|  |
| --- |
|  |

|  |
| --- |
|  |
| 接口说明-集成认证 |
| 版本 V1.5 |

**会签**

陕西 西安

2021．02

|  |  |  |  |
| --- | --- | --- | --- |
| 作者 | ZhengWei(HY) | 日期 | 2021-02-02 |
| 审核 |  | 日期 |  |
| 批准 |  | 日期 |  |

**版本**

|  |  |  |  |
| --- | --- | --- | --- |
| 版本 | 日期 | 更新 | 签字 |
| 1.0 | 2021-02-02 | 创建 | ZhengWei(HY) |
| 1.1 | 2021-03-03 | 添加判定访问级令牌的有效性 | ZhengWei(HY) |
| 1.2 | 2022-05-11 | 添加绑定会话关系、跨系统获取会话票据、会话保活 | ZhengWei(HY) |
| 1.3 | 2022-06-18 | 获取登录用户接口添加访问级令牌  在线用户简报、用户域字段 | ZhengWei(HY) |
| 1.4 | 2022-10-07 | 登录用户信息添加other、systemID字段;拦截器逻辑 | ZhengWei(HY) |
| 1.5 | 2023-08-02 | 所有接口添加sysid系统编号 | ZhengWei(HY) |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**目 录**

[1. 应用场景 5](#_Toc141878464)

[2. 流程 6](#_Toc141878465)

[2.1 纯接口服务流程 6](#_Toc141878466)

[2.2 用户登录流程 7](#_Toc141878467)

[2.3 应用端拦截器的逻辑 8](#_Toc141878468)

[3. 接口格式 9](#_Toc141878469)

[3.1 生成应用密钥对 9](#_Toc141878470)

[3.2 获取访问Token 10](#_Toc141878471)

[3.3 判定访问级令牌的有效性 12](#_Toc141878472)

[3.4 获取会话级的令牌 13](#_Toc141878473)

[3.5 获取登录用户 16](#_Toc141878474)

[3.7 注销登录用户 19](#_Toc141878475)

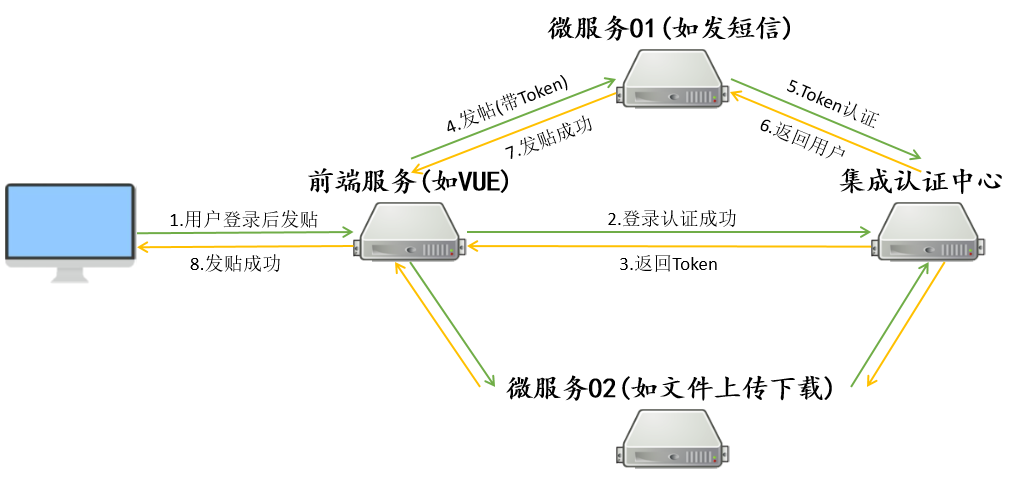
[3.8 绑定会话关系 20](#_Toc141878476)

[3.9 跨系统获取会话票据 21](#_Toc141878477)

[3.10 会话保活 22](#_Toc141878478)

[3.11 在线用户简报 23](#_Toc141878479)

