

# “浙里办”统一单点登录接入指引

版本号：1.0.5

# 目录

一、 概述 .....	3
二、 接入流程 .....	3
三、 接入示例 .....	4
1. 移动端接入 .....	4
2. PC 端接入 .....	5
四、 接口调用方式 .....	7
1. 请求说明 .....	7
2. 鉴权参数说明 .....	7
3. 接口签名计算代码示例 .....	8
五、 接口说明 .....	11
1. 基于单点登录票据换取请求 token .....	11
(1) 请求地址 .....	11
(2) 入参 .....	11
(3) 出参 .....	11
(4) 错误码 .....	11
2. 基于 token 获取用户信息 .....	12
(1) 请求地址 .....	12
(2) 入参 .....	12
(3) 出参 .....	12
(4) 错误码 .....	14
六、 技术支持 .....	14

# 一、概述

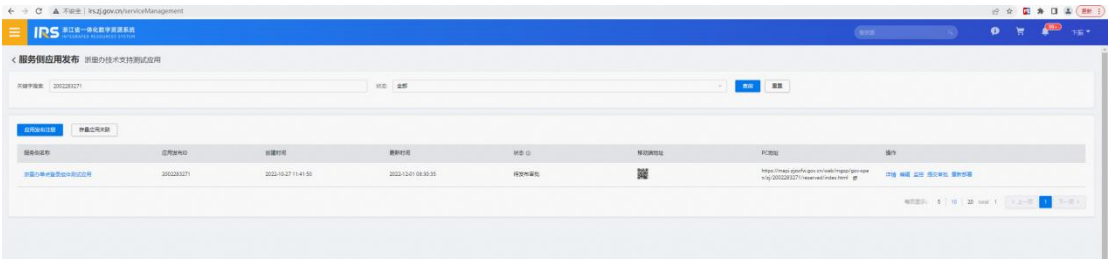
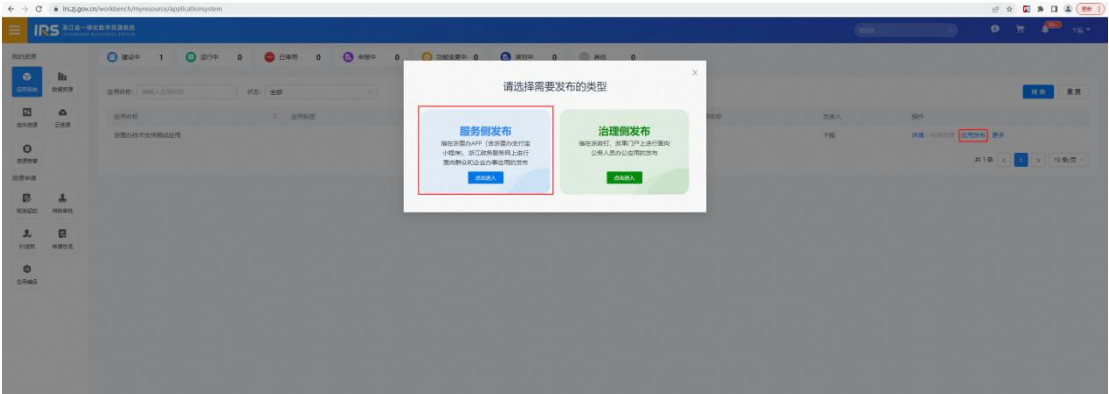
本规范规定了“浙里办”统一单点登录的接入流程、接口调用方式。适用于便民惠企服务接入“浙里办”时单点登录技术接入指引。

# 二、接入流程

- 1) IRS 应用管理员通过“浙里办”单点登录组件详情页面提交申请审核。  
<https://csss.zj.gov.cn/comDetail?comId=958289093569708032>
- 2) 组件申请通过后，IRS 应用管理员通过 IRS 系统工作台获取 AK&SK。  
<https://irs.zj.gov.cn/workbench/myresource/appcomponent>



- 3) IRS 应用管理员通过 IRS 应用发布注册获取应用发布 ID（appid）  
<https://irs.zj.gov.cn/workbench/myresource/applicationsystem>



