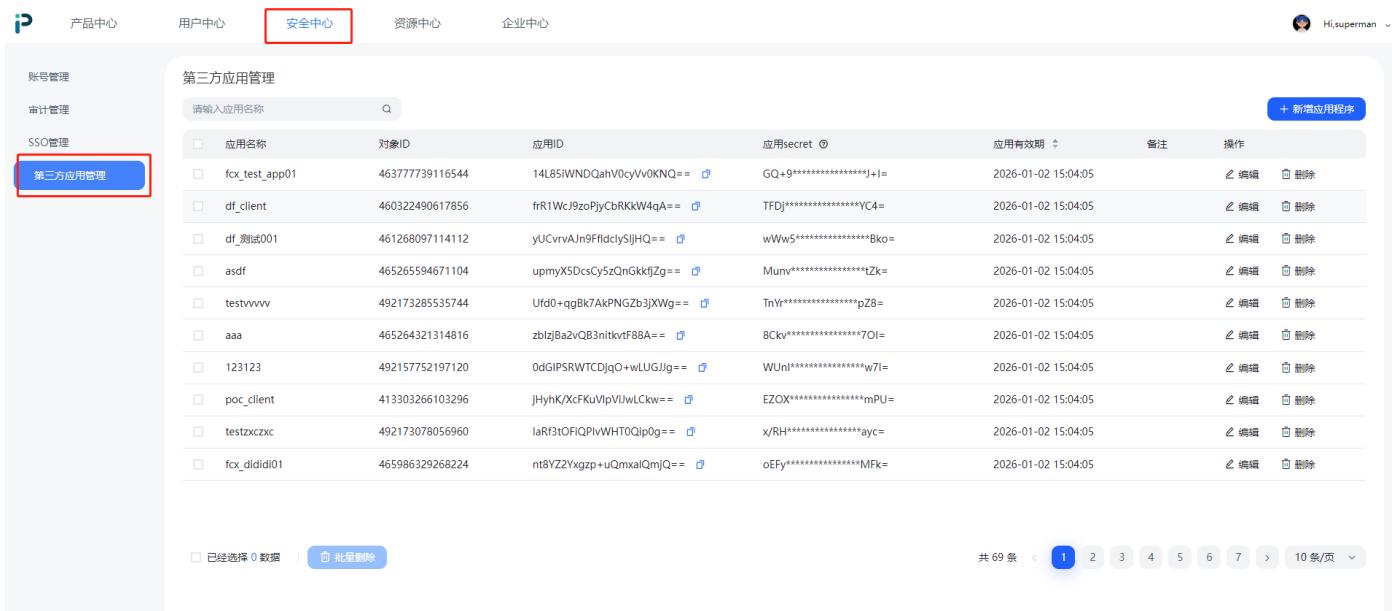


# 1、统一 OAuth 流程

1. 登入统一平台，安全中心创建第三方应用管理，获取应用ID及应用secret。用于后续获取token接口



应用名称	对象ID	应用ID	应用secret	应用有效期	备注	操作
fcx_test_app01	463777739116544	14L85iVNDQahV0cyVv0KNQ==	GG+9*****j+ =	2026-01-02 15:04:05		<a href="#">编辑</a> <a href="#">删除</a>
df_client	460322490617856	fr81WcJ9zoPjyCbRKkW4qA==	TFDj*****VC4=	2026-01-02 15:04:05		<a href="#">编辑</a> <a href="#">删除</a>
df_测试001	461268097114112	yUCvrvAJn9FfdclySijHQ==	wWw5*****Bko=	2026-01-02 15:04:05		<a href="#">编辑</a> <a href="#">删除</a>
asdf	465265594671104	upmyX5DcsCy5zQnGkkfjZg==	Munv*****tZk=	2026-01-02 15:04:05		<a href="#">编辑</a> <a href="#">删除</a>
testvvvv	492173285535744	Ufd0+qgBk7AkPNGZb3jXWg==	TnY*****pZB=	2026-01-02 15:04:05		<a href="#">编辑</a> <a href="#">删除</a>
aaa	465264321314816	zblzjBa2vQB3nitkvtf88A==	8Cky*****7O =	2026-01-02 15:04:05		<a href="#">编辑</a> <a href="#">删除</a>
123123	492157752197120	0dGIPSRWTCDjqO+wLUGJg==	WUnl*****w7 =	2026-01-02 15:04:05		<a href="#">编辑</a> <a href="#">删除</a>
poc_client	413303266103296	JhYhK/XcFKuVlpVijwLCkw==	EZOX*****mPU=	2026-01-02 15:04:05		<a href="#">编辑</a> <a href="#">删除</a>
testzxczxc	492173076056960	laRf3tOFQPlvWHT0Qip0g==	x/RH*****8yc=	2026-01-02 15:04:05		<a href="#">编辑</a> <a href="#">删除</a>
fcx_didid01	465986329268224	nt8YZ2Yxgzp+uQmxalQm Q==	oEfY*****Mfk=	2026-01-02 15:04:05		<a href="#">编辑</a> <a href="#">删除</a>