# 应用场景

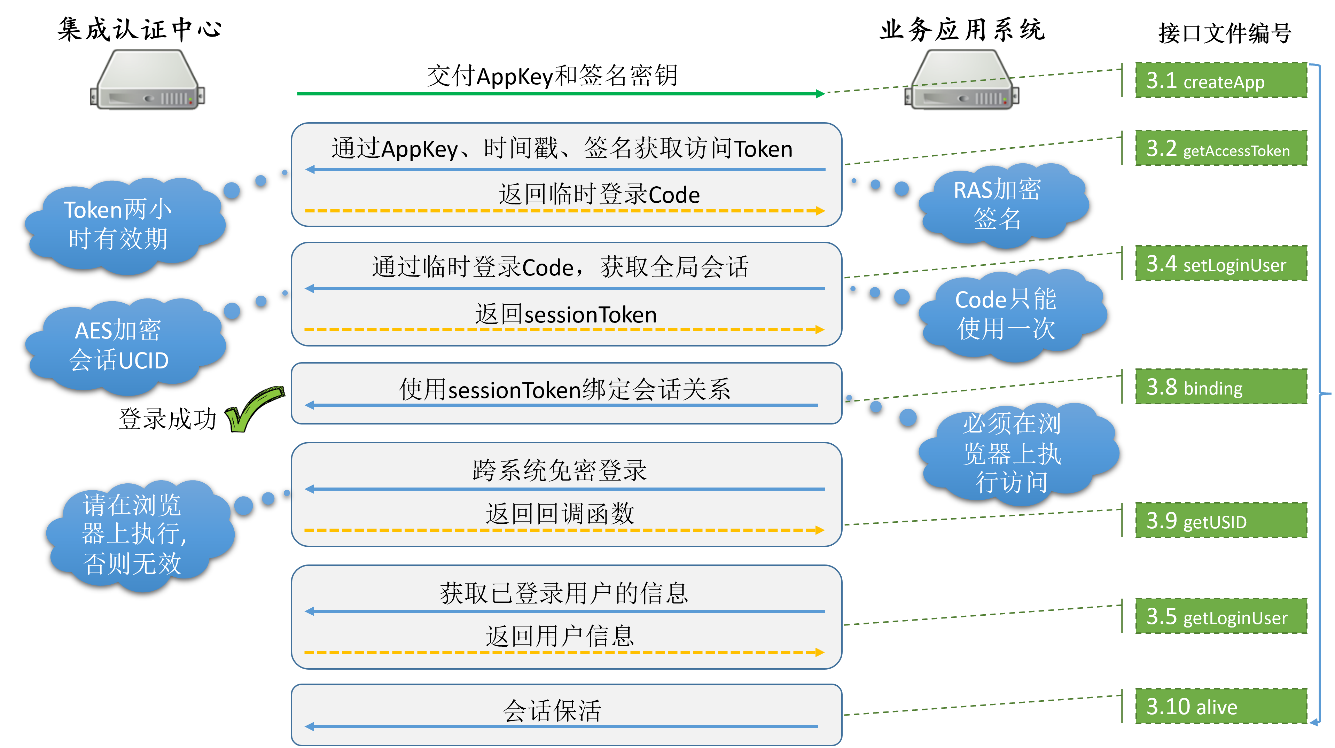


# 流程

### 2.1 纯接口服务流程



### 2.2 用户登录流程

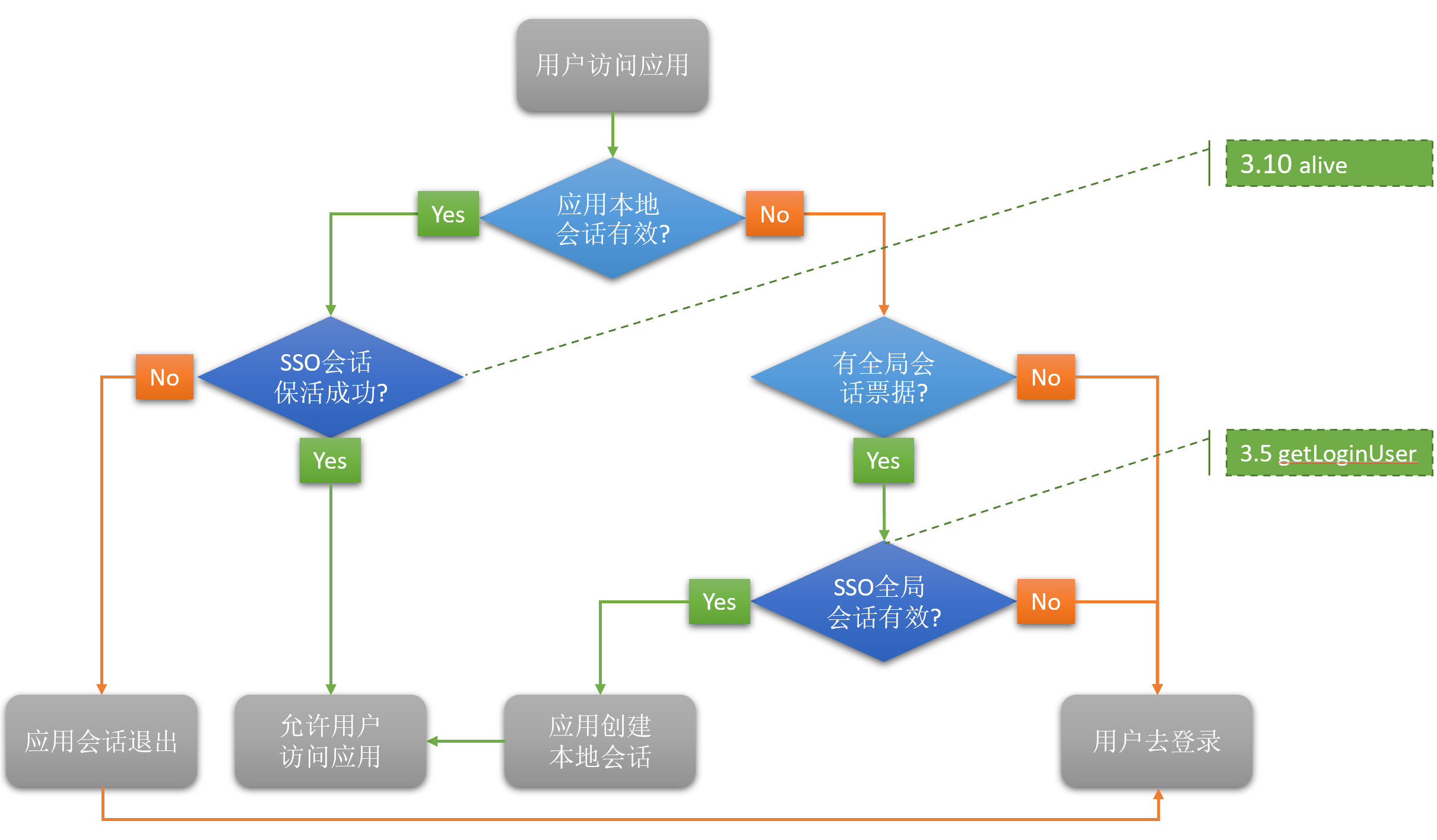


单点登录（Single Sign On），简称为 [SSO](https://baike.baidu.com/item/SSO/3451380?fromModule=lemma_inlink)，是比较流行的企业[业务整合](https://baike.baidu.com/item/%E4%B8%9A%E5%8A%A1%E6%95%B4%E5%90%88/349259?fromModule=lemma_inlink)的解决方案之一。SSO的定义是在多个应用系统中，用户只需要登录一次就可以访问所有相互信任的应用系统。

实现机制为，当用户第一次访问应用系统的时候，因为还没有登录，会被引导到认证系统中进行登录；根据用户提供的登录信息，认证系统进行身份校验，如果通过校验，应该返回给用户一个认证的凭据（会话级Token）；用户再访问别的应用的时候，就会将这个token带上，作为自己认证的凭据，应用系统接受到请求之后会把token送到认证系统进行校验，检查token的合法性。如果通过校验，用户就可以在不用再次登录的情况下访问应用系统2和应用系统3了。

在安全方面的签名signature使用RAS非对称算法，RAS是共认的非常安全的加密算法，签名加密公式为URLEncode( Base64EnCode( RAS( parameters) ) )，也同时满足互联网地对字符的协议要求。

### 2.3 应用端拦截器的逻辑



接口3.10 alive 会话保活无须用户每次访问都做，可间隔一段时间做一次，如间隔30秒。

举例代码如下所示：

**private** **static** **final** ExpireMap<String ,Date> $SSOAliveMap = **new** ExpireMap<String ,Date>();

// 业务方法

