Journal of Computer Applications
计算机应用,2014,34(S1):50 -54
ISSN 1001-9081
CODEN JYIIDU
2014-06-30
http://www.joca.cn

文章编号:1001-9081(2014)S1-0050-05

社会化网络服务中 OAuth2.0 的应用研究与实现

卢慧锋*,赵文涛,孙志峰,游 超
(国防科学技术大学 计算机学院,长沙 410073)
(*通信作者电子邮箱 jluhuifeng@163.com)

摘 要:随着以 Facebook 和新浪微博为代表的社会化网络服务(SNS)的日益普及,对认证与授权技术也提出了更高要求。OAuth 协议为开放平台解决用户身份验证和用户资源授权问题提供了一个安全、开放的标准。分析了 OAuth 协议的原理与工作流程,利用新浪微博提供的 API 接口,设计并实现了读书分享社区应用的 Web 客户端和手机客户端的授权认证功能,进一步提高了读书分享社区应用的社会化程度,同时简化了应用的用户登录流程、账号管理和用户隐私保护等问题。

关键词:社会化网络服务;OAuth 协议;授权;认证
中图分类号:TP393 **文献标志码**:A

Research and implementation of OAuth2.0 in social network services

LU Huifeng*, ZHAO Wentao, SUN Zhifeng, YOU Chao
(*College of Computer, National University of Defense Technology, Changsha Hunan* 410073, *China*)

Abstract: Along with the Social Network Services (SNS) become more and more popular such as Facebook and Sina micro-blog, there are higher requests to the authentication and authorization technology. The OAuth protocol provides a safe, open standard to solve the user authentication and user authorization problem for the open platform. This paper analyzed the principle and work process of OAuth protocol, designed and implemented Web client and mobile phone client authorization and authentication functions of shared reading community application using the API interface provided by Sina micro-blog. This method strengthens the sociality of shared reading community and simplifies the problems in the application of user login process, account management and user privacy protection.

Key words: Social Network Service (SNS); OAuth protocol; authorization; authentication

0 引言

社会性网络服务(Social Networking Service,SNS)专指旨在帮助人们建立社会性网络的互联网应用服务[1],SNS 网站是一种基于社交网络服务,为用户提供信息展示、交流与共享的平台,并注重用户关系管理服务形式的网站。该类网站的服务使互联网应用模式从传统的"人机对话"转变为"人与人对话"[2]。

随着 SNS 应用的日益普及,SNS 中对用户的认证与授权提出了更高的要求。授权与认证体系在保证网络安全通信中起到了至关重要的作用[3],授权是由其中一方决定另一方可以拥有的权限,而认证指的是一方验证另一方的身份是否合法或是否得到合法的授权,若认证通过,则允许其使用授权方指定的权限,否则拒绝其申请。将授权与认证相结合,为网络中的安全通信提供了基础的保证。

开放性是当今互联网快速发展的主要促进因素,网络系统的交互性也在日益增强[4]。在这种环境下,互联网服务之间的整合已经成为必然的趋势,因此开放平台(Open Platform)应运而生[5]。

网站的服务商将自己的服务封装成一系列计算机容易识别的数据接口开放出去,供第三方开发者使用,这种行为称为 OPEN API,提供开放 API 的平台就称为开放平台[6]。

本文利用社会化网站的开放接口,基于 OAuth2.0 协议,针对读书分享社区这一典型 SNS 案例,研究并实现了 SNS 认证授权应用方案,实现了社会化网站之间用户账号的互通,简化了用户登录系统的操作。

本文简要介绍了 SNS、开放平台、授权与认证的相关概念;阐述了 OAuth 协议的工作原理和具体流程;提出了基于 OAuth 协议的社会化网络系统设计方案;以新浪微博开放平台提供的 API 接口,以读书分享社区的授权登录为案例,实现了 OAuth 协议的具体应用。

