原理, 并通过在国内某大型水利信息门户中的应用实践, 说明该研究结果对于门户建设具有较高的理论意义和参考价值.

关 键 词: 单点登录; 门户; 轻量级; 面向服务架构

中图分类号: TP 302.1

文献标识码: A **文 章 编 号**: 1001-5965(2008)06-0721-04

Lightweight single sign-on service mechanism for portal

Luo Xin Wu Jing Xiong Zhang Pu Juhua

(School of Computer Science and Technology, Beijing University of Aeronautics and Astronautics, Beijing 100191, China)

Abstract: An integration framework with easy and efficient implementation is necessary in construction process of information portal. To overcome the limitation that traditional single sign-on mechanisms can't achieve fast integration in dynamic and loose coupling environment, a novel lightweight single sign-on service mechanism for portal (LSSO-service) was proposed based on service oriented architecture. LSSO-service's business logic functions were divided according to a distribute integration framework with services higher than object layer. By this means, the simple, adaptive, loose-coupled and flexible single sign-on service could work well, and those application systems based on different techniques might be quickly integrated into portal with a unique entry. The design idea and functional principle of LSSO-service were both presented. To validate the high significance and feasibility of the proposed approach, an integrated prototype was carried out.

Key words: single sign-on (SSO); portal; lightweight; service oriented architecture (SOA)

信息门户 (information portal)[1]是为用户提供单一入口, 使用户能够按照个性化需求, 提取存储在企业或机构内部和外部的信息, 从而便于进行决策的应用程序, 它整合了一系列用于管理、分析、发布信息的应用系统 (AS, Application System). 其中, 提供单一入口是信息门户的前提, 即所谓的单点登录 (SSO, Single Sign-on)[2-3]: 用户只需在网络中主动进行一次身份认证, 随后便可以访问其被授权的所有网络资源, 而不需要再主动参与其它的身份认证过程.

目前, 以微软. Net Passport SSO机制和自由联盟 Liberty SSO机制为代表, 已相继推出了一些商业化、相对成熟的 SSO机制, 然而这些机制在实际应用中仍有不足: ①编程接口复杂, 开放度低, 整合时需对 AS进行大范围改造; ②对于采用不同技术的 AS往往存在可移植性问题; ③当门户由于 AS频繁变动而要求对 AS进行快速整合时, 现有 SSO机制由于自身的高复杂性难以满足

收稿日期: 2007-05-21

基金项目: "十一五"国家科技支撑计划重大项目(2006BAB04A13)

作者简介: 罗 辛(1983-), 男, 重庆人, 博士生, luoxin21@gmail.com.

需求[4-8]. 因而,迫切需要提供一种结构简单、快速机动、完善通用、松散耦合的 SSO 服务机制.

要设计出符合上述要求的 SSO 服务机制,必须选择一种协同高效、复用扩展、松散耦合的软件结构. 而面向服务架构(SOA, Service Oriented Architecture)[9-10]将应用程序封装为服务单元,并通过具备通用性的服务调用接口联系起来,强调技术无关性,能进一步抽象和标准化可复用资源;并且服务的接口定义独立于实现平台,具有松耦合、可复用和可扩展等特点,非常适合于作为本文所构想的 SSO 服务机制设计指导思想. 本文提出一种轻量级门户单点登录服务机制(LSSO-Service, Lightweight Single Sign-on Service)机制,将整个 SSO 流程分解成松耦合的功能单元并封装成一组认证服务,为待整合的 AS提供通用的调用接口,待整合的 AS通过这些接口调用服务执行 SSO流程,从而降低整合难度,提高门户灵活性. 下面将从该机制的设计思想、工作原理及其在某信息门户中的应用实例几方面进行详细阐述.