// ....

// 业务方法

String v\_SSOAliveKey = "SSO#" + "工号";

**if** ( $SSOAliveMap.get(v\_SSOAliveKey)== **null** )

{

// 调保活接口的开始

// ....

// 调保活接口的结束

// 保活成功时，每人每30秒调用一次会话保活接口就OK

$SSOAliveMap.put(v\_SSOAliveKey ,**new** Date() ,30);

}

# 接口格式

所有的 API 都是基于 HTTP 1.1 [RFC2616]， 采用 REST 风格定义。每一个请求都包含  
HTTP 的 POST、 GET 等请求方法，对应于请求的每一个响应都含有标准的 HTTP 状态  
码(status code)。大部分请求响应还包含了 JSON 格式的消息正文。

### 3.1 生成应用密钥对

此接口对于一个应用只调用一次就好，须将返回信息汇总管理员，并添加到ms.xsso.appKey.Config.xml配置文件中才能生效（请重启服务）。

**使用对象：**应用服务器

#### 3.1.1 接口请求参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **名称** | 生成应用密钥对 | | | | |
| **说明** | 生成应用密钥对 | | | | |
| **地址** | <https://127.0.0.1:80/>服务名称/xsso/createApp | | | | |
| **方式** | GET请求 | | | | |
| **接口请求参数** | | | | | |
| **序号** | **参数名称** | **参数说明** | **参数类型** | **是否必须** | **备注** |
| 01 | sysid | 系统编号 | String | 是 |  |

