



中华人民共和国国家标准

GB/T 36610—2018

用于微博客的法人和其他组织统一社会 信用代码实名认证服务接口规范

Service interface specification for microblogs of legal entities and other
organizations certification by unified social credit identifier

2018-09-17 发布

2019-04-01 实施

国家市场监督管理总局
中国国家标准化管理委员会 发布

目 次

前言	III
引言	IV
1 范围	1
2 规范性引用文件	1
3 术语和定义	1
4 服务接口	2
附录 A (资料性附录) 数据加解密示例	9
参考文献	13

前 言

本标准按照 GB/T 1.1—2009 给出的规则起草。

本标准由全国信息分类与编码标准化技术委员会(SAC/TC 353)提出并归口。

本标准起草单位:全国组织机构代码数据服务中心、中国电子技术标准化研究院、辽宁省标准化研究院、哈尔滨市标准化研究院、黑龙江省标准化研究院、江苏省质量和标准化研究院、济南市质量技术监督局、辽宁凯隆标准化技术服务有限公司。

本标准主要起草人:孙镇、赵捷、孙泰、柯志勇、袁辉、金江、徐克超、钱晓东、司琳华、李晟飞、宫政、李一峰、孟炬、施晓林、徐一鸣、朱峰、姜欣、张劲男、贺君。

引 言

本标准详细描述了统一社会信用代码、组织机构代码数据服务部门向微博客商业网站提供组织机构微博客实名信息对比的要求,主要包括接口调用、数据定义、数据加密等要求。统一社会信用代码、组织机构代码作为组织机构的“身份证”,通过使用统一社会信用代码、组织机构代码库信息用于微博客实名认证,无论对微博客网站、组织机构来说都有重要的意义,是深化微博客实名制应用,以促进微博健康发展的关键。

用于微博客的法人和其他组织统一社会信用代码实名认证服务接口规范

1 范围

本标准规定了用于微博客的法人和其他组织统一社会信用代码实名认证服务接口的格式规范,详细定义了微博客商业网站访问统一社会信用代码、组织机构代码信息资源的接口要求,包括:接口调用、数据加密要求、格式要求、数据类型定义等。

本标准适用于所有访问统一社会信用代码、组织机构代码信息的微博客商业网站在接受组织机构微博账户注册时,进行信息对比认证的过程。

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。

GB 11714—1997 全国组织机构代码编制规则

GB 32100—2015 法人和其他组织统一社会信用代码编码规则

3 术语和定义

下列术语和定义适用于本文件。

3.1

微博客 microblog

一种通过关注机制分享咨询的广播式的社交网络平台。

3.2

微博客网站 microblog website

承载微博客系统的门户网站。

3.3

组织机构 organization

企业、事业单位、机关、社会团体及其他依法成立的单位的通称。

[GB/T 20091—2006,定义 2.2]

3.4

组织机构代码 organization code

主体标识码 subject identification code

按照 GB 11714 编制,赋予每一个组织机构在全国范围内唯一的、始终不变的识别标识码。

[GB 32100—2015,定义 3.4]

3.5

统一社会信用代码 unified social credit identifier

每一个法人和其他组织在全国范围内唯一的、终身不变的身份识别码。

[GB 32100—2015,定义 3.5]

3.6

服务接口 service interface

计算机软件系统之间进行共享数据、交换信息的支持程序,以报文的形式实现。

3.7

网络服务描述语言 web services description language; WSDL

一种可扩展置标语言(XML)应用,它将 Web 服务描述定义为一组服务访问点,使用者可以通过这些服务访问点对包含面向文档信息或面向过程调用的服务进行访问。

4 服务接口

4.1 实名备案接口

4.1.1 概述

实名备案接口用于对接微博客网站,通过备案后可进行后续实名认证服务,在接口使用过程中应进行数据加密、解密处理。数据加解密示例,参见附录 A。

4.1.2 接口定义

实名备案:public String recordDMInfo(String operatorCode,String opInfo)。

调用时应完全吻合方法名称的大小写、请求参数个数、请求参数的位置和请求参数类型。

4.1.3 接口参数说明

实名备案接口参数说明,见表 1。

表 1 实名备案接口参数

序号	参数名	数据类型	描述
1	operatorCode	String	微博客运营商实名备案申请表中统一社会信用代码或组织机构代码,省略连字符(-)
2	opInfo	String	规定格式的 XML 标准字符串

