

# EZVIZ 设备托管 产品文档

# 乐享安全智能生活 V1.5



# 目录

1.	产品	简介	2
1	.1	什么是设备托管	2
1	.2	名词解释	2
2.	产品	定价	2
2	.1	计费方式	2
2	.2	保证金	3
3.	快速	这门	3
3	.1	使用流程	3
3	.2	步骤详解	1
	3.2.1	开通设备托管	1
	3.2.2	9.开发者配置授权页	5
	3.2.3	3.用户登录、提交授权设备信息	5
	3.2.4	1 开发者获取授权信息	7
	3.2.5	5 开发者以用获取授权信息	3
附 <b>:</b>	设备	托管协议文档	9
1.	开发	者 H5 授权登录页地址	9
2.	平台	集成开发者回调地址逻辑10	)
3.	相关	接口说明12	2
3	.1	开发者通过授权码获取授权信息(确认设备授权)12	2
3	.2	刷新设备托管 токеn 接口14	1
3	.3	通过开发者 APPKEY 和 APPSECRET 获取设备托管 ALL 权限 TOKEN16	5
3	.4	获取设备托管列表17	7
3	.5	开发者取消设备托管19	9
3	.6	其余支持接口	כ

# 1.产品简介

### 1.1 什么是设备托管

C用户可以通过设备托管将设备的权限(预览、回放、对讲等)授权给开发者用户,用于设备的二次开发。在保障设备所属于 C 用户的同时,满足了开发者用户同步进行二次开发的需求。



有了设备托管功能,开发者用户不仅可以同时进行多账户设备二次开发,而且可以保留 C用户使用萤石 APP 的全部能力,降低成本;对于 C 用户不但可以掌握设备所属权、拥有 设备的全部使用权限,更可以自主控制授权权限信息,保障隐私安全。举个例子,零售店铺 管理系统,设备所属于各个门店的 C 账号,管理系统需要通过二次开发集成视频相关功能 进行管理者观看。

### 1.2 名词解释

C端用户:需要使用开发者服务,或者需要将设备托管/分享/给予给另一个平台的用户 授权码回调:设备授权会发起回调-通过开发者提供的回调地址,以回调地址+参数的形式将 用户的设备授权信息传给开发者服务器

# 2.产品定价

### 2.1 计费方式

托管服务开通后,通过开发者账户进行二次开发,开发者账户所需交付的费用为:

#### 开发者账号总费用=账号企业版套餐费用+托管服务费

其中,企业版付费的设备接入数包括,账号下原先计费数+托管设备数;带宽量包括账号下设备产生带宽+托管设备通过 开发者账号取流产生带宽。托管服务费=0.1元/通道/月。

# 2.2 保证金

被授权设备数(通道)	保证金(元)
50	无需保证金
1000	3万
2000	5 万
5000	10 万
10000	15 万

# 3.快速入门

3.1 使用流程



### 3.2 步骤详解

### 3.2.1 开通设备托管

- Appkey: 开发者账户的 appkey。每个 appkey 对应一个回调地址;
- 回调地址:地址协议形式为HTTPS/HTTP,同时整个请求地址由开发者自定义.开放平台 会通过开发者的回调地址,将授权码以参数形式给到开发者,参数名为 auth\_code.开发者 只需要通过 auth\_code 来获取需要的授权信息;
- **设备托管能力集:**(您可以在以下表格中选择开发者账户需要获取的托管权限)

权限	权限解释
Get	获取设备名称序列号在线状态
Capture	抓图
Real	预览设备实时画面
Replay	预览设备本地录像
Ptz	云台控制设备
Alarm	获取设备告警消息(ys.alarm)
CloudStorage	获取云存储录像
Talk	对讲
ISAPI	海康设备isapi下行指令

\*默认支持获取设备在线状态及获取设备能力集的权限

### 3.2.2 开发者配置授权页

#### 3.2.2.1 页面配置



图 1-2 授权页面(界面由萤石开放平台提供)

图 1-2 中红色框部分可通过开发者服务账号及应用信息配置同步,包括 logo、应用名、公司 名以及服务介绍等。

路径分别为:

- 萤石开放平台官网→开发者服务→<u>我的应用</u>(同步应用服务介绍、应用名及 logo)
- 萤石开放平台官网**●**开发者服务**●**<u>账号信息</u>(同步企业名)

#### 3.2.2.2 路径配置

```
授权登录页地址:
```

https://openauth.ys7.com/trust/device?client\_id=69b11dbe62154353876ea92c3375a50a&response

