ICS 35.240.99
CCS L67

MH

中华人民共和国民用航空行业标准

MH/T XXXXX—XXXX

民航统一认证接口规范

Specification of civil aviation unified authentication interface

（征求意见稿）

在提交反馈意见时，请将您知道的相关专利连同支持性文件一并附上。

XXXX－XX－XX 发布　　XXXX－XX－XX 实施

中国民用航空局　　发 布

目　　次

前　言

本文件按照GB/T 1.1—2020《标准化工作导则　第1部分：标准化文件的结构和起草规则》的规定起草。

本文件由中国民用航空局综合司提出。

本文件由中国民航科学技术研究院归口。

本文件起草单位：中国民用航空局信息中心。

本文件主要起草人：曾曦、朱伯宇、曹媛。

民航统一认证接口规范

1 范围

本文件规定了中国民用航空局（以下简称“民航局”）各应用系统对接民航局信息中心统一认证平台（包括认证Authentication、 授权Authorization、账号Accounting 、审计Audit，简称4A）和面向社会公众的统一身份认证系统（包括认证Authentication、账号Accounting，简称2A）时的接入模式、接入流程和接口定义。

本文件适用于民航各政务信息系统之间，以及民航各政务信息系统与国家政务服务平台间统一认证的应用过程。

2 规范性引用文件

下列文件中的内容通过文中的规范性引用而构成本文件必不可少的条款。其中，注日期的引用文件，仅该日期对应的版本适用于本文件；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。

C 0110-2018 国家政务服务平台统一身份认证系统接入要求

C 0111-2018 国家政务服务平台统一身份认证系统身份认证技术要求

C 0112-2018 国家政务服务平台统一身份认证系统信任传递要求

C 0114-2018 国家政务服务平台可信身份等级定级要求

3 术语和定义

下列术语和定义适用于本文件。

3.1

自然人 natural person

自然人是基于出生而取得民事主体资格的人，其外延包括本国公民、港澳台居民、外国公民和无国籍人等。

[来源：C 0111-2018，3.3]

3.2

法人 legal person

具有民事权利能力和民事行为能力，依法独立享有民事权利和承担民事义务的组织。

[来源：C 0111-2018，3.4]

3.3

信任传递 trust transitivity

实现用户、业务系统的强身份鉴别，跨域条件下的信任传递。本标准中特指在统一身份认证系统（2A）中，将某个政务系统的已登录身份传递到另一个系统完成登录的过程。

[来源：C 0111-2018，3.15，有修改]

3.4

单点登录 single sign-on

当用户访问多个应用系统时，只需提交一次认证信息就可访问多个应用系统。

[来源：C 0111-2018，3.16]

3.5

令牌 token

认证子系统产生的，用于标识用户的登录身份信息。

注：在统一身份认证体系中，用户登录会话与令牌相绑定。

[来源：C 0111-2018，3.17，有修改]

3.6

票据　ticket

基于随机数的一次性会话验证凭据，用于验证通讯请求的合法性。

[来源：C 0111-2018，3.18]

4 缩略语

下列缩略语适用于本文件。

HTTP　超文本传输协议（Hyper Text Transfer Protocol）

URL　统一资源定位器（Uniform Resource Locator）

SDK　软件开发工具包（Software Development Kit）

5 概述

5.1 统一认证平台（4A）接口

统一认证平台（4A）接口面向民航局各应用系统的业务人员，采用认证、授权、账号、审计的“4A”管理，融合集中认证管理、集中权限管理、集中帐号管理、集中审计管理四要素，实现统一认证平台与民航局各应用系统间的单点登录。该接口适用于民航局统一认证平台门户网站单点登录到民航局各应用系统的使用场景。统一认证平台到民航局各应用系统的数据同步方向为正向，反之为逆向。

民航局各应用系统采用票据的方式接入统一认证平台（4A），实现从统一认证门户跳转至民航局各应用系统、并有选择地实现机构和账号的正向和逆向数据同步。

5.2 统一身份认证系统（2A）接口

统一身份认证系统（2A）接口面向社会公众的被服务人员，采用账号和认证的“2A”管理，实现社会公众登录民航局各应用系统办理相关服务事项。该接口基于国家政务服务平台标准建设，适用于民航局与面向社会公众提供在线办理服务系统对接的场景，和民航局与自然人、法人注册系统对接的场景，为民航局对接以上系统提供用户管理、用户实名核验、用户数据汇聚、单点登录等能力。该接口支持自然人或企业法人通过统一身份认证系统登录到民航局各应用系统，并支持自然人或企业法人与国家政务服务平台身份信息互信互认。

统一身份认证系统有统一模式和协同模式两种对接方式，民航局各应用系统对接时需选择其中一种对接模式。

6 统一认证平台（4A）对接要求

6.1 总体要求

统一认证平台（4A）对接的总体要求如下：

a) 统一认证平台（4A）采用票据方式单点登录到应用系统；

b) 应用系统集成统一认证平台（4A）提供的SDK开发包；

c) 应用系统拦截统一认证平台（4A）到应用系统的票据登录请求，解密票据字段定义为 pname；

d) 解密票据的方法：public static String com.linkage.util.EncodeUtil.decodeURL(String pname)；

e) 解密后的票据参数分项见表1；

f) 应用系统将统一认证平台的pname解析完成后得到票据ticket，并将该票据发送至统一认证平台票据服务进行验证。

表1　解密后的票据信息

参数名称	类型和长度	必填	说明	描述
MAIN_ID	String(64)	是	当前登录统一认证平台的帐号	此帐号为统一认证平台的登录账号区别于应用系统的账号
op	String(64)	是	表示经统一认证平台单点登录	此参数可没有，根据应用实际情况设置
loginName	String(64)	是	登录应用系统的帐号	此帐号为应用系统的账号区别于登录统一认证平台的账号
ticket	String(200)	是	当前票据	应用系统需要将此票据发回单点登录服务器进行验证

6.2　接入流程

6.2.1　流程图

统一认证平台（4A）接口接入流程见图1。

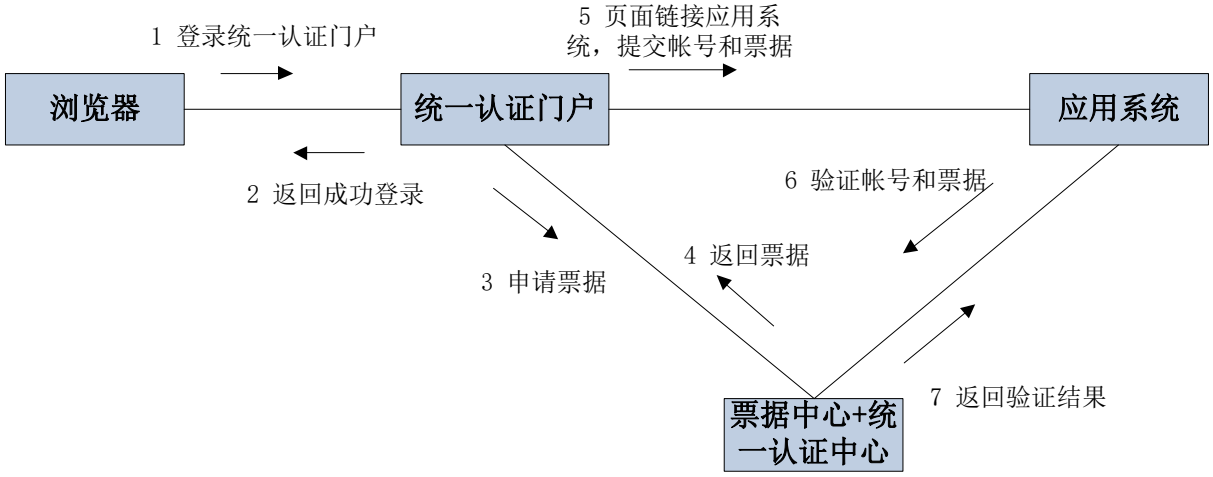

图1　统一认证平台（4A）接口接入流程

6.2.2　流程说明

接入统一认证平台（4A）接口应符合如下流程。

a）　用户通过浏览器登录统一认证平台（4A）门户。

b）　统一认证平台（4A）门户向用户返回统一认证平台（4A）门户登录结果。

c）　如登录成功，统一认证平台（4A）门户向票据中心申请票据。

d）　票据中心向统一认证平台（4A）门户返回票据。

e）　统一认证平台（4A）门户使用应用系统的链接地址，并附带用户账号和票据登录到应用系统。访问地址示例如下：

示例：http://应用系统 URL/login.do?pname=c3RhZmZOYW1lPUFBNjAwMDAzJk1BSU5fSUQ9UzBfcW1hbmdlJnRpY2tldDOyYzExZDQyYy00NTA5LTAwMTYzNTd5STZONjRIVTVqcG1ZbFk4dkQ1dFE9PQ%3D%3Dz%26%26%26cf4。