1 LSSO-Service基本结构

LSSO-Service是一种轻量级门户单点登录服务机制. 所谓轻量级,指该机制具备结构简单、快速机动、完善通用、松散耦合的特性. LSSO-Service 基本结构如图 1所示,主要包括嵌入 AS的认证代理和认证服务器两部分,其中认证服务器维护用户的身份信息和 SSO服务,认证代理通过认证服务器提供的接口调用 SSO服务进行单点登录.

1) 认证服务器. 认证服务器维护用户身份信息并提供 SSO服务,它向 AS提供两类 SSO相关的服务接口: ①服务查询接口. 认证代理调用服务查询接口在服务注册仓库中查询相应的认证服务调用接口;服务开发方更新用户身份认证服务后(图 1虚线所示)调用服务注册接口将更新的认证服务调用接口在服务注册仓库进行注册;认证代理通过服务查询接口获取最新的服务调用接口. ②认证服务调用接口. 认证代理通过这组接口调用用户身份认证服务,这组服务是结合 SOA 设计思想,对 SSO 流程进行分解、封装而成,其具体设计将在后文叙述.

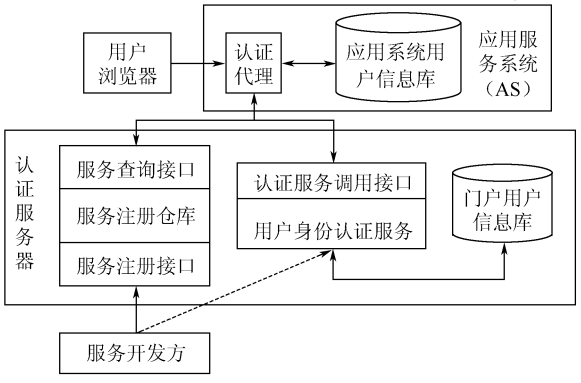


图 1 LSSO-Service机制总体结构

2) AS和认证代理. 在 LSSO-Service中, AS通过认证代理调用 SSO服务. 由于符合 SOA 思想的 LSSO-Service认证服务具备统一、通用的调用接口,故认证代理的工作仅限于结合 AS的具体实现方式将认证服务的调用接口转化为 AS可直接使用的形式. 对 AS而言, LSSO-Service仅意味着一组认证服务调用接口. 只要嵌入认证代理, AS就能通过这组接口使用 LSSO-Service认证服务,充分体现了 LSSO-Service快速机动的特点.

2 LSSO-Service认证服务设计

LSSO-Service中,认证服务器提供的 SSO服务是整个机制的核心. 其中权限管理、数据结构、安全策略和服务划分是 SSO服务设计的重点.

2.1 LSSO-Service权限管理策略

AS中用户权限的分配极其严格. 门户系统中集成的 AS在业务逻辑和工作流程上的差异通常会以最直观的方式反映到权限划分上. LSSO-Service以降低 AS整合难度和门户权限管理复杂度,提高软件灵活性和可复用性为目标,设计了以二次鉴权制度为主、角色映射管理制度为辅的 LSSO-Service权限管理策略. 二次鉴权制度是指将单个 AS视作一个门户资源,门户角色仅负责提供用户对于 AS的入口访问权,用户在 AS中的权限由 AS另行分配、管理. 角色映射管理制度是指在门户角色和 AS内部角色之间建立映射关系. 用户在访问 AS时根据其所持有的门户角色和角色映射之积,得出对应的 AS内部角色.

通过 LSSO-Service权限管理策略进行权限管理,在 LSSO-Service上避免了由于权限问题导致的对 AS和门户的修改,同时为用户在门户和子系统中的权限一致性提供了保障.

