

Table of Contents

旷视盘古平台简介	1.1
版本更新说明	1.2
平台对接指南	1.3
接口调用说明	1.3.1
错误码定义	1.3.2
数据字典	1.3.3
接口使用流程	1.3.4
API接口文档	1.4
人员管理	1.4.1
上传人员图片	1.4.1.1
上传人员图片（不卡质量）	1.4.1.2
入库质量检测	1.4.1.3
批量添加人员	1.4.1.4
批量删除人员	1.4.1.5
修改人员	1.4.1.6
搜索人员列表	1.4.1.7
根据uuid列表返回人员详细信息列表	1.4.1.8
根据uuid返回人员信息	1.4.1.9
根据员工、访客UUID获取二维码	1.4.1.10
人员组管理	1.4.2
批量添加组	1.4.2.1
删除组	1.4.2.2
修改组	1.4.2.3
组列表	1.4.2.4
查询组详情	1.4.2.5
批量移除组内人员	1.4.2.6
查询组内人员详情	1.4.2.7
组内批量增加或删除人员	1.4.2.8
获取默认组信息（不建议使用）	1.4.2.9
设备管理	1.4.3
设备添加	1.4.3.1
发现设备列表	1.4.3.2
面板机设备导入	1.4.3.3
删除设备预校验	1.4.3.4
删除设备	1.4.3.5

修改设备信息	1.4.3.6
查询设备列表	1.4.3.7
查询设备信息（不建议使用）	1.4.3.8
查询设备详情	1.4.3.9
查询core设备可绑定设备	1.4.3.10
查询非core设备可绑定设备	1.4.3.11
按类型分组设备列表（不建议使用）	1.4.3.12
查询设备型号关系（不建议使用）	1.4.3.13
设备绑定	1.4.3.14
解绑设备预校验	1.4.3.15
设备解绑	1.4.3.16
门禁、人员管控设备列表查询	1.4.3.17
添加门禁、人员管控设备	1.4.3.18
移除门禁、人员管控设备	1.4.3.19
设备可布控列表查询	1.4.3.20
时间计划管理	1.4.4
添加时间计划集合	1.4.4.1
修改时间计划	1.4.4.2
批量删除时间计划	1.4.4.3
时间计划列表	1.4.4.4
查询时间计划详情列表	1.4.4.5
门禁点管理	1.4.5
查询绑定B3R的端设备（分页）	1.4.5.1
查询门禁点设备列表信息（分页）	1.4.5.2
查询同步失败人员信息	1.4.5.3
查询绑定设备类型	1.4.5.4
查询设备配置信息	1.4.5.5
批量添加和移除门禁点	1.4.5.6
移除门禁点	1.4.5.7
绑定开关	1.4.5.8
远程常开	1.4.5.9
远程常闭	1.4.5.10
远程开门	1.4.5.11
恢复正常	1.4.5.12
配置设备	1.4.5.13
查询同步列表（不建议使用）	1.4.5.14
同步失败详情（不建议使用）	1.4.5.15
门禁设备告警	1.4.6

门禁设备告警列表查询	1.4.6.1
通行权限管理	1.4.7
添加通行权限	1.4.7.1
删除通行权限	1.4.7.2
编辑通行权限	1.4.7.3
通行权限列表	1.4.7.4
通行记录	1.4.8
查询通行记录列表	1.4.8.1
考勤设备管理	1.4.9
批量添加考勤设备	1.4.9.1
批量删除考勤设备	1.4.9.2
考勤设备列表查询	1.4.9.3
考勤组管理	1.4.10
创建考勤组	1.4.10.1
批量删除考勤组	1.4.10.2
更新考勤组	1.4.10.3
考勤员工分页列表查询	1.4.10.4
考勤组分页列表查询	1.4.10.5
考勤组详情查询	1.4.10.6
考勤记录	1.4.11
员工月考勤记录分页列表查询	1.4.11.1
员工考勤记录日历模式查询	1.4.11.2
考勤记录分页列表查询	1.4.11.3
考勤记录详情分页列表查询	1.4.11.4
考勤记录调整	1.4.11.5
考勤统计	1.4.12
考勤统计记录分页列表查询	1.4.12.1
出勤调整	1.4.13
出勤调整	1.4.13.1
出勤调整记录分页列表查询	1.4.13.2
更新出勤调整记录	1.4.13.3
布控规则管理	1.4.14
添加规则	1.4.14.1
修改规则	1.4.14.2
批量删除规则	1.4.14.3
规则分页查询	1.4.14.4
人员识别记录	1.4.15
员工、访客、重点人员&全局陌生人列表	1.4.15.1

员工、访客、重点人员&全局陌生人详情	1.4.15.2
人员轨迹	1.4.16
人员轨迹查询	1.4.16.1
人员聚类	1.4.17
人员聚类任务添加	1.4.17.1
人员聚类任务状态查询	1.4.17.2
人员聚类任务详情查看	1.4.17.3
人员聚类任务列表查看	1.4.17.4
人员聚类人员详情查看	1.4.17.5
人员聚类任务删除	1.4.17.6
人脸比对	1.4.18
1:1查询	1.4.18.1
1: N列表查询	1.4.18.2
1: N列表详情	1.4.18.3
车辆基础信息	1.4.19
车辆基础信息列表	1.4.19.1
车辆基础信息批量删除	1.4.19.2
车辆基础信息批量新增	1.4.19.3
车辆基础信息更新	1.4.19.4
车辆基础详情信息	1.4.19.5
结构化、车辆管控布控	1.4.20
结构化、车辆管控布控列表	1.4.20.1
结构化、车辆管控布控删除	1.4.20.2
结构化、车辆管控布控新增	1.4.20.3
结构化、车辆管控布控更新	1.4.20.4
结构化、车辆管控布控详情	1.4.20.5
车辆管控记录	1.4.21
车辆识别记录列表	1.4.21.1
车辆识别记录详情	1.4.21.2
车辆轨迹	1.4.22
车辆轨迹查询	1.4.22.1
结构化检索	1.4.23
人体数据检索	1.4.23.1
人体数据详情查询	1.4.23.2
人脸数据检索	1.4.23.3
人脸数据详情查询	1.4.23.4
机动车数据检索	1.4.23.5
机动车数据详情查询	1.4.23.6

非机动车数据检索	1.4.23.7
非机动车数据详情查询	1.4.23.8
警戒记录	1.4.24
警戒记录列表查询	1.4.24.1
视频监控	1.4.25
云台控制	1.4.25.1
视频实时预览	1.4.25.2
视频抓图	1.4.25.3
盘古页面回调数据	1.4.26
事件记录回调	1.4.26.1
事件记录回调说明	1.4.26.1.1
事件记录回调数据结构	1.4.26.1.2
回调数据示例	1.4.26.1.3
通行记录回调	1.4.26.2
考勤记录回调	1.4.26.3
业务数据推送	1.4.27
业务数据记录订阅	1.4.27.1
查询业务数据记录订阅	1.4.27.2
解除业务数据记录订阅	1.4.27.3
业务推送数据结构	1.4.27.4
通行识别记录推送	1.4.27.4.1
安防识别记录推送	1.4.27.4.2
结构化记录推送	1.4.27.4.3
车辆记录推送	1.4.27.4.4
警戒记录推送	1.4.27.4.5

- 1. 盘古平台简介
 - 1.1. 产品概述
 - 1.2. 系统架构
 - 1.3. 系统配置流程

1. 盘古平台简介

1.1. 产品概述

盘古AIOT视图综合应用平台（以下简称盘古）是一款旷视自主研发的面向楼宇园区提供极致AI体验的开放性综合应用平台。

通过接入旷视敏观相机、神行面板机等AI端硬件产品，配合魔盒和AI算法引擎等边缘计算中心，盘古支持多种人脸、人体、车辆识别算法应用，最终为用户提供了门禁、考勤、迎宾、访客等一脸通应用，重点人员识别、陌生人管控、人员聚类、人车结构化等综合管控应用，人员入侵，车辆禁停等警戒管理应用。

同时，盘古提供丰富的OpenAPI接口，支持将AI数据和业务数据上传给行业业务平台，为AI赋能百业提供AI数据汇聚和分析能力。

1.2. 系统架构



盘古系统架构整体遵循“云-边-端”的三层架构：

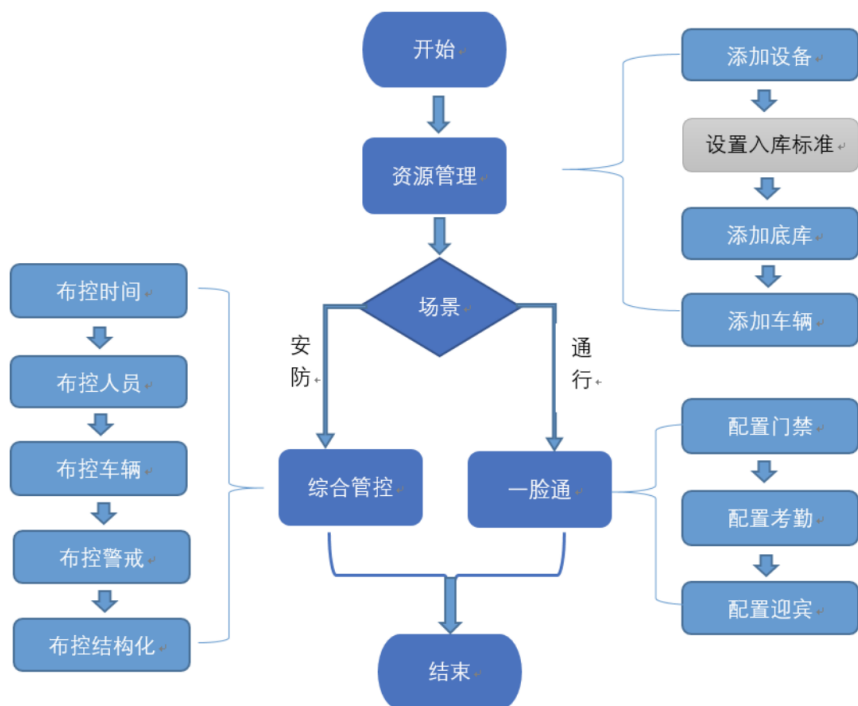
云：盘古平台，可管理边、端设备，提供一脸通，综合管控等业务应用。

边：提供边缘处理的能力的计算单元和存算单元，例如智能分析盒、存算一体机。

端：旷视科技的各类AI端设备，例如各种摄像机和面板机。

1.3. 系统配置流程

基本流程参考下图，详细配置流程请参考盘古系统用户帮助文档。



- 1. 更新说明
 - 1.1. V1.0.5更新说明
 - 1.2. V1.0.2更新说明
 - 1.3. V1.0.1更新说明

1. 更新说明

- V1.0.5更新说明
- V1.0.2更新说明
- V1.0.1更新说明

1.1. V1.0.5更新说明

- 1.门禁点支持配置二次识别间隔
- 2.增加获取员工二维码，二维码支持有效期，支持二维码有效期刷新
- 3.增加控制门禁点常开、常闭、取消常开、取消常闭
- 4.支持北京健康码，新增健康码配置项，在面板机参数配置中对健康码参数进行配置；员工和访客通行记录中新增健康码状态信息
- 4.设备绑定优化，绑定时按设备模式和服务器模式进行绑定
- 5.设备添加支持PDNS，新增主动模式，可实现设备与盘古跨网接入
- 6.1：N接口优化，增加上传人员图片接口（不卡质量判断）；新增质量判断接口

1.2. V1.0.2更新说明

- 1.人员布控中新增对员工、访客的布控。并将陌生人拆分为通道陌生人和全局陌生人
- 2.区域警戒新增多项行为警戒，包括：人员值岗/离岗-离岗、人员值岗/离岗-超员、人员值岗/离岗-少员、人员奔跑、摔倒检测、人员扭打、抽烟检测、看手机、接打电话、人员聚众、防碰撞预警、车辆逆行
- 3.新增云台控制功能，可以对摄像机进行视角、方位、焦距、光圈的调整
- 4.视频预览画面支持手动抓图
- 5.设备，支持单设备多路AI：支持一个物理设备，可使用不同的模式添加到盘古
- 6.设备，网络摄像头，支持onvif协议接入到盘古

1.3. V1.0.1更新说明

- 1.门禁告警新增测温模块失效
- 2.设备分为门禁设备和人员管控设备，系统中新增的设备中，只有被添加到门禁设备中后，才可以一脸通中使用，只有被添加到人员管控设备中后，才可以在综合管控中使用
- 3.门禁参数新增门禁配置（门锁控制、门磁类型、认证方式、开门超时时长、门禁控制时间、延迟门锁控制）和通讯配置（开启韦根、使用模式、韦根格式）；识别参数新增“FACE AE”“口罩识别”“口罩阈值”；门禁设备新增测温参数
- 4.人员管控设备的配置分为重点人员参数和陌生人参数，包括水平角度、垂直角度、旋转角度、模糊度、识别间隔、识别阈值、陌生人识别阈值、活体开关（陌生

人无活体）等

5.系统新增测温 and 口罩检测功能，在设备-门禁设备-测温参数中进行配置，通行状态新增“未通行（高温）”和“未通行（未佩戴口罩）”

- 1. 接口调用方式
 - 1.1. 鉴权方式
 - 1.2. 签名计算规则
 - 1.3. 签名示例
 - 1.4. 可用签名及秘钥
 - 1.5. 签名SDK
 - 1.5.1. Java版本

1. 接口调用方式

1.1. 鉴权方式

OpenAPI请求采用接口签名方式，需要在HTTP HEADER中加入以下请求头：

序号	header名	描述
1	ctimestamp	客户端本地unix毫秒时间戳
2	cnonce	6位随机字符串，可含字母，客户端自行生成
3	cappkey	标识第三方系统身份，由盘古分配
4	csign	按签名规则计算的签名

1.2. 签名计算规则

签名算法

MD5，32位小写

通过以下几部分组合计算：

- A: 请求URI，以/开头，如/v1/api/device/list
- B: request的请求method，大写，如GET、POST、PUT、DELETE
- C: 请求参数按照参数key正序排序后连接的字符串：如请求参数中有akey=1,ckey=2,bkey=3,则连接的参数字符串为：akey=1&bkey=3&ckey=2(注意：参数值采用url编码前的原值)，目前，所有接口都是body传参，所以这里是""
- D: 请求的requestbody中的内容的MD5(32位小写，下同)，如无内容，则为""【注，对于文件上传类url，该字段用空字符串替代】
- E: cappkey对应的秘钥，如：appkey1对应的秘钥为sdfajk3242324fa!dj7
- F: header中的ctimestamp、cnonce、cappkey

sign计算公式：

csign = MD5(A-B-C-D-E-ctimestamp-cnonce-cappkey)

1.3. 签名示例

这里以查询设备列表为例：

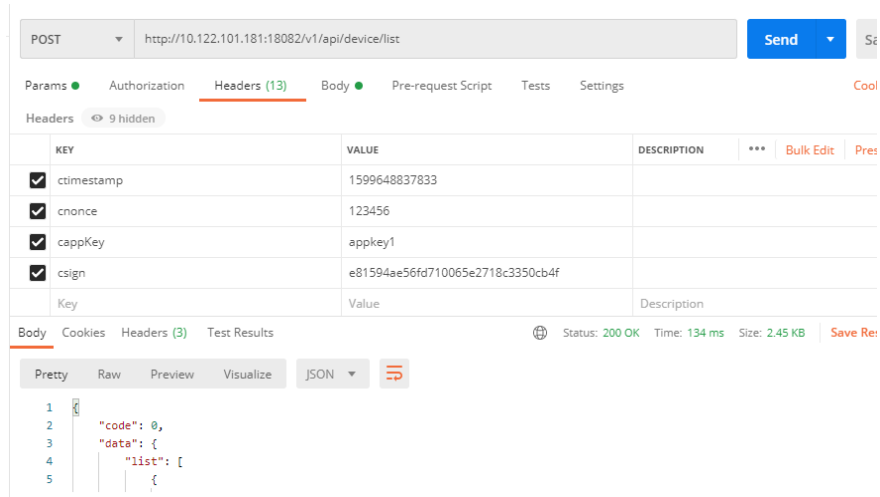
- A="/v1/api/device/list"
- B="POST"
- C=""
- 请求体数据是:

```
{
  "pageNum":1,
  "pageSize":10
}
```

则D为: 1e673403a5bda635cea1b76fcc6755e2

- E="sdfajk3242324fa!dj7"
- ctimestamp="1599648837833"
- cnonce="123456"
- cappkey="appkey1"
- 则csign=MD5("/v1/api/device/list-POST-1e673403a5bda635cea1b76fcc6755e2-sdfajk3242324fa!dj7-1599648837833-123456-appkey1"), 结果为:
e81594ae56fd710065e2718c3350cb4f

那么, 我们可以如下请求:



1.4. 可用签名及密钥

目前, 盘古里面有两组cappkey和密钥, 如下:

cappkey	secret
appkey1	sdfajk3242324fa!dj7
appkey2	zxcv&jlkfd234adf#xi9

1.5. 签名SDK

1.5.1. Java版本

```

package com.megvii.sign.sdk;
import com.alibaba.fastjson.JSONObject;
import org.apache.commons.lang3.StringUtils;
/**
描述:
Author: liuxing
Date: 2020-09-11
*/
public class SignUtils {
    private static final String SIGN_FORMAT = "%s-%s-%s-%s-%s-%s-%s-%s";
    /**
签名算法生成卡密昂
@param ctimestamp
@param cnonce
@param requestBody 请求体对象
@param requestParam
@param url
@param method
@param cappKey
@param secret
@return
*/
    public static String sign(String ctimestamp, String cnonce, Object requestBody, String requestParam, String url, String method, String cappKey, String secret) {
        String requestBodyMd5 = "";
        if (requestBody != null) {
            String requestBodyJson = JSONObject.toJSONString(requestBody);
            if (StringUtils.isNotEmpty(requestBodyJson)) {
                requestBodyMd5 = DigestUtil.encryptMd5(requestBodyJson);
            }
        }
        String signStr = String.format(SIGN_FORMAT, url, method, requestParam, requestBodyMd5, ctimestamp, cnonce, cappKey, secret);
        String sign = DigestUtil.encryptMd5(signStr);
        return sign;
    }
}

package com.megvii.sign.sdk;
import javax.crypto.Cipher;
import javax.crypto.SecretKey;
import javax.crypto.SecretKeyFactory;
import javax.crypto.spec.DESedeKeySpec;
import java.security.MessageDigest;
import java.security.NoSuchAlgorithmException;
public class DigestUtil {
    /** 算法名称 */
    private static final String ALGORITHM = "DESede";
    /** 16进制字母 */
    private static final char[] HEX_DIGITS={'0','1','2','3','4','5','6','7','8','9','a','b','c','d','e','f'};
    /**
    * @param decrypt 要加密的字符串
    * @return 加密的字符串
    * SHA1加密
    */
    public final static String SHA1(String decrypt) {
        try {
            MessageDigest digest = java.security.MessageDigest
                .getInstance("SHA-1");
            digest.update(decrypt.getBytes());
            byte messageDigest[] = digest.digest();
            // Create Hex String
            StringBuffer hexString = new StringBuffer();
            // 字节数组转换为 十六进制 数
            for (int i = 0; i < messageDigest.length; i++) {
                String shaHex = Integer.toHexString(messageDigest[i] & 0xFF);
                if (shaHex.length() < 2) {
                    hexString.append(0);
                }
            }
        } catch (NoSuchAlgorithmException e) {
            e.printStackTrace();
        }
        return hexString.toString();
    }
}

```

```

    }
    hexString.append(shaHex);
}
return hexString.toString();
}
catch (NoSuchAlgorithmException e) {
    e.printStackTrace();
}
return "";
}
}
/**
 * 3DES加密,key必须是长度大于等于 3*8 = 24 位
 * @param src
 * @param key
 * @return
 * @throws Exception
 */
public static String encrypt3DES(String src, String key) {
    byte[] keyBytes = key.getBytes();
    try {
        DESedeKeySpec dks = new DESedeKeySpec(keyBytes);
        SecretKeyFactory keyFactory = SecretKeyFactory.getInstance(ALGORITHM);
        SecretKey securekey = keyFactory.generateSecret(dks);
        Cipher cipher = Cipher.getInstance("DESede/ECB/PKCS5Padding");
        cipher.init(Cipher.ENCRYPT_MODE, securekey);
        byte[] b=cipher.doFinal(src.getBytes());
        return StringUtil.byte2hex(b);
    }
    catch (Exception e) {
        return null;
    }
}
}
/**
 * 3DESECB解密,key必须是长度大于等于 3*8 = 24 位
 * @param src
 * @param key
 * @return
 * @throws Exception
 */
public static String decrypt3DES(String src, String key) {
    byte[] keyBytes = key.getBytes();
    try {
        //--通过base64,将字符串转成byte数组
        byte[] bytesrc = StringUtil.hex2byte(src);
        //--解密的key
        DESedeKeySpec dks = new DESedeKeySpec(keyBytes);
        SecretKeyFactory keyFactory = SecretKeyFactory.getInstance(ALGORITHM);
        SecretKey securekey = keyFactory.generateSecret(dks);
        //--Cipher对象解密
        Cipher cipher = Cipher.getInstance("DESede/ECB/PKCS5Padding");
        cipher.init(Cipher.DECRYPT_MODE, securekey);
        byte[] retbyte = cipher.doFinal(bytesrc);
        return new String(retbyte);
    }
    catch (Exception e) {
        return null;
    }
}
}
/**
 *
 * Description:md5加密
 * @Create_by:JH
 * @Create_date:2014-9-9
 * @Last_Edit_By:
 * @Edit_Description
 * @Create_Version:exinhua 1.0

```

```

*/
public static String encryptMd5(String src) {
    if(StringUtil.isEmpty(src)) {
        return src;
    }
    try {
        byte[] btInput = src.getBytes();
        // 获得MD5摘要算法的 MessageDigest 对象
        MessageDigest mdInst = MessageDigest.getInstance("MD5");
        // 使用指定的字节更新摘要
        mdInst.update(btInput);
        // 获得密文
        byte[] md = mdInst.digest();
        // 把密文转换成十六进制的字符串形式
        return byteToHexString(md);
    }
    catch (Exception e) {
        throw new RuntimeException("error occurred when encrypt", e);
    }
}
/**
 * 对输入进行sha1加密后，在进行16进制转换
 * @param value
 * @return
 */
public static String hexSHA1(String value) {
    try {
        MessageDigest md = MessageDigest.getInstance("SHA-1");
        md.update(value.getBytes("utf-8"));
        byte[] digest = md.digest();
        return byteToHexString(digest);
    }
    catch (Exception ex) {
        throw new RuntimeException(ex);
    }
}
/**
 * 字节数组转换成16进制表示
 * @param bytes
 * @return
 */
public static String byteToHexString(byte[] bytes) {
    int j = bytes.length;
    char[] str = new char[j * 2];
    int k = 0;
    for (int i = 0; i < j; i++) {
        byte byte0 = bytes[i];
        str[k++] = HEX_DIGITS[byte0 >>> 4 & 0xf];
        str[k++] = HEX_DIGITS[byte0 & 0xf];
    }
    return new String(str);
}
}
package com.megvii.sign.sdk;
/**
 * 字符串工具类
 */
public class StringUtil {
    public static final String[] HEX_ARRAYS = { "0", "1", "2", "3", "4", "5",
        "6", "7", "8", "9", "a", "b", "c", "d", "e", "f" };
    /**
     * 判断value是否是null或者其length为0
     *
     * @param value
     * @return
     */

```

```

*/
public static Boolean isEmpty(String value) {
    return value == null || value.length() == 0;
}
/**
 * 判断value是否不是null且其length>0
 *
 * @param value
 * @return
 */
public static Boolean isNotEmpty(String value) {
    return !isEmpty(value);
}
/**
 * trim操作
 *
 * @param value
 * @return
 */
public static String trim(String value) {
    if (value == null) {
        return null;
    }
    return value.trim();
}
/**
 * byte string to hex string
 *
 * @param b
 * @return
 */
public static String byte2hex(byte[] b) {
    if (b == null) {
        return null;
    }
    if (b.length == 0) {
        return "";
    }
    StringBuilder retBuilder = new StringBuilder();
    for (int n = 0; n < b.length; ++n) {
        retBuilder.append(HEX_ARRAYS[(b[n] & 0xF0) >> 4]);
        retBuilder.append(HEX_ARRAYS[b[n] & 0x0F]);
    }
    return retBuilder.toString();
}
/**
 * hex string to byte string
 *
 * @param str
 * @return
 */
public static byte[] hex2byte(String str) {
    if (str == null)
        return null;
    str = str.trim();
    int len = str.length();
    if ((len == 0) || (len % 2 == 1))
        return null;
    byte[] b = new byte[len / 2];
    byte tmp = 0;
    char[] strc = str.toCharArray();
    for (int i = 0; i < len; i++) {
        byte t = 0;
        if (strc[i] >= '0' && strc[i] <= '9') {
            t = (byte) (strc[i] - '0');
        } else if (strc[i] >= 'A' && strc[i] <= 'F') {

```

```

        t = (byte) (strs[i] - 'A' + 10);
    } else if (strs[i] >= 'a' && strs[i] <= 'f') {
        t = (byte) (strs[i] - 'a' + 10);
    }
    if ((i & 0x1) == 1) {
        tmp <<= 4;
        tmp += t;
        b[i / 2] = tmp;
        tmp = 0;
    } else {
        tmp = t;
    }
}
return b;
}
/**
 * 判断一个字符串是否只包含数字(10进制)字符
 *
 * @param str
 * @return
 */
public static Boolean isNumeric(String str) {
    if (isEmpty(str)) {
        return false;
    }
    char[] arr = str.toCharArray();
    for (char c : arr) {
        if (!Character.isDigit(c)) {
            return false;
        }
    }
    return true;
}
/**
 * 混淆字符串(显示前pn个字符和后tn个字符, 其余全部用*填充)
 *
 * @param source
 * @param pn
 * @param tn
 * @return
 */
public static String mix(String source, int pn, int tn) {
    return mix(source, pn, tn, '*');
}
/**
 * 混淆字符串(显示前pn个字符和后tn个字符, 其余前部用mixChar填充)
 *
 * @param source
 * @param pn
 * @param tn
 * @param mixChar
 * @return
 */
public static String mix(String source, int pn, int tn, char mixchar) {
    if (source == null || source.length() <= pn + tn) {
        return source;
    }
    int len = source.length();
    StringBuilder tmp = new StringBuilder(source.length());
    char[] mobileAs = source.toCharArray();
    for (int i = 0; i < pn; i++) {
        tmp.append(mobileAs[i]);
    }
    for (int i = 0; i < len - (pn + tn); i++) {
        tmp.append(mixchar);
    }
}

```



```

        for (int i = len - tn; i < len; i++) {
            tmp.append(mobileAs[i]);
        }
        return tmp.toString();
    }
    /**
     * 获取字符串长度(全角2,半角1)
     * @param value
     * @return
     */
    public static final int getLength(String value) {
        if(StringUtil.isEmpty(value)) {
            return 0;
        }
        int len = 0;
        for (char c : value.toCharArray()) {
            len++;
            if(isSbcCase(c)) {
                len++;
            }
        }
        return len;
    }
    /**
     * 判断字符是否是半角
     * @param c
     * @return
     */
    public static final Boolean isDbcCase(char c) {
        int k = 0x80;
        return c / k == 0 ? true : false;
    }
    /**
     * 判断是否是全角
     * @param c
     * @return
     */
    public static final Boolean isSbcCase(char c) {
        return isDbcCase(c);
    }
    /**
     * 判断是否都是数字
     * @param value
     * @return
     */
    public static final Boolean isAllDigits(String value) {
        if(StringUtil.isEmpty(value)) {
            return true;
        }
        for (char c : value.toCharArray()) {
            if(c < '0' || c > '9') {
                return false;
            }
        }
        return true;
    }
    /**
     * 在指定的字符串前面填充0,直到字符串的长度达到指定的长度
     * 示例: 输入6, 2 ---> 返回06
     * @param data 需要被填充的字符串
     * @param fillStr 需要填充得字符串
     * @param length 目标字符串的长度
     * @return 返回填充后的字符串
     */
    public static String frontFillStr(String data, String fillStr, int length) {
        StringBuilder sb = new StringBuilder();

```

```
        for (int i = 0; i < length-data.length(); i++) {  
            sb.append(fillStr);  
        }  
        return sb.append(data).toString();  
    }  
}
```

请求示例

```

package test;
import lombok.AllArgsConstructor;
import lombok.Builder;
import lombok.Data;
import lombok.NoArgsConstructor;
/**
 * 描述:
 * Author: liuxing
 * Date: 2020-09-14
 */
@Data
@AllArgsConstructor
@NoArgsConstructor
@Builder
public class DeviceReq {
    private String name;
    private Integer pageNum;
    private Integer deviceType;
    private Integer pageSize;
}

package test;
import com.alibaba.fastjson.JSONObject;
import com.megvii.sign.sdk.SignUtilis;
import okhttp3.*;
import java.util.concurrent.TimeUnit;
/**
 * 描述: 模拟一次请求的过程
 * Author: liuxing
 * Date: 2020-09-14
 */
public class SignUtilTest {
    public static void main(String[] args) {
        String ctimestamp = "1599648837833";
        // 时间戳, 用户自己生成
        String cnonce = "123456";
        // 6位随机码, 用户自己生成
        String requestParam = "";
        // 请求参数, form表单方式, 盘古大部分接口都是body传参, 故这里大部分都是"", 如果有参数
        String url = "/v1/api/device/list";
        // 要请求的url, 看要请求的接口说明
        String method = "POST";
        // 请求方式, 看要请求的接口说明
        String cappKey = "appkey1";
        // appkey, 固定,
        String secret = "sdfajk3242324fa!djq7";
        // 密钥, 固定
        DeviceReq deviceReq = DeviceReq.builder().name("core").build();
        // 构建请求体, 看要请求的接口说明
        // 计算签名
        String csign = SignUtilis.sign(ctimestamp, cnonce, deviceReq, requestParam, url)
        // 发起请求
        String realUrl = "http://10.122.101.181:18082" + url;
        postJson(ctimestamp, cnonce, cappKey, csign, deviceReq, realUrl);
    }

    private static void postJson(String ctimestamp, String cnonce, String cappKey, Str
        String bodyJson = JSONObject.toJSONString(requestData);
        OkHttpClient httpClient = new OkHttpClient.Builder()
            .connectTimeout(3000, TimeUnit.SECONDS)
            .readTimeout(30000, TimeUnit.SECONDS)
            .writeTimeout(30000, TimeUnit.SECONDS)
            .build();
        MediaType mediaType = MediaType.parse("application/json;charset=UTF-8");
        RequestBody requestBody = RequestBody.create(mediaType, bodyJson);
        Request request = new Request.Builder()
            .url(url)

```

```
        .addHeader("cnonce", cnonce)
        .addHeader("ctimestamp", ctimestamp)
        .addHeader("cappKey", cappKey)
        .addHeader("csign", csign)
        .post(requestBody)
        .build();
    Call call = httpClient.newCall(request);
    try {
        Response response = call.execute();
        System.out.println("请求结果: " + response.body().string());
        System.out.println(response);
    }
    catch (Exception e) {
        e.printStackTrace();
    }
}
```

- o 1.1. 公共错误码
- o 1.2. 人员模块错误码
- o 1.3. 设备模块错误码
- o 1.4. 通行权限模块错误码
- o 1.5. 通行业务错误码
- o 1.6. 考勤业务错误码
- o 1.7. 通行记录错误码
- o 1.8. 安防人员业务错误码
- o 1.9. 结构化业务错误码
- o 1.10. 警戒业务错误码
- o 1.11. 事件业务错误码

1.1. 公共错误码

错误码	描述	备注
0	成功	
-1	系统异常	
100001	未登录	
100002	无访问权限	
100003	参数不合法	
100004	转化异常	
100005	导出异常	
100006	需要重定向	
100007	没有可以导出的数据	
100008	签名错误	
100009	不能超过90天	
100010	系统繁忙	
100011	该操作耗时较长，请稍后在页面再次确认操作结果	
999999	extra	

1.2. 人员模块错误码

错误码	描述	备注
101001	feature入库失败	
101002	feature修改失败	
101003	feature删除失败	
101004	feature查询失败	
101005	机构不存在	
101006	数据插入失败	
101007	手机号已存在	
101008	证件号已存在	
101008	邮箱已存在	
101010	卡号已存在	
101011	员工编号已存在	
101012	该人员不存在	
101013	删除人员失败	
101014	父级机构不存在	
101015	公司ID不一致	
101016	顶级机构不允许删除	
101017	机构删除失败	
101018	组织架构不允许变更	
101019	生成二维码失败	
101020	访客访问时间不存在	
101021	访客访问时间已经失效	
101022	照片名称不能为空	
101023	人员姓名不能为空	
101024	姓名的长度不能超过40位	
101025	手机号码格式错误	
101026	密码格式错误	
101027	邮箱格式错误	
101028	职位格式错误	
101029	备注格式错误	
101030	照片名称重复	
101031	所属机构格式错误	
101032	证件号格式错误	

错误码	描述	备注
101033	导出组织信息失败	
101034	未获取到有效人员信息	
101035	导出人员信息异常	
101036	导入人员信息异常	
101037	token校验出错	
101038	token过期	
101039	读取文件出错误	
101040	imageuri为空	
101041	获取照片失败	
101042	创建文件路径失败	
101043	上传照片失败，请核查人员信息是否已成功导入	
101044	公司不存在	
101045	人员类型不存在	
101046	同级机构名称已存在	
101047	发送消息到mq异常	
101048	机构层级过深	
101049	机构ID为空	
101050	模糊度不通过	
101051	人脸大小不通过	
101052	垂直角度不通过	
101053	水平角度不通过	
101054	旋转角度不通过	
101101	面部存在遮挡	
101055	亮度不通过	
101056	标准差亮度不通过	
101057	机构和公司不匹配	
101058	公司名字重复	
101059	受访员工不存在	
101099	访客时间不能为空	
101100	访客时间范围有误	
101060	卡号格式错误	
101061	添加组织异常	

错误码	描述	备注
101062	添加人员异常	
101063	顶级机构不唯一	
101064	部门层级不能超过10级	
101065	读取导出人员模板失败	
101066	图片格式或名称错误	
101067	人员数量超过最大值	
101068	该区域存在设备，不能删除	
101069	访客被拜访员工不能为空	
101070	人脸分析错误，可能是图片人脸质量较差或者没有人脸	
101071	员工编号格式错误	
101072	员工生日格式错误	
101073	员工入职日期格式错误	
101074	来访目的格式错误	
101075	未获取到有效照片，请确保照片在压缩包一级目录下	
101076	更新人员信息失败	
101077	汉字转化拼音失败	
101078	区域名称已存在	
101079	未上传照片，请核对照片是否存在或者照片名称是否与导入人员信息表中的一致	
101080	此文件目录无效	
101081	该设备已进行布控	
101082	图片大小超过10MB	
101083	父区域不存在	
101084	区域不存在	
101085	照片名称有误	
101086	人员信息重复,存在重复信息的人员	
101087	人员照片覆盖	
101088	匹配项必选	
101089	车牌号已存在	
101090	唯一标识已存在	
101091	规则名称已存在	
101092	人员关系删除失败	
101093	人员关系增加失败	

错误码	描述	备注
101094	区域已经初始化过	
101095	人员所在组数量已经达到20个上限	
101096	规则不存在	
101097	文件大小过大	
101098	姓名格式不正确	
101099	UUID已存在	
101100	唯一标识格式不正确	
101101	访客所属单位格式不正确	
101102	访客车牌号格式不正确	
101103	有子区域的区域不允许删除	
101104	人脸分析错误，可能是图片中存在多个人脸	
101105	人员uuid类型不符	
101106	人员添加出现未知异常	
101107	任务状态出现错误，停止执行	
101108	视频监控系统地址未配置	
101109	视频监控系统地址无效	
101110	视频监控系统API调用异常	
101111	手机号不允许为空	

1.3. 设备模块错误码

错误码	描述	备注
102020	新增设备配置失败	
102023	删除设备配置失败	
102024	修改设备配置失败	
102025	设备记录不存在	
102035	添加设备版本信息失败	
102046	设备恢复出厂设置失败	
102049	设备类型不存在	
102050	设备型号不存在	
102051	设备snCode不存在	
102052	设备mac不存在	
102053	设备IP不存在	
102054	设备导入失败	
102059	设备型号错误	
102060	该设备类型无法进行绑定	
102061	设备绑定失败	
102062	设备分组查询失败	
102063	无设备可绑	
102064	snCode已存在	
102065	关联查询存储设备信息失败	
102066	设备解绑失败	
102067	SN号已被绑定	
102071	结构化设备更新布控失败	
102072	结构化设备删除布控失败	
102073	警戒设备布控失败	
102074	警戒设备删除布控失败	
102075	警戒设备更新布控失败	
102076	设备类型和型号不匹配	
102077	设备模式和型号不匹配	
102079	算力设备没有绑定显示设备	
102082	门禁点移除失败	
102083	查询设备绑定状态失败	
102085	设备删除预校验失败	
102086	该设备未绑定	

错误码	描述	备注
102087	设备名称已存在	
102088	IP地址已存在	
102089	Mac地址已存在	
102090	视频流地址已存在	
102091	设备离线状态	
102092	设备布控查询类型失败	
102097	设备非离线状态，不可强制删除	
102099	智能相机布控总数超限	
102107	超过企业最大设备接入数,添加设备失败	
102108	设备接入必须在线	
102110	设备未绑定，不能进行业务数据查询	
102113	设备不存在	
102114	core能绑定人脸识别的总数超过限制	
102115	core能绑定的人脸抓拍机总数超过限制	
102116	core能绑定的结构化总数超过限制	
102117	core能绑定的警戒总数超过限制	
102118	设备绑定部分失败	
102119	设备已布控不能修改	
102123	设备绑定或解绑失败	
102124	设备管控查询失败	
102125	设备管控移除失败	
102129	解除失败，已进行陌生人布控	
102135	设备已设置联动，请先解除布控	
102137	IP规则0.0.0.0~255.255.255.255之间	
102138	用户名由中文、字母和数字组成	
102139	用户名长度1-50位	
102142	设备已设置门禁点布控，请先解除布控	
102143	设备已设置人员布控，请先解除布控	
102144	设备已设置警戒布控，请先解除布控	
102145	设备已设置结构化布控，请先解除布控	
102149	添加人员管控的设备中，存在类型不合法	
102150	IP的设备模式已存在	
102154	多路AI设备已绑定智能服务器，无法进行修改操作	

1.4. 通行权限模块错误码

错误码	描述	备注
103001	新增Group失败	
103002	更新Group失败	
103003	新增GroupPersonRelation失败	
103004	删除GroupPersonRelation失败	
103004	参数不能为空	
103005	删除Group失败	
103100	数据错误	
103101	参数错误	
103102	版本错误	
103103	无数据更新	
103104	设备ID错误	
103105	设备更新时，不能同时触发删除与更新操作	
103106	时间计划不能超过128条	
103107	通行时间段不能超过3条	
103108	节假日不能超过32条	
103109	单个设备的通行权限最多下发128个组	
103109	通行权限已被删除，请刷新页面重新请求	
103109	时间计划已被删除，请刷新页面重新请求	
103110	组信息异常，核实数据是否被异常删除	
103111	全员组操作参数错误	
103112	工作日的周计划最多设置7天	
103113	工作日的周计划不允许为空	
103114	该名字不被允许	
103115	该时间计划不允许被删除	
103116	时间计划名称不允许重复	
103117	存在员工没有分配部门	
103118	存在访客没有设置访问对象	
103119	员工数据不全，请核实数据	
103120	访客数据不全，请核实数据	
103121	设备未绑定组	
103122	公司无全员组	
103123	无法删除默认组	
103124	组名重复	

错误码	描述	备注
103125	默认组不允许修改	
103126	组不存在	
103127	人员类型与组类型不匹配	
103128	人员不存在	
103129	部分人员不存在	
103130	人员类型不一致	
103132	公司人员达到最大值，无法添加	
103133	本类组数目达到最大值，无法添加	
103134	组名称长度应为2~32	
103135	时间计划名称长度应为2~32	
103136	时间计划不存在	
103137	默认时间计划名称不允许修改	
103138	时间计划传入的时间段异常	
103139	时间计划传入的时间段异常,开始时间必须小于结束时间	
103140	设备更新时间为空	
103141	设备全量数据无更新	
103142	非重点人员组	
103143	B3R通行权限最多支持64个组	
103144	绑定B3R的设备通行权限最多支持20个组	
103145	CORE通行权限最多支持10000个组	
103145	面板机通行权限最多支持128个组	
103146	人组关系删除发送mq消息异常	
103147	设备没有布控	
103148	兼容版本重复	
103149	时间计划uuid不允许重复	
103150	组uuid不允许重复	
103151	B3R最多支持人数是10万	
103152	CORE最多支持人数是100万	
103153	W4K面板机最多支持人数是5万	
103154	W5K面板机最多支持人数是10万	
103155	通行权限不存在	
103156	设备不在人员管控设备中	

错误码	描述	备注
103157	设备不在门禁设备中	
103158	人员布控编辑失败	
103159	人员布控添加失败	
103162	W3K面板机最多支持人数是5万	
103163	布控类型不符	
103160	组内人员数量最多支持10万	
103161	人员布控删除失败	
103162	添加失败，未指定中心设备	
103163	绑定B3R的设备通行权限最多支持64个组	
103164	非重点人员组/员工组/访客组	
103165	%s设备通行权限最多支持%s个组	
103166	%s设备最多支持人数是%s	
103167	组未进行布控	

1.5. 通行业务错误码

错误码	描述	备注
103000	读取文件出错误	
103001	imageuri为空	
103002	设备未绑定	
103003	验证码不存在	
103004	主题不存在	
103005	参数异常	
103006	迎宾设备不存在	
103007	视频流设备不存在,请刷新后重新选择	
103008	展示的视频流设备不存在,请刷新后重新选择	
103009	没有找到相应的迎宾记录	
103010	迎宾设备已绑定	
103011	绑定失败,设备已绑定	
103012	设备不存在	
103012	设备不存在	
103013	区域不存在	

1.6. 考勤业务错误码

错误码	描述	备注
104401	请选择要添加的考勤设备	
104402	不允许跨区域添加考勤设备	
104403	不允许批量添加同一个设备	
104404	删除设备失败，参数为空	
104405	出勤调整失败，非考勤组人员	
104406	获取人员信息失败	
104407	考勤记录不存在	
104408	考勤时段不存在	
104409	人员信息不存在	
104410	人员信息不存在	
104411	考勤组不存在	
104412	考勤组名字重复	
104413	未设置考勤时段	
104414	考勤时段编号设置错误	
104415	最多允许设置3个考勤时段	
104416	跨天时段不能超过次日12时	
104417	考勤时段累计不能超过24小时	
104418	考勤时段间隔必须大于1分钟	
104419	考勤时段的起始时间不能晚于结束时间	
104420	考勤时段不支持重合	
104421	最多可设置32个特殊日期	
104022	特殊日期不能重复	
104023	特殊日期不可选择历史日期	
104024	不存在跨天时段，不允许设置跨天统计	
104025	只允许一个时间段跨天	
104026	考勤周计划重复	
104027	设备不存在	
104028	添加出勤调整记录失败，考勤记录ID为空	
104029	出勤调整失败，不符合调整规定	
104030	开始时间不能大于结束时间	
104031	开始时间不能等于结束时间	
104032	起止时间跨度不能超过3个月	
104033	权限不足	

错误码	描述	备注
104034	参数无效	
104035	参数有重复，请仔细核对！	
104036	重复添加！	
104037	本月5号前只能调整上一个自然月及其以后的考勤记录	
104038	本月5号以后只能调整本月及本月以后的考勤记录	
104039	不支持服务器类设备的添加	
104040	名字包含非法字符	
104041	导出文件异常	
104042	考勤时间段为空	
104043	考勤时间段长度大于三个月	
104044	考勤周期枚举类型设置错误，请检查！	
104045	非门禁点设备！	

1.7. 通行记录错误码

错误码	描述	备注
105001	批量删除失败	
105002	导出记录最长记录为3个月	

1.8. 安防人员业务错误码

121001	获取高频人员列表失败	
121002	高频处理出现错误	
121003	人员入ES索引错误	
121004	查询人员详情出现错误	
121005	查询人员列表出现错误	
121006	人员聚类任务名称重复添加	
121007	人员聚类任务添加达到上限	
121008	人员聚类任务添加错误	
121009	查询高频人员详情出现错误	
121010	调用中台接口错	
121011	查询轨迹信息错误	
121012	轨迹列表获取失败	
121013	过人库Track错误	
121014	深度实时检索并发上限3次，该操作已超上限，请稍后再试	
121015	导出超出上限1000条	
121016	导出条目出现异常	
121017	导出操作出现异常	
121018	导出数量为空	
121019	获取列表失败	
121020	识别记录推送MQ失败	
121021	获取图片失败	
121022	本地缓冲操作错误	
121023	PersonId和ProfileId不能同时为空	
121024	获取人次数失败	
121025	以图搜图查看失败	
121026	图片URI和ProfileId不能同时为空	
121027	获取档案信息失败	
121028	PersonId和ProfileId不能同时存在	
121029	以图搜图查看记录详情错误	
121030	添加一人一档强id错误	
121031	添加人员底库失败，人脸过多	
121032	添加人员底库失败，未检测到人脸	
121033	添加人员底库失败，未检测到人脸	