4.1.4 接口请求规范

实名备案接口请求信息应符合表 2 的规定。

表 2 实名备案请求 opInfo

实名备案请求 opInfo
<pre> <? xml version="1.0" encoding="UTF-8" ?> <record> <WZMC>网站名称</WZMC > <WZYM>网站域名</WZYM> <ZZJGMC>申请单位名称</ZZJGMC> <ZZJGDM>组织机构代码</ZZJGDM> <TYDM>统一社会信用代码</TYDM> </pre>

表 2 (续)

实名备案请求 opInfo
<pre> <FDDBR>法定代表人姓名</FDDBR> <ZJHM>法定代表人证件号码</ZJHM> <ZJLX>证件类型</ZJLX> <JGDZ>机构地址</JGDZ> <SQR>申请人姓名</SQR> <SQR_ZJHM>申请人证件号码</SQR_ZJHM> <MEMO>备用字段</MEMO> </record> </pre>

4.1.5 接口返回规范

实名备案接口返回信息应符合表 3 的规定。

表 3 实名备案返回 recordDMInfo

实名备案返回 recordDMInfo
<pre> <? xml version="1.0" encoding="UTF-8" ?> <record> <BAZT>备案状态</BAZT> <ZTSM>备案状态说明</ZTSM> <BAH>备案号</BAH> <WZYM>网站域名</WZYM> <MEMO>备用字段</MEMO> </record> </pre>



全国组织机构代码统一社会信用代码数据服务中心
<https://www.cods.org.cn>

4.2 实名认证接口

4.2.1 概述

实名认证接口用于对接微博客用户,微博客用户在微博客网站完成注册后跳转至实名认证接口进行实名认证,实名认证完成后将实名认证结果反馈至微博客网站,在接口使用过程中应进行数据加密、解密处理。数据加解密示例,参见附录 A。

4.2.2 接口定义

实名认证:public String verifyDMInfo(String companyCode,String dmInfo)。

调用时应完全吻合方法名称的大小写、请求参数个数、请求参数的位置和请求参数类型。

4.2.3 接口参数说明

实名认证接口参数说明,见表 4。

表 4 实名认证接口参数

序号	参数名	数据类型	描述
1	companyCode	String	微博客网站接收组织机构类用户申请微博客账号的申请表中填写的统一社会信用代码或组织机构代码,省略连字符(-)
2	dmInfo	String	规定格式的 XML 标准字符串

4.2.4 接口请求规范

实名认证接口请求信息应符合表 5 的规定。

表 5 实名认证请求 dmInfo

实名认证请求 dmInfo
<pre> <? xml version="1.0" encoding="UTF-8" ?> <verify> <WBWZBM>组织机构微博账户</WBWZBM> <ZZJGDM>组织机构代码</ZZJGDM> <TYDM>统一社会信用代码</TYDM> <ZZJGMC>组织机构名称</ZZJGMC> <FDDBR>法定代表人姓名</FDDBR> <ZJHM>法定代表人证件号码</ZJHM> <ZJLX>证件类型</ZJLX> <JGDZ>机构地址</JGDZ> <SQR>申请人姓名</SQR> <SQR_ZJHM>申请人证件号码</SQR_ZJHM> <MEMO>备用字段</MEMO> </ verify > </pre>

4.2.5 接口返回规范

实名认证接口返回信息应符合表 6 的规定。

表 6 实名认证返回 verifyDMInfo

实名认证返回 verifyDMInfo
<pre> <? xml version="1.0" encoding="UTF-8" ?> <VERIFY> <FHBM>请求返回状态码</ FHBM> <JGSM >状态码说明信息</ JGSM > <BDHZM >比对回执码</ BDHZM > <WBWZBH>组织机构微博账户</ WBWZBH> <MEMO>备用字段</MEMO> </VERIFY> </pre>