1 OAuth 协议

1.1 OAuth 起源

OAuth 起源于去中心化的网上身份认证系统 OpenID[7]。对于支持 OpenID 的网站,用户只需要预先在一个作为 OpenID 身份提供者(Identity Provider, IdP)的网站上注册即可确认数字身份。2007 年 4 月,OAuth 讨论组成立并撰写了一个开放协议的提议草案。2010 年 4 月份,OAuth 1.0 正式成为互联网标准协议,具体为 RFC5849:The OAuth 1.0 Protocol。OAuth 2.0 是 OAuth 协议的下一版本,但不向前兼容 OAuth 1.0。和 OAuth 1.0 相比,OAuth 2.0 关注客户端开发者的简易性,同时为 Web 应用、桌面应用和手机应用提供专门的认证流程[8]。

收稿日期:2013-08-19;**修回日期**:2013-10-20。

作者简介:卢慧锋(1987 -),男,黑龙江鹤岗人,硕士研究生,主要研究方向:信息安全、网络安全; 赵文涛(1973 -),男,内蒙古凉城人,副教授,博士,主要研究方向:信息安全、网络安全; 孙志峰(1978 -),男,河北冀州人,助教,硕士研究生,主要研究方向:信息安全、网络安全;游超(1986 -),男,湖南长沙人,硕士研究生,主要研究方向:信息安全、网络安全。

1.2　OAuth 协议工作原理

OAuth 协议中涉及的角色包括服务提供方、用户和第三方应用[9]。

1)服务提供方(Server Provider):一般是提供服务或数据交换的大型网站,用户利用账号和口令登录到该网站可以存储受保护的资源。

2)用户(User):存放受保护资源的拥有者。他们可以使用第三方应用访问自己的数据,或使用服务提供方的业务应用。

3)第三方应用(Consumer):要访问服务提供方资源的第三方应用。在认证过程之前,客户端要向服务提供者申请客户端标识。

OAuth 协议解决服务提供方、用户和第三方应用之间的三角关系,即当用户需要第三方应用为其提供某种服务,并且该服务需要服务提供方获取该用户受保护的资源时,OAuth 协议确定用户的认证与授权过程,授权成功后,第三方应用才可以获取该用户的资源。OAuth 协议基本原理如图 1 所示。

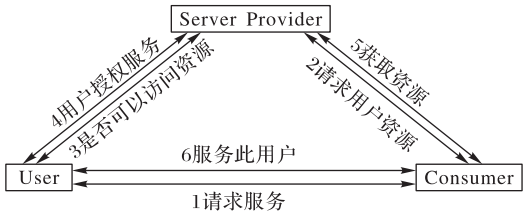

图 1　OAuth 协议基本原理

如图 1 所示, OAuth 协议的基本原理是第三方应用与不同的服务提供方已经建立关系的前提下,第三方应用与服务提供方共享一个密码并且将公钥公开给服务提供方,服务提供方使用该公钥来确定第三方应用的身份,服务提供方将用户重定向到第三方应用的登录页面。

1.3　OAuth 协议工作流程

OAuth 协议需要经过准备阶段、授权阶段与资源调用三个阶段。在准备阶段,客户端在授权服务提供方进行注册以获取到相应的客户端证书;在授权阶段,授权服务提供方提供统一的授权服务地址(Authorization Endpoint)进行授权服务,同时也提供统一的令牌服务地址(Token Endpoint)进行令牌申请服务;在资源调用阶段,第三方应用向服务提供方发送资源调用申请,在发送申请时需要明确指定服务提供方需要的参数和访问令牌[10]。

OAuth 授权阶段需要经过临时令牌的请求、用户授权、访问令牌的请求等三个过程。过程如下所述:

1)第三方应用向服务提供方请求临时令牌 Request_Token。如表 1 所示,在请求临时令牌过程中,第三方应用需要附加请求授权的相应类型、服务提供方颁发的唯一标识、客户端回调地址、客户端标识状态等参数。

表 1　临时令牌请求参数表

请求数据	数据描述
Response_Type	授权请求的响应类型
Client_Id	客户端标识符
Redirect_Uri	客户端回调地址
Scope	待授予的权限
State	客户端状态标识

2)服务提供方验证第三方应用的合法请求,如果验证通过,向第三方应用返回未经用户授权的临时令牌 Request_Token 和与其相对应的 Token_Secret。服务提供方响应参数如表 2 所示。