2.2 基本数据结构定义

LSSO-Service要能正常运转,首先依赖于机制在通信过程中传输的数据结构. LSSO-Service 基本数据结构定义如下:

1) 用户身份令牌(UIT, User Identity Token),由认证服务器生成,用以标识用户是否已登录;

2) 用户角色令牌(URT, UserRoleToken),已登录用户所具备的门户角色信息;

3) 用户令牌存根(UTS, UserToken Stub),用

户名、UIT和 URT组成的三元组,由认证服务器生成并保存,用以标识 UIT和 URT的有效性;

4) 全局用户身份信息 (GUI, Global User Identity Information),门户用户信息库保存的用户信息;

5) AS编号 (ASID, Application System ID),由 AS具体编号、AS名与 AS域名所组成的三元组,代表每个 AS在门户中的唯一编号;

6) 局部用户角色映射 (LURM, Local User Role Mapping),由门户角色编号、ASID和 AS内部角色编号组成的三元组,维护门户角色与 AS内部角色之间的映射.

以上定义的数据结构用以装载 SSO认证流程中交换的相关数据.

2.3 安全策略设计

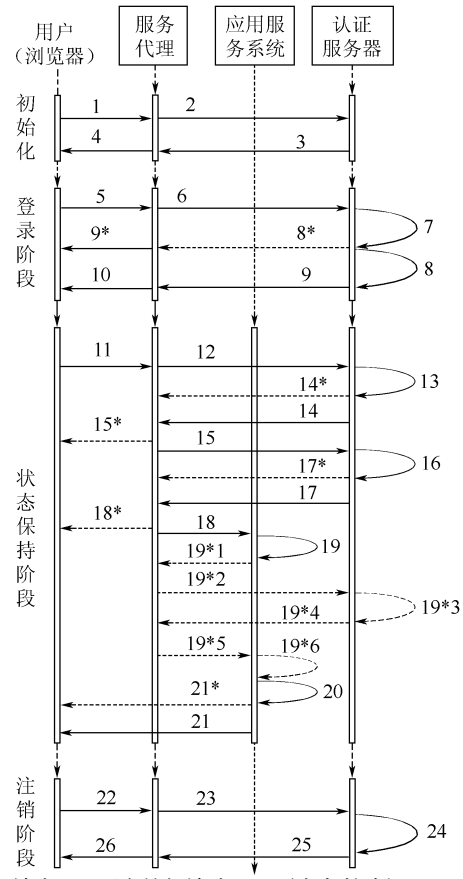
鉴于 LSSO-Service的数据结构装载信息均为敏感信息,安全要求高,故 LSSO-Service采用了以下 2种安全策略: ①利用 SSL(Secure Sockets Layer)协议保证传输安全. AS内嵌的服务代理在调用认证服务时必须出示认证服务器颁发的 SSL数字证书. ②利用非对称加密保证数据元素级安全. 采用 RSA (Rivest Shamir Adleman)算法与 AES (The Advanced Encryption Standard)算法相结合的方式,对服务代理和认证服务器交互过程中传输的数据进行加密.

以上安全策略可有效地保证机制在数据传输级别和数据元素级别的数据安全[10]. 在实际运用中可与机制的具体实现所提供的安全框架共同保护机制的信息安全.

2.4 认证流程及原子服务划分

LSSO-Service认证流程分为初始化、登录、状态保持和注销 4个阶段,具体流程如图 2所示. LSSO-Service以 SOA思想为指导,将认证流程分解成一组原子服务,这组原子服务对外提供统一、通用的调用接口,便于认证代理调用,实现 SSO功能;同时为保证其原子性和完整性[11-12],对认证流程进行原子服务划分时,需满足: ①单一性原则. 一个原子服务只包括唯一一对认证代理与认证服务器之间的请求/响应通信. ②完备性原则. 一个原子服务对认证代理的一次调用请求而言是功能完备的,不能与其它原子服务在功能上有所交叉.