4.3 冒用申诉接口

4.3.1 概述

冒用申诉接口用于冒用申诉,微博客用户在微博客网站提交冒用申诉后跳转至冒用申诉接口进行实名认证,实名认证完成后将实名认证结果反馈至微博客网站,在接口使用过程中应进行数据加密、解密处理。数据加解密示例,参见附录 A。

4.3.2 接口定义

冒用申诉:public String appealDMInfo(String appealCode,String apInfo)。

调用时应完全吻合方法名称的大小写、请求参数个数、请求参数的位置和请求参数类型。

4.3.3 接口参数说明

冒用申诉接口参数说明,见表 7。

表 7 冒用申诉接口参数

序号	参数名	数据类型	描述
1	appealCode	String	微博客网站接收组织机构类用户申诉微博客账号被冒用的申请表中填写的统一社会信用代码或组织机构代码,省略连字符(-)
2	apInfo	String	规定格式的 XML 标准字符串

全国组织机构代码统一社会信用代码数据服务中心
<https://www.cods.org.cn>

4.3.4 接口请求规范

冒用申诉接口请求信息应符合表 8 的规定。

表 8 冒用申诉请求 apInfo

冒用申诉请求 apInfo
<pre><? xml version="1.0" encoding="UTF-8" ?> <appeal> <WBWZBM>组织机构微博账户</WBWZBM> <ZZJGDM>组织机构代码</ZZJGDM> <TYDM>统一社会信用代码</TYDM> <ZZJGMC>组织机构名称</ZZJGMC> <FDDBR>法定代表人姓名</FDDBR> <ZJHM>法定代表人证件号码</ZJHM> <ZJLX>证件类型</ZJLX> <JGDZ>机构地址</JGDZ> <SQR>申请人姓名</SQR> <SQR_ZJHM>申请人证件号码</SQR_ZJHM> <MEMO>备用字段</MEMO> </appeal></pre>

4.3.5 接口返回规范

冒用申诉接口返回信息应符合表 9 的规定。

表 9 冒用申诉返回 appealDMInfo

冒用申诉返回 appealDMInfo
<pre> <? xml version="1.0" encoding="UTF-8" ?> < appeal > <FHBM>请求返回状态码</ FHBM> < JGSM >状态码说明信息</ JGSM > < BDHZM >比对回执码</ BDHZM > < WBWZBH>组织机构微博账户</ WBWZBH> <MEMO>备用字段</MEMO> </ appeal > </pre>

4.4 数据类型定义

4.4.1 概述

对实名备案接口、实名认证接口和冒用申诉接口使用过程中涉及的数据类型进行定义。

4.4.2 请求信息

请求信息数据类型定义应符合表 10 的规定。

表 10 请求信息表

节点名称	中文名称	含义说明	数据类型	值域
WZMC	网站名称	微博客运营商的网站名称	字符串	必填项,长度限制在 130 字符内,加密
WZYM	网站域名	微博客运营商的网站域名	字符串	必填项,长度限制在 130 字符内,加密
WBWZBH	组织机构微博账户	组织机构在商业网站申请的微博账户	字符串	必填项,长度限制在 50 字符内,加密
JGDM	组织机构代码	申请微博账户关联的组织机构代码	字符串	必填项,长度为 9 字符,应符合 GB 11714—1997 规定,加密
TYDM	统一社会信用代码	申请微博账户关联的统一社会信用代码	字符串	必填项,长度为 18 字符,应符合 GB 32100—2015 规定,加密
JGMC	组织机构名称	申请微博账户关联的组织机构名称	字符串	必填项,长度限制在 130 字符内,加密
FDDBR	法定代表人姓名	申请微博账户关联的组织机构法人姓名	字符串	必填项,长度限制在 130 字符内,加密

表 10 (续)

节点名称	中文名称	含义说明	数据类型	值域
ZJHM	法定代表人证件号码	申请微博账户关联的法定代表人证件号码	字符串	必填项,长度限制在 50 字符内,加密
ZJLX	证件类型	申请微博账户关联的法定代表人证件类型	字符串	必填项,长度限制在 20 字符内,加密
JGDZ	机构地址	申请微博账户关联的组织机构注册地址	字符串	必填项,长度限制在 200 字符内,加密
SQR	申请人姓名	组织机构微博账户申请人姓名	字符串	必填项,长度限制在 130 字符内,加密
SQR_ZJHM	申请人证件号码	组织机构微博账户申请人有效身份证件号码	字符串	必填项,长度限制在 50 字符内,加密
MEMO	备用字段	扩展位,备用	字符串	非必填,为空时项节点保留,长度限制在 120 字符内,非空时加密