\_type=code&state=test&access\_token=AAAAAA

参数说明:

client_id	开发者的 appKey			
response_type 请求类型,默认类型为 code				
state	开发者自定义信息,会通过回调接口返回给开发者(只允许字母或数字字符,			
	不允许特殊字符)			
access_token	可通过萤石通行证获取用户授权的 access_token, 若未输入该参数或参数错误			
	将需要用户重新登录;针对该 token 的获取方式,请详见萤石通行证接入文档			

\*可以将拼接过参数的 url 生成二维码或某注册页面、账户配置页面的链接;

State: 用于最终 C 端用户授权行为发生时,回调给开发者服务端,所以可以用 state
 区分业务类别、用户身份

例:

物业管理平台,自行提供家庭登记页面,当最终用户填写了家庭信息后,前往 state=家庭信息标识的授权页,

将设备授权给开发者,这样开发者收到授权设备后,可以将这些设备标记为这个家庭

 Access\_token:如果 access\_token 未传,或者说该参数过期和异常情况下,会出现如下 页面;需要用户额外再登录授权,来保障安全性。



#### 3.2.3 用户登录、提交授权设备信息



图 1-3 授权页面(界面由萤石开放平台提供)

- 用户通过萤石通行证进入账号授权页
- 在同意萤石账户用户授权须知后,登录自己的萤石账号
- 在成功拿到用户授权的 token 后,就可以直接唤起托管页面
- 确认授权后,可以查看到自己萤石账号下所有的设备,在'我的设备'标签中,选择 想要授权的设备,然后点击提交,开发者会通过被萤石服务器回调而收到授权码和授 权信息。

选择授权设备时,可以同步对每台设备配置C用户自定义设备信息,用于开发者后续
 获取托管设备列表时,参考备注信息对设备进行对应处置

### 3.2.4 开发者获取授权信息

开发者通过向萤石云平台提供回调接口后,用户进行授权操作时,平台会通过以下接口 给到开发者授权信息:

- 授权码回调: Url:{callBackAddress}?auth\_code=value&state=customInfo
- 授权信息(设备)回调:

Url:{callBackAddress}?opt\_type=opt&deviceSerials=deviceSerials&deviceTrustId=deviceTr ustId&state=customInfo

- 在授权过程中开放平台会先将 auth\_code 以及 state 给开发者,, auth\_code 有效时长为 10 分钟
- 开发者拿到 auth\_code,后通过 auth\_code 和开发者的 access\_token 请求到开放平台获取 授权信息接口(获取设备授权信息)
- 收到授权信息后,返回给开放平台服务成功: {"code": "200"},返回其他字符则会接口调用失败.
- 获取 auth\_code 这次操作返回平台成功后,平台认为该次授权成功,最后返回给用户授权成功
- 5. 授权成功后平台会再回调一次,将授权设备和用户设备授权 id- deviceTrustId 传给开发者



图 1-4 开发者获取授权信息

接口协议详情见<u>协议文档</u>(文末)

#### 3.2.5 开发者以用获取授权信息

#### 3.2.5.1 关于托管中的 token

托管/授权关系中,开发者会同时存在4种 token:

- 开发者 token:开发者自身的 token,目前暂时无法操作授权设备,仅用于调用授权设备控制类(包含获取授权设备列表)接口外的其他接口;
- 授权设备刷新 token:与开发者&授权用户关系存在唯一关联,用于刷新授权设备 token,刷新 token 目前会长久存在,刷新 token 只能通过授权回调中的 auth\_code 获得;
- 授权设备 token: 主要通过刷新 token 获取,在获取刷新 token 时,也会给到,用于授权设备的实际操作,同样与开发者&授权用户关系存在唯一关联,通过该 token 获取托管设备列表时,可以获取到所有跟该 token 关联的用户所授权给该开发者的设备,同样,该 token 所能操作的设备权限仅限于该 token 对应用户授权的设备;
- 授权设备 ALL 权限 token: 可以通过开发者 appkey 及 sercret 获取,该 token 和授权设备 token 用法类似,但是不与某用户、用户关系绑定,可以操作所有开发者下的授权 设备和能够获取到所有授权设备。

#### 3.2.5.2 授权设备的操作方式

PS: 授权设备实际上也包含多种 scope-权限,目前默认只能生成设备 ALL 权限的授权页(权限无法根据每次托管修改, 授权页面也为统一界面,无法根据每次授权修改),所以该版本暂时不需要关心