## 1.1 获取授权码

### 基本信息

**Path:** /api/dp/auth/oauth/authorize

**Method:** GET

### 请求参数

#### Query

参数名称	是否必须	示例	备注
client_id	是	jHyhK/XcFKuVlpVijwLCkw==	clientID 为应用注册的id
response_type	是	code	
scope	是	all或default	

### 请求示例

```
1 curl --request GET \
2   --url 'https://jkstack.dev.jingkunsystem.com:29041/api/dp/auth/oauth/authorize?
3   response_type=code&client_id=CfjisL9AeNc066tkxSiKig==&scope=default' \
4   --header 'content-type: multipart/form-data'
```

## 返回参数

```
1  {
2      "code": 200,
3      "message": "",
4      "data": {
5          "code": "N2I1OTEXMTYTYTA2ZS0ZZJVHLWJLMTKMMNIMDFMMJFMYJNM"
6      }
7  }
```

## 1.2 获取访问令牌

### 基本信息

**Path:** /api/dp/auth/oauth/token

**Method:** POST

### 请求参数

#### Body

参数名称	参数类型	是否必须	示例	备注
code	String	是	NMYXNZDLNjYTZDJjNI0ZNZCYLWFjMjMTYZUZYDMXYjHKZTNH	
client_id	String	是	jHyhK/XcFKuVlpVijwLCkw==	
client_secret	String	是	EZOXzdyqbGt1CpxtHrt5te35zS9iE6jfcgyvpPpNmPU=	
grant_type	String	是	authorization_code	此模式为authorization_code
user_id	String	否	1	根据用户的unqieID

### 请求示例

```
1  {
2      "code": "MDZMZTCZZJITMGNMNY0ZNTDLLTHMNMMTNJDMZTEYMGJJYMQW",
3      "client_id": "CfjisL9AeNc066tkxSiKig==",
4      "client_secret": "xB5hQnmlNnc+iab3RODs6yHlcHqFHAM2e4PawddAxtI=",
5      "grant_type": "authorization_code"
6  }
```

### 返回示例

```
1  {
2      "code": 200,
3      "message": "",
4      "data": {
5          "token_type": "Bearer",
6          "access_token":
7              "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJhdWQiOiJUaGUGQVBQIHRoYXQgY2FuIGJ1IGFjY2Vzc2
8              VkIGJhc2VkJG9uIHRoZSB0b2tlbiIsImV4cCI6MTY0ODY0Mjc3NSwiaXNzIjoiVGh1IGlzc3VpbmcgYXV0aG
9              9yaxR5IG9mIHRoZSB0b2tlbiIsImp0aSI6IlRoZSB1bmlxdWVseSBpZGVudGlmaWVzIiwib3JpZ19pYXQiOj
E2ODg2MjQ3NzUsInNjb3BlIjoiVGh1IHNjb3BlIG9mIHRoZSBUEkgdGhhCBjYW4gYmUgYWNjZXNzzWQgYm
FZZWQgb24gdGh1IHRva2VuIiwig3ViIjolMDgwNDUxNjcyNTE0NTZ9.NY6sVKBF7D5mWFMsWw7bc-
Sp80cslrrH3nCXrKK1y4"
8      }
8  }
```

## 2、获取预案列表

请求 URL:

- 完整url 参考调试地址
- 路径: /api/dpa/process/processPlan/queryPlanByUserId

请求方式:

- GET

请求头:

- Authorization: Bearer 请求token

参数: 无

返回示例

```
1  {
2      "message": "success",
3      "code": 200,
4      "data": [
5          {
6              "processPlanId": "acf42dd3",
7              "title": "测试预案",
8              "status": "DEFAULT",
9              "stepNums": 1,
10             "version": 0,
11             "createTime": 1626944836628,
12             "createUser": "U100000",
13             "updateTime": 1626944836628,
14             "updateUser": ""
```

```
15     }
16 ]
17 }
```

#### 返回参数说明

参数名	类型	说明
code	int	返回code值, 200:成功, 997:失败
message	String	描述
data	List	返回数据
— name	String	字段名
— type	String	类型