表 2　临时令牌请求服务响应参数表

返回信息	信息描述
Request_Token	访问许可
Token_Secret	客户端状态标识

3)第三方应用向服务提供方请求经过用户授权的令牌 Request_Token,如表 3 所示,该请求需要指明客户端标识、客户端回调地址和此前从服务方获取到的未授权 Request_Token 等参数。

表 3　用户授权令牌请求参数表

请求数据	数据描述
Client_Id	客户端标识符
Redirect_Uri	客户端回调地址
Request_Token	访问许可

4)服务方在对客户端请求参数进行一系列验证之后,引导用户登录授权,如果用户同意授权,生成全局唯一授权码,引导用户重新回到第三方应用。如表 4 所示,服务提供方返回用户授权令牌和全局唯一授权码。

表 4　用户授权令牌请求服务响应参数表

返回信息	信息描述
Request_Token	用户授权令牌
OAuth_Verifier	授权令牌校验码

5)第三方应用向服务提供方发送获取到的用户授权 Request_Token,请求访问令牌 Access_Token。如表 5 所示,第三方应用需要附加客户端标识符、用户授权令牌、授权令牌校验码、请求授权类型和授权用户的用户名等参数。

表 5　访问令牌请求参数表

请求数据	数据描述
Client_Id	客户端标识符
Request_Token	用户授权令牌
OAuth_Verifier	授权令牌校验码
Grant_type	授权类型
Username	授权用户的用户名

6)服务提供方认证第三方的请求,如果通过验证则向第三方应用颁发 Access_Token 以及对应的 TokenSecret。

7)第三方应用通过获得的 Access Token 访问用户授权的资源[11]。

2　设计方案

OAuth 协议为开放平台解决用户身份验证和用户资源授权问题提供了一个安全、开放的标准。读书分享社区作为一个典型的 SNS 应用,要求以读者为中心,体现社会化阅读特点。因此,本文选用读书分享社区作为 SNS 中 OAuth 应用的典型案例进行分析和设计。

2.1　需求分析

读书分享社区是一个社会化阅读分享平台。该社区的主要特点是以读者为中心,分享交流社区化,这些特点是社会化阅读的发展趋势。该社区包括 Web 客户端和安卓手机客户

端,用户在 Windows 平台和安卓手机平台都可以随时随地地通过阅览书籍获取新知识,发现新事物。该社区的主要功能是用户可以查阅社区中的所有书籍、收藏自己喜爱的书籍、对未看完的书籍加书签、与朋友分享书籍、进入社区发布帖子和对帖子进行回复等。为了进一步提高读书分享社区社会化程度,在社区的用户授权认证模块中,加入如下功能:

1)用户能够使用新浪微博账号登录读书分享社区。

2) 用户可以通过读书分享社区系统发布新微博与新浪微博中的好友进行交互。

2.2 系统框架设计

该系统采用分层架构 + 软件开发工具包架构模式,分层的设计使系统具有良好的可维护性和可扩展性[12]。由于安卓手机客户端使用 Sqlite 数据库,而 Web 客户端使用的是 Mysql 数据库,因此我们将数据访问层抽象化,这样就无需为特定的数据库而编写代码,只需修改配置文件即可访问相应的数据库。系统架构如图 2 所示。

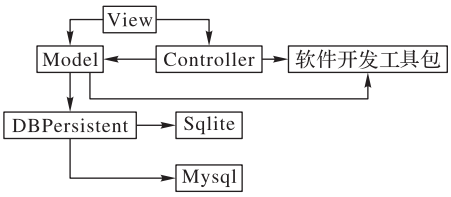

图 2 系统架构示意图

该系统架构图中,自底向上分别为:

1)数据库访问层:提供数据存储、访问、增加、修改、删除等功能。

2)数据持久层 DBPersistent:定义数据库访问接口。

3)业务逻辑模型层 Model:系统内部业务逻辑的实现,完成数据存储的相关操作。

4)控制器层 Controller:处理用户操作,对用户的操作调用相应的业务逻辑,并且调用相应的页面展示。

5)页面展示层:Web 客户端通过控制器(Controller),安卓手机客户端通过活动(Activity)调用相应业务逻辑完成相应页面的显示。

采用分层架构模式的优点显而易见,通过分层架构加软件开发包的架构模式对于业务逻辑的微小变动不至于牵动整个程序的更改,提高了代码的可重用性。软件开发工具包是经过多次验证的成熟代码,使用软件开发包,提高了工作效率,更易于系统的扩展。在认证和授权模块中,控制器层调用业务逻辑层的申请授权接口向新浪微博平台申请授权,如果申请授权成功,客户端通过调用业务逻辑层的获取用户信息接口将用户信息显示到页面上。

2.3 用例分析

在授权认证工作流程中,包括新浪微博平台、用户和读书分享社区客户端三个参与者[13]。用户与读书分享社区客户端之间的交互包括启动登录流程、退出应用、账号绑定、发布新微博。用户与新浪微博开放平台之间的交互主要是授权管理。图 3 为读书分享社区授权认证模块用户与客户端和新浪微博平台交互的用例图。

根据 OAuth 授权认证的工作流程,读书分享社区客户端需要在用户进入登录界面点击登录按钮时引导用户到新浪微博平台进行授权。Web 客户端直接将页面跳转到新浪微博平台进行授权,手机客户端引导用户打开 google 内置的浏览器程序,跳转到新浪微博开放平台的授权地址进行授权。在此过程中客户端需要向新浪微博开放平台发起用户未授权令牌申请、用户已授权令牌申请、访问令牌申请,客户端得到访问令牌后即可调用新浪微博的用户资源。如图 4 所示为读书分享社区与新浪微博服务端交互的用例图。

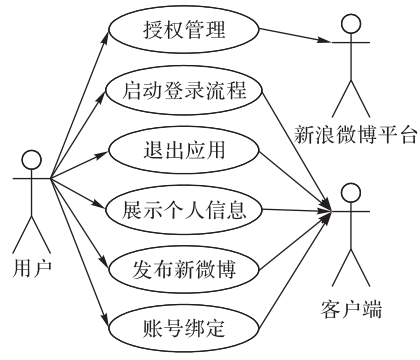

图 3 读书分享社区授权认证模块用户与客户端和新浪微博平台交互的用例图

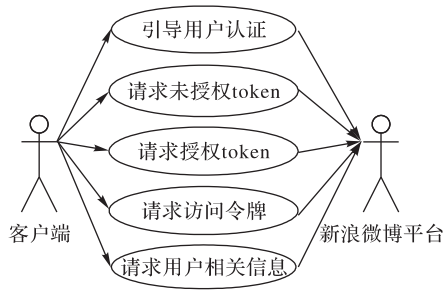

图 4 读书分享社区客户端与新浪微博服务端交互用例

2.4 类设计

根据用例分析,鉴于 Web 客户端和手机客户端在进行发送 Http 请求的过程上略有不同,而且 PHP 语言相对于 Java 语言在编写代码上更为灵活的特性,我们对 Web 客户端和手机客户端分别进行类设计,安卓手机客户端类图如图 5 所示。

在手机客户端系统中创建两个不同的令牌类:AccessToken 和 RequestToken 表示客户端需要请求的令牌,这两个类都有共同的操作 getToken、setExpires、getParameter、setVerifier 用来获取令牌信息、设置有效期、获取参数和设置标识符,因此我们将共同的功能放在超类 Token 中。在安卓手机客户端定义一个 HttpConnectUtility 类用于发送 Http 请求。Constant 类定义了 AppKey、AppSecert 和 RequestUri 等属性来标识客户端的信息、客户端的密钥以及客户端的回调地址标识客户端信息。Weibo 类和 WeiboDialog 类是具体的业务逻辑类;OauthActivity 类完成控件的显示,并对用户的操作做出响应;SharePreference 类将获取的用户信息存储安卓手机中。

在 Web 客户端用户直接通过点击 HTML 页面的 a 标签连接或提交表单即可发送 Http 的 GET 和 POST 请求,即使用户不进行操作,程序也可以通过调用 Curl 工具包的函数发送 Http 请求,因此我们省略了 Http 类的设计。Web 客户端类图如图 6 所示。