#### 3.1.2 接口响应参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **接口返回参数** | | | | | |
| **序号** | **参数名称** | **参数说明** | **参数类型** | **是否必须** | **备注** |
| 01 | code | 是否成功标识 | String | 是 | 200为成功 |
| 02 | message | 消息说明 | String | 是 |  |
| **Data对象** | | | | | |
| 03 | appKey | 应用编码 | String | 是 | 注册应用方妥善保存 |
| 04 | privateKey | 私钥加密验证 | String | 是 | 注册应用方妥善保存 |
| 05 | publicKey | 公钥加密验证 | String | 否 | 本服务妥善保存在：ms.xsso.appKey.Config.xml配置文件中。 |

#### 3.1.3 接口举例-请求报文

无

#### 3.1.4 接口举例-响应报文

{

    "code": "200",

    "message": "成功",

    "data": {

        "data": {

            "appKey": "70A78B5680FD4C7C9A9D9522F41F5E5E",

            "privateKey": "",

            "publicKey": "MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQD0ry8BiVBTOg7APlYAwlmYpiqQAOYkje2BpDbtebfUow2LWxX1Gv8kSKO/u0A7CgD2NnI3JyKITUy2m+o/vXJ6qQ7aP1geOPz2FQV9uNvIFUcf1dCO1axASEMeJgWLhOhDikeymCevf7sv/2PMzd80xvFFnpQdv9xTmX4wdSu1sQIDAQAB"

        }

    }

}

### 3.2 获取访问Token

应用创建成功后，会生成应用的 appKey 和 privateKey 的值，租户管理员可将这个两个值交给开发者，开发者可用 appKey 和 privateKey 获取 访问级令牌token。

**使用对象：**应用服务器

#### 3.2.1 接口请求参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **名称** | 获取访问Token | | | | |
| **说明** | 获取系统（或接口）访问级的令牌 access\_token ，是应用访问开放平台业务接口的凭证，有效期为2小时,过期后需要重新获取。 | | | | |
| **地址** | <https://127.0.0.1:80/>服务名称/xsso/getAccessToken | | | | |
| **方式** | GET请求 | | | | |
| **接口请求参数** | | | | | |
| **序号** | **参数名称** | **参数说明** | **参数类型** | **是否必须** | **备注** |
| 01 | appKey | 应用 appKey | String | 是 |  |
| 02 | timestamp | 毫秒时间戳 | Long | 是 | 3分钟内有效。当超过当前时间时，容错1分钟内认为有效。 |
| 03 | signature | 校验签名 | String | 是 |  |
| 04 | sysid | 系统编号 | String | 是 |  |

**signature校验签名的生成方式**

其中，签名字段signature计算使用RAS算法，具体计算方式如下：

URLEncode( Base64EnCode( RAS( parameters) ) )

其中，parameters 按照参数名称排序，参数名称与参数值依次拼接(signature字段除外)，形成待计算签名的字符串。

示例 ：若发送请求参数appKey为0904202FB2EA4AC3A899B423928A0F8E,时间戳timestamp为1609221942639，则待加密字符串的值为 appKey0904202FB2EA4AC3A899B423928A0F8Etimestamp1609221942639

之后对 parameters使用 RAS 计算签名，RAS 的 key 为自建应用的 privateKey。计算出的二进制签名先进行 Base64，之后进行 URLEncode，即得到 signatrue 字段的值。

#### 3.2.2 接口响应参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **接口返回参数** | | | | | |
| **序号** | **参数名称** | **参数说明** | **参数类型** | **是否必须** | **备注** |
| 01 | code | 是否成功标识 | String | 是 | 200为成功 |
| 02 | message | 消息说明 | String | 是 |  |
| **Data对象** | | | | | |
| 03 | accessToken | 访问级的令牌 | String | 否 | 成功时，返回 |
| 04 | expire | 过期时长 | String | 否 | 成功时，返回。  单位：秒  默认为：7200 |
| 05 | code | 临时登录Code | String | 否 | 成功时，返回。  在其后的使用中，仅且只能使用一次，做为其它接口的入参。 |

#### 3.2.3 接口举例-请求报文

[https://127.0.0.1/](http://127.0.0.1/)msXSSO/**getAccessToken**?**appKey**=0904202FB2EA4AC3A899B423928A0F8E&**timestamp**=1609221942639&**signature**=fPAHpZghqNLpssMr%2FeMlbTizQSpCX91hRMRKdrKs0djDCyH4Hb0ozv7yfFSgjGBTGLhd6K%2FAqxE3UQQMmk33OCiuHztNbyRaxSGgrQ75HIVmvriQFYtPTUGOtYR4vmKzGOcxghv4s%2FTdWsGGRWwcK8y%2Bg9ZULRHVUtu8EMErud4%3D