121034	添加人员底库失败，人脸质量过低	
121035	添加人员底库失败	
121036	删除底库人员失败	
121037	人员服务内部调用错	
121038	处理时间计划处理失败	
121039	识别pipeline错误	
121040	开始时间大于结束时间	
121041	入参pageNum或者pageSize不可小于等于0	
121042	personUuid和profileUuid不能同时为空,也不能同时存在,请检查参数	
121043	图片质量不合格，可能是由于人脸过多	
121044	flag如果参数传递 参数只能为true或者false	
121045	字符串长度超出	
121046	聚类任务不存在	
121047	档案不存在	
121048	设备不存在	
121049	聚类任务不存在	
121050	聚类任务未完成	
121051	图片URL换取EFS--Uri失败	
121052	检索失败，未指定中心设备	
121053	检索失败，中心设备离线	
121054	uri或特征不存在，不能搜索1:N	
121055	请求次数过多，请稍后再试	
121056	查询参数有误	
121057	查询数据报表请求参数不全	
121100	人脸太多	
121101	未检测到人脸，可能是图片质量太差	
121102	图片解码错误	
121103	图片质量太差	
121104	检索失败，未指定CORE服务器作为中心设备	
121057	调用CORE接口错误	
121200	已经执行成功，无需重复执行	
121201	不存在布控的设备	
121202	操作失败，设备升级还未成功	

121203	历史陌生人记录为空	
121204	数据还未清洗	
121205	操作失败，数据清洗还未成功	

1.9. 结构化业务错误码

错误码	描述	备注
122001	已布控	
122002	结构化布控规则不存在	
122003	结构化新增布控请求失败	
122004	结构化更新布控请求失败	
122005	结构化布控详情请求失败	
122006	结构化删除布控请求失败	
122007	结构化布控列表请求失败	
122008	该区域内已录入此车牌	
122009	车辆不存在	
122010	车辆基础信息新增失败	
122011	车辆基础信息更新失败	
122012	车辆信息列表获取失败	
122013	车辆详情信息获取失败	
122014	车辆详情信息删除失败	
122015	布控AIOT绑定失败	
122016	结构化布控AIOT解绑失败	
122017	结构化布控AIOT更新失败	
122018	读取导入文件出错误	
122019	车牌号码不能为空	
122020	车牌号码必须在1-16个字符内且只能包含数字、中英文	
122021	车牌颜色8个字符内仅支持汉字	
122022	车辆颜色8个字符内仅支持汉字	
122023	车辆信息描述支持0-255个中英文字符	
122024	电话支持6-18位数字	
122025	唯一标识最大16位数字	
122026	该区域内唯一标识已存在	
122027	批量导入车辆信息数据转换异常	
122028	未获取到有效车辆信息	
122029	获取车辆识别记录详情异常	
122030	获取车辆识别记录列表异常	
122031	获取车辆轨迹异常	
122032	车辆列表导出异常	

错误码	描述	备注
122033	excel生成异常	
122034	超出导出上限，最多支持5000条数据导出	
122035	记录导出异常	
122036	车辆轨迹信息导出异常	
122037	枚举转换异常	
122038	车辆记录信息导出异常	
122039	该区域已存在此规则名称	
120040	车辆记录不存在	
120041	时间计划不存在	
120042	感知设备绑定关系不存在	
120043	车辆事件消息下发失败	
120044	该区域已存在此手机号	
120045	车辆类型枚举不存在	
120046	车身颜色枚举不存在	
120047	车牌颜色枚举不存在	
120048	车辆品牌枚举不存在	
120049	功能用车枚举不存在	
120050	专项作业枚举不存在	
120051	车主身份枚举不存在	
120052	车辆身份枚举不存在	
120053	开始时间不能大于结束时间	
120054	分页参数必须大于0	
120055	车牌类型不存在	
120056	非结构化设备	
120057	结构化设备不存在	
120058	人员不存在	
120059	布控类型不存在	
120060	枚举中方法不存在	
120061	超过系统单次导入支持上限10W条，请重新导入	

1.10. 警戒业务错误码

错误码	描述	备注
123001	解析告警消息失败	
123002	消息格式验证失败	
123003	处理告警消息失败	
123101	获取设备布控数据失败	
123102	获取布控列表失败	
123103	区域警戒新增布控失败	
123104	区域警戒删除布控失败	
123105	区域警戒更新布控失败	
123106	区域警戒指定布控不存在	
123107	区域警戒指定布控已存在	
123108	区域警戒指定布控不唯一	
123109	设备状态为离线，无法修改区域警戒布控	
122041	时间计划不存在	
123201	警戒记录查询失败	
123202	警戒记录导出失败	
123203	警戒记录详情导出失败	
123204	警戒导出失败	
123205	警戒进出人数统计失败	
123301	获取设备信息失败	
123302	开始时间不能大于结束时间	
124303	分页参数必须大于0	
124304	警戒设备不存在	
124305	非警戒设备	
124306	警戒类型不存在,请确认参数是否正确	
123301	获取设备信息失败	

1.11. 事件业务错误码

错误码	描述	备注
106001	事件规则不存在	
106002	参数错误:%s	会显示具体参数错误内容
106003	事件记录%s不存在	会指明具体不存在的记录标志
106004	事件记录%s已经处理过了	
106005	服务器信息错误%s	已经废弃
106006	创建文件目录失败	
106007	导出Excel文件失败	
106008	导出事件记录条目异常	
106009	导出任务不存在，任务ID为%s	
106010	缓存第三方订阅消息消息失败,消息trackID = %s	
106011	删除历史记录失败	
106012	事件规则名为空	
106013	事件规则名只能包含中文、数字、字母	
106014	事件规则名长度错误	
106015	事件规则名称冲突	
106016	更新任务状态异常	

- 1. 人员
 - 1.1. 人员类别
 - 1.2. 性别
 - 1.3. 访客类型
 - 1.4. 访客状态
 - 1.5. 人员导入导出业务类型
 - 1.6. 人员导出类型
 - 1.7. 删除区域状态
 - 1.8. 公司ID
 - 1.9. 校验人员开关
 - 1.10. 公司的父级机构id
 - 1.11. 人员通知
 - 1.12. 机构类型
 - 1.13. 人脸入库登记
 - 1.14. 区域状态
 - 1.15. 图片上传类型枚举
 - 1.16. 人员操作类型
 - 1.17. 区域过滤类型
- 2. 组管理
 - 2.1. 组类型
 - 2.2. 区域过滤类型
- 3. 设备
 - 3.1. 设备模式
 - 3.2. 服务器模式
 - 3.3. 设备类型
 - 3.4. 设备型号
 - 3.5. 协议类型
 - 3.6. 是否支持主动模式
- 4. 门禁点管理
 - 4.1. 绑定类型
 - 4.2. 人员照片显示
 - 4.3. 报警输入
 - 4.4. 报警输出
 - 4.5. 通行认证方式
 - 4.6. 使用模式
- 5. 聚类
 - 5.1. 聚类任务类型
 - 5.2. 聚类任务状态
- 6. 考勤
 - 6.1. 打卡状态
 - 6.2. 考勤结果
 - 6.3. 出勤调整类型
 - 6.4. 考勤状态
 - 6.5. 考勤时段编号
 - 6.6. 跨天打卡考勤统计标识
- 7. 结构化&车辆
 - 7.1. 头盔状态
 - 7.2. 骑行状态

- 7.3. 包的类型
- 7.4. 各种颜色(帽子、上衣服饰、下衣服饰)
- 7.5. 帽子状态
- 7.6. 下身服饰
- 7.7. 上衣服饰
- 7.8. 鼻子有无遮挡
- 7.9. 头部状态
- 7.10. 胡子状态
- 7.11. 眼部状态
- 7.12. 嘴部状态
- 7.13. 年龄
- 7.14. 性别
- 7.15. 结构化类别
- 7.16. 车辆类型
- 7.17. 车身颜色
- 7.18. 车辆品牌
- 7.19. 车辆专项作业
- 7.20. 车牌类型
- 7.21. 车牌颜色
- 7.22. 车辆身份
- 7.23. 车辆功能用车
- 7.24. 车牌渲染风格
- 7.25. 车牌基础类型
- 7.26. 基础车辆类型
- 7.27. 人员类型
- 7.28. 车辆&结构化布控类型
- 7.29. 口罩状态
- 7.30. 眼镜状态
- 7.31. 人体跌倒状态
- 7.32. 人体奔跑状态
- 7.33. 人体抽烟状态
- 7.34. 人体看手机状态
- 7.35. 人体打电话状态
- 7.36. 打伞状态
- 8. 警戒
 - 8.1. 警戒类型
- 9. 事件
 - 9.1. 事件类型
 - 9.2. 事件等级
 - 9.3. 事件状态
 - 9.4. 订阅的业务类型
 - 9.5. 识别记录

1. 人员

1.1. 人员类别

枚举值	枚举描述	备注
1	员工	
2	访客	
3	重点人员	

1.2. 性别

枚举值	枚举描述	备注
0	未知	
1	男	
2	女	

1.3. 访客类型

枚举值	枚举描述	备注
1	普通访客	
2	VIP访客	

1.4. 访客状态

枚举值	枚举描述	备注
true	有效	
false	无效	

1.5. 人员导入导出业务类型

枚举值	枚举描述	备注
1	导入	
2	导出	
3	上传人员照片	

1.6. 人员导出类型

枚举值	枚举描述	备注
1	员工导出	
2	访客导出	
3	重点人员导出	

1.7. 删除区域状态

枚举值	枚举描述	备注
1	开始异步删除区域，可在任务中心查看结果，切勿重复删除	同步删除
2	删除成功	异步删除

1.8. 公司ID

枚举值	枚举描述	备注
1	nacos	
2	auth_center	

1.9. 校验人员开关

枚举值	枚举描述	备注
true	open	
false	close	

1.10. 公司的父级机构id

枚举值	枚举描述	备注
0L	company	公司的父级机构ID

1.11. 人员通知

枚举值	枚举描述	备注
1	增加	
2	删除	

1.12. 机构类型

枚举值	枚举描述	备注
1	普通组织或机构	
2	公司	

1.13. 人脸入库登记

枚举值	枚举描述	备注
1	低	
2	中	
3	高	
4	自定义	

1.14. 区域状态

枚举值	枚举描述	备注
1	已删除	
2	正常	

1.15. 图片上传类型枚举

枚举值	枚举描述	备注
1	登录页logo	
2	导航栏logo	
3	登录页背景图	

1.16. 人员操作类型

枚举值	枚举描述	备注
1	添加	
2	修改	
3	删除	

1.17. 区域过滤类型

枚举值	枚举描述	备注
1	当前区域	
2	当前区域及下级区域	
3	指定的区域	

2. 组管理

2.1. 组类型

枚举值	枚举描述	备注
1	员工组	
2	访客组	
3	重点人员组	

2.2. 区域过滤类型

枚举值	枚举描述	备注
1	当前区域	
2	当前区域及下级区域	
3	指定的区域	

3. 设备

3.1. 设备模式

枚举值	枚举描述	备注
1	人脸识别模式	
2	结构化模式	
3	警戒模式	
4	人脸抓拍模式	
5	混合模式	B3R类型设备

3.2. 服务器模式

枚举值	枚举描述	备注
2	结构化模式	
3	警戒模式	
4	人脸抓拍模式	
6	人脸识别标准模式	
7	人脸识别高性能模式	

3.3. 设备类型

枚举值	枚举描述	备注
1	智能摄像头	设备列表： MegEye-C3S-123 MegEye-C4H-141 MegEye-C4H-241 MegEye-C4N-888 MegEye-C4S-222 MegEye-C4S-322 MegEye-C4R-122 MegEye-C4R-222 MegEye-C4R-322 海康摄像头 大华摄像头 宇视摄像头 科达摄像头 华为摄像头
2	智能面板机	设备列表： MegEye-W5K-I8 MegEye-W4K-E7 MegEye-W4K-I51 MegEye-W4K-I52 MegEye-W3K-45S MegEye-W4K-I31 MegEye-W5K-EA MegEye-W5K-EA-D MegEye-W5K-EA-T MegEye-W5K-EA-DT MegEye-W4K-E81 SmartEye-W4K-E7
3	智能服务器	设备列表： CORE MegBox-B3R-412 MegBox-B3R-1012 MegBox-B3R-1012(H1) MegCube-B4H16-311
4	网络摄像机	设备列表： MegEye-C3S-123 MegEye-C4H-141 MegEye-C4H-241 MegEye-C4N-888 MegEye-C4S-222 MegEye-C4S-322 MegEye-C4R-122 MegEye-C4R-222 MegEye-C4R-322 IPC
5	存储服务器	设备列表： MegNVR-N4K-001 MegNVR-N4M-001 MegNVR-N8K-281 MegNVR-N8M-281

3.4. 设备型号

枚举值	枚举描述	备注
1	MegEye-W5K-I8	
2	MegEye-W4K-E7	
3	MegEye-W4K-I51	
4	MegEye-C3S-123	
5	MegEye-C4H-141	
6	MegEye-C4H-241	
7	MegEye-C4N-888	
8	MegEye-C4S-222	
9	MegEye-C4S-322	
10	MegEye-C4R-122	
11	MegEye-C4R-222	
12	MegEye-C4R-322	
18	CORE	
19	IPC	
22	MegBox-B3R-412	
23	MegBox-B3R-1012	
24	MegNVR-N4K-001	
25	MegNVR-N4M-001	
26	MegNVR-N8K-281	
27	MegNVR-N8M-281	
28	MegNVR-N8P-281	
29	海康摄像头	
30	大华摄像头	
31	宇视摄像头	
32	科达摄像头	
33	华为摄像头	
34	MegEye-W4K-I52	
35	MegEye-W3K-45S	
36	MegEye-W4K-I31	
38	MegBox-B3R-1012(H1)	
39	MegCube-B4H16-311	
40	MegEye-W5K-EA	
41	MegEye-W5K-EA-D	

枚举值	枚举描述	备注
42	MegEye-W5K-EA-T	
43	MegEye-W5K-EA-DT	
44	MegEye-W4K-E81	
45	SmartEye-W4K-E7	

3.5. 协议类型

枚举值	枚举描述	备注
1	rtsp	添加设备时需传递rtsp流地址
2	onvif	添加设备时无需传递rtsp流地址，端口与rtsp不同

3.6. 是否支持主动模式

枚举值	枚举描述	备注
1	非主动模式	
2	主动模式	设备接入是必须传递sn枚举值，无须传递IP

4. 门禁点管理

4.1. 绑定类型

枚举值	枚举描述	备注
1	B3R	
2	CORE	
3	面板机	

4.2. 人员照片显示

枚举值	枚举描述	备注
1	底库照片	
2	抓拍照片	
3	不显示照片	

4.3. 报警输入

枚举值	枚举描述	备注
0	不启用	
1	消防信号	
2	报警信号	

4.4. 报警输出

枚举值	枚举描述	备注
1	黑名单报警联动报警	
2	门磁超时联动报警	
3	报警输入1联动报警	
4	报警输入2联动报警	

4.5. 通行认证方式

枚举值	枚举描述	备注
0	人脸	
1	人脸或刷卡	
2	人脸及刷卡	
3	人脸及密码	

4.6. 使用模式

枚举值	枚举描述	备注
0	单人	
1	多人	

5. 聚类

5.1. 聚类任务类型

枚举值	枚举描述	备注
0	高频陌生人	
1	人员聚类任务	

5.2. 聚类任务状态

枚举值	枚举描述	备注
0	处理中	
1	已完成	
2	已删除	

6. 考勤

6.1. 打卡状态

枚举值	枚举描述	备注
0	不打卡	考勤组-考勤规则，设置特殊日期时使用。
1	打卡	考勤组-考勤规则，设置特殊日期时使用。

6.2. 考勤结果

枚举值	枚举描述	备注
0	--	--"表示暂无结果
1	正常	
2	缺勤	
3	迟到	
4	早退	
5	迟到早退	
6	漏打卡	
7	出勤调整	

6.3. 出勤调整类型

枚举值	枚举描述	备注
1	补卡	
2	请假	
3	调休	
4	调班	

6.4. 考勤状态

枚举值	枚举描述	备注
0	"--"	"--"表示暂无结果
1	正常	
2	旷工	
3	出勤调整	
4	异常	

6.5. 考勤时段编号

枚举值	枚举描述	备注
1	考勤时段1	考勤组-考勤规则-设置考勤时段使用。
2	考勤时段2	
3	考勤时段3	最多3个考勤时段

6.6. 跨天打卡考勤统计标识

枚举值	枚举描述	备注
0	正常打卡统计	考勤打卡跨天的，跨天的考勤记录按跨天的第一天计算，示例：6号跨7号打卡，此时考勤记录的日期为6号，按6号统计考勤
1	跨天打卡统计	考勤打卡跨天时，跨天的考勤记录按跨天的第二天计算，示例：6号跨7号打卡，此时考勤记录的日期为7号，按7号统计考勤
2	已删除	

7. 结构化&车辆

7.1. 头盔状态

枚举值	枚举描述	备注
0	未知	
1	无安全帽	
2	有安全帽	

7.2. 骑行状态

枚举值	枚举描述	备注
0	未知	
1	未骑行	
2	骑行	

7.3. 包的类型

枚举值	枚举描述	备注
0	未知	
1	未携带包	
2	双肩包	
3	手提包	
4	单肩包	
5	拉杆箱	
6	腰包	
7	有背包，类型未知	

7.4. 各种颜色(帽子、上衣服饰、下衣服饰)

枚举值	枚举描述	备注
0	未知	
1	白色	
2	灰色	
3	棕色	
4	红色	
5	蓝色	
6	黄色	
7	绿色	
8	粉色	
9	橙色	
10	青色	
11	紫色	
12	浅蓝色	
13	深蓝色	
14	黑色	
15	彩色	

7.5. 帽子状态

枚举值	枚举描述	备注
0	未知	
1	无帽子	
2	有帽子	

7.6. 下身服饰

枚举值	枚举描述	备注
0	未知	
1	长裤	
2	短裤	
3	长裙	
4	短裙	

7.7. 上衣服饰

枚举值	枚举描述	备注
0	未知	
1	长袖	
2	短袖	

7.8. 鼻子有无遮挡

枚举值	枚举描述	备注
0	未知	
1	未遮挡	
2	有遮挡	

7.9. 头部状态

枚举值	枚举描述	备注
0	未知	
1	长发	
2	短发	
3	少量头发（秃顶）	
4	光头	

7.10. 胡子状态

枚举值	枚举描述	备注
0	未知	
1	无胡子	
2	有胡子	

7.11. 眼部状态

枚举值	枚举描述	备注
0	未知	
1	睁眼	
2	闭眼	

7.12. 嘴部状态

枚举值	枚举描述	备注
0	未知	
1	闭嘴	
2	张嘴	

7.13. 年龄

枚举值	枚举描述	备注
0	未知	
1	儿童	
2	少年	
3	青年	
4	中年	
5	中老年	
6	老年	

7.14. 性别

枚举值	枚举描述	备注
0	未知	
1	男性	
2	女性	

7.15. 结构化类别

枚举值	枚举描述	备注
1	人脸	
2	人体	
3	机动车	
4	非机动车	

7.16. 车辆类型

枚举值	枚举描述	备注
0	未知	
1	两厢轿车	
2	轿车	
3	轿跑	
4	小型轿车	
5	微型轿车	
6	MPV	
7	SUV	
8	大型客车	
9	中型客车	
10	面包车	
11	微型面包车	
12	货车	
13	中型货车	
14	油罐车	
15	吊车	
16	渣土车	
17	小货车	
18	皮卡	
19	微卡	
20	两轮车	
21	三轮车	
22	行人	
29	挂车	
30	小型车(大范围)	
31	大型车(大范围)	
100	自行车	
101	电动车	
102	电动车	

7.17. 车身颜色

枚举值	枚举描述	备注
0	白色	
1	红色	
2	黄色	
5	绿色	
7	紫色	
8	粉色	
9	黑色	
10	橙色	
11	银色	
12	香槟色	
13	灰色	
15	蓝色	
21	青色	
29	棕色	
32	深蓝	
40	彩色	
99	未知	

7.18. 车辆品牌

枚举值	枚举描述	备注
0	其它	
1	大众	
2	奥迪	
3	丰田	
4	本田	
5	奔驰	
6	雪佛兰	
7	奇瑞	
8	别克	
9	长城	
10	现代	
11	尼桑	
12	福特	
13	宝马	
14	雪铁龙	
15	起亚	
16	铃木	
17	马自达	
18	比亚迪	
20	标致	
21	长安	
22	雷克萨斯	
23	中华	
24	斯柯达	
25	海马	
26	夏利	
27	五菱	
28	东风	
29	哈飞	
30	一汽	
31	宝骏	
32	帝豪(吉利)	
33	MG名爵	

枚举值	枚举描述	备注
34	东南	
35	皇冠(丰田)	
36	金杯	
37	三菱	
38	荣威	
39	吉利	
40	英伦(吉利)	
41	吉利全球鹰(吉利)	
42	哈飞赛豹	
43	长丰	
44	北汽威旺	
45	北京汽车	
46	新凯	
47	吉奥汽车	
48	玛莎拉蒂	
49	罗孚	
50	奥斯汀	
52	道奇	
55	捷豹	
57	阿尔法	
58	兰博基尼	
59	布加迪	
60	林肯	
61	法拉利汽车	
62	昌河	
63	菲亚特	
64	福田	
65	讴歌	
66	莲花汽车	
67	华普汽车	
68	红旗	
69	瑞麟	
70	一汽奔腾(一汽)	

枚举值	枚举描述	备注
71	威麟汽车	
72	众泰	
73	力帆	
74	JEEP(吉普)	
75	中兴	
76	开瑞	
77	路虎	
78	迈巴赫	
79	雷诺	
80	欧宝	
81	野马	
82	吉普	
83	依维柯	
84	英菲尼迪	
85	斯巴鲁	
86	阿斯顿·马丁	
87	安凯客车	
88	保时捷	
89	宾利	
90	福迪	
91	福建奔驰	
92	吉姆斯	
93	观致	
94	广汽	
95	双龙	
96	海格	
97	悍马	
98	华泰	
99	黄海	
100	九龙客车	
101	理念	
102	奔驰SMART	
103	陆风	

枚举值	枚举描述	备注
104	纳智捷	
105	欧朗	
106	启辰	
107	哈弗HAVAL	
108	豪沃	
109	上海汇众(上汽集团)	
110	金龙	
112	江淮	
113	JMC(江铃)	
114	凯迪拉克	
116	宝马MINI	
117	陕汽	
118	少林	
119	沃尔沃	
120	五十铃(庆铃)	
121	跃进	
122	宇通	
123	中通	
125	申沃	
126	扬子江	
128	北奔重卡	
139	红岩汽车(上汽依维柯红岩)	
130	劳斯莱斯	
131	特斯拉	
132	腾势	
134	英致	
135	西雅特	
136	亚星客车	
137	大宇客车	
138	长安谛艾仕	
139	凯翼	
140	上汽大通	
141	本田思铭	

枚举值	枚举描述	备注
142	凯马	

7.19. 车辆专项作业

枚举值	枚举描述	备注
0	未知	
7	混凝土搅拌机	
8	消防车	

7.20. 车牌类型

枚举值	枚举描述	备注
0	未知	
1	普通小车、私家车牌（蓝色、黑色）	
2	普通大车、卡车牌（黄色）	
3	警车牌（白色）	
4	武警车牌	
5	港澳车牌	
6	双层黄牌	
7	军牌	
8	领使馆牌	
9	电动自行车车牌	
10	双层军牌	
11	“学”牌	
12	摩托车牌	
13	临时车牌	
14	农用车牌 拖拉机	
15	挂车车牌	
16	新能源牌	
17	运输车	
18	商用车	
19	教练车	
20	政府车	
21	出租车	

7.21. 车牌颜色

枚举值	枚举描述	备注
0	未知	
1	蓝色	
2	黄色	
3	白色	
4	黑色	
5	绿色	
6	渐变绿底黑字	
7	黄绿双拼底黑字	

7.22. 车辆身份

枚举值	枚举描述	备注
1	重点车辆	
2	员工车辆	
3	访客车辆	
4	陌生车辆	
5	已知车辆	

7.23. 车辆功能用车

枚举值	枚举描述	备注
0	未知	
1	挖掘机	
2	渣土车	
3	校车	
4	救护车	
5	警车	
6	集装箱车	

7.24. 车牌渲染风格

枚举值	枚举描述	备注
0	未知	
1	蓝色	
2	黄色	
3	白色	
4	黑色	
5	新能源	

7.25. 车牌基础类型

枚举值	枚举描述	备注
1	标准民用车与军用车	
2	006车	
3	警车	
4	军用车双行尾牌	
5	使馆车	
6	农用车	
7	摩托车	
8	新能源车	

7.26. 基础车辆类型

枚举值	枚举描述	备注
1	小型车	
2	大型车	
3	摩托车	
4	其他车	

7.27. 人员类型

枚举值	枚举描述	备注
1	员工	
2	访客	
3	重点人员	

7.28. 车辆&结构化布控类型

枚举值	枚举描述	备注
1	结构化布控	
2	车辆布控	

7.29. 口罩状态

枚举值	枚举描述	备注
1	未知	
2	无口罩	
3	有口罩	

7.30. 眼镜状态

枚举值	枚举描述	备注
0	未知	
1	无眼镜	
2	有眼镜	

7.31. 人体跌倒状态

枚举值	枚举描述	备注
0	未知	
1	未跌倒	
2	跌倒	

7.32. 人体奔跑状态

枚举值	枚举描述	备注
0	未知	
1	未奔跑	
2	奔跑	

7.33. 人体抽烟状态

枚举值	枚举描述	备注
0	未知	
1	未抽烟	
2	抽烟	

7.34. 人体看手机状态

枚举值	枚举描述	备注
0	未知	
1	未看手机	
2	看手机	

7.35. 人体打电话状态

枚举值	枚举描述	备注
0	未知	
1	未打电话	
2	打电话	

7.36. 打伞状态

枚举值	枚举描述	备注
0	未知	
1	不打伞	
2	打伞	

8. 警戒

8.1. 警戒类型

枚举值	枚举描述	备注
1	人员越界	
2	人员入侵	
3	车辆越界	
4	车辆禁停	
5	人员越界-翻墙检测	
6	人员徘徊	
7	机动车离开	
8	非机动车离开	
10	人员值岗/离岗-离岗	
11	人员值岗/离岗-超员	
12	人员值岗/离岗-少员	
13	人员奔跑	
14	摔倒检测	
15	人员扭打	
16	抽烟检测	
17	看手机	
18	接打电话	
19	人员聚众	
20	防碰撞预警	
21	车辆逆行	
22	睡岗检测	
31	机动车越界	
32	非机动车越界	
41	机动车禁停	
42	非机动车禁停	

9. 事件

9.1. 事件类型

枚举值	枚举描述	备注
101	重点人员	
102	全局陌生人员	
103	高频重点人员	
104	员工	
105	访客	
106	通道陌生人	
201	重点车辆	
202	陌生车辆	
203	已知车辆	
204	访客车辆	
205	员工车辆	
301	人员入侵	
302	人员越界	
303	车辆越界	
304	车辆禁停	
305	人员越界-翻墙检测	
306	机动车越界	
307	非机动车越界	
308	机动车禁停	
309	非机动车禁停	
310	机动车离开	
312	人员徘徊	
313	人员值岗/离岗	
314	人员奔跑	
315	人员摔倒	
316	人员扭打	
317	抽烟	
318	看手机	
319	接打电话	
320	人员聚众	
321	车辆逆行	
322	防碰撞预警	
323	人员值岗/离岗-超员	

枚举值	枚举描述	备注
324	人员值岗/离岗-少员	
325	睡岗检测	
4002	设备离线	
401	设备拆除	
402	门磁超时	
16	密码破解	
404	消防告警	
405	存储空间告警	
406	测温模块失效	
407	人员底库超限	
501	重点人员出入事件	
502	高温事件	
503	通道陌生人出入事件	
504	健康码检测未通过出入事件	

9.2. 事件等级

枚举值	枚举描述	备注
1	高	
2	中	
3	低	

9.3. 事件状态

枚举值	枚举描述	备注
1	已处理	
0	未处理	

9.4. 订阅的业务类型

枚举值	枚举描述	备注
1	安防识别记录	
2	车辆识别记录	
3	警戒记录	
5	通行记录	
12	结构化全属性识别记录	

9.5. 识别记录

枚举值	枚举描述	备注
1	员工	
2	访客	
3	重点人员	
4	全局陌生人	
5	通道陌生人	

- 1. 通行业务
- 2. 考勤业务
- 3. 人员布控
- 4. 人员聚类
- 5. 人脸比对
 - 5.1. 1: 1
 - 5.2. 1: N
- 6. 结构化布控
- 7. 车辆布控
- 8. 警戒布控
- 9. 事件记录回调
- 10. 通行记录回调
- 11. 考勤记录回调
- 12. 业务数据推送

1. 通行业务

序号	步骤	接口	备注
1	添加边设备（接入算力服务器）	/v1/api/device/add	设备添加
2	添加端设备，添加端设备有以下两种类型：		
	视频流、抓拍机的添加	设备添加： /v1/api/device/add	
	面板机的接入	面板机注册至盘古系统 通过发现设备查询： /v1/api/device/queryReportDevice 设备导入： /v1/api/device/importDevice	
3	绑定端设备到边设备	/v1/api/device/bind	设备绑定
4	添加门禁、人员管控设备	/v1/api/device/control/add	将设备添加为门禁设备，布控前置操作
5	添加通行权限	/v1/api/pass/authority/add	进行通行布控
6	通行记录查询	/v1/api/pass_record/list	分页查询通行记录

2. 考勤业务

序号	步骤	接口	备注
1	查询门禁点设备	/v1/api/door/list	获取门禁点设备信息。
	添加考勤设备	/v1/api/attendance/device/batchAdd	设备不一定被布控，需要调用布控接口确认。
	删除考勤设备	/v1/api/attendance/device/batchDelete	
2	添加考勤组	/v1/api/attendance/group/add	添加一个考勤组，设置考勤规则，批量绑定考勤员工。
	查询未添加到考勤组的员工信息	/v1/api/person/list	分页获取员工信息。
	查询已绑定考勤组的员工信息	/v1/api/attendance/group/person/list	分页获取绑定考勤组的员工信息。
	更新考勤组	/v1/api/attendance/group/update	同添加考勤组接口。

序号	步骤	接口	备注
	删除考勤组	/v1/api/attendance/group/batchDelete	删除考勤组会同步删除考勤组中的人。
3	查询考勤记录	/v1/api/attendance/record/list	
	查询考勤通行详情	/v1/api/attendance/record/detail/list	获取考勤打卡时间区间的通行记录。
	查询员工月考勤记录	/v1/api/attendance/record/personByMonth/list	通过考勤统计记录获取考勤记录，时间跨度为一个月。
	查询员工月考勤记录（日历模式）	/v1/api/attendance/record/calendar	
	考勤记录调整	/v1/api/attendance/record/adjust	对指定的异常考勤记录进行调整。

序号	步骤	接口	备注
4	查询考勤统计记录	/v1/api/attendance/statistic/list	按时间段、天的时间维度分别统计员工的出勤情况。
5	添加出勤调整	/v1/api/attendance/adjust/add	《查询已绑定考勤组的员工信息》- 获取考勤调整人员信息
	查询出勤调整记录	/v1/api/attendance/adjust/list	分页查询获取。
	更新出勤调整记录	/v1/api/attendance/adjust/update	修改指定的出勤调整记录。

3. 人员布控

序号	步骤	接口	备注
1	添加边设备（接入算力服务器）	/v1/api/device/add	设备添加
2	添加端设备，添加端设备有以下两种类型：		
	视频流、抓拍机的添加	设备添加： /v1/api/device/add	
	面板机的接入	面板机注册至盘古系统 通过发现设备查询： /v1/api/device/queryReportDevice 设备导入： /v1/api/device/importDevice	
3	绑定端设备到边设备	/v1/api/device/bind	设备绑定
4	添加人员管控设备	/v1/api/device/control/add	将设备添加为人员管控设备，布控前置操作
5	添加人员布控规则	/v1/api/pass/rule/addBatch	进行人员布控
6	人员布控记录查询	/v1/api/person//record/list	分页查询人员布控记录

4. 人员聚类

序号	步骤	接口	备注
1	添加边设备 (接入算力服务器)	/v1/api/device/add	设备添加
2	添加端设备， 添加端设备有以下两种类型：		
	视频流、抓拍机的添加	设备添加： /v1/api/device/add	
	面板机的接入	发现设备查询： /v1/api/device/queryReportDevice 设备导入： /v1/api/device/importDevice	
3	绑定端设备到边设备	/v1/api/device/bind	设备绑定
4	添加人员管控设备	/v1/api/device/control/add	添加人员管控设备
5	新建陌生人员规则	/v1/api/pass/rule/addBatch	添加规则
6	运行.... 产生记录.....		
7	添加陌生人员聚类任务	/v1/api/cluster/task/add	人员聚类任务添加
8	查看聚类任务状态	/v1/api/cluster/task/status	当任务状态是已完成之后，方可通过后面的接口查看聚类的结果
9	查看某个任务的结果	/v1/api/cluster/task/detail/query	人员聚类任务详情查看
10	查看聚类结果中某个档案的详情	/v1/api/cluster/person/detail/query	人员聚类人员详情查看

5. 人脸比对

5.1. 1: 1

序号	步骤	接口	备注
1	页面添加算力设备（接入算力服务器）并且再页面指定算力中心	/v1/api/device/add	设备添加
2	上传人员图片，不卡质量	/v1/api/person/config/uploadImage	上传图片获取URI
3	接口调用	/v1/api/photo/searchOneToOnePhoto	查看比分

5.2. 1: N

序号	步骤	接口	备注
1	页面添加算力设备（接入算力服务器）并且再页面指定算力中心	/v1/api/device/add	设备添加
2	添加人员	/v1/api/person/batchAdd	上传人员到底库
3	获取需要检索的图片URI	/v1/api/person/config/uploadImage	上传图片获取URI
4	传递阈值、选择检索的底库类型、【在lite场景下不支持陌生人的底库类型搜索】	/v1/api/photo/search	获取到当前上传图片与底库中比分最大的20人

6. 结构化布控

步骤	接口	备注
添加边设备（接入算力服务器）	设备添加： <code>/v1/api/device/add</code>	采用视频流做结构化才需要
添加端设备（只能用结构化相机或者视频流相机）	设备添加： <code>/v1/api/device/add</code>	
绑定端设备到边设备	设备绑定： <code>/v1/api/device/bind</code>	视频流需要和算力绑定，结构化相机不需要
结构化布控	结构化&车辆管控布控新增： <code>/v1/api/vehicle/rule/add</code>	
运行.... 产生记录.....		
结构化数据检索	这里检索的结构比较多，具体见“结构化检索模块”部分	

7. 车辆布控

步骤	接口	备注
添加车辆信息	车辆基础信息批量新增： <code>/v1/api/vehicle/batchAdd</code>	
添加边设备（接入算力服务器）	设备添加： <code>/v1/api/device/add</code>	采用视频流做结构化才需要
添加端设备（只能用结构化相机或者视频流相机）	设备添加： <code>/v1/api/device/add</code>	
绑定端设备到边设备	设备绑定： <code>/v1/api/device/bind</code>	视频流需要和算力绑定，结构化相机不需要
车辆布控	结构化&车辆管控布控新增： <code>/v1/api/vehicle/rule/add</code>	
运行.... 产生记录.....		
查看车辆识别记录	车辆识别列表： <code>/v1/api/vehicle/record/list</code>	
车辆记录详情	车辆识别记录详情： <code>/v1/api/vehicle/record/query</code>	
查看车辆轨迹	车辆轨迹查询： <code>/v1/api/vehicle/record/track/query</code>	

8. 警戒布控

步骤	接口	备注
添加边设备（接入算力服务器）	设备添加： <code>/v1/api/device/add</code>	采用视频流做警戒算法仓才需要
添加端设备（只能用警戒相机或者视频流相机）	设备添加： <code>/v1/api/device/add</code>	
绑定端设备到边设备	设备绑定： <code>/v1/api/device/bind</code>	视频流需要和算力绑定，警戒相机不需要
警戒布控（目前openapi未开放此接口，只能在盘古页面上布控）	警戒布控新增：请在盘古页面新增	目前openapi未开放此接口，只能在盘古页面上布控
运行.... 产生警戒记录.....		
警戒记录查询	<code>/v1/api/areaAlert/record/list</code>	

9. 事件记录回调

序号	步骤	备注
1	用户在【系统管理】-【功能配置】-【事件配置】上配置事件订阅地址	
2	【事件管理】-【事件联动】页面添加规则	联动反馈一栏，必须选中‘联动事件订阅’
3	事件产生	
4	订阅地址接收到数据	参看api接口“事件回调的数据结构”部分

10. 通行记录回调

序号	步骤	接口	备注
1	客户自行开发接受数据的接口		例如: http://xxx 或者 https://xxx
2	在一脸通配置接口添加识别记录回调地址	/v1/api/sys/config/saveAFaceConfig	<pre> { "identifyRecordCallback": "http://xxx" } 或者 { "identifyRecordCallback": "https://xxx" } </pre>
3	用户接受到的数据类型		参看"通行记录"->"通行记录数据结构"部分

11. 考勤记录回调

序号	步骤	接口	备注
1	客户自行开发接受数据的接口		例如: http://xxx 或者 http://xxx
2	在一脸通配置接口添加考勤回调地址	/v1/api/sys/config/saveAFaceConfig	<pre> { "attendanceRecordCall" "http://xxx" } 或者 { "attendanceRecordCall" "https://xxx" } </pre>
3	用户接受到的数据类型		参看"考勤记录管理"->"考勤回调数据结构"部分

12. 业务数据推送

序号	步骤	备注
1	用户调【订阅接口】，订阅想要的	参看api接口“业务数据订阅”部分
2	事件产生	
3	订阅地址接收到数据	参看api接口“业务数据推送的数据结构”部分



- 1. 人员管理
 - 1.1. 上传人员图片
 - 1.1.1. 基本信息
 - 1.1.2. 请求参数
 - 1.1.3. 返回数据
 - 1.1.4. HTTP请求示例
 - 1.1.5. HTTP响应示例
 - 1.2. 上传人员图片（不卡质量）
 - 1.2.1. 基本信息
 - 1.2.2. 请求参数
 - 1.2.3. 返回数据
 - 1.2.4. HTTP请求示例
 - 1.2.5. HTTP响应示例
 - 1.3. 入库质量检测
 - 1.3.1. 基本信息
 - 1.3.2. 请求参数
 - 1.3.3. 返回数据
 - 1.3.4. 质量不过关相关错误码
 - 1.3.5. HTTP请求示例
 - 1.3.6. HTTP响应示例
 - 1.4. 批量添加人员
 - 1.4.1. 基本信息
 - 1.4.2. 请求参数
 - 1.4.3. 返回数据
 - 1.4.4. HTTP请求示例
 - 1.4.5. HTTP响应示例
 - 1.5. 批量删除人员
 - 1.5.1. 基本信息
 - 1.5.2. 请求参数
 - 1.5.3. 返回数据
 - 1.5.4. HTTP请求示例
 - 1.5.5. HTTP响应示例
 - 1.6. 修改人员
 - 1.6.1. 基本信息
 - 1.6.2. 请求参数
 - 1.6.3. 返回数据
 - 1.6.4. HTTP请求示例
 - 1.6.5. HTTP响应示例
 - 1.7. 搜索人员列表
 - 1.7.1. 基本信息
 - 1.7.2. 请求参数
 - 1.7.3. 返回数据
 - 1.7.4. HTTP请求示例
 - 1.7.5. HTTP响应示例
 - 1.8. 根据uuid列表返回人员详细信息列表
 - 1.8.1. 基本信息
 - 1.8.2. 请求参数
 - 1.8.3. 返回数据

- 1.8.4. HTTP请求示例
- 1.8.5. HTTP响应示例
- 1.9. 根据uuid返回人员信息
 - 1.9.1. 基本信息
 - 1.9.2. 请求参数
 - 1.9.3. 返回数据
 - 1.9.4. HTTP请求示例
 - 1.9.5. HTTP响应示例
- 1.10. 根据员工、访客UUID获取二维码
 - 1.10.1. 基本信息
 - 1.10.2. 请求参数
 - 1.10.3. 返回数据
 - 1.10.4. HTTP请求示例
 - 1.10.5. HTTP响应示例

1. 人员管理

1.1. 上传人员图片

1.1.1. 基本信息

Path: /v1/api/person/uploadImage

Method: POST

接口描述: 上传人脸图片文件，接口内部包含了人脸质量判断，图片压缩等处理，适用于需要上传人脸的所有场景，

1.1.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	multipart/form-data	是		

Body

参数名称	参数类型	是否必须	示例	备注
file	file	是		上传人脸图片文件，接口内部包含了人脸质量判断，图片压缩等处理，适用于需要上传人脸的所有场景，最大10m,支持格式:jpg/jpeg/bmp/png/JPG/JPEG/BMP/PNG

1.1.3. 返回数据

名称	类型	是否必须	备注
uri	string	非必须	上传返回的uri

1.1.4. HTTP请求示例

请求 path

```
/v1/api/person/uploadImage
```

请求 body

```
upload file stream (OMITTED)
```

1.1.5. HTTP响应示例

响应 200

```
{
  "code": 0,
  "data": {
    "uri": "_ZzEwMF9mb3JldmVyQnVja2V0_9a00bafda96b4829a9ff6cd470b1c675"
  },
  "msg": "成功"
}
```

1.2. 上传人员图片（不卡质量）

1.2.1. 基本信息

Path: /v1/api/person/config/uploadImage

Method: POST

接口描述: 上传人员图片(使用非系统入库参数,适用于1: N图片上传等场景, 不建议入库)

1.2.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	multipart/form-data	是		

Body

参数名称	参数类型	是否必须	示例	备注
file	file	是		上传人脸图片文件，接口内部包含了人脸质量判断(按照最低标准进行判断)，图片压缩等处理，适用于需要上传人脸等场景，不建议入库使用, 最大10m,支持格式:jpg/jpeg/bmp/png/JPG/JPEG/BMP/PNG

1.2.3. 返回数据

名称	类型	是否必须	备注
uri	string	非必须	上传返回的uri

1.2.4. HTTP请求示例

请求 path

```
/v1/api/person/config/uploadImage
```

请求 body

```
upload file stream (OMITTED)
```

1.2.5. HTTP响应示例

响应 200

```
{
  "code": 0,
  "data": {
    "uri": "_ZzEwMF9mb3JldmVyQnVja2V0_9a00bafda96b4829a9ff6cd470b1c675"
  },
  "msg": "成功"
}
```

1.3. 入库质量检测

1.3.1. 基本信息

Path: /v1/api/misc/file/qualityJudge

Method: POST

接口描述: 上传人脸图片文件质量检测，接口内部包含了人脸质量判断，不做上传，只是个质量判断。

1.3.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	multipart/form-data	是		

Body

参数名称	参数类型	是否必须	示例	备注
file	file	是		上传人脸图片文件质量检测，接口内部包含了人脸质量判断，不做上传，只是个质量判断，最大10m,支持格式:jpg/jpeg/bmp/png/JPG/JPEG/BMP/PNG

1.3.3. 返回数据

OK

1.3.4. 质量不过关相关错误码

code	msg
101082	图片大小超过10MB
101066	图片格式或名称错误
101050	模糊度不通过
101052	垂直角度不通过
101053	水平角度不通过
101054	旋转角度不通过
101055	亮度不通过
101056	标准差亮度不通过
101051	人脸大小不通过
101101	面部存在遮挡