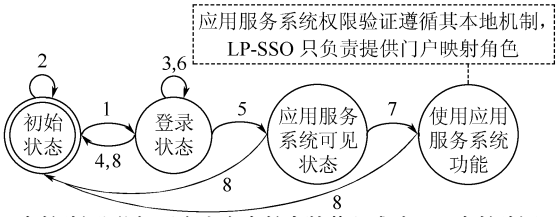
基于上述原则,将图 2所示的 LSSO-Service认证流程分解为 5种原子服务,分别是用户身份令牌获取、用户身份令牌验证、用户一级权限校验、门户映射角色获取和身份令牌废弃. 此外, LSSO Service还需要一组辅助服务维持其正常工作,分别是用户门户身份获取、用户信息同步、具备 AS入口访问权的门户角色获取、用户角色映射注册、认证服务调用接口查询和认证服务调用接口注册. 结合 LSSO-Service认证流程和其分解



1—访问请求; 2—用户访问请求; 3—要求身份验证; 4—要求身份验证,定向至登录入口; 5—身份验证信息 (用户名、密码); 6—请求验证身份验证信息; 7—身份验证; 8—生成 UIT, URT和 UTS; 8* —身份验证失败; 9—返回 UIT; 9* —要求重新进行验证; 10—认证成功,获得 UIT,允许后续操作; 11—带 UIT的访问请求; 12—请求校验 UIT有效性; 13—校验 UIT有效性; 14—UIT验证成功; 14* —UIT验证失败; 15—发送 UIT和 ASID,请求一级鉴权; 15* —要求重新进行验证; 16—根据 UIT和 UTS获得 URT,结合 ASID进行一级权限鉴别; 17—用户具备对应用服务系统的访问权限; 17* —用户不具备对应用服务系统的访问权限; 18—应用服务访问请求; 18* —提示无权操作; 19—检验是否已缓存用户角色; 19* 1—未缓存用户权限,请求门户映射角色; 19* 2—发送 UIT和 ASID,请求门户映射角色; 19* 3—根据 UIT和 UTS获得 URT, GUI, LURM,结合 ASID进行求积,取得门户角色映射到该应用服务系统的角色; 19* 4—返回相应的门户映射角色; 19* 5—返回门户映射角色; 19* 6—根据用户名取得用户在应用服务系统中的角色,与门户映射求并,得出用户最终角色并缓存; 20—根据用户角色进行权限验证; 21—访问请求回应; 21* —不具备执行所请求操作的权限,提示无权操作; 22—带 UIT的注销请求; 23—请求废弃 UIT; 24—废弃 UIT, URT, UTS; 25—废弃确认; 26—注销完成; 带* 符号、以虚线标识的步骤代表逻辑判断为 false的分支.

图 2　LSSO-Service认证流程时序图

成的原子服务, 对机制进行状态划分, 最后得到LSSO-Service认证流程状态转换图, 如图 3所示.



1—身份验证通过, 用户门户身份令牌获取成功; 2—身份验证不通过, 用户门户身份令牌获取失败; 3—检验门户身份令牌, 未过期, 验证通过; 4—检验门户身份令牌, 已过期, 验证不通过; 5—具备 AS入口访问权, 一级权限校验通过; 6—不具备 AS入口访问权, 一级权限校验不通过; 7—AS通过服务代理取出门户映射角色; 8—门户身份令牌废弃.

图 3 LSSO-Service认证流程状态转换图

3 LSSO-Service的应用

LSSO-Service的特点主要表现在: ①服务形式提供 SSO功能并可进行动态更新, 耦合度低, 扩展性、复用性高; ②与认证服务器交互由代理完成, 结构简单, 对 AS改动要求极小, 易用性好; ③具备灵活的权限管理策略, 完整的数据结构定义, 充分的信息安全策略和清晰的认证流程.

下面将介绍机制在大型水利建管信息门户 ——南水北调中线干线建设管理信息系统中的应用设计.

本文基于 LSSO-Service设计了 SSO服务, 使用 JAVA编程语言实现, 在 Apache AXIS2 Web服务引擎上进行发布. 认证服务器的服务注册仓库和用户数据均采用 LDAP (Lightweight Directory Access Protocol)存储. 基于 JAVA, C#和 Python三种 AS设计语言提供服务代理, 根据 AS具体实现选择服务代理进行配置整合.

