基于 CAS 和 Liferay 门户的统一身份认证系统

陈国慧[1,2],谭海波[1],吕 波[1],李晓风[1,2]
(1. 中国科学院 合肥物质科学研究院,安徽 合肥 230031;
2. 中国科学技术大学,安徽 合肥 230026)

摘 要: 统一身份认证系统是提供了用户身份信息库和接口服务以供其他应用系统接入,并验证用户身份的系统。通过对目前主流单点登录系统、单位内用户权限与组织架构数据库系统需求多样化的特点的介绍和对比,选择通过集成 CAS 单点登录系统与 Liferay 门户框架,设计和实现统一身份认证系统,来解决单位当前存在的多应用系统用户账户不统一,以及由此导致的冗余信息管理困难的问题。系统设计分为数据库系统、CAS 单点登录系统和 Liferay 门户三部分,并具体给出了每部分的重点设计环节和实现过程,完成一次登录即可使用多个子系统,并实现统一的信息及应用集成,达到消除业务共享不畅的目的。该系统不仅提供统一的用户界面和集成的应用资源,还由于其高度的可配置性和扩展性,方便开发人员根据需要研发新的应用,便于管理人员对系统中的用户、组织、角色和站点进行管理,同时也有利于用户进行个性化定制,满足各方面使用和管理需求,起到更加充分利用网络资源的作用。

关键词: CAS; Liferay; 统一身份认证; 门户; 单点登录; 关系数据库
中图分类号: TP302 **文献标识码:** A **文章编号:** 1673-629X(2018)12-0106-05
doi: 10.3969/j.issn.1673-629X.2018.12.023

Uniform Identity Authentication System Based on CAS and Liferay Portal

CHEN Guo-hui[1,2], TAN Hai-bo[1], LYU Bo[1], LI Xiao-feng[1,2]
(1. Hefei Institutes of Physical Sciences, Chinese Academy of Sciences, Hefei 230031, China;
2. University of Science and Technology of China, Hefei 230026, China)

Abstract: The unified identity authentication system is a system that provides users identity information database and application program interface for other application systems to access and verifies user identity. By introducing and comparing requirements diversification characteristics of the current mainstream single sign-on system and internal user rights and organizational structure database system, through the integration of the central authentication service and Liferay portal framework, we design and implement the unified identity authentication system to solve the problem of the disunity of user accounts in multiple application systems and the difficulty in managing redundant information. The system consists of three parts: database system, central authentication service and Liferay portal. The key design steps and implementation of each part are given in detail. Multiple subsystems can be used after one login, and unified information and application integration can be realized to attain the goal of eliminating inconvenience of business sharing. The system not only provides a unified user interface and integrated application resources, but also can be easily extended by developers according to their requirements because of its high configurability and extensibility. It is convenient for managers to manage users, organizations, roles and sites in the system, and at the same time, also beneficial for users to make personalized customization, meeting the use and management requirements of all aspects and playing the role of making full use of network resources.

Key words: CAS; Liferay; uniform identity authentication; portal; single sign on; rational database

0 引 言

随着计算机技术、网络技术的飞速发展,用户可使用的信息化应用,诸如办公应用、邮件服务、财务管理、设备管理等系统的数量日益增多。然而这些系统大多都是独立设计开发,相互之间缺乏信息共享、业务互动以及文件共享等方面的设计,在时代的背景下,逐渐成

收稿日期: 2018-01-22 **修回日期:** 2018-05-23 **网络出版时间:** 2018-07-21
基金项目: 安徽省科技重大专项项目(711245801052)
作者简介: 陈国慧(1989-),男,硕士研究生,研究方向为计算机应用; 谭海波,博士,研究员,硕导,研究方向为计算机应用和网络安全。
网络出版地址: http://kns.cnki.net/kcms/detail/61.1450.TP.20180920.1536.030.html

