ICS 01.140.20
A 14

C A D A L 项 目 标 准

CADAL 20902—2012

数字图书馆访问控制规范

Digital Library Access Control Specification

第一稿

2012-05-08 发布　　2012-05-09 实施

CADAL 项目管理中心　　发　布

目　　次

前　言

《数字图书馆安全标准规范集》包括以下 4 个部分：

——第 1 部分：数字图书馆数字对象存储安全规范；

——第 2 部分：数字图书馆访问控制规范；

——第 3 部分：数字图书馆数字资源长期保存规范；

——第 4 部分：数字图书馆安全传输标准。

本标准是其中的第 2 部分。

本部分的制定依据了标准化工作导则第 1 部分(GB/T 1.1—2009)。

本部分是由大学数字图书馆国际合作计划(CADAL)项目管理中心提出。

本标准由 CADAL 项目管理中心归口。

本部分起草单位：数字图书馆教育部工程研究中心。

本部分起草人：洪鑫、张寅、王宇奇。

引　　言

数字资源的访问控制是数字图书馆中实施版权保护的一个重要环节。

本标准是在 CADAL 数字图书馆数字资源访问控制实践的基础上编制的。

本标准充分说明了符合 CADAL 项目的要求数字资源以及用户数据访问方式，为今后的数字图书馆访问控制提供了可参考的规范。

数字图书馆访问控制规范

1 范围

本标准确定了数字资源访问控制规范。

本规范规定了数字资源访问时所要遵循的安全规范和协议。

本规范适用于数字资源的安全管理，适用于对数字资源的访问过程进行安全管理。

本规范涉及的数据包括门户的数字资源及用户数据。

2 规范性引用文件

下列文件对于本文件的应用是必不可少的。凡是注日期的引用文件，仅所注日期的版本适用于本文件。凡是不注日期的引用文件，其最新版本(包括所有的修改单)适用于本文件。

The OAuth 2.0 Protocol Framework

3 术语和定义

3.1 明文 Clear-Text

明文是待伪装或加密的消息。在通信系统中它可能是比特流，如文本、位图、数字化的语音或者数字化的视频图像等。一般可以简单地认为，明文是有意义的字符或比特集，或通过某种公开的编码标准就能获得的消息。

3.2 数字签名 Digital Signature

数字签名是以电子形式存在于数据信息之中的，或作为其附件的或逻辑上与之有联系的数据，可用于辨别数据签署人的身份，并表明签署人对数据信息中包含的信息的认可。

3.3 互联网数字分配机构 The Internet Assigned Numbers Authority 缩写：IANA

互联网数字分配机构是负责协调一些使 Internet 正常运作的机构。同时，由于 Internet 已经成为一个全球范围的不受集权控制的全球网络，为了使网络在全球范围内协调，存在对互联网一些关键的部分达成技术共识的需要，而这就是 IANA 的任务。

3.4 访问控制 Access Control 缩写：AC

访问控制是指按用户身份及其所归属的某项定义组来限制用户对某些信息项的访

问，或限制其对某些控制功能的使用。访问控制通常用于系统管理员控制用户对服务器、目录、文件等网络资源的访问。功能主要是：防止非法的主体进入受保护的网络资源，允许合法用户访问受保护的网络资源，防止合法的用户对受保护的网络资源进行非授权的访问。

4 访问控制规范

本标准中的访问控制对象包括数字资源以及用户数据，采用访问控制规则的方式来定义哪些用户可以访问哪些数字资源。具体的访问控制方式包括 IP 控制、用户名密码控制。IP 控制指只有来自参建机构 IP 范围的用户才能够访问数字资源；用户名密码控制是指获得授权的用户在提供正确的用户名和密码之后，可以访问授权的资源。用户自身的隐私数据必须经过用户的授权，才能够让第三方应用访问和使用。对用户隐私数据的访问控制，本标准推荐采用 OAUTH 2.0 认证技术。

4.1 数字资源访问控制标准

4.1.1 IP 控制

IP 控制是指只有来自参建机构的 IP 范围的用户才能够访问数字资源。每一个参建单位会分配一个唯一标识符。每个标识符对应的 IP 段也会被单独存放在一个数据库表中。

