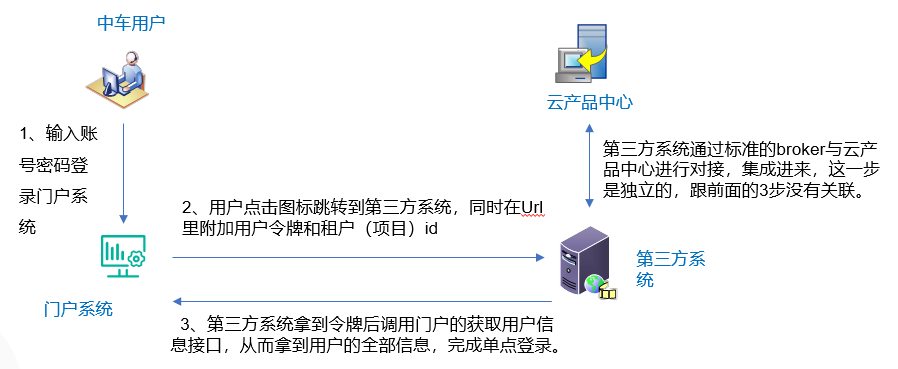
# 一、单点登录的流程及事例



跳转示例：http://ip:port/index?sessionId=aegaegtatetaetettaettaetwt&projectId=fadsf4d25

sessionId:当前登录用户令牌

projectId：项目id或租户id

门户获取用户信息接口:

<http://ip:port/indnet/portalUser/getUserPermit> post请求

参数为空，注意要把获取到的令牌放到请求的Header里。

格式为：

Authorization:令牌的值

调用成功后，门户返回的格式如下：

{

"state": true,

"message": "",

"value": {

"userId": "用户id",

"account": "用户帐号",

"userName": "用户名",

"email": "邮箱",

"orgId": "公司id",

"orgName": "\*\*公司名",

"deptId": "部门id",

"deptName": "部门名",

"roles": [

{

"id": "角色id",

"name": "角色名",

"code": "角色编码",

"enabled": "是否可用0-不可用，1可用",

"isCommon": "是否公共角色0-否1是",

"orgId": "所属组织"

}

],

"roleCodes": "所有角色编码",

"roleName": "所有角色名",

"menuCheck": [

"菜单名1",

"菜单名1"

],

"authorization": "当前token",

"userKey": "加密key",

"userCheckKey": "加密key",

//用户所属租户信息 数组类型

"tenants": [

{

                "id": "租户id",

                "name": "租户名",

                "description": "隆轩项目D",

                "orgId": "所属组织id",

                "status": 1, // 是否启用0-否；1-是

                 //用户所属租户下的项目信息 数组类型

"projects": [

                    {

                        "id": "项目id",

                        "name": "项目名",

                        "code": "项目编码",

                        "status": 1, // 是否启用0-否；1-是

                        "userProRoles": [用户在此项目的角色编码],

                    }

]

}

]

}

}

# 二、门户提供的可供调用的API接口

## 请求方式

门户提供接口，供其他系统调用，调用方式为POST

## 查询验证用户是否在IAM存在

请求url：http(s)://ip:port/user/checkUserIam

其中ip和port为门户的ip或域名，port为端口号

请求头：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数编码 | 参数值 | 是否必须 | 说明 |
| Authorization | Token | 是 | 门户登录时生成的token值 |
| Content-Type | application/json | 否 |  |

请求参数

|  |  |  |  |
| --- | --- | --- | --- |
| 参数编码 | 参数值 | 是否必须 | 说明 |
| userId | String | 是 | 用户编码 |
| name | String | 是 | 用户名 |

响应编码：200

响应报文代表：如果存在则返回用户信息，如果为空则不存在

响应编码非200值，data会有对应错误信息

## 用户信息查询

用户登录完门户后会生成tokenId，当用户打开第三方平台的时候，把tokenId放到请求的header参数传到paas平台，会视提供通过tokenId查询用户信息的接口，该接口返回用户信息、权限信息、租户信息、组织机构信息等。

请求url：http(s)://ip:port/user/getUserInfo

其中ip和port为门户的ip或域名，port为端口号

请求头：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数编码 | 参数值 | 是否必须 | 说明 |
| Authorization | Token | 是 | 门户登录时生成的token值 |
| Content-Type | application/json | 否 |  |