- 4) 非 IRS 应用发布部署应用系统，应用管理员携带 AK&SK 联系“浙里办”技术支持对接分配 appid。

## 三、接入示例



### 1. 移动端接入

移动端接入“浙里办”APP、“浙里办”微信小程序、“浙里办”支付宝小程序，单点登录可通过使用 ZWJSBridge 参考下述代码块：

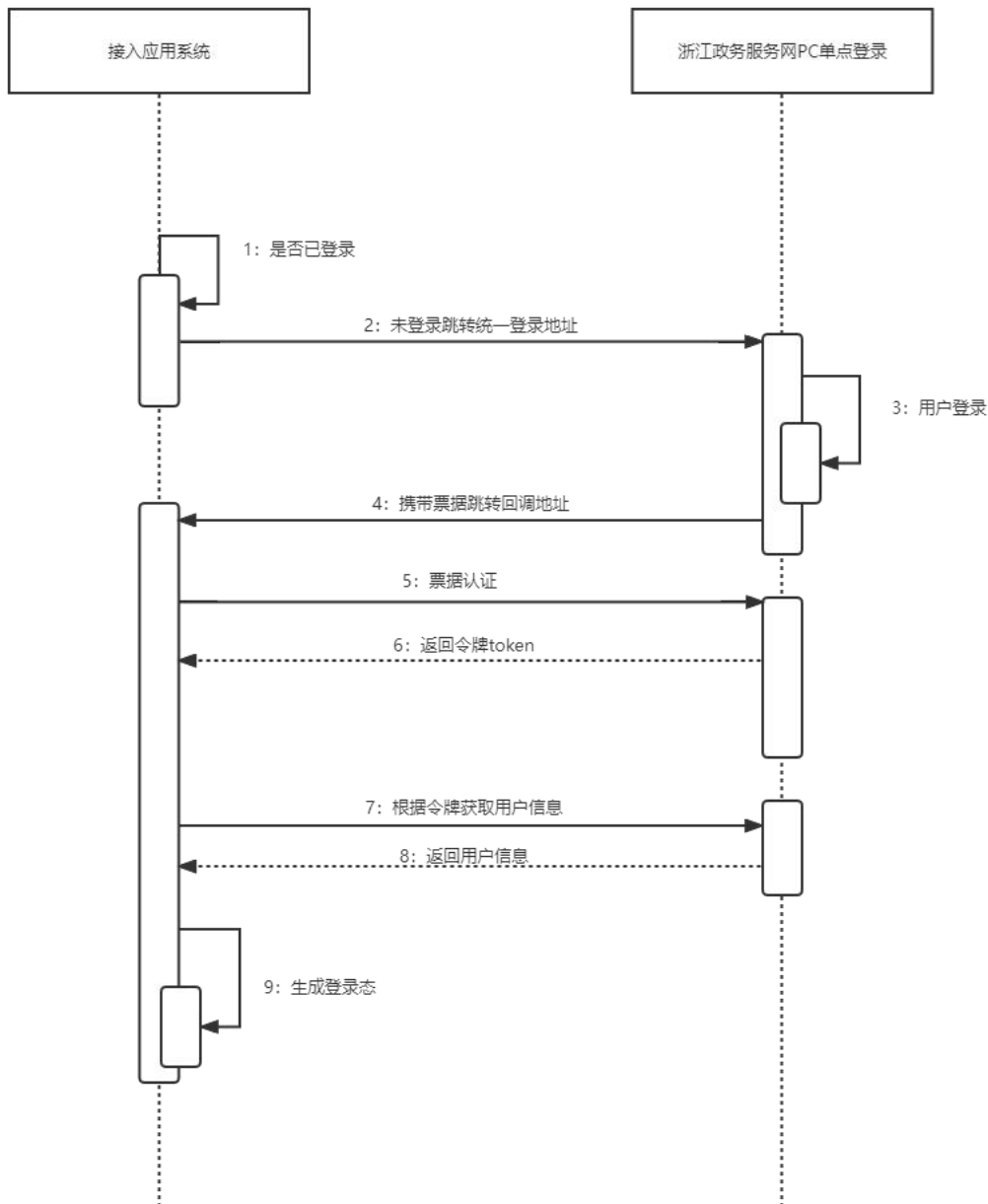
```
if(ZWJSBridge.ssoTicket){
    const ssoFlag = await ZWJSBridge.ssoTicket({});
    if(ssoFlag && ssoFlag.result === true){
        //使用 IRS“浙里办”统一单点登录组件
        if(ssoFlag.ticketId){
            // TODO 应用方服务端单点登录接口
        }
    }
}
```

```
    } else {  
        //异常情况：当前环境不支持“浙里办”统一单点登录  
    }  
}else{  
    //异常情况：ZWJSBridge 加载异常  
}
```