IP 组与学校对应关系见表 1。IP 组号与 IP 地址段对应关系见表 2。

表 1 IP 组与学校对应关系

字段名	含 义
GroupID	参建机构唯一标识符
Name	参建机构名称

表 2 IP 组号与 IP 地址段对应关系

字段名	含 义
IpAddress	IP 段
GroupID	所属机构 ID

当有新的机构加入参建单位时，要及时更新门户服务器的 IP 库，以保证新增机构的用户可以访问门户的数字资源。数字资源访问控制整体流程如图 1 所示。

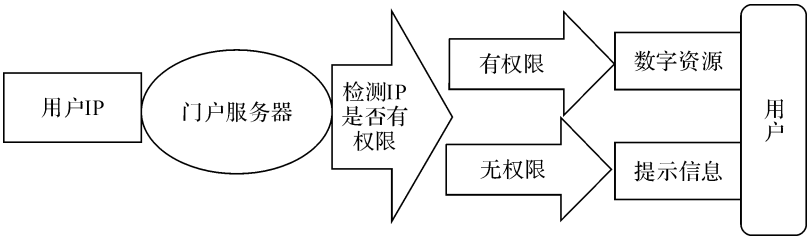

图 1　数字资源访问控制流程

4.1.2　用户名密码控制

用户名密码控制过程中，用户只有提供正确的用户名与密码，才能够访问授权的资源。总门户服务器的数据库中会存放相应的用户名密码以及其他一些与用户相关的重要信息。具体的信息记录见表 3。

表 3　用户信息

字段名	含　义
UserId	用户唯一标识符
UserName	用户名
Password	密码
School	用户所在的学校
Country	用户所在的国家
Occupation	用户职业

4.2　用户数据访问控制标准

本标准要求用户数据的访问控制采用 OAuth 2.0 认证技术。

首先第三方需要到 CADAL 上申请一个 client_id，对应一个 client_key。这些信息在服务器端是存放在 oauth_user 这个数据库的表中的，详见后文表 4。

4.2.1　步骤一

第三方在自己的网页上将 CADAL 的用户引导到 CADAL 上，同时提供一个重定向的 URI（这个 URI 是用户在决定是否授权给该第三方应用后，第三方应用需要跳转的地址）。

登录流程开始于重定向用户浏览器到 CADAL OAuth 2.0 的 Authorize Endpoint，必须传递三个参数：

(1)client_id：必须参数。在 CADAL 注册时获得的 API Key。

(2)redirect_uri：流程结束后要跳转回的 URL. redirect_uri 所在的域名必须上报 CADAL 开发者，以便 CADAL 开发者检查跳转的合法性。

(3)response_type：必须参数。这里固定为“code”。

http: //www. cadal. zju. edu. cn/oauthAPI/authorize? client_id＝YOUR_API_KEY&

redirect_uri =YOUR_CALLBACK_URL&response_type=code.

用户在 CADAL 服务器上输入证书(用户名与密码),此时用户的合法性是在 CADAL 服务器进行验证的(服务端需要从数据库中查询该用户),第三方无法看到用户名与密码,所以这一步是安全的。(这一步在 OAuth 2.0 中是核心,因为用户的私有证书是在服务器端进行验证的,对第三方来说是透明的。)

如果用户已经登录,CADAL OAuth 2.0 会检验存储在用户浏览器中的 cookie;如果用户没有登录,CADAL OAuth 2.0 会为用户展示登录页面,让用户输入用户名与密码。

用户授权第三方,服务器将生成一个授权码(authorization code)并发送给第三方应用,同时服务器端将此授权码存到数据库中的 oauth_auth 表中。

如果用户不同意授权(点击关闭),应用将不会被授权。CADAL OAuth 2.0 会将用户的浏览器重定向(通过 HTTP 302)到 redirect_uri 参数对应的 URL 上,并在 Query 中带上相应的错误信息。

http: //YOUR_CALLBACK_URL? error=nvalid_request&error_description=The+request+is+missing+a+required+parameter: +client_id.

如果用户同意授权(点击连接),应用将会被授权。CADAL OAuth 2.0 会将用户的浏览器重定向(通过 HTTP 302)到 redirect_uri 参数对应的 URL 上,并在 Query 中使用'code'参数返回一个 Authorization Code。

http: //YOUR_CALLBACK_URL? code=A_CODE_GENERATED_BY_SERVER.

Authorize Code 可以在 redirect_uri 后端程序中获得(例如 Java request. getParameter("code");)。获得到 Authorization Code 后,你可以进行流程的下一步——应用验证。

4.2.2 步骤二

第三方将自己的 client_id 和 client_key 与授权码同时发送给服务器端,以换取访问令牌(access token)。

应用验证需要使用 HTTP POST 请求 CADAL OAuth 2.0 的 Access Token Endpoint,还需要带上一系列需要的参数:

grant_type:使用 Authorization Code 作为 Access Grant 时,此值为"authorization_code"。

code:上述过程中获得的 Authorization Code;

client_id:在 CADAL 注册应用时获得的 API Key;

client_secret:在 CADAL 注册应用时获得的 Secret Key。

redirect_uri:必须与获取 Authorization Code 时传递的"redirect_uri"保持一致。

https: //www. cadal. zju. edu. cn/oauth/token? grant_type=authorization_code&client_id=YOUR_API_KEY&redirect_uri=YOUR_CALLBACK_URL&client_secret=YOUR_SECRET_KEY&code=THE_CODE_FROM_ABOVE.