Web 客户端主要包含 SaeOauth2 类、SaeClient 客户端类、UserModel 用户模型类、Config 配置信息类、以及 Session 保存用户信息类。SaeOauth2 类用于请求临时令牌、请求访问令牌等业务逻辑。UserModel 类主要完成请求用户的信息和用户的微博信息等业务逻辑。Config 配置信息类包括客户端的标识、用户请求授权的的地址、访问令牌的地址和客户端的回调

地址等主要信息。LoginAction、CallbackAction 和 UserAction 都继承于 Action 主控制类,处理用户的请求操作。这种多态的方法使得系统更容易扩展,创建新的用户操作类只需创建新的 Action 子类即可完成。

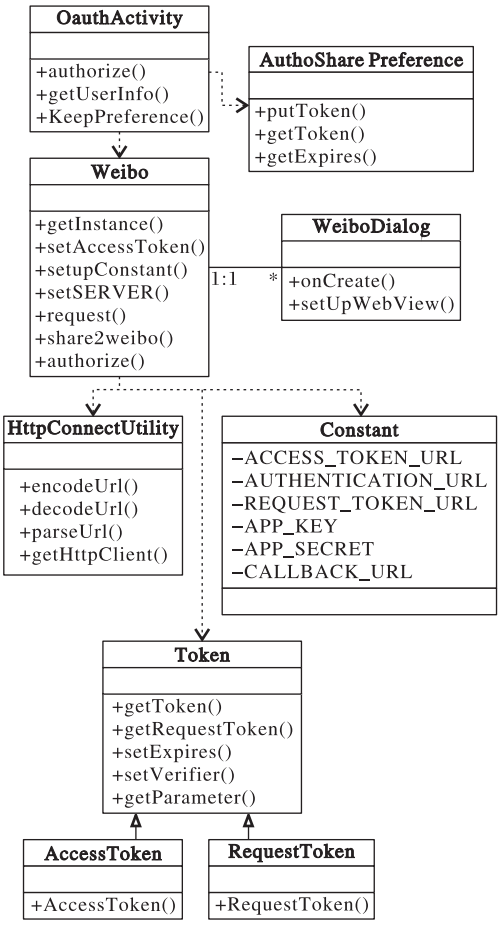

图 5　安卓手机客户端类图

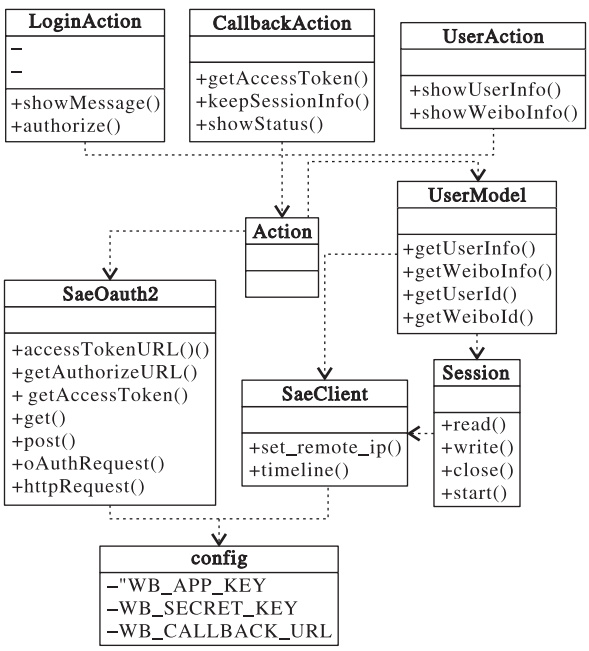

图 6　Web 客户端类图

3　具体实现

3.1　实现平台与环境

本课题 Web 客户端基于 WAMP 平台,使用 PHP 语言进行实现;手机客户端基于 Android 平台,使用 Java 语言进行实现。完成该课题的完整开发环境如表 6 所述。

表 6　实现平台与环境

名称	平台环境
操作系统	Windows7, sp1
开发工具	Eclipse Ide for Java Develpers
插件	Android Development Tools(ADT), Plugin for Eclipse(安卓手机客户端)
开发语言	Java(安卓手机客户端), PHP(Web 客户端)