- 1) 移动端票据（ticketId）由用户进入服务时携带，H5 微应用可通过 ZWJSBridge.ssoTicket 获取相关参数：result 指容器环境是否支持“浙里办”统一单点登录，ticketId 指“浙里办”统一单点登录票据。
- 2) IRS 应用发布注册未审核上架“浙里办”的 H5 微应用，通过单点登录获取为测试用户数据。发布上架后，可获取正式用户数据。

## 2. PC 端接入

PC 端接入浙江政务服务网单点登录，通过请求统一单点登录入口 url 地址返回票据（ticketId）至应用系统配置的回调地址进行业务处理获取用户信息。流程图示：



1) 提交 IRS 工单（分类：“浙里办”应用发布）  
<https://irsform.zj.gov.cn/flowable-web/zhejiangOrder/startOrder> 附  
 组件 AK、PC 端回调地址、IRS 应用编码、IRS 应用发布 appid 信息，配置 PC 端回调  
 地址。

2) PC 端统一登录地址

[https://portal.zjzfwf.gov.cn/uc/sso/login?appId=\\${appId}&sp=\\${sp}](https://portal.zjzfwf.gov.cn/uc/sso/login?appId=${appId}&sp=${sp})

3) PC 端统一登出地址 <https://portal.zjzfwf.gov.cn/uc/unifiedLogout>

4) 跳转统一登录地址时 appId 传入在 IRS 应用发布注册的应用发布 ID，sp 会作为返  
 回参数，传给回调地址。注意：sp 地址中如需要带参数和符号如？、&、#等，需进行  
 url 编码处理。

5) 通过统一登录地址登录成功后，浙里办用户中心会将票据 ticket（参数名为  
 ticketId）以及重定向地址（参数名为 returnUrl）以 GET 方式回调应用系统 PC 端回  
 调地址，接入系统收到票据之后调用票据认证接口进行票据认证。

# 四、接口调用方式

## 1. 请求说明

请以 POST 方式提交请求，参数以 application/json 形式提交。

## 2. 鉴权参数说明

“浙里办”单点登录，HTTP 请求都必须在请求头（HTTP Header）中设置如下 4 个参数：

参数名	是否必填	类型	说明
X-BG-HMAC-SIGNATURE	是	string	API 输入参数签名结果
X-BG-HMAC-ALGORITHM	是	string	签名的摘要算法，当前仅支持 hmac-sha256。
X-BG-HMAC-ACCESS-KEY	是	string	分配给应用的 accessKey，例如：12345678。
X-BG-DATE-TIME	是	string	时间戳，时区为 GMT+8，格式为：Tue, 09 Nov 2021 08:49:20 GMT。服务端允许客户端请求最大时间误差为 100 秒。

其中 X-BG-HMAC-SIGNATURE 的计算公式为：

signature = HMAC-SHA256-HEX(secret\_key,signing\_string)