为信息孤岛[1]。由于各应用系统的分散管理,导致它们的使用和维护产生诸多问题: 开发人员需要维护多个异构的应用系统,所使用的架构、技术可能不同,使维护工作变得困难; 管理人员需要维护各应用系统中的用户、组织和角色等信息,多个系统很容易产生信息冗余和冲突; 用户使用各自孤立的应用系统需要使用不同账号和密码,系统数量越多,越是增加用户账号管理成本和系统使用成本。而当前各行业、各领域工作协同已成趋势化,这些问题亟待解决,因此构建统一的身份认证系统,提供集中的信息管理和应用集成已成当务之急[2]。

为解决上述问题,设计并实现了一种基于 CAS[3] 单点登录[4] 和 Liferay[5] 门户[6] 的统一身份认证系统[7],提供统一身份认证库、统一的用户界面和应用资源集成平台。

1　技术基础

1.1　数据库系统

通过需求分析,系统不仅要提供对用户、组织和权限的管理,并且由于院、所、中心、实验室等研究机构之间经常性的跨组织、跨学科交流,需要实现和提供一个灵活方便的人员、课题组、权限以及内容的管理和使用平台。

经过调查研究和比对的结果,该系统选择关系数据库 MySQL 管理系统数据。统一身份认证系统通常采用 LDAP 作为用户身份信息存储方式,主要利用其查询数据速度快和树状层次结构存储的优点[7]。但是在 LDAP 中增加、删除和修改数据存在速度过慢的问题,不能满足系统对数据修改效率的要求。而且 LDAP 并不支持事务机制,在该系统架构下,可能会出现数据不一致的问题。MySQL 数据库对数据增加、删除和修改则没有速度过慢的问题,且其支持事务机制可以解决数据不一致的问题,满足系统对数据管理的性能和安全方面的需求[8]。

1.2　单点登录

该系统使用中央认证服务(central authentication service,CAS) 作为单点登录实现的技术。单点登录(single sign-on,SSO) 是服务于企业业务整合的解决方案,在多应用系统中,用户只需一次登录就可以访问所有相互信任的应用系统[9]。单点登录的实现技术有多种,由于商用产品成本过高,基于项目成本和功能需求,主要考虑开源产品,目前主流技术有 Kerberos[10] 和 CAS 等[11]。Kerberos 技术的安全性较高,但每一个子应用都需要实现 Kerberos 体系,其实现比较复杂,部署和使用成本过高。CAS 系统由于其简单有效、安全可靠、文档齐全的特性,使其部署简单,并且有良好的社区支持,在实际中应用广泛[3]。经过比对,该系统选择 CAS 技术作为单点登录模块。

1.3　门　户

门户也被称为信息门户,是一个提供统一用户界面和集成应用资源的系统,提供统一信息管理和集中的应用入口,通常以网页的形式展现[6]。门户最主要的两大功能是应用的集成和内容的展现,提供访问不同来源内容的能力和统一的信息管理平台,通常还包括单点登录、权限控制、内容管理、信息发布、文件管理等功能[12]。

该系统使用 Liferay 搭建门户系统。Liferay 是开源的门户项目,利用 Spring、Hibernate、Struts 等框架[13],实现了 JSR168 规范[14] 中提出的门户标准。Liferay 支持对用户、组织、角色、站点、权限等的管理,还支持公告、文章、文档、图片等内容的发布、展示和管理,允许用户个性化定制个人空间等功能,是一个通用的、统一的工作平台。同时,以 Liferay 作为开发平台,由于其良好的可扩展性和定制性,可极大提高开发人员的开发效率和管理质量。