## 3、根据预案 ID 查询工作流执行时输入

请求方式:

- POST

请求URL:

- 完整url 参考调试地址
- 路径: /api/dpa/process/processPlan/queryModelByPlanId

请求头:

- Content-Type: application/json
- Authorization: Bearer 请求token

请求示例:

```
1 {
2     "processPlanId": "预案ID"
3 }
```

返回示例:

```
1 {
2     "code": 200,
3     "message": "success",
4     "data": [
5         {
6             "modelName": "工作流名称",
7             "processModelId": "工作流 ID",
8         }
9     ]
10 }
```

```

8     "mapParam": [
9         {
10            "name": "DPA_MODEL_WORKDIR",
11            "type": "running"
12        }
13    ]
14 }
15 ]
16 }

```

## 4、预案定时场景执行

请求 URL:

- 完整url 参考调试地址
- 路径: /api/dpa/process/processPlan/saveTimingParam

请求方式:

- POST

请求头:

- Content-Type: application/x-www-form-urlencoded
- Authorization: Bearer 请求token

参数:

参数名	必选	类型	说明
processPlanId	是	String	预案 ID
callbackUrl	是	String	需要回调的URL
dimension	是	String	周期维度 (seconds:秒, minutes:分, hours:小时, day:日, weeks: 周, month:月)
intervals	是	String	周期间隔
executeStrategy	是	String	执行策略:1.single (单次时间) 2.cycle(周期)
sceneName	是	String	场景名称
modelVarReq	否	ArrayList	执行时定义的参数对象,如有设置必填
--modelId	否	String	工作流 ID
--mapParam	否	Object	执行参数——包含字段参考接口 3 返回数据

返回示例

```
1  {
2      "message": "success",
3      "code": 200,
4      "data": {
5          "instanceId": "in1606749734084",
6          "modelId": "md1606743910508",
7          "status": "FAIL"
8      }
9 }
```

## 5、预案手动触发执行

请求 URL:

- 完整url 参考调试地址
- 路径: /api/dpa/process//processPlan/opExecutePlanByApi/\${预案 ID}

请求方式:

- POST

请求头:

- Content-Type: application/json
- Authorization: Bearer 请求token

参数:

参数名	必选	类型	说明
processPlanId	是	String	预案 ID
callbackUrl	否	String	需要回调的 URL—回调规则--参照接口 7
modelVarReq	否	ArrayList	执行时定义的参数对象,如有设置必填
-- modelId	否	String	工作流 ID
-- mapParam	否	Object	执行参数——包含字段参考接口 3 返回数据

返回示例

```
1  {
2      "code":200,
3      "message":"success",
4      "data":[
5          {
6              "processPlanId":"acf42dd3",
7              "planInstanceId":"in10000001"
8          }
9      ]
10 }
```

返回参数说明

```
1 | processPlanId: 预案 ID
2 | planInstanceId: 预案执行 ID(每次执行都不一样, 可做唯一标识)
```

## 6、预案查询执行出参 (api,shell 节点)

请求 URL:

- 完整url 参考调试地址
- 路径: /api/dpa/process//processPlan/queryPlanRunOutParam

请求方式:

- GET

请求头:

- Authorization: Bearer 请求token

请求参数:

参数名	必选	类型	说明
processPlanId	是	String	预案 ID
planInstanceId	是	String	预案执行 ID 回调返回或执行时会返回

返回示例

```
1  {
2      "code":200,
3      "message":"success",
4      "data":[
5          {
6              "modelName":"工作流名称",
7              "processModelId":"工作流 ID",
```

```

8     "mapParam": [
9         {
10            "result": "节点出参值",
11            "nodeName": "节点名称"
12        }
13    ]
14 }
15 ]
16 }

```

## 7、回调接口定义

请求方式：

- POST

请求头：

- Content-Type: application/json

发送的请求字段：

参数名	必选	类型	说明
planInstanceId	是	String	任务实例 ID
processPlanId	是	String	预案 ID
status	是	String	状态:DEFAULT 默认 RUNNING 执行中 PENDING 挂起 SUCCESS 成功 FAIL 失败

返回示例

```
1 | SUCCESS
```

备注: 如果接收到请求, 返回字符串 SUCESS 表示接收成功 否则我们会重新回调, 回调重试 3 次。处理过的逻辑可以不用在处理直接返回SUCCESS, 可用 instanceId 做校验

**响应状态码说明：**

状态码	说明
200	OK
400	参数校验异常
403	身份验证失败
404	请求路径异常
500	服务器内部错误
503	服务器异常未发现
997	未知错误
998	用户不存在
999	业务逻辑异常