4.4.3 返回信息

全国组织机构代码统一社会信用代码数据服务中心
<https://www.cods.org.cn>

返回信息数据类型定义应符合表 11 的规定。

表 11 返回信息表

英文名称	中文名称	含义	数据类型	值域
BAZT	备案状态	微博客运营商申请备案的状态	字符串	必填项,长度限制在 10 字符内,加密
ZTSM	备案状态说明	备案状态的原因说明	字符串	必填项,长度限制在 150 字符内,加密
BAH	备案号	微博客运营商备案号	字符串	必填项,长度限制在 50 字符内,加密
FHBM	请求返回状态码	微博商业网站请求服务后返回的比对结果	字符串	必填项,长度为 2 位,加密
JGSM	状态码说明信息	微博商业网站请求服务后返回的比对结果的详细说明	字符串	必填项,长度限制为 130 位,加密
BDHZM	比对回执码	当比对结果一致时,由全国组织机构代码数据服务中心服务器生成的与组织机构唯一对应的回执码;当比对结果不一致时空值	字符串	选填项,长度限制为 100 位,非空时加密

表 11 (续)

英文名称	中文名称	含义	数据类型	值域
WBWZBH	组织机构微博账户		字符串	必填项,长度限制为 50 位,加密
MEMO	备用字段		字符串	非必填项,为空时节点保留,长度限制在 120 字符内,加密

4.4.4 状态信息

比对结果中返回状态码字段定义应符合表 12 的规定。

表 12 状态码表

状态码	含义
0	信息匹配一致
1	信息匹配不一致
2	代码证书不在有效期
3	代码证书已失效
-1	系统异常
-2	服务器 IP 校验失败
-3	companyCode 无效
-4	XML 结构不完整
-5	×××节点值 AES 解密错误
-6	×××必填项节点为空值

附 录 A
(资料性附录)
数据加解密示例

数据传输过程中可使用国家通用加密算法标准,密钥的交换机制不在本标准考虑。以下为 AES 加密算法示例,对数据进行加密,接口返回值为 AES 加密后字符串。使用全国组织机构代码统一社会信用代码数据服务中心提供的密钥解密。JAVA 语言的加解密代码示例,见表 A.1。PHP 语言的加解密代码示例,见表 A.2。

表 A.1 JAVA 语言的加解密代码示例

JAVA 语言的加解密代码示例
<pre> import javax.crypto.Cipher; import javax.crypto.spec.SecretKeySpec; public class Arithmetic { public static SecretKeySpec getKey(String seed){ SecretKeySpec key = null; try { key = new SecretKeySpec(hex2byte(seed.getBytes()), "AES"); } catch (Exception e) { e.printStackTrace(); } return key; } /* * * 加密 String 明文输入,String 密文输出 * @param strMing * @return * / public static String getEncString(String strMing, String seed) { byte[] byteMi = null; byte[] byteMing = null; String strMi = ""; try { return byte2hex(getEncCode(strMing.getBytes(),seed)); } catch (Exception e) { e.printStackTrace(); } finally { byteMing = null; byteMi = null; } return strMi; } /* * * 解密 以 String 密文输入,String 明文输出 </pre>

表 A.1 (续)

JAVA 语言的加解密代码示例
<pre> * @param strMi * @return * / public static String getDesString(String strMi, String seed) { byte[] byteMing = null; byte[] byteMi = null; String strMing = ""; try { return new String(getDesCode(hex2byte(strMi.getBytes()), seed)); } catch (Exception e) { e.printStackTrace(); } finally { byteMing = null; byteMi = null; } return strMing; } /* * * 加密以 byte[]明文输入,byte[]密文输出 * @param byteS * @return * / private static byte[] getEncCode(byte[] byteS, String seed) { SecretKeySpec key = getKey(seed); byte[] byteFina = null; Cipher cipher; try { cipher = Cipher.getInstance("AES/ECB/PKCS5Padding"); cipher.init(Cipher.ENCRYPT_MODE, key); byteFina = cipher.doFinal(byteS); } catch (Exception e) { e.printStackTrace(); } finally { cipher = null; } return byteFina; } /* * * 解密以 byte[]密文输入,以 byte[]明文输出 * @param byteD * @return * / private static byte[] getDesCode(byte[] byteD, String seed) { </pre>