- 1. 通过获取托管设备列表接口,可以获取到参数中 token 权限覆盖的授权设备
- 2. 使用 ezopen 播放和 sdk 播放函数播放时,操作的设备需传入有该设备操作权限的 token,授权设备 ALL 权限 token 或该设备对应的授权设备 token
- 3. 当不操作授权设备时,开放平台的所有需要用到 token 的接口,均使用开发者 token 即 可
- 目前支持的能力:预览(不支持直播地址)、回放、告警消息(alarm)订阅、云台控制、子账号、抓图、探测器网关

# 附:设备托管协议文档

# 1. 开发者 H5 授权登录页地址

开放平台会提供开发者授权的授权登录页地址,开发者可以根据地址生成二维码等,来引导 C用户完成授权流程.

授权登录页地址:

 $https://openauth.ys7.com/trust/device?client_id=\{appKey\}\&response\_type=code\&state=\{state\}ate=sponse\_type=code\&state=\{state\}ate=sponse\_type=code\&state=\{state\}ate=sponse\_type=code\&state=\{state\}ate=sponse\_type=code\&state=sponse\_type=code\&state=sponse\_type=code\&state=sponse\_type=code\&state=sponse\_type=code\&state=sponse\_type=code\&state=sponse\_type=code\_t$ 

access\_token={access\_token}

参数说明:

client_id	开发者的 appKey
response_type	请求类型,默认类型为 code
state	开发者自定义信息,会通过回调接口返回给开发者
access_token	可通过萤石通行证获取用户授权的 access_token, 若未输
	入该参数或参数错误将需要用户重新登录;针对该 token
	的获取方式,请详见萤石通行证接入文档

\*如果 access\_token 未传,或者说该参数过期和异常情况下,会出现如下页面;需要用户额外再登录授权,来保障安全性。

诸登录您的蓝石账户			
<b>並</b> 石	₽	董石开放	
<ul> <li>账号</li> <li>使得</li> </ul>		0	
	登录		

#### 授权登录页展示:

<b>並石开放平台</b> 約州並石明络有限公司	清登录您的萤石账户	我的设备已经视过错
<mark>防介绍:</mark> 此应用为童石开放平台的首方示例,如有任何建议和路问,请 限系统们:open-team@hävision.com	董石 豊石开放…	DS-2CD7A47HWD(8) [NVR 序列号: C70740483-1
如同意 董石开放平台 获取以下权限:	▲ 就号 ★ 安岡 ●	NVK 序列号: 129103497-3     RANG 序列号: C19564467-1     RANG 序列号: C19564467-1     RANG 序列号: 809019554
智不變权		超权

以上界面由萤石开放平台提供,其中 logo、应用名、公司名以及服务介绍等可通过开发者服务账号及应用信息配置同步,如下图:

#### 路径分别为:

- **萤石开放平台官网→开发者服务→我的应用**(同步应用服务介绍、应用名及应用 logo)
- **萤石开放平台官网→开发者服务→账号信息**(同步企业名)

C用户通过授权登录页可选择授权设备及备注设备信息,完成授权后授权信息将调取回调地 址返回给开发者

# 2. 平台集成开发者回调地址逻辑

### 2.1 授权码回调

开发者需要提供<u>回调地址</u>给开放平台后台管理员,地址协议形式为HTTPS/HTTP,同时整个请 求地址由开发者自定义.开放平台会通过开发者的回调地址,将授权码以参数形式给到开发者,参 数名为 auth\_code.开发者只需要通过 auth\_code 来获取需要的授权信息.开发者接口设计逻辑 可参照下图:



如上图:

- 1. 在授权过程中开放平台会将 auth\_code 以及 state 给开发者, auth\_code 有效时长为 10 分钟
- 2. 开发者拿到 auth\_code, 通过 auth\_code 和开发者的 access\_token 请求到开放平台获取授权 信息接口
- 3. 收到授权信息后,返回给开放平台服务成功,返回其他字符则会接口调用失败.

接口协议如下:

#### ▶ 请求地址

Url:{callBackAddress}?auth\_code=value&state=customInfo

#### ▶ 请求方式

Get

▶ 返回信息

{

"code": "200"

}

注:返回上述信息,则开放平台认为开发者信息获取成功.返回其他任何信息,均认为失败.