3.2　关键功能实现

Web 端请求授权认证时需要引入配置信息和 SaeOAuth2 业务处理类,通过调用 SaeOAuth2 类中的 getAccessToken 方法获取新浪微博的访问令牌,该方法封装了获取获取临时令牌和授权码等内部业务逻辑,完成 Web 客户端请求授权认证代码如下:

```
<?php
include_once('config.php');
include_once('SaeOAuth2.class.php'); $OAuth =
  new SaeOAuth(APP_KEY, APP_SECRET);
if(isset($_REQUEST['code'])){
  $keys = array();
  $keys['code'] = $_REQUEST['code'];
  $keys['redirect_uri'] = WB_CALLBACK_URL;
  try{
    $token = $OAuth->getAccessToken('code', $key);
  }catch(OAuthException $e){
    $this->error("发送请求错误");
  }if($token){
    $_SESSION['token'] = $token;
    setcookie('weibojs_'.$o->client_id, http_build_query($
      token));
      $successUrl = SHARE_PATH."?m=user&a=showInfo";
      $message = "授权完成,<a href='.user.php'>"进入你
      的微博列表页面</a><br/>3 秒后跳转";
    $this->success($message, $successUrl);
  }else{
    $message = "授权失败";
    $this->error($message);
  }
}
```

客户端引导用户到新浪微博登录界面,新浪微博平台会向用户显示授权读书分享社区访问帐号,界面如图 7 所示。

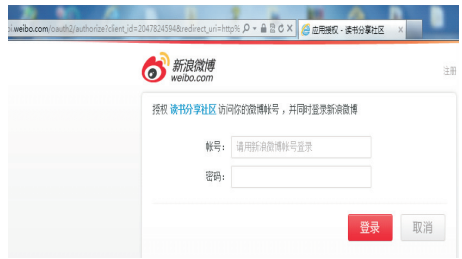

图 7　Web 客户端请求授权认证的界面

Web 端获取用户信息后通过 Session 将用户信息保存到 Session 中,调用 SaeClient 的 home_timeline 方法可以获得用户的基本信息,具体实现如图 8 所示。Web 客户端获取用户信息示例如下:

```
<?php
session_start();                                    //包含加载文件
```

```
require_once( SHARE_PATH. 'config. class. php');
require_once( SHARE_PATH. 'saetv2. ex. class. php');
$ saeClient = new SaeClient( APP_KEY, APP_SECRET,
$ _SESSION[ 'token'][ 'access_token']);
$ message = $ saeClient -> home_timeline();
$ uid_get = $ sae -> get_uid();
$ uid = $ uid_get[ 'uid'];
$ user_message = $ sae -> show_user_by_id( $ uid);
```

用户登录成功后进入读书分享社区的用户个人空间,界面如图 8 所示。

图 8 用户登录成功界面

安卓手机客户端用户点击授权按钮,在 OauthActivity 类中给该按钮绑定监听器,调用 Weibo 对象的 authorize 方法进行授权,具体实现如下:

```
authBtn = ( Button) findViewById( R. id. auth);
authBtn. setOnClickListener( new OnClickListener() {
  public void onClick() {
    mWeibo. authorize( MainActivity. this,
      new AuthDialogLisener());
}};
```

安卓手机客户端程序将用户引导到手机内置浏览器程序的新浪微博地址,同 Web 客户端的界面相同,如图 9 所示。

图 9 安卓手机客户端请求授权界面

安卓手机客户端授权认证成功处理 WeiboDialogListener 中的 onComplete 方法,如果授权认证失败则调用该类的 onCancel 方法,具体实现如下:

```
Bundle value = Utility. parseUrl( url);
String error = value. getString( "error");
String errot_code = value. getString( "error_code");
if ( error == null && error_code == null) {
  mListener. onComplete( values);
} elseif( error. equals( "access_denied") {
  //用户获授权服务器拒绝数据库访问
  mListener. onCancel();
}
```