2　系统架构

该系统主要由数据库系统、CAS 单点登录系统和 Liferay 门户三个部分构成,系统架构如图 1 所示。

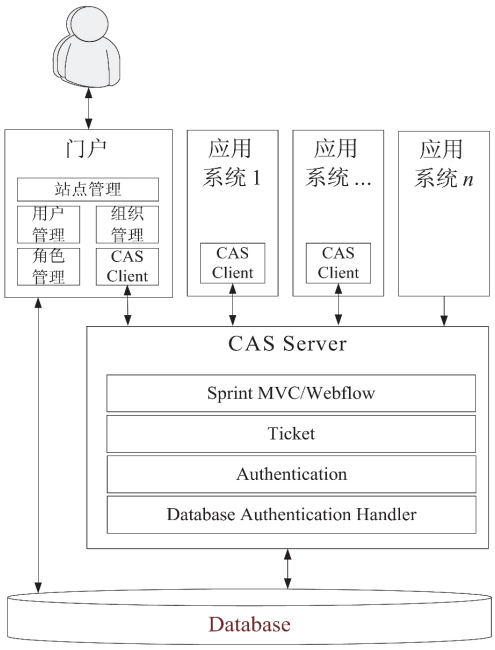

图 1　统一身份认证系统架构

数据库系统存储统一身份认证基础库,主要包括用户、组织、角色信息等数据,是整个系统的基础数据源,为统一身份认证提供数据支持。CAS 单点登录系统,提供单点登录服务,主要由 CAS Server 和 CAS Client 两个部分组成。CAS Server 作为认证中心,负责对用户的认证工作,对用户名和密码凭证进行验证

处理; CAS Client 被集成于应用系统中,负责将对用户身份认证的工作重定向到 CAS Server 进行处理。Liferay 门户提供统一的用户界面和内容展示,包括对用户、组织、角色以及站点等信息的管理,其本身集成了 CAS Client 包服务,也可作为普通应用系统使用统一身份认证功能。

3 系统实现

3.1 数据库设计

该系统使用 MySQL 数据库作为数据库管理系统。根据需求,系统主要数据表包括用户、组织、角色、用户组和资源动作表等,关系如图 2 所示。

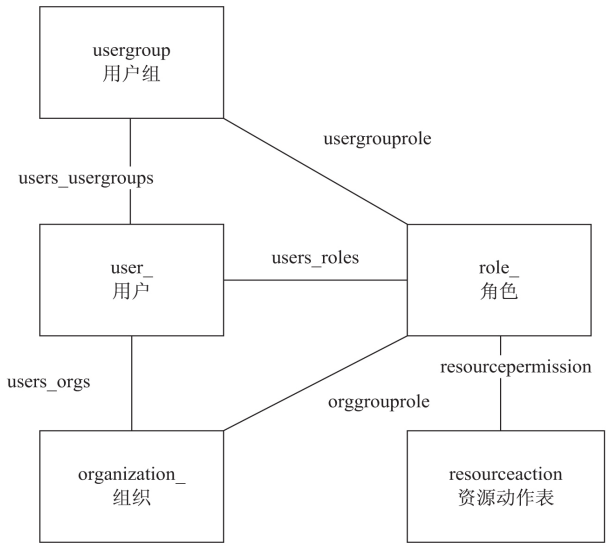

图 2 主要数据表关系

用户表 user_是系统的关键实体,统一身份认证的信息来源; 组织表 organization_是组织架构表,存储人员的组织架构关系; 用户组 usergroup 存储用户组信息,可用于灵活管理流动人员和对人员进行分组; 角色表 role_存储的是角色的定义,系统基于角色提供权限,角色被赋予用户或用户组,使其具有特定的权限,角色主要分为管理员、用户、临时用户等; 资源动作表 resourceaction 存储角色的权限,及对用户访问门户中资源的增加、删除、修改和查询的能力。

3.2 单点登录的实现

3.2.1 CAS 单点登录流程

实现单点登录功能需要解决三个关键问题: 登录信息传递问题、登录状态判断问题和登出信息的传递问题。

CAS 通过 Ticket(票据)和对 Ticket 的交互处理,解决用户登录状态问题。主要票据有全局票据(ticket granting ticket,TGT)和服务票据(service ticket,ST)等。TGT 是认证中心 CAS Server 为用户签发的登录票据,用户拥有了 TGT,就可以证明自己已经在认证中心登录成功,即完成全局登录; 全局会话(ticket granting cookie,TGC)是 TGT 对象的 ID,被作为浏览器与 CAS Server 之间的会话(Cookie)[15]返回给用户,用作浏览器和 CAS Server 间通讯的访问凭证,且其只能通过安全通道传输(HTTPS)[16]; ST 是 CAS 为用户签发的访问某一特定应用的票据。