1.3.5. HTTP请求示例

请求 path

/v1/api/misc/file/qualityJudge

请求 body

```
upload file stream (OMITTED)
```

1.3.6. HTTP响应示例

响应 **200**

```
{
  "code": 0,
  "msg": "成功"
}
```

相关错误码

1.4. 批量添加人员

1.4.1. 基本信息

Path: /v1/api/person/batchAdd

Method: POST

接口描述: 一次添加多个人员

1.4.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
personList	object []	必须	用户列表
└─name	string	必须	用户名, 长度:[1,40]
└─type	integer	必须	用户类别, 1 员工 2 访客 3 重点人员
└─sex	integer	非必须	性别, 0 未知, 1 男, 2 女
└─uuid	string	非必须	uuid, 若填写需保证唯一, 最大长度32位
└─imageUri	string	非必须	用户识别照片的uri
└─code	string	非必须	员工编码, 员工可填, 若填写需保证唯一, 格式: 允许大小写英文字母, 数字, 长度:[1,32]
└─identifyNum	string	非必须	身份证号, 格式: 允许大小写英文字母, 数字, 长度:[1,32]
└─uniqueIdentify	string	非必须	唯一标识, 若填写需要保证唯一, 格式: 最大40位数字、字母和ascii码的字符
└─visitFirm	string	非必须	访客所属单位, 访客可填, 格式: 汉字, 大小写英文字母, 数字, 长度: [1,40]
└─visitStartTimeStamp	integer	非必须	拜访起始时间(时间戳, 毫秒), 访客必填
└─visitEndTimeStamp	integer	非必须	拜访结束时间(时间戳, 毫秒), 访客必填
└─visitReason	string	非必须	拜访原因, 访客可填, 格式: 任意字符, 长度: [1, 255]
└─visitedUuid	string	非必须	受访人的uuid, 访客必填
└─visitType	integer	非必须	访客类型, 访客可填, 1 普通访客, 2 VIP (当为2, 此字段必填)
└─email	string	非必须	邮箱, 格式: 符合一般邮箱格式即可

名称	类型	是否必须	备注
└─phone	string	非必须	手机号, 格式:数字, 长度: [6,18], 访客类型手机号必填, 其他人员类型非必填
└─cardNum	string	非必须	卡号,若填写需保证唯一, 格式: 数字, 长度:[1,20]
└─password	string	非必须	密码,仅对员工生效 ,
└─entryTimeStamp	integer	非必须	入职时间(时间戳, 毫秒),员工填写生效, 其他填写不生效
└─postion	string	非必须	职位, 员工可填,格式: 任意字符, 长度: [0,64], 员工填写生效, 其他填写不生效
└─birthdayStamp	integer	非必须	生日(时间戳, 毫秒), 员工填写生效, 其他填写不生效
└─ext	string	非必须	用户自定义说明
└─groupList	string []	非必须	加入的组uuid集合
└─		非必须	

1.4.3. 返回数据

名称	类型	是否必须	备注
successes	object []	非必须	成功列表
└─name	string	非必须	用户名
└─code	integer	非必须	结果状态码
└─type	integer	非必须	用户类别, 1 员工 2 访客 3 重点人员
└─msg	string	非必须	结果信息提示
└─sex	integer	非必须	性别, 0 未知, 1 男, 2 女
└─uuid	string	非必须	uuid, 若填写需保证唯一, 最大长度32位
└─imageUri	string	非必须	用户识别照片的uri
└─staffCode	string	非必须	员工编码, 员工可填, 若填写需保证唯一, 格式: 允许大小写英文字母, 数字, 长度:[1,32]
└─identifyNum	string	非必须	身份证号, 格式: 允许大小写英文字母, 数字, 长度:[1,32]
└─visitStartTimeStamp	integer	非必须	拜访起始时间(时间戳, 毫秒), 访客必填
└─visitEndTimeStamp	integer	非必须	拜访结束时间(时间戳, 毫秒), 访客必填
└─visitReason	string	非必须	拜访原因, 访客可填, 格式: 任意字符, 长度: [1, 255]
└─visitedUuid	string	非必须	受访人的uuid, 访客必填
└─visitType	integer	非必须	访客类型, 访客可填, 1 普通访客, 2 VIP

名称	类型	是否必须	备注
└─email	string	非必须	邮箱, 格式: 符合一般邮箱格式即可
└─phone	string	非必须	手机号, 格式: 数字, 长度: [6,18]
└─cardNum	string	非必须	卡号, 若填写需保证唯一, 格式: 数字, 长度: [1,16]
└─password	string	非必须	密码, 员工可填, 格式: 数字, 长度: [4,6]
└─entryTimeStamp	integer	非必须	入职时间(时间戳, 毫秒), 员工可填
└─postion	string	非必须	职位, 员工可填, 格式: 任意字符, 长度: [1,64]
└─birthdayStamp	integer	非必须	生日(时间戳, 毫秒)
└─ext	string	非必须	备注, 格式: 任意字符, 长度: [1,255]
└─groupList	string []	非必须	加入的组uuid集合
└─		非必须	
failures	object []	非必须	失败列表
└─name	string	非必须	用户名
└─code	integer	非必须	结果状态码
└─type	integer	非必须	用户类别, 1 员工 2 访客 3 重点人员
└─msg	string	非必须	结果信息提示

名称	类型	是否必须	备注
sex	integer	非必须	性别, 0 未知, 1 男, 2 女
uuid	string	非必须	uuid, 若填写需保证唯一, 最大长度32位
imageUri	string	非必须	用户识别照片的uri
staffCode	string	非必须	员工编码, 员工可填, 若填写需保证唯一, 格式: 允许大小写英文字母, 数字, 长度:[1,32]
identifyNum	string	非必须	身份证号, 格式: 允许大小写英文字母, 数字, 长度:[1,32]
visitStartTimeStamp	integer	非必须	拜访起始时间(时间戳, 毫秒), 访客必填
visitEndTimeStamp	integer	非必须	拜访结束时间(时间戳, 毫秒), 访客必填
visitReason	string	非必须	拜访原因, 访客可填, 格式: 任意字符, 长度: [1, 255]
visitedUuid	string	非必须	受访人的uuid, 访客必填
visitType	integer	非必须	访客类型, 访客可填, 1 普通访客, 2 VIP
email	string	非必须	邮箱, 格式: 符合一般邮箱格式即可
phone	string	非必须	手机号, 格式: 数字, 长度: [6,18]
cardNum	string	非必须	卡号, 若填写需保证唯一, 格式: 数字, 长度:[1,16]
password	string	非必须	密码, 员工可填, 格式: 数字, 长度:[4,6]
entryTimeStamp	integer	非必须	入职时间(时间戳, 毫秒), 员工可填

名称	类型	是否必须	备注
position	string	非必须	职位, 员工可填, 格式: 任意字符, 长度: [1,64]
birthdayStamp	integer	非必须	生日(时间戳, 毫秒)
ext	string	非必须	备注, 格式: 任意字符, 长度: [1,255]
groupList	string []	非必须	加入的组uuid集合

1.4.4. HTTP请求示例

请求 path

```
/v1/api/person/batchAdd
```

请求 body

```
{
  "personList": [
    {
      "name": "我是员工",
      "type": 1,
      "sex": 1,
      "uuid": "ngtwn",
      "imageUri": "_ZzEwMF9mb3JldmVyQnVja2V0_91b2035456f9407ea1b8d944003bbe06",
      "code": "877849",
      "identifyNum": "911102",
      "email": "MgpbsXyhik@126.com",
      "phone": "17792842739",
      "cardNum": "220306",
      "password": "255075",
      "entryTimeStamp": 1622535057000,
      "postion": "海龙",
      "birthdayStamp": 1591113600000,
      "uniqueIdentify": "650677",
      "ext": "备注",
      "groupList": [
        "976055aae31445c5a583ebc1a53c7514"
      ]
    }
  ]
}
```

1.4.5. HTTP响应示例

响应 200

```
{
  "code": 0,
  "data": {
    "failures": [],
    "successes": [
      {
        "birthdayStamp": 1591113600000,
        "cardNum": "220306",
        "email": "MgpbsXyhik@126.com",
        "entryTimeStamp": 1622535057000,
        "ext": "备注",
        "groupList": [
          "976055aae31445c5a583ebc1a53c7514"
        ],
        "identifyNum": "911102",
        "imageUri": "_ZzEwMF9mb3JldmVyQnVja2V0_91b2035456f9407ea1b8d944003bbe0",
        "name": "我是员工",
        "password": "255075",
        "phone": "17792842739",
        "postion": "海龙",
        "sex": 1,
        "staffCode": "877849",
        "type": 1,
        "uuid": "ngtwr"
      }
    ]
  },
  "msg": "成功"
}
```

1.5. 批量删除人员

1.5.1. 基本信息

Path: /v1/api/person/batchDelete

Method: POST

接口描述: 一次删除多个人员

1.5.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
uuidList	string []	必须	用户UUID列表
└─		非必须	

1.5.3. 返回数据

名称	类型	是否必须	备注
successes	object []	非必须	成功列表
└─uuid	string	非必须	uuid
└─code	integer	非必须	结果状态码
└─msg	string	非必须	结果信息提示
failures	object []	非必须	失败列表
└─uuid	string	非必须	uuid
└─code	integer	非必须	结果状态码
└─msg	string	非必须	结果信息提示

1.5.4. HTTP请求示例

请求 path

```
/v1/api/person/batchDelete
```

请求 body

```
{
  "uuidList": [
    "7e6916df6a05444692b0d48621892551",
    "d3eef33925544fac9c83f650be8e727b"
  ]
}
```

1.5.5. HTTP响应示例

响应 200

```
{
  "code": 0,
  "data": {
    "failures": [],
    "successes": [
      {
        "uuid": "7e6916df6a05444692b0d48621892551"
      },
      {
        "uuid": "d3eef33925544fac9c83f650be8e727b"
      }
    ]
  },
  "msg": "成功"
}
```

1.6. 修改人员

1.6.1. 基本信息

Path: /v1/api/person/update

Method: POST

接口描述: 修改人员信息

1.6.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

1.6.3. 返回数据

OK

1.6.4. HTTP请求示例

请求 path

/v1/api/person/update

请求 body

```

{
  "uuid": "363227b5360f4a29b28d0a9543c14914",
  "name": "我是update员工",
  "type": 1,
  "sex": 1,
  "imageUri": "_ZzEwMF9mb3JldmVyQnVja2V0_2fb0ce763b5c4cb29a7d64af407a2f43",
  "code": "910025",
  "identifyNum": "185864",
  "email": "OpMJvwJc62@126.com",
  "phone": "13403743852",
  "cardNum": "247870",
  "password": "451548",
  "entryTimeStamp": 1622535055000,
  "postion": "海龙",
  "birthdayStamp": 1591113600000,
  "uniqueIdentify": "992514",
  "ext": "备注",
  "groupList": [
    "976055aae31445c5a583ebc1a53c7514"
  ]
}

```

1.6.5. HTTP响应示例

响应 200

```

{
  "code": 0,
  "msg": "成功"
}

```

1.7. 搜索人员列表

1.7.1. 基本信息

Path: /v1/api/person/list

Method: POST

接口描述: 搜索人员, 下列字段支持模糊搜索: name、phone、identifyNum、email、code、cardNum

1.7.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码,不传默认1
pageSize	integer	非必须	分页大小, 不传默认20
type	integer	非必须	用户类别, 1 员工 2 访客 3 重点人员
name	string	非必须	用户姓名
phone	string	非必须	手机号
identifyNum	string	非必须	身份证号
email	string	非必须	邮箱
code	string	非必须	员工编号
cardNum	string	非必须	卡号
uniqueIdentify	string	非必须	唯一标识
staffEntryMinStamp	integer	非必须	员工入职日期最小值(时间戳,毫秒)
staffEntryMaxStamp	integer	非必须	员工入职日期最大值(时间戳,毫秒)
visitorCreateMinStamp	integer	非必须	访客创建日期最小值(时间戳,毫秒)
visitorCreateMaxStamp	integer	非必须	访客创建日期最大值(时间戳,毫秒)
filterVisitedType	integer	非必须	筛选访客状态类型, 0 不筛选, 1 有效, 2 无效

名称	类型	是否必须	备注
simpleInfo	boolean	非必须	是否仅返回简单信息(id+name), 默认false
isPinyinOrder	boolean	非必须	是否按拼音排序, 默认true
visitType	integer	非必须	访客类型(1 普通访客, 2 VIP访客)
visitedName	string	非必须	受访员工姓名

1.7.3. 返回数据

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码
pageSize	integer	非必须	分页大小
total	integer	非必须	总数
list	object []	非必须	
└─uuid	string	非必须	用户UUID
└─name	string	非必须	用户名
└─type	integer	非必须	用户类别, 1 员工 2 访客 3 重点人员
└─sex	integer	非必须	性别, 0 未知, 1 男, 2 女
└─imageUri	string	非必须	用户识别照片的url
└─code	string	非必须	员工编码, 员工非必须, 唯一
└─identifyNum	string	非必须	身份证号
└─visitStartTime	string	非必须	拜访起始时间(时间戳, 毫秒), 访客必须
└─visitEndTime	string	非必须	拜访结束时间(时间戳, 毫秒), 访客必须
└─visitReason	string	非必须	拜访原因, 访客非必须
└─visitedUuid	string	非必须	受访人的UUID, 访客必须

名称	类型	是否必须	备注
└─visitedName	string	非必须	受访人的姓名, 访客必须
└─visitType	integer	非必须	访客类型, 访客必须, 1 普通访客, 2 VIP
└─visitedStatus	boolean	非必须	访客状态, 访客必须, true 有效, false 无效
└─email	string	非必须	邮箱
└─phone	string	非必须	手机号
└─cardNum	string	非必须	卡号, 员工非必须, 唯一
└─password	string	非必须	密码, 员工非必须
└─entryTime	string	非必须	入职时间(时间戳, 毫秒), 员工非必须
└─postion	string	非必须	职位, 员工非必须
└─birthday	string	非必须	生日(时间戳, 毫秒), 员工非必须
└─uniqueIdentify	string	非必须	唯一标识, 唯一
└─ext	string	非必须	备注
└─groupList	object []	非必须	人员所属组列表
└─uuid	string	非必须	组uuid
└─name	string	非必须	组名

名称	类型	是否必须	备注
visitFirm	string	非必须	访客所属单位, 访客非必须
isPassRecord	boolean	非必须	是否有通行记录
isAttendanceRecord	boolean	非必须	是否有考勤记录

1.7.4. HTTP请求示例

请求 **path**

```
/v1/api/person/list
```

请求 **body**

```
{
  "type": 1,
  "name": "备用员工1"
}
```

1.7.5. HTTP响应示例

响应 **200**

```

{
  "code": 0,
  "data": {
    "list": [
      {
        "code": "13925810",
        "ext": "",
        "groupList": [
          {
            "name": "员工默认组"
          }
        ],
        "identifyNum": "1234",
        "isAttendanceRecord": false,
        "isPassRecord": false,
        "name": "123456",
        "password": "123456",
        "postion": "",
        "sex": 0,
        "type": 1,
        "uuid": "950d737b2051410986698adb493965d5"
      }
    ],
    "pageNum": 1,
    "pageSize": 20,
    "total": 1
  },
  "msg": "成功"
}

```

1.8. 根据uuid列表返回人员详细信息列表

1.8.1. 基本信息

Path: /v1/api/person/uuid/list

Method: POST

接口描述: 通过uuid列表一次获取多个人员信息

1.8.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
uuidList	string []	必须	人员UUID列表
└─			

1.8.3. 返回数据

名称	类型	是否必须	备注
	object []	非必须	
└─uuid	string	非必须	用户UUID
└─name	string	非必须	用户名
└─type	integer	非必须	用户类别, 1 员工 2访客 3重点人员
└─sex	integer	非必须	性别, 0 未知, 1 男, 2 女
└─imageUri	string	非必须	用户识别照片的url
└─code	string	非必须	员工编码,员工非必须,唯一
└─identifyNum	string	非必须	身份证号
└─visitStartTime	string	非必须	拜访起始时间(时间戳,毫秒), 访客必须
└─visitEndTime	string	非必须	拜访结束时间(时间戳,毫秒), 访客必须
└─visitReason	string	非必须	拜访原因, 访客非必须
└─visitedUuid	string	非必须	受访人的UUID,访客必须
└─visitedName	string	非必须	受访人的姓名, 访客必须
└─visitType	integer	非必须	访客类型, 访客必须, 1 普通访客, 2 VIP
└─visitedStatus	boolean	非必须	访客状态, 访客必须, true 有效, false 无效
└─email	string	非必须	邮箱
└─phone	string	非必须	手机号
└─cardNum	string	非必须	卡号,员工非必须, 唯一
└─password	string	非必须	密码, 员工非必须
└─entryTime	string	非必须	入职时间(时间戳, 毫秒), 员工非必须
└─postion	string	非必须	职位, 员工非必须

名称	类型	是否必须	备注
└─birthday	string	非必须	生日(时间戳,毫秒), 员工非必须
└─uniqueIdentify	string	非必须	唯一标识, 唯一
└─ext	string	非必须	备注
└─groupList	object []	非必须	人员所属组列表
└─uuid	string	非必须	组uuid
└─name	string	非必须	组名
└─visitFirm	string	非必须	访客所属单位, 访客非必须

1.8.4. HTTP请求示例

请求 path

```
/v1/api/person/uuid/list
```

请求 body

```
{
  "uuidList": [
    "0ff02877cceb4b2eab1a737854165b21"
  ]
}
```

1.8.5. HTTP响应示例

响应 200

```

{
  "code": 0,
  "data": [
    {
      "birthday": 1591113600000,
      "cardNum": "717277",
      "code": "158138",
      "email": "zcsEou3S0G@126.com",
      "entryTime": 1622535059000,
      "ext": "备注",
      "groupList": [
        {
          "name": "员工默认组"
        }
      ],
      "identifyNum": "840369",
    }
  ]
}

```

1.9. 根据uuid返回人员信息

1.9.1. 基本信息

Path: /v1/api/person/query

Method: POST

接口描述: 通过**uuid**获取个人信息

1.9.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
uuid	string	必须	员工、访客uuid

1.9.3. 返回数据

1.9.4. HTTP请求示例

请求 path

```
/v1/api/person/query
```

请求 body

```
{
  "uuid": "72ec703308e840cfb05287ac9f268be5"
}
```

1.9.5. HTTP响应示例

响应 200

```
{
  "code": 0,
  "data": {
    "birthday": 1591113600000,
    "cardNum": "006910",
    "code": "480914",
    "email": "BSSQPEa7Jk@126.com",
    "entryTime": 1622535059000,
    "ext": "备注",
    "groupList": [
      {
        "name": "员工默认组",
        "uuid": "976055aae31445c5a583ebc1a53c7514"
      }
    ],
    "identifyNum": "347876",
    "imageUri": "http://10.171.4.130/pub/_ZzEwMF9mb3JldmVyQnVja2V0_a53fe0cb7607458",
    "name": "备用员工1",
    "password": "060844",
    "phone": "13262729364",
    "postion": "海龙",
    "sex": 1,
    "type": 1,
    "uuid": "72ec703308e840cfb05287ac9f268be5"
  },
  "msg": "成功"
}
```

1.10. 根据员工、访客UUID获取二维码

1.10.1. 基本信息

Path: /v1/api/person/visitorCode

Method: POST

接口描述: 员工、访客获取二维码使用

1.10.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
uuid	string	必须	员工、访客uuid

1.10.3. 返回数据

名称	类型	是否必须	备注
qrCode	string	非必须	员工、访客二维码图片(base64编码的字符串)
expireTime	integer	非必须	二维码图片的过期分钟数

1.10.4. HTTP请求示例

请求 path

```
/v1/api/person/visitorCode
```

请求 body

```
{
  "uuid": "72ec703308e840cfb05287ac9f268be5"
}
```

1.10.5. HTTP响应示例

响应 200

```
{
  "code": 0,
  "data": {
    "expireTime": 10,
    "qrCode": "data:image/png;base64,R0lGODlhyADIAPAAAAAAP//ywAAAAyADIAEAC/4yPc"
  },
  "msg": "成功"
}
```

- 1. 组管理
 - 1.1. 批量添加组
 - 1.1.1. 基本信息
 - 1.1.2. 请求参数
 - 1.1.3. 返回数据
 - 1.1.4. HTTP请求示例
 - 1.1.5. HTTP响应示例
 - 1.2. 删除组
 - 1.2.1. 基本信息
 - 1.2.2. 请求参数
 - 1.2.3. 返回数据
 - 1.2.4. HTTP响应示例
 - 1.3. 修改组
 - 1.3.1. 基本信息
 - 1.3.2. 请求参数
 - 1.3.3. 返回数据
 - 1.3.4. HTTP请求示例
 - 1.3.5. HTTP响应示例
 - 1.4. 组列表
 - 1.4.1. 基本信息
 - 1.4.2. 请求参数
 - 1.4.3. 返回数据
 - 1.4.4. HTTP请求示例
 - 1.4.5. HTTP响应示例
 - 1.5. 查询组详情
 - 1.5.1. 基本信息
 - 1.5.2. 请求参数
 - 1.5.3. 返回数据
 - 1.5.4. HTTP请求示例
 - 1.5.5. HTTP响应示例
 - 1.6. 批量移除组内人员
 - 1.6.1. 基本信息
 - 1.6.2. 请求参数
 - 1.6.3. 返回数据
 - 1.6.4. HTTP请求示例
 - 1.6.5. HTTP响应示例
 - 1.7. 查询组内人员详情
 - 1.7.1. 基本信息
 - 1.7.2. 请求参数
 - 1.7.3. 返回数据
 - 1.7.4. HTTP请求示例
 - 1.7.5. HTTP响应示例
 - 1.8. 组内批量增加或删除人员
 - 1.8.1. 基本信息
 - 1.8.2. 请求参数
 - 1.8.3. 返回数据
 - 1.8.4. HTTP请求示例
 - 1.8.5. HTTP响应示例

- 1.9. 获取默认组信息（不建议使用）
 - 1.9.1. 基本信息
 - 1.9.2. 请求参数
 - 1.9.3. 返回数据
 - 1.9.4. HTTP请求示例
 - 1.9.5. HTTP响应示例

1. 组管理

1.1. 批量添加组

1.1.1. 基本信息

Path: /v1/api/pass/group/batchAdd

Method: POST

接口描述: 批量添加多个组

1.1.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
groupList	object []	非必须	人员组列表
├─uuid	string	非必须	人员组唯一标志uuid, 长度1~40, 不支持特殊字符
├─name	string	必须	组名称, 唯一, 长度2~32, 不支持特殊字符
├─type	integer	必须	组类型, 1: 员工组; 2: 访客组; 3: 重点人员组

1.1.3. 返回数据

名称	类型	是否必须	备注
successes	object []	非必须	成功列表
└─uuid	string	非必须	人员组唯一标志uuid
└─code	integer	非必须	结果状态码
└─name	string	非必须	人员组名称
└─msg	string	非必须	结果信息提示
└─type	integer	非必须	1 员工 2 访客 3 重点人员
failures	object []	非必须	失败列表
└─uuid	string	非必须	人员组唯一标志uuid
└─code	integer	非必须	结果状态码
└─name	string	非必须	人员组名称
└─msg	string	非必须	结果信息提示
└─type	integer	非必须	1 员工 2 访客 3 重点人员

1.1.4. HTTP请求示例

请求 path

```
/v1/api/pass/group/batchAdd
```

请求 body

```
{
  "groupList": [
    {
      "name": "我是员工组",
      "type": 1,
      "uuid": "empgroupUuid"
    }
  ]
}
```

1.1.5. HTTP响应示例

响应 200

```
{
  "code": 0,
  "data": {
    "failures": [],
    "successes": [
      {
        "name": "我是员工组",
        "type": 1,
        "uuid": "empgroupUuid"
      }
    ]
  },
  "msg": "成功"
}
```

1.2. 删除组

1.2.1. 基本信息

Path: /v1/api/pass/group/delete

Method: POST

接口描述: 删除组信息

1.2.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
uuid	string	必须	组uuid

1.2.3. 返回数据

OK

HTTP请求示例

请求 path

/v1/api/pass/group/delete

请求 body

```
{
  "uuid": "158feab6a61746698ce6aa43264e8e4a"
}
```

1.2.4. HTTP响应示例

响应 **200**

```
{
  "code": 0,
  "msg": "成功"
}
```

1.3. 修改组

1.3.1. 基本信息

Path: /v1/api/pass/group/update

Method: POST

接口描述: 修改组名字

1.3.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
uuid	string	必须	人员组唯一标志uuid
name	string	必须	组名称

1.3.3. 返回数据

OK

1.3.4. HTTP请求示例

请求 **path**

```
/v1/api/pass/group/update
```

请求 body

```
{
  "uuid": "2075f5ceb40d456db65af7e6515b59c4",
  "name": "update员工组"
}
```

1.3.5. HTTP响应示例

响应 200

```
{
  "code": 0,
  "msg": "成功"
}
```

1.4. 组列表

1.4.1. 基本信息

Path: /v1/api/pass/group/list

Method: POST

接口描述: 查询1-员工； 2-访客； 3-重点人员 不同分类下的所有的组，支持通过name模糊查询

1.4.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
pageNum	integer	非必须	分页页码，默认为 1
pageSize	integer	非必须	分页大小，默认为 20
type	integer	必须	组类别：1-员工； 2-访客； 3-重点人员
name	string	非必须	组名

1.4.3. 返回数据

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码
pageSize	integer	非必须	分页大小
total	integer	非必须	总数
list	object []	非必须	
└─uuid	string	非必须	组UUID
└─name	string	非必须	组名
└─type	integer	非必须	组类型, 1: 员工组; 2: 访客组; 3: 重点人员组
└─personTotal	integer	非必须	总人数

1.4.4. HTTP请求示例

请求 path

```
/v1/api/pass/group/list
```

请求 body

```
{
  "type": 1,
  "name": "员工组备用",
  "pageNum": 1,
  "pageSize": 10
}
```

1.4.5. HTTP响应示例

响应 200


```
{
  "code": 0,
  "data": {
    "list": [
      {
        "isAll": 0,
        "name": "员工组备用",
        "personTotal": 1,
        "type": 1,
        "uuid": "5b7065c4a84f4f41b70d3d534e1a3a9e"
      }
    ],
    "pageNum": 1,
    "pageSize": 10,
    "total": 1
  },
  "msg": "成功"
}
```

1.5. 查询组详情

1.5.1. 基本信息

Path: /v1/api/pass/group/query

Method: POST

接口描述: 查询指定组的详细信息，包括组内人员总数

1.5.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
uuid	string	非必须	组uuid

1.5.3. 返回数据

名称	类型	是否必须	备注
uuid	string	非必须	uuid
name	string	非必须	组名
type	integer	非必须	组类型, 1: 员工组; 2: 访客组; 3: 重点人员组
personTotal	integer	非必须	组内员工数

1.5.4. HTTP请求示例

请求 path

```
/v1/api/pass/group/query
```

请求 body

```
{
  "uuid": "00eea45c96e64954a8de2d89aa5adcb9"
}
```

1.5.5. HTTP响应示例

响应 200

```
{
  "code": 0,
  "data": {
    "isAll": 0,
    "name": "员工组备用",
    "personTotal": 1,
    "type": 1,
    "uuid": "00eea45c96e64954a8de2d89aa5adcb9"
  },
  "msg": "成功"
}
```

1.6. 批量移除组内人员

1.6.1. 基本信息

Path: /v1/api/pass/group/batchDeletePerson

Method: POST

接口描述: 一次删除组内多个人员

1.6.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
groupUuid	string	必须	人员组唯一标志uuid
personUids	string []	必须	人员uuid列表

1.6.3. 返回数据

OK

1.6.4. HTTP请求示例

请求 path

/v1/api/pass/group/batchDeletePerson

请求 body

<pre>{ "groupUuid": "6eed172bb5084455bbd7d29d53564289", "personUids": ["57f1acb42a0046ff9c2e957756b3e7b8"] }</pre>
--

1.6.5. HTTP响应示例

响应 200

<pre>{ "code": 0, "msg": "成功" }</pre>

1.7. 查询组内人员详情

1.7.1. 基本信息

Path: /v1/api/pass/group/queryPersonListInGroup

Method: POST

接口描述: 查询 指定组内人员详情，支持通过name、personCode模糊查询

1.7.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
pageNum	integer	非必须	分页页码，默认为 1
pageSize	integer	非必须	分页大小，默认为 20
uuid	string	必须	人员组唯一标志uuid
type	integer	必须	组类型, 1: 员工组; 2: 访客组; 3: 重点人员组
name	string	非必须	人员姓名
personCode	string	非必须	人员编号

1.7.3. 返回数据

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码
pageSize	integer	非必须	分页大小
total	integer	非必须	总数
list	object []	非必须	
└─groupUuid	string	非必须	组uuid
└─personUuid	string	非必须	人员uuid
└─imageUri	string	非必须	用户图片
└─personName	string	非必须	人员名
└─personCode	string	非必须	员工编号(员工组)
└─visitTargetUuid	string	非必须	拜访用户uuid
└─visitTarget	string	非必须	拜访用户姓名
└─visitStartTime	string	非必须	拜访起始时间(时间戳)
└─visitEndTime	string	非必须	拜访结束时间(时间戳)
└─status	string	非必须	访客状态：无效、有效
└─phone	string	非必须	手机号码
└─identifyNum	string	非必须	身份证号码
└─type	integer	非必须	人员类型, 1: 员工; 2: 访客; 3: 重点人员
└─visitReason	string	非必须	拜访原因, 访客可填, 格式: 任意字符, 长度: 1-255

1.7.4. HTTP请求示例

请求 path

```
/v1/api/pass/group/queryPersonListInGroup
```

请求 body

```
{
  "type": 1,
  "uuid": "1b3c5dedd5984c628b1beb6b80d0e102",
  "pageSize": 10,
  "pageNum": 1
}
```

1.7.5. HTTP响应示例

响应 200

```
{
  "code": 0,
  "data": {
    "list": [
      {
        "groupUuid": "1b3c5dedd5984c628b1beb6b80d0e102",
        "imageUri": "http://10.171.4.130/pub/_ZzEwMF9mb3JldmVyQnVja2V0_7bd1e32",
        "personCode": "40906240",
        "personName": "备用员工1",
        "personUuid": "14cc526644b740d9a94009e0ff25e51e",
        "phone": "13684505620",
        "type": 1
      }
    ],
    "pageNum": 1,
    "pageSize": 10,
    "total": 1
  },
  "msg": "成功"
}
```

1.8. 组内批量增加或删除人员

1.8.1. 基本信息

Path: /v1/api/pass/group/batchUpdatePerson

Method: POST

接口描述: 组内批量增加或删除人员，personUuids为空，删除组内所有人员，personUuids不为空，通过 groupUuid获取组内已有人员为originPersonUuids，进行差集计算，例：待删除: originPersonUuids-personUuids，待添加: personUuids-originPersonUuids

1.8.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
groupUuid	string	非必须	人员组唯一标志uuid
personUuids	string []	非必须	人员uuid列表

1.8.3. 返回数据

OK

1.8.4. HTTP请求示例

请求 path

```
/v1/api/pass/group/batchUpdatePerson
```

请求 body

```
{
  "groupUuid": "b0b9ed9f106f4b7cb12ae5ccd44c70b8",
  "personUuids": [
    "6e386a7b0dcd4df38c57bc990d5ce62e",
    "c90a3919a62442059d5319e1b9c30bcd"
  ]
}
```

1.8.5. HTTP响应示例

响应 200

```
{
  "code": 0,
  "msg": "成功"
}
```

1.9. 获取默认组信息（不建议使用）

1.9.1. 基本信息

Path: /v1/api/pass/group/getDefaultGroup

Method: POST

接口描述: 获取每个分类下一个默认的组

1.9.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

1.9.3. 返回数据

名称	类型	是否必须	备注
type	integer	非必须	组类别：1-员工； 2-访客； 3-重点人员

1.9.4. HTTP请求示例

请求 **path**

```
/v1/api/pass/group/getDefaultGroup
```

请求 **body**

```
{
  "type": 1
}
```

1.9.5. HTTP响应示例

响应 **200**

```
{
  "code": 0,
  "data": {
    "isAll": 0,
    "name": "员工默认组",
    "personTotal": 0,
    "type": 1,
    "uuid": "976055a31445c5a583ebc1a53c7514"
  },
  "msg": "成功"
}
```


- 1. 设备管理
 - 1.1. 设备添加
 - 1.1.1. 基本信息
 - 1.1.2. 请求参数
 - 1.1.3. 返回数据
 - 1.1.4. HTTP请求示例
 - 1.1.5. HTTP响应示例
 - 1.2. 发现设备列表
 - 1.2.1. 基本信息
 - 1.2.2. 请求参数
 - 1.2.3. 返回数据
 - 1.2.4. HTTP请求示例
 - 1.2.5. HTTP响应示例
 - 1.3. 面板机设备导入
 - 1.3.1. 基本信息
 - 1.3.2. 请求参数
 - 1.3.3. 返回数据
 - 1.3.4. HTTP请求示例
 - 1.3.5. HTTP响应示例
 - 1.4. 删除设备预校验
 - 1.4.1. 基本信息
 - 1.4.2. 请求参数
 - 1.4.3. 返回数据
 - 1.4.4. HTTP请求示例
 - 1.4.5. HTTP响应示例
 - 1.5. 删除设备
 - 1.5.1. 基本信息
 - 1.5.2. 请求参数
 - 1.5.3. 返回数据
 - 1.5.4. HTTP请求示例
 - 1.5.5. HTTP响应示例
 - 1.6. 修改设备信息
 - 1.6.1. 基本信息
 - 1.6.2. 请求参数
 - 1.6.3. 返回数据
 - 1.6.4. HTTP请求示例
 - 1.6.5. HTTP响应示例
 - 1.7. 查询设备列表
 - 1.7.1. 基本信息
 - 1.7.2. 请求参数
 - 1.7.3. 返回数据
 - 1.7.4. HTTP请求示例
 - 1.7.5. HTTP响应示例
 - 1.8. 查询设备信息（不建议使用）
 - 1.8.1. 基本信息
 - 1.8.2. 请求参数
 - 1.8.3. 返回数据
 - 1.8.4. HTTP请求示例

- 1.8.5. HTTP响应示例
- 1.9. 查询设备详情
 - 1.9.1. 基本信息
 - 1.9.2. 请求参数
 - 1.9.3. 返回数据
 - 1.9.4. HTTP请求示例
 - 1.9.5. HTTP响应示例
- 1.10. 查询算力设备可绑定的端设备列表（Core）
 - 1.10.1. 基本信息
 - 1.10.2. 请求参数
 - 1.10.3. 返回数据
 - 1.10.4. HTTP请求示例
 - 1.10.5. HTTP响应示例
- 1.11. 查询算力设备可绑定的端设备列表（非Core）
 - 1.11.1. 基本信息
 - 1.11.2. 请求参数
 - 1.11.3. 返回数据
 - 1.11.4. HTTP请求示例
 - 1.11.5. HTTP响应示例
- 1.12. 按设备类型分组查询设备列表（不建议使用）
 - 1.12.1. 基本信息
 - 1.12.2. 请求参数
 - 1.12.3. 返回数据
 - 1.12.4. HTTP请求示例
 - 1.12.5. HTTP响应示例
- 1.13. 查询设备型号关系（不建议使用）
 - 1.13.1. 基本信息
 - 1.13.2. 请求参数
 - 1.13.3. 返回数据
 - 1.13.4. HTTP请求示例
 - 1.13.5. HTTP响应示例
- 1.14. 设备绑定
 - 1.14.1. 基本信息
 - 1.14.2. 请求参数
 - 1.14.3. 返回数据
 - 1.14.4. HTTP请求示例
 - 1.14.5. HTTP响应示例
- 1.15. 解绑设备预校验
 - 1.15.1. 基本信息
 - 1.15.2. 请求参数
 - 1.15.3. 返回数据
 - 1.15.4. HTTP请求示例
 - 1.15.5. HTTP响应示例
- 1.16. 设备解绑
 - 1.16.1. 基本信息
 - 1.16.2. 请求参数
 - 1.16.3. 返回数据
 - 1.16.4. HTTP请求示例

- 1.16.5. HTTP响应示例
- 1.17. 门禁、人员管控设备列表查询
 - 1.17.1. 基本信息
 - 1.17.2. 请求参数
 - 1.17.3. 返回数据
 - 1.17.4. HTTP请求示例
 - 1.17.5. HTTP响应示例
- 1.18. 添加门禁、人员管控设备
 - 1.18.1. 基本信息
 - 1.18.2. 请求参数
 - 1.18.3. 返回数据
 - 1.18.4. HTTP请求示例
 - 1.18.5. HTTP响应示例
- 1.19. 移除门禁、人员管控设备
 - 1.19.1. 基本信息
 - 1.19.2. 请求参数
 - 1.19.3. 返回数据
 - 1.19.4. HTTP响应示例
- 1.20. 设备可布控列表查询
 - 1.20.1. 基本信息
 - 1.20.2. 请求参数
 - 1.20.3. 返回数据
 - 1.20.4. HTTP请求示例
 - 1.20.5. HTTP响应示例

1. 设备管理

1.1. 设备添加

1.1.1. 基本信息

Path: /v1/api/device/add

Method: POST

接口描述: 设备新增接口，仅支持网络摄像机、智能摄像头、智能服务器、存储服务器的添加，面板机设备需通过面板机设备导入添加至系统，仅添加网络摄像机支持Onvif协议，并且Onvif协议下网络摄像机可以无须传递rtsp视频流地址，智能摄像机、网络摄像机、智能服务器中的MegBox-B3R-1012(H1)、MegCube-B4H16-311设备添加时必须添加设备模式，部分设备（详细请查看数据字典）支持以主动模式添加，主动模式下无需填写IP地址，但必须填写snCode和端口号；

1.1.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
name	string	必须	设备名称（长度1-50位）
location	string	必须	设备位置（长度1-50位）
deviceType	integer	必须	设备类型 1、智能摄像头 3、智能服务器 4、网络摄像机 5、存储服务器
deviceModel	integer	必须	设备型号（详细型号描述请查看数据字典）
deviceMode	integer	非必须	设备模式（网络摄像机、智能摄像头、智能服务器【MegBox-B3R-1012(H1)、MegCube-B4H16-311】时必填） 1、人脸识别 2、结构化 3、警戒 4、人脸抓拍 5、混合模式
mac	string	非必须	mac地址
userName	string	非必须	用户名
userPassword	string	非必须	用户密码
rtspUrl	string	非必须	视频流地址
protocol	integer	非必须	协议类型:1rtspt协议 2onvif协议（网络摄像机必填）
activeMode	integer	非必须	是否支持主动模式 0：非主动模式，1：主动模式
ipAddress	string	非必须	设备IP地址，非PDNS接入时必填
port	integer	非必须	端口，PDNS接入时必填
snCode	string	非必须	设备sn码，PDNS接入时必填

1.1.3. 返回数据

名称	类型	是否必须	备注
successes	object []	非必须	成功列表
└─ uuid	string	非必须	设备uuid
└─ code	integer	非必须	结果状态码
└─ name	string	非必须	设备名称
└─ location	string	非必须	设备位置
└─ msg	string	非必须	结果信息提示
└─ deviceType	integer	非必须	设备类型 1、智能摄像头 3、智能服务器 4、网络摄像机 5、存储服务器
└─ deviceModel	integer	非必须	设备型号
└─ deviceMode	integer	非必须	设备模式 1、人脸识别 2、结构化 3、警戒 4、人脸抓拍 5、混合模式
└─ ipAddress	string	非必须	设备IP
└─ mac	string	非必须	mac地址
└─ userName	string	非必须	用户名
└─ userPassword	string	非必须	用户密码
└─ rtspUrl	string	非必须	视频流地址
└─ port	integer	非必须	端口

名称	类型	是否必须	备注
 protocol	integer	非必须	协议类型:1rtspt协议 2onvif协议
 activeMode	integer	非必须	是否是主动模式接入, 0: 非主动模式, 1: 主动模式
 snCode	string	非必须	设备sn码, PDNS接入时必须填
failures	object []	非必须	失败列表
 uuid	string	非必须	设备uuid
 code	integer	非必须	结果状态码
 name	string	非必须	设备名称
 location	string	非必须	设备位置
 msg	string	非必须	结果信息提示
 deviceType	integer	非必须	设备类型 1、智能摄像头 3、智能服务器 4、网络摄像机 5、存储服务器
 deviceModel	integer	非必须	设备型号 (详细型号描述请查看数据字典)
 deviceMode	integer	非必须	设备模式 (网络摄像机、智能摄像头、智能服务器设备B3R的型号为MegBox-B3R-1012(H1)、MegCube-B4H16-311的必填) 1、人脸识别 2、结构化 3、警戒 4、人脸抓拍 5、混合模式
 ipAddress	string	非必须	设备IP
 mac	string	非必须	mac地址

名称	类型	是否必须	备注
 userName	string	非必须	用户名
 userPassword	string	非必须	用户密码
 rtspUrl	string	非必须	视频流地址
 port	integer	非必须	端口
 protocol	integer	非必须	协议类型:1rtspt协议 2onvif协议
 activeMode	integer	非必须	是否是主动模式接入，0：非主动模式，1：主动模式
 snCode	string	非必须	设备sn码，PDNS接入时必须填

1.1.4. HTTP请求示例

请求 path

```
/v1/api/device/add
```

请求 body

```
{
  "name": "智能摄像头-MegEye-C3S-123-人脸抓拍",
  "location": "OpenAPI测试",
  "deviceType": 1,
  "deviceModel": 4,
  "deviceMode": 4,
  "ipAddress": "10.171.4.76",
  "userName": "admin",
  "userPassword": "admin",
  "rtspUrl": "rtsp://10.171.4.76/user=admin&password=admin&channel=1&stream=0.sdp",
  "port": "34567"
}
```

1.1.5. HTTP响应示例

响应 200

```
{
  "code":0,
  "data":{
    "failures":[
    ],
    "successes":[
      {
        "activeMode":0,
        "deviceMode":4,
        "deviceModel":4,
        "deviceType":1,
        "ipAddress":"10.171.4.76",
        "location":"OpenAPI测试",
        "name":"智能摄像头-MegEye-C3S-123-人脸抓拍",
        "port":34567,
        "protocol":1,
        "rtspUrl":"rtsp://10.171.4.76/user=admin&password=admin&channel=1&stream=1",
        "userName":"admin",
        "userPassword":"admin",
        "uuid":"27a76e6a158b4e62817a7dbd6d3c43b0"
      }
    ]
  },
  "msg":"成功"
}
```

1.2. 发现设备列表

1.2.1. 基本信息

Path: /v1/api/device/queryReportDevice

Method: POST

接口描述: 查询面板机注册系统但未进行面板机导入的设备列表，通过面板机注册至盘古系统后，可在发现设备列表查询到此设备并将设备导入至系统；

1.2.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码,不传默认1
snCode	string	非必须	设备编码
deviceType	integer	非必须	设备类型 2 智能面板机
pageSize	integer	非必须	分页大小, 不传默认20
discoveryMethod	integer	非必须	发现方式 1-主动上报

1.2.3. 返回数据

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码
pageSize	integer	非必须	分页大小
total	integer	非必须	总数
list	object []	非必须	
└─ uuid	string	非必须	设备uuid
└─ deviceType	integer	非必须	设备类型 2 智能面板机
└─ snCode	string	非必须	设备编码
└─ gmtCreate	string	非必须	操作时间
└─ discoveryMethod	integer	非必须	发现方式 1-主动上报

1.2.4. HTTP请求示例

请求 **path**

```
/v1/api/device/queryReportDevice
```

请求 **body**

```
{
  "pageNum":1,
  "snCode":"M014200332004000094",
  "deviceType":2,
  "pageSize":10,
  "discoveryMethod":1
}
```

1.2.5. HTTP响应示例

响应 **200**

```

{
  "code":0,
  "data":{
    "list":[
      {
        "deviceType":2,
        "discoveryMethod":1,
        "gmtCreate":1622602983000,
        "snCode":"M014200332004000094",
        "uuid":"8b2e4da9392b4847aff5faf9dcb17a3c"
      }
    ],
    "pageNum":1,
    "pageSize":10,
    "total":1
  },
  "msg":"成功"
}

```

1.3. 面板机设备导入

1.3.1. 基本信息

Path: /v1/api/device/importDevice

Method: POST

接口描述: 面板机注册至系统后，在发现设备列表发现此设备，通过此接口导入至盘古系统，仅面板机支持，其他类型设备需通过设备添加接口添加；

1.3.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
name	string	必须	设备名称
location	string	必须	设备位置
deviceType	integer	必须	设备类型 2 智能面板机
deviceModel	integer	必须	设备型号 1 MegEye-W5K-I8 2 MegEye-W4K-E7 3 MegEye-W4K-I5 34 MegEye-W4K-I52 35 MegEye-W3K-45S 36 MegEye-W4K-I31
deviceMode	integer	必须	设备模式 1-人脸识别
snCode	string	必须	设备编码
discoveryMethod	integer	必须	发现方式 1-主动上报

1.3.3. 返回数据

OK

1.3.4. HTTP请求示例

请求 path

```
/v1/api/device/importDevice
```

请求 body

```
{
  "name": "智能面板机-MegEye-W5K-I8",
  "deviceMode": 1,
  "deviceType": 2,
  "location": "OpenAPI测试",
  "deviceModel": 1,
  "snCode": "M014200332004000094",
  "discoveryMethod": 1
}
```