请求参数：无

响应编码：200

响应报文：

{

"state": true,

"message": "",

"value": {

"userId": "用户id",

"account": "用户帐号",

"userName": "用户名",

"email": "邮箱",

"orgId": "公司id",

"orgName": "\*\*公司名",

"deptId": "部门id",

"deptName": "部门名",

"roles": [

{

"id": "角色id",

"name": "角色名",

"code": "角色编码",

"enabled": "是否可用0-不可用，1可用",

"isCommon": "是否公共角色0-否1是",

"orgId": "所属组织"

}

],

"roleCodes": "所有角色编码",

"roleName": "所有角色名",

"menuCheck": [

"菜单名1",

"菜单名1"

],

"authorization": "当前token",

"userKey": "加密key",

"userCheckKey": "加密key",

//用户所属租户信息 数组类型

"tenants": [

{

                "id": "租户id",

                "name": "租户名",

                "description": "隆轩项目D",

                "orgId": "所属组织id",

                "status": 1, // 是否启用0-否；1-是

                 //用户所属租户下的项目信息 数组类型

"projects": [

                    {

                        "id": "项目id",

                        "name": "项目名",

                        "code": "项目编码",

                        "status": 1, // 是否启用0-否；1-是

                        "userProRoles": [用户在此项目的角色编码],

                    }

]

}

]

}

}

## 消息中心接收消息接口

请求url：http(s)://ip:port/user/messageInfo/

其中ip和port为门户的ip或域名，port为端口号

请求头：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数编码 | 参数值 | 是否必须 | 说明 |
| Authorization | Token | 是 | 门户登录时生成的token值 |
| Content-Type | application/json | 否 |  |

请求参数

|  |  |  |  |
| --- | --- | --- | --- |
| 参数编码 | 参数值 | 是否必须 | 说明 |
| infoType | String | 是 | 消息类型 |
| infoResource | String | 是 | 消息来源 |
| infoTitle | String | 是 | 消息标题 |
| data | String | 是 | 消息体 |
| userId | List<string> | 是 | 接收消息的用户 |

响应编码：200

响应报文：

{

code:200

message:””

}

响应编码非200值，message会有对应错误信息

# 三、门户的同步信息接口

## 请求方式

第三方系统需要提供一个信息接收接口url，门户会在相关信息发生变更的时候调用配置好的第三方接口url，并把相关的信息作为入参传递过去，请求方式为POST。

响应统一为：

响应编码：200

响应报文：

{

code:200

message:””

}

## 请求参数

请求头参数

|  |  |  |
| --- | --- | --- |
| 参数编码 | 参数值 | 说明 |
| Authorization | Basic secretKey | 会视门户登录时生成的token值 |
| Content-Type | application/json |  |

请求参数

|  |  |  |
| --- | --- | --- |
| 参数编码 | 参数值 | 说明 |
| Flag | String | 11:项目增加  12:项目删除  13:项目修改  21:菜单权限增加或修改  22: 菜单权限删除  31:组织机构增加  32:组织机构删除  33:组织机构修改  41:角色增加  42:角色删除  43:角色修改  51:用户信息增加  52:用户信息删除  53:用户信息修改  61:租户信息增加  62:租户信息删除  63:租户信息修改  71:子公司信息增加或修改  72:子公司信息删除  511:用户和角色关系的增加或修改  512:用户和角色关系的删除  521:用户和组织关系的增加或修改  522:用户和组织关系的删除  531:用户和项目关系的增加  532:用户和项目关系的删除  533:用户和项目关系的修改 |
| Data | Object | data数据部分根据flag值确定 |

### 4.5操作租户下的用户对象

#### 新增租户下的用户信息

当flag=51时传入userinfo对象参数，新增用户信息

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 类型 | 是否必须 | 说明 |
| Username | String | 是 | 用户姓名 |
| userAccount | String | 是 | 用户账号 |
| sexId | Long | 否 | 性别 1:男，2:女 |
| loginId | String | 是 | 登录id，唯一id，由由会视生成 |
| Password | String | 否 | 密码 |
| Email | String | 否 | 邮箱 |
| Mobile | String | 否 | 手机 |
| Memo | String | 否 | 备注 |
| isAdmin | Int | 否 | 是否管理员，1:是/0:否 |
| isNormal | Int | 否 | 是否正常，1:是/0:否 |
| updateDate | String | 否 | 更新日期,格式yyyyMMddHHmmss,如2023年2月9日13：30:00则为20230209133000 |
| updateUserId | String | 否 | 更新人，该字段来自用户唯一标识 |
| createDate | String | 否 | 创建日期,格式yyyyMMddHHmmss,如2023年2月9日13：30:00则为20230209133000 |
| createUserId | String | 否 | 创建人，该字段来自用户唯一标识 |
| pwdExpireDate | String | 否 | 密码过期时间,格式yyyyMMddHHmmss,如2023年2月9日13：30:00则为20231209133000 |
| imgUrl | String | 否 | 用户头像 |
| reserve3 | String | 否 | 密码加密盐值 |
| tenantId | String | 否 | 租户id，关联4.6，不传的话，就默认系统tenant\_system |