#### 3.2.4 接口举例-响应报文

{

    "code": "200",

    "message": "成功",

    "data": {

        "data": {

            "accessToken": "B8EE43B314E94A5CBC46F7C266043FFA",

            "expire": 7200,

            "code": "51FDA1E47E794AE9A2ED2A51E9B001C7"

        }

    }

}

### 3.3 判定访问级令牌的有效性

**使用对象：**应用服务器 或 终端用户

#### 3.3.1 接口请求参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **名称** | 判定访问级令牌的有效性 | | | | |
| **说明** | 用于系统级的（如接口）的访问权限的判定。 | | | | |
| **地址** | <https://127.0.0.1:80/>服务名称/xsso/getAppKey | | | | |
| **方式** | GET请求 | | | | |
| **接口请求参数** | | | | | |
| **序号** | **参数名称** | **参数说明** | **参数类型** | **是否必须** | **备注** |
| 01 | token | 访问级的令牌 | String | 是 | 。 |
| 02 | sysid | 系统编号 | String | 是 |  |

#### 3.3.2 接口响应参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **接口返回参数** | | | | | |
| **序号** | **参数名称** | **参数说明** | **参数类型** | **是否必须** | **备注** |
| 01 | code | 是否成功标识 | String | 是 | 200为成功 |
| 02 | message | 消息说明 | String | 是 |  |
| **Data对象** | | | | | |
| 03 | appKey | 应用 appKey | String | 否 | 成功时，返回 |
| 04 | expire | 过期时长 | Integer | 否 | 成功时，返回 |

#### 3.3.3 接口举例-请求报文

https://127.0.0.1/msXSSO/xsso/getAppKey?token=3E8DE247F5E84EC592ECF633807BF3A2

#### 3.3.4 接口举例-响应报文

{

    "code": "200",

    "message": "成功",

    "data": {

        "data": {

"appKey": "0904202FB2EA4AC3A899B423928A0F8E"

            "expire": 7200

        }

    }

}

### 3.4 获取会话级的令牌

**使用对象：**应用服务器

#### 3.4.1 接口请求参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **名称** | 获取会话级的令牌 | | | | |
| **说明** | 用于用户成功登录后，设置用户信息，生成与用户信息绑定的会话级的令牌。 | | | | |
| **地址** | <https://127.0.0.1:80/>服务名称/xsso/setLoginUser | | | | |
| **方式** | POST请求 | | | | |
| **接口请求参数** | | | | | |
| **序号** | **参数名称** | **参数说明** | **参数类型** | **是否必须** | **备注** |
| 01 | code | 临时登录Code | String | 是 | Code使用后失效，仅能使用一次。 |
| 02 | sysid | 系统编号 | String | 是 |  |
| **接口请求体** | | | | | |
| 01 | appKey | 应用编码 | String | 是 |  |
| 02 | userId | 用户在第三方用户中心的标识 | String | 是 |  |
| 03 | userNo | 用户在第三方用户中心的工号 | String | 否 |  |
| 04 | userCode | 用户在第三方系统的编码 | String | 否 |  |
| 05 | openID | 微信用户应用ID | String | 否 |  |
| 06 | unionID | 微信用户统一ID | String | 否 |  |
| 07 | mobile | 用户在第三方系统的手机号 | String | 否 |  |
| 08 | phone | 用户在第三方系统的手机号 | String | 否 |  |
| 09 | email | 用户在第三方系统的邮箱 | String | 否 |  |
| 10 | userName | 用户在第三方系统的用户名 | String | 否 |  |
| 11 | nickname | 用户在第三方系统的昵称 | String | 否 |  |
| 12 | userType | 用户在第三方系统的用户类型 | String | 否 |  |
| 13 | userLevel | 用户在第三方系统的用户级别 | String | 否 |  |
| 14 | userSource | 户在第三方系统的用户来源 | String | 否 |  |
| 15 | userLogoUrl | 用户在第三方系统的头像地址 | String | 否 |  |
| 16 | loginAccount | 用户在第三方系统的登陆账号 | String | 否 |  |
| 17 | orgCode | 用户在第三方系统的组织编码 | String | 否 |  |
| 18 | orgName | 用户在第三方系统的组织名称 | String | 否 |  |
| 19 | sex | 用户性别 | String | 否 |  |
| 20 | sessionID | 会话ID | String | 否 |  |
| 21 | ip | 用户IP地址 | String | 否 |  |
| 22 | systemID | 系统编码 | String | 否 |  |
| 23 | other | 其它信息 | String | 否 |  |
| 24 | userDomain | 用户在第三方系统的用户域 | Map  <String ,String> | 否 | 可用于解决一号多重身份的应用场景 |
| 25 | datas | 附加用户数据 | Map  <String ,String> | 否 |  |