表 A.1 (续)

JAVA 语言的加解密代码示例
<pre> SecretKeySpec key = getKey(seed); Cipher cipher; byte[] byteFina = null; try { cipher = Cipher.getInstance("AES/ECB/PKCS5Padding"); cipher.init(Cipher.DECRYPT_MODE, key); byteFina = cipher.doFinal(byteD); } catch (Exception e) { e.printStackTrace(); } finally { cipher = null; } return byteFina; } /* * * 二行制转字符串 * @param b * @return * / public static String byte2hex(byte[] b) { // 一个字节的数, // 转成 16 进制字符串 String hs = ""; String stmp = ""; for (int n = 0; n < b.length; n++) { // 整数转成十六进制表示 stmp = (Integer.toHexString(b[n] & 0XFF)); if (stmp.length() == 1) hs = hs + "0" + stmp; else hs = hs + stmp; } return hs.toUpperCase(); // 转成大写 } public static byte[] hex2byte(byte[] b) { if ((b.length % 2) != 0) throw new IllegalArgumentException("长度不是偶数"); byte[] b2 = new byte[b.length / 2]; for (int n = 0; n < b.length; n += 2) { String item = new String(b, n, 2); // 两位一组,表示一个字节,把这样表示的 16 进制字符串,还原成一个进制字节 b2[n / 2] = (byte) Integer.parseInt(item, 16); } return b2; </pre>

表 A.1 (续)

JAVA 语言的加解密代码示例
<pre> } public static void main(String[] args) { String seed = "B4DA0006EA6C5EAAF742D4161E01FDC1"; String test1 = Arithmetic.getEncString("明文字符串",seed);// System.out.println("sss:"+test1); String test2 = Arithmetic.getDesString(test1, seed); System.out.println("ttt:"+test2); } } </pre>

表 A.2 PHP 语言的加解密代码示例

PHP 语言的加解密代码示例
<pre> /* * 实现 AES 加密 * \$ text : 要加密的字符串 */ function encryptdString(\$ text){ \$ key = pack("H * ", \$ this->key); \$ pad = 16 - (strlen(\$ text) % 16); \$ text = str_repeat(chr(\$ pad), \$ pad); return bin2hex(mcrypt_encrypt(MCRYPT_RIJNDAEL_128, \$ key, \$ text, MCRYPT_MODE_ECB)); } /* * 实现 AES 解密 * \$ crypttext : 要解密的字符串 */ function decryptString(\$ crypttext){ \$ key = pack("H * ", \$ this->key); \$ crypttext = pack("H * ", \$ crypttext); \$ text = mcrypt_decrypt(MCRYPT_RIJNDAEL_128, \$ key, \$ crypttext, MCRYPT_MODE_ECB); \$ pad = 16-(strlen(\$ text) % 16); \$ text = str_repeat(chr(\$ pad), \$ pad); return \$ text; } </pre>

参 考 文 献

- [1] GB/T 7408—2005 数据元和交换格式 信息交换 日期和时间表示法
 - [2] GB/T 18793—2002 信息技术 可扩展置标语言(XML)1.0
 - [3] GB/T 20091—2006 组织机构类型
-

中华人民共和国
国家标准
用于微博客的法人和其他组织统一社会
信用代码实名认证服务接口规范
GB/T 36610—2018

*

中国标准出版社出版发行
北京市朝阳区和平里西街甲2号(100029)
北京市西城区三里河北街16号(100045)

网址: www.spc.org.cn

服务热线: 400-168-0010

2018年9月第一版

*

书号: 155066·1-61358

版权专有 侵权必究



GB/T 36610—2018