入参示例：

1. {
2. "flag": "51",
3. "data": [{
4. "userName":"123",
5. “userAccount”:”abc”
6. "sexId":1,
7. "loginId": "用户唯一id",
8. "password":"123",
9. "email":"bdev@172.com",
10. "mobile":"16601330458",
11. "memo":"备注",
12. "isAdmin":0,
13. "isNormal":1,
14. "updateDate":"20230209133000",
15. "updateUserId":"456789",
16. "createDate":"20230209133000",
17. "createUserId":"123456",
18. "pwdExpireDate":"20231209133000",
19. "imgUrl":"/root/img/test.jpg",
20. "reserve3":"32",
21. "tenantId":"123456789"
22. },
23. {
24. "userName":"123",
25. "sexId":1,
26. "loginId": "用户唯一id",
27. "password":"123",
28. "email":"bdev@172.com",
29. "mobile":"16601330458",
30. "memo":"备注",
31. "isAdmin":0,
32. "isNormal":1,
33. "updateDate":"20230209133000",
34. "updateUserId":"456789",
35. "createDate":"20230209133000",
36. "createUserId":"123456",
37. "pwdExpireDate":"20231209133000",
38. "imgUrl":"/root/img/test.jpg",
39. "reserve3":"32",
40. "tenantId":"123456789"
41. }
42. ]
43. }

#### 删除租户下的用户信息

当flag= 52时传入userinfo对象参数，删除用户信息

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 类型 | 是否必须 | 说明 |
| loginId | String | 是 | 登录id，唯一id，由由会视生成 |
| tenantId | String | 是 | 租户ID |
| userAccount | String | 是 | 用户账号 |
| updateDate | String | 否 | 更新时间，格式yyyyMMddHHmmss,如2023年2月9日13:30:00则为20230209133000 |
| updateUserId | String | 否 | 更新人登录id,该字段来自用户唯一标识 |

入参示例：

1. {
2. "flag": "52",
3. "data": [
4. {
5. "loginId": "用户唯一id",
6. "updateDate":"20230209133000",
7. "updateUserId":"456789"
8. },
9. {
10. "loginId": "用户唯一id",
11. "updateDate":"20230209133000",
12. "updateUserId":"456789"
13. }]
14. }

#### 修改租户下的用户信息

当flag=53时传入userinfo对象参数，修改用户信息

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 类型 | 是否必须 | 说明 |
| userAccount | String | 是 | 用户账号 |
| Username | String | 是 | 用户姓名 |
| sexId | Long | 否 | 性别 1:男，2:女 |
| loginId | String | 否 | 登录id，唯一id，由由会视生成 |
| Password | String | 否 | 密码 |
| Email | String | 否 | 邮箱 |
| Mobile | String | 否 | 手机 |
| Memo | String | 否 | 备注 |
| isAdmin | Int | 否 | 是否管理员，1:是/0:否 |
| isNormal | Int | 否 | 是否正常，1:是/0:否 |
| updateDate | String | 否 | 更新日期,格式yyyyMMddHHmmss,如2023年2月9日13：30:00则为20230209133000 |
| updateUserId | String | 否 | 更新人，该字段来自用户唯一标识 |
| createDate | String | 否 | 创建日期,格式yyyyMMddHHmmss,如2023年2月9日13：30:00则为20230209133000 |
| createUserId | String | 否 | 创建人，该字段来自用户唯一标识 |
| pwdExpireDate | String | 否 | 密码过期时间,格式yyyyMMddHHmmss,如2023年2月9日13：30:00则为20231209133000 |
| imgUrl | String | 否 | 用户头像 |
| reserve3 | String | 否 | 密码加密盐值 |
| tenantId | String | 否 | 租户id，关联4.6 |