各字段解释如下：

secret\_key 为接口申请完成后获取到的 secret\_key

signing\_string 由请求方法、URI、请求参数等拼接获得，具体如下：

HTTP\_METHOD+\n+HTTP\_URI+\n+QUERY\_STREAM+\n+X-BG-HMAC-ACCESS-KEY+\n+X-BG-DATE+\n

参数解释如下图，详细代码可参考签名计算代码

参数名	说明
HTTP METHOD	指 HTTP 协议中定义的 GET、PUT、POST 等请求方法，必须使用全大写的形式。
HTTP URI	请求路径，要求必须以 “/” 开头，不以 “/” 开头的需要补充上，空路径为 “/”
X-BG-DATE	请求头中的 Date （ GMT 格式 ）格式为：“Tue, 09 Nov 2021 08:49:20 GMT”
QUERY_STREAM	是对于 URL 中的 query（ query 即 URL 中?后面的 key1=valve1&key2=valve2 字符串）进行编码后的结果。以 key 按照字典顺序（ ASCII 码由小到大）排序，并使用 & 符号连接起来，生成相

参数名	说明
	应的 query_string。

### 3. 接口签名计算代码示例

```
import javafx.util.Pair;
import lombok.extern.slf4j.Slf4j;

import javax.crypto.Mac;
import javax.crypto.spec.SecretKeySpec;
import javax.xml.bind.DatatypeConverter;
import java.io.UnsupportedEncodingException;
import java.net.URI;
import java.net.URL;
import java.net.URLEncoder;
import java.text.DateFormat;
import java.text.SimpleDateFormat;
import java.util.*;
import java.util.stream.Collectors;

@Slf4j
public class HmacAuthUtil {

    /**
     * 构造请求 header
     * @param urlStr 请求 url ， 全 路 径 格 式 ， 比 如 ：
https://ibcdsg.zj.gov.cn:8443/restapi/prod/IC33000020220228000002/sso/servlet/simpleauth
     * @param requestMethod 请求方法,大写格式，如：GET，POST
     * @param accessKey 应用的 AK
     * @param secretKey 应用的 SK
     * @return
     */
    public static Map<String, String> generateHeader(String urlStr,String requestMethod, String accessKey, String secretKey) {

log.info("params,urlStr={},requestMethod={},accessKey={},secretKey={}",urlStr,requestMethod,accessKey,secretKey);
        Map<String, String> header = new HashMap<>();
        try {
            DateFormat dateFormat = new SimpleDateFormat("EEE, dd MMM yyyy HH:mm:ss
```



```

z", Locale.US);
    dateFormat.setTimeZone(TimeZone.getTimeZone("GMT"));
    String date = dateFormat.format(new Date());
    URL url = new URL(urlStr);
    URI uri = new URI(url.getProtocol(), url.getHost(), url.getPath(),
url.getQuery(), null);

    String canonicalQueryString = getCanonicalQueryString(uri.getQuery());

    String message = requestMethod.toUpperCase() + "\n" + uri.getPath() + "\n"
+ canonicalQueryString + "\n" + accessKey + "\n" + date + "\n";

    Mac hasher = Mac.getInstance("HmacSHA256");
    hasher.init(new SecretKeySpec(secretKey.getBytes(), "HmacSHA256"));

    byte[] hash = hasher.doFinal(message.getBytes());

    // to lowercase hexits
    DatatypeConverter.printHexBinary(hash);

    // to base64
    String sign = DatatypeConverter.printBase64Binary(hash);
    header.put("X-BG-HMAC-SIGNATURE", sign);
    header.put("X-BG-HMAC-ALGORITHM", "hmac-sha256");
    header.put("X-BG-HMAC-ACCESS-KEY", accessKey);
    header.put("X-BG-DATE-TIME", date);
} catch (Exception e) {
    log.error("generate error", e);
    throw new RuntimeException("generate header error");
}
log.info("header info, {}", header);
return header;
}

private static String getCanonicalQueryString(String query) {
    if (query == null || query.trim().length() == 0) {
        return "";
    }
    List<Pair<String, String>> queryParamList = new ArrayList<>();
    String[] params = query.split("&");
    for (String param : params) {
        int eqIndex = param.indexOf("=");
        String key = param.substring(0, eqIndex);
        String value = param.substring(eqIndex+1);