f）　应用系统收到票据后，对票据进行解密，并向票据中心验证票据真实性；同时，应用系统对收到的用户账号有效性进行验证。解密后的票据信息见表 1。

g）　票据中心向应用系统返回验证结果，如结果正确，则用户登录成功；如结果错误，则用户登录失败。验证票据的返回码应符合附录 A 中表 A.1 的规定。

6.3 数据接口方式

应用系统与统一认证平台（4A）数据接口采用轻量级的rest api+JSON POST方式实现数据的传输。

6.4 接口服务说明

6.4.1 应用系统验证票据服务

应用系统向票据中心验证用户登录请求时携带票据。具体接口调用方式如下：

a) 调用地址：http://统一认证平台URL/pname；

b) 接口参数说明见表2；

表2 验证票据参数说明

类别	参数	必填	数据类型	说明
入参	pracct	是	字符串	主账号
	slacct	是	字符串	从账号
	ticket	是	字符串	票据，ticket由应用系统通过解密pname获得，解密方法在统一认证平台提供的SDK包里：public static String com.linkage.util.EncodeUtil.decodeURL(String pname)
	clientIp	是	字符串	登录资源地址：在统一认证平台（4A）管理台中配置的应用系统单点登录地址
	clientPort	是	字符串	登录资源端口号：在统一认证平台（4A）管理台中配置的应用系统服务端口号
	userIp	是	字符串	用户地址：在统一认证平台（4A）管理台中配置的应用系统服务器地址
出参	resultCode	是	字符串	结果代码，应符合附录A中表A.1的规定
	resultMsg	是	字符串	结果提示信息
	userName	是	字符串	登录系统帐号名

c) 验证票据返回码应符合附录A中表A.1的规定；

d) pname 解密方法decodeURL示例。

示例：

```
public static String decodeURL(String enCode) {
    if (enCode != null) {
        String[] splitStr = enCode.split("z&&&");
        if (splitStr != null && splitStr.length > 1) {
            try {
                String encodeUrl = splitStr[0];
                if (encodeUrl != null && getMd5(encodeUrl).equals(splitStr[1])) {
                    return new String(BaseEncode.decode(encodeUrl), "UTF-8");
                }
                return "";
            } catch (UnsupportedEncodingException var3) {
                LOGGER.error(var3.getMessage(), var3);
```

```
                return "";
            }
        }
    }
    return "";
}
```

6.4.2 统一认证平台（4A）到应用系统用户名密码登录认证服务

统一认证平台（4A）向应用系统发起用户名密码登录认证请求，该接口为统一认证平台（4A）到应用系统验证登录用户名密码的合法性提供验证服务。具体接口调用方式如下：

a) 调用地址：http://应用系统URL/doSSO；

b) 接口参数说明见表3；

表3 统一认证平台（4A）到应用系统用户名密码登录认证参数说明

类别	参数	必填	数据类型	说明
入参	pracct	是	字符串	统一认证平台（4A）主帐号名称
	slacct	是	字符串	应用系统从帐号名称
	ticket	是	字符串	验证票据
	clientIp	是	字符串	登录资源地址，该地址为统一认证平台（4A）管理台维护，作为应用系统白名单判断使用
	clientPort	是	字符串	登录资源端口号，该地址为统一认证平台（4A）管理台维护，作为应用系统判断使用
	userIp	是	字符串	用户地址，用户终端地址
出参	resultCode	是	字符串	结果代码，应符合附录A中表A.1的规定
	resultMsg	是	字符串	返回信息
	userName	是	字符串	登录应用系统帐号名

c) 调用方法示例。

示例：

```
Client client = Client.getInstance(“文件路径”);
CResult result = client.doSSO(pracct, slacct,ticket, clientIp, clientPort,userIp);
if (result.getResultCode() == 0) {
      String username = result.getUsername();
      System.out.println(“用户” + username + result.getResultMsg());
} else if (result.getResultCode() == -1) {
      System.out.println(result.getResultMsg());
} else {
      System.out.println(“认证不通过！原因：” + result.getResultMsg());

}
```

6.4.3 应用系统到统一认证平台用户名密码登录认证服务

应用系统向统一认证平台（4A）发起用户名密码登录认证请求，该接口为应用系统到统一认证平台（4A）验证登录用户名密码的合法性提供验证服务。具体接口调用方式如下：

a) 调用地址：http://统一认证平台URL/loginAuthenByPassWd；

b) 接口参数说明见表4；

表4 应用系统到统一认证平台（4A）用户名密码登录认证参数说明

类别	参数	必填	数据类型	说明
入参	clientIp	是	字符串	客户端IP，应用系统服务器IP，该IP为统一认证平台（4A）白名单配置项
	userIp	是	字符串	用户IP，暨用户终端的IP
	operatorName	是	字符串	应用系统登录帐号，与验证的用户名相同
	appSysNum	是	字符串	应用系统编码，该编码由统一认证平台（4A）统一分配
	userName	是	字符串	用户名
	passWd	是	字符串	用户密码，加密方法在统一认证平台（4A）提供的SDK包里：com.linkage.fa.service.util.AESUtil.AES_Encrypt(String plainText)
出参	errCode	是	字符串	返回码，应符合附录A中表A.2的规定
	errDes	是	字符串	返回信息

c) AES_Encrypt 加密方法示例。

示例：

```
public static String AES_Encrypt(String plainText) {
    return AES_Encrypt("asiainfoSecurity", plainText);
}
public static String AES_Encrypt(String keyStr, String plainText) {
     if (keyStr != null && plainText != null && !"".equals(keyStr) && !"".equals(plainText)) {
          StringBuffer _key = new StringBuffer();
          if (keyStr.length() < 16) {
               for(int i = 0; i < 16 - keyStr.length(); ++i) {
                    _key.append("=");
               }
               keyStr = keyStr + _key;
          } else if (keyStr.length() > 16) {
```

```
            keyStr = keyStr.substring(0, 16);
        }
        byte[] encrypt = (byte[])null;
        try {
            Key key = generateKey(keyStr);
            Cipher cipher = Cipher.getInstance("AES/ECB/PKCS5Padding");
            cipher.init(1, key);
            encrypt = cipher.doFinal(plainText.getBytes());
        } catch (Exception var6) {
            LOGGER.error(var6.getMessage(), var6);
        }
        return new String(Base64.encodeBase64(encrypt));
    } else {
        return "";
    }
}
```

6.4.4 应用系统获取短信验证码服务

应用系统向统一认证平台（4A）发起获取短信验证码请求。具体接口调用方式如下：

a) 调用地址：http://统一认证平台URL/sendShortMessage；

b) 接口参数说明见表5；

表5 应用系统到统一认证平台（4A）用户名密码登录认证参数说明

类别	参数	必填	数据类型	说明
入参	clientIp	是	字符串	客户端IP，应用系统服务器IP，该IP为统一认证平台（4A）白名单配置项
	userIp	是	字符串	用户IP，即用户终端的IP
	operatorName	是	字符串	应用系统登录帐号，与验证的用户名相同
	appSysNum	是	字符串	应用系统编码，该编码由统一认证平台（4A）统一分配
	telephone	是	字符串	手机号码
出参	errCode	是	字符串	返回码，应符合附录A中表A.2的规定

表 5　应用系统到统一认证平台（4A）用户名密码登录认证参数说明（续）

类别	参数	必填	数据类型	说明
出参	errDes	是	字符串	返回信息

c）　该功能可通过发送 JSON 报文实现，无需调用统一认证平台（4A）SDK，报文示例如下。

示例：

```
输入报文：
{
    "clientIp":"客户端 IP（调用端服务器 IP）",
    "operatorName":"操作人帐号",
    "userIp":"用户 IP（用户机器 IP）",
    "appSysNum":"123",
    "telephone":"手机号"
}
输出报文：
{
    "errCode":"错误代码",
    "errDes":"错误描述"

}
```