## 2.2 授权信息(设备)回调

接口协议如下:

#### ▶ 请求地址

Url:{callBackAddress}?opt\_type=opt&deviceSerials=deviceSerials&deviceTrustId=deviceTrustId

参数说明:

Opt	"device_authorize" => 设备授权		
	"device_cancel" => 取消授权		
deviceSerials	用户操作的设备序列号 一般为 deviceSerial: channelNo, 多个设备直接 用 逗号(,)分隔		
deviceTrustId	授权用户 id, 表示设备来源		

#### ▶ 请求方式

Get

▶ 返回信息

{ "code": "200"

}

注: 返回上述信息,则开放平台认为开发者信息获取成功.返回其他任何信息,均认为失败.

# 3.相关接口说明

### 3.1 开发者通过授权码获取授权信息(确认设备授权)

#### ▶ 接口功能

该接口主要是开发者通过授权码获取授权信息,该 token 主要是用来获取单个用户下对应的托管设备.

#### ▶ 请求地址

https://open.ys7.com/api/lapp/trust/device/token/get

#### ➢ 请求参数 Body

Key	类型	必选	备注
access_token	String	Y	开发者 accessToken
auth_code	String	Y	授权码

#### ➢ HTTP 请求报文

POST /trust/device/token/get HTTP/1.1

Host: open.ys7.com

Content-Type: application/x-www-form-urlencoded

access\_token =at.dunwhxt2azk02hcn7phqygsybbw0wv6p&auth\_code=dshfksdhf

#### ▶ 返回信息

#### {

"data": {

"access\_token": "du.2svbvjn82ycx5weh54slfbuebmn6im7o-3np7vpbklx-0zxzdhk-fkzgcpc84", "expires\_in": 1541656437269,

"refresh\_token": "rt.99oz60qc8hn1jfwp9ml4nttid9u3w9h0-8s2wqprgxs-0zhci16-sssvtcmdk",

"openId": "b4a3edff6af84a71b8a12912094359b5",

"device\_trust\_id": "fdb5d9b9ad884edb84f62339ecc62d70"

},

```
"code": "200",
```

"msg": "操作成功"

}

> 返回字段说明:

字段名	类型	描述
access_token String		返回授权托管 token
expires_in	Long	该 token 的过期时间, 单位为毫秒数
refresh_token	String	该 token 用来刷新授权托管 token
openId	String	授权码
device_trust_id	String	设备授权 Id, 用来标识授权用户

#### ▶ 返回码:

code 信息			
200	成功		
1001	用户不存在		
10002	token 过期或异常		
10004	用户不存在		
10015	用户未授权		
70007	授权码不存在		
80000	授权异常请重试		
80002	授权码和 token 不匹配		

# 3.2 刷新设备托管 token 接口

#### ▶ 接口功能

该接口主要是用来刷新过期的设备托管 token

#### ▶ 请求地址

https://openauth.ys7.com/oauth/token/refreshToken

#### ▶ 请求参数

Key	类型	必选	备注
refresh_token	String	Y	授权用户刷新 token
client_id	String	Y	开发者 appKey
open_id	String	Y	授权码
grant_type	String	Y	该参数请填写字符串"refresh_token"
access_token	String	Y	开发者 token

#### ➢ HTTP 请求报文

GET/oauth/token/refreshToken HTTP/1.1 Host: open.ys7.com Content-Type: application/x-www-form-urlencoded refresh\_token =dr.dunwhxt2azk02hcn7phqygsybbw0wv6p& client\_id =dshfksdhf&open\_id=asjdkasd&grant\_type= refresh\_token&access\_token=at.sadasdasd

#### ▶ 返回信息

{

```
"data": {
```

"access\_token": "du.2svbvjn82ycx5weh54slfbuebmn6im7o-3np7vpbklx-0zxzdhk-fkzgcpc84",

```
"expires_in": 1541656437269,
```

 $"refresh\_token":"rt.99 oz 60 qc 8 hn 1 jf wp 9 ml 4 ntt id 9 u 3 w 9 h0 - 8 s 2 wq pr g x s - 0 z hc i 16 - s s s v t c m d k", the set of th$ 

```
"openId": "b4a3edff6af84a71b8a12912094359b5",
```

},

"code": "200",

```
"msg": "操作成功"
```

}

#### ➢ 返回字段说明

字段名	类型	描述	
access_token	String	返回授权托管 token	
expires_in	Long	该 token 的过期时间, 单位为毫秒数	
refresh_token	String	该 token 用来刷新授权托管 token	
openId	String	授权码	