4 结语

社会化网络已经成为人们生活的重要组成部分,如今无论是 Web 应用还是移动应用,实时的交互和广泛的传播成为互联网用户的主要需求之一,即时分享成为了社会化网络时尚的标签。读书分享社区是一个基于社会化网络的阅读系统,本文分析了 OAuth2.0 协议,实现了用户使用新浪微博账号登录到读书分享社区系统的功能,简化了用户登录流程、提高读书分享社区的社会化程度。下一步的工作是完成读书分享社区总体功能,实现一个社会化元素丰富、用户体现度高的读书社交系统。

参考文献:

[1] 朱睿敏, 谢东亮. SIP 在移动 SNS 架构中的应用简析[J]. 中国科技论文在线, 2007, 36(4):577 - 580.

[2] 刘文娟, 袁文芳. 校内网的 SNS 人际传播特征分析[J]. 东南传播, 2009(5):129 - 131.

[3] 许志敏, 薛质. 授权认证系统的应用研究[J]. 中国传媒科技, 2006(4):28 - 31.

[4] 何洪波, 黄文, 王闰强. 中国科学院网络化科学传播平台门户的设计理念和相关技术[C]// 数字博物馆研究与实践(2009). 北京: 中国传媒大学出版社, 2010:206 - 210.

[5] 谭晨辉, 刘青炎. OpenAPI 出现、起源与现状[J]. 程序员, 2008(7):38 - 41.

[6] 钱丹浩. 项目化嵌入式教学的开发系统平台构建[J]. 单片机与嵌入式系统应用, 2010(11):22 - 24, 35.

[7] Wikip edia OpenID[EB/OL]. [2013 - 08 - 01]. http://zh. wikipedia. org/wiki/OpenID.

[8] 刘大红, 刘明. 第三方应用与开放平台 OAuth 认证互连技术研究[J]. 电脑知识与技术, 2012, 8(22): 5367 - 5369.

[9] 付韬, 马春光, 李迎涛, 等. 基于开放平台的 OAuth 认证授权技术研究[J]. 保密科学技术, 2012(9):58 - 62.

[10] 时子庆, 刘金兰, 谭小华. 基于 OAuth 的认证授权技术[J]. 计算机系统应用, 2012, 21(3):1 - 4.

[11] 刘镝, 张智江, 张尼. 基于国内开放平台的 OAuth 的认证框架研究[J]. 信息通信技术, 2011(6):60 - 64.

[12] 熊静, 喻钢, 张旭, 等. 基于面向对象与构件技术的联锁逻辑的仿真与建模方法研究[C]// 2008 中国信息技术与应用学术论坛论文集(二). 重庆:《计算机科学》杂志社, 2008.

[13] 王晓光. 微博客用户行为特征与关系特征实证分析——以"新浪微博"为例[J]. 图书情报工作, 2010, 54(14):66 - 70.

(上接第 49 页)

[6] ZHANG Y, GUO H. An improved RFID privacy protection scheme based on Hash-chain [C]// Proceedings of the 2010 International Conference on Logistics Engineering and Intelligent Transportation Systems. New York: IEEE Press, 2010:1 - 4.

[7] 杨超, 张红旗. 基于秘密共享方案 RFID 认证协议[J]. 计算机应用, 2012, 32(12):194 - 197.

[8] 钟杰卓. 基于 Hash 链的 RFID 安全协议研究与设计[J]. 现代计算机: 专业版, 2010(8):139 - 141.

[9] 熊宛星, 薛开平, 洪佩琳, 等. 基于二维区间 Hash 链的 RFID 安全协议[J]. 中国科学技术大学学报, 2011, 41(7):594 - 598.

[10] 马巧梅. 基于 Hash 链的 RFID 改进协议[J]. 网络安全技术与应用, 2012(7):62 - 64.

[11] DIMITRIOU T. A lightweight RFID protocol to protect against traceability and cloning attacks [C]// Proceedings of the 1st International Conference on Security and Privacy for Emerging Areas in Communication Networks. Washington, DC: IEEE Computer Society, 2005:59 - 66.

[12] 张顺, 陈海进. 轻量级的无线射频识别安全认证协议[J]. 计算机应用, 2012, 32(7):230 - 234.