6.4.5　应用系统短信验证码登录认证服务

应用系统向统一认证平台（4A）发起逆向短信验证码登录请求，该服务为应用系统到统一认证平台（4A）验证短信验证码是否有效，应用系统根据返回的结果获得短信码的合法性。具体接口调用方式如下：

a）　调用地址：http://统一认证平台 URL/loginAuthenBySMS；

b）　接口参数说明见表 6；

表6　应用系统短信验证码登录认证参数说明

类别	参数	必填	数据类型	说明
入参	clientIp	是	字符串	客户端IP，应用系统服务器IP，该IP为统一认证平台（4A）白名单配置项
	userIp	是	字符串	用户IP，即用户终端的IP
	operatorName	是	字符串	应用系统登录帐号，与验证的用户名相同
	appSysNum	是	字符串	应用系统编码，该编码由统一认证平台（4A）统一分配
	telephone	是	字符串	手机号码
	verifyCode	是	字符串	验证码，该验证码为获取短信验证码接口中获取到的值
出参	errCode	是	字符串	返回码，应符合附录A中表A.2的规定

表 6　应用系统短信验证码登录认证参数说明（续）

类别	参数	必填	数据类型	说明
出参	errDes	是	字符串	返回信息

c）　该功能可通过发送 JSON 报文实现，无需调用统一认证平台（4A）SDK，报文示例如下。

示例：

```
输入报文：
{
    "clientIp":"客户端 IP（调用端服务器 IP）",
    "operatorName":"操作人帐号",
    "userIp":"用户 IP（用户机器 IP）",
    "appSysNum":"123",
    "sessionUuid":"会话 id",
    "signedToken":"token",
    "telephone":"手机号",
    "verifyCode":"短信验证码"
}
输出报文：
{
    "errCode":"错误代码",
    "errDes":"错误描述"

}
```

7　统一身份认证系统（2A）对接要求

7.1　总体要求

7.1.1　统一模式要求

统一模式要求是以统一身份认证系统（2A）为中心的认证方式，各应用系统需采用统一身份认证系统（2A）的数据模型、登录和注册界面。应用系统无存量用户且不维护用户身份信息，所有自然人、法人信息由统一身份认证系统（2A）统一存储和管理。

总体要求如下：

a）　应用系统集成统一身份认证系统（2A）SDK，并对接信任传递流程（具体流程见 7.2.1）；

b）　用户登录应用系统时，需在浏览器端跳转至统一身份认证系统（2A）登录界面，并在统一身份认证系统（2A）登录地址中附带 backUrl 参数，参数值为应用系统信任传递回调页面地址；

c）　用户通过统一身份认证系统（2A）的登录界面输入账号密码登录成功后，浏览器端自动跳转回应用系统 backUrl 的参数地址，同时附带 ticket（一次性票据）参数；

d）　应用系统接收 ticket（一次性票据）参数后，通过信任传递流程获取已登录的当前用户身份信息。

7.1.2　协同模式要求

协同模式要求应用系统本身存在用户模型及存量用户，利用统一身份认证系统（2A）的实名核验能力和数据汇聚能力完成认证。应用系统保留自己的数据模型、登录和注册方式，其中，数据模型应符合附录B中表B.4和表B.5的规定。

总体要求如下：

a）　应用系统需集成统一身份认证系统（2A）SDK，按照隐性登录流程（具体流程见 7.2.2）与统一身份认证系统（2A）同步用户身份信息；

b）　应用系统通过隐性登录接口向统一身份认证系统（2A）同步当前登录的用户信息，此时应用系统和统一身份认证系统（2A）均标记同一用户为已登录状态；

c） 应用系统向统一身份认证系统（2A）传递已登录的用户身份信息，已登录的用户身份信息由统一身份认证系统（2A）入库并统一管理。

7.2 接入流程

7.2.1 信任传递流程

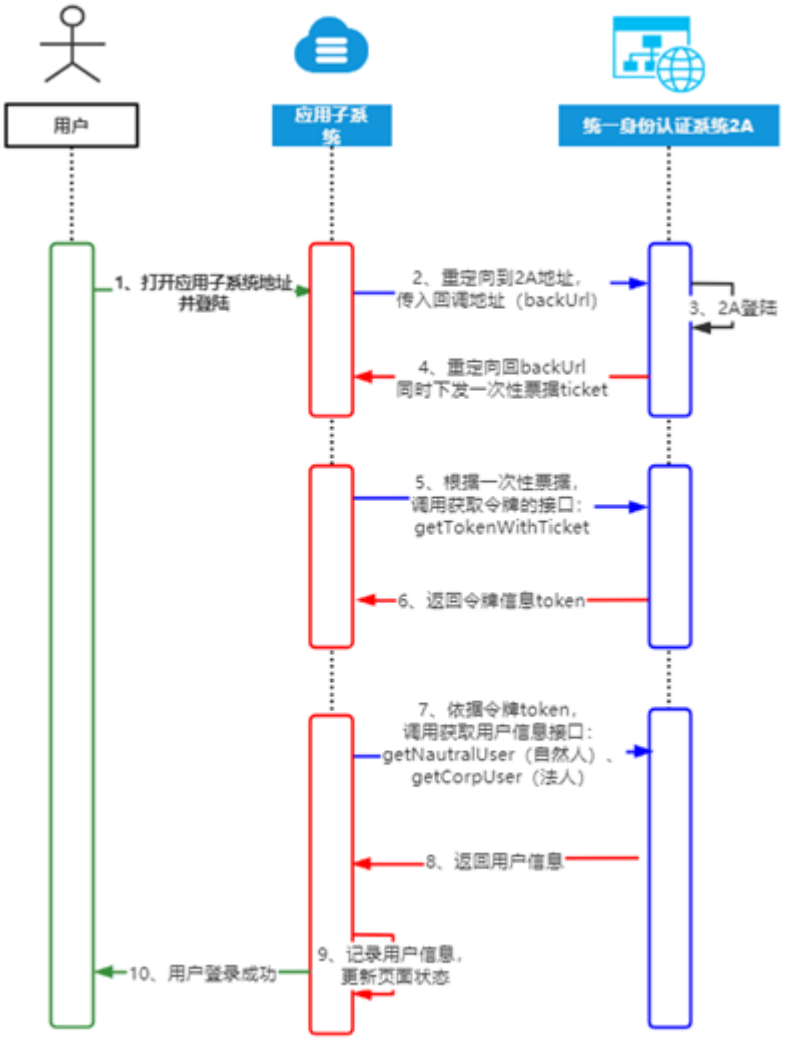

图2 信任传递流程

信任传递流程应符合C 0110-2018《国家政务服务平台统一身份认证系统接入要求》和C 0112-2018《国家政务服务平台统一身份认证系统信任传递要求》中的规定。信任传递流程见图2，流程说明如下。

a） 用户通过浏览器访问应用系统并使用登录功能。

b） 应用系统通过浏览器重定向到统一身份认证系统（2A）登录页面，同时 URL 中传入回调地址 backUrl。

c） 用户在统一身份认证系统（2A）登录页面输入账号、密码认证。

d） 统一身份认证系统（2A）经过业务处理后，重定向回应用系统，重定向地址为 backUrl 参数。同时统一身份认证系统（2A）下发一次性票据 ticket。

e） 应用系统根据一次性票据 ticket 调用统一身份认证系统（2A）获取令牌接口（getTokenWithTicket 接口）。

f） 统一身份认证系统（2A）经过逻辑处理后返回 token 令牌，应用系统接收返回值 token 令牌；

g） 应用系统根据 token 令牌调用统一身份认证系统（2A）获取用户信息接口 getNautralUser（自然人）、getCorpUser（法人），获得用户身份信息。
h） 统一身份认证系统（2A）经过业务处理后根据应用系统传入的 token 值，返回用户身份信息；
i） 应用系统根据获得的用户身份信息，在应用系统中标记业务系统账号已登录状态。
j） 用户成功登录应用系统。