1.3.5. HTTP响应示例

响应 200

```
{
  "code":0,
  "msg":"成功"
}
```

1.4. 删除设备预校验

1.4.1. 基本信息

Path: /v1/api/device/preDelete

Method: POST

接口描述: 删除设备前，调用此接口进行预删除校验，防止绑定、布控等正在使用的设备删除；

1.4.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
uuid	string	必须	设备uuid

1.4.3. 返回数据

名称	类型	是否必须	备注
tag	integer	非必须	标识 0-可删除 1-已绑定 2-已布控 3-强制删除（1、3状态可执行删除操作）
deployType	integer	非必须	布控类型 1-警戒布控 2-结构化布控 3-门禁点布控 4-事件联动布控 5-员工、访客、重点人员布控 6-车辆布控 7-全局陌生人员布控（tag为2时此值存在）

1.4.4. HTTP请求示例

请求 path

```
/v1/api/device/preDelete
```

请求 body

```
{
  "uuid": "eabe5bbbd4e24faba672aa45d09a6607"
}
```

1.4.5. HTTP响应示例

响应 200

```
{
  "code": 0,
  "data": {
    "tag": 1
  },
  "msg": "成功"
}
```

1.5. 删除设备

1.5.1. 基本信息

Path: /v1/api/device/delete

Method: POST

接口描述: 已添加设备的删除接口，删除设备前需调用删除设备预校验接口；

1.5.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
uuid	string	必须	设备uuid

1.5.3. 返回数据

OK

1.5.4. HTTP请求示例

请求 path

/v1/api/device/delete

请求 **body**

```
{
  "uuid": "eabe5bbbd4e24faba672aa45d09a6607"
}
```

1.5.5. HTTP响应示例

响应 **200**

```
{
  "code": 0,
  "msg": "成功"
}
```

1.6. 修改设备信息

1.6.1. 基本信息

Path: /v1/api/device/update

Method: POST

接口描述: 修改设备信息接口，已绑定算力设备的相机只允许修改基础信息（名称、位置、mac地址），未绑定的相机可以修改几乎全部信息（不可修改设备类型、设备型号、设备模式、协议类型、SN码、主动模式）；

1.6.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
uuid	string	必须	设备uuid
name	string	必须	设备名称
location	string	必须	设备位置
mac	string	非必须	mac地址
userName	string	非必须	用户名
userPassword	string	非必须	用户密码
rtspUrl	string	非必须	视频流地址
ipAddress	string	非必须	设备IP
port	integer	非必须	端口
activeMode	integer	非必须	

1.6.3. 返回数据

OK

1.6.4. HTTP请求示例

请求 path

/v1/api/device/update

请求 body

```
{
  "uuid": "ae61bbd608ef415cb9a8060e5434cee6",
  "name": "设备更新API测试名称",
  "location": "OpenAPI测试",
  "ipAddress": "10.171.4.50",
  "mac": "",
  "userName": "admin",
  "userPassword": "admin123",
  "port": 3000,
  "rtspUrl": "rtsp://admin:admin123@10.171.4.50:554/1/1"
}
```

1.6.5. HTTP响应示例

响应 200

```
{
  "code": 0,
  "msg": "成功"
}
```

1.7. 查询设备列表

1.7.1. 基本信息

Path: /v1/api/device/list

Method: POST

接口描述: 分页查询设备列表，返回设备的基础信息；

1.7.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
name	string	非必须	设备名称
pageNum	integer	非必须	当前页码,不传默认1
deviceType	integer	非必须	设备类型 1 智能摄像头 2 智能面板机 3 智能服务器 4 网络摄像机 5 存储服务器
pageSize	integer	非必须	分页大小，不传默认20

1.7.3. 返回数据

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码
pageSize	integer	非必须	分页大小
total	integer	非必须	总数
list	object []	非必须	
└─ uuid	string	非必须	设备uuid
└─ location	string	非必须	设备位置
└─ name	string	非必须	设备名称
└─ deviceType	integer	非必须	设备类型 1 智能摄像头 2 智能面板机 3 智能服务器 4 网络摄像机 5 存储服务器
└─ status	integer	非必须	设备状态 3 在线 4 离线
└─ ipAddress	string	非必须	设备IP
└─ mac	string	非必须	mac地址
└─ deviceMode	integer	非必须	设备模式 1、人脸识别 2、结构化 3、警戒 4、人脸抓拍 5、混合模式 6、人脸识别标准模式 7、人脸识别高性能模式
└─ upgradeStatus	integer	非必须	升级状态 0 升级中 1 升级成功 2 升级失败 4 可升级 5 升级完成
└─ upgradeAction	integer	非必须	升级操作 0-更新版本（置灰）1-更新版本 2-历史版本
└─ currentVersion	string	非必须	当前版本

名称	类型	是否必须	备注
checkFirm	integer	非必须	设备型号是否是固件 0 表示不是 1 表示是固件

1.7.4. HTTP请求示例

请求 path

```
/openapi/v1/device/list
```

请求 body

```
{
  "name": "智能摄像头-MegEye-C4R-322-人脸抓拍",
  "deviceType": 1,
  "pageNum": 1,
  "pageSize": 10
}
```

1.7.5. HTTP响应示例

响应 200

```
{
  "code": 0,
  "data": {
    "list": [
      {
        "checkFirm": 0,
        "deviceMode": 4,
        "deviceType": 1,
        "ipAddress": "10.171.4.50",
        "location": "OpenAPI测试",
        "name": "智能摄像头-MegEye-C4R-322-人脸抓拍",
        "status": 4,
        "upgradeAction": 0,
        "uuid": "92dca22d1d3a439bbf649cd58245e837"
      }
    ],
    "pageNum": 1,
    "pageSize": 10,
    "total": 1
  },
  "msg": "成功"
}
```

1.8. 查询设备信息（不建议使用）

1.8.1. 基本信息

Path: /v1/api/device/queryDeviceInfo

Method: POST

接口描述: 查询设备详情接口，返回设备部门信息，主要是编辑回显使用，已不建议使用，建议使用/v1/api/device/query接口

1.8.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
uuid	string	必须	设备uuid

1.8.3. 返回数据

名称	类型	是否必须	备注
uuid	string	非必须	uuid
name	string	非必须	设备名称
ipAddress	string	非必须	设备IP
location	string	非必须	设备位置
deviceType	integer	非必须	设备类型 1 智能摄像头 2 智能面板机 3 智能服务器 4 网络摄像机 5 存储服务器
mac	string	非必须	mac地址
deviceMode	integer	非必须	设备模式 1、人脸识别 2、结构化 3、警戒 4、人脸抓拍 5、混合模式
userName	string	非必须	用户名
userPassword	string	非必须	用户密码
deviceModel	integer	非必须	设备型号
discoveryMethod	integer	非必须	发现方式 1-主动上报（面板机）， 2-主动添加（非面板机）
rtspUrl	string	非必须	视频流地址
port	integer	非必须	端口号
protocol	integer	非必须	协议类型:1rtspt协议 2onvif协议
snCode	string	非必须	设备编码

名称	类型	是否必须	备注
activeMode	integer	非必须	是否支持主动模式 1 主动模式 0 非主动 默认0
isSupportActiveSdk	integer	非必须	是否支持activeSdk 1支持 0不支持
cmdStatus	integer	非必须	常开常闭状态码 17 正常 15 常开 16 常闭

1.8.4. HTTP请求示例

请求 path

```
/openapi/v1/device/queryDeviceInfo
```

请求 body

```
{
  "uuid": "eabe5bbbd4e24faba672aa45d09a6607"
}
```

1.8.5. HTTP响应示例

响应 200

```
{
  "code": 0,
  "data": {
    "activeMode": 0,
    "cmdStatus": 17,
    "deviceMode": 4,
    "deviceModel": 12,
    "deviceType": 1,
    "discoveryMethod": 2,
    "ipAddress": "10.171.4.50",
    "isSupportActiveSdk": 1,
    "location": "OpenAPI测试",
    "name": "智能摄像头-MegEye-C4R-322-人脸抓拍",
    "port": 3000,
    "rtspUrl": "rtsp://admin:admin123@10.171.4.50:554/1/1",
    "userName": "admin",
    "userPassword": "admin123",
    "uuid": "348847fb76c845918cb22d7a99f67666"
  },
  "msg": "成功"
}
```

1.9. 查询设备详情

1.9.1. 基本信息

Path: /v1/api/device/query

Method: POST

接口描述: 查询设备详细信息接口，包含设备全部信息；

1.9.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
uuid	string	必须	设备uuid

1.9.3. 返回数据

1.9.4. HTTP请求示例

请求 path

```
/openapi/v1/device/list
```

请求 body

```
{
  "uuid": "6b9a32333c4c42a1912d2fb31b38d01d"
}
```

1.9.5. HTTP响应示例

响应 200

```

{
  "code": 0,
  "data": {
    "activeMode": 0,
    "bindDevices": [
      {
        "uuid": "754332333c4c42a1912d2fb31b381122",
        "name": "相机设备",
        "deviceType": 1,
        "status": 4
      }
    ],
    "cmdStatus": 17,
    "deviceType": 3,
    "ipAddress": "10.171.2.15",
    "isSupportActiveSdk": 1,
    "location": "OpenAPI测试",
    "name": "智能服务器-MegBox-B3R-1012",
    "protocol": 1,
    "status": 3,
    "uuid": "6b9a32333c4c42a1912d2fb31b38d01d"
  },
  "msg": "成功"
}

```

1.10. 查询算力设备可绑定的端设备列表（Core）

1.10.1. 基本信息

Path: /v1/api/device/list/enbind/core/group

Method: POST

接口描述: 算力设备为Core时，查询当前设备可以绑定的端设备列表信息（包含可绑定与已绑定设备），按照服务器模式划分数据；

1.10.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
uuid	string	必须	设备uuid（只允许传Core设备的uuid）

1.10.3. 返回数据

名称	类型	是否必须	备注
Object	object []	非必须	
 serverType	integer	非必须	服务器模式 2 结构化模式 3 警戒模式 4 人脸抓拍模式 6 人脸识别标准模式 7 人脸识别高性能模式
 serverName	string	非必须	服务器名字
 serverList	object []	非必须	设备类型信息
 deviceType	integer	非必须	设备类型 1 智能摄像头 4 网络摄像机
 deviceList	object []	非必须	设备列表信息
uuid 	string	非必须	设备uuid
name 	string	非必须	设备名称
bindType 	integer	非必须	是否绑定设备 0-未绑定 1-绑定

1.10.4. HTTP请求示例

请求 path

```
/v1/api/device/list/enbind/core/group
```

请求 body

```
{
  "uuid": "6b9a32333c4c42a1912d2fb31b38d01d"
}
```

1.10.5. HTTP响应示例

响应 200


```

{
  "code": 0,
  "data": [
    {
      "serverList": [
        {
          "deviceList": [
            {
              "bindType": 1,
              "name": "网络摄像机-MegEye-C4H-142-结构化",
              "uuid": "81d65fe049fa48afa50afd42559c3581"
            }
          ],
          "deviceType": 4
        }
      ],
      "serverName": "结构化模式",
      "serverType": 2
    },
    {
      "serverList": [
        {
          "deviceList": [
            {
              "bindType": 0,
              "name": "网络摄像机-MegEye-C4S-222-警戒",
              "uuid": "4cb7507b930946b99319823f4b6ef0f8"
            }
          ],
          "deviceType": 4
        }
      ],
      "serverName": "警戒模式",
      "serverType": 3
    },
    {
      "serverList": [
        {
          "deviceList": [
            {
              "bindType": 1,
              "name": "智能摄像头-MegEye-C3S-123-人脸抓拍",
              "uuid": "f75deadcaf8f4a69a7753b7f830cc756"
            }
          ],
          "deviceType": 1
        }
      ],
      "serverName": "人脸抓拍模式",
      "serverType": 4
    },
    {
      "serverList": [
        {
          "deviceList": [
            {
              "bindType": 0,
              "name": "网络摄像机-MegEye-C4R-122-人脸识别",
              "uuid": "cb57d05ebd4e403c8f72ba92aa7eb4e5"
            }
          ],
          "deviceType": 4
        }
      ],
      "serverName": "人脸识别高性能模式",
    }
  ]
}

```

```
        "serverType": 7
    }
  ],
  "msg": "成功"
}
```

1.11. 查询算力设备可绑定的端设备列表（非Core）

1.11.1. 基本信息

Path: /v1/api/device/list/enbind/group

Method: POST

接口描述: 算力设备为非Core（B3R等）设备时，查询此算力设备可绑定的设备列表（包含可绑定与已绑定设备）；

1.11.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
uuid	string	必须	设备uuid（仅可传递非Core的算力设备uuid）

1.11.3. 返回数据

名称	类型	是否必须	备注
Object	object	非必须	
└ deviceType	integer	非必须	设备类型 1 智能摄像头 4 网络摄像机
└ deviceList	object	非必须	设备列表信息
└─ uuid	string	非必须	设备uuid
name └	string	非必须	设备名称
bindType └	integer	非必须	是否绑定设备 0-未绑定 1-绑定

1.11.4. HTTP请求示例

请求 **path**

```
/v1/api/device/list/enbind/group
```

请求 **body**

```
{  
  "uuid": "6b9a32333c4c42a1912d2fb31b38d01d"  
}
```

1.11.5. HTTP响应示例

响应 **200**

```

{
  "code": 0,
  "data": [
    {
      "deviceList": [
        {
          "bindType": 0,
          "name": "智能摄像头-MegEye-C3S-123-人脸抓拍",
          "uuid": "3ff7113b697244ce9f203114e4ce22df"
        },
        {
          "bindType": 0,
          "name": "智能摄像头-MegEye-C4H-142-人脸抓拍",
          "uuid": "1bbd8d1fa8de454eb90eaacef53654b4"
        },
        {
          "bindType": 0,
          "name": "智能摄像头-MegEye-C4S-222-人脸抓拍",
          "uuid": "21b049c1987d45c686f2d85b7eb23ce4"
        },
        {
          "bindType": 0,
          "name": "智能摄像头-MegEye-C4H-242-人脸抓拍",
          "uuid": "3a2fb9a7b24246c6bbc6a1c86833e386"
        }
      ],
      "deviceType": 1
    },
    {
      "deviceList": [
        {
          "bindType": 0,
          "name": "网络摄像机-MegEye-C4H-142-人脸识别",
          "uuid": "52c9c2d2310e4aa2b480650c59e66818"
        },
        {
          "bindType": 0,
          "name": "网络摄像机-IPC-人脸识别",
          "uuid": "c23a575a536e48f3b5e1cb249adf15fc"
        }
      ],
      "deviceType": 4
    }
  ],
  "msg": "成功"
}

```

1.12. 按设备类型分组查询设备列表（不建议使用）

1.12.1. 基本信息

Path: /v1/api/device/listBygroup

Method: POST

接口描述: 按照设备类型分组查询设备列表信息，考虑到使用场景非常局限，后期将此接口不建议使用。使用/v1/api/device/list代替

1.12.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

1.12.3. 返回数据

名称	类型	是否必须	备注
	object []	非必须	
 deviceType	integer	非必须	设备类型 1 智能摄像头 2 智能面板机 3 智能服务器 4 网络摄像机 5 存储服务器
 deviceList	object []	非必须	设备列表信息
uuid 	string	非必须	设备uuid
name 	string	非必须	设备名称

1.12.4. HTTP请求示例

请求 path

```
/v1/api/device/listBygroup
```

请求 body

```
{}
```

1.12.5. HTTP响应示例

响应 200

```

{
  "code": 0,
  "data": [
    {
      "deviceList": [],
      "deviceType": 1
    },
    {
      "deviceList": [],
      "deviceType": 2
    },
    {
      "deviceList": [
        {
          "name": "CORE",
          "uuid": "15d988e28956461fb74bb4ad22bfd5f"
        }
      ],
      "deviceType": 3
    },
    {
      "deviceList": [],
      "deviceType": 4
    },
    {
      "deviceList": [],
      "deviceType": 5
    }
  ],
  "msg": "成功"
}

```

1.13. 查询设备型号关系（不建议使用）

1.13.1. 基本信息

Path: /v1/api/device/queryModelRelation

Method: POST

接口描述: 可以查询系统所有设备型号支持的设备模式、是否支持主动模式等信息，考虑到此接口过于依赖业务，后期将不建议使用；

1.13.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
relationType	integer	必须	映射类型 1: 返回全部映射关系 2: 返回除智能面板机外的其他四种设备/服务器映射关系 3: 返回智能面板机的映射关系

1.13.3. 返回数据

名称	类型	是否必须	备注
	object []	非必须	
└─ deviceType	integer	非必须	设备类型id, 1 智能摄像头 2 智能面板机 3 智能服务器 4 网络摄像机 5 存储服务器
└─ deviceTypeStr	string	非必须	类型名称 智能摄像头 智能面板机 智能服务器 网络摄像机 存储服务器
└─ modelList	object []	非必须	设备型号和模型关系
└─ deviceModel	integer	非必须	设备型号id
└─ deviceModelStr	string	非必须	设备型号名称
└─ isSupportActiveSdk	integer	非必须	是否支持avtiveSDK, 0 不支持avtiveSDK, 1支持avtiveSDK
└─ modeList	object []	非必须	设备模式
└─ deviceMode	integer	非必须	模式id(服务器无模式) 1-人脸识别 2- 结构化 3- 警戒 4-人脸抓拍
└─ deviceModeStr	string	非必须	模式名称 人脸识别 结构化 警戒 人脸抓拍

1.13.4. HTTP请求示例

请求 path

```
/openapi/v1/device/list
```

请求 **body**

```
{  
  "relationType": 1  
}
```

1.13.5. HTTP响应示例

响应 **200**


```

{
  "code": 0,
  "data": [
    {
      "deviceType": 1,
      "deviceTypeStr": "智能摄像头",
      "modelList": [
        {
          "deviceModel": 4,
          "deviceModelStr": "MegEye-C3S-123",
          "isSupportActiveSdk": 1,
          "modelList": [
            {
              "deviceMode": 4,
              "deviceModeStr": "人脸抓拍模式"
            }
          ]
        },
        {
          "deviceModel": 5,
          "deviceModelStr": "MegEye-C4H-141",
          "isSupportActiveSdk": 1,
          "modelList": [
            {
              "deviceMode": 4,
              "deviceModeStr": "人脸抓拍模式"
            }
          ]
        },
        {
          "deviceModel": 6,
          "deviceModelStr": "MegEye-C4H-241",
          "isSupportActiveSdk": 1,
          "modelList": [
            {
              "deviceMode": 4,
              "deviceModeStr": "人脸抓拍模式"
            },
            {
              "deviceMode": 3,
              "deviceModeStr": "警戒模式"
            }
          ]
        },
        {
          "deviceModel": 7,
          "deviceModelStr": "MegEye-C4N-888",
          "isSupportActiveSdk": 0,
          "modelList": [
            {
              "deviceMode": 2,
              "deviceModeStr": "结构化模式"
            }
          ]
        },
        {
          "deviceModel": 8,
          "deviceModelStr": "MegEye-C4S-222",
          "isSupportActiveSdk": 1,
          "modelList": [
            {
              "deviceMode": 4,
              "deviceModeStr": "人脸抓拍模式"
            }
          ]
        }
      ]
    }
  ]
}

```

```

    },
    {
      "deviceModel": 9,
      "deviceModelStr": "MegEye-C4S-322",
      "isSupportActiveSdk": 1,
      "modelList": [
        {
          "deviceMode": 4,
          "deviceModeStr": "人脸抓拍模式"
        }
      ]
    },
    {
      "deviceModel": 10,
      "deviceModelStr": "MegEye-C4R-122",
      "isSupportActiveSdk": 1,
      "modelList": [
        {
          "deviceMode": 4,
          "deviceModeStr": "人脸抓拍模式"
        }
      ]
    },
    {
      "deviceModel": 11,
      "deviceModelStr": "MegEye-C4R-222",
      "isSupportActiveSdk": 1,
      "modelList": [
        {
          "deviceMode": 4,
          "deviceModeStr": "人脸抓拍模式"
        },
        {
          "deviceMode": 3,
          "deviceModeStr": "警戒模式"
        }
      ]
    },
    {
      "deviceModel": 12,
      "deviceModelStr": "MegEye-C4R-322",
      "isSupportActiveSdk": 1,
      "modelList": [
        {
          "deviceMode": 4,
          "deviceModeStr": "人脸抓拍模式"
        },
        {
          "deviceMode": 3,
          "deviceModeStr": "警戒模式"
        }
      ]
    },
    {
      "deviceModel": 29,
      "deviceModelStr": "海康摄像头",
      "isSupportActiveSdk": 0,
      "modelList": [
        {
          "deviceMode": 4,
          "deviceModeStr": "人脸抓拍模式"
        }
      ]
    },
    {
      "deviceModel": 30,

```

```

        "deviceModelStr": "大华摄像头",
        "isSupportActiveSdk": 0,
        "modelList": [
            {
                "deviceMode": 4,
                "deviceModeStr": "人脸抓拍模式"
            }
        ]
    },
    {
        "deviceModel": 31,
        "deviceModelStr": "宇视摄像头",
        "isSupportActiveSdk": 0,
        "modelList": [
            {
                "deviceMode": 4,
                "deviceModeStr": "人脸抓拍模式"
            }
        ]
    },
    {
        "deviceModel": 32,
        "deviceModelStr": "科大摄像头",
        "isSupportActiveSdk": 0,
        "modelList": [
            {
                "deviceMode": 4,
                "deviceModeStr": "人脸抓拍模式"
            }
        ]
    },
    {
        "deviceModel": 33,
        "deviceModelStr": "华为摄像头",
        "isSupportActiveSdk": 0,
        "modelList": [
            {
                "deviceMode": 4,
                "deviceModeStr": "人脸抓拍模式"
            }
        ]
    }
]
},
{
    "deviceType": 2,
    "deviceTypeStr": "智能面板机",
    "modelList": [
        {
            "deviceModel": 1,
            "deviceModelStr": "MegEye-W5K-I8",
            "isSupportActiveSdk": 0,
            "modelList": [
                {
                    "deviceMode": 1,
                    "deviceModeStr": "人脸识别模式"
                }
            ]
        }
    ]
},
{
    "deviceModel": 2,
    "deviceModelStr": "MegEye-W4K-E7",
    "isSupportActiveSdk": 0,
    "modelList": [
        {
            "deviceMode": 1,

```

```

        "deviceModeStr": "人脸识别模式"
    }
}
},
{
    "deviceModel": 3,
    "deviceModelStr": "MegEye-W4K-I51",
    "isSupportActiveSdk": 0,
    "modelList": [
        {
            "deviceMode": 1,
            "deviceModeStr": "人脸识别模式"
        }
    ]
},
{
    "deviceModel": 34,
    "deviceModelStr": "MegEye-W4K-I52",
    "isSupportActiveSdk": 0,
    "modelList": [
        {
            "deviceMode": 1,
            "deviceModeStr": "人脸识别模式"
        }
    ]
},
{
    "deviceModel": 35,
    "deviceModelStr": "MegEye-W3K-45S",
    "isSupportActiveSdk": 0,
    "modelList": [
        {
            "deviceMode": 1,
            "deviceModeStr": "人脸识别模式"
        }
    ]
},
{
    "deviceModel": 36,
    "deviceModelStr": "MegEye-W4K-I31",
    "isSupportActiveSdk": 0,
    "modelList": [
        {
            "deviceMode": 1,
            "deviceModeStr": "人脸识别模式"
        }
    ]
}
]
},
{
    "deviceType": 3,
    "deviceTypeStr": "智能服务器",
    "modelList": [
        {
            "deviceModel": 18,
            "deviceModelStr": "CORE",
            "isSupportActiveSdk": 0,
            "modelList": []
        },
        {
            "deviceModel": 22,
            "deviceModelStr": "MegBox-B3R-412",
            "isSupportActiveSdk": 1,
            "modelList": []
        }
    ]
},

```

```

    {
      "deviceModel": 23,
      "deviceModelStr": "MegBox-B3R-1012",
      "isSupportActiveSdk": 1,
      "modelList": []
    },
    {
      "deviceModel": 37,
      "deviceModelStr": "MegBox-B3R-1012(H1)",
      "isSupportActiveSdk": 1,
      "modelList": [
        {
          "deviceMode": 1,
          "deviceModeStr": "人脸识别模式"
        },
        {
          "deviceMode": 5,
          "deviceModeStr": "混合模式"
        }
      ]
    },
    {
      "deviceModel": 39,
      "deviceModelStr": "MegCube-B4H16-311",
      "isSupportActiveSdk": 1,
      "modelList": [
        {
          "deviceMode": 5,
          "deviceModeStr": "混合模式"
        }
      ]
    }
  ],
  {
    "deviceType": 4,
    "deviceTypeStr": "网络摄像机",
    "modelList": [
      {
        "deviceModel": 4,
        "deviceModelStr": "MegEye-C3S-123",
        "isSupportActiveSdk": 1,
        "modelList": [
          {
            "deviceMode": 2,
            "deviceModeStr": "结构化模式"
          },
          {
            "deviceMode": 3,
            "deviceModeStr": "警戒模式"
          },
          {
            "deviceMode": 1,
            "deviceModeStr": "人脸识别模式"
          }
        ]
      }
    ]
  },
  {
    "deviceModel": 5,
    "deviceModelStr": "MegEye-C4H-141",
    "isSupportActiveSdk": 1,
    "modelList": [
      {
        "deviceMode": 2,
        "deviceModeStr": "结构化模式"
      }
    ]
  }
]

```

```

        {
            "deviceMode": 3,
            "deviceModeStr": "警戒模式"
        },
        {
            "deviceMode": 1,
            "deviceModeStr": "人脸识别模式"
        }
    ]
},
{
    "deviceModel": 6,
    "deviceModelStr": "MegEye-C4H-241",
    "isSupportActiveSdk": 1,
    "modelList": [
        {
            "deviceMode": 2,
            "deviceModeStr": "结构化模式"
        },
        {
            "deviceMode": 3,
            "deviceModeStr": "警戒模式"
        },
        {
            "deviceMode": 1,
            "deviceModeStr": "人脸识别模式"
        }
    ]
},
{
    "deviceModel": 7,
    "deviceModelStr": "MegEye-C4N-888",
    "isSupportActiveSdk": 0,
    "modelList": [
        {
            "deviceMode": 2,
            "deviceModeStr": "结构化模式"
        },
        {
            "deviceMode": 3,
            "deviceModeStr": "警戒模式"
        },
        {
            "deviceMode": 1,
            "deviceModeStr": "人脸识别模式"
        }
    ]
},
{
    "deviceModel": 8,
    "deviceModelStr": "MegEye-C4S-222",
    "isSupportActiveSdk": 1,
    "modelList": [
        {
            "deviceMode": 2,
            "deviceModeStr": "结构化模式"
        },
        {
            "deviceMode": 3,
            "deviceModeStr": "警戒模式"
        },
        {
            "deviceMode": 1,
            "deviceModeStr": "人脸识别模式"
        }
    ]
}
]

```

```

    },
    {
      "deviceModel": 9,
      "deviceModelStr": "MegEye-C4S-322",
      "isSupportActiveSdk": 1,
      "modelList": [
        {
          "deviceMode": 2,
          "deviceModeStr": "结构化模式"
        },
        {
          "deviceMode": 3,
          "deviceModeStr": "警戒模式"
        },
        {
          "deviceMode": 1,
          "deviceModeStr": "人脸识别模式"
        }
      ]
    },
    {
      "deviceModel": 10,
      "deviceModelStr": "MegEye-C4R-122",
      "isSupportActiveSdk": 1,
      "modelList": [
        {
          "deviceMode": 2,
          "deviceModeStr": "结构化模式"
        },
        {
          "deviceMode": 3,
          "deviceModeStr": "警戒模式"
        },
        {
          "deviceMode": 1,
          "deviceModeStr": "人脸识别模式"
        }
      ]
    },
    {
      "deviceModel": 11,
      "deviceModelStr": "MegEye-C4R-222",
      "isSupportActiveSdk": 1,
      "modelList": [
        {
          "deviceMode": 2,
          "deviceModeStr": "结构化模式"
        },
        {
          "deviceMode": 3,
          "deviceModeStr": "警戒模式"
        },
        {
          "deviceMode": 1,
          "deviceModeStr": "人脸识别模式"
        }
      ]
    },
    {
      "deviceModel": 12,
      "deviceModelStr": "MegEye-C4R-322",
      "isSupportActiveSdk": 1,
      "modelList": [
        {
          "deviceMode": 2,
          "deviceModeStr": "结构化模式"
        }
      ]
    }
  ]
}

```

```

        },
        {
            "deviceMode": 3,
            "deviceModeStr": "警戒模式"
        },
        {
            "deviceMode": 1,
            "deviceModeStr": "人脸识别模式"
        }
    ]
},
{
    "deviceModel": 21,
    "deviceModelStr": "IPC",
    "isSupportActiveSdk": 0,
    "modelList": [
        {
            "deviceMode": 2,
            "deviceModeStr": "结构化模式"
        },
        {
            "deviceMode": 3,
            "deviceModeStr": "警戒模式"
        },
        {
            "deviceMode": 1,
            "deviceModeStr": "人脸识别模式"
        }
    ]
}
],
{
    "deviceType": 5,
    "deviceTypeStr": "存储服务器",
    "modelList": [
        {
            "deviceModel": 24,
            "deviceModelStr": "MegNVR-N4K-001",
            "isSupportActiveSdk": 0,
            "modelList": []
        },
        {
            "deviceModel": 25,
            "deviceModelStr": "MegNVR-N4M-001",
            "isSupportActiveSdk": 0,
            "modelList": []
        },
        {
            "deviceModel": 26,
            "deviceModelStr": "MegNVR-N8K-281",
            "isSupportActiveSdk": 0,
            "modelList": []
        },
        {
            "deviceModel": 27,
            "deviceModelStr": "MegNVR-N8M-281",
            "isSupportActiveSdk": 0,
            "modelList": []
        },
        {
            "deviceModel": 28,
            "deviceModelStr": "MegNVR-N8P-281",
            "isSupportActiveSdk": 0,
            "modelList": []
        }
    ]
}

```



```
    ]
  }
],
"msg": "成功"
}
```

1.14. 设备绑定

1.14.1. 基本信息

Path: /v1/api/device/bind

Method: POST

接口描述: 算力设备/存储设备与端设备进行绑定，绑定后可进行布控等操作，是全量绑定，已绑定的设备和入参进行差集运算，得到需要解绑的设备和需要绑定的设备；

1.14.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
uuid	string	必须	设备uuid（算力设备/存储设备uuid）
deviceUids	string []	非必须	端设备uuid（非Core设备绑定时传递此值）
└		非必须	
serverModeToDeviceUuidReqs	object []	非必须	绑定的端设备信息集合（core设备绑定需要关联服务器模式，所以core设备绑定时此参数必须，如果不是绑定core，不要传递此参数，注意：此参数有值，会忽略deviceUids字段的值）
└ deviceUuid	string	必须	绑定的设备uuid
└ serverMode	integer	必须	服务器类型 2 结构化模式 3 警戒模式 4 人脸抓拍模式 6 人脸识别标准模式 7 人脸识别高性能模式

1.14.3. 返回数据

OK

1.14.4. HTTP请求示例

请求 path

/v1/api/device/bind

请求 body

Core设备绑定请求body

```
{
  "uuid": "15d988e28956461fb74bb4ad22bfd5f",
  "serverModeToDeviceUuidReqs": [
    {
      "deviceUuid": "fc4c7c24120b4e28af18f34347ac234b",
      "serverMode": 4
    }
  ]
}
```

非Core设备绑定请求body

```
{
  "uuid": "353ac229d99a45369da81246c483d654",
  "deviceUuids": [
    "652adc10cdf44c85b54e048b922ffa7f"
  ]
}
```

1.14.5. HTTP响应示例

响应 200

```
{
  "code": 0,
  "msg": "成功"
}
```

1.15. 解绑设备预校验

1.15.1. 基本信息

Path: /v1/api/device/preUnbind

Method: POST

接口描述: 算力设备/存储设备与端设备进行解绑需要调用此接口进行预校验，判断是否可以解绑，支持批量校验。

1.15.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
computeUuid	string	必须	算力设备uuid/存储设备uuid
deviceUulds	string []	必须	设备uuid集合
└─		非必须	
singleCheck	boolean	必须	单一解绑 true 批量解绑校验 false

1.15.3. 返回数据

名称	类型	是否必须	备注
tag	integer	非必须	标识 0-可解绑 1-已绑定 2-已布控 3-强制删除
deployType	integer	非必须	布控类型 1-警戒布控 2-结构化布控 3-门禁点布控 4-事件联动布控 5-员工、访客、重点人员布控 6-车辆布控 7-全局陌生人员布控（当tag为2时存在值）

1.15.4. HTTP请求示例

请求 path

```
/v1/api/device/preUnbind
```

请求 body

```
{
  "computeUuid": "6fe4345dd07c466da847155fc8eecb5e",
  "deviceUuIds": [
    "0c1fe75ccba64b06997f5958c1b1d9f0",
    "25c12cfd25714d16a9ddb8224315b2b5",
    "bc8cae66dc6843dd961d4ac902f4a2d9"
  ],
  "singleCheck": false
}
```

1.15.5. HTTP响应示例

响应 200

```
{
  "code": 0,
  "data": {
    "tag": 0
  },
  "msg": "成功"
}
```

1.16. 设备解绑

1.16.1. 基本信息

Path: /v1/api/device/unbind

Method: POST

接口描述: 算力设备/存储设备与端设备进行解绑操作（单一解绑），已布控设备不允许解绑；

1.16.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
uuid	string	必须	设备uuid
unBindDeviceUuid	string	必须	被解绑设备id

1.16.3. 返回数据

OK

1.16.4. HTTP请求示例

请求 path

/v1/api/device/unbind

请求 body

```
{
  "uuid": "15ca93618388448ab1afc9115d5a580b",
  "unBindDeviceUuid": "bd022d30511849a29326e51d5959b757"
}
```

1.16.5. HTTP响应示例

响应 200

```
{
  "code": 0,
  "msg": "成功"
}
```

1.17. 门禁、人员管控设备列表查询

1.17.1. 基本信息

Path: /v1/api/device/control/list

Method: POST

接口描述: 查询可添加、已添加的门禁/人员管控设备列表，方便进行添加门禁/人员管控设备；

1.17.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
location	string	非必须	设备位置
pageNum	integer	非必须	当前页码,不传默认1
name	string	非必须	设备名称
pageSize	integer	非必须	分页大小，不传默认20
deviceType	integer	非必须	设备类型 1、智能摄像头 2、智能面板机 4、网络摄像机
deviceModel	integer	非必须	设备型号
queryFlag	integer	非必须	可添加设备查询标识 0 -查询可绑定的列表 1 查询已绑定的设备 不传递查询全部
flagType	integer	必须	查询标识 0 - 查询门禁点设备 1 - 查询人员管控设备

1.17.3. 返回数据

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码
pageSize	integer	非必须	分页大小
total	integer	非必须	总数
list	object []	非必须	
└─ deviceUuid	string	非必须	设备uuid（不包含算力设备与存储设备）
└─ location	string	非必须	设备位置
└─ name	string	非必须	设备名称
└─ deviceType	integer	非必须	设备类型 1 智能摄像头 2 智能面板机 4 网络摄像机
└─ deviceModel	integer	非必须	设备型号（不包含算力设备与存储设备）
└─ status	integer	非必须	设备状态 3、在线 4、离线
└─ switchIp	string	非必须	绑定开关,通行设备列表中使用
└─ isPass	integer	非必须	是否绑定门禁设备 0、不是 1、是
└─ personControlFlag	integer	非必须	人员管控设备标识 0、不是 1、是

1.17.4. HTTP请求示例

请求 path

```
/v1/api/device/control/list
```


请求 body

```
{
  "queryFlag": 1,
  "flagType": 1
}
```

1.17.5. HTTP响应示例

响应 200

```
{
  "code": 0,
  "data": {
    "list": [
      {
        "deviceModel": 4,
        "deviceType": 1,
        "deviceUuid": "9d156e7a817a4db99361d9ffe424f15b",
        "isPass": 0,
        "location": "OpenAPI测试",
        "name": "智能摄像头-MegEye-C3S-123-人脸抓拍",
        "personControlFlag": 1,
        "status": 3
      },
      {
        "deviceModel": 5,
        "deviceType": 4,
        "deviceUuid": "926fc3ad2ab94181b6cd90e11eb09144",
        "isPass": 0,
        "location": "OpenAPI测试",
        "name": "网络摄像机-MegEye-C4H-142-人脸识别",
        "personControlFlag": 1,
        "status": 3
      }
    ],
    "pageNum": 1,
    "pageSize": 20,
    "total": 2
  },
  "msg": "成功"
}
```

1.18. 添加门禁、人员管控设备

1.18.1. 基本信息

Path: /v1/api/device/control/add

Method: POST

接口描述: 将已绑定的设备添加至门禁/人员管控设备，添加完成后设备方可进行通行、安防布控；

1.18.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
removeDeviceUuids	string []	必须	移除人员管控设备id
└─		非必须	
addDeviceUuids	string []	必须	添加人员管控设备id
└─		非必须	

1.18.3. 返回数据

OK

1.18.4. HTTP请求示例

请求 path

/v1/api/device/control/add

请求 body

<pre>{ "removeDeviceUuids": [], "addDeviceUuids": ["de5a5008983e4632bcd1d2d07e16a044", "41d1feb198154927b6bade54ec12482"] }</pre>

1.18.5. HTTP响应示例

响应 200

<pre>{ "code": 0, "msg": "成功" }</pre>

1.19. 移除门禁、人员管控设备

1.19.1. 基本信息

Path: /v1/api/device/control/remove

Method: POST

接口描述: 将设备从门禁/人员管控设备中移除，已布控设备可强制移除；

1.19.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
uuid	string	必须	设备uuid

1.19.3. 返回数据

OK

HTTP请求示例

请求 path

/v1/api/device/control/remove

请求 body

```
{
  "uuid": "95350c2ebf104c3d8a936d2bf39b296e"
}
```

1.19.4. HTTP响应示例

响应 200

```
{
  "code": 0,
  "msg": "成功"
}
```

1.20. 设备可布控列表查询

1.20.1. 基本信息

Path: /v1/api/device/queryDeviceControl

Method: POST

接口描述: 根据设备模式与布控类型查询设备列表信息，可用于查询可以进行布控安防、通行、警戒、结构化的设备列表；

1.20.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
deviceModes	integer []	必须	设备模式 1-人脸识别 2- 结构化 3- 警戒 4-人脸抓拍
└		非必须	
queryType	integer	非必须	查询类型 0-查询全部（不包含算力、存储设备） 1-查询全局陌生人员布控，但是面板机要求是有门禁权限 2-查询员工、访客、重点人员布控

1.20.3. 返回数据

名称	类型	是否必须	备注
	object []	非必须	
uuid	string	非必须	设备uuid
name	string	非必须	设备名称
rtspUrl	string	非必须	视频流地址
type	integer	非必须	是否是all-in-one模式(type=0)下面的警戒模式还是sensor+compute的警戒模式(type=1),其他模式(type=2)
activeMode	integer	非必须	是否支持主动模式 1 主动模式
snCode	string	非必须	设备sn码

1.20.4. HTTP请求示例

请求 path

```
/v1/api/device/queryDeviceControl
```

请求 body

```
{
  "deviceModes": [
    1,
    4
  ],
  "queryType": 2
}
```

1.20.5. HTTP响应示例

响应 200

```
{
  "code": 0,
  "data": [
    {
      "activeMode": 1,
      "uuid": "95350c2ebf104c3d8a936d2bf39b296e",
      "name": "C4S222PDNS",
      "rtspUrl": "rtsp://admin:admin123@192.168.0.222:554/1/1",
      "snCode": "ID0000801941431621590738",
      "type": 2
    },
    {
      "activeMode": 0,
      "uuid": "11110c2ebf104c3d8a936d2bf39b3r4r",
      "name": "Core模拟视频2",
      "rtspUrl": "rtsp://10.122.94.220:8554/jinyu_mulit_19/20210115_02.ts",
      "type": 1
    }
  ],
  "msg": "成功"
}
```

- 1. 时间计划管理
 - 1.1. 添加时间计划集合
 - 1.1.1. 基本信息
 - 1.1.2. 请求参数
 - 1.1.3. 返回数据
 - 1.1.4. HTTP请求示例
 - 1.1.5. HTTP响应示例
 - 1.2. 修改时间计划
 - 1.2.1. 基本信息
 - 1.2.2. 请求参数
 - 1.2.3. HTTP请求示例
 - 1.2.4. HTTP响应示例
 - 1.3. 批量删除时间计划
 - 1.3.1. 基本信息
 - 1.3.2. 请求参数
 - 1.3.3. 返回数据
 - 1.3.4. HTTP请求示例
 - 1.3.5. HTTP响应示例
 - 1.4. 时间计划列表
 - 1.4.1. 基本信息
 - 1.4.2. 请求参数
 - 1.4.3. 返回数据
 - 1.4.4. HTTP请求示例
 - 1.4.5. HTTP响应示例
 - 1.5. 查询时间计划详情列表
 - 1.5.1. 基本信息
 - 1.5.2. 请求参数
 - 1.5.3. 返回数据
 - 1.5.4. HTTP请求示例
 - 1.5.5. HTTP响应示例

1. 时间计划管理

1.1. 添加时间计划集合

1.1.1. 基本信息

Path: /v1/api/pass/schedule/addBatch

Method: POST

接口描述: 添加时间计划

1.1.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
scheduleList	object []	必须	时间计划添加集合
└─ uuid	string	非必须	uuid
└─ name	string	必须	时间计划名称,唯一, 长度1-32, 不支持特殊字符
└─ type	integer	必须	用途类型: 1 通行模块, 2 安防模块, 3 事件模块
└─ holiday	object []	非必须	节假日配置, 安防、事件时间计划不支持假日。 示例: [{"date":"2020-10-01","during":[{"from":"08:00","to":"10:00"}, {"from":"10:30","to":"14:00"}],"state":true}]
└─ date	string	必须	时间: '2020-01-01'
└─ state	boolean	必须	状态开关: false 关闭 true 开启
└─ during	object []	非必须	时间段数组, 如果state为true, 则该参数必须传, 且不能是空集合, 参数按照集合里面对象参数说明为准
└─ from	string	必须	开始时间, 格式: 00:00
└─ to	string	必须	截止时间, 格式: 23:59
└─ workDay	object []	必须	工作日配置 示例: [{"week":1,"during":[{"from":"08:00","to":"10:00"}, {"from":"10:30","to":"14:00"}]}]
└─ week	integer	必须	星期几, 1-7分别表示周一至周日
└─ during	object []	必须	时间段数组, 一天可以有多个时间段
└─ from	string	必须	开始时间, 时分, 如"00:00"
└─ to	string	必须	结束时间, 时分, 如"23:59"

1.1.3. 返回数据

名称	类型	是否必须	备注
successes	object []	非必须	成功列表
└─ code	integer	非必须	结果状态码
└─ msg	string	非必须	结果信息提示
└─ uuid	string	非必须	uuid
└─ name	string	非必须	时间计划名称
└─ type	integer	非必须	用途类型：1 通行模块, 2 安防模块, 3 事件模块
└─ holiday	object []	非必须	节假日配置，安防、事件时间计划不支持假日。 示例：[{"date":"2020-10-01","during":[{"from":"08:00","to":"10:00"}, {"from":"10:30","to":"14:00"}],"state":true}]
└─ date	string	非必须	时间：'2020-01-01'
└─ state	boolean	非必须	状态开关：false 关闭 true 开启
└─ during	object []	非必须	时间段数组，如果state为true，则该参数必须传，且不能是空集合,参数按照集合里面对象参数说明为准
└─ from	string	非必须	开始时间，格式：00:00
└─ to	string	非必须	截止时间，格式：23:59
└─ workDay	object []	非必须	工作日配置 示例：[{"week":1,"during":[{"from":"08:00","to":"10:00"}, {"from":"10:30","to":"14:00"}]}]
└─ week	integer	非必须	星期几，1-7分别表示周一至周日

名称	类型	是否必须	备注
└─ during	object []	非必须	时间段数组，一天可以有多个时间段
└─ from	string	非必须	开始时间，时分，如“00:00”
└─ to	string	非必须	结束时间，时分，如“23:59”
failures	object []	非必须	失败列表
└─ code	integer	非必须	结果状态码
└─ msg	string	非必须	结果信息提示
└─ uuid	string	非必须	uuid
└─ name	string	非必须	时间计划名称
└─ type	integer	非必须	用途类型：1 通行模块, 2 安防模块, 3 事件模块
└─ holiday	object []	非必须	节假日配置，安防、事件时间计划不支持假日。 示例：[{"date":"2020-10-01","during":[{"from":"08:00","to":"10:00"}, {"from":"10:30","to":"14:00"}],"state":true}]
└─ date	string	非必须	时间：'2020-01-01'
└─ state	boolean	非必须	状态开关：false 关闭 true 开启
└─ during	object []	非必须	时间段数组，如果state为true，则该参数必须传，且不能是空集合,参数按照集合里面对象参数说明为准
└─ from	string	非必须	开始时间，格式：00:00
└─ to	string	非必须	截止时间，格式：23:59