#### 3.4.2 接口响应参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **接口返回参数** | | | | | |
| **序号** | **参数名称** | **参数说明** | **参数类型** | **是否必须** | **备注** |
| 01 | code | 是否成功标识 | String | 是 | 200为成功 |
| 02 | message | 消息说明 | String | 是 |  |
| **Data对象** | | | | | |
| 03 | sessionToken | 会话级的令牌 | String | 否 | 成功时，返回 |
| 04 | expire | 过期时长 | Integer | 否 | 成功时，返回 |

#### 3.4.3 接口举例-请求报文

https://127.0.0.1/msXSSO/xsso/setLoginUser?code=3E8DE247F5E84EC592ECF633807BF3A2

{

    "appKey": "D22DCADC06BB4EAE868D5728AACAD370",

    "userId": "9527",

    "userNo": "工号",

    "userCode": "编码",

    "openID": "微信用户应用ID",

    "unionID": "微信用户统一ID",

    "mobile": "手机号",

    "phone": "手机号",

    "email": "邮箱",

    "userName": "用户名",

    "nickname": "昵称",

    "userType": "用户类型",

    "userLevel": "用户级别",

    "userSource": "用户来源",

    "loginAccount": "登陆账号",

    "orgCode": "组织编码",

    "orgName": "组织名称",

    "sex": "用户性别",

"sessionID": "会话ID",

"systemID": "系统编号",

"other": "其它信息",

    "datas": {

        "附加用户数据1": "附加用户数据1",

        "附加用户数据2": "附加用户数据2"

    }

}

#### 3.4.4 接口举例-响应报文

{

    "code": "200",

    "message": "成功",

    "data": {

        "data": {

            "expire": 7177

        }

    }

}

### 3.5 获取登录用户

**使用对象：**应用服务器

**使用限制：**白名单有限使用

#### 3.5.1 接口请求参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **名称** | 获取登录用户 | | | | |
| **说明** | 获取登录用户。 | | | | |
| **地址** | <https://127.0.0.1:80/>服务名称/xsso/getLoginUser | | | | |
| **方式** | GET请求 | | | | |
| **接口请求参数** | | | | | |
| **序号** | **参数名称** | **参数说明** | **参数类型** | **是否必须** | **备注** |
| 01 | USID | 会话级的令牌 | String | 是 | 二选一，与token同义，  为兼容老接口而存在 |
| 02 | token | 会话级的令牌 | String | 是 | 二选一，与USID同义 |
| 03 | accessToken | 获取访问Token | String | 是 | 见 3.2 接口 |
| 04 | sysid | 系统编号 | String | 是 |  |

#### 3.5.2 接口响应参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **接口返回参数** | | | | | |
| **序号** | **参数名称** | **参数说明** | **参数类型** | **是否必须** | **备注** |
| 01 | code | 是否成功标识 | String | 是 | 200为成功 |
| 02 | message | 消息说明 | String | 是 |  |
| **Data对象** | | | | | |
| 01 | appKey | 应用编码 | String | 是 |  |
| 02 | userId | 用户在第三方用户中心的标识 | String | 是 |  |
| 03 | userNo | 用户在第三方用户中心的工号 | String | 否 |  |
| 04 | userCode | 用户在第三方系统的编码 | String | 否 |  |
| 05 | openID | 微信用户应用ID | String | 否 |  |
| 06 | unionID | 微信用户统一ID | String | 否 |  |
| 07 | mobile | 用户在第三方系统的手机号 | String | 否 |  |
| 08 | phone | 用户在第三方系统的手机号 | String | 否 |  |
| 09 | email | 用户在第三方系统的邮箱 | String | 否 |  |
| 10 | userName | 用户在第三方系统的用户名 | String | 否 |  |
| 11 | nickname | 用户在第三方系统的昵称 | String | 否 |  |
| 12 | userType | 用户在第三方系统的用户类型 | String | 否 |  |
| 13 | userLevel | 用户在第三方系统的用户级别 | String | 否 |  |
| 14 | userSource | 户在第三方系统的用户来源 | String | 否 |  |
| 15 | userLogoUrl | 用户在第三方系统的头像地址 | String | 否 |  |
| 16 | loginAccount | 用户在第三方系统的登陆账号 | String | 否 |  |
| 17 | orgCode | 用户在第三方系统的组织编码 | String | 否 |  |
| 18 | orgName | 用户在第三方系统的组织名称 | String | 否 |  |
| 19 | sex | 用户性别 | String | 否 |  |
| 20 | sessionID | 会话ID | String | 否 |  |
| 21 | loginTime | 登录时间 | Date | 否 | 系统自动赋值 |
| 22 | aliveTime | 保活时间 | Date | 否 | 系统自动赋值 |
| 23 | ip | 用户IP地址 | String | 否 |  |
| 24 | systemID | 系统编号 | String | 否 |  |
| 25 | other | 其它信息 | String | 否 |  |
| 26 | userDomain | 用户在第三方系统的用户域 | Map  <String ,String> | 否 | 可用于解决一号多重身份的应用场景 |
| 27 | datas | 附加用户数据 | Map  <String ,String> | 否 |  |