7.2.2 隐形登录流程

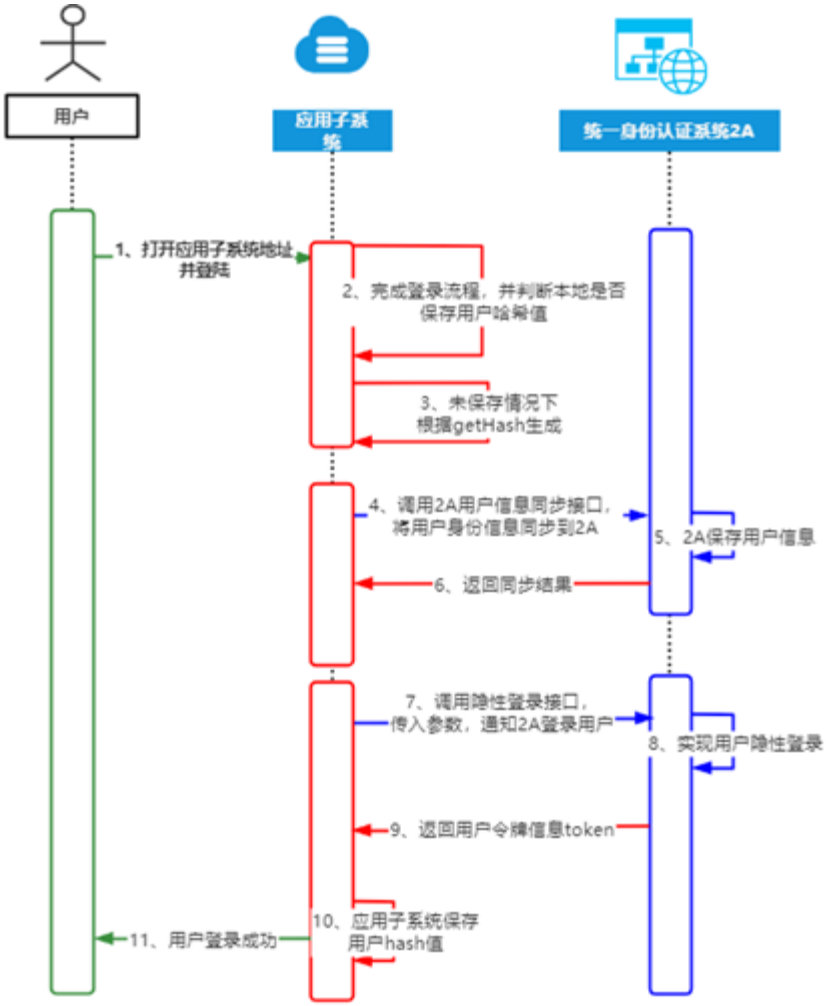

图3 隐性登录流程

隐性登录流程见图 3，流程说明如下。

a） 用户通过浏览器访问应用系统并使用登录功能。
b） 用户在应用系统输入账号、密码认证。用户登录成功后，应用系统判断本地是否保存该用户的哈希值。
c） 如应用系统未保存用户的哈希值，则应用系统调用 getHash 散列方法生成用户的哈希值；如应用系统已保存用户的哈希值，则应用系统直接获取用户的哈希值。
d） 应用系统调用统一身份认证系统（2A）用户信息实时同步接口，将用户身份信息同步到统一身份认证系统（2A）。
e） 统一身份认证系统（2A）保存应用系统同步的用户信息。
f） 统一身份认证系统（2A）返回数据同步结果给应用系统。
g） 应用系统调用隐性登录接口（见 7.5），通知统一身份认证系统（2A）该用户已登录。

h） 统一身份认证系统（2A）内部实现用户隐性登录流程，该流程将用户在国家政务服务平台的登录状态置为已登录状态。
i） 统一身份认证系统（2A）返回用户令牌信息（token）至应用系统。
j） 应用系统保存该用户的哈希值。
k） 用户成功登录应用系统。

7.2.3 登出流程

登出流程具体说明如下。
a） 用户登出应用系统时，应用系统调用统一身份认证系统（2A）平台登出接口（见 7.6.3），同时登出统一身份系统（2A）以及国家政务服务平台。
b） 用户登出国家政务服务平台时，同时登出统一身份认证系统（2A）以及所有应用系统。
c） 用户登出统一身份认证系统（2A）时，同时登出国家政务服务平台以及所有应用系统。
d） 统一身份认证系统（2A）登出流程应满足 C 0110-2018《国家政务服务平台统一身份认证系统接入要求》中的规定。

7.3 实名核验服务

7.3.1 社团法人实名核验服务

实现社团法人的社会组织名称、登记号、法定代表人的实名验证。具体接口调用方式如下：
a） 调用 SDK 方法：EnterpriseResult verifyAssociation(EnterpriseRequest request)；
b） 请求参数说明见表 7；
c） 返回值应符合附录 B 中表 B.1 的规定；

表7 社团法人实名核验参数说明

类别	参数	必填	数据类型	说明
入参	userName	是	字符串	社会组织名称
	uniscId	是	字符串	登记号
	name	是	字符串	法定代表人姓名
出参	code	是	字符串	响应码，应符合附录B中表B.1的规定
	msg	是	字符串	响应信息

d） 调用方法示例。

示例：

```
package com.asiainfo.test;
import com.alibaba.fastjson.JSON;
import gov.zwfw.iam.client.TacsHttpClient;
import gov.zwfw.iam.exception.TacsException;
import gov.zwfw.iam.real.client.RealClient;
import gov.zwfw.iam.real.request.NaturalRequest;
import gov.zwfw.iam.real.response.RealResult;
public class TestSimpleSt {
     public static void main(String[] args) {
     //读取配置文件 tacs.cer
     TacsHttpClient.init(证书路径，环境地址);
    EnterpriseRequest request = new EnterpriseRequest ();
    request.setEntName("中国营养保健食品协会");
    request.setUniscId("5082");
    request.setName("刘学聪");
     RealClient realClient = null;
```

```
    try {
         realClient = TacsRealClient.getInstance();
          EnterpriseResult result = realClient.verifyAssociation(request);
         } catch (TacsException e1) {
              // TODO Auto-generated catch block
              e1.printStackTrace();
         }

}
```

7.3.2 企业法人实名核验接口

实现企业法人信息实名验证,传入企业名称、或统一社会信用代码等,返回企业照面信息比对结果。具体接口调用方式如下：

a) 调用 SDK 方法：EnterpriseResult verifyEnterpriseInfo (EnterpriseRequest request)；

b) 接口参数说明见表 8；

表8 企业法人实名核验参数说明

类别	参数	必填	数据类型	说明
入参	entName	是	字符串	主体名称
	uniscId	是	字符串	统一社会信用代码
	regno	否	字符串	注册号
	enttypeCn	否	字符串	企业类型
	regcap	否	字符串	注册资本
	regcapcurCn	否	字符串	注册资本币种
	estdate	否	字符串	成立日期
	apprdate	否	字符串	核准日期
	regstateCn	否	字符串	登记状态（中文）
	regorgCn	否	字符串	登记机关（中文）
	opfrom	否	字符串	经营期限自
	opto	否	字符串	经营期限至
	dom	否	字符串	住所
	name	否	字符串	法定代表人
	opscope	否	字符串	经营范围
出参	code	是	字符串	响应码，应符合附录B中表B.1的规定
	msg	是	字符串	响应信息

c) 接口返回值应符合附录 B 中表 B.1 的规定；

d) 调用方法示例。

示例：

```
import gov.zwfw.iam.client.TacsHttpClient;
import gov.zwfw.iam.data.request.EnterpriseRequest;
import gov.zwfw.iam.exception.TacsException;
import gov.zwfw.iam.real.client.RealClient;
import gov.zwfw.iam.real.client.TacsRealClient;
import gov.zwfw.iam.real.response.EnterpriseResult;
public class Demo {
    public void main(String[] args) {
        TacsHttpClient.init("证书路径","环境地址");
        EnterpriseRequest request = new EnterpriseRequest ();
        request.setEntName("亚信科技（成都）有限公司");
        RealClient realClient = null;
        try {
            realClient = TacsRealClient.getInstance();
            EnterpriseResult result = realClient.verifyEnterpriseInfo(request);
        } catch (TacsException e) {
            e.printStackTrace();
        }
    }

}
```