名称	类型	是否必须	备注
└─ workDay	object []	非必须	工作日配置 示例: [{"week":1,"during": [{"from":"08:00","to":"10:00"}, {"from":"10:30","to":"14:00"}]}
└─ week	integer	非必须	星期几, 1-7分别表示周一至周日
└─ during	object []	非必须	时间段数组, 一天可以有多个时间段
└─ from	string	非必须	开始时间, 时分, 如“00:00”
└─ to	string	非必须	结束时间, 时分, 如“23:59”

1.1.4. HTTP请求示例

请求 path

```
/v1/api/pass/schedule/addBatch
```

请求 body

```

{
  "scheduleList": {
    "name": "通行时间计划",
    "type": 1,
    "uuid": "passSchedule",
    "workDay": [
      {
        "week": 1,
        "during": [
          {
            "from": "08:00",
            "to": "10:00"
          },
          {
            "from": "10:30",
            "to": "14:00"
          },
          {
            "from": "15:01",
            "to": "20:00"
          }
        ]
      },
      {
        "week": 2,
        "during": [
          {
            "from": "08:00",
            "to": "10:00"
          },
          {
            "from": "10:30",
            "to": "20:00"
          }
        ]
      },
      {
        "week": 3,
        "during": [
          {
            "from": "08:00",
            "to": "20:00"
          }
        ]
      },
      {
        "week": 4,
        "during": [
          {
            "from": "08:00",
            "to": "10:00"
          },
          {
            "from": "10:30",
            "to": "14:00"
          },
          {
            "from": "15:01",
            "to": "20:00"
          }
        ]
      },
      {
        "week": 5,
        "during": [

```

```

        {
            "from": "08:00",
            "to": "10:00"
        },
        {
            "from": "10:30",
            "to": "19:00"
        }
    ]
},
"holiday": [
    {
        "date": "2020-10-01",
        "during": [
            {
                "from": "08:00",
                "to": "10:00"
            },
            {
                "from": "10:30",
                "to": "14:00"
            },
            {
                "from": "15:01",
                "to": "20:00"
            }
        ]
    },
    {
        "date": "2020-10-02",
        "during": [
            {
                "from": "08:00",
                "to": "10:00"
            },
            {
                "from": "10:30",
                "to": "20:00"
            }
        ]
    },
    {
        "date": "2020-10-03",
        "during": [
            {
                "from": "08:00",
                "to": "20:00"
            }
        ]
    }
]
}

```

1.1.5. HTTP响应示例

响应 200

```

{
  "code": 0,
  "data": {
    "failures": [],
    "successes": [
      {
        "holiday": [
          {
            "date": "2020-10-01",
            "during": [
              {
                "from": "08:00",
                "to": "10:00"
              },
              {
                "from": "10:30",
                "to": "14:00"
              },
              {
                "from": "15:01",
                "to": "20:00"
              }
            ],
            "state": true
          },
          {
            "date": "2020-10-02",
            "during": [
              {
                "from": "08:00",
                "to": "10:00"
              },
              {
                "from": "10:30",
                "to": "20:00"
              }
            ],
            "state": true
          },
          {
            "date": "2020-10-03",
            "during": [
              {
                "from": "08:00",
                "to": "20:00"
              }
            ],
            "state": true
          }
        ],
        "name": "通行时间计划",
        "type": 1,
        "uuid": "passSchedule",
        "workDay": [
          {
            "during": [
              {
                "from": "08:00",
                "to": "10:00"
              },
              {
                "from": "10:30",
                "to": "14:00"
              }
            ],
            "state": true
          }
        ]
      }
    ]
  }
}

```

```

        "from": "15:01",
        "to": "20:00"
    }
],
    "week": 1
},
{
    "during": [
        {
            "from": "08:00",
            "to": "10:00"
        },
        {
            "from": "10:30",
            "to": "20:00"
        }
    ],
    "week": 2
},
{
    "during": [
        {
            "from": "08:00",
            "to": "20:00"
        }
    ],
    "week": 3
},
{
    "during": [
        {
            "from": "08:00",
            "to": "10:00"
        },
        {
            "from": "10:30",
            "to": "14:00"
        },
        {
            "from": "15:01",
            "to": "20:00"
        }
    ],
    "week": 4
},
{
    "during": [
        {
            "from": "08:00",
            "to": "10:00"
        },
        {
            "from": "10:30",
            "to": "19:00"
        }
    ],
    "week": 5
}
]
}
},
    "msg": "成功"
}

```

1.2. 修改时间计划

1.2.1. 基本信息

Path: /v1/api/pass/schedule/update

Method: POST

接口描述: 修改时间计划

1.2.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
uuid	string	必须	uuid
name	string	必须	时间计划名称,唯一, 长度1-32, 不支持特殊字符
type	integer	必须	用途类型: 1 通行模块, 2 安防模块, 3 事件模块
holiday	object []	非必须	节假日配置, 安防、事件时间计划不支持节假日。 示例: [{"date":"2020-10-01","during":[{"from":"08:00","to":"10:00"}, {"from":"10:30","to":"14:00"}],"state":true}]
└─ date	string	必须	时间: '2020-01-01'
└─ state	boolean	必须	状态开关: false 关闭 true 开启
└─ during	object []	非必须	时间段数组, 如果state为true, 则该参数必须传, 且不能是空集合,参数按照集合里面对象参数说明为准
└─ from	string	必须	开始时间, 格式: 00:00
└─ to	string	必须	截止时间, 格式: 23:59
workDay	object []	必须	工作日配置 示例: [{"week":1,"during":[{"from":"08:00","to":"10:00"}, {"from":"10:30","to":"14:00"}]}]
└─ week	integer	必须	星期几, 1-7分别表示周一至周日
└─ during	object []	必须	时间段数组, 一天可以有多个时间段
└─ from	string	必须	开始时间, 时分, 如“00:00”
└─ to	string	必须	结束时间, 时分, 如“23:59”

1.2.3. HTTP请求示例

请求 path

```
/v1/api/pass/schedule/update
```

请求 body

```

{
  "name": "update通行时间计划",
  "type": 1,
  "uuid": "ff8dc9acb1b6471fbc1bf80fed24e17c",
  "workDay": [
    {
      "week": 1,
      "during": [
        {
          "from": "08:00",
          "to": "10:00"
        },
        {
          "from": "10:30",
          "to": "14:00"
        },
        {
          "from": "15:01",
          "to": "20:00"
        }
      ]
    },
    {
      "week": 2,
      "during": [
        {
          "from": "08:00",
          "to": "10:00"
        },
        {
          "from": "10:30",
          "to": "20:00"
        }
      ]
    },
    {
      "week": 3,
      "during": [
        {
          "from": "08:00",
          "to": "20:00"
        }
      ]
    },
    {
      "week": 4,
      "during": [
        {
          "from": "08:00",
          "to": "10:00"
        },
        {
          "from": "10:30",
          "to": "14:00"
        },
        {
          "from": "15:01",
          "to": "20:00"
        }
      ]
    },
    {
      "week": 5,
      "during": [
        {

```

```

        "from": "08:00",
        "to": "10:00"
    },
    {
        "from": "10:30",
        "to": "19:00"
    }
]
},
{
    "week": 6,
    "during": [
        {
            "from": "08:00",
            "to": "10:00"
        },
        {
            "from": "10:30",
            "to": "14:00"
        },
        {
            "from": "15:01",
            "to": "20:00"
        }
    ]
},
{
    "week": 7,
    "during": [
        {
            "from": "08:00",
            "to": "10:00"
        },
        {
            "from": "10:30",
            "to": "20:00"
        }
    ]
}
],
"holiday": [
    {
        "date": "2022-10-05",
        "during": [
            {
                "from": "08:00",
                "to": "10:00"
            },
            {
                "from": "10:30",
                "to": "14:00"
            },
            {
                "from": "15:01",
                "to": "20:00"
            }
        ]
    },
    {
        "date": "2024-08-22",
        "during": [
            {
                "from": "08:00",
                "to": "10:00"
            },
        ]
    }
]

```

```

        "from": "10:30",
        "to": "20:00"
      }
    ]
  },
  {
    "date": "2025-08-15",
    "during": [
      {
        "from": "08:00",
        "to": "20:00"
      }
    ]
  },
  {
    "date": "2023-08-08",
    "during": [
      {
        "from": "08:00",
        "to": "20:00"
      }
    ]
  }
]
}

```

1.2.4. HTTP响应示例

响应 200

```

{"code": 0, "msg": "成功"}

```

1.3. 批量删除时间计划

1.3.1. 基本信息

Path: /v1/api/pass/schedule/delete

Method: POST

接口描述: 删除时间计划

1.3.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
uuidList	string []	必须	时间计划uuid集合

1.3.3. 返回数据

名称	类型	是否必须	备注
successes	object []	非必须	成功列表
└─ uuid	string	非必须	uuid
└─ code	integer	非必须	结果状态码
└─ msg	string	非必须	结果信息提示
failures	object []	非必须	失败列表
└─ uuid	string	非必须	uuid
└─ code	integer	非必须	结果状态码
└─ msg	string	非必须	结果信息提示

1.3.4. HTTP请求示例

请求 **path**

```
/v1/api/pass/schedule/delete
```

请求 **body**

```
{
  "uuidList": [
    "faede9d7207e45d687cf2a5591fece52"
  ]
}
```

1.3.5. HTTP响应示例

响应 **200**

```
{
  "code": 0,
  "data": {
    "failures": [],
    "successes": [
      {
        "uuid": "faede9d7207e45d687cf2a5591fece52"
      }
    ]
  },
  "msg": "成功"
}
```

1.4. 时间计划列表

1.4.1. 基本信息

Path: /v1/api/pass/schedule/page

Method: POST

接口描述: 根据条件查询时间计划

1.4.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
pageNum	integer	非必须	分页页码，默认为 1
name	string	非必须	时间计划名称，支持模糊查询
pageSize	integer	非必须	分页大小，默认为 20
type	integer	必须	用途类型: 1 通行模块, 2 安防模块, 3 事件模块

1.4.3. 返回数据

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码
pageSize	integer	非必须	分页大小
total	integer	非必须	总数
list	object []	非必须	
├─ uuid	string	非必须	uuid
├─ name	string	非必须	时间计划名称
├─ type	integer	非必须	用途类型: 1 通行模块, 2 安防模块, 3 事件模块

1.4.4. HTTP请求示例

请求 path

/v1/api/pass/schedule/page

请求 body

```
{
  "type": 1,
  "name": "通行时间计划备用1",
  "pageNum": 1,
  "pageSize": 15
}
```

1.4.5. HTTP响应示例

响应 200

```
{
  "code": 0,
  "data": {
    "list": [
      {
        "name": "通行时间计划备用1",
        "type": 1,
        "uuid": "d77146e899424cb0970e4dbb3fd408b4"
      }
    ],
    "pageNum": 1,
    "pageSize": 15,
    "total": 1
  },
  "msg": "成功"
}
```

1.5. 查询时间计划详情列表

1.5.1. 基本信息

Path: /v1/api/pass/schedule/details

Method: POST

接口描述: 根据uuid查询时间计划详情

1.5.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
uuidList	string []	必须	时间计划uuid集合

1.5.3. 返回数据

名称	类型	是否必须	备注
	object []	非必须	
uuid	string	非必须	uuid
name	string	非必须	时间计划名称,唯一, 长度1-32, 不支持特殊字符
type	integer	非必须	用途类型: 1 通行模块, 2 安防模块, 3 事件模块
holiday	object []	非必须	节假日配置, 安防、事件时间计划不支持假日。 示例: [{"date":"2020-10-01","during":[{"from":"08:00","to":"10:00"}, {"from":"10:30","to":"14:00"}],"state":true}]
└─ date	string	非必须	时间: '2020-01-01'
└─ state	boolean	非必须	状态开关: false 关闭 true 开启
└─ during	object []	非必须	时间段数组, 如果state为true, 则该参数必须传, 且不能是空集合,参数按照集合里面对象参数说明为准
└─ from	string	非必须	开始时间, 格式: 00:00
└─ to	string	非必须	截止时间, 格式: 23:59
workDay	object []	非必须	工作日配置 示例: [{"week":1,"during":[{"from":"08:00","to":"10:00"}, {"from":"10:30","to":"14:00"}]}]
└─ week	integer	非必须	星期几, 1-7分别表示周一至周日
└─ during	object []	非必须	时间段数组, 一天可以有多个时间段
└─ from	string	非必须	开始时间, 时分, 如"00:00"

名称	类型	是否必须	备注
└ to	string	非必须	结束时间，时分，如“23:59”

1.5.4. HTTP请求示例

请求 path

```
/v1/api/pass/schedule/details
```

请求 body

```
{
  "uuidList": [
    "624a6ffde9684236b3629214938bc1a7"
  ]
}
```

1.5.5. HTTP响应示例

响应 200

```

{
  "code": 0,
  "data": [
    {
      "holiday": [
        {
          "date": "2020-10-01",
          "during": [
            {
              "from": "08:00",
              "to": "10:00"
            },
            {
              "from": "10:30",
              "to": "14:00"
            },
            {
              "from": "15:01",
              "to": "20:00"
            }
          ],
          "state": true
        },
        {
          "date": "2020-10-02",
          "during": [
            {
              "from": "08:00",
              "to": "10:00"
            },
            {
              "from": "10:30",
              "to": "20:00"
            }
          ],
          "state": true
        },
        {
          "date": "2020-10-03",
          "during": [
            {
              "from": "08:00",
              "to": "20:00"
            }
          ],
          "state": true
        }
      ],
      "name": "通行时间计划备用1",
      "type": 1,
      "uuid": "624a6ffde9684236b3629214938bc1a7",
      "workDay": [
        {
          "during": [
            {
              "from": "08:00",
              "to": "10:00"
            },
            {
              "from": "10:30",
              "to": "14:00"
            },
            {
              "from": "15:01",
              "to": "20:00"
            }
          ]
        }
      ]
    }
  ]
}

```

```

    }
  ],
  "week": 1
},
{
  "during": [
    {
      "from": "08:00",
      "to": "10:00"
    },
    {
      "from": "10:30",
      "to": "20:00"
    }
  ],
  "week": 2
},
{
  "during": [
    {
      "from": "08:00",
      "to": "20:00"
    }
  ],
  "week": 3
},
{
  "during": [
    {
      "from": "08:00",
      "to": "10:00"
    },
    {
      "from": "10:30",
      "to": "14:00"
    },
    {
      "from": "15:01",
      "to": "20:00"
    }
  ],
  "week": 4
},
{
  "during": [
    {
      "from": "08:00",
      "to": "10:00"
    },
    {
      "from": "10:30",
      "to": "19:00"
    }
  ],
  "week": 5
}
]
}
],
"msg": "成功"
}

```

- 1. 门禁点管理
 - 1.1. 查询绑定B3R的端设备（分页）
 - 1.1.1. 基本信息
 - 1.1.2. 请求参数
 - 1.1.3. 返回数据
 - 1.1.4. HTTP请求示例
 - 1.1.5. HTTP响应示例
 - 1.2. 查询门禁点设备列表信息（分页）
 - 1.2.1. 基本信息
 - 1.2.2. 请求参数
 - 1.2.3. 返回数据
 - 1.2.4. HTTP请求示例
 - 1.2.5. HTTP响应示例
 - 1.3. 查询同步失败人员信息
 - 1.3.1. 基本信息
 - 1.3.2. 请求参数
 - 1.3.3. 返回数据
 - 1.3.4. HTTP请求示例
 - 1.3.5. HTTP响应示例
 - 1.4. 查询绑定设备类型
 - 1.4.1. 基本信息
 - 1.4.2. 请求参数
 - 1.4.3. 返回数据
 - 1.4.4. HTTP请求示例
 - 1.4.5. HTTP响应示例
 - 1.5. 查询设备配置信息
 - 1.5.1. 基本信息
 - 1.5.2. 请求参数
 - 1.5.3. 返回数据
 - 1.5.4. HTTP请求示例
 - 1.5.5. HTTP响应示例
 - 1.6. 批量添加和移除门禁点
 - 1.6.1. 基本信息
 - 1.6.2. 请求参数
 - 1.6.3. 返回数据
 - 1.6.4. HTTP请求示例
 - 1.6.5. HTTP响应示例
 - 1.7. 移除门禁点
 - 1.7.1. 基本信息
 - 1.7.2. 请求参数
 - 1.7.3. 返回数据
 - 1.7.4. HTTP请求示例
 - 1.7.5. HTTP响应示例
 - 1.8. 绑定开关
 - 1.8.1. 基本信息
 - 1.8.2. 请求参数
 - 1.8.3. 返回数据
 - 1.8.4. HTTP请求示例

- 1.8.5. HTTP响应示例
- 1.9. 远程常开
 - 1.9.1. 基本信息
 - 1.9.2. 请求参数
 - 1.9.3. 返回数据
 - 1.9.4. HTTP请求示例
 - 1.9.5. HTTP响应示例
- 1.10. 远程常闭
 - 1.10.1. 基本信息
 - 1.10.2. 请求参数
 - 1.10.3. 返回数据
 - 1.10.4. HTTP请求示例
 - 1.10.5. HTTP响应示例
- 1.11. 远程开门
 - 1.11.1. 基本信息
 - 1.11.2. 请求参数
 - 1.11.3. 返回数据
 - 1.11.4. HTTP请求示例
 - 1.11.5. HTTP响应示例
- 1.12. 恢复正常
 - 1.12.1. 基本信息
 - 1.12.2. 请求参数
 - 1.12.3. 返回数据
 - 1.12.4. HTTP请求示例
 - 1.12.5. HTTP响应示例
- 1.13. 配置设备
 - 1.13.1. 基本信息
 - 1.13.2. 请求参数
 - 1.13.3. 返回数据
 - 1.13.4. HTTP请求示例
 - 1.13.5. HTTP响应示例
- 1.14. 查询同步列表（废弃）
 - 1.14.1. 基本信息
 - 1.14.2. 请求参数
 - 1.14.3. 返回数据
 - 1.14.4. HTTP请求示例
 - 1.14.5. HTTP响应示例
- 1.15. 同步失败详情（废弃）
 - 1.15.1. 基本信息
 - 1.15.2. 请求参数
 - 1.15.3. 返回数据
 - 1.15.4. HTTP请求示例
 - 1.15.5. HTTP响应示例

1. 门禁点管理

1.1. 查询绑定B3R的端设备（分页）

1.1.1. 基本信息

Path: /v1/api/door/queryB3rInBind

Method: POST

接口描述: 查询在绑定在同一B3R下的设备

1.1.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码,不传默认1
uuid	string	必须	当前设备uuid
pageSize	integer	非必须	分页大小, 不传默认20

1.1.3. 返回数据

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码
pageSize	integer	非必须	分页大小
total	integer	非必须	总数
list	object []	非必须	
└─uuid	string	非必须	设备uuid
└─location	string	非必须	设备位置
└─name	string	非必须	设备名称
└─deviceType	integer	非必须	设备类型 1、智能摄像头 2、智能面板机 4、网络摄像机
└─deviceModel	integer	非必须	设备型号 1、MegEye-W5K-I8 2、MegEye-W4K-E7 3、MegEye-W4K-I5 4、MegEye-C3S-123 5、MegEye-C4H-142 6、MegEye-C4H-242 7、MegEye-C4N-888 8、MegEye-C4S-222 9、MegEye-C4S-322 10、MegEye-C4R-122 11、MegEye-C4R-222 12、MegEye-C4R-322 13、MegEye-D1S-X2X 14、MegEye-D1S-X4X 15、MegID-W2K 16、MegID-W2K-E 17、MegEye-W3K-45S 21、IPC
└─deviceMode	integer	非必须	设备模式（服务器无） 1、人脸识别 4、人脸抓拍

1.1.4. HTTP请求示例

请求 path

```
/v1/api/door/queryB3rInBind
```

请求 body


```
{
  "uuid": "51971496c44d4c638aad1636a2dff64f"
}
```

1.1.5. HTTP响应示例

响应 200

```
{
  "code": 0,
  "data": {
    "list": [
      {
        "deviceModel": 4,
        "deviceType": 4,
        "location": "OpenAPI测试",
        "name": "网络摄像机-MegEye-C3S-123-人脸识别",
        "uuid": "51971496c44d4c638aad1636a2dff64f"
      },
      {
        "deviceModel": 10,
        "deviceType": 4,
        "location": "OpenAPI测试",
        "name": "网络摄像机-MegEye-C4R-122-人脸识别",
        "uuid": "a1233f7f1ae94bb0822d5a8e63b349eb"
      }
    ],
    "pageNum": 1,
    "pageSize": 20,
    "total": 2
  },
  "msg": "成功"
}
```

1.2. 查询门禁点设备列表信息（分页）

1.2.1. 基本信息

Path: /v1/api/door/list

Method: POST

接口描述: 分页查询门禁点设备的信息

1.2.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码
pageSize	integer	非必须	分页大小
total	integer	非必须	总数
list	object []	非必须	
└─uuid	string	非必须	设备uuid
└─location	string	非必须	设备位置
└─name	string	非必须	设备名称
└─deviceType	integer	非必须	设备类型 1、智能摄像头 2、智能面板机 4、网络摄像机
└─deviceModel	integer	非必须	设备型号 1、MegEye-W5K-I8 2、MegEye-W4K-E7 3、MegEye-W4K-I5 4、MegEye-C3S-123 5、MegEye-C4H-142 6、MegEye-C4H-242 7、MegEye-C4N-888 8、MegEye-C4S-222 9、MegEye-C4S-322 10、MegEye-C4R-122 11、MegEye-C4R-222 12、MegEye-C4R-322 13、MegEye-D1S-X2X 14、MegEye-D1S-X4X 15、MegID-W2K 16、MegID-W2K-E 17、MegEye-W3K-45S 21、IPC、34、MegEye-W4K-I52 35、MegEye-W4K-I31
└─deviceMode	integer	非必须	设备模式（服务器无） 1、人脸识别 4、人脸抓拍

1.2.3. 返回数据

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码
pageSize	integer	非必须	分页大小
total	integer	非必须	总数
list	object []	非必须	
└─uuid	string	非必须	设备uuid
└─location	string	非必须	设备位置
└─name	string	非必须	设备名称
└─deviceType	integer	非必须	设备类型 1、智能摄像头 2、智能面板机 4、网络摄像机
└─deviceModel	integer	非必须	设备型号 1、MegEye-W5K-I8、2 MegEye-W4K-E7 3、MegEye-W4K-I5 4、MegEye-C3S-123 5、MegEye-C4H-142 6、MegEye-C4H-242 7、MegEye-C4N-888 8、MegEye-C4S-222 9、MegEye-C4S-322 10、MegEye-C4R-122 11、MegEye-C4R-222 12、MegEye-C4R-322 13、MegEye-D1S-X2X 14、MegEye-D1S-X4X 15、MegID-W2K 16、MegID-W2K-E 17、MegEye-W3K-45S 21、IPC
└─status	integer	非必须	设备状态 3 在线 4 离线
└─synStatus	integer	非必须	同步状态 0 未同步 1 已同步 2 同步失败（以后不在返回）
└─switchIp	string	非必须	绑定开关
└─isPass	integer	非必须	是否绑定门禁设备 0 不是 1是

名称	类型	是否必须	备注
failNum	integer	非必须	同步失败的数量，（以后不在返回）
cmdStatus	integer	非必须	常开常闭状态

1.2.4. HTTP请求示例

请求 path

```
/v1/api/door/list
```

请求 body

```
{
  "passModel": false,
  "location": "OpenAPI测试",
  "pageNum": 1,
  "name": "智能面板机-MegEye-W5K-I8",
  "pageSize": 5,
  "deviceType": 2,
  "deviceModel": 1
}
```

1.2.5. HTTP响应示例

响应 200

```
{
  "code": 0,
  "data": {
    "list": [
      {
        "cmdStatus": 16,
        "deviceModel": 1,
        "deviceType": 2,
        "isPass": 1,
        "location": "OpenAPI测试",
        "name": "智能面板机-MegEye-W5K-I8",
        "status": 4,
        "uuid": "8b2e4da9392b4847aff5faf9dcb17a3c"
      }
    ],
    "pageNum": 1,
    "pageSize": 5,
    "total": 1
  },
  "msg": "成功"
}
```

1.3. 查询同步失败人员信息

1.3.1. 基本信息

Path: /v1/api/door/queryFeatureFailList

Method: POST

接口描述: 查询同步设备失败的人员信息

1.3.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码,不传默认1
uuid	string	非必须	设备uuid（盘古V1.0.2版本无须传递，查询整个区域的数据）
pageSize	integer	非必须	分页大小，不传默认20

1.3.3. 返回数据

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码
pageSize	integer	非必须	分页大小
total	integer	非必须	总数
list	object []	非必须	
├personUuid	string	非必须	人员personUuid
├name	string	非必须	人员名
├imageUri	string	非必须	用户识别照片的uri
├type	integer	非必须	用户类别, 1 员工 2 访客 3 重点人员
├reasonStr	string	非必须	失败原因描述

1.3.4. HTTP请求示例

请求 path

```
/v1/api/door/queryFeatureFailList
```

请求 body

```
{
  "pageNum": 1,
  "uuid": "8b2e4da9392b4847aff5faf9dcb17a3c",
  "pageSize": 5
}
```

1.3.5. HTTP响应示例

响应 200

```
{
  "code": 0,
  "data": {
    "list": [],
    "pageNum": 1,
    "pageSize": 5,
    "total": 0
  },
  "msg": "成功"
}
```

1.4. 查询绑定设备类型

1.4.1. 基本信息

Path: /v1/api/door/queryBindType

Method: POST

接口描述:

1.4.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
uuid	string	必须	端设备uuid

1.4.3. 返回数据

名称	类型	是否必须	备注
bindType	integer	必须	绑定类型，1-B3R 2-CORE 3-面板机

1.4.4. HTTP请求示例

请求 path

```
/v1/api/door/queryBindType
```

请求 body

```
{
  "uuid": "8b2e4da9392b4847aff5faf9dcb17a3c"
}
```

1.4.5. HTTP响应示例

响应 200

```
{'code': 0, 'data': {'bindType': 3}, 'msg': '成功'}
```

1.5. 查询设备配置信息

1.5.1. 基本信息

Path: /v1/api/door/queryConfig

Method: POST

接口描述: 查询已经绑定的设备的一些具体的配置信息

1.5.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

1.5.3. 返回数据

名称	类型	是否必须	
uuid	string	必须	设备uuid
customConfig	object	必须	个性化配置信息
└─companyName	string	非必须	公司名称，由
└─showPersonName	boolean	非必须	人员名称是否
└─pictureType	integer	非必须	人员照片显示 片；2：不显示
└─standByTime	integer	非必须	进入待机时间(
└─fillLightness	integer	非必须	通行补光亮度
└─recognitionInterval	number	非必须	二次识别间隔
└─rebootSchedule	string	非必须	定时重启 定时 期}{小时}:{分钟} 日，取 * 时代 隔，例如：*/2 启，0,1,2,3,4/ 三、周四、周 */标识未开启定
└─alarmInput1	integer	非必须	报警输入1 0： 2：报警信号
└─alarmInput2	integer	非必须	报警输入2 0： 2：报警信号
└─alarmOutput1	integer []	非必须	报警输出 0：联 报警联动报警； 3：报警输入1 动报警
└─		非必须	
└─recognitionTimes	integer	非必须	二次识别次数

名称	类型	是否必须	
└─lockSwitch	boolean	非必须	门锁控制 true
└─sensorType	integer	非必须	门磁类型 0 不
└─verificationMode	integer	非必须	通行认证方式, 及刷卡;3:人脸
└─openTimeOut	integer	非必须	开门超时时长,
└─openDuration	number	非必须	门禁控制时间,
└─openDelay	number	非必须	延迟门锁控制,
└─wiegandMode	integer	非必须	韦根模式 0:关
└─wiegandFormat	integer	非必须	韦根格式 0:26
└─wiegandOutput	integer	非必须	韦根输出内容
└─standbyWpUrl	string	非必须	自定义待机界
algorithmConfig	object	必须	人脸参数信息
└─recognitionMode	integer	非必须	使用模式（面 多人
└─recognitionThreshold	number	非必须	识别阈值 最大
└─livenessModel	integer	必须	通行活体模式 1 开单目 2开双
└─livenessThreshold	number	非必须	活体检测阈值 要传）最大值
└─maxDistance	integer	非必须	最大识别距离 2: 1.5 米; 3:

名称	类型	是否必须	
└pitchThreshold	number	非必须	垂直角度 0-90
└yawThreshold	number	非必须	水平角度 0-90
└rollThreshold	number	非必须	旋转角度 0-90
└sharpnessThreshold	number	非必须	模糊度 0-1，精度
└smallestFace	number	非必须	最小人脸（非
└strangerRecognitionThreshold	number	非必须	陌生人识别阈
└faceAeMode	boolean	非必须	FACE AE，开
└maskSwitch	boolean	非必须	口罩识别模式，
└maskThreshold	number	非必须	口罩识别阈值
safeGuardConfig	object	必须	员工、访客、
└recognitionMode	integer	非必须	使用模式 0：主
└recognitionThreshold	number	非必须	识别阈值 最大
└livenessModel	integer	非必须	通行活体模式
└livenessThreshold	number	非必须	活体检测阈值
└maxDistance	integer	非必须	最大识别距离 米

名称	类型	是否必须	
└pitchThreshold	number	非必须	垂直角度 0-90
└yawThreshold	number	非必须	水平角度 0-90
└rollThreshold	number	非必须	旋转角度 0-90
└sharpnessThreshold	number	非必须	模糊度 0.1-1.0
└smallestFace	number	非必须	最小人脸，范围
└strangerRecognitionThreshold	number	非必须	陌生人识别阈值
└recognitionInterval	number	非必须	二次识别间隔
strangerConfig	object	必须	全局陌生人参数
└recognitionMode	integer	非必须	使用模式 0: 主
└recognitionThreshold	number	非必须	识别阈值 最大
└maxDistance	integer	非必须	最大识别距离 米
└pitchThreshold	number	非必须	垂直角度 0-90
└yawThreshold	number	非必须	水平角度 0-90
└rollThreshold	number	非必须	旋转角度 0-90
└sharpnessThreshold	number	非必须	模糊度 0.1-1.0

名称	类型	是否必须	
└─smallestFace	number	非必须	最小人脸，范围
└─strangerRecognitionThreshold	number	非必须	陌生人识别阈值
└─recognitionInterval	number	非必须	二次识别间隔
temperatureConfig	object	非必须	测温参数
└─temperatureSwitch	boolean	必须	体温检测开关，true开启，false关闭
└─highTemperatureOff	boolean	必须	高温禁入开关，true开启，false关闭，温度高于阈值时
└─temperatureMode	boolean	必须	快速测试开关，true开启，false关闭，只做测温不做
└─highThreshold	number	必须	高温阈值：默认36.5，true开启，false关闭，填写，范围36
└─maskAlertSwitch	boolean	必须	未佩戴口罩提醒开关，true开启，false关闭，开启时对未佩
└─unMaskOff	boolean	必须	口罩佩戴禁入开关，true开启，false关闭，未戴口罩人员
healthkitConfig	object	非必须	设备配置健康码
└─healthKitMode	integer	非必须	是否开启健康码
└─healthKitProvinceList	object []	非必须	健康码省份列表，格式：[{ "cityCode": "110101", "cityName": "北京" }]
└─cityCode	string	非必须	城市code
└─cityName	string	非必须	城市名称
└─healthKitProvince	string	非必须	健康码使用的省份code（与cityCode值）

名称	类型	是否必须	
healthKitCoordinateMode	integer	非必须	坐标上传开启
healthKitLongitude	number	非必须	坐标经度 经度范围：-180.0000
healthKitLatitude	number	非必须	坐标纬度 纬度
healthKitGreen	integer	非必须	健康码绿码 0:
healthKitRed	integer	非必须	健康码红码 0:
healthKitYellow	integer	非必须	健康码黄码 0:
healthKitUnknown	integer	非必须	健康码未知 0:
healthKitTimeout	integer	非必须	健康码超时 0:
healthKitImageUrl	string	非必须	二维码图片uri
healthKitIdcardVerify	integer	非必须	身份证模块开:
healthKitIdcardAutosave	integer	非必须	自动保存身份i
healthKitScheduleVerify	integer	非必须	通行时间权限:
healthKitHighTemperatureNoEntry	integer	非必须	高温禁行开关
healthKitHighTemperatureThreshold	number	非必须	高温阈值 默认
healthKitServerAddress	string	非必须	健康码平台地

名称	类型	是否必须	
healthkitTemperatureSensitivity	integer	非必须	健康码测温灵敏度

1.5.4. HTTP请求示例

请求 **path**

```
/v1/api/door/queryConfig
```

请求 **body**

```
{
  "uuid": "8b2e4da9392b4847aff5faf9dcb17a3c"
}
```

1.5.5. HTTP响应示例

响应 **200**

```

{
  "code": 0,
  "data": {
    "algorithmConfig": {
      "faceAeMode": false,
      "livenessModel": 0,
      "livenessThreshold": 60,
      "maskSwitch": true,
      "maskThreshold": 50,
      "maxDistance": 1,
      "pitchThreshold": 30,
      "recognitionMode": 0,
      "recognitionThreshold": 70,
      "rollThreshold": 30,
      "sharpnessThreshold": 0.8,
      "smallestFace": 20,
      "yawThreshold": 30
    },
    "customConfig": {
      "alarmInput1": 0,
      "alarmInput2": 0,
      "alarmOutput1": [
        0,
        1
      ],
      "companyName": "个性化配置信息公司名称",
      "fillLightness": 30,
      "lockSwitch": true,
      "openDelay": 1,
      "openDuration": 0.1,
      "openTimeOut": 1,
      "pictureType": 0,
      "rebootSchedule": "*/",
      "recognitionInterval": 1,
      "recognitionTimes": 2,
      "sensorType": 0,
      "showPersonName": false,
      "standByTime": 10,
      "standbyWpUrl": "_ZzEwMF8xZA==_f9f0f4ecc2a44762a7394b3cd5c81bc2",
      "verificationMode": 0,
      "wiegandFormat": 0,
      "wiegandMode": 1,
      "wiegandOutput": 0
    },
    "healthkitConfig": {
      "healthKitProvinceList": []
    },
    "temperatureConfig": {
      "highTemperatureOff": true,
      "highThreshold": 36.5,
      "maskAlertSwitch": false,
      "temperatureMode": false,
      "temperatureSwitch": true,
      "unMaskOff": false
    },
    "uuid": "8b2e4da9392b4847aff5faf9dcb17a3c"
  },
  "msg": "成功"
}

```

1.6. 批量添加和移除门禁点

1.6.1. 基本信息

Path: /v1/api/door/add

Method: POST

接口描述: 用于批量添加和移除门禁点

1.6.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
addDeviceUuids	string []	必须	新添设备uuid
└─		非必须	
removeDeviceUuids	string []	必须	删除设备uuid
└─		非必须	

1.6.3. 返回数据

OK

1.6.4. HTTP请求示例

请求 path

/v1/api/door/add

请求 body

<pre>{ "addDeviceUuids": ["eb4c9ed7d25348fba14ab429832d818b", "16fee62a23f84321bafc38bca7749896", "51971496c44d4c638aad1636a2dff64f", "f45c0378f8114b5082f1c39376d46556", "bbc1305df66b4b00a23cf8bb3cce8640", "a1233f7f1ae94bb0822d5a8e63b349eb", "51971496c44d4c638aad1636a2dff64f", "8b2e4da9392b4847aff5faf9dcb17a3c"], "removeDeviceUuids": [] }</pre>
--

1.6.5. HTTP响应示例

响应 200

```
{
  "code": 0,
  "msg": "成功"
}
```

1.7. 移除门禁点

1.7.1. 基本信息

Path: /v1/api/door/remove

Method: POST

接口描述:

1.7.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
pageNum	integer	必须	当前页码
pageSize	integer	必须	分页大小
total	integer	必须	总数
list	object []	非必须	
└─groupUuid	string	必须	组uuid
└─personUuid	string	必须	人员uuid
└─imageUri	string	非必须	用户影像
└─personName	string	必须	人员名
└─personCode	string	非必须	员工编号(员工组)
└─visitTargetUuid	string	非必须	拜访用户uuid
└─visitTarget	string	非必须	拜访用户姓名
└─visitStartTime	string	非必须	拜访起始时间(时间戳)
└─visitEndTime	string	非必须	拜访结束时间(时间戳)
└─status	string	非必须	访客状态：无效、有效
└─phone	string	非必须	手机号码
└─identifyNum	string	非必须	身份证号码
└─type	integer	非必须	人员类型
└─visitReason	string	非必须	拜访原因, 访客可填,格式: 任意字符, 长度: 1-255

1.7.3. 返回数据

OK

1.7.4. HTTP请求示例

请求 path

/v1/api/door/remove

请求 **body**

```
{
  "uuid": "8b2e4da9392b4847aff5faf9dcb17a3c"
}
```

1.7.5. HTTP响应示例

响应 **200**

```
{
  "code": 0,
  "msg": "成功"
}
```

1.8. 绑定开关

1.8.1. 基本信息

Path: /v1/api/door/bindSwitch

Method: POST

接口描述: 绑定开关，不传表示移除门禁开关

1.8.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
switchIp	string	非必须	开关IP,不传则表示移除门禁开关
uuid	string	必须	当前设备uuid

1.8.3. 返回数据

```
OK
```

1.8.4. HTTP请求示例

请求 **path**

```
/v1/api/door/bindSwitch
```

请求 **body**

```
{
  "uuid": "eb4c9ed7d25348fba14ab429832d818b",
  "switchIp": "10.20.30.40"
}
```

1.8.5. HTTP响应示例

响应 **200**

```
{
  "code": 0,
  "msg": "成功"
}
```

1.9. 远程常开

1.9.1. 基本信息

Path: /v1/api/door/alwaysOpenDoor

Method: POST

接口描述: 远程控制设备，将面板机设备远程长开，不支持非面板机设备

1.9.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
uuid	string	必须	当前设备uuid

1.9.3. 返回数据

```
OK
```

1.9.4. HTTP请求示例

请求 **path**

```
/v1/api/door/alwaysOpenDoor
```

请求 body

```
{
  "uuid": "51971496c44d4c638aad1636a2dff64f"
}
```

1.9.5. HTTP响应示例

响应 200

面板机

```
{
  "code": 0,
  "msg": "成功"
}
```

非面板机不支持

```
{
  "msg": "目前仅面板机支持远程长开、远程常闭、恢复正常",
  "code": 200314
}
```

1.10. 远程常闭

1.10.1. 基本信息

Path: /v1/api/door/alwaysCloseDoor

Method: POST

接口描述: 远程控制设备，将面板机设备远程长闭，不支持非面板机设备

1.10.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
uuid	string	必须	当前设备uuid

1.10.3. 返回数据

```
OK
```

1.10.4. HTTP请求示例

请求 path

```
/v1/api/door/alwaysCloseDoor
```

请求 body

```
{
  "uuid": "51971496c44d4c638aad1636a2dff64f"
}
```

1.10.5. HTTP响应示例

响应 200

面板机

```
{
  "code": 0,
  "msg": "成功"
}
```

非面板机不支持

```
{
  "msg": "目前仅面板机支持远程长开、远程常闭、恢复正常",
  "code": 200314
}
```

1.11. 远程开门

1.11.1. 基本信息

Path: /v1/api/door/openDoor

Method: POST

接口描述: 远程控制设备，将设备远程开门

1.11.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
uuid	string	必须	当前设备uuid

1.11.3. 返回数据

OK

1.11.4. HTTP请求示例

请求 **path**

```
/v1/api/door/openDoor
```

请求 **body**

```
{
  "uuid": "51971496c44d4c638aad1636a2dff64f"
}
```

1.11.5. HTTP响应示例

响应 **200**

```
{
  "code": 0,
  "msg": "成功"
}
```

1.12. 恢复正常

1.12.1. 基本信息

Path: /v1/api/door/normal

Method: POST

接口描述: 远程控制设备，将远程长开、远程常闭 的面板机设备恢复正常，不支持非面板机设备

1.12.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
uuid	string	必须	当前设备uuid

1.12.3. 返回数据

OK

1.12.4. HTTP请求示例

请求 **path**

```
/v1/api/door/normal
```

请求 **body**

```
{
  "uuid": "51971496c44d4c638aad1636a2dff64f"
}
```

1.12.5. HTTP响应示例

响应 **200**

面板机

```
{
  "code": 0,
  "msg": "成功"
}
```

非面板机不支持

```
{
  "msg": "目前仅面板机支持远程长开、远程常闭、恢复正常",
  "code": 200314
}
```

1.13. 配置设备

1.13.1. 基本信息

Path: /v1/api/door/config

Method: POST

接口描述:

1.13.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
uuid	string	必须	设备uuid
customConfig	object	非必须	门禁参数，用
└─companyName	string	非必须	公司名称，由32个字母组成
└─showPersonName	boolean	非必须	人员名称是否启 false 关闭
└─pictureType	integer	非必须	人员照片显示片；1：抓拍照显示图片
└─standByTime	integer	非必须	进入待机时间(min 0表示永不
└─fillLightness	integer	非必须	通行补光亮度，100 最小值: 0
└─recognitionInterval	number	非必须	二次识别间隔
└─rebootSchedule	string	非必须	定时重启 定时格式: {星期}/{钟}, 0~6 依次周日, 取 * 时多天之间用逗如: */23:20 代23:20 重启, 0,1,2,3,4/5:5, 一、周二、周三、周四、周五的每天早启, */标识未开启
└─alarmInput1	integer	非必须	报警输入1 0: 1: 消防信号; 号
└─alarmInput2	integer	非必须	报警输入2 0: 1: 消防信号; 号
└─alarmOutput1	integer []	非必须	报警输出 0: 显示; 1: 黑名单报警; 2: 门磁报警; 3: 报警报警; 4: 报警报警

名称	类型	是否必须	备注
└		非必须	
└recognitionTimes	integer	非必须	二次识别次数
└lockSwitch	boolean	非必须	门锁控制 true
└sensorType	integer	非必须	门磁类型 0 不 2 常闭
└verificationMode	integer	非必须	通行认证方式, 脸或刷卡;2:人 人脸及密码
└openTimeOut	integer	非必须	开门超时时长, 围: 0-255单位
└openDuration	number	非必须	门禁控制时间, 最小0.1,最大6
└openDelay	number	非必须	延迟门锁控制, 0-60之间的数, 一位小数, 单位
└wiegandMode	integer	非必须	韦根模式 0:关 入;2:韦根输出
└wiegandFormat	integer	非必须	韦根格式 0:26
└wiegandOutput	integer	非必须	韦根输出内容 号;1:人员编号
└standbyWpUrl	string	非必须	自定义待机界面
algorithmConfig	object	非必须	识别参数,用于
└recognitionMode	integer	非必须	使用模式 0: 单 多人
└recognitionThreshold	number	非必须	识别阈值 最大 小值: 0

名称	类型	是否必须	备注
└─livenessModel	integer	非必须	通行活体模式 目 2开双目
└─livenessThreshold	number	非必须	活体检测阈值 100 最小值: 0
└─maxDistance	integer	非必须	最大识别距离 2: 1.5 米; 3:
└─pitchThreshold	number	非必须	垂直角度 0-90
└─yawThreshold	number	非必须	水平角度 0-90
└─rollThreshold	number	非必须	旋转角度 0-90
└─sharpnessThreshold	number	非必须	模糊度 0.1-1.0
└─smallestFace	number	非必须	最小人脸 1-50
└─strangerRecognitionThreshold	number	非必须	陌生人识别阈值 100 最小值: 0
└─faceAeMode	boolean	非必须	FACE AE, 开 启 false、关闭
└─maskSwitch	boolean	非必须	口罩识别模式: 启, false关闭
└─maskThreshold	number	非必须	口罩识别阈值 100 最小值: 0
safeGuardConfig	object	非必须	员工、访客、 员参数
└─recognitionThreshold	number	非必须	识别阈值 最大 小值: 0
└─livenessModel	integer	非必须	通行活体模式 目 2开双目

名称	类型	是否必须	备注
└─livenessThreshold	number	非必须	活体检测阈值 100 最小值: 0
└─pitchThreshold	number	非必须	垂直角度 0-90
└─yawThreshold	number	非必须	水平角度 0-90
└─rollThreshold	number	非必须	旋转角度 0-90
└─sharpnessThreshold	number	非必须	模糊度 0.1-1.0
└─smallestFace	number	非必须	最小人脸, 摄像时需要传值 1-
└─strangerRecognitionThreshold	number	非必须	陌生人识别阈值 100 最小值: 0
└─recognitionInterval	number	非必须	二次识别间隔
strangerConfig	object	非必须	全局陌生人参数
└─recognitionThreshold	number	非必须	识别阈值 最大 小值: 0
└─pitchThreshold	number	非必须	垂直角度 0-90
└─yawThreshold	number	非必须	水平角度 0-90
└─rollThreshold	number	非必须	旋转角度 0-90
└─sharpnessThreshold	number	非必须	模糊度 0.1-1.0
└─smallestFace	number	非必须	最小人脸, 摄像时需要传值