#### 3.5.3 接口举例-请求报文

https://127.0.0.1/msXSSO/xsso/getLoginUser? token=41CCE7E3C0FA498985970C021179CEF2

#### 3.5.4 接口举例-响应报文

{

    "code": "200",

    "message": "成功",

    "data": {

        "data": {

            "appKey": "D22DCADC06BB4EAE868D5728AACAD370",

            "userId": "9527",

            "userNo": "工号",

            "userCode": "编码",

            "openID": "微信用户应用ID",

            "unionID": "微信用户统一ID",

            "mobile": "手机号",

            "phone": "手机号",

            "email": "邮箱",

            "userName": "用户名",

            "nickname": "昵称",

            "userType": "用户类型",

            "userLevel": "用户级别",

            "userSource": "用户来源",

            "loginAccount": "登陆账号",

            "orgCode": "组织编码",

            "orgName": "组织名称",

            "sex": "用户性别",

            "sessionID": "会话ID",

"systemID": "系统编号",

"other": "其它信息",

            "datas": {

                "附加用户数据1": "附加用户数据1",

                "附加用户数据2": "附加用户数据2"

            }

        }

    }

}

### 3.7 注销登录用户

**使用对象：**应用服务器

#### 3.7.1 接口请求参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **名称** | 注销登录用户 | | | | |
| **说明** | 用户退出登录 | | | | |
| **地址** | <https://127.0.0.1:80/>服务名称/xsso/logoutUser | | | | |
| **方式** | GET请求 | | | | |
| **接口请求参数** | | | | | |
| **序号** | **参数名称** | **参数说明** | **参数类型** | **是否必须** | **备注** |
| 01 | USID | 会话级的令牌 | String | 是 | 二选一，与token同义，  为兼容老接口而存在 |
| 02 | token | 会话级的令牌 | String | 是 | 二选一，与USID同义 |
| 03 | sysid | 系统编号 | String | 是 |  |

#### 3.7.2 接口响应参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **接口返回参数** | | | | | |
| **序号** | **参数名称** | **参数说明** | **参数类型** | **是否必须** | **备注** |
| 01 | code | 是否成功标识 | String | 是 | 200为成功 |
| 02 | message | 消息说明 | String | 是 |  |

#### 3.7.3 接口举例-请求报文

https://127.0.0.1/msXSSO/xsso/logoutUser? token=41CCE7E3C0FA498985970C021179CEF2

#### 3.7.4 接口举例-响应报文

{

    "code": "200",

    "message": "成功",

}

### 3.8 绑定会话关系

**使用对象：**终端用户

**使用限制：**仅限终端用户使用（如，浏览器、手机、PC），应用服务器无须调用

#### 3.8.1 接口请求参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **名称** | 绑定会话关系 | | | | |
| **说明** | 终端用户使用浏览器访问，一般在用户登录成功后执行，可通过Ajax方式回调JS脚本实现 | | | | |
| **地址** | <https://127.0.0.1:80/>服务名称/xsso/binding | | | | |
| **方式** | GET请求 | | | | |
| **接口请求参数** | | | | | |
| **序号** | **参数名称** | **参数说明** | **参数类型** | **是否必须** | **备注** |
| 01 | USID | 会话级的令牌 | String | 是 | 二选一，与token同义，  为兼容老接口而存在 |
| 02 | token | 会话级的令牌 | String | 是 | 二选一，与USID同义 |
| 03 | sysid | 系统编号 | String | 是 |  |

#### 3.8.2 接口响应参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **接口返回参数** | | | | | |
| **序号** | **参数名称** | **参数说明** | **参数类型** | **是否必须** | **备注** |
| 01 | code | 是否成功标识 | String | 是 | 200为成功 |
| 02 | message | 消息说明 | String | 是 |  |