7.3.3 事业单位法人实名核验服务

实现事业单位法人的社会组织名称、登记号、法定代表人的实名验证，传入事业单位法人的社会组织名称、登记号、法定代表人返回该法人的实名验证信息。具体接口调用方式如下：

a) 调用 SDK 方法：EnterpriseResult verifyCause(EnterpriseRequest request)；

b) 接口参数说明见表 9；

表9 事业单位法人实名核验参数说明

类别	参数	必填	数据类型	说明
入参	username	是	字符串	事业单位名称
	uniscId	是	字符串	统一社会信用代码
	name	否	字符串	法定代表人姓名
出参	code	是	字符串	响应码，应符合附录B中表B.1的规定
	msg	是	字符串	响应信息

c) 接口返回值应符合附录 B 中表 B.1 的规定；

d) 调用方法示例。

示例：

```
package com.asiainfo.test;
import com.alibaba.fastjson.JSON;
import gov.zwfw.iam.client.TacsHttpClient;
import gov.zwfw.iam.exception.TacsException;
import gov.zwfw.iam.real.client.RealClient;
import gov.zwfw.iam.real.request.NaturalRequest;
import gov.zwfw.iam.real.response.RealResult;
public class TestSimpleSt {
     public static void main(String[] args) {
          //读取配置文件 tacs.cer
          TacsHttpClient.init(证书路径, 环境地址);
```

```
EnterpriseRequest request = new EnterpriseRequest ();
request.setEntName("中国营养保健食品协会");
request.setUniscId("5082");
request.setName("张三");
         RealClient realClient = null;
         try {
              realClient = TacsRealClient.getInstance();
EnterpriseResult result = realClient.verifyCause(request);
         } catch (TacsException e1) {
              e1.printStackTrace();
         }

}
```

7.3.4 个体工商户实名核验服务

实现个体工商户法人的身份证号、姓名、工商注册号的实名验证。传入个体工商户法人的身份证号、姓名、工商注册号返回该法人的实名验证信息。具体调用方式如下：

a) 调用 SDK 方法：EnterpriseResult verifyIndividual (EnterpriseRequest request)；

b) 接口参数说明见表 10；

表10 个体工商户实名核验参数说明

类别	参数	必填	数据类型	说明
入参	certNo	是	字符串	身份证号
	name	是	字符串	姓名
	uniscId	否	字符串	工商注册号
出参	code	是	字符串	响应码，应符合附录B中表B.1的规定
	msg	是	字符串	响应信息

c) 接口返回值应符合附录 B 中表 B.1 的规定；

d) 调用方法示例。

示例：

```
package com.asiainfo.test;
import com.alibaba.fastjson.JSON;
import gov.zwfw.iam.client.TacsHttpClient;
import gov.zwfw.iam.exception.TacsException;
import gov.zwfw.iam.real.client.RealClient;
import gov.zwfw.iam.real.request.NaturalRequest;
import gov.zwfw.iam.real.response.RealResult;

public class TestSimpleSt {

    public static void main(String[] args) {
         //读取配置文件 tacs.cer
         TacsHttpClient.init(证书路径，环境地址);
          EnterpriseRequest request = new EnterpriseRequest ();
          request.setCertNo("125484568541254615");
          request.setUniscId("565456518919");
          request.setName("张三");
         RealClient realClient = null;
         try {
              realClient = TacsRealClient.getInstance();
```

```
            EnterpriseResult result = realClient. verifyIndividual (request);
        } catch (TacsException e1) {
            e1.printStackTrace();
        }

}
```

7.3.5 自然人初级实名核验服务

实现基于自然人姓名、身份证号码、身份证起始日期四项要素实名信息核验能力。具体接口调用方式如下：

a) 调用 SDK 方法：EnterpriseResult verifyIndividual (EnterpriseRequest request)；

b) 接口参数说明见表 11；

表11 自然人初级实名核验参数说明

类别	参数	必填	数据类型	说明
入参	userName	是	字符串	用户姓名
	certNo	是	字符串	用户身份证号
	effDate	是	字符串	身份证号有效起始日期
	expDate	是	字符串	身份证号有效结束日期
出参	code	是	字符串	响应码，应符合附录B中表B.1的规定
	msg	是	字符串	响应信息
	data	是	字符串	响应数据
	realLevel	是	字符串	实名等级应符合附录B中表B.2的规定，初级实名等级值为“3”

c) 接口返回值应符合附录 B 中表 B.1 的规定；

d) 调用方法示例。

示例：

```
public void main(String[] args) {
    TacsHttpClient.init("证书路径");
    TacsRealClient realInstance = TacsRealClient.getInstance();
    NaturalRequest naturalRequest = new NaturalRequest();
    naturalRequest.setCertNo("23230XXXXXXXXXXX");
    naturalRequest.setUserName("陈 XX");
    RealResult realResult = realInstance.simpleTwoPattern(naturalRequest);
    System.out.println(JSON.toJSONString(realResult));

}
```

7.3.6 自然人中级实名核验服务

实现自然人的姓名、身份证号码、人像两要素中级核验接口，核验通过返回该自然人的实名验证等级信息。具体接口调用方式如下：

a) 调用 SDK 方法：RealResult simpleTwoPattern(NaturalRequest request)；

b) 接口参数说明见表 12；

表12 自然人中级实名核验参数说明

类别	参数	必填	数据类型	说明
入参	userName	是	字符串	用户姓名
	certNo	是	字符串	用户身份证号
	effDate	是	字符串	身份证号有效起始日期
	expDate	是	字符串	身份证号有效结束日期
	pictureControlsVersion	是	字符串	人像控件版本号
	pictureFile	是	字符串	人像数据（base64编码）
出参	code	是	字符串	响应码，应符合附录B中表B.1的规定
	msg	是	字符串	响应信息
	data	是	字符串	响应数据
	realLevel	是	字符串	实名等级应符合附录B中表B.2的规定，中级实名等级值为“4”

c) 接口返回值应符合附录 B 中表 B.1 的规定；

d) 调用方法示例。

示例：

```
public void main(String[] args) {
    TacsHttpClient.init("证书路径");
    TacsRealClient realInstance = TacsRealClient.getInstance();
    NaturalRequest naturalRequest = new NaturalRequest();
    naturalRequest.setCertNo("23230XXXXXXXXXXX");
    naturalRequest.setUserName("陈 XX");
    naturalRequest. setPictureControlsVersion("XXXXX");
    naturalRequest .setPictureFile("XXXXX");
    RealResult realResult = realInstance. simpleTwoPatternWithPicture (naturalRequest);
    System.out.println(JSON.toJSONString(realResult));

}
```

7.3.7 企业法人实名核验服务(四要素)

实现企业法人信息实名验证。传入企业名称、或统一社会信用代码、法定代表人姓名、法定代表人身份证号等，返回企业照面信息比对结果。具体接口调用方式如下：

a) 调用 SDK 方法：EnterpriseResult verifyEnterpriseInfoWithFourEle(EnterpriseRequest request)；

b) 接口参数说明见表 13；

表13 企业法人实名核验参数说明

类别	参数	必填	数据类型	说明
入参	entName	是	字符串	主体名称
	uniscId	是	字符串	统一社会信用代码

表 13　企业法人实名核验参数说明（续）

类别	参数	必填	数据类型	说明
入参	name	是	字符串	法定代表人姓名
	certNo	是	字符串	法定代表人身份证号
出参	code	是	字符串	响应码，应符合附录B中表B.1的规定
	msg	是	字符串	响应信息

c)　接口返回值应符合附录 B 中表 B.1 的规定；

d)　调用方法示例。

示例：

```
public class Test {
     public static void main(String[] args) {
          EnterpriseRequest request = new EnterpriseRequest ();
          request.setEntName("亚信科技（成都）有限公司");
         request.setUniscId("91510100732356360H");
         request.setName("何 XX");
         request.setCertNo("******");
          TacsHttpClient.init(证书路径, 环境地址);
          RealClient realClient = null;
          try {
               realClient = TacsRealClient.getInstance();
               EnterpriseResult result = realClient. verifyEnterpriseInfoWithFourEle (request);
          } catch (TacsException e) {
               e.printStackTrace();
          }
     }

}
```

7.3.8　出入境证件核验服务

实现核验护照港澳往来通行证等证件。核验护照港澳往来通行证等证件。具体接口调用方式如下：

a)　调用 SDK 方法：AuthResult immiCheck (UserRequest request)；

b)　接口参数说明见表 14；

表14　出入境证件核验参数说明

类别	参数	必填	数据类型	说明
入参	idNumber	是	字符串	证件号
	idType	是	字符串	证件类型，证件类型应符合附录B中表B.3的规定
	name	是	字符串	姓名
	nation	是	字符串	国籍编码，国籍三位字母编码参见最新版国别国籍编码大全
出参	code	是	字符串	响应码，应符合附录B中表B.1的规定
	msg	是	字符串	响应信息

c)　接口返回值应符合附录 B 中表 B.1 的规定；

d)　调用方法示例。

示例：

```
public void main(String[] args) {
TacsHttpClient.init(证书路径，环境地址);
     TacsAuthClient client = TacsAuthClient.getInstance();
     AuthRequest request = new AuthRequest ();
     request.setIdType("*****");
    request.setIdNumber("*****");
    request.setName("*****");
    request.setNation("*****");
     AuthResult result = cient.immiCheck(request);

}
```