名称	类型	是否必须	备注
└strangerRecognitionThreshold	number	非必须	陌生人识别阈值 100 最小值: 0
└recognitionInterval	number	非必须	二次识别间隔
temperatureConfig	object	非必须	测温参数
└temperatureSwitch	boolean	必须	体温检测开关, 闭, 开启时启
└highTemperatureOff	boolean	必须	高温禁入开关, 闭, 开启时当, 于阈值时不开
└temperatureMode	boolean	必须	快速测试开关, false, 开启时做识别
└highThreshold	number	必须	高温阈值: 默认数值用户可手, 围36.5-38.5°
└maskAlertSwitch	boolean	必须	未佩戴口罩提醒, 默认关闭 false, 未佩戴口罩进
└unMaskOff	boolean	必须	口罩佩戴禁入, 关闭: false 关闭, 默认关闭, 测到未戴口罩, 门
healthkitConfig	object	非必须	健康码配置信
└healthKitMode	integer	非必须	是否开启健康码, 闭 1 开启
└healthKitProvince	string	非必须	健康码使用的, 递省份列表中(值)
└healthKitCoordinateMode	integer	非必须	坐标上传开启, 开启
└healthKitLongitude	number	非必须	坐标经度 经度, 围: -180.0000
└healthKitLatitude	number	非必须	坐标纬度 纬度, 围: -90.0000-

名称	类型	是否必须	备注
healthKitGreen	integer	非必须	健康码绿码 0: 关闭
healthKitRed	integer	非必须	健康码红码 0: 关闭
healthKitYellow	integer	非必须	健康码黄码 0: 关闭
healthKitUnknown	integer	非必须	健康码未知 0: 关闭
healthKitTimeout	integer	非必须	健康码超时 0: 关闭
healthKitImageUrl	string	非必须	二维码图片uri
healthKitIdcardVerify	integer	非必须	身份证模块开启 1: 开启
healthKitIdcardAutosave	integer	非必须	自动保存身份信息 0:关闭 1: 开启
healthKitScheduleVerify	integer	非必须	通行时间权限 1: 开启
healthKitHighTemperatureNoEntry	integer	非必须	高温禁行开关 开启
healthKitHighTemperatureThreshold	number	非必须	高温阈值 默认
healthKitServerAddress	string	非必须	健康码平台地址
healthkitTemperatureSensitivity	integer	非必须	健康码测温灵敏度 低; 1: 中; 2: 高

1.13.3. 返回数据

OK

1.13.4. HTTP请求示例

请求 **path**

```
/v1/api/door/openDoor
```

请求 **body**

```
{  
  "uuid": "51971496c44d4c638aad1636a2dff64f"  
}
```

1.13.5. HTTP响应示例

响应 **200**


```

{
  "code": 0,
  "data": {
    "algorithmConfig": {
      "livenessModel": 0,
      "livenessThreshold": 60,
      "pitchThreshold": 30,
      "recognitionMode": 0,
      "recognitionThreshold": 70,
      "rollThreshold": 30,
      "sharpnessThreshold": 0.8,
      "smallestFace": 20,
      "strangerRecognitionThreshold": 10,
      "yawThreshold": 30
    },
    "customConfig": {
      "recognitionInterval": 1,
      "recognitionTimes": 2
    },
    "healthkitConfig": {
      "healthKitProvinceList": []
    },
    "safeGuardConfig": {
      "livenessModel": 0,
      "livenessThreshold": 90,
      "pitchThreshold": 90,
      "recognitionInterval": 1,
      "recognitionThreshold": 60,
      "rollThreshold": 90,
      "sharpnessThreshold": 1,
      "smallestFace": 45,
      "strangerRecognitionThreshold": 58,
      "yawThreshold": 90
    },
    "strangerConfig": {
      "pitchThreshold": 90,
      "recognitionInterval": 2,
      "recognitionThreshold": 70,
      "rollThreshold": 90,
      "sharpnessThreshold": 1,
      "smallestFace": 45,
      "strangerRecognitionThreshold": 60,
      "yawThreshold": 90
    },
    "uuid": "16fee62a23f84321bafc38bca7749896"
  },
  "msg": "成功"
}

```

1.14. 查询同步列表（废弃）

1.14.1. 基本信息

Path: /v1/api/door/querySynList

Method: POST

接口描述： 获取绑定在同一个算力设备下的其他设备

1.14.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码,不传默认1
uuid	string	必须	当前设备uuid
location	string	必须	设备位置
pageSize	integer	非必须	分页大小, 不传默认20
name	string	必须	设备名称
deviceModel	integer	非必须	设备型号 1、MegEye-W5K-I8 2、MegEye-W4K-E7 3、MegEye-W4K-I5 4、MegEye-C3S-123 5、MegEye-C4H-142 6、MegEye-C4H-242 7、MegEye-C4N-888 8、MegEye-C4S-222 9、MegEye-C4S-322 10、MegEye-C4R-122 11、MegEye-C4R-222 12、MegEye-C4R-322 13、MegEye-D1S-X2X 14、MegEye-D1S-X4X 15、MegID-W2K 16、MegID-W2K-E 17、MegEye-W3K-45S 21、IPC

1.14.3. 返回数据

名称	类型	是否必须	备注
pageNum	integer	必须	当前页码
pageSize	integer	必须	分页大小
total	integer	必须	总数
list	object []	非必须	
└─uuid	string	必须	设备uuid
└─location	string	必须	设备位置
└─name	string	必须	设备名称
└─deviceType	integer	必须	设备类型 1、智能摄像头 2、智能面板机 4、网络摄像机
└─deviceModel	integer	必须	设备型号 1、MegEye-W5K-I8 2、MegEye-W4K-E7 3、MegEye-W4K-I5 4、MegEye-C3S-123 5、MegEye-C4H-142 6、MegEye-C4H-242 7、MegEye-C4N-888 8、MegEye-C4S-222 9、MegEye-C4S-322 10、MegEye-C4R-122 11、MegEye-C4R-222 12、MegEye-C4R-322 13、MegEye-D1S-X2X 14、MegEye-D1S-X4X 15、MegID-W2K 16、MegID-W2K-E 17、MegEye-W3K-45S 21、IPC

1.14.4. HTTP请求示例

请求 path

```
/v1/api/door/querySynList
```

请求 body

```
{
  "location": "",
  "pageNum": 1,
  "uuid": "8b2e4da9392b4847aff5faf9dcb17a3c",
  "pageSize": 5,
  "name": "",
  "deviceModel": 1
}
```

1.14.5. HTTP响应示例

响应 200

```
{
  "code": 0,
  "data": {
    "list": [],
    "pageNum": 1,
    "pageSize": 5,
    "total": 0
  },
  "msg": "成功"
}
```

1.15. 同步失败详情（废弃）

1.15.1. 基本信息

Path: /v1/api/door/syncDetail

Method: POST

接口描述: 查询同步设备的详细信息

1.15.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码,不传默认1
uuid	string	非必须	设备uuid（盘古V1.0.2版本无须传递，查询整个区域的数据）
pageSize	integer	非必须	分页大小，不传默认20

1.15.3. 返回数据

名称	类型	是否必须	备注
pageNum	integer	必须	当前页码
pageSize	integer	必须	分页大小
total	integer	必须	总数
list	object []	非必须	
└─uuid	string	必须	personUuid
└─name	string	必须	人员名
└─imageUrl	string	非必须	用户识别照片的url
└─type	integer	必须	用户类别, 1 员工 2 访客 3 重点人员
└─reasonStr	string	非必须	失败原因描述

1.15.4. HTTP请求示例

请求 path

```
/v1/api/door/syncDetail
```

请求 body

```
{
  "pageNum": 1,
  "uuid": "8b2e4da9392b4847aff5faf9dcb17a3c",
  "pageSize": 5,
  "personName": "",
  "type": 1
}
```

1.15.5. HTTP响应示例

响应 200

```
{
  "code": 0,
  "data": {
    "list": [],
    "pageNum": 1,
    "pageSize": 5,
    "total": 0
  },
  "msg": "成功"
}
```

- 1. 门禁设备告警
 - 1.1. 门禁设备告警列表查询
 - 1.1.1. 基本信息
 - 1.1.2. 请求参数
 - 1.1.3. 返回数据
 - 1.1.4. HTTP请求示例
 - 1.1.5. HTTP响应示例

1. 门禁设备告警

1.1. 门禁设备告警列表查询

1.1.1. 基本信息

Path: /v1/api/device_alarm/list

Method: POST

接口描述: 分页查询门禁设备告警列表，支持条件搜索；

1.1.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码,不传默认1
pageSize	integer	非必须	分页大小, 不传默认20
deviceUuids	string []	非必须	设备uuid列表
└─		非必须	
deviceName	string	非必须	设备名称, 支持模糊搜索
alarmStatus	integer	非必须	历史报警状态。0 - 报警恢复; 1 - 报警
alarmType	integer	非必须	报警类型。500000 - 存储空间占用超过 90% 报警; 500100 - 防拆报警; 500102 - 密码破解报警; 500104 - 门禁开放超时报警; 500200 - 报警输入报警; 500500 - 消防报警; 500108 - 测温模块失效
startAlarmTime	string	非必须	报警开始时间(包含)。格式为"2020-01-01 10:00:00", 或者填一个long类型的unix timestamp(毫秒)也行
endAlarmTime	string	非必须	报警结束时间(不包含)。格式为"2020-01-01 10:00:00", 或者填一个long类型的unix timestamp(毫秒)也行

1.1.3. 返回数据

名称	类型	是否必须	备注
pageNum	integer	必须	当前页码
pageSize	integer	必须	分页大小
total	integer	必须	总数
list	object []	非必须	
 deviceUuid	string	非必须	设备uuid
 deviceName	string	非必须	设备名称
 deviceLocation	string	非必须	设备位置
 deviceType	integer	非必须	设备的类型。1 - 智能摄像头; 2 - 智能面板机; 4 - 网络摄像机;
 deviceModel	integer	非必须	设备的型号。设备型号 1 MegEye-W5K-I8 2 MegEye-W4K-E7 3 MegEye-W4K-I5 4 MegEye-C3S-123 5 MegEye-C4H-142 6 MegEye-C4H-242 7 MegEye-C4N-888 8 MegEye-C4S-222 9 MegEye-C4S-322 10 MegEye-C4R-122 11 MegEye-C4R-222 12 MegEye-C4R-322 13 MegEye-D1S-X2X 14 MegEye-D1S-X4X 15 MegID-W2K 16 MegID-W2K-E 17 MegEye-W3K-45S 18 Core 19 B3R 20 NVR 21 IPC
 deviceStatus	integer	非必须	设备在发送报警信息时的状态。0 - 未激活; 1 - 在线未绑定; 2 - 离线未绑定; 3 - 在线; 4 - 离线; 5 - 解绑中
 alarmStatus	integer	非必须	历史报警状态。0 - 报警恢复; 1 - 报警
 alarmType	integer	非必须	报警类型。500000 - 存储空间占用超过 90% 报警; 500100 - 防拆报警; 500102 - 密码破解报警; 500104 - 门禁开放超时报警; 500200 - 报警输入报警; 500500 - 消防报警500108 - 测温模块失效
 alarmTime	string	非必须	报警时间。unix时间戳, 毫秒

名称	类型	是否必须	备注
 gmtCreate	string	非必须	记录创建时间。 unix 时间戳，毫秒
 gmtModified	string	非必须	记录修改时间。 unix 时间戳，毫秒
 alarmStatusNow	integer	非必须	当前报警状态。 0 - 报警恢复; 1 - 报警中

1.1.4. HTTP请求示例

请求 path

```
/v1/api/device_alarm/list
```

请求 body

```
{
  "pageSize": 10,
  "pageNum": 1,
  "deviceUuids": [
    "c089dea7471643ef98c5dbe0981de616"
  ],
  "alarmType": 500100,
  "startAlarmTime": 1621785600000,
  "endAlarmTime": 1624377599000,
  "zoneLabel": "一级区域"
}
```

1.1.5. HTTP响应示例

响应 200

```

{
  "code": 0,
  "data": {
    "list": [
      {
        "alarmStatus": 1,
        "alarmStatusNow": 0,
        "alarmTime": 1622449062000,
        "alarmType": 500108,
        "deviceLocation": "OpenAPI测试",
        "deviceModel": 1,
        "deviceName": "智能面板机-MegEye-WSK-I8",
        "deviceStatus": 3,
        "deviceType": 2,
        "deviceUuid": "8b2e4da9392b4847aff5faf9dcb17a3c",
        "gmtCreate": 1622449070000,
        "gmtModified": 1622449070000
      },
      {
        "alarmStatus": 0,
        "alarmStatusNow": 0,
        "alarmTime": 1622449062000,
        "alarmType": 500108,
        "deviceLocation": "OpenAPI测试",
        "deviceModel": 1,
        "deviceName": "智能面板机-MegEye-WSK-I8",
        "deviceStatus": 3,
        "deviceType": 2,
        "deviceUuid": "8b2e4da9392b4847aff5faf9dcb17a3c",
        "gmtCreate": 1622449071000,
        "gmtModified": 1622449071000
      }
    ],
    "pageNum": 1,
    "pageSize": 20,
    "total": 5
  },
  "msg": "成功"
}

```

- 1. 通行权限管理
 - 1.1. 添加通行权限
 - 1.1.1. 基本信息
 - 1.1.2. 请求参数
 - 1.1.3. HTTP请求示例
 - 1.1.4. HTTP响应示例
 - 1.2. 删除通行权限
 - 1.2.1. 基本信息
 - 1.2.2. 请求参数
 - 1.2.3. HTTP请求示例
 - 1.2.4. HTTP响应示例
 - 1.3. 编辑通行权限
 - 1.3.1. 基本信息
 - 1.3.2. 请求参数
 - 1.3.3. HTTP请求示例
 - 1.3.4. HTTP响应示例
 - 1.4. 通行权限列表
 - 1.4.1. 基本信息
 - 1.4.2. 请求参数
 - 1.4.3. 返回数据
 - 1.4.4. HTTP请求示例
 - 1.4.5. HTTP响应示例

1. 通行权限管理

1.1. 添加通行权限

1.1.1. 基本信息

Path: /v1/api/pass/authority/add

Method: POST

接口描述: 添加通行权限

1.1.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
deviceUuids	string []	必须	设备uuid列表
groupUuids	string []	必须	组uuid列表
scheduleUuid	string	必须	时间计划uuid

1.1.3. HTTP请求示例

请求 path

```
/v1/api/pass/authority/add
```

请求 body

```
{
  "deviceUuids": [
    "eb4c9ed7d25348fba14ab429832d818b",
    "16fee62a23f84321bafc38bca7749896",
    "588d56169e3d4587b8c0010316c30cab",
    "f45c0378f8114b5082f1c39376d46556",
    "bbc1305df66b4b00a23cf8bb3cce8640"
  ],
  "groupUuids": [
    "976055aae31445c5a583ebc1a53c7514",
    "90a41fddf76a4daa9dae47762ecfd359",
    "127d6dadbf1c4e3f93be34865ad0d135"
  ],
  "scheduleUuid": "5ce434bd925b4d56a8dded0542d02d81"
}
```

1.1.4. HTTP响应示例

响应 200

```
{
  "code": 0,
  "msg": "成功"
}
```

1.2. 删除通行权限

1.2.1. 基本信息

Path: /v1/api/pass/authority/delete

Method: POST

接口描述: 删除通行权限

1.2.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
deviceUuid	string	必须	设备UUID
scheduleUuid	string	必须	时间计划UUID

1.2.3. HTTP请求示例

请求 path

```
/v1/api/pass/authority/delete
```

请求 body

```
{
  "deviceUuid": "bbc1305df66b4b00a23cf8bb3cce8640",
  "scheduleUuid": "5ce434bd925b4d56a8dded0542d02d81"
}
```

1.2.4. HTTP响应示例

响应 200

```
{
  "code": 0,
  "msg": "成功"
}
```

1.3. 编辑通行权限

1.3.1. 基本信息

Path: /v1/api/pass/authority/update

Method: POST

接口描述: 修改通行权限

1.3.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
deviceId	string	必须	设备UUID
oldScheduleUuid	string	必须	操作前的时间计划UUID
newScheduleUuid	string	必须	新修改的时间计划UUID
oldGroupUuids	string []	必须	操作前的组ID数组
newGroupUuids	string []	必须	新修改的组ID数组

1.3.3. HTTP请求示例

请求 path

```
/v1/api/pass/authority/update
```

请求 body

```
{
  "deviceId": "bbc1305df66b4b00a23cf8bb3cce8640",
  "oldScheduleUuid": "be19a2939eed4171887a7721e9ff40d9",
  "newScheduleUuid": "5ce434bd925b4d56a8dded0542d02d81",
  "oldGroupUuids": [
    "1c54f72571ca4aaba198c1b54d70720e",
    "5774ea9a80a941249c6784a2329fb706",
    "434a4e3b149e4749b3ed5aea7d8a07f9"
  ],
  "newGroupUuids": [
    "976055aae31445c5a583ebc1a53c7514",
    "90a41fddf76a4daa9dae47762ecfd359",
    "127d6dadbf1c4e3f93be34865ad0d135"
  ]
}
```

1.3.4. HTTP响应示例

响应 200

```
{
  "code": 0,
  "msg": "成功"
}
```

1.4. 通行权限列表

1.4.1. 基本信息

Path: /v1/api/pass/authority/list

Method: POST

接口描述: 根据条件分页查询通行权限

1.4.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
pageNum	integer	非必须	分页页码, 默认为 1
pageSize	integer	非必须	分页大小, 默认为 20
deviceName	string	非必须	设备名称, 支持模糊查询
deviceUuids	string []	非必须	设备id列表
groupUuid	string	非必须	组uuid
scheduleUuid	string	非必须	时间计划uuid

1.4.3. 返回数据

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码
pageSize	integer	非必须	分页大小
total	integer	非必须	总数
list	object []	非必须	
├ deviceUuid	string	非必须	设备uuid
├ deviceName	string	非必须	设备名称
├ groups	object []	非必须	组信息
├ groupUuid	string	非必须	组uuid
├ groupName	string	非必须	组名
├ scheduleUuid	string	非必须	时间计划Uuid
├ scheduleName	string	非必须	时间计划名

1.4.4. HTTP请求示例

请求 path

```
/v1/api/pass/authority/list
```

请求 body

```
{
  "pageNum": 1,
  "pageSize": 10,
  "deviceUids": [],
  "groupUuid": "employ_groupUuid"
}
```

1.4.5. HTTP响应示例

响应 200


```

{
  "code": 0,
  "data": {
    "list": [
      {
        "deviceName": "智能摄像头-MegEye-C4H-242-人脸抓拍",
        "deviceUuid": "bbc1305df66b4b00a23cf8bb3cce8640",
        "groups": [
          {
            "groupId": 1,
            "groupName": "员工默认组",
            "groupUuid": "976055aae31445c5a583ebc1a53c7514"
          },
          {
            "groupId": 2,
            "groupName": "访客默认组",
            "groupUuid": "90a41fddf76a4daa9dae47762ecfd359"
          },
          {
            "groupId": 3,
            "groupName": "重点人员默认组",
            "groupUuid": "127d6dadbf1c4e3f93be34865ad0d135"
          }
        ],
        "scheduleName": "默认时间计划",
        "scheduleUuid": "5ce434bd925b4d56a8dded0542d02d81"
      },
      {
        "deviceName": "智能摄像头-MegEye-C3S-123-人脸抓拍",
        "deviceUuid": "f45c0378f8114b5082f1c39376d46556",
        "groups": [
          {
            "groupId": 2,
            "groupName": "访客默认组",
            "groupUuid": "90a41fddf76a4daa9dae47762ecfd359"
          },
          {
            "groupId": 3,
            "groupName": "重点人员默认组",
            "groupUuid": "127d6dadbf1c4e3f93be34865ad0d135"
          },
          {
            "groupId": 1,
            "groupName": "员工默认组",
            "groupUuid": "976055aae31445c5a583ebc1a53c7514"
          }
        ],
        "scheduleName": "默认时间计划",
        "scheduleUuid": "5ce434bd925b4d56a8dded0542d02d81"
      },
      {
        "deviceName": "网络摄像机-MegEye-C4R-322-人脸识别",
        "deviceUuid": "16fee62a23f84321bafc38bca7749896",
        "groups": [
          {
            "groupId": 3,
            "groupName": "重点人员默认组",
            "groupUuid": "127d6dadbf1c4e3f93be34865ad0d135"
          },
          {
            "groupId": 1,
            "groupName": "员工默认组",
            "groupUuid": "976055aae31445c5a583ebc1a53c7514"
          }
        ],
      }
    ]
  }
}

```

```

        "groupId": 2,
        "groupName": "访客默认组",
        "groupUuid": "90a41fddf76a4daa9dae47762ecfd359"
    }
],
    "scheduleName": "默认时间计划",
    "scheduleUuid": "5ce434bd925b4d56a8dded0542d02d81"
},
{
    "deviceName": "网络摄像机-IPC-人脸识别",
    "deviceUuid": "eb4c9ed7d25348fba14ab429832d818b",
    "groups": [
        {
            "groupId": 2,
            "groupName": "访客默认组",
            "groupUuid": "90a41fddf76a4daa9dae47762ecfd359"
        },
        {
            "groupId": 3,
            "groupName": "重点人员默认组",
            "groupUuid": "127d6dadbf1c4e3f93be34865ad0d135"
        },
        {
            "groupId": 1,
            "groupName": "员工默认组",
            "groupUuid": "976055aae31445c5a583ebc1a53c7514"
        }
    ],
    "scheduleName": "默认时间计划",
    "scheduleUuid": "5ce434bd925b4d56a8dded0542d02d81"
},
{
    "deviceName": "网络摄像机-MegEye-C4R-222-人脸识别",
    "deviceUuid": "588d56169e3d4587b8c0010316c30cab",
    "groups": [
        {
            "groupId": 2,
            "groupName": "访客默认组",
            "groupUuid": "90a41fddf76a4daa9dae47762ecfd359"
        },
        {
            "groupId": 3,
            "groupName": "重点人员默认组",
            "groupUuid": "127d6dadbf1c4e3f93be34865ad0d135"
        },
        {
            "groupId": 1,
            "groupName": "员工默认组",
            "groupUuid": "976055aae31445c5a583ebc1a53c7514"
        }
    ],
    "scheduleName": "默认时间计划",
    "scheduleUuid": "5ce434bd925b4d56a8dded0542d02d81"
}
],
    "pageNum": 1,
    "pageSize": 10,
    "total": 5
},
    "msg": "成功"
}

```

- 1. 通行记录
 - 1.1. 分页查询通行记录
 - 1.1.1. 基本信息
 - 1.1.2. 请求参数
 - 1.1.3. 返回数据
 - 1.1.4. HTTP请求示例
 - 1.1.5. HTTP响应示例

1. 通行记录

1.1. 分页查询通行记录

1.1.1. 基本信息

Path: /v1/api/pass_record/list

Method: POST

接口描述: 分页查询通行记录，支持条件筛选；

1.1.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码,不传默认1
pageSize	integer	非必须	分页大小, 不传默认20
personUuid	string	非必须	人员uuid
personName	string	非必须	人员姓名, 支持模糊查询
personCode	string	非必须	员工编码
visitPersonName	string	非必须	受访人员姓名, 支持模糊查询
deviceUuids	string []	非必须	设备uuid列表
└		非必须	
deviceName	string	非必须	设备名称, 支持模糊查询
deviceLocation	string	非必须	设备位置, 支持模糊查询
recognitionType	integer	非必须	识别人员类型。1 - 员工; 2 - 普通访客; 3 - 重点人员 (黑名单); 4 - 通道陌生人; 5 - 未识别; 6 - VIP访客
verificationMode	integer	非必须	验证类型。0 - 人脸; 1 - 人脸或刷卡; 2 - 人脸及刷卡; 3 - 人脸及密码; 4 - 远程开门; 5 - 二维码
passType	integer	非必须	通行类型。0 - 未通行; 1 - 通行; 2 - 复合认证未通过; 3 - 未通行 (非活体攻击); 4 - 未通行 (高温); 5 - 未通行 (未佩戴口罩) 6 - 未通行 (健康码检测未通过)
livenessType	integer	非必须	活体结果。0 - 非活体; 1 - 活体; 2 - 未知

名称	类型	是否必须	备注
startPassTime	string	非必须	时间段查询，开始时间（包含），格式为"2020-01-01 10:00:00"，或者填一个long类型的unix timestamp(毫秒)也行。
endPassTime	string	非必须	时间段查询，结束时间（不包含），格式为"2020-01-01 10:00:00"，或者填一个long类型的unix timestamp(毫秒)也行
maskType	integer	非必须	口罩状态，0：未检测，1：已佩戴，2：未佩戴
healthyState	integer	非必须	健康码状态 0绿码、1黄码、2红码、3未知、4超时

1.1.3. 返回数据

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码
pageSize	integer	非必须	分页大小
total	integer	非必须	总数
list	object []	非必须	
└─ personUuid	string	非必须	人员uuid
└─ personName	string	非必须	人员姓名
└─ personType	integer	非必须	人员类型。1 - 员工; 2 - 访客; 3 - 重点人员
└─ personCode	string	非必须	员工编码
└─ visitType	integer	非必须	访客类型。1 - 普通访客, 2 - VIP访客
└─ visitPeronUuid	string	非必须	访问人员的uuid
└─ visitPersonType	integer	非必须	访问人员的类型。1 - 员工; 2 - 访客; 3 - 重点人员
└─ visitPersonCode	string	非必须	访问人员的员工编码
└─ visitPersonName	string	非必须	访问人员的姓名
└─ visitReason	string	非必须	来访原因
└─ visitFirm	string	非必须	访客单位

名称	类型	是否必须	备注
└─ visitNumPlate	string	非必须	访客车牌
└─ deviceSetUuid	string	非必须	device set的uuid
└─ deviceUuid	string	非必须	抓拍设备的uuid
└─ deviceName	string	非必须	抓拍设备的名称
└─ deviceLocation	string	非必须	抓拍设备的位置
└─ deviceType	integer	非必须	抓拍设备的类型。1 - 智能摄像头; 2 - 智能面板机; 3 - 智能服务器; 4 - 网络摄像机; 5 - 存储服务器
└─ deviceModel	integer	非必须	抓拍设备的型号。设备型号 1 MegEye-W5K-I8 2 MegEye-W4K-E7 3 MegEye-W4K-I5 4 MegEye-C3S-123 5 MegEye-C4H-142 6 MegEye-C4H-242 7 MegEye-C4N-888 8 MegEye-C4S-222 9 MegEye-C4S-322 10 MegEye-C4R-122 11 MegEye-C4R-222 12 MegEye-C4R-322 13 MegEye-D1S-X2X 14 MegEye-D1S-X4X 15 MegID-W2K 16 MegID-W2K-E 17 MegEye-W3K-45S 18 Core 19 B3R 20 NVR 21 IPC
└─ deviceStatus	integer	非必须	抓拍设备在抓拍时的状态。0 - 未激活; 1 - 在线未绑定; 2 - 离线未绑定; 3 - 在线; 4 - 离线; 5 - 解绑中
└─ recognitionType	integer	非必须	识别人员类型。1 - 员工; 2 - 访客; 3 - 重点人员 (黑名单); 4 - 通道陌生人; 5 - 未识别; 6 - VIP访客
└─ verificationMode	integer	非必须	验证类型。0 - 人脸; 1 - 人脸或刷卡; 2 - 人脸及刷卡; 3 - 人脸及密码; 4 - 远程开门; 5 - 二维码

名称	类型	是否必须	备注
└─ passType	integer	非必须	通行类型。0 - 未通行; 1 - 通行; 2 - 复合认证未通过; 3 - 未通行（非活体攻击）; 4 - 未通行（高温）; 5 - 未通行（未佩戴口罩） 6 - 未通行（健康码检测未通过）
└─ livenessType	integer	非必须	活体结果。0 - 非活体; 1 - 活体; 2 - 未知
└─ recognitionScore	number	非必须	识别分数
└─ livenessScore	number	非必须	活体分数
└─ passTime	string	非必须	抓拍时间。ISO 8601的精确到秒的格式化时间。形如 2020-04-15T17:16:20+08:00
└─ uuid	string	非必须	本条识别记录的uuid
└─ personImageUrl	string	非必须	人员底库图片url
└─ snapshotUrl	string	非必须	抓拍图url
└─ passTimestamp	integer	非必须	抓拍时间。unix时间戳，毫秒
└─ recognitionTypeDesc	string	非必须	识别人员类型的描述
└─ verificationModeDesc	string	非必须	验证类型的描述
└─ passTypeDesc	string	非必须	通行类型的描述
└─ temperature	number	非必须	人体温度（未检测的情况下，默认是0.0）
└─ temperatureType	integer	非必须	人体高温状态，0：未检测，1：正常，2：高温

名称	类型	是否必须	备注
 temperatureTypeDesc	string	非必须	人体高温状态的描述
 maskType	integer	非必须	口罩状态，0：未检测，1：已佩戴，2：未佩戴
 maskTypeDesc	string	非必须	口罩状态的描述
 healthyState	integer	非必须	健康码状态 健康码状态 0绿码、1黄码、2红码、3未知、4超时
 healthkitPasstype	integer	非必须	0、未通过 1、通过 2、未使用

1.1.4. HTTP请求示例

请求 path

```
/v1/api/pass_record/list
```

请求 body

```
{
  "pageSize": 10,
  "pageNum": 1,
  "personName": "zhangsna",
  "recognitionType": 1,
  "personCode": "11113333",
  "orgParentFullName": "",
  "passType": 1,
  "verificationMode": 0,
  "deviceUids": [
    "c089dea7471643ef98c5dbe0981de616"
  ],
  "startPassTime": 1621785600000,
  "endPassTime": 1624377599000
}
```

1.1.5. HTTP响应示例

响应 200

```

{
  "code": 0,
  "data": {
    "list": [
      {
        "deviceLocation": "OpenAPI测试",
        "deviceModel": 21,
        "deviceName": "网络摄像机-IPC-人脸识别",
        "deviceSetUuid": "f73f9e33029e48b69709c790a819d58e",
        "deviceStatus": 3,
        "deviceType": 4,
        "deviceUuid": "eb4c9ed7d25348fba14ab429832d818b",
        "livenessScore": 94.824066,
        "livenessType": 1,
        "maskType": 0,
        "maskTypeDesc": "未检测",
        "passTime": "2021-05-31T16:23:23.383+08:00",
        "passTimestamp": 1622449403383,
        "passType": 1,
        "passTypeDesc": "通行",
        "personCode": "266712",
        "personImageUri": "_ZzEwMF9mb3JldmVyQnVja2V0_8413f112861d4376947c93db2",
        "personImageUrl": "http://127.0.0.1/pub/_ZzEwMF9mb3JldmVyQnVja2V0_8413",
        "personName": "备用员工1",
        "personType": 1,
        "personUuid": "5cc0b3e93f6842bead21096cd6b0ca38",
        "recognitionScore": 80.67127,
        "recognitionType": 1,
        "recognitionTypeDesc": "员工",
        "snapshotUri": "_ZzEwMF82bQ==_7a21e1362e3545039352ed4cfb0879a6",
        "snapshotUrl": "http://127.0.0.1/pub/_ZzEwMF82bQ==_7a21e1362e354503935",
        "temperature": 0.0,
        "temperatureType": 0,
        "temperatureTypeDesc": "未检测",
        "uuid": "95413f44d26942b7b34dc4ce1abfa460",
        "verificationMode": 0,
        "verificationModeDesc": "人脸"
      }
    ],
    "pageNum": 1,
    "pageSize": 20,
    "total": 4
  },
  "msg": "成功"
}

```

- 1. 考勤设备管理
 - 1.1. 批量删除考勤设备
 - 1.1.1. 基本信息
 - 1.1.2. 请求参数
 - 1.1.3. 返回数据
 - 1.1.4. HTTP请求示例
 - 1.1.5. HTTP响应示例
 - 1.2. 批量添加考勤设备
 - 1.2.1. 基本信息
 - 1.2.2. 请求参数
 - 1.2.3. 返回数据
 - 1.2.4. HTTP请求示例
 - 1.2.5. HTTP响应示例
 - 1.3. 考勤设备列表查询
 - 1.3.1. 基本信息
 - 1.3.2. 请求参数
 - 1.3.3. 返回数据
 - 1.3.4. HTTP请求示例
 - 1.3.5. HTTP响应示例

1. 考勤设备管理

1.1. 批量删除考勤设备

1.1.1. 基本信息

Path: /v1/api/attendance/device/batchDelete

Method: POST

接口描述: 用于考勤设备的添加、删除和变更，考勤设备为门禁设备的子集，可用于考勤的设备为端设备列如面板机、摄像头以及网络摄像头等（详细请查看数据字典）；未布控的设备可以添加到考勤设备中，需要布控后才能用于考勤。

1.1.2. 请求参数

Headers

参数名称	参数值	是否必须	备注
Content-Type	application/json	是	

Body

名称	类型	是否必须	备注
uuids	string []	必须	考勤设备uuids

1.1.3. 返回数据

名称	类型	是否必须	备注
successes	object []	非必须	成功列表
└─ uuid	string	非必须	考勤设备uuid
└─ code	integer	非必须	
└─ msg	string	非必须	
failures	object []	非必须	失败列表
└─ uuid	string	非必须	考勤设备uuid
└─ code	integer	非必须	
└─ msg	string	非必须	

1.1.4. HTTP请求示例

请求 path

```
/v1/api/attendance/device/batchDelete
```

请求 body

```
{
  'uuids': ['e7d7e9d8f0984cddb099b002997aa59b', 'e0f99441af3a49979e705830e1b4d609']
}
```

1.1.5. HTTP响应示例

响应 200

```
{
  'code': 0,
  'data': {
    'failures': ['e0f99441af3a49979e705830e1b4d609'],
    'successes': [{
      'code': 0,
      'msg': '成功',
      'uuid': 'e7d7e9d8f0984cddb099b002997aa59b'
    }]
  },
  'msg': '成功'
}
```

1.2. 批量添加考勤设备

1.2.1. 基本信息

Path: /v1/api/attendance/device/batchAdd

Method: POST

接口描述: 从门禁点设备中选取一个/多个门禁点设备添加为考勤设备。

1.2.2. 请求参数

Headers

参数名称	参数值	是否必须	备注
Content-Type	application/json	是	

Body

名称	类型	是否必须	备注
uuids	string []	必须	门禁点设备uuids

1.2.3. 返回数据

名称	类型	是否必须	备注
successes	object []	非必须	添加成功列表
└─ uuid	string	非必须	门禁点设备uuid
└─ attendanceDeviceUuid	string	非必须	考勤设备uuid, 添加成功时必有
└─ code	integer	非必须	
└─ msg	string	非必须	
failures	object []	非必须	添加失败列表
└─ uuid	string	非必须	门禁点设备uuid
└─ attendanceDeviceUuid	string	非必须	考勤设备uuid, 添加成功时必有
└─ code	integer	非必须	
└─ msg	string	非必须	

1.2.4. HTTP请求示例

请求 path

```
/v1/api/attendance/device/batchAdd
```

请求 **body**

```
{
  'uuids': ['b1dafa998fde488ea63d5965a14a15d9', 'v1dafb9944de488ea64d5965a14a15rc']
}
```

1.2.5. HTTP响应示例

响应 **200**

```
{
  'code': 0,
  'data': {
    'failures': ['e0f99441af3a49979e705830e1b4d609'],
    'successes': [{
      'attendanceDeviceUuid': 'e0f99441af3a49979e705830e1b4d609',
      'uuid': 'b1dafa998fde488ea63d5965a14a15d9'
    }]
  },
  'msg': '成功'
}
```

1.3. 考勤设备列表查询

1.3.1. 基本信息

Path: /v1/api/attendance/device/list

Method: POST

接口描述: 获取考勤设备列表信息。

1.3.2. 请求参数

Headers

参数名称	参数值	是否必须	备注
Content-Type	application/json	是	

Body

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码,不传默认1
pageSize	integer	非必须	分页大小, 不传默认20
name	string	非必须	设备名称, 支持模糊查询
location	string	非必须	设备位置, 支持模糊查询
deviceType	integer	非必须	设备类型: 1-智能摄像头、2-智能面板机、4-网络摄像机
deviceModel	integer	非必须	设备型号 (详细型号描述请查看数据字典)

1.3.3. 返回数据

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码
pageSize	integer	非必须	分页大小
total	integer	非必须	总数
list	object []	非必须	
└─ uuid	string	非必须	考勤设备uuid
└─ name	string	非必须	考勤设备名称
└─ location	string	非必须	设备位置
└─ deviceType	integer	非必须	设备类型: 1-智能摄像头、2-智能面板机、4-网络摄像机
└─ deviceModel	integer	非必须	设备型号 (详细型号描述请查看数据字典)

1.3.4. HTTP请求示例

请求 **path**

```
/v1/api/attendance/device/list
```

请求 **body**

```
{
  "name": "",
  "location": "",
  "deviceModel": null,
  "deviceType": null,
  "pageNum": "1",
  "pageSize": 10
}
```

1.3.5. HTTP响应示例

响应 200

```
{
  'code': 0,
  'data': {
    'list': [{
      'deviceModel': 8,
      'deviceType': 1,
      'location': '测试',
      'name': '智能摄像头-MegEye-C4S-222-人脸抓拍',
      'uuid': '2612f42f1ab6442c861d06bcb9e8e618'
    }
  ],
  'pageNum': 1,
  'pageSize': 20,
  'total': 1
},
  'msg': '成功'
}
```


- 1. 考勤组管理
 - 1.1. 创建考勤组
 - 1.1.1. 基本信息
 - 1.1.2. 请求参数
 - 1.1.3. 返回数据
 - 1.1.4. HTTP请求示例
 - 1.1.5. HTTP响应示例
 - 1.2. 批量删除考勤组
 - 1.2.1. 基本信息
 - 1.2.2. 请求参数
 - 1.2.3. 返回数据
 - 1.2.4. HTTP请求示例
 - 1.2.5. HTTP响应示例
 - 1.3. 更新考勤组
 - 1.3.1. 基本信息
 - 1.3.2. 请求参数
 - 1.3.3. 返回数据
 - 1.3.4. HTTP请求示例
 - 1.3.5. HTTP响应示例
 - 1.4. 考勤员工分页列表查询
 - 1.4.1. 基本信息
 - 1.4.2. 请求参数
 - 1.4.3. 返回数据
 - 1.4.4. HTTP请求示例
 - 1.4.5. HTTP响应示例
 - 1.5. 考勤组分页列表查询
 - 1.5.1. 基本信息
 - 1.5.2. 请求参数
 - 1.5.3. 返回数据
 - 1.5.4. HTTP请求示例
 - 1.5.5. HTTP响应示例
 - 1.6. 考勤组详情查询
 - 1.6.1. 基本信息
 - 1.6.2. 请求参数
 - 1.6.3. 返回数据
 - 1.6.4. HTTP请求示例
 - 1.6.5. HTTP响应示例

1. 考勤组管理

1.1. 创建考勤组

1.1.1. 基本信息

Path: /v1/api/attendance/group/add

Method: POST

接口描述：此接口提供了考勤规则创建以及考勤员工的绑定功能，考勤规则以考勤周、日以及小时的维度提供规则制定；此外考勤组也提供了特殊日期例如法定节假日，公司活动等定制考勤规则；考勤组名称不能重复且一个只能在一个考勤组，人员变动以当前考勤组为准，就考勤规则被覆盖。

1.1.2. 请求参数

Headers

参数名称	参数值	是否必须	备注
Content-Type	application/json	是	

Body

名称	类型	是否必须	
name	string	必须	考勤组名称，唯一，长度2~32字符，不
groupPersons	string []	必须	考勤组员工uuids数组
attendanceTime	object []	必须	考勤时间段
└─ order	integer	必须	时段编号：1-时段一，2-时段二，3-时段
└─ onSt	string	必须	上班开始时间eg:09:00，需要精确到分
└─ onEt	string	必须	上班结束时间eg:09:00，需要精确到分
└─ offSt	string	必须	下班开始时间eg:09:00，需要精确到分
└─ offEt	string	必须	下班结束时间eg:09:00，需要精确到分
specialDate	object []	非必须	特殊日期（考勤周期weekPlan的辅助设 将来的某一天（或多天）是否打卡；2- 立即生效，可将考勤组的创建日期设置
└─ date	string	必须	特殊日期，毫秒级时间戳
└─ isPunch	integer	非必须	是否打卡，默认不打卡，注：0-不打卡
weekPlan	string []	必须	考勤周期LIST，注： ['MONDAY','TUESDAY','WEDNESDAY
daySpan	integer	必须	跨天统计标记:0-第一天；1-第二天

1.1.3. 返回数据

名称	类型	是否必须	
successPerson	string []	非必须	添加成功的人员uuid列表
failurePerson	string []	非必须	添加失败的人员uuid列表
name	string	非必须	考勤组名称
attendanceTime	object []	非必须	考勤时间段
└─ order	integer	非必须	时段编号：1-时段一，2-时段二，3-时段三
└─ onSt	string	非必须	上班开始时间eg:09:00，需要精确到分钟
└─ onEt	string	非必须	上班结束时间eg:09:30，需要精确到分钟
└─ offSt	string	非必须	下班开始时间eg:011:30，需要精确到分钟
└─ offEt	string	非必须	下班结束时间eg:12:00，需要精确到分钟
specialDate	object []	非必须	特殊日期，设置了特殊日期时必有
└─ date	string	非必须	特殊日期，毫秒级时间戳
└─ isPunch	integer	非必须	是否打卡，默认不打卡，注：0-不打卡
weekPlan	string []	非必须	考勤周期LIST，注： ['MONDAY','TUESDAY','WEDNESDAY']
daySpan	integer	非必须	跨天统计标记:0-第一天；1-第二天
uuid	string	非必须	考勤组uuid，添加成功时必有

1.1.4. HTTP请求示例

请求 path

```
/v1/api/attendance/group/add
```

请求 body

```
{
  'name': '考勤4组and跨天第二天',
  'daySpan': 1,
  'groupPersons': ['59615f62a6c8492e877468da97cdaeb2'],
  'attendanceTime': [{
    'onSt': '10:27',
    'onEt': '11:27',
    'offSt': '12:27',
    'offEt': '08:27',
    'order': 1
  }],
  'specialDate': [{
    'date': '2023-03-11',
    'isPunch': 0
  }, {
    'date': '2024-04-29',
    'isPunch': 0
  }],
  'weekPlan': ['MONDAY', 'TUESDAY', 'WEDNESDAY', 'THURSDAY', 'FRIDAY', 'SATURDAY', 'SUNDAY']
}
```

1.1.5. HTTP响应示例

响应 200

```
{
  'code': 0,
  'data': {
    'attendanceTime': [{
      'offEt': '08:27',
      'offSt': '12:27',
      'onEt': '11:27',
      'onSt': '10:27',
      'order': 1
    }
  ],
  'daySpan': 1,
  'failurePerson': [],
  'name': '考勤4组and跨天第二天',
  'specialDate': [{
    'date': 1678464000000,
    'isPunch': 0
  }, {
    'date': 1714320000000,
    'isPunch': 0
  }
  ],
  'successPerson': ['59615f62a6c8492e877468da97cdaeb2'],
  'uuid': 'bfa7cc41e25a42ffb129bc204705d4d',
  'weekPlan': ['MONDAY', 'TUESDAY', 'WEDNESDAY', 'THURSDAY', 'FRIDAY', 'SATURDAY'],
  'msg': '成功'
}
```

1.2. 批量删除考勤组

1.2.1. 基本信息

Path: /v1/api/attendance/group/batchDelete

Method: POST

接口描述: 删除一个/多个考勤组，已经生成的考勤记录和考勤统计结果不受此影响

1.2.2. 请求参数

Headers

参数名称	参数值	是否必须	备注
Content-Type	application/json	是	

Body

名称	类型	是否必须	备注
uuids	string []	必须	考勤组uuids

1.2.3. 返回数据

名称	类型	是否必须	备注
successes	object []	非必须	成功列表
└─ uuid	string	非必须	考勤组uuid
└─ code	integer	非必须	
└─ msg	string	非必须	
failures	object []	非必须	失败列表
└─ uuid	string	非必须	考勤组uuid
└─ code	integer	非必须	
└─ msg	string	非必须	

1.2.4. HTTP请求示例

请求 path

```
/v1/api/attendance/group/batchDelete
```

请求 body

```
{
  'uuids': [ '19ec0adec1244605ab6f7b864b53ef9b', '4c7ab506d8e64823835aa9f2cf7b7146',
}
```