```

```

        Pair<String, String> pair = new Pair<String, String>(key, value);
        queryParamList.add(pair);
    }

    List<Pair<String, String>> sortedParamList =
queryParamList.stream().sorted(Comparator.comparing(param -> param.getKey() + "=" +
Optional.ofNullable(param.getValue()).orElse(""))).collect(Collectors.toList());
    List<Pair<String, String>> encodeParamList = new ArrayList<>();
    sortedParamList.stream().forEach(param -> {
        try {
            String key = URLEncoder.encode(param.getKey(), "utf-8");
            String value =
URLEncoder.encode(Optional.ofNullable(param.getValue()).orElse(""), "utf-8")
                .replaceAll("\\%2B", "%20")
                .replaceAll("\\+", "%20")
                .replaceAll("\\%21", "!")
                .replaceAll("\\%27", "'")
                .replaceAll("\\%28", "(")
                .replaceAll("\\%29", ")")
                .replaceAll("\\%7E", "~")
                .replaceAll("\\%25", "%")
                ;
            encodeParamList.add(new Pair<>(key, value));
        } catch (UnsupportedEncodingException e) {
            throw new RuntimeException("encoding error");
        }
    });
    StringBuilder queryParamString = new StringBuilder(64);
    for (Pair<String, String> encodeParam : encodeParamList) {
        queryParamString.append(encodeParam.getKey()).append("=").append(Optional.ofNulla
ble(encodeParam.getValue()).orElse(""));
        queryParamString.append("&");
    }

    return queryParamString.substring(0, queryParamString.length() - 1);
}
}

```

# 五、接口说明

## 1. 基于单点登录票据换取请求 token

### (1) 请求地址

政务外网地址：  
https://bcdsg.zj.gov.cn:8443/restapi/prod/IC33000020220329000007/uc/sso/access\_token

互联网地址：  
https://ibcdsg.zj.gov.cn:8443/restapi/prod/IC33000020220329000007/uc/sso/access\_token

### (2) 入参

参数	类型	描述
ticketId	String	单点登录票据
appId	String	应用发布 ID

### (3) 出参

参数	类型	描述
errorCode	String	错误码
errorMsg	String	错误信息
success	Boolean	请求是否成功
data	Object	响应体
- accessToken	String	获取用户信息 token

### (4) 错误码

错误码	描述
C-USER-SSO-TICKET-INVALID	ticket 非法

## 2. 基于 token 获取用户信息

### (1) 请求地址

政务外网地址:

`https://bcdsg.zj.gov.cn:8443/restapi/prod/IC33000020220329000008/uc/sso/getUserInfo`

互联网地址:

`https://ibcdsg.zj.gov.cn:8443/restapi/prod/IC33000020220329000008/uc/sso/getUserInfo`

### (2) 入参

参数	类型	描述
token	String	获取用户信息 token

### (3) 出参

参数	类型	描述
success	Boolean	请求是否成功
errorCode	String	错误码
errorMsg	String	错误信息
data	Object	响应体
- userType	String	用户类型, PERSON 个人/LEGAL_PERSON 法人
- personInfo	Object	个人用户信息, 当前登录自然人的信息
-- userId	String	主键
-- userName	String	个人姓名
-- idType	String	ID_CARD: 身份证, PASSPORT: 护照, OFFICER_CARD: 军官证, MAINLAND_TRAVEL_PERMIT_FOR_HONGKONG_AND_MACAO_RESIDENTS: 港澳居民来往内地通行证, MAINLAND_TRAVEL_PERMIT_FOR_TAIWAN_RESIDENTS: 台湾居民来往大陆通行证