7.4 认证服务接口 SDK

7.4.1 自然人认证服务

获取自然人用户基本信息(明文)。通过令牌唯一识别号获取自然人用户信息接口。具体接口调用方式如下：

a) 调用方法：AuthResult getNaturalUser(AuthRequest request)；
b) 接口参数说明见表 15；

表15 自然人认证参数说明

类别	参数	必填	数据类型	说明
入参	tokenSNO	是	字符串	令牌唯一识别号
出参	code	是	字符串	响应码，应符合附录B中表B.1的规定
	msg	是	字符串	响应信息
	user	是	字符串	自然人用户基本信息（明文），应符合附录B中表B.4的规定

c) 接口返回值应符合附录 B 中表 B.1 的规定；
d) 调用方法示例。

示例：

```
public void sample()
    {
     //初始化客户端
     TacsHttpClient.init(证书路径，环境地址);
     //获取客户端
     TacsAuthClient authClient = TacsAuthClient.getInstance();
     //发起调用
     AuthRequest authRequest = new AuthRequest ();
      authRequest.setTokenSNO("A86A37C4C3C0D1DA63735C60C911B4988A69A654E4B2D9EA0C5C2A2383168FFE");
     //获取结果
      AuthResult authResult = authClient. getNaturalUser(authRequest);
       System.out.println(auth.getNaturalUser());

}
```

7.4.2 法人认证接口服务

获取法人用户基本信息(明文)。通过令牌唯一识别号获取法人用户信息接口。具体接口调用方式如下：

a) 调用方法：AuthResult getCorpUser(AuthRequest request)；
b) 接口参数说明见表 16；

表16 法人认证参数说明

类别	参数	必填	数据类型	说明
入参	tokenSNO	是	字符串	令牌唯一识别号
出参	code	是	字符串	响应码，应符合附录B中表B.1的规定
	msg	是	字符串	响应信息
	corporator	是	字符串	法人用户基本信息（明文），应符合附录B中表B.5的规定

c) 接口返回值应符合附录 B 中表 B.1 的规定；

d) 调用方法示例。

示例：

```
public void sample()
{
     //初始化客户端
     TacsHttpClient.init(证书路径, 环境地址);
     //获取客户端
     TacsAuthClient authClient = TacsAuthClient.getInstance();
     //发起调用
      AuthRequest authRequest = new AuthRequest();
      authRequest.setTokenSNO("A86A37C4C3C0D1DA63735C60C911B4988A69A654E4B2D9EA0C5C2A2383168FFE");
      //获取结果
      AuthResult authResult = authClient.getCorpUser(authRequest);

}
```

7.4.3 获取法人账号资料服务

获取法人账号信息(明文)。通过令牌唯一识别号获取法人账号信息接口。具体接口调用方式如下：

a) 调用方法：AuthResult getCorpUserAccount(AuthRequest request)；

b) 接口参数说明见表 17；

表17 获取法人账号资料参数说明

类别	参数	必填	数据类型	说明
入参	tokenSNO	是	字符串	令牌唯一识别号
出参	code	是	字符串	响应码，应符合附录B中表B.1的规定
	msg	是	字符串	响应信息
	corpAccount	是	字符串	法人账号基本信息（明文），应符合附录B中表B.7的规定

c) 接口返回值应符合附录 B 中表 B.1 的规定；

d) 调用方法示例。

示例：

```
import gov.zwfw.iam.auth.client.TacsAuthClient;
import gov.zwfw.iam.auth.request.AuthRequest;
import gov.zwfw.iam.auth.response.AuthResult;
import gov.zwfw.iam.client.TacsHttpClient;
import gov.zwfw.iam.exception.TacsException;
public class Demo {
    public void main(String[] args) {
        TacsHttpClient.init("证书路径","环境地址");
```

```
        TacsAuthClient authClient = TacsAuthClient.getInstance();
        //发起调用
        AuthRequest authRequest = new AuthRequest ();
        authRequest.setTokenSNO("***");
        //获取结果
        try {
            AuthResult authResult = authClient. getCorpUserAccount(authRequest);
        } catch (TacsException e) {
            e.printStackTrace();
        }
    }

}
```

7.4.4 获取令牌服务

根据一次性票据信息（ticketSNO）获取当前用户的登录令牌服务。具体接口调用方式如下：

a) 调用方法：AuthResult getTokenWithTicket(AuthRequest request)；

b) 接口参数说明见表 18；

表18 通用认证参数说明

类别	参数	必填	数据类型	说明
入参	ticketSNO	是	字符串	令牌唯一识别号
出参	code	是	字符串	响应码，应符合附录B中表B.1的规定
	msg	是	字符串	响应信息
	token	是	字符串	令牌信息，应符合附录B中表B.8的规定

c) 接口返回值应符合附录 B 中表 B.1 的规定；

d) 调用方法示例。

示例：

```
public void sample()
    {
     //初始化客户端
     TacsHttpClient.init(证书路径，环境地址);
     //获取客户端
      TacsAuthClient authClient = TacsAuthClient.getInstance();
      //发起调用
       AuthRequest authRequest = new AuthRequest();
      authRequest.setTicketSNO("24e356235dab4918b657b054c60f5b4d116820A6A6286C96401A51F3E715D32D ");
       //获取结果
       AuthResult authResult = authClient. getTokenWithTicket (authRequest);

    }
```

7.4.5 令牌延期服务

通过令牌唯一识别码（refreshToken）延期令牌失效时间接口。具体接口调用方式如下：

a) 调用方法：AuthResult delayToken(AuthRequest request)；

b) 接口参数说明见表 19；

表19 令牌延期参数说明

类别	参数	必填	数据类型	说明
入参	refreshToken	是	字符串	刷新令牌唯一识别号
出参	code	是	字符串	响应码，应符合附录B中表B.1的规定
	msg	是	字符串	响应信息
	token	是	字符串	令牌信息，应符合附录B中表B.8的规定

c) 接口返回值应符合附录 B 中表 B.1 的规定；

d) 调用方法示例。

示例：

```
public void sample()
{
     //初始化客户端
     TacsHttpClient.init(证书路径，环境地址);
     //获取客户端
        TacsAuthClient authClient = TacsAuthClient.getInstance();
        //发起调用
        AuthRequest authRequest = new AuthRequest ();
       authRequest.setRefreshToken("A86A37C4C3C0D1DA63735C60C911B4988A69A654E4B2D9EA0C5C2A2383168FFE")
;
        //获取结果
       AuthResult authResult = authClient.delayToken(authRequest);

}
```

7.5 用户中心 SDK

7.5.1 自然人隐性登录服务

实现传入证件号hash值和认证授权码获取自然人登录令牌，传入证件号hash值和认证授权码获取登录令牌接口。具体接口调用方式如下：

a) 调用方法：UserResult getNaturalToken(UserRequest request)；

b) 接口参数说明见表 20；

表20 自然人隐性登录参数说明

类别	参数	必填	数据类型	说明
入参	authCode	是	字符串	认证授权码
	certKey	是	字符串	证件号hash值
出参	code	是	字符串	响应码，应符合附录B中表B.1的规定
	msg	是	字符串	响应信息
	token	是	字符串	令牌信息，应符合附录B中表B.8的规定

c) 接口返回值应符合附录 B 中表 B.1 的规定；

d) 调用方法示例。

示例：

```
public void sample()
    {
```

```
    //初始化客户端
    TacsHttpClient.init(证书路径，环境地址);
    //获取客户端
    TacsNaturalClient userClient = TacsNaturalClient.getInstance();
    //发起调用
    UserRequest userRequest = new UserRequest();
    userRequest.setAuthCode("01");
    userRequest.setCertKey("01");
    //获取结果
     UserResult userResult = userClient. getNaturalToken(userRequest);

   }
```