1.2.5. HTTP响应示例

响应 200

```
{
  'code': 0,
  'data': {
    'failures': [],
    'successes': [{
      'code': 0,
      'msg': '成功',
      'uuid': '19ec0adec1244605ab6f7b864b53ef9b'
    }, {
      'code': 0,
      'msg': '成功',
      'uuid': '4c7ab506d8e64823835aa9f2cf7b7146'
    }, {
      'code': 0,
      'msg': '成功',
      'uuid': '938f9abc54a64375ab17d54da4993001'
    }
  ]
},
'msg': '成功'
}
```

1.3. 更新考勤组

1.3.1. 基本信息

Path: /v1/api/attendance/group/update

Method: POST

接口描述: 更新考勤组主要提供考勤组名称、考勤规则、考勤人员的更新功能，对已经生成的考勤记录和考勤统计结果不受影响，接下来的考勤以变更后的规则生成考勤记录。

1.3.2. 请求参数

Headers

参数名称	参数值	是否必须	备注
Content-Type	application/json	是	

Body

名称	类型	是否必须	
uuid	string	必须	考勤组主键uuid
name	string	必须	考勤组名称，唯一，长度2~32字符，不
groupPersons	string []	必须	考勤组员工uuids数组
attendanceTime	object []	必须	考勤时间段
└─ order	integer	必须	时段编号：1-时段一，2-时段二，3-时段
└─ onSt	string	必须	上班开始时间eg:09:00，需要精确到分
└─ onEt	string	必须	上班结束时间eg:09:00，需要精确到分
└─ offSt	string	必须	下班开始时间eg:09:00，需要精确到分
└─ offEt	string	必须	下班结束时间eg:09:00，需要精确到分
specialDate	object []	非必须	特殊日期（考勤周期weekPlan的辅助设 来的某一天（或多天）是否打卡
└─ date	string	必须	特殊日期，毫秒级时间戳
└─ isPunch	integer	非必须	是否打卡，默认不打卡，注：0-不打卡
weekPlan	string []	必须	考勤周期LIST，注： ['MONDAY','TUESDAY','WEDNESDAY
daySpan	integer	必须	考勤记录跨天统计标志，注：0-第一天

1.3.3. 返回数据

名称	类型	是否必须	
name	string	非必须	考勤组名称
attendanceTime	object []	非必须	考勤时间段
└─ order	integer	非必须	时段编号：1-时段一，2-时段二，3-时
└─ onSt	string	非必须	上班开始时间eg:09:00，需要精确到分
└─ onEt	string	非必须	上班结束时间eg:09:00，需要精确到分
└─ offSt	string	非必须	下班开始时间eg:09:00，需要精确到分
└─ offEt	string	非必须	下班结束时间eg:09:00，需要精确到分
specialDate	object []	非必须	特殊日期：设置了特殊日期时必有
└─ date	string	非必须	特殊日期，毫秒级时间戳
└─ isPunch	integer	非必须	是否打卡，默认不打卡，注：0-不打卡
weekPlan	string []	非必须	考勤周期LIST，注： ['MONDAY','TUESDAY','WEDNESDA
daySpan	integer	非必须	考勤记录跨天统计标志，注：0-第一天
successPerson	string []	非必须	添加成功的人员uuid列表
failurePerson	string []	非必须	添加失败的人员uuid列表
uuid	string	非必须	考勤组主键uuid

1.3.4. HTTP请求示例

请求 path

```
/v1/api/attendance/group/update
```

请求 body

```
{
  'uuid': '62a9aa86bc914884884f77adb863a97c',
  'name': 'update考勤3组and3个时段',
  'daySpan': 0,
  'groupPersons': ['0ebea90f7a6f4044b7aa8453975031f1'],
  'attendanceTime': [{
    'onSt': '05:00',
    'onEt': '06:30',
    'offSt': '08:00',
    'offEt': '08:30',
    'order': 1
  }, {
    'onSt': '10:00',
    'onEt': '10:30',
    'offSt': '14:00',
    'offEt': '14:30',
    'order': 2
  }
],
  'specialDate': [{
    'date': '2023-10-28',
    'isPunch': 1
  }, {
    'date': '2023-03-10',
    'isPunch': 1
  }
],
  'weekPlan': ['MONDAY', 'TUESDAY', 'WEDNESDAY', 'THURSDAY', 'FRIDAY']
}
```

1.3.5. HTTP响应示例

响应 200

```

{
  'code': 0,
  'data': {
    'attendanceTime': [{
      'offEt': '08:30',
      'offSt': '08:00',
      'onEt': '06:30',
      'onSt': '05:00',
      'order': 1
    }, {
      'offEt': '14:30',
      'offSt': '14:00',
      'onEt': '10:30',
      'onSt': '10:00',
      'order': 2
    }
  ],
  'daySpan': 0,
  'failurePerson': [],
  'name': 'update考勤3组and3个时段',
  'specialDate': [{
    'date': 1698422400000,
    'isPunch': 1
  }, {
    'date': 1678377600000,
    'isPunch': 1
  }
  ],
  'successPerson': ['0ebea90f7a6f4044b7aa8453975031f1'],
  'uuid': '62a9aa86bc914884884f77adb863a97c',
  'weekPlan': ['MONDAY', 'TUESDAY', 'WEDNESDAY', 'THURSDAY', 'FRIDAY']
},
'msg': '成功'
}

```

1.4. 考勤员工分页列表查询

1.4.1. 基本信息

Path: /v1/api/attendance/group/person/list

Method: POST

接口描述: 获取考勤组/所有考勤组员工信息。

1.4.2. 请求参数

Headers

参数名称	参数值	是否必须	备注
Content-Type	application/json	是	

Body

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码,不传默认1
pageSize	integer	非必须	分页大小, 不传默认20
personName	string	非必须	员工名称, 支持模糊查询
personCode	string	非必须	员工编号, 支持模糊查询

1.4.3. 返回数据

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码
pageSize	integer	非必须	分页大小
total	integer	非必须	总数
list	object []	非必须	
├ groupUuid	string	非必须	考勤组uuid
├ groupName	string	非必须	考勤组名称
├ personUuid	string	非必须	员工uuid
├ personName	string	非必须	员工姓名
├ personCode	string	非必须	员工编号

1.4.4. HTTP请求示例

请求 path

```
/v1/api/attendance/group/person/list
```

请求 body

```
{
  'pageNum': 1,
  'pageSize': 15,
  'personName': '',
  'personCode': ''
}
```

1.4.5. HTTP响应示例

响应 200

```
{
  'code': 0,
  'data': {
    'list': [{
      'groupName': '考勤组备用3c',
      'groupUuid': '7da239698cf04f80804dd09772d4992a',
      'personCode': '196682',
      'personName': '考勤备用员工03',
      'personUuid': '194bbb3827a7457bbff8008cb6ac5212'
    }, {
      'groupName': '考勤组备用2b',
      'groupUuid': 'fe147fd0897440d99b0063cccad19f6a',
      'personCode': '582575',
      'personName': '考勤备用员工02',
      'personUuid': 'bf8ddcd90af54eee975f1bdb7d69f8d7'
    }, {
      'groupName': '考勤组备用1a',
      'groupUuid': '3259c3de22914008b7212f419c321e94',
      'personCode': '431575',
      'personName': '考勤备用员工01',
      'personUuid': 'a948148276694a96bc4c2e4c2701b86f'
    }
  ],
  'pageNum': 1,
  'pageSize': 15,
  'total': 3
},
'msg': '成功'
}
```

1.5. 考勤组分页列表查询

1.5.1. 基本信息

Path: /v1/api/attendance/group/list

Method: POST

接口描述:

1.5.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码,不传默认1
pageSize	integer	非必须	分页大小，不传默认20
groupUuid	string	非必须	考勤组uuid
personName	string	非必须	员工名称，支持模糊查询
personCode	string	非必须	员工编号，支持模糊查询

1.5.3. 返回数据

名称	类型	是否必须	
pageNum	integer	非必须	当前页码
pageSize	integer	非必须	分页大小
total	integer	非必须	总数
list	object []	非必须	
└─ uuid	string	非必须	考勤组uuid
└─ name	string	非必须	考勤组名称
└─ attendanceTime	object []	非必须	考勤时间段
order └─	integer	非必须	时段编号：1-时段一，2-时段二，3-时段三
└─ onSt	string	非必须	上班开始时间eg:09:00，需要精确到分钟
└─ onEt	string	非必须	上班结束时间eg:09:30，需要精确到分钟
└─ offSt	string	非必须	下班开始时间eg:11:00，需要精确到分钟
└─ offEt	string	非必须	下班结束时间eg:12:00，需要精确到分钟
└─ specialDate	object []	非必须	特殊日期：日期-是否打卡
└─ date	string	非必须	特殊日期，毫秒级时间戳
isPunch └─	integer	非必须	是否打卡，默认不打卡，注：0-不打卡，1-打卡

名称	类型	是否必须	
 weekPlan	string []	非必须	考勤周期，注： ['MONDAY','TUESDAY','WEDNESDAY']
 daySpan	integer	非必须	是否跨天: 0-第一天; 1-第二天
 totalPerson	integer	非必须	考勤组总人数

1.5.4. HTTP请求示例

请求 path

```
/v1/api/attendance/group/list
```

请求 body

```
{
  'pageNum': 1,
  'pageSize': 15,
  'groupUuid': '',
  'name': '考勤组备用1a'
}
```

1.5.5. HTTP响应示例

响应 200


```
{
  'code': 0,
  'data': {
    'list': [{
      'attendanceTime': [{
        'offEt': '13:27',
        'offSt': '12:27',
        'onEt': '11:27',
        'onSt': '10:27',
        'order': 1
      }
    ],
    'daySpan': 0,
    'name': '考勤组备用1a',
    'specialDate': [{
      'isPunch': 1
    }
  ],
  'totalPerson': 1,
  'uuid': '788d7394e79f47a28e132acb97845022',
  'weekPlan': ['MONDAY', 'TUESDAY', 'WEDNESDAY', 'THURSDAY', 'FRIDAY', 'SATURDAY', 'SUNDAY']
},
  'pageNum': 1,
  'pageSize': 15,
  'total': 1
},
  'msg': '成功'
}
```

1.6. 考勤组详情查询

1.6.1. 基本信息

Path: /v1/api/attendance/group/query/details

Method: POST

接口描述: 获取考勤组考勤规则详情和考勤组人数。

1.6.2. 请求参数

Headers

参数名称	参数值	是否必须	备注
Content-Type	application/json	是	

Body

名称	类型	是否必须	备注
uuid	string	必须	考勤组uuid

1.6.3. 返回数据

名称	类型	是否必须	
name	string	非必须	考勤组名称，查询成功时必有
groupPersons	object []	非必须	考勤组人员信息，查询成功时必有
└─ personUuid	string	非必须	员工uuid
└─ personName	string	非必须	员工姓名（仅返回前10人）
attendanceTime	object []	非必须	考勤时间段，查询成功时必有
└─ order	integer	非必须	时段编号：1-时段一，2-时段二，3-时段三
└─ onSt	string	非必须	上班开始时间eg:09:00，需要精确到分
└─ onEt	string	非必须	上班结束时间eg:09:00，需要精确到分
└─ offSt	string	非必须	下班开始时间eg:09:00，需要精确到分
└─ offEt	string	非必须	下班结束时间eg:09:00，需要精确到分
specialDate	object []	非必须	特殊日期，设置了特殊日期且查询成功
└─ date	string	非必须	特殊日期，毫秒级时间戳
└─ isPunch	integer	非必须	是否打卡，默认不打卡，注：0-不打卡
weekPlan	string []	非必须	考勤周期LIST，查询成功时必有，注：['MONDAY','TUESDAY','WEDNESDAY']
daySpan	integer	非必须	考勤记录跨日统计选记，查询成功时必有

名称	类型	是否必须	
totalPerson	integer	非必须	考勤组总人数，查询成功时必有
uuid	string	非必须	考勤组uuid，查询成功时必有
code	integer	非必须	
msg	string	非必须	

1.6.4. HTTP请求示例

请求 **path**

```
/v1/api/attendance/group/query/details
```

请求 **body**

```
{
  'uuid': '8ac675e54e30497fac16dab9797643c4'
}
```

1.6.5. HTTP响应示例

响应 **200**

```
{
  'code': 0,
  'data': {
    'attendanceTime': [{
      'offEt': '13:27',
      'offSt': '12:27',
      'onEt': '11:27',
      'onSt': '10:27',
      'order': 1
    }
  ],
  'daySpan': 0,
  'groupPersons': [{
    'personName': '考勤备用员工01',
    'personUuid': '2daa3b6a383e4800b245de279efa03cb'
  }
  ],
  'name': '考勤组备用1a',
  'specialDate': [{
    'date': '2025-03-08',
    'isPunch': 1
  }
  ],
  'totalPerson': 1,
  'uuid': '8ac675e54e30497fac16dab9797643c4',
  'weekPlan': ['MONDAY', 'TUESDAY', 'WEDNESDAY', 'THURSDAY', 'FRIDAY', 'SATURDAY'],
  'msg': '成功'
}
```

- 1. 考勤记录管理
 - 1.1. 员工月考勤记录分页列表查询
 - 1.1.1. 基本信息
 - 1.1.2. 请求参数
 - 1.1.3. 返回数据
 - 1.1.4. HTTP请求示例
 - 1.1.5. HTTP响应示例
 - 1.2. 员工考勤记录日历模式查询
 - 1.2.1. 基本信息
 - 1.2.2. 请求参数
 - 1.2.3. 返回数据
 - 1.2.4. HTTP请求示例
 - 1.2.5. HTTP响应示例
 - 1.3. 考勤记录分页列表查询
 - 1.3.1. 基本信息
 - 1.3.2. 请求参数
 - 1.3.3. 返回数据
 - 1.3.4. HTTP请求示例
 - 1.3.5. HTTP响应示例
 - 1.4. 考勤记录详情分页列表查询
 - 1.4.1. 基本信息
 - 1.4.2. 请求参数
 - 1.4.3. 返回数据
 - 1.4.4. HTTP请求示例
 - 1.4.5. HTTP响应示例
 - 1.5. 考勤记录调整
 - 1.5.1. 基本信息
 - 1.5.2. 请求参数
 - 1.5.3. 返回数据
 - 1.5.4. HTTP请求示例
 - 1.5.5. HTTP响应示例
 - 1.6. 考勤记录回调数据结构
 - 1.6.1. 回调数据示例

1. 考勤记录管理

1.1. 员工月考勤记录分页列表查询

1.1.1. 基本信息

Path: /v1/api/attendance/record/personByMonth/list

Method: POST

接口描述: 获取考勤员工一个月的考勤记录。

1.1.2. 请求参数

Headers

参数名称	参数值	是否必须	备注
Content-Type	application/json	是	

Body

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码,不传默认1
pageSize	integer	非必须	分页大小, 不传默认20
uuid	string	必须	考勤统计记录uuid, 通过考勤统计记录uuid查询考勤记录

1.1.3. 返回数据

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码
pageSize	integer	非必须	分页大小
total	integer	非必须	总数
list	object []	非必须	
└─ uuid	string	非必须	考勤记录uuid
└─ personName	string	非必须	员工姓名
└─ personCode	string	非必须	员工编号
└─ attendanceDate	string	非必须	考勤日期，时间戳
└─ attTimeResult	object []	非必须	考勤时段结果：打卡时间&考勤结果
order └─	integer	非必须	时段编号：1-时段一，2-时段二，3-时段三
inPunch └─	string	非必须	上班打卡时间eg:09:00:00，需要精确到秒
outPunch └─	string	非必须	下班打卡时间eg:11:00:00，需要精确到秒
result └─	integer	非必须	考勤结果：0-暂无结果（'-'表示）、1-正常、2-缺勤、3-迟到、4-早退、5-迟到早退、6-漏打卡
└─ status	integer	非必须	考勤状态，注：：0-暂无结果（'-'表示），1-正常，2-旷工，3-出勤调整，4-异常

1.1.4. HTTP请求示例

请求 path

```
/v1/api/attendance/record/personByMonth/list
```

请求 body

```
{
  'pageNum': 1,
  'pageSize': 20,
  'uuid': '0d03c5c7eb804a8d8e9f873f56c0053f'
}
```

1.1.5. HTTP响应示例

响应 200

```
{
  {
    'code': 0,
    'data': {
      'list': [{
        'attTimeResult': [{
          'inPunch': '15:49:41',
          'order': 1,
          'outPunch': '--',
          'result': 6
        }
      ],
      'attendanceDate': 1622390400000,
      'personCode': '118347',
      'personName': '考勤记录备用员工01',
      'status': 4,
      'uuid': '196d09c65707480f94a5e6876374c099'
    }
  ],
  'pageNum': 1,
  'pageSize': 20,
  'total': 1
},
  'msg': '成功'
}
```

1.2. 员工考勤记录日历模式查询

1.2.1. 基本信息

Path: /v1/api/attendance/record/calendar

Method: POST

接口描述: 用于日历模式展示数据，以自然月为单位展示。

1.2.2. 请求参数

Headers

参数名称	参数值	是否必须	备注
Content-Type	application/json	是	

Body

名称	类型	是否必须	备注
uuid	string	必须	考勤统计记录uuid，通过考勤统计记录查询考勤记录

1.2.3. 返回数据

名称	类型	是否必须	备注
	object []	非必须	
└─ uuid	string	非必须	考勤记录uuid
└─ personName	string	非必须	员工姓名
└─ personCode	string	非必须	员工编号
└─ attendanceDate	string	非必须	考勤日期，时间戳
└─ attTimeResult	object []	非必须	考勤时段结果：打卡时间&考勤结果
order └─	integer	非必须	时段编号：1-时段一，2-时段二，3-时段三
inPunch └─	string	非必须	上班打卡时间eg:09:00:00，需要精确到秒
outPunch └─	string	非必须	下班打卡时间eg:11:00:00，需要精确到秒
result └─	integer	非必须	考勤结果：0-暂无结果（'-1'表示）、1-正常、2-缺勤、3-迟到、4-早退、5-迟到早退、6-漏打卡
└─ status	integer	非必须	考勤状态，注：：0-暂无结果（'-1'表示），1-正常，2-旷工，3-出勤调整，4-异常

1.2.4. HTTP请求示例

请求 path

```
/v1/api/attendance/record/calendar
```

请求 body

```
{
  'uuid': '0d03c5c7eb804a8d8e9f873f56c0053f'
}
```

1.2.5. HTTP响应示例

响应 200

```
{
  'code': 0,
  'data': [{
    'attTimeResult': [{
      'inPunch': '15:49:41',
      'order': 1,
      'outPunch': '--',
      'result': 6
    }
  ],
  'attendanceDate': 1622390400000,
  'personCode': '118347',
  'personName': '考勤记录备用员工01',
  'status': 4,
  'uuid': '196d09c65707480f94a5e6876374c099'
},
  'msg': '成功'
}
```

1.3. 考勤记录分页列表查询

1.3.1. 基本信息

Path: /v1/api/attendance/record/list

Method: POST

接口描述: 分页获取考勤记录。

1.3.2. 请求参数

Headers

参数名称	参数值	是否必须	备注
Content-Type	application/json	是	

Body

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码,不传默认1
pageSize	integer	非必须	分页大小, 不传默认20
personName	string	非必须	员工姓名, 支持模糊查询
personCode	string	非必须	员工编号, 支持模糊查询
startTime	string	非必须	开始时间, 毫秒级时间戳
endTime	string	非必须	截止时间, 毫秒级时间戳

1.3.3. 返回数据

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码
pageSize	integer	非必须	分页大小
total	integer	非必须	总数
list	object []	非必须	
└─ uuid	string	非必须	考勤记录uuid
└─ personName	string	非必须	员工姓名
└─ personCode	string	非必须	员工编号
└─ attendanceDate	string	非必须	考勤日期，时间戳
└─ attTimeResult	object []	非必须	考勤时段结果：打卡时间&考勤结果
order └─	integer	非必须	时段编号：1-时段一，2-时段二，3-时段三
inPunch └─	string	非必须	上班打卡时间eg:09:00:00，需要精确到秒
outPunch └─	string	非必须	下班打卡时间eg:11:00:00，需要精确到秒
result └─	integer	非必须	考勤结果：0-暂无结果（'-'表示）、1-正常、2-缺勤、3-迟到、4-早退、5-迟到早退、6-漏打卡
└─ status	integer	非必须	考勤状态，注：：0-暂无结果（'-'表示），1-正常，2-旷工，3-出勤调整，4-异常

1.3.4. HTTP请求示例

请求 path

```
/v1/api/attendance/record/list
```

请求 **body**

```
{
  "personName": "",
  "personCode": "",
  "startTime": 1622217600000,
  "endTime": 1624895999000,
  "pageNum": 1,
  "pageSize": 10
}
```

1.3.5. HTTP响应示例

响应 **200**

```

{
  'code': 0,
  'data': {
    'list': [{
      'attTimeResult': [{
        'inPunch': '--',
        'order': 1,
        'outPunch': '--',
        'result': 2
      }, {
        'inPunch': '16:19:04',
        'order': 2,
        'outPunch': '--',
        'result': 0
      }
    ],
    'attendanceDate': 1622390400000,
    'personCode': '994466',
    'personName': '考勤记录备用员工04',
    'status': 4,
    'uuid': '010a8b2f13b84bab8427b69af1a3003a'
  }, {
    'attTimeResult': [{
      'inPunch': '15:52:46',
      'order': 1,
      'outPunch': '--',
      'result': 6
    }
  ],
  'attendanceDate': 1622390400000,
  'personCode': '089162',
  'personName': '考勤记录备用员工01',
  'status': 4,
  'uuid': 'a0b542af2d4046be8169e6ff3f6c30c9'
}, {
  'attTimeResult': [{
    'inPunch': '--',
    'order': 1,
    'outPunch': '--',
    'result': 2
  }, {
    'inPunch': '--',
    'order': 2,
    'outPunch': '--',
    'result': 2
  }, {
    'inPunch': '15:49:37',
    'order': 3,
    'outPunch': '--',
    'result': 0
  }
],
  'attendanceDate': 1622390400000,
  'personCode': '708594',
  'personName': '考勤记录备用员工03',
  'status': 4,
  'uuid': 'abb26b51297542288ad1b5bbd3b8e4ea'
}, {
  'attTimeResult': [{
    'inPunch': '--',
    'order': 1,
    'outPunch': '--',
    'result': 2
  }, {
    'inPunch': '15:49:42',

```

```

        'order': 2,
        'outPunch': '--',
        'result': 0
    }
],
    'attendanceDate': 1622390400000,
    'personCode': '716629',
    'personName': '考勤记录备用员工02',
    'status': 4,
    'uuid': '91127128a4634a4dac4fa5d7453400ae'
}
],
    'pageNum': 1,
    'pageSize': 20,
    'total': 12
},
    'msg': '成功'
}

```

1.4. 考勤记录详情分页列表查询

1.4.1. 基本信息

Path: /v1/api/attendance/record/detail/list

Method: POST

接口描述: 获取考勤记录详细信息，展示考勤时间区间内通行的详细信息。

1.4.2. 请求参数

Headers

参数名称	参数值	是否必须	备注
Content-Type	application/json	是	

Body

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码,不传默认1
pageSize	integer	非必须	分页大小, 不传默认20
uuid	string	必须	考勤记录uuid

1.4.3. 返回数据

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码
pageSize	integer	非必须	分页大小
total	integer	非必须	总数
list	object []	非必须	
└─ personUuid	string	非必须	员工uuid
└─ personName	string	非必须	员工姓名
└─ personCode	string	非必须	员工编号
└─ deviceName	string	非必须	设备名称
└─ deviceLocation	string	非必须	设备位置
└─ snapshotUri	string	非必须	抓拍照片url
└─ personImageUri	string	非必须	底库照片url
└─ passTime	string	非必须	通行时间，毫秒级时间戳

1.4.4. HTTP请求示例

请求 path

```
/v1/api/attendance/record/detail/list
```

请求 body

```
{
  'pageNum': 1,
  'pageSize': 20,
  'uuid': '010a8b2f13b84bab8427b69af1a3003a'
}
```

1.4.5. HTTP响应示例

响应 200


```
{
  'code': 0,
  'data': {
    'list': [{
      'deviceLocation': 'OpenAPI测试',
      'deviceName': '网络摄像机-MegEye-C4R-222-人脸识别',
      'passTime': 1622449196901,
      'personCode': '994466',
      'personImageUri': 'http://10.171.4.130/pub/_ZzEwMF9mb3JldmVyQnVja2V0_5',
      'personName': '考勤记录备用员工04',
      'personUuid': '87a3e7456cc34303af431b79476936a7',
      'snapshotUri': 'http://10.171.4.130/pub/_ZzEwMF82bQ==_1c63b1b776634f00',
    }, {
      'deviceLocation': 'OpenAPI测试',
      'deviceName': '网络摄像机-MegEye-C4R-322-人脸识别',
      'passTime': 1622449194902,
      'personCode': '994466',
      'personImageUri': 'http://10.171.4.130/pub/_ZzEwMF9mb3JldmVyQnVja2V0_5',
      'personName': '考勤记录备用员工04',
      'personUuid': '87a3e7456cc34303af431b79476936a7',
      'snapshotUri': 'http://10.171.4.130/pub/_ZzEwMF82bQ==_0d49b7c36dc54227',
    }
  ],
  'pageNum': 1,
  'pageSize': 20,
  'total': 4
},
'msg': '成功'
}
```

1.5. 考勤记录调整

1.5.1. 基本信息

Path: /v1/api/attendance/record/adjust

Method: POST

接口描述: 以天为时间单位，针对考勤记录做调整，考勤结果为正常或已经是出勤调整的无法调整，考勤结果和考勤统计实时变更。

1.5.2. 请求参数

Headers

参数名称	参数值	是否必须	备注
Content-Type	application/json	是	

Body

名称	类型	是否必须	备注
uuid	string	必须	考勤记录uuid
adjustType	integer	必须	出勤调整类型，注：：1-补卡、2-请假、3-调休、4-调班
adjustReason	string	非必须	调整原因，注：调整原因长度为0-50个字符

1.5.3. 返回数据

名称	类型	是否必须	备注
adjustRecordUuid	string	非必须	出勤调整记录uuid，调整成功时必有值
uuid	string	非必须	考勤记录uuid
adjustType	integer	非必须	出勤调整类型，注：：1-补卡、2-请假、3-调休、4-调班
adjustReason	string	非必须	调整原因

1.5.4. HTTP请求示例

请求 path

```
/v1/api/attendance/record/adjust
```

请求 body

```
{
  'uuid': '34b68646a6634fbea214535685df198c',
  'adjustType': 1,
  'adjustReason': '出勤调整'
}
```

1.5.5. HTTP响应示例

响应 200

```
{
  'code': 0,
  'data': {
    'adjustReason': '出勤调整',
    'adjustRecordUuid': '341e150d24f7484384ba24471abebe55',
    'adjustType': 1,
    'uuid': '34b68646a6634fbea214535685df198c'
  },
  'msg': '成功'
}
```

1.6. 考勤记录回调数据结构

名称	类型	是否必须	备注
uuid	string	非必须	考勤记录uuid
personName	string	非必须	员工姓名
personCode	string	非必须	员工编号
attendanceDate	string	非必须	考勤日期，时间戳
attTimeResult	object []	非必须	考勤时段结果：打卡时间&考勤结果
└─ order	integer	非必须	时段编号：1-时段一，2-时段二，3-时段三
└─ inPunch	string	非必须	上班打卡时间eg:09:00:00，需要精确到秒
└─ outPunch	string	非必须	下班打卡时间eg:11:00:00，需要精确到秒
└─ result	integer	非必须	考勤结果：0-暂无结果（'-'表示）、1-正常、2-缺勤、3-迟到、4-早退、5-迟到早退、6-漏打卡
status	integer	非必须	考勤状态，注：：0-暂无结果（'-'表示），1-正常，2-旷工，3-出勤调整，4-异常

1.6.1. 回调数据示例

响应 200

```
{
  "attTimeResult": [{
    "adjust": -1,
    "inPunch": "09:00:00",
    "order": 1,
    "outPunch": "10:00:00",
    "period": 2,
    "result": 4
  }],
  "attendanceDate": 1624896000000,
  "personCode": "5677978",
  "personName": "彭斯亮",
  "personUuid": "2a12603fc2db4ebeb6dd8d6004e0c93",
  "status": 4,
  "uuid": "26862b1abeb64c4d869add1d225e5ebe"
}
```

- 1. 考勤统计管理
 - 1.1. 考勤统计记录分页列表查询
 - 1.1.1. 基本信息
 - 1.1.2. 请求参数
 - 1.1.3. 返回数据
 - 1.1.4. HTTP请求示例
 - 1.1.5. HTTP响应示例

1. 考勤统计管理

1.1. 考勤统计记录分页列表查询

1.1.1. 基本信息

Path: /v1/api/attendance/statistic/list

Method: POST

接口描述: 统计考勤人员一个月内的考勤状况，分别以考勤时间段、考勤天为时间维度，细粒度的展示了员工的月出勤、日出勤和时段出勤情况以及请假等。

1.1.2. 请求参数

Headers

参数名称	参数值	是否必须	备注
Content-Type	application/json	是	

Body

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码,不传默认1
pageSize	integer	非必须	分页大小，不传默认20
name	string	非必须	员工姓名 支持模糊查询
groupUuid	string	非必须	考勤组uuid
startTime	string	非必须	查询开始时间，毫秒级时间戳
endTime	string	非必须	查询截止时间，毫秒级时间戳

1.1.3. 返回数据

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码
pageSize	integer	非必须	分页大小
total	integer	非必须	总数
list	object []	非必须	
└─ uuid	string	非必须	考勤统计记录 uuid
└─ personName	string	非必须	员工姓名
└─ personCode	string	非必须	员工编号
└─ groupName	string	非必须	考勤组名称
└─ workDays	integer	非必须	应出勤天数
└─ normalDays	integer	非必须	正常天数
└─ adjustDays	integer	非必须	调整天数
└─ absentDays	integer	非必须	旷工天数
└─ lateTimes	integer	非必须	迟到次数
└─ leaveEarlyTimes	integer	非必须	早退次数
└─ lateLeaveEarlyTimes	integer	非必须	迟到早退次数
└─ omitPunchTimes	integer	非必须	漏打卡次数
└─ absentTimes	integer	非必须	缺勤次数

1.1.4. HTTP请求示例

请求 path

```
/v1/api/attendance/statistic/list
```

请求 body

```
{
  "name": "",
  "personCode": "",
  "startTime": 1622476800000,
  "endTime": 1625068799000,
  "pageNum": 1,
  "pageSize": 10
}
```

1.1.5. HTTP响应示例

响应 200

```
{
  'code': 0,
  'data': {
    'list': [{
      'absentDays': 0,
      'absentTimes': 0,
      'adjustDays': 0,
      'groupName': '考勤组记录备用1a',
      'lateLeaveEarlyTimes': 0,
      'lateTimes': 0,
      'leaveEarlyTimes': 0,
      'normalDays': 0,
      'omitPunchTimes': 1,
      'personCode': '118347',
      'personName': '考勤记录备用员工01',
      'uuid': '0d03c5c7eb804a8d8e9f873f56c0053f',
      'workDays': 1
    }, {
      'absentDays': 0,
      'absentTimes': 0,
      'adjustDays': 0,
      'groupName': '考勤组记录备用1a',
      'lateLeaveEarlyTimes': 0,
      'lateTimes': 0,
      'leaveEarlyTimes': 0,
      'normalDays': 0,
      'omitPunchTimes': 1,
      'personCode': '089162',
      'personName': '考勤记录备用员工01',
      'uuid': '59418e8ab0ca43579b920ddc70ac3f9d',
      'workDays': 1
    }, {
      'absentDays': 0,
      'absentTimes': 1,
      'adjustDays': 0,
      'groupName': '考勤组记录备用2b',
      'lateLeaveEarlyTimes': 0,
      'lateTimes': 0,
      'leaveEarlyTimes': 0,
      'normalDays': 0,
      'omitPunchTimes': 0,
      'personCode': '279983',
      'personName': '考勤记录备用员工04',
      'uuid': '4cd3bfff54d614ad390704bdff5ad555d',
      'workDays': 1
    }
  ],
  'pageNum': 1,
  'pageSize': 20,
  'total': 8
},
'msg': '成功'
}
```

- 1. 出勤调整管理
 - 1.1. 出勤调整
 - 1.1.1. 基本信息
 - 1.1.2. 请求参数
 - 1.1.3. 返回数据
 - 1.1.4. HTTP请求示例
 - 1.1.5. HTTP响应示例
 - 1.2. 出勤调整记录分页列表查询
 - 1.2.1. 基本信息
 - 1.2.2. 请求参数
 - 1.2.3. 返回数据
 - 1.2.4. HTTP请求示例
 - 1.2.5. HTTP响应示例
 - 1.3. 更新出勤调整记录
 - 1.3.1. 基本信息
 - 1.3.2. 请求参数
 - 1.3.3. 返回数据
 - 1.3.4. HTTP请求示例
 - 1.3.5. HTTP响应示例

1. 出勤调整管理

1.1. 出勤调整

1.1.1. 基本信息

Path: /v1/api/attendance/adjust/add

Method: POST

接口描述: 出勤调整主要对考勤员工异常的考勤记录进行正常状态修正，且考勤组有该员工时可操作；可修正的状态有补卡、请假、调休与调班；调整的时间范围是自然月5号（包含5号）以前可以修改上一个月异常的考勤记录，5号以后仅修改本月的考勤记录；也可对未来将要生成的考勤结果调整，出勤调整的优先级比正常打卡高；调整实时生效，考勤统计实时更新。

1.1.2. 请求参数

Headers

参数名称	参数值	是否必须	备注
Content-Type	application/json	是	

Body

名称	类型	是否必须	备注
personUuid	string	必须	员工uuid
adjustType	integer	必须	调整类型：1-补卡、2-请假、3-调休、4-调班
adjustTimeBegin	string	必须	调整时段开始时间，毫秒级时间戳
adjustTimeEnd	string	必须	调整时段截止时间，毫秒级时间戳
adjustReason	string	非必须	调整原因，注：调整原因长度为0-50个字符

1.1.3. 返回数据

名称	类型	是否必须	备注
personUuid	string	非必须	员工uuid
adjustType	integer	非必须	调整类型：1-补卡、2-请假、3-调休、4-调班
adjustTimeBegin	string	非必须	调整时段开始时间，毫秒级时间戳
adjustTimeEnd	string	非必须	调整时段截止时间，毫秒级时间戳
adjustReason	string	非必须	调整原因，注：调整事由长度为0-50个字符
uuid	string	非必须	考勤调整记录uuid，出勤调整成功时必有

1.1.4. HTTP请求示例

请求 path

```
/v1/api/attendance/adjust/add
```

请求 body

```
{
  'personUuid': '0b0dc7d3ff5b45e687140301d192957d',
  'adjustType': 1,
  'adjustTimeBegin': 1622434654000,
  'adjustTimeEnd': 1622535454000,
  'adjustReason': 'adjustReason01'
}
```

1.1.5. HTTP响应示例

响应 200

```
{
  'code': 0,
  'data': {
    'adjustReason': 'adjustReason01',
    'adjustTimeBegin': 1622434654000,
    'adjustTimeEnd': 1622535454000,
    'adjustType': 1,
    'personUuid': '0b0dc7d3ff5b45e687140301d192957d',
    'uuid': 'affc2c30d6ea43728730fc06ccfee28c'
  },
  'msg': '成功'
}
```

1.2. 出勤调整记录分页列表查询

1.2.1. 基本信息

Path: /v1/api/attendance/adjust/list

Method: POST

接口描述: 获取调整记录列表。

1.2.2. 请求参数

Headers

参数名称	参数值	是否必须	备注
Content-Type	application/json	是	

Body

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码,不传默认1
pageSize	integer	非必须	分页大小, 不传默认20
personName	string	非必须	姓名 支持模糊查询
adjustType	integer	非必须	调整类型: 1-补卡、2-请假、3-调休、4-调班
startTime	string	非必须	查询开始时间, 毫秒级时间戳
endTime	string	非必须	查询截止时间, 毫秒级时间戳

1.2.3. 返回数据

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码
pageSize	integer	非必须	分页大小
total	integer	非必须	总数
list	object []	非必须	
 personName	string	非必须	员工姓名
 personCode	string	非必须	员工编号
 groupName	string	非必须	考勤组名称
 adjustTimeBegin	string	非必须	调整时段开始时间，毫秒级时间戳
 adjustTimeEnd	string	非必须	调整时段截止时间，毫秒级时间戳
 adjustType	integer	非必须	调整类型：1-补卡、2-请假、3-调休、4-调班
 adjustReason	string	非必须	调整原因

1.2.4. HTTP请求示例

请求 **path**

```
/v1/api/attendance/adjust/list
```

请求 **body**

```
{
  "personName": "考勤备用员工",
  "adjustType": 1,
  "personCode": "",
  "pageNum": 1,
  "pageSize": 10,
  "startTime": 1622217600742,
  "endTime": 1624895999742
}
```

1.2.5. HTTP响应示例

响应 **200**

```

{
  'code': 0,
  'data': {
    'list': [{
      'adjustReason': 'adjustReason04',
      'adjustTimeBegin': 1622434654000,
      'adjustTimeEnd': 1622535454000,
      'adjustType': 4,
      'groupName': '考勤组备用1a',
      'personCode': '323183',
      'personName': '考勤备用员工01'
    }, {
      'adjustReason': 'adjustReason03',
      'adjustTimeBegin': 1622434654000,
      'adjustTimeEnd': 1622535454000,
      'adjustType': 3,
      'groupName': '考勤组备用3c',
      'personCode': '636041',
      'personName': '考勤备用员工03'
    }, {
      'adjustReason': 'adjustReason02',
      'adjustTimeBegin': 1622434654000,
      'adjustTimeEnd': 1622535454000,
      'adjustType': 2,
      'groupName': '考勤组备用2b',
      'personCode': '134634',
      'personName': '考勤备用员工02'
    }
  ],
  'pageNum': 1,
  'pageSize': 20,
  'total': 20
},
'msg': '成功'
}

```

1.3. 更新出勤调整记录

1.3.1. 基本信息

Path: /v1/api/attendance/adjust/update

Method: POST

接口描述: 对已经调整的调整记录进行修改，此操作同出勤调整添加操作，但是对已经生效的出勤调整产生影响，从而导致考勤结果和考勤统计记录的变化；具体表现为两次调整区间相交部分，相交的时间区间内的考勤解果不会有影响，不相交区间内的考勤结果的若是被第一次考勤调整则将被还原成调整前的状态，未被第一调整的时间区间内的考勤结果将被调整。

1.3.2. 请求参数

Headers

参数名称	参数值	是否必须	备注
Content-Type	application/json	是	

Body

名称	类型	是否必须	备注
uuid	string	必须	出行调整记录主键uuid
adjustType	integer	必须	调整类型：1-补卡、2-请假、3-调休、4-调班
adjustTimeBegin	string	必须	调整时段开始时间
adjustTimeEnd	string	必须	调整时段截止时间
adjustReason	string	非必须	调整事由

1.3.3. 返回数据

名称	类型	是否必须	备注
code	string	非必须	消息提示编码
msg	string	非必须	消息提示

1.3.4. HTTP请求示例

请求 path

```
/v1/api/attendance/adjust/update
```

请求 body

```
{
  'personUuid': '2fd98a319ea04fe497b786bc1df68002',
  'adjustType': 1,
  'adjustTimeBegin': 1622434655000,
  'adjustTimeEnd': 1622535455000,
  'adjustReason': 'adjustReason01'
}
```

1.3.5. HTTP响应示例

响应 200

```
{
  'code': 0,
  'msg': '成功'
}
```

- 1. 规则管理
 - 1.1. 添加人员布控规则
 - 1.1.1. 基本信息
 - 1.1.2. 请求参数
 - 1.1.3. 返回数据
 - 1.1.4. HTTP请求示例
 - 1.1.5. HTTP响应示例
 - 1.2. 修改人员布控规则
 - 1.2.1. 基本信息
 - 1.2.2. 请求参数
 - 1.2.3. HTTP请求示例
 - 1.2.4. HTTP响应示例
 - 1.3. 批量删除人员布控规则
 - 1.3.1. 基本信息
 - 1.3.2. 请求参数
 - 1.3.3. 返回数据
 - 1.3.4. HTTP请求示例
 - 1.3.5. HTTP响应示例
 - 1.4. 人员布控规则分页查询
 - 1.4.1. 基本信息
 - 1.4.2. 请求参数
 - 1.4.3. 返回数据
 - 1.4.4. HTTP请求示例
 - 1.4.5. HTTP响应示例

1. 规则管理

1.1. 添加人员布控规则

1.1.1. 基本信息

Path: /v1/api/pass/rule/addBatch

Method: POST

接口描述: 添加人员布控规则

1.1.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
ruleList	object []	必须	规则列表
└─ name	string	必须	名称, 格式: 汉字,大小写英文,数字, 长度:[1,50]
└─ scheduleUUid	string	必须	时间计划uuid
└─ type	integer	非必须	规则类型, 与typeList二选一, 如果同时存在以typeList为主。 值的范围: 1 重点人员 2 陌生人, 如果传递其他值不进行处理
└─ typeList	integer []	非必须	规则类型: 与type二选一, 如果同时存在以typeList为主。 值的范围: 1 重点人员, 2 全局陌生人 3 员工, 4 访客
└─ deviceUUidList	string []	必须	设备uuid列表
└─ groupUUidList	string []	非必须	组uuid列表, 当规则类型为 1 (重点人员), 3 (员工), 4 (访客) 必填; 全局陌生人布控无须填写

1.1.3. 返回数据

名称	类型	是否必须	备注
successes	object []	非必须	成功列表
└─ code	integer	非必须	结果状态码
└─ msg	string	非必须	结果信息提示
└─ uuid	string	非必须	uuid
└─ name	string	非必须	名称, 格式: 汉字,大小写英文,数字, 长度:[1,50]
└─ scheduleUUID	string	非必须	时间计划uuid
└─ type	integer	非必须	规则类型, 与typeList二选一, 如果同时存在以typeList为主。 值的范围: 1 重点人员 2 陌生人, 如果传递其他值不进行处理
└─ typeList	integer []	非必须	规则类型: 与type二选一, 如果同时存在以typeList为主。 值的范围: 1 重点人员, 2 全局陌生人 3 员工, 4 访客
└─ deviceUUIDList	string []	非必须	设备uuid列表
└─ groupUUIDList	string []	非必须	组uuid列表, 当规则类型为 1 (重点人员), 3 (员工), 4 (访客) 必填
successes	object []	非必须	成功列表
└─ code	integer	非必须	结果状态码
└─ msg	string	非必须	结果信息提示
└─ uuid	string	非必须	uuid

名称	类型	是否必须	备注
└─ name	string	非必须	名称, 格式: 汉字,大小写英文,数字, 长度:[1,50]
└─ scheduleUUID	string	非必须	时间计划uuid
└─ type	integer	非必须	规则类型, 与typeList二选一, 如果同时存在以typeList为主。 值的范围: 1 重点人员 2 陌生人, 如果传递其他值不进行处理
└─ typeList	integer []	非必须	规则类型: 与type二选一, 如果同时存在以typeList为主。 值的范围: 1 重点人员, 2 全局陌生人 3 员工, 4 访客
└─ deviceUUIDList	string []	非必须	设备uuid列表
└─ groupUUIDList	string []	非必须	组uuid列表, 当规则类型为 1 (重点人员), 3 (员工), 4 (访客) 必填

1.1.4. HTTP请求示例

请求 path

```
/v1/api/pass/rule/addBatch
```

请求 body

```
{
  "ruleList": [
    {
      "name": "员工访客布控规则",
      "type": 1,
      "typeList": [
        3,
        4
      ],
      "scheduleUUID": "e20fddee76914431ad8a4e9a49fa47e9",
      "deviceUUIDList": [
        "588d56169e3d4587b8c0010316c30cab"
      ],
      "groupUUIDList": [
        "edb068db35d849bebd8692a45692ebed",
        "9dade394f15e4907af906dcfe9718294"
      ]
    }
  ]
}
```

1.1.5. HTTP响应示例

响应 200

```
{
  "code": 0,
  "data": {
    "failures": [],
    "successes": [
      {
        "deviceUUidList": [
          "588d56169e3d4587b8c0010316c30cab"
        ],
        "groupUUidList": [
          "edb068db35d849bebd8692a45692ebed",
          "9dade394f15e4907af906dcfe9718294"
        ],
        "name": "员工访客布控规则",
        "scheduleUUid": "e20fddee76914431ad8a4e9a49fa47e9",
        "type": 1,
        "typeList": [
          3,
          4
        ],
        "uuid": "Fb4b0bcb684a4dc180191edeec787560"
      }
    ]
  },
  "msg": "成功"
}
```