#### 3.8.3 接口举例-请求报文

https://127.0.0.1/msXSSO/xsso/binding?token=41CCE7E3C0FA498985970C021179CEF2

#### 3.8.4 接口举例-响应报文

{

    "code": "200",

    "message": "成功",

}

### 3.9 跨系统获取会话票据

**使用对象：**终端用户

**使用限制：**仅限终端用户使用（如，浏览器、手机、PC），应用服务器无须调用

#### 3.9.1 接口请求参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **名称** | 终端用户跨系统获取会话票据 | | | | |
| **说明** | 终端用户使用浏览器访问，获取已从其它系统登录过的会话票据 | | | | |
| **地址** | <https://127.0.0.1:80/>服务名称/xsso/getUSID | | | | |
| **方式** | GET请求 | | | | |
| **接口请求参数** | | | | | |
| **序号** | **参数名称** | **参数说明** | **参数类型** | **是否必须** | **备注** |
| 01 | SSOCallBack | 回调JS函数名 | String | 是 |  |
| 02 | sysid | 系统编号 | String | 是 |  |

#### 3.9.2 接口响应参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **接口返回参数** | | | | | |
| **序号** | **参数名称** | **参数说明** | **参数类型** | **是否必须** | **备注** |
| 01 | SSOCallBack入参定义的回调函数名称 | SSOCallBack入参定义的回调函数名称 | String | 是 |  |

#### 3.9.3 接口举例-请求报文

https://127.0.0.1/msXSSO/xsso/getUSID?SSOCallBack=getUSID

#### 3.9.4 接口举例-响应报文

getUSID('45BB44A6E158444B678E48302DFB03E5');

### 3.10 会话保活

**使用对象：**应用服务器 或 终端用户

#### 3.10.1 接口请求参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **名称** | 会话保活 | | | | |
| **说明** | 会话保活，一般在用户登录成功后并在每次有效请求后执行（或累积到60秒，每分钟执行一次）。可由应用服务或终端发起均可 | | | | |
| **地址** | <https://127.0.0.1:80/>服务名称/xsso/alive | | | | |
| **方式** | GET请求 | | | | |
| **接口请求参数** | | | | | |
| **序号** | **参数名称** | **参数说明** | **参数类型** | **是否必须** | **备注** |
| 01 | USID | 会话级的令牌 | String | 否 | 二选一，与token同义，  为兼容老接口而存在  允许在终端用户上不传任何值 |
| 02 | token | 会话级的令牌 | String | 否 | 二选一，与USID同义  允许在终端用户上不传任何值 |
| 03 | sysid | 系统编号 | String | 是 |  |

#### 3.10.2 接口响应参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **接口返回参数** | | | | | |
| **序号** | **参数名称** | **参数说明** | **参数类型** | **是否必须** | **备注** |
| 01 | code | 是否成功标识 | String | 是 | 200为成功 |
| 02 | message | 消息说明 | String | 是 |  |

#### 3.10.3 接口举例-请求报文

https://127.0.0.1/msXSSO/xsso/alive?token=45BB44A6E158444B678E48302DFB03E5

#### 3.10.4 接口举例-响应报文

{

    "code": "200",

    "message": "成功",

}

### 3.11 在线用户简报

**使用对象：**应用服务器 或 终端用户

#### 3.11.1 接口请求参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **名称** | 在线用户简报 | | | | |
| **说明** | 统计在线用户数量。并能控制统计多长时间内活动的在线用户数量 | | | | |
| **地址** | <https://127.0.0.1:80/>服务名称/xsso/report | | | | |
| **方式** | GET请求 | | | | |
| **接口请求参数** | | | | | |
| **序号** | **参数名称** | **参数说明** | **参数类型** | **是否必须** | **备注** |
| 01 | onlineMaxTimeLen | 在线最大时长 | Integer | 否 | 按活动时间计算，取多少时间范围内的在线用户（单位：秒） |
| 02 | accessToken | 获取访问Token | String | 是 | 见 3.2 接口 |
| 03 | sysid | 系统编号 | String | 是 |  |

#### 3.11.2 接口响应参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **接口返回参数** | | | | | |
| **序号** | **参数名称** | **参数说明** | **参数类型** | **是否必须** | **备注** |
| 01 | code | 是否成功标识 | String | 是 | 200为成功 |
| 02 | message | 消息说明 | String | 是 |  |
| **Data对象** | | | | | |
| 01 | userType | 用户类型 | String | 是 | 见 3.4 接口中的用户类型 |
| 02 | onlineUserCount | 在线用户数据 | Integer | 是 | 本用户类型下的在线用户数据 |
| 03 | onlineUserAllCount | 所有在线用户数 | Integer | 是 |  |

#### 3.11.3 接口举例-请求报文

https://127.0.0.1/msXSSO/xsso/report?token=45BB44A6E158444B678E48302DFB03E5

#### 3.11.4 接口举例-响应报文

{

    "code": "200",

    "message": "成功",

    "data": {

        "data": [

            {

                "userType": "-",

                "onlineUserCount": 50,

                "onlineUserAllCount": 53

            },

            {

                "userType": "industry",

                "onlineUserCount": 3,

                "onlineUserAllCount": 53

            }

        ]

    }

}