7.5.2 法人隐性登录接口

实现传入证件号hash值和认证授权码获取法人登录令牌，传入证件号hash值和认证授权码获取法人登录令牌接口。具体接口调用方式如下：

a) 调用方法：CorpResult getCorpToken(CorpRequest request)；

b) 接口参数说明见表 21；

表21 法人隐性登录参数说明

类别	参数	必填	数据类型	说明
入参	accountType	是	字符串	账号类型
	authCode	是	字符串	认证授权码
	corpKey	是	字符串	法人机构代码散列值
	corpType	是	字符串	法人类型
	certKey	是	字符串	信任传递标识散列值
出参	code	是	字符串	响应码，应符合附录B中表B.1的规定
	msg	是	字符串	响应信息
	token	是	字符串	应符合附录B中表B.8的规定

c) 接口返回值应符合附录 B 中表 B.1 的规定；

d) 调用方法示例。

示例：

```
public void sample()
    {
      //初始化客户端
      TacsHttpClient.init(证书路径，环境地址);
      //获取客户端
      TacsCorpClient corpClient = TacsCorpClient.getInstance();
      //发起调用
      CorpRequest corpRequest = new CorpRequest ();
      corpRequest.setAuthCode("*****");
      corpRequest.setCertKey("*****");
      corpRequest.setCorpKey("*****");
      corpRequest.setCorpType("CP01");
      corpRequest.setAccountType("2001");
      //获取结果
       CorpResult corpResult = corpClient. getCorpToken (corpRequest);
```

```
    }
```

7.5.3 登出服务

实现用户登出服务。具体接口调用方式如下：

a) 调用方法：UserResult logout(UserRequest request)；

b) 接口参数说明见表 22；

表22 登出服务参数说明

类别	参数	必填	数据类型	说明
入参	certKey	是	字符串	证件号hash值
出参	code	是	字符串	响应码，应符合附录B中表B.1的规定
	msg	是	字符串	响应信息

c) 接口返回值应符合附录 B 中表 B.1 的规定；

d) 调用方法示例。

示例：

```
public void sample() {
     TacsHttpClient.init(证书路径, 环境地址);
     TacsNaturalClient client = TacsNaturalClient.getInstance();
     UserRequest userRequest = new UserRequest();
     userRequest.setTokenSNO("*********");
     UserResult result = client.logout(userRequest);

}
```

7.6 散列函数服务

7.6.1 身份证散列函数服务

实现用户登出服务。具体接口调用方式如下：

a) 调用方法：HashResult genCertHash (HashRequest request)；

b) 接口参数说明见表 23；

表23 身份证散列函数参数说明

类别	参数	必填	数据类型	说明
入参	certNo	是	字符串	身份证号
	userName	是	字符串	姓名
出参	code	是	字符串	响应码，应符合附录B中表B.1的规定
	msg	是	字符串	响应信息
	data	是	字符串	散列值

c) 接口返回值应符合附录 B 中表 B.1 的规定；

d) 调用方法示例。

示例：

```
public void sample()
{
    TacsHttpClient.init(证书路径, 环境地址);
```

```
    TacsHashClient tacsHashClient = TacsHashClient.getInstance();
    HashRequest hashRequest = new HashRequest();
    hashRequest.setCertNo("232301199212141515");
    hashRequest.setUserName("***");
    HashResult result = tacsHashClient.genCertHash(hashRequest);

}
```

7.6.2 手机号散列函数服务

实现传入手机号返回手机号散列值。具体接口调用方式如下：

a) 调用方法：HashResult genMobileHash (HashRequest request)；

b) 接口参数说明见表 24；

表24 手机号散列函数参数说明

类别	参数	必填	数据类型	说明
入参	mobile	是	字符串	手机号
出参	code	是	字符串	响应码，应符合附录B中表B.1的规定
	msg	是	字符串	响应信息
	data	是	字符串	散列值

c) 接口返回值应符合附录 B 中表 B.1 的规定；

d) 调用方法示例。

示例：

```
public void sample()
 {
    TacsHttpClient.init(证书路径，环境地址);
    TacsHashClient tacsHashClient = TacsHashClient.getInstance();
    HashRequest hashRequest = new HashRequest();
    hashRequest.setMobile("13666666666");
    HashResult result = tacsHashClient.genMobileHash(hashRequest);

}
```

7.6.3 统一社会信用代码散列函数服务

实现传入统一社会信用代码返回统一社会信用代码散列值。具体接口调用方式如下：

a) 调用方法：HashResult genCreditCodeHash (HashRequest request)；

b) 接口参数说明见表 25；

表25 统一社会信用代码散列函数参数说明

类别	参数	必填	数据类型	说明
入参	certificateSno	是	字符串	统一信用代码
出参	code	是	字符串	响应码，应符合附录B中表B.1的规定
	msg	是	字符串	响应信息
	data	是	字符串	散列值

c) 接口返回值应符合附录 B 中表 B.1 的规定；

d) 调用方法示例。

示例：

```
public void sample()
{
    TacsHttpClient.init(证书路径，环境地址);
    TacsHashClient tacsHashClient = TacsHashClient.getInstance();
    HashRequest hashRequest = new HashRequest();
    hashRequest. setCertificateSno ("91430111MA4L16JQ9B");
    HashResult result = tacsHashClient. genCreditCodeHash(hashRequest);

}
```

7.6.4 通用散列函数服务

实现传入字符串返回一个散列值。具体接口调用方式如下：

a) 调用方法：HashResult genCommon(HashRequest request)；

b) 接口参数说明见表 26；

表26 通用散列函数参数说明

类别	参数	必填	数据类型	说明
入参	common	是	字符串	输入字符串
出参	code	是	字符串	响应码，应符合附录B中表B.1的规定
	msg	是	字符串	响应信息
	data	是	字符串	散列值

c) 接口返回值应符合附录 B 中表 B.1 的规定；

d) 调用方法示例。

示例：

```
public void sample()
{
    TacsHttpClient.init(证书路径，环境地址);
    TacsHashClient tacsHashClient = TacsHashClient.getInstance();
    HashRequest hashRequest = new HashRequest();
    hashRequest. setCommon("23123132131231");
    HashResult result = tacsHashClient. genCommon(hashRequest);

}
```

附 录 A
（规范性）
统一认证平台（4A）代码集

A.1 验证票据返回码

验证票据返回码以1位数字表示，其代码见表A.1。

表A.1 验证票据返回码

结果代码	含义	英文描述
0	通过认证	the user passed authentication
1	访问请求被限制	the request has been limited
2	单点登录票据已经被成功创建	the sso ticket has been created successfully
3	验证的票据是无效的，或者已经过期	the sso ticket is invalid , it's overdue
9	登录目标资源非法	the authen target is illegal
43	登录目标资源被禁止	the target has been forbid

A.2 统一认证平台（4A）返回码

统一认证平台返回码以1位数字表示，其代码见表A.2。

表A.2 统一认证平台（4A）返回码说明

结果代码	含义
0	调用成功
1	必填字段验证失败
2	签名随机数生成异常
3	签名随机数生成为空
4	会话已过期
5	获取会话信息异常
6	签名随机数验证失败
7	签名随机数验证异常
8	从帐号错误
9	从帐号密码错误
10	从帐号密码加解密异常
11	保存session信息失败

表A.2 统一认证平台（4A）返回码说明（续）

结果代码	含义
12	短信验证码验证失败
13	短信验证码验证异常
99	其他错误或异常

附 录 B
（规范性）
统一身份认证系统（2A）代码集

B.1 统一身份认证系统（2A）返回码

统一身份认证系统（2A）返回码以5位数字表示，其代码见表B.1。

表B.1 统一身份认证系统（2A）返回码

结果代码	含义
80000	请求成功ok
90001	注册失败
90002	用户不存在
90003	用户名或密码不能为空
90004	用户名或密码不正确
90005	手机号格式不正确
90006	校验码不正确
90007	用户名已经被使用
90008	输入不能为空
90009	修改密码失败
90010	获取用户信息错误
90011	发送验证码失败
90012	证件号已经被使用
90013	手机号码已经被使用
90014	身份证不合法
90015	手机验证码错误
90016	用户名称或手机号或身份证填写不正确

表B.1　统一身份认证系统（2A）返回码（续）

结果代码	含义
90017	用户名称不合法，首字母不能位数字，不能包含汉字，长度6到20个字符
90018	身份证号格式不正确
90019	企业统一信用代码已经被使用
90020	请先注册自然人账号
90021	输入的数据为空
90022	请使用您注册的手机号码
90023	请输入注册时的手机号码
90024	密码不正确
90025	登录已失效，请重新登录
90026	身份证信息校验不通过
90027	请刷新验证码并从新填写
90028	登录成功
90029	登录失败
90030	验证码输入错误
90031	输入数据格式有误
90032	请刷新页面
90033	请勿重复提交
90035	未给用户授权
90036	登陆令牌失效
90037	该用户已登录，请勿重复登录
90040	验证码发送太频繁，请稍候再操作