#### ▶ 返回码

code	提示信息
200	成功
10001	参数错误
10017	client_id 不存在
10005	client_id 异常
70003	refresh_token 不存在
70004	refresh_token 已过期
70005	refresh_token 与 client_id 不匹配

70006 refresh\_token 与 open\_id 不匹配

# 3.3 通过开发者 appKey 和 appSecret 获取设备托管 ALL 权限 token

#### ▶ 接口功能

该接口主要是开发者通过 appKey 和 appSecret 获取设备托管权限 token,

该 token 主要是获取账户下所有被托管的设备,来源是多个 C 用户

#### ▶ 请求地址

https://open.ys7.com/api/lapp/trust/device/v2/token/get

#### ➢ 请求参数 Body

Key	类型	必选	备注
аррКеу	String	Y	开发者应用的 key
appSecret	String	Y	开发者应用的 secret

#### ➢ HTTP 请求报文

POST /trust/device/token/get HTTP/1.1 Host: open.ys7.com Content-Type: application/x-www-form-urlencoded appKey =sadasdasdasdadas &appSecret =dshfksdhf

#### ▶ 返回信息

```
{
```

```
"data": {
```

```
"accessToken": "da.3q98ktuz49rerl7ebfewu7br07ove6y9-4u8opzpclp-1rwgjkn-zyy52a5vn",
"expireTime": 1553851609869
```

```
},
"code": "200",
```

```
"msg": "操作成功!"
```

}

#### 运回字段说明:

字段名	类型	描述
accessToken	String	返回授权托管 token
expireTime	Long	该 token 的过期时间, 单位为毫秒数

▶ 返回码:

code	信息
200	成功
10001	参数异常
10005	appKey 异常
10017	appKey 不存在
10030	appkey 和 appsecret 不匹配

# 3.4 获取设备托管列表

#### ▶ 接口功能

该接口主要是用来获取授权托管给开发者的设备列表

#### ▶ 请求地址

https://open.ys7.com/api/lapp/trust/device/list

#### ➢ 请求参数 Body

Key	类型	必选	备注
accessToken	String	Y	设备托管授权的 access_token
pageStart	int	N	分页起始页,从0开始
pageSize	int	N	分页大小,默认为10,最大为50

#### ➢ HTTP 请求报文

POST /api/lapp/trust/device/list HTTP/1.1 Host: open.ys7.com Content-Type: application/x-www-form-urlencoded access\_token =du.dunwhxt2azk02hcn7phqygsybbw0wv6p& pageStart=1&pageSize=2

#### ▶ 返回信息

```
{
    "page": {
         "total": 1,
         "page": 0,
         "size": 10
    },
    "data": [
        {
             "deviceSerial": "210699125",
             "deviceName": "C2W(210699125)1()3~",
             "cameraName": "C2W(210699125)1()3~",
             "channelNo": 1,
       "customName":"客厅东北角"
             "deviceTrustId":"test12334rw"
        }
    ],
    "code": "200",
    "msg": "操作成功!"
}
```

```
▶ 返回字段说明
```

字段名	类型	描述
deviceSerial	String	设备序列号
deviceName	Long	设备名称
cameraName	String	通道名称
channelNo	Int	通道号
customName	String	C 用户自定义设备信息
deviceTrustId	String	设备授权 Id, 用来标识 授权用户
status	int	在线状态: 0:不在线,

open.ys7.com

1:在线

返回码

code	提示信息
200	成功
10001	参数错误
10002	accessToken 过期或异常
10004	用户不存在
10005	client_id 异常
49999	数据异常

## 3.5 开发者取消设备托管

#### ▶ 接口功能

开发者对托管的设备进行取消托管操作

#### ▶ 请求地址

https://open.ys7.com/api/lapp/trust/cancel

#### ➢ 请求参数 Body

Key	类型	必选	备注
aaaagTakan	Stain - V	v	开发者托管 access_token(da 前缀)或者
access loken String	Sung	3 1	设备托管授权的 access_token(du 前缀)
deviceSerials	String	Y	格式:"设备序列号:通道号",多个设备以逗号分隔,最大100个

#### ➢ HTTP 请求报文

POST /api/lapp/trust/cancel HTTP/1.1
Host: open.ys7.com
Content-Type: application/x-www-form-urlencoded
access\_token =du.dunwhxt2azk02hcn7phqygsybbw0wv6p&deviceSerials=D1:C1,D2:C2