服务的安全策略综合 LSSO-Service安全策略和 Apache AXIS2 Web服务引擎所提供的 WS-Security安全框架, 对服务代理和认证服务器间交换的 SOAP报文进行了 XML-Encryption和 XML-Signature处理, 以确保信息安全. 在用户登录后, 用户身份令牌将由认证服务代理缓存在用户与应用系统服务器建立的会话中. 当用户需要进行 AS切换, 与服务器重新建立会话时, 用户身份令牌以采用 AES算法加密的 URL参数形式传递至新建会话, 并由认证服务代理加以缓存, 具体认证流程见图 2目前南水北调建管系统的 SSO服务运转正常, 能够适应对 AS进行快速整合的系统需求.

4 结束语

本文针对信息门户的需求特点, 基于 SOA软件架构思想, 提出了一种轻量级单点登录服务机制 LSSO-Service 实践证明, 该机制流程清晰, 结构简单, 复用性和扩展性较好, 适用于为门户提供量身定制、简单实用的 SSO服务, 便于门户快速整合. 由于其具备松散耦合、易于扩展、整合难度低的特性, 克服了传统单点登录机制无法满足在动态松耦合环境下实现快速整合的缺陷, 因此对于门户建设具有较高的理论意义和参考价值.

参考文献 (References)

[1] Staff W. Enterprise information portals: What they are, why they're important, and why you may need one[M]. Waltham, Massachusetts: Unitas Corporation, 2001: 5

[2] Satoh F, Itoh T. Single sign on architecture with dynamic tokens [C]//2004 Symposium on Applications and the Internet. Tokyo: IEEE Computer Society, 2004: 197-200

[3] Pashalidis A, Mitchell C. A taxonomy of single sign-on systems [C]//Proceedings of 8th Australasian Conference on the Information Security and Privacy. Wollongong, Australia: Springer, 2003: 249-264

[4] Pashalidis A, Mitchell C. Using GSM/UMTS for single sign-on [J]. Mobile Future and Symposium on Trends in Communications, 2003, 12: 138-145

[5] Oppliger R. Microsoft. NET passport: a security analysis[J]. IEEE Computer Society, 2003, 36: 29-25

[6] Kormann D P, Aviel D. Rubin, risks of the passport single sign-on protocol[J]. Computer Networks, 2000, 33: 33-51

[7] Beznosov K, Flinn D J, Kawamoto S. Introduction to Web service and their security[J]. Information Security Technical Report, 2005, 10: 2-14

[8] Hodges J. Liberty architecture overview V1.1 [EB/OL]. Yorba Linda, CA: Liberty Alliances, 2003. http://projectliberty.org/specs/

[9] Bayem S. CAS2.0: proxiable credentials service [EB/OL]. New Haven: Yale University, 2004. http://www.yale.edu/tp/auth/cas20.html

[10] Newcomer E, Lomow G. Understanding SOA with Web services [M]. 徐涵译. 北京: 电子工业出版社, 2006: 313-347
Newcomer E, Lomow G. Understanding SOA with Web services [M]. Translated by Xu Han. Beijing: Publishing House of Electrics Industry, 2006: 313-347(in Chinese)

[11] William M, Corvine A, Ralf B, et al. Managing information access to an enterprise information system using J2EE and services oriented architecture[M]. Chantilly, Virginia: International Business Machines Corporation, 2005

[12] 吴晶, 熊璋, 朱成军, 等. 面向服务的企业门户架构及其实现的研究[J]. 计算机应用研究, 2007, 24(1): 112-115
Wu Jing, Xiong Zhang, Zhu Chengjun, et al. Research on service oriented architecture for enterprise portal and its implementation[J]. Application Research of Computers, 2007, 24(1): 112-115(in Chinese)

(责任编辑: 文丽芳)