表B.1　统一身份认证系统（2A）返回码（续）

结果代码	含义
90041	操作太频繁，请稍候再操作
90042	账户已经被冻结,请两小时后重试
90043	数据验证失败，数据已经被修改，请刷新重试
90044	查询数据不存在
90045	请您先取消该账号授权
90046	二次认证密码失败
90047	刷脸登录失败
90048	验证用户与当前用户不匹配
90051	子账号状态异常，不能进行操作
90052	无此用户信息
90053	手机号绑定失败
90054	实名认证失败
90055	授权码为空
90056	修改身份证信息和登录信息不符
90057	不能以自然人身份登录
90058	密码强度不够，不符合要求
90059	旧密码不正确
90060	没有登录权限，不能登录到该企业中心
90061	证件格式不正确
90062	法人类型不正确
90063	切换的法人目标不存在

表B.1　统一身份认证系统（2A）返回码（续）

结果代码	含义
90064	同一身份证只能在同一企业下注册一次
90065	身份证时间间隔有误
90066	数据修改失败
90093	中级实名认证失败
90067	当前登录用户没有查询权限
90068	授权码有误或者已过期
90069	实名有效期已过
90070	经办人账号已过有效日期
90071	授权码有误或已使用
90072	企业法人实名核验不通过
90073	工商查询结果不通过
90074	经办人账号不存在
90075	虎符验签失败
90076	虎符绑定失败
90077	虎符保护失败
90078	虎符登陆失败
90079	省用户请先设置国家平台密码
90080	操作失败，请刷新重试
90081	自然人账号不存在
90082	安全验证不通过
90083	操作频繁，请稍后重试

表B.1　统一身份认证系统（2A）返回码（续）

结果代码	含义
90084	当前机器IP已被锁定2个小时
90085	手机验证码和手机号不匹配
90090	非法操作
90091	非法字符集
90092	非法请求
90093	您不是省同步用户无需账号激活
90094	二维码失效， 请点击刷新二维码
90095	访问地址不合法无法登录
90096	用户数据不存在，请先在国家平台注册用户
90097	绑定过程出错，请返回首页重新扫码绑定
90098	您的账号不是待激活状态
90099	账号没有绑定，请绑定账号
90101	输入信息过长,请重试
90102	当前设备已被锁定2个小时
90103	查询信任传递certKey失败
90104	身份证有效期截止日期有误

B.2　统一身份认证系统（2A）实名核验等级

统一身份认证系统（2A）实名核验等级应满足C 0114-2018《国家政务服务平台可信身份等级定级要求》中的规定，实名核验等级以1位数字表示，其代码见表B.2。

表B.2 统一身份认证系统（2A）实名核验等级

编码	核验要素
1	邮箱、账号口令。
2	手机号、短信码。
3	身份证实名核验（公民身份号码、姓名、有效期限）。

表B.2 统一身份认证系统（2A）实名核验等级（续）

编码	核验要素
4	在三级基础上进行实人核验，使用人脸或其它生物特征进行核验。
5	在四级基础上，使用身份证专用识别设备或具有射频功能的手机配合专用 APP 进行实证核验。

B.3 统一身份认证系统（2A）证件类型

统一身份认证系统（2A）证件类型以3位数字表示，其代码见表B.3。

表B.3 统一身份认证系统（2A）证件类型

编码	核验要素
111	身份证
414	普通护照
513	往来港澳通行证
517	往来台湾通行证
516	港澳居民往来内地通行证
511	台湾居民往来内地通行证
553	外国人永久居留证
554	外国人居留证或居留许可

B.4 统一身份认证系统（2A）自然人数据结构

统一身份认证系统（2A）自然人数据结构应满足C 0111-2018《国家政务服务平台统一身份认证系统身份认证技术要求》中的规定，见表B.4。

表B.4 统一身份认证系统（2A）自然人数据结构

英文名称	中文名称	数据类型	最大长度(字节)	属性说明
userName	用户姓名	String	128	用户姓名
loginName	用户名	String	64	系统登录账号名
certType	证件类型	String	6	应符合表B.3的规定
certNo	身份证号	String	64	身份证号
userMobile	手机号	String	64	手机号
userRealLvl	实名等级	String	6	应符合表B.2的规定
userRealDate	实名认证时间	String	64	实名认证时间
certEffDate	身份证生效日期	String	8	YYYYMMDD
certExpDate	身份证失效日期	String	8	YYYYMMDD
userStatus	用户状态	String	1	1：正常状态 9：待激活状态

B.5 统一身份认证系统（2A）企业法人数据结构

统一身份认证系统（2A）企业法人数据结构应满足C 0111-2018《国家政务服务平台统一身份认证系统身份认证技术要求》中的规定，见表B.5。

表B.5 统一身份认证系统（2A）企业法人数据结构

英文名称	中文名称	数据类型	最大长度	属性说明
certificateSno	统一社会信用代码/登记证号/组织号	String	128	—
corpName	企业/社会组织/事业单位/个体工商户名称	String	64	—
corpType	法人类型	String	64	应符合表B.6的规定
legalMobile	法定代表人手机号	String	64	—
legalName	法定代表人姓名	String	64	—
legalCertNo	法定代表人身份证号	String	64	—
legalCertnoBeginDate	法定代表人身份证生效日期	String	8	YYYYMMDD
legalCertnoEndDate	法定代表人身份证失效日期	String	8	YYYYMMDD

B.6 统一身份认证系统（2A）法人类型

统一身份认证系统证件（2A）类型以3位字母和数字表示，其代码见表B.6。

表B.6 统一身份认证系统（2A）法人类型

编码	核验要素
C01	法人企业
C02	社团法人
C03	事业机关法人
C04	个体工商户

B.7 统一身份认证系统（2A）法人账号信息

统一身份认证系统（2A）法人账号信息见表B.7。

表B.7 统一身份认证系统（2A）法人账号信息

英文名称	中文名称	数据类型	最大长度	属性说明
acctType	法人账号类型	String	64	2001：法人主账号 2004：经办人账号

表 B.7　统一身份认证系统（2A）法人账号信息(续)

英文名称	中文名称	数据类型	最大长度	属性说明
acctNo	法人账号名	String	64	—
trustAcctNo	信任标识	String	64	—
agentMobile	手机号	String	64	—
agentName	姓名	String	64	—
agentCert	身份证号	String	64	—
agentCertBeginDate	身份证有效期起始日期	String	64	YYYYMMDD
agentCertEndDate	身份证有效期结束日期	String	64	YYYYMMDD
corporatorStatus	法人状态	String	6	—
agentChannel	信息来源	String	64	—

B.8　统一身份认证系统（2A）TOKEN 数据结构

统一身份认证系统（2A）TOKEN数据结构见表B.8。

表B.8 统一身份认证系统（2A）TOKEN 数据结构

英文名称	中文名称	数据类型	最大长度	属性说明
tokenSNO	令牌号	String	128	令牌唯一识别码
subjectSNO	身份识别码	String	64	身份证号码/法人统一信用代码 hash 值，对法人账号而言，为信任标识账号散列值。
subjectType	身份识别码类型	String	8	应符合表 B.8 的规定
termType	令牌类型	String	8	令牌类型
refreshToken	刷新令牌	String	64	—
effDate	令牌生效时间	String	64	令牌生效时间
expDate	令牌失效时间	String	64	令牌失效时间

B.9　统一身份认证系统（2A）身份识别码类型

统一身份认证系统（2A）身份识别码类型以2位字母表示，其代码见表B.9。

表B.9 统一身份认证系统（2A）身份识别码类型

编码	身份类型
NA	自然人
CP	法人
PY	经办人
C04	个体工商户

B.10 统一身份认证系统（2A）AuthResult 类数据格式

统一身份认证系统（2A）AuthResult类数据格式见表B.10。

表B.10 AuthResult 类数据格式表

英文名称	中文名称	数据类型	说明
token	令牌	Token	应符合表 B.7 的规定
code	响应码	String	应符合表 B.1 的规定
msg	响应信息	String	应符合表 B.1 的规定
user	自然人基本信息	NaturalUser	应符合表 B.4 的规定
corporator	法人主体信息	Corporator	应符合表 B.5 的规定
corpAccount	法人账号信息	CorpAccount	应符合表 B.7 的规定