		证, FOREIGN_PERMANENT_RESIDENT_ID_CARD: 外国人永久居留身份证, FOREIGN_PASSPORT: 外籍人士护照, DIPLOMACY_PASSPORT: 外交护照, OFFICIAL_PASSPORT: 公务护照, SOLDIER_CARD: 士兵证, OFFICER_RETIRE_CARD: 军官离退休证, GANG_AO_TAI_RESIDENCE_CART: 港澳台居民居住证, GANG_AO_ID_CART: 港澳居民身份证, UNIFIED_SOCIAL_ID: 统一社会信用代码, OTHER: 其他
-- outerIdType	String	外部证件类型
-- idNo	String	证件编号
-- attnUserType	String	法人经办人时用户类型， 评级
-- phone	String	手机号
-- email	String	邮箱
-- nation	String	民族
-- gender	String	性别
-- birthday	String	生日
-- certKey	String	身份散列值
-- attributes	Object	额外属性
legalPersonInfo	Object	法人用户信息， 比如公司相关的信息
-- name	String	法人名称
-- unifiedSocialId	String	社会统一信用代码
-- orgType	String	法人类型
-- attnName	String	经办人姓名
-- attnPhone	String	经办人手机号
-- attnIdType	String	经办人证件类型
-- attnIdNo	String	经办人证件号码
-- attnUserType	String	经办人用户等级
-- principal	String	法人代表人姓名
-- gender	Integer	法人代表人性别
-- nation	Integer	法人代表人民族
-- idType	Integer	法人代表人证件类型

-- outerIdType	String	法人代表人证件类型
-- idNo	String	法人代表人证件号码
-- principalUserId	String	法人代表唯一键
-- corpId	String	法人唯一键
-- attributes	Object	额外属性
-- organizationInfoList	Array	所属组织信息
-- orgId	String	组织主键
-- oid	String	Alias for orgId
-- parentId	String	父组织主键
-- pid	String	Alias for parentId
-- name	String	组织机构简称
--fullName	String	组织机构全称
--devCoding	String	组织后缀
--leafFlag	Boolean	是否叶子标志
--orderBy	Integer	排序号，从小到大

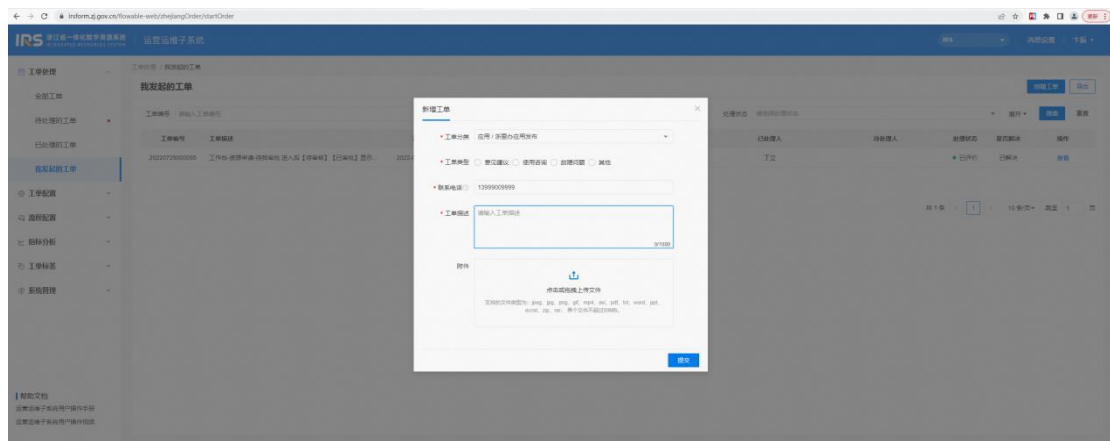
#### (4) 错误码

错误码	描述
C-USER-SSO-TOKEN-INVALID	token 非法
C-USER-SSO-USER-EMPTY	用户信息为空

## 六、技术支持

### 1) “浙里办”技术支持-工单系统

<https://irsform.zj.gov.cn/flowable-web/>



## 2) “浙里办”技术支持-钉钉群

钉钉公开群号：31376954、34340559、31419900、34143965、31140598



## 3) “浙里办”技术支持-浙政钉群

浙政钉公开群号：1630638093、1643885873



## “浙里办”应用发布-2

188 人



请使用浙政钉扫码加入

### 4) “浙里办”技术支持-联系电话

卞振: +86-18868422737

王韶欣: +86-18790696328

罗豪: +86-15629051003