用户首次访问某应用时,如系统 A,从发出访问请求到获得资源流程如图 3 所示。

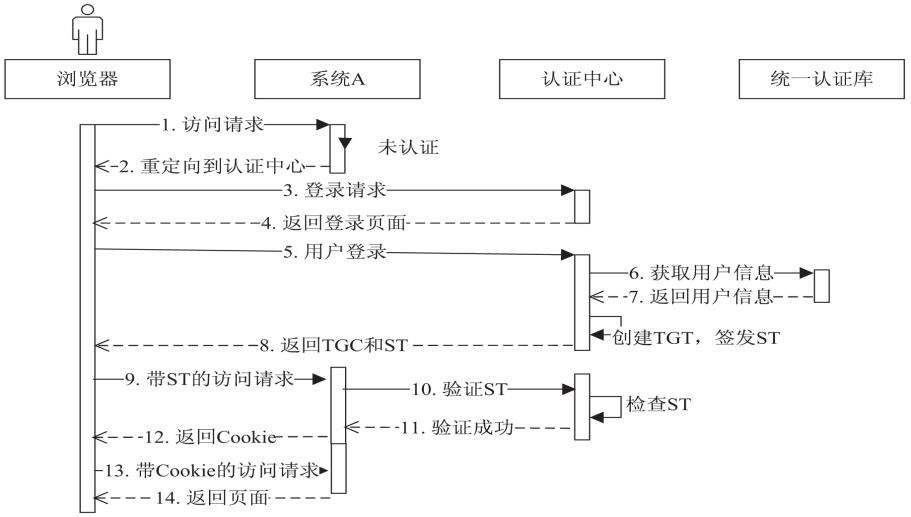

图 3 单点登录流程

具体流程说明如下:

(1) 用户使用浏览器请求访问系统 A 中受限资源,系统 A 检查发现用户没有经过认证,需要对用户身份进行认证;

(2) 系统 A 将请求重定向到认证中心;

(3) 浏览器向认证中心发送登录请求;

(4) 认证中心返回登录页面;

(5) 用户输入用户名和密码后提交至认证中心;

(6) 认证中心从统一认证库中获取用户信息;

(7) 统一认证库返回用户信息给认证中心,认证

中心验证成功后,创建 TGT,并签发 ST;

(8) 认证中心将 TGT 的 ID,即 TGC 作为 Cookie,返回给用户浏览器,并重定向至系统 A,重定向链接中将 ST 设置为参数返回给系统 A;

(9) 浏览器被重定向至系统 A,此时请求携带参数 ST;

(10) 系统 A 判断请求存在 ST 票据,直接向认证中心发出验证 ST 请求;

(11) 认证中心检查该 ST 是否存在,若存在,则证明用户身份已经获得认证,返回验证结果;

(12) 系统 A 收到验证结果,创建局部会话的 Cookie,返回给用户;

(13) 用户再次请求访问系统 A 的受限资源,携带 Cookie,系统 A 检查到会话已经存在该 Cookie,验证通过;

(14) 将请求的资源返回给用户。

经过上述一系列流程后,系统完成对用户的初次认证,并最终将用户请求的资源返回给用户。整个过程完成了浏览器、应用系统和认证中心三者之间的两两互连,从而解决登录信息的传递问题。三者之间的关系为: 浏览器与认证中心通过 TGC 维持全局会话; 应用系统通过 ST 与认证中心保持联系,验证用户的身份; 浏览器与应用系统则通过普通 Cookie 维持局部会话。

用户首次访问未登录的系统 B 时,与访问系统 A 时的区别在于第 3 步登录请求时,该请求直接携带 TGC 信息,认证中心会直接签发新的 ST,而不必再经过过程 4 ~ 7,直接省去了用户输入用户名和密码登录的过程。