入参示例：

{

"flag": "53",

"data": {

"userName":"123",

"sexId":1,

"loginId": "用户唯一id",

"password":"123",

"email":"[bdev@172.com](mailto:bdev@172.com)",

"mobile":"16601330458",

"memo":"备注",

"isAdmin":0,

"isNormal":1,

"updateDate":"20230209133000",

"updateUserId":"456789",

"createDate":"20230209133000",

"createUserId":"123456",

"pwdExpireDate":"20231209133000",

"imgUrl":"/root/img/test.jpg",

"reserve3":"32",

"tenantId":"123456789"

}

}

### 4.6操作租户对象

#### 新增租户信息

当flag=61时增加租户时data对象

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 类型 | 是否必须 | 说明 |
| tenantId | String | 是 | 该用户所属租户id |
| tenantCode | String | 是 | 租户编码（可以和tenantId一样） |
| Remark | String | 否 | 备注 |
| status | Int | 是 | 0停用 1启用 |
| tenantName | String | 是 | 租户名字 |
| isSystem | Int | 否 | 是否系统1:是/0:否 |
| groupCode | String | 是 | 子公司编码code，关联4.7 (对应租户的Org\_id) |
| createTime | String | 否 | 创建时间，格式yyyyMMddHHmmss,如2023年2月9日13:30:00则为20230209133000 |
| createUserId | String | 否 | 创建人登录id,该字段来自用户唯一标识 |
| updateTime | String | 否 | 更新时间，格式yyyyMMddHHmmss,如2023年2月9日13:30:00则为20230209133000 |
| updateUserId | String | 否 | 更新人登录id,该字段来自用户唯一标识 |
| tenantManageId | String | 是 | 租户管理员id,该字段来自用户唯一标识 |

入参示例：

{

"flag": "61",

"data": {

"tenantId": "123456789",

"tenantCode": "123456789",

"remark": "备注",

"tenantName": "租户名字",

"isSystem":0,

"groupCode":"456"，

"createTime":"20230214120000",

"createUserId":"123",

"updateTime":"20230214142020",

"updateUserId":"456",

"tenantManageId":"123456789"

}

}

1. 测试参数
2. {
3. "flag": "61",
4. "data": {
5. "tenantId": "1547409848181821440",
6. "tenantCode": "1547409848181821440",
7. "remark": "备注",
8. "tenantName": "租户名字测试1",
9. "isSystem":0,
10. "groupCode":"test1",
11. "createTime":"20230214120000",
12. "createUserId":"1547409851726008320",
13. "updateTime":"20230214142020",
14. "updateUserId":"1547409851726008320"
15. }
16. }

#### 删除租户信息

当flag=62时删除租户时data对象

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 类型 | 是否必须 | 说明 |
| tenantId | String | 是 | 该用户所属租户id |
| updateTime | String | 否 | 更新时间，格式yyyyMMddHHmmss,如2023年2月9日13:30:00则为20230209133000 |
| updateUserId | String | 否 | 更新人登录id,该字段来自用户唯一标识 |

入参示例：

{

"flag": "62",

"data": {

"tenantId": "123456789",

"updateTime":"20230214142020",

"updateUserId":"456"

}

}

#### 修改租户信息

当flag=63时修改租户时data对象

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 类型 | 是否必须 | 说明 |
| tenantId | String | 是 | 该用户所属租户id |
| tenantCode | String | 否 | 租户编码（可以和tenantId一样） |
| Remark | String | 否 | 备注 |
| status | Int | 是 | 0停用 1启用 |
| tenantName | String | 否 | 租户名字 |
| isSystem | Int | 否 | 是否系统1:是/0:否 |
| groupCode | String | 否 | 子公司编码code，关联4.7 |
| createTime | String | 否 | 创建时间，格式yyyyMMddHHmmss,如2023年2月9日13:30:00则为20230209133000 |
| createUserId | String | 否 | 创建人登录id,该字段来自用户唯一标识 |
| updateTime | String | 否 | 更新时间，格式yyyyMMddHHmmss,如2023年2月9日13:30:00则为20230209133000 |
| updateUserId | String | 否 | 更新人登录id,该字段来自用户唯一标识 |
| tenantManageId | String | 否 | 租户管理员id,该字段来自用户唯一标识 |

入参示例：

{

"flag": "63",

"data": {

"tenantId": "123456789",

"tenantCode": "123456789",

"remark": "备注",

"tenantName": "租户名字",

"isSystem":0,

"groupCode":"456"，

"createTime":"20230214120000",

"createUserId":"123",

"updateTime":"20230214142020",

"updateUserId":"456",

"tenantManageId":"123456789"

}

}