服务器端在验证授权码正确的情况下将 Access Token 发送给第三方,并且在数据库中的 oauth_token 表中存储该访问令牌。

http: //YOUR_CALLBACK_URL? token_code=ACCESS_TOKEN.

如果应用验证通过(client_id 与 client_secret 匹配,redirect_uri 与获取 Authorization

Code 时传递的 redirect_uri 保持一致），并且从用户获取的 Authorization Code 也正确，CADAL OAuth 2.0 会返回 Access Token 相关的信息：

{

"access_token": "10000|5. a6b7dbd428f731035f771b8d15063f61. 86400. 1292922000－222209506",

"expires_in": 87063,

"refresh_token": "10000|0. 385d55f8615fdfd9edb7c4b5ebdc3e39－222209506",

}

access_token: 获取的 Access Token；

expires_in: Access Token 的有效期，以秒为单位；

refresh_token：用于刷新 Access Token 的 Refresh Token，长期有效，不会过期。

如果应用验证过程中出错，人人 OAuth 2.0 将返回 HTTP 401（应用验证失败）或 HTTP 400（参数错误），并在 HTTP Body 中返回错误信息

{

"error": "invalid_request",

"error_description": "The request is missing a required parameter: client_id"

}

4.2.3 步骤三

第三方通过访问令牌即可到服务器获取到 session_key，服务器端用 oauth_session 表存放 session_key。

第三方通过 session_key 调用所需的 API 返回数据（这个 API 需要服务器端完成，将用户的个人相关信息返回给第三方）。

4.2.4 数据库表标准设计

为了使用方便，在服务器部分设计了四个数据表：oauth_user、oauth_auth、oauth_token 和 oauth_session，其中 oauth_auth、oauth_token、oauth_session 三个表格并不需要永久存储，他们的有效期都只有一个小时，可以通过缓存等方式重新处理，但是当前为了设计方便，暂存储于数据库。由于这些 code 过期或者使用过后将不再有效，所以没有存储的必要，在 CADAL 当前的设计中，是定期删除该数据库中过期或者失效的数据。

4.2.4.1 oauth_user

存储应用的 API id 和 API key，只有申请通过的用户才可以使用 API，包括四个字段：client_id 表示应用 API id，client_key 表示 API key，username 表示申请应用 API 的用户在 CADAL 门户网站的用户名，datetime 表示申请时间，详见表 4。

表 4　oauth_user

Column	类　型	非　空
client_id	character varying(50)	NOT NULL
client_key	character varying(50)	NOT NULL
username	character varying(50)	NOT NULL
datetime	timestamp with time zone	

4.2.4.2　oauth_auth

存储一段时间内的授权码，包括四个字段：client_id 表示应用 API id；client_key 表示 API key；authorization 表示 code，每次都通过 UUID 随机生成，基本可认定不会出现重复 code；datetime 表示该 code 生成的时间；详见表 5。其中，authorization 表示图中的授权码，即服务器会将此码发送给应用，应用要凭这个授权码到服务器换取访问令牌。

表 5　oauth_auth

Column	类　型	非　空
client_id	character varying(50)	NOT NULL
client_key	character varying(50)	NOT NULL
username	character varying(50)	NOT NULL
datetime	timestamp with time zone	NOT NULL

4.2.4.3　oauth_token

存储一段时间内的访问令牌，包括三个字段：oauth_token 表示 token code，used 表示是否已被使用，datetime 表示该 code 生成的时间，详见表 6。

表 6　oauth_token

Column	类　型	非　空
oauth_token	character varying(50)	NOT NULL
used	character varying(2)	NOT NULL
datetime	timestamp with time zone	NOT NULL

4.2.4.4 oauth_session

存储一段时间内的 session key，应用需要通过该 session key 调用所需的 API，在这一步当中还缺少 refresh token，可以刷新该 session key 的二次使用，目前还未实现；包括三个字段：session_key 表示 session key，used 表示是否已被使用，datetime 表示该 code 生成的时间，详见表 7。

表 7 oauth_session

Column	类 型	非 空
session_key	character varying(50)	NOT NULL
used	character varying(2)	NOT NULL
datetime	timestamp with time zone	NOT NULL

参 考 文 献

[1] The OAuth 2.0 Protocol Framework[DB/OL]. http: //oauth.net/2/.

[2] IANA The Internet Assigned Numbers Authority 互联网数字分配机构[DB/OL]. http: www.iana.org.