用户登出应用系统时,除销毁该应用的局部会话外,还通知认证中心结束全局会话,认证中心再通知其他应用系统销毁各自局部会话,实现其他应用系统的自动登出。

3.2.2　CAS Server 的部署和配置

(1) 部署环境配置。

该系统选择将 CAS Server 部署到 Tomcat[17] 服务器。为增强系统安全性和启用 CAS Server 的单点登录功能,需要对 CAS Server 的安全证书进行管理。该过程使用 Java 自带的证书管理工具 keytool 生成证书并导入到 CAS Server 的执行环境中,在 Tomcat 配置文件 server. xml 中添加对 HTTPS 连接支持的配置,关键配置如下:

```
<Connector
protocol="org. apache. coyote. http11. Http11NioProtocol"
port="8443" SSLEnabled="true"
scheme="https"  secure="true"
lientAuth="false"  sslProtocol="TLS"
keystoreFile="  pathname"
keystorePass="pass"
maxThreads="150"  >
</Connector>
```

(2) CAS Server 依赖包配置。

CAS Server 使用 Maven[18] 进行项目管理。项目默认没有提供 MySQL 数据库的连接和认证的依赖包,部署时需要添加相关依赖包的配置,配置位于 CAS Server 项目的 pom. xml 文件中。

(3) CAS Server 数据库、密码规则配置。

CAS 支持对数据库的连接和使用数据库作为认证数据源,有多种密码验证规则可供选择。对 CAS Server 的数据库连接、获取认证数据和密码验证规则设置在配置文件 application. properties 中,添加的配置如图 4 所示。

```
连接 MySQL 数据库
    cas.authn.jdbc.query[0].url=jdbc:mysql://domain-name:33
06/lportal?useUnicode=true&characterEncoding=UTF-8&auto
Reconnect=true&useSSL=true
数据库用户名密码
    cas.authn.jdbc.query[0].user= username
    cas.authn.jdbc.query[0].password=pwd
获取用户信息的 SQL 语句和密码字段
    cas.authn.jdbc.query[0].sql=SELECT  *  FROM  user_
WHERE emailAddress =?
    cas.authn.jdbc.query[0].fieldPassword=PASSWORD_
配置密码加密算法，以 MD5 为例
    cas.authn.jdbc.query[0].passwordEncoder.type=DEFAULT
    cas.authn.jdbc.query[0].passwordEncoder.characterEncodi
ng=UTF-8
    cas.authn.jdbc.query[0].passwordEncoder.encodingAlgorit
hm=MD5
```

图 4　CAS 中数据库和密码验证规则配置

3.2.3　CAS Client 的使用

CAS Client 以包或库的形式被集成到待认证的应用系统,保护应用系统中访问受限的资源。应用系统不必在应用内对用户身份进行认证,而是将认证工作重定向到 CAS Server 进行处理。CAS Client 能够支持多种编程语言开发的应用系统,包括 Java,. Net,PHP,Perl 等。

CAS Client 与 Web 应用整合时只需关注两个方面,分别是请求路径是否需要跳转到登录页面和重定向用户到请求的资源。认证工作可通过设置过滤器(Filter)[19] 配置完成,将配置中的过滤器和监听器配置成的站点和重定向目标配置成 CAS Server 对应的接口。

3.3　门户的配置与使用

(1) Liferay 的部署配置。

为避免数据的冗余,该系统中 Liferay 与 CAS 使用同一份用户信息。Liferay 使用数据库数据需要与

MySQL 进行对接，并使密码验证规则相互匹配。需要在 Liferay 的启动配置文件 portal-setup-wizard. properties 中添加相关配置，如图 5 所示。

(2) Liferay 对接 CAS 设置。

为使 Liferay 启用 CAS 单点登录功能，需要在其系统设置中与 CAS 进行对接。需要在认证的设置中配置 CAS 单点登录的登录 URL、退出 URL、服务器名和服务器 URL 等。

该系统提供普通用户自定义门户桌面内容的功能，用户可进行个性化设置，发布公告、文章、文档、图片等。管理员可对用户、组织和角色等信息进行管理。

数据库的驱动、连接和用户名密码配置
jdbc.default.driverClassName=com.mysql.jdbc.Driver jdbc.default.url=jdbc:mysql://domain-name/lportal?characterEncoding=UTF-8&dontTrackOpenResources=true&holdResultsOpenOverStatementClose=true&useFastDateParsing=false&useUnicode=true jdbc.default.username=username jdbc.default.password=pwd
门户的系统管理员
admin.email.from.address=Email Address
密码加密配置，有多重组合可以选择，需与 CAS 中的保持策略一致。
passwords.encryption.algorithm=MD5 passwords.digest.encoding=hex