#### ▶ 返回信息

{

```
"code": "200",
"msg": "操作成功!"
```

}

#### 返回码

code	提示信息
200	成功
10001	参数错误
10002	accessToken 过期或异常
10004	用户不存在
10031	子账户或萤石用户没有权限
49999	数据异常

# 3.6 其余支持接口

支持托管的接口(可使用托管 token 进行调用):

接口	描述	需要的 权限项
/api/hikvision/v2/**	设备 ISAPI 协议透传	PIPE
/api/device/camera	获取单个通道信息	GET
/api/device/setVideoLevel	设置视频清晰度	CONFIG
/api/sdk/device/info	SDK 获取设备信息	GET
/api/device/capturePicture	抓图接口	CAPTURE
/api/device/getVersionInfo	获取设备版本信息	GET

接口	描述	需要的 权限项
/api/device/upgrade/status	获取设备升级状态	GET
/api/device/upgrade	设备升级	UPGRADE
/api/device/ptz/start	开始云台控制	PTZ
/api/device/ptz/stop	停止云台控制	PTZ
/api/device/preset/add	添加预置点	PTZ
/api/device/preset/move	调用预置点	PTZ
/api/device/preset/clear	清除预置点	PTZ
/api/lapp/camera/video/sound/status	获取视频声音开关状 态	GET
/api/lapp/camera/video/sound/set	设备视频声音开关	CONFIG
/api/lapp/camera/name/update	修改通道名称	UPDATE
/api/lapp/cloud/storage/device/info	获取设备云存储信息	GET
/api/lapp/cloud/storage/enable	开启或关闭设备云存 储	CONFIG
/api/lapp/device/ptz/start	开始云台控制	PTZ
/api/lapp/device/ptz/stop	停止云台控制	PTZ
/api/lapp/device/ptz/mirror	镜像翻转	PTZ
/api/lapp/device/preset/add	添加预置点	PTZ
/api/lapp/device/preset/move	调用预置点	PTZ

接口	描述	需要的 权限项
/api/lapp/device/preset/clear	清除预置点	PTZ
/api/lapp/device/capture	设备抓拍图片	CAPTURE
/api/lapp/device/upgrade	升级设备	UPGRADE
/api/lapp/device/sound/switch/set	设备 wifi 配置声音 开关设置	CONFIG
/api/lapp/device/name/update	修改设备名称	UPDATE
/api/lapp/device/ipc/add	NVR 设备关联 IPC	CONFIG
/api/lapp/device/ipc/delete	NVR 设备删除 IPC	CONFIG
/api/lapp/device/defence/plan/get	获取设备布撤防计划	GET
/api/lapp/device/defence/plan/set	设置设备布撤防计划	CONFIG
/api/lapp/device/scene/switch/set	镜头遮蔽开关设置	CONFIG
/api/lapp/device/ssl/switch/set	声源定位开关设置	CONFIG
/api/lapp/device/fullday/record/swi tch/set	全天录像开关设置	CONFIG
/api/lapp/device/light/switch/set	摄像机指示灯开关设 置	CONFIG
/api/lapp/device/algorithm/config/g et	获取智能算法配置	GET
/api/lapp/device/algorithm/config/s et	设置智能算法模式	CONFIG

接口	描述	需要的 权限项
/api/lapp/device/alarm/sound/set	设置报警声音模式	CONFIG
/api/lapp/device/password/update	修改视频预览密码	CONFIG
/api/lapp/device/notify/switch	开启或关闭设备下线 通知	CONFIG
/api/lapp/device/wifi/qrcode	生成设备配网二维码	CONFIG
/api/lapp/device/capacity	获取设备能力集列表	GET
/api/lapp/device/encodeType/change	透传设备修改编码格 式	CONFIG
/api//lapp/live/address/limited	获取指定有效期的直 播地址	GET
/api/lapp/passengerflow/hourly	查询设备当天每小时 的客流数据	GET
/api/lapp/passengerflow/config/set	客流统计配置	CONFIG
/api/lapp/passengerflow/config/get	获取客流统计配置信 息	GET
/api/lapp/passengerflow/daily	查询设备当天的统计 客流数据	GET
/api/lapp/passengerflow/switch/set	设置客流统计开关	CONFIG
/lapp/trust/device/list	获取托管监控点列表	GET
/lapp/trust/device/v2/list	获取托管监控点列表	GET

接口	描述	需要的 权限项
	(包括非通道设备)	