图 5 数据库和密码验证规则配置

4 结束语

基于 CAS 单点登录和 Liferay 门户，以关系数据库为数据存储基础设计与实现了统一身份认证系统，达到整合用户身份信息、集中应用系统入口和统一系统信息管理的目的。该系统为用户间进行信息共享和交流提供了一个良好的平台，并进一步提高了各系统间协同工作的能力，同时降低了用户信息管理的难度和成本。在未来的工作中，将继续深化对统一身份认证系统的开发、建设和推广工作。不仅要将现有的应用系统纳入到统一身份认证平台中进行管理，而且要对新的应用系统进行推广并对开发人员提供帮助，使统一身份认证系统得到更广泛的支持和认可，削弱乃至消除信息孤岛的存在。

参考文献：

[1] 王晓光. 我国电子政务中“信息孤岛”问题及对策研究[D]. 济南：山东大学，2012.

[2] 张 璟. 基于 Web Services 和 J2EE 企业应用系统集成方法研究[D]. 青岛：山东科技大学，2005.

[3] 景民昌，唐弟官. 开放源码的 CAS 单点登录系统研究[J]. 现代情报，2009，29(3)：125-127.

[4] 梁志罡. 基于 Web service 的混合架构单点登录的设计[J]. 计算机应用，2010，30(12)：3363-3365.

[5] SCHUH G，BRAEKLING A，VALDEZ A C，et al. Using liferay as an interdisciplinary scientific collaboration portal a comparative usability study of version 6.1 and 6.2 [C]//8th international conference on social computing and social media held as part of 18th international conference on human-computer interaction. Toronto，Canada：[s. n.]，2016：405-414.

[6] 孟晓川，马自卫. 基于 Liferay 的多维化门户系统在数字图书馆中的研究与实现[J]. 现代图书情报技术，2008(12)：8-14.

[7] 贺玉明，李晋宏，唐 辉. LDAP 在数字校园统一身份认证系统中的应用[J]. 计算机技术与发展，2011，21(5)：139-142.

[8] SCHWARTZ B，ZAITSEV P，TKACHENKO V. High performance MySQL：optimization，backups，replication，and more[M]. 3rd ed. USA：O'Reilly Media，Inc.，2012：1-34.

[9] 杨延双，杨 武，史吉文. 跨域的单点登录在集成系统中的设计与实现[C]//第四届中国软件工程大会：四川大学学报(工程科学版)编辑部会议论文集. 成都《四川大学学报(工程科学版)》编辑部，2007：108-111.

[10] 齐忠厚. Kerberos 协议原理及应用[J]. 计算机工程与科学，2000，22(5)：11-13.

[11] 胡雅琴. 单点登录技术现状调查与分析[J]. 软件产业与工程，2014(1)：53-56.

[12] DIAS C. Corporate portals：a literature review of a new concept in information management[J]. International Journal of Information Management，2001，21：269-287.

[13] 李 洋，孙永维，许 冰，等. 基于 Ajax，Struts，Hibernate 和 Spring 的 J2EE 架构[J]. 吉林大学学报：信息科学版，2011，29(6)：576-584.

[14] ABDELNUR A，HEPPER S. Java Portlet specification[M]. [s. l.]：Sun Microsystems，2003：19-28.

[15] 胡忠望，刘卫东. Cookie 应用与个人信息安全研究[J]. 计算机应用与软件，2007，24(3)：50-53.

[16] MORI T，INOUE T，SHIMODA A，et al. Statistical estimation of the names of HTTPS servers with domain name[J]. Computer Communications，2016，94：1-10.

[17] 边清刚，潘东华. Tomcat 和 Apache 集成支持 JSP 技术探讨[J]. 计算机应用研究，2003，20(6)：12-14.

[18] 许晓斌. Maven 实战[M]. 北京：机械工业出版社，2011：27-44.

[19] 李 建. Java Web 开发中过滤器组件应用及实例解析[J]. 电脑开发与应用，2009，22(11)：58-60.