1.2. 修改人员布控规则

1.2.1. 基本信息

Path: /v1/api/pass/rule/update

Method: POST

接口描述: 修改人员布控规则

1.2.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
uuid	string	必须	uuid
name	string	非必须	名称, 格式: 汉字,大小写英文,数字, 长度:[1,50]
scheduleUUID	string	必须	时间计划UUID
deviceList	string []	必须	设备列表, 会进行差集计算, 属于全量修改, 不支持增量。
groupList	string []	非必须	组列表, 当规则类型为1（重点人员），3（员工），4（访客）必填。会进行差集计算, 属于全量修改, 不支持增量。
typeList	integer []	非必须	规则类型: 1 重点人员, 2 全局陌生人, 3 员工, 4 访客。会进行差集计算, 属于全量修改, 不支持增量。

1.2.3. HTTP请求示例

请求 path

```
/v1/api/pass/rule/update
```

请求 body

```
{
  "uuid": "28af6d0fd7064978a9cd4553861dda7e",
  "name": "update员工访客重点人员规则布控",
  "scheduleUUID": "8ff53895e76d4b86b3b0d01e8a9caae0",
  "deviceList": [
    "a1233f7f1ae94bb0822d5a8e63b349eb"
  ],
  "groupList": [
    "bf8c07e5aa40433fb2ed2f95bf5ca733",
    "976055aae31445c5a583ebc1a53c7514"
  ],
  "typeList": [
    1,
    3
  ]
}
```

1.2.4. HTTP响应示例

响应 200

```
{
  "code": 0,
  "msg": "成功"
}
```

1.3. 批量删除人员布控规则

1.3.1. 基本信息

Path: /v1/api/pass/rule/deleteBatch

Method: POST

接口描述: 删除人员布控规则

1.3.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
ruleIdList	string []	必须	规则uuid列表

1.3.3. 返回数据

名称	类型	是否必须	备注
successes	object []	非必须	成功列表
├─ uuid	string	非必须	uuid
├─ code	integer	非必须	结果状态码
├─ msg	string	非必须	结果信息提示
failures	object []	非必须	失败列表
├─ uuid	string	非必须	uuid
├─ code	integer	非必须	结果状态码
├─ msg	string	非必须	结果信息提示

1.3.4. HTTP请求示例

请求 path

```
/v1/api/pass/rule/deleteBatch
```

请求 body

```
{
  "ruleIdList": [
    "e8751b5f47c24edba681ad23d796f684",
    "e8b4681e627d4073b7a69d5d4b1215e9"
  ]
}
```

1.3.5. HTTP响应示例

响应 200

```
{
  "code": 0,
  "data": {
    "failures": [],
    "successes": [
      {
        "uuid": "e8751b5f47c24edba681ad23d796f684"
      },
      {
        "uuid": "e8b4681e627d4073b7a69d5d4b1215e9"
      }
    ]
  },
  "msg": "成功"
}
```

1.4. 人员布控规则分页查询

1.4.1. 基本信息

Path: /v1/api/pass/rule/list

Method: POST

接口描述: 根据条件分页查询人员布控规则

1.4.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码,不传默认1
pageSize	integer	非必须	分页大小, 不传默认20
type	integer	必须	类型, 1 非陌生人 (员工、访客、重点人员), 2 陌生人
name	string	非必须	名称, 支持模糊查询
scheduleUuid	string	非必须	时间计划uuid
deviceUuids	string []	非必须	设备uuid列表

1.4.3. 返回数据

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码
pageSize	integer	非必须	分页大小
total	integer	非必须	总数
list	object []	非必须	
└─ uuid	string	非必须	uuid
└─ name	string	非必须	名称
└─ type	integer	非必须	规则类型, 与typeList二选一, 如果同时存在以typeList为主。 值的范围: 1 重点人员 2 陌生人, 如果传递其他值不进行处理
└─ typeList	integer []	非必须	规则类型: 与type二选一, 如果同时存在以typeList为主。 值的范围: 1 重点人员, 2 全局陌生人 3 员工, 4 访客
└─ scheduleUuid	string	非必须	时间计划uuid
└─ scheduleName	string	非必须	时间计划名称
└─ deviceList	object []	非必须	设备列表
└─ deviceUuid	string	非必须	设备uuid
└─ name	string	非必须	设备名字

名称	类型	是否必须	备注
└ groupList	object []	非必须	组列表, 当规则类型为 1（重点人员），3（员工），4（访客）必须, 全局陌生人时是空列表
└ name	string	非必须	组名字
└ uuid	string	非必须	组uuid

1.4.4. HTTP请求示例

请求 path

```
/v1/api/pass/rule/list
```

请求 body

```
{
  "pageNum": 1,
  "pageSize": 20,
  "name": "员工访客重点人员规则布控",
  "type": 1,
  "deviceUUIs": []
}
```

1.4.5. HTTP响应示例

响应 200


```

{
  "code": 0,
  "data": {
    "list": [
      {
        "deviceList": [
          {
            "deviceUuid": "bbc1305df66b4b00a23cf8bb3cce8640",
            "name": "智能摄像头-MegEye-C4H-242-人脸抓拍"
          }
        ],
        "groupList": [
          {
            "name": "员工默认组",
            "uuid": "976055aae31445c5a583ebc1a53c7514"
          },
          {
            "name": "访客默认组",
            "uuid": "90a41fddf76a4daa9dae47762ecfd359"
          },
          {
            "name": "重点人员默认组",
            "uuid": "127d6dadbf1c4e3f93be34865ad0d135"
          }
        ],
        "name": "员工访客重点人员规则布控2",
        "scheduleName": "默认时间计划",
        "scheduleUuid": "4178cc7d6fd3449a93fd053b7fc4a065",
        "type": 134,
        "typeList": [
          1,
          3,
          4
        ],
        "uuid": "0ac1be52b9f54fe4b7c7fa6b9c7f3392"
      },
      {
        "deviceList": [
          {
            "deviceUuid": "eb4c9ed7d25348fba14ab429832d818b",
            "name": "网络摄像机-IPC-人脸识别"
          }
        ],
        "groupList": [
          {
            "name": "员工默认组",
            "uuid": "976055aae31445c5a583ebc1a53c7514"
          },
          {
            "name": "访客默认组",
            "uuid": "90a41fddf76a4daa9dae47762ecfd359"
          },
          {
            "name": "重点人员默认组",
            "uuid": "127d6dadbf1c4e3f93be34865ad0d135"
          }
        ],
        "name": "员工访客重点人员规则布控1",
        "scheduleName": "默认时间计划",
        "scheduleUuid": "4178cc7d6fd3449a93fd053b7fc4a065",
        "type": 134,
        "typeList": [
          1,
          3,
          4
        ]
      }
    ]
  }
}

```

```
    ],  
    "uuid": "3c00767684e142d2b7d08268a7533ec9"  
  }  
],  
  "pageNum": 1,  
  "pageSize": 20,  
  "total": 2  
},  
"msg": "成功"  
}
```

- 1. 人员识别记录
 - 1.1. 员工、访客、重点人员&全局陌生人列表
 - 1.1.1. 基本信息
 - 1.1.2. 请求参数
 - 1.1.3. 返回数据
 - 1.2. 员工、访客、重点人员&全局陌生人详情
 - 1.2.1. 基本信息
 - 1.2.2. 请求参数
 - 1.2.3. 返回数据

1. 人员识别记录

1.1. 员工、访客、重点人员&全局陌生人列表

1.1.1. 基本信息

Path: /v1/api/person/record/list

Method: POST

接口描述: 根据人员类型获取识别记录

1.1.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页/数字类型 默认第一页
personTag	integer	必须	人员类型 1:员工 2:访客 3:重点人员 4:全局陌生人 5:通道陌生人
pageSize	integer	非必须	每页条数/数字类型 默认10条
personName	string	非必须	人员姓名 限制长度40位
deviceUuids	string []	非必须	设备UUID集合
└		非必须	
startTime	integer	必须	开始时间/毫秒
endTime	integer	必须	结束时间/毫秒

1.1.3. 返回数据

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码
pageSize	integer	非必须	分页大小
total	integer	非必须	总数
list	object []	非必须	
└─idUuid	string	非必须	记录主键UUID
└─personUuid	string	非必须	人员UUID
└─deviceUuid	string	非必须	设备UUID
└─captureImageUrl	string	非必须	抓拍图
└─groupImageUrl	string	非必须	底库图
└─personName	string	非必须	人员姓名
└─deviceName	string	非必须	设备名称
└─captureTime	integer	非必须	抓拍时间
└─location	string	非必须	设备位置
└─score	number	非必须	识别分数
└─personTag	integer	非必须	人员类型 1:员工 2: 访客 3:重点人员 4: 全局陌生人 5: 通道陌生人

HTTP请求示例

```
{
  "personTag":1,
  "deviceUuids":[
    "11",
    "22",
    "33"
  ],
  "startTime":1622523871000,
  "endTime":1622624671000
}
```

HTTP响应示例

响应200

```
{
  "code":0,
  "data":{
    "list":[
      {
        "captureImageUrl":"https://IP/pub/_ZzEwMF82bQ==_cd6e3ee8727d44e2a690ef",
        "captureTime":1624591499749,
        "deviceId":"243454566464",
        "deviceName":"人脸识别-B3R-安防",
        "groupImageUrl":"https://IP/pub/_ZzEwMF9mb3JldmVyQnVja2V0_4897c592565f",
        "idUuid":"840b8c3e902c4a1eb540a74f3979c64f",
        "location":"N6",
        "personUuid":"111111",
        "personName":"谋某",
        "personTag":1,
        "score":75.64
      }
    ]
  }
}
```

1.2. 员工、访客、重点人员&全局陌生人详情

1.2.1. 基本信息

Path: /v1/api/person/record/detail

Method: POST

接口描述: 获取人员识别记录详情

1.2.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
idUuid	string	必须	记录主键UUID

1.2.3. 返回数据

名称	类型	是否必须	备注
deviceUuid	string	非必须	设备UUID
personUuid	string	非必须	人员UUID
captureImageUrl	string	非必须	抓拍图
groupImageUrl	string	非必须	底库图
fullImageUrl	string	非必须	全景图
personTag	integer	非必须	人员类型 1:员工 2: 访客 3:重点人员 4: 全局陌生人 5: 通道陌生人
score	number	非必须	比对分数
deviceName	string	非必须	设备名称
captureTime	integer	非必须	识别时间
location	string	非必须	设备位置
personName	string	非必须	人员姓名
ext	string	非必须	备注
groupList	object []	非必须	人员所属组列表
└groupUuid	string	非必须	组uuid
└groupName	string	非必须	组名

名称	类型	是否必须	备注
phone	string	非必须	手机号
identifyNum	string	非必须	身份证号
email	string	非必须	邮箱
uniqueIdentify	string	非必须	唯一标志

HTTP请求示例

```
{
  "idUuid": "2222222222222222",
}
```

HTTP响应示例

响应200

```
{
  "code": 0,
  "data": {
    "captureImageUrl": "https://IP/pub/_ZzEwMF82bQ==_33c608d8a0bb4323a4dd3873092dfc",
    "captureTime": 1624591416353,
    "deviceUuid": "122334444",
    "deviceName": "人脸识别-B3R-安防",
    "fullImageUrl": "",
    "groupImageUrl": "https://IP/pub/_ZzEwMF9mb3JldmVyQnVja2V0_4897c592565f48aaa6f2",
    "groupList": [
      {
        "groupUuid": "976055aae31445c5a583ebc1a53c7514",
        "groupName": "员工默认组"
      }
    ],
    "location": "N6",
    "personUuid": "2222222222222222",
    "personName": "某某",
    "personTag": 1,
    "score": 73.94
  },
  "msg": "成功"
}
```

- 1. 人员轨迹
 - 1.1. 人员轨迹查询
 - 1.1.1. 基本信息
 - 1.1.2. 请求参数
 - 1.1.3. 返回数据

1. 人员轨迹

1.1. 人员轨迹查询

1.1.1. 基本信息

Path: /v1/api/person/track/query

Method: POST

接口描述: 获取实名人员或者陌生人轨迹，陌生人轨迹需先调用1：N接口指定陌生人库搜索得到档案ID

1.1.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
pageNum	integer	非必须	非必须 当前页/数字类型 默认第一页
personUuid	string	非必须	人员UUID 注:【不能和profileUuid同时存在 二者只能选择入参一个】
pageSize	integer	非必须	非必须 每页条数/数字类型 默认10条
profileUuid	string	非必须	档案UUID 注:【不能和personUuid同时存在 二者只能选择入参一个,且在盘古Lite版本不支持此字段,请勿传值】
number	string	非必须	非必须 身份证号码
deviceUids	string []	非必须	非必须 设备UUID集合
└		非必须	
startTime	integer	必须	必须 开始时间/毫秒
endTime	integer	必须	必须 结束时间/毫秒

1.1.3. 返回数据

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码
pageSize	integer	非必须	分页大小
total	integer	非必须	记录总数
list	object []	非必须	
└─deviceUuid	string	非必须	设备UUID
└─personUuid	string	非必须	人员UUID
└─captureImageUrl	string	非必须	抓拍图
└─captureTime	integer	非必须	抓拍时间
└─location	string	非必须	设备位置
└─age	integer	非必须	人员年龄
└─gender	integer	非必须	人员性别 0:男 1: 女
└─personName	string	非必须	人员姓名
└─personTag	integer	非必须	人员类型 1:员工 2: 访客 3:重点人员 4.全局陌生人
└─deviceName	string	非必须	设备名称

HTTP请求示例

```
{
  "personUuid": "1111",
  "deviceUuids": [
    "11",
    "22",
    "33"
  ],
  "startTime": 1622523871000,
  "endTime": 1622624671000
}
```

HTTP响应示例

响应200

```
{
  "code": 0,
  "data": {
    "list": [
      {
        "captureImageUrl": "https://IP/pub/_ZzEwMF82bQ==_44f0c0d3dde846c691eedf",
        "captureTime": 1624566275672,
        "deviceUuid": "2222222",
        "deviceName": "人脸识别",
        "location": "N6",
        "personName": "某某某",
        "personTag": 1
      }
    ]
  }
}
```

- 1. 人员聚类
 - 1.1. 人员聚类任务添加
 - 1.1.1. 基本信息
 - 1.1.2. 请求参数
 - 1.1.3. 返回数据
 - 1.1.4. HTTP请求示例
 - 1.1.5. HTTP响应示例
 - 1.2. 人员聚类任务状态查询
 - 1.2.1. 基本信息
 - 1.2.2. 请求参数
 - 1.2.3. 返回数据
 - 1.2.4. HTTP请求示例
 - 1.2.5. HTTP响应示例
 - 1.3. 人员聚类任务详情查看
 - 1.3.1. 基本信息
 - 1.3.2. 请求参数
 - 1.3.3. 返回数据
 - 1.3.4. HTTP请求示例
 - 1.3.5. HTTP响应示例
 - 1.4. 人员聚类任务列表查看
 - 1.4.1. 基本信息
 - 1.4.2. 请求参数
 - 1.4.3. 返回数据
 - 1.4.4. HTTP请求示例
 - 1.4.5. HTTP响应示例
 - 1.5. 人员聚类人员详情查看
 - 1.5.1. 基本信息
 - 1.5.2. 请求参数
 - 1.5.3. 返回数据
 - 1.5.4. HTTP请求示例
 - 1.5.5. HTTP响应示例
 - 1.6. 人员聚类任务删除
 - 1.6.1. 基本信息
 - 1.6.2. 请求参数
 - 1.6.3. 返回数据
 - 1.6.4. HTTP请求示例
 - 1.6.5. HTTP响应示例

1. 人员聚类

1.1. 人员聚类任务添加

1.1.1. 基本信息

Path: /v1/api/cluster/task/add

Method: POST

接口描述： 新增人员聚类任务

1.1.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
taskName	string	必须	聚类规则名称，长度1-50位
deviceUids	string []	必须	设备uuid集合。必填项，不能为空
timespan	integer	必须	选择时间，范围0~90天
frequency	integer	必须	出现次数，范围2~100次

1.1.3. 返回数据

名称	类型	是否必须	备注
taskUuid	string	非必须	任务uuid

1.1.4. HTTP请求示例

请求path

```
/v1/api/cluster/task/add
```

请求body

```
{
  "taskName": "验证聚类任务",
  "deviceUids": [
    "eb4c9ed7d25348fba14ab429832d818b"
  ],
  "timespan": 2,
  "frequency": 2
}
```

1.1.5. HTTP响应示例

响应 200

```
{
  "code": 0,
  "data": {
    "taskUuid": "57070c5a72124ba281ad90535de072c1"
  },
  "msg": "成功"
}
```

1.2. 人员聚类任务状态查询

1.2.1. 基本信息

Path: /v1/api/cluster/task/status

Method: POST

接口描述: 查询指定的人员聚类任务的状态

1.2.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
taskUuid	string	必须	任务uuid

1.2.3. 返回数据

名称	类型	是否必须	备注
statusType	integer	非必须	任务状态类型, 0-处理中;1-已完成;2-已删除;
statusDesc	string	非必须	任务状态

1.2.4. HTTP请求示例

请求path

```
/v1/api/cluster/task/status
```

请求body


```
{
  "taskUuid": "cd7e6d3825294ddeb3a3dbb3ff2b43d3"
}
```

1.2.5. HTTP响应示例

响应 200

```
{
  "code": 0,
  "data": {
    "statusDesc": "处理中",
    "statusType": 0
  },
  "msg": "成功"
}
```

1.3. 人员聚类任务详情查看

1.3.1. 基本信息

Path: /v1/api/cluster/task/detail/query

Method: POST

接口描述: 查询某个聚类任务的详情

1.3.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
taskUuid	string	必须	任务id
pageNum	integer	非必须	当前页, 默认值1
pageSize	integer	非必须	每页条数, 默认值10
orderReq	object	非必须	
└─filed	string	非必须	排序的filed字段名称 列表返回的字段列如count(出现次数)、latestTime(最近出现时间)等, 默认count
└─order	string	非必须	排序的类型 asc:升序 desc:降序 。默认desc 降序

1.3.3. 返回数据

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码
pageSize	integer	非必须	分页大小
total	integer	非必须	总数
list	object []	非必须	
└─profileId	string	非必须	抓拍轨迹id
└─url	string	非必须	抓拍图
└─frequency	integer	非必须	出现次数
└─latestTime	integer	非必须	最近出现时间, 时间戳, 单位ms

1.3.4. HTTP请求示例

请求path

```
/v1/api/cluster/task/detail/query
```

请求body

```
{
  "taskUuid": "cd7e6d3825294ddeb3a3dbb3ff2b43d3",
  "orderReq": {
    "order": "desc",
    "filed": "count"
  },
  "pageNum": 1,
  "pageSize": 10
}
```

1.3.5. HTTP响应示例

响应 200

```
{
  "code": 0,
  "data": {
    "list": [
      {
        "frequency": 153,
        "latestTime": 1624346093000,
        "profileId": "6974051166623629329",
        "url": "http://10.122.101.7/v5/resources/data?uri=weed://86,22594a3c69"
      },
      {
        "frequency": 142,
        "latestTime": 1624346104000,
        "profileId": "6974051166623629337",
        "url": "http://10.122.101.7/v5/resources/data?uri=weed://86,19fd763091"
      }
    ],
    "pageNum": 1,
    "pageSize": 10,
    "total": 2
  },
  "msg": "成功"
}
```

1.4. 人员聚类任务列表查看

1.4.1. 基本信息

Path: /v1/api/cluster/task/list

Method: POST

接口描述: 分页查看人员聚类任务列表，也可通过名称模糊匹配

1.4.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
name	string	非必须	任务名称, 做模糊匹配
pageNum	integer	非必须	当前页, 默认值1
pageSize	integer	非必须	每页条数, 默认值10

1.4.3. 返回数据

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码
pageSize	integer	非必须	分页大小
total	integer	非必须	总数
list	object []	非必须	
├-uuid	string	非必须	任务uuid
├-name	string	非必须	任务名称
├-frequency	integer	非必须	出现次数
├-deviceInfos	object []	非必须	设备信息
├-deviceUuid	integer	非必须	设备uuid
├-deviceName	string	非必须	设备名称
├-timespan	integer	非必须	时间范围, 1-90表示近多少天
├-createTime	integer	非必须	任务创建时间
├-statusType	integer	非必须	任务状态类型, 0-处理中;1-已完成;2-已删除;
├-statusDesc	string	非必须	任务状态

1.4.4. HTTP请求示例

请求path

```
/v1/api/cluster/task/list
```

请求body

```
{
  "name": "",
  "pageNum": 1,
  "pageSize": 10
}
```

1.4.5. HTTP响应示例

响应 200

```
{
  "code": 0,
  "data": {
    "list": [
      {
        "createdTime": 1622538582000,
        "deviceInfos": [
          {
            "deviceName": "网络摄像机-IPC-人脸识别"
          }
        ],
        "frequency": 2,
        "name": "U3diWcbXZj",
        "statusDesc": "处理中",
        "statusType": 0,
        "timespan": 2,
        "uuid": "3975bc2d4fc04ce29cb8e75bef38fff6"
      },
      {
        "createdTime": 1622457149000,
        "deviceInfos": [
          {
            "deviceName": "智能摄像头-MegEye-C3S-123-人脸抓拍"
          },
          {
            "deviceName": "智能摄像头-MegEye-C4H-242-人脸抓拍"
          },
          {
            "deviceName": "网络摄像机-MegEye-C4R-222-人脸识别"
          }
        ],
        "frequency": 2,
        "name": "122333",
        "statusDesc": "已完成",
        "statusType": 1,
        "timespan": 3,
        "uuid": "a18f45fe9c2d465dbdb4539b22ad84f3"
      }
    ],
    "pageNum": 1,
    "pageSize": 10,
    "total": 2
  },
  "msg": "成功"
}
```

1.5. 人员聚类人员详情查看

1.5.1. 基本信息

Path: /v1/api/cluster/person/detail/query

Method: POST

接口描述: 查询指定聚类任务结果中某个人员的详情

1.5.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
profileId	string	必须	档案id
taskId	string	必须	任务uuid
deviceUids	string []	非必须	设备uuid集合
startTime	integer	非必须	起始时间-时间戳(ms)
endTime	integer	非必须	结束时间-时间戳(ms)
pageNum	integer	非必须	当前页, 默认值1
pageSize	integer	非必须	每页条数, 默认值10

1.5.3. 返回数据

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码
pageSize	integer	非必须	分页大小
total	integer	非必须	总数
list	object []	非必须	
├profileId	string	非必须	人员档案id
├url	string	非必须	抓拍图
├deviceName	string	非必须	设备名称
├deviceLocation	string	非必须	设备位置
├timestamp	integer	非必须	出现时间

1.5.4. HTTP请求示例

请求path

```
/v1/api/cluster/person/detail/query
```

请求body

```
{
  "profileId": "6974051166623629329",
  "taskUuid": "efb21ab7b0bf4396aabe315acd967d10",
  "pageSize": 10,
  "pageNum": 1,
  "endTime": 1624377600000,
  "startTime": 1621785600000,
  "deviceUids": [
    "68599084226b48f29e9126bb8adcf4f7",
    "d909d84343a64dc5936c458edaf3e8cf"
  ]
}
```

1.5.5. HTTP响应示例

响应 200

```

{
  "code": 0,
  "data": {
    "list": [
      {
        "deviceLocation": "ss",
        "deviceName": "core安防",
        "profileId": "6974051166623629329",
        "timestamp": 1624291126000,
        "url": "https://10.122.101.7/v5/resources/data?uri=weed://328,2256a9e6"
      },
      {
        "deviceLocation": "ss",
        "deviceName": "core安防",
        "profileId": "6974051166623629329",
        "timestamp": 1624291028000,
        "url": "https://10.122.101.7/v5/resources/data?uri=weed://328,22545da9"
      },
      {
        "deviceLocation": "ss",
        "deviceName": "core安防",
        "profileId": "6974051166623629329",
        "timestamp": 1624290930000,
        "url": "https://10.122.101.7/v5/resources/data?uri=weed://328,2251be16"
      },
      {
        "deviceLocation": "ss",
        "deviceName": "core安防",
        "profileId": "6974051166623629329",
        "timestamp": 1624290832000,
        "url": "https://10.122.101.7/v5/resources/data?uri=weed://328,224fbdd1"
      },
      {
        "deviceLocation": "ss",
        "deviceName": "core安防",
        "profileId": "6974051166623629329",
        "timestamp": 1624290734000,
        "url": "https://10.122.101.7/v5/resources/data?uri=weed://328,224d6597"
      },
      {
        "deviceLocation": "ss",
        "deviceName": "core安防",
        "profileId": "6974051166623629329",
        "timestamp": 1624290636000,
        "url": "https://10.122.101.7/v5/resources/data?uri=weed://328,224ab57b"
      },
      {
        "deviceLocation": "ss",
        "deviceName": "core安防",
        "profileId": "6974051166623629329",
        "timestamp": 1624290538000,
        "url": "https://10.122.101.7/v5/resources/data?uri=weed://328,2248e516"
      },
      {
        "deviceLocation": "ss",
        "deviceName": "core安防",
        "profileId": "6974051166623629329",
        "timestamp": 1624290440000,
        "url": "https://10.122.101.7/v5/resources/data?uri=weed://328,224686ea"
      },
      {
        "deviceLocation": "ss",
        "deviceName": "core安防",
        "profileId": "6974051166623629329",
        "timestamp": 1624290342000,

```



```
        "url": "https://10.122.101.7/v5/resources/data?uri=weed://328,2243d64f",
      },
      {
        "deviceLocation": "ss",
        "deviceName": "core安防",
        "profileId": "6974051166623629329",
        "timestamp": 1624290244000,
        "url": "https://10.122.101.7/v5/resources/data?uri=weed://328,2241cf61",
      }
    ],
    "pageNum": 1,
    "pageSize": 10,
    "total": 153
  },
  "msg": "成功"
}
```

1.6. 人员聚类任务删除

1.6.1. 基本信息

Path: /v1/api/cluster/task/delete

Method: POST

接口描述: 删除指定的聚类任务

1.6.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
taskId	string	必须	任务uuid

1.6.3. 返回数据

```
OK
```

1.6.4. HTTP请求示例

请求path

```
/v1/api/cluster/task/delete
```

请求body

```
{  
  "taskUuid": "682c729639c148ff83e2e6d11cca0972"  
}
```

1.6.5. HTTP响应示例

响应 200

```
{  
  "code": 0,  
  "msg": "成功"  
}
```

- 1. 人脸比对
 - 1.1. 1:1查询
 - 1.1.1. 基本信息
 - 1.1.2. 请求参数
 - 1.1.3. 返回数据
 - 1.2. 1: N列表查询
 - 1.2.1. 基本信息
 - 1.2.2. 请求参数
 - 1.2.3. 返回数据
 - 1.3. 1: N列表详情
 - 1.3.1. 基本信息
 - 1.3.2. 请求参数
 - 1.3.3. 返回数据

1. 人脸比对

1.1. 1:1查询

1.1.1. 基本信息

Path: /v1/api/photo/searchOneToOnePhoto

Method: POST

接口描述: 上传两张人脸图片进行人脸相似度比分

1.1.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
image1	string	必须	1、支持图片1 Base64编码 2、支持efs地址，使用该接口上传图片获取efs地址：/v1/api/person/uploadImage 3、必须同时使用Base64或者esf类型对比，不可以混合使用
image2	string	必须	1、支持图片2 Base64编码 2、支持efs地址，使用该接口上传图片获取efs地址：/v1/api/person/uploadImage 3、必须同时使用Base64或者esf类型对比，不可以混合使用
imageType	integer	必须	类型：1 efs地址，2 Base64编码 默认1

1.1.3. 返回数据

名称	类型	是否必须	备注
score	number	非必须	比较分数

HTTP请求示例

1、base64请求实例

```
{
  "image1": "图片base64处理字符串",
  "image2": "图片base64处理字符串",
  "imageType": 2
}
```

2、EFS图片请求实例

URI获取 调用下面【上传人员图片，不卡质量】接口 /v1/api/person/uploadImage

```
{
  "image1": "_ZzEwMF82bQ==_8125493cca594b55999767dec7a12640",
  "image2": "_ZzEwMF82bQ==_8125493cca594b55999767dec7a12640",
  "imageType": 1
}
```

HTTP响应示例

响应200

```
{
  "code": 0,
  "data": {
    "score": 99.98237
  },
  "msg": "成功"
}
```

1.2. 1： N列表查询

1.2.1. 基本信息

Path: /v1/api/photo/search

Method: POST

接口描述: 上传图片进行比分获取底库最相似人

1.2.2. 请求参数

Headers

名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
uri	string	必须	图片寻址uri，通过【上传人员图片接口】获取uri
groupTypes	integer []	必须	底库来源 1-员工; 2-访客; 3-重点人员; 4-陌生人(备注： 盘古Lite不支持4-陌生人);
└─		非必须	
threshold	integer	必须	阈值大小, 数值为正整数，阈值区间(0, 100)
pageNum	integer	非必须	当前页, 默认值1
pageSize	integer	非必须	每页条数, 默认值0

1.2.3. 返回数据

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码
pageSize	integer	非必须	分页大小
total	integer	非必须	总数
list	object []	非必须	
└─url	string	非必须	底库图
└─personUuid	string	非必须	人员uuid
└─profileId	string	非必须	档案id
└─name	string	非必须	姓名
└─groupType	integer	非必须	人员身份类型， 1-员工;2-访客;3-黑名单;4-陌生人(备注：盘古Lite不支持4-陌生人)
└─groupTypeDesc	string	非必须	人员身份
└─score	number	非必须	比对分数
└─count	integer	非必须	返回详情数量

HTTP请求示例

```
{
  "uri": "_ZzEwMF9mb3JldmVyQnVja2V0_1e381bd5932846519d0c605a99d65752",
  "groupTypes": [
    3
  ],
  "threshold": 70,
  "pageNum": 1,
  "pageSize": 10
}
```

HTTP响应示例

响应200

```
{
  "code":0,
  "data":{
    "list":[
      {
        "count":8,
        "groupType":3,
        "groupTypeDesc":"黑名单",
        "name":"备用黑名单1",
        "personUuid":"224eb049f12a47e58f518f5ae34b143d",
        "score":99.98075,
        "url":"http://10.171.4.130/pub/_ZzEwMF9mb3JldmVyQnVja2V0_aed08f2ac8854
      }
    ],
    "pageNum":1,
    "pageSize":10,
    "total":1
  },
  "msg":"成功"
}
```

1.3. 1： N列表详情

1.3.1. 基本信息

Path: /v1/api/photo/search/query/detail

Method: POST

接口描述: 获取比对之后的详细

1.3.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
pageNum	integer	非必须	非必须 当前页/数字类型 默认第一页
personUuid	string	非必须	人员Uuid, 和档案ID其中一个必填
pageSize	integer	非必须	非必须 每页条数/数字类型 默认10条
profileId	string	非必须	档案ID, 和人员Uuid其中一个必填, 备注: 盘古Lite不支持此字段, 请勿传值!
deviceUuids	string []	非必须	设备uuids
└		非必须	
startTime	integer	必须	开始时间-时间戳(ms)
endTime	integer	必须	结束时间-时间戳(ms)

1.3.3. 返回数据

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码
pageSize	integer	非必须	分页大小
total	integer	非必须	总数
list	object []	非必须	
└captureImageUrl	string	非必须	抓拍图
└captureTime	integer	非必须	抓拍时间
└deviceName	string	非必须	设备名称
└location	string	非必须	设备位置
└fsType	string	非必须	表示存储来源, 不需要关注

HTTP请求示例


```
{
  "personUuid": "2d33d80d880a4d2a865b315aeb8f555c",
  "deviceUuids": [

  ],
  "pageNum": 1,
  "pageSize": 100,
  "startTime": 1622523854000,
  "endTime": 1622624654000
}
```

HTTP响应示例

响应200

```
{
  "code": 0,
  "data": {
    "list": [
      {
        "captureImageUrl": "http://10.171.4.130/pub/_ZzEwMF82bQ==_b04be96fb9104",
        "captureTime": 1622538507731,
        "deviceName": "网络摄像机-MegEye-C4R-322-人脸识别",
        "location": "OpenAPI测试"
      }
    ],
    "pageNum": 1,
    "pageSize": 100,
    "total": 1
  },
  "msg": "成功"
}
```

- 1. 车辆基础信息
 - 1.1. 车辆基础信息列表
 - 1.1.1. 基本信息
 - 1.1.2. 请求参数
 - 1.1.3. 返回数据
 - 1.2. 车辆基础信息批量删除
 - 1.2.1. 基本信息
 - 1.2.2. 请求参数
 - 1.2.3. 返回数据
 - 1.3. 车辆基础信息批量新增
 - 1.3.1. 基本信息
 - 1.3.2. 请求参数
 - 1.3.3. 返回数据
 - 1.4. 车辆基础信息更新
 - 1.4.1. 基本信息
 - 1.4.2. 请求参数
 - 1.4.3. 返回数据
 - 1.5. 车辆基础详情信息
 - 1.5.1. 基本信息
 - 1.5.2. 请求参数
 - 1.5.3. 返回数据

1. 车辆基础信息

1.1. 车辆基础信息列表

1.1.1. 基本信息

Path: /v1/api/vehicle/list

Method: POST

接口描述: 根据查询条件分页查询基础车辆信息列表。

1.1.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
plateNumber	string	非必须	车辆号码
plateType	integer	非必须	车牌类型枚举 (1:标准民用车与军用车; 2:006车; 3:警车; 4:军用车双行尾牌; 5:使馆车; 6:农用车; 7:摩托车; 8:新能源车)
vehicleType	integer	非必须	车辆类型枚举 (1:小型车; 2:大型车; 3:摩托车; 4:其他车)
personUuld	string	非必须	车主唯一uuld
pageNum	integer	非必须	查询页码,默认第1页
pageSize	integer	非必须	页面大小,默认每页大小20

1.1.3. 返回数据

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码
pageSize	integer	非必须	分页大小
total	integer	非必须	总数
list	object []	非必须	
└─vehicleUuld	string	非必须	车辆唯一uuld
└─plateNumber	string	非必须	车辆号码
└─plateType	integer	非必须	车牌类型枚举 (1:标准民用与军用车; 2:006车; 3:警车; 4:军用车双行尾牌; 5:使馆车; 6:农用车; 7:摩托车; 8:新能源车)
└─plateTypeMsg	string	非必须	车牌类型信息
└─plateColor	string	非必须	车牌颜色
└─vehicleType	integer	非必须	车辆类型枚举 (1:小型车; 2:大型车; 3:摩托车; 4:其他车)
└─vehicleTypeMsg	string	非必须	车辆类型信息
└─vehicleColor	string	非必须	车辆颜色
└─describe	string	非必须	描述
└─personUuld	string	非必须	车主唯一uuld
└─personName	string	非必须	车主姓名

名称	类型	是否必须	备注
 personTelephone	string	非必须	手机号码
 uniqueIdentify	string	非必须	唯一标识
 personType	integer	非必须	车主类型
 personTypeMsg	string	非必须	车主类型描述

HTTP请求示例

请求**body**

```
{
  "pageNum":1,
  "pageSize":20
}
```

HTTP响应示例

响应**200**

```

{
  "code":0,
  "data":{
    "list":[
      {
        "personName":"备用员工1",
        "personType":1,
        "personTypeMsg":"员工",
        "personUuid":"226a3eaa92bd483fb26f25b0cb18686b",
        "plateColor":"白色",
        "plateNumber":"鄂U622VC",
        "plateType":1,
        "plateTypeMsg":"标准民用车与军用车",
        "vehicleUuid":"462137a00b3d4cffbfff7b3d80f2a48ed"
      },
      {
        "personName":"备用访客1",
        "personType":2,
        "personTypeMsg":"访客",
        "personUuid":"186848c9155546209eabe0f4a5f68c87",
        "plateColor":"白色",
        "plateNumber":"陕N629V0",
        "plateType":1,
        "plateTypeMsg":"标准民用车与军用车",
        "vehicleType":1,
        "vehicleTypeMsg":"小型车",
        "vehicleUuid":"46744675f28240c0ac8864327f850cca"
      },
      {
        "describe":"test",
        "personName":"备用黑名单1",
        "personType":3,
        "personTypeMsg":"重点人员",
        "personUuid":"5f57928c8cea4617918e0f1022b04ca6",
        "plateColor":"白色",
        "plateNumber":"辽FC3C10",
        "plateType":1,
        "plateTypeMsg":"标准民用车与军用车",
        "vehicleUuid":"37d9fd76433645068080634abe9bf65d"
      }
    ],
    "pageNum":1,
    "pageSize":20,
    "total":3
  },
  "msg":"成功"
}

```

1.2. 车辆基础信息批量删除

1.2.1. 基本信息

Path: /v1/api/vehicle/batchDelete

Method: POST

接口描述: 根据车辆唯一uuid集合，批量删除基础车辆信息。

1.2.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
vehicleUuldList	string []	必须	车辆唯一uuid集合

1.2.3. 返回数据

名称	类型	是否必须	备注
successVehicles	object []	非必须	批量删除车辆成功返回体
└─vehicleUuld	string	非必须	车辆唯一uuid
failVehicles	object []	非必须	批量删除车辆失败返回体
└─vehicleUuld	string	非必须	车辆唯一uuid
└─msg	string	非必须	异常原因
└─code	integer	非必须	异常码

HTTP请求示例

请求body

```
{
  "vehicleUuIdList":[
    "90504145f6f94860909c2457106c05b3",
    "a2815c2f57154d0fbc6cfeac1f4d965b",
    "69815069a32840368b06635038e4a7de"
  ]
}
```

HTTP响应示例

响应200

```
{
  "code":0,
  "data":{
    "failVehicles":[

    ],
    "successVehicles":[
      {
        "vehicleUuId":"90504145f6f94860909c2457106c05b3"
      },
      {
        "vehicleUuId":"a2815c2f57154d0fbc6cfeac1f4d965b"
      },
      {
        "vehicleUuId":"69815069a32840368b06635038e4a7de"
      }
    ]
  },
  "msg":"成功"
}
```

1.3. 车辆基础信息批量新增

1.3.1. 基本信息

Path: /v1/api/vehicle/batchAdd

Method: POST

接口描述: 基础车辆新增接口，必须输入车牌号码。

1.3.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
addVehicleBeans	object []	必须	车辆批量新增请求体
└─plateNumber	string	必须	车辆号码,车牌号码必须在1-16个字符内且只能包含数字、中英文
└─plateType	integer	非必须	车牌类型枚举 (1:标准民用车与军用车; 2:006车; 3:警车; 4:军用车双行尾牌; 5:使馆车; 6:农用车; 7:摩托车; 8:新能源车)
└─plateColor	string	非必须	车牌颜色,长度必须在0-8个字符且仅支持汉字
└─vehicleType	integer	非必须	车辆类型枚举 (1:小型车; 2:大型车; 3:摩托车; 4:其他车)
└─vehicleColor	string	非必须	车辆颜色,长度必须在0-8个字符且仅支持汉字
└─describe	string	非必须	描述,车辆信息描述长度必须在0-255个中英文字符
└─personUuld	string	非必须	车主唯一uuld
└─personTelephone	string	非必须	手机号码,仅支持6-18位数字
└─uniqueIdentify	string	非必须	用户唯一标识,最大16位数字

1.3.3. 返回数据

名称	类型	是否必须	备注
successVehicles	object []	非必须	批量新增车辆成功返回体
└─vehicleUuld	string	非必须	车辆新增返回唯一uuid
└─plateNumber	string	非必须	车辆号码
└─plateType	integer	非必须	车牌类型,1-标准民用车与军用车、2-006车、3-警车、4-军用车双行尾牌、5-使馆车、6-农用车、7-摩托车、8-新能源车
└─plateColor	string	非必须	车牌颜色
└─vehicleType	integer	非必须	车辆类型,1-小型车、2-大型车、3-摩托车、4-其他车
└─vehicleColor	string	非必须	车辆颜色
└─describe	string	非必须	描述
└─personUuld	string	非必须	车主唯一uuid
└─personTelephone	string	非必须	手机号码
└─uniqueIdentify	string	非必须	用户指定唯一标识
failVehicles	object []	非必须	批量新增车辆失败返回体
└─vehicleUuld	string	非必须	车辆新增返回唯一uuid
└─plateNumber	string	非必须	车辆号码
└─plateType	integer	非必须	车牌类型,1-标准民用车与军用车、2-006车、3-警车、4-军用车双行尾牌、5-使馆车、6-农用车、7-摩托车、8-新能源车

名称	类型	是否必须	备注
plateColor	string	非必须	车牌颜色
vehicleType	integer	非必须	车辆类型,1-小型车、2-大型车、3-摩托车、4-其他车
vehicleColor	string	非必须	车辆颜色
describe	string	非必须	描述
personUuld	string	非必须	车主唯一uuld
personTelephone	string	非必须	手机号码
uniqueIdentify	string	非必须	用户指定唯一标识
msg	string	非必须	异常原因
code	integer	非必须	异常码

HTTP请求示例

请求body

```
{
  "addVehicleBeans":[
    {
      "plateNumber":"湘Q93202"
    },
    {
      "plateNumber":"琼C27425"
    }
  ]
}
```

HTTP响应示例

响应200

```
{
  "code":0,
  "data":{
    "failVehicles":[

    ],
    "successVehicles":[
      {
        "plateNumber":"湘Q93202",
        "vehicleUuid":"8808d955359d46a8a48459fcd57680d2"
      },
      {
        "plateNumber":"琼C27425",
        "vehicleUuid":"b4b46b71bd0f4e5097d534a3a07bf4fc"
      }
    ]
  },
  "msg":"成功"
}
```

1.4. 车辆基础信息更新

1.4.1. 基本信息

Path: /v1/api/vehicle/update

Method: POST

接口描述: 根据车辆唯一uuid或者车牌号更新基础车辆信息

1.4.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
vehicleUuld	string	必须	车辆更新唯一uuid
plateNumber	string	必须	车辆号码,车牌号码必须在1-16个字符内且只能包含数字、中英文
plateType	integer	非必须	车牌类型枚举 (1:标准民用车与军用车; 2:006车; 3:警车; 4:军用车双行尾牌; 5:使馆车; 6:农用车; 7:摩托车; 8:新能源车)
plateColor	string	非必须	车牌颜色,长度必须在0-8个字符且仅支持汉字
vehicleType	integer	非必须	车辆类型枚举 (1:小型车; 2:大型车; 3:摩托车; 4:其他车)
vehicleColor	string	非必须	车辆颜色,车辆颜色长度必须在0-8个字符且仅支持汉字
describe	string	非必须	描述,车辆信息描述长度必须在0-255个中英文字符
personUuld	string	非必须	车主唯一uuld
personTelephone	string	非必须	手机号码,仅支持6-18位数字
uniqueIdentify	string	非必须	用户指定唯一标识,最大16位数字

1.4.3. 返回数据

OK

HTTP请求示例

请求body

```
{
  "addVehicleBeans":[
    {
      "plateNumber":"湘Q93202"
    },
    {
      "plateNumber":"琼C27425"
    }
  ]
}
```

HTTP响应示例

响应200

```
{
  "code":0,
  "data":{
    "failVehicles":[

    ],
    "successVehicles":[
      {
        "plateNumber":"湘Q93202",
        "vehicleUuId":"8808d955359d46a8a48459fcd57680d2"
      },
      {
        "plateNumber":"琼C27425",
        "vehicleUuId":"b4b46b71bd0f4e5097d534a3a07bf4fc"
      }
    ]
  },
  "msg":"成功"
}
```

1.5. 车辆基础详情信息

1.5.1. 基本信息

Path: /v1/api/vehicle/query

Method: POST

接口描述: 根据车辆唯一uuid查询车辆详细信息

1.5.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

1.5.3. 返回数据

名称	类型	是否必须	备注
vehicleUuld	string	非必须	车辆唯一uuld
plateNumber	string	非必须	车辆号码
plateType	integer	非必须	车牌类型枚举 (1:标准民用车与军用车; 2:006车; 3:警车; 4:军用车双行尾牌; 5:使馆车; 6:农用车; 7:摩托车; 8:新能源车)
plateTypeMsg	string	非必须	车牌类型信息
plateColor	string	非必须	车牌颜色
vehicleType	integer	非必须	车辆类型枚举 (1:小型车; 2:大型车; 3:摩托车; 4:其他车)
vehicleTypeMsg	string	非必须	车辆类型信息
vehicleColor	string	非必须	车辆颜色
describe	string	非必须	描述
personUuld	string	非必须	车主唯一uuld
personName	string	非必须	车主姓名
personTelephone	string	非必须	手机号码
uniqueIdentify	string	非必须	唯一标识
personType	integer	非必须	车主类型
personTypeMsg	string	非必须	车主类型描述

HTTP请求示例

请求body

```
{
  "vehicleUuid": "41e5b09d4928432db3f5e5ae5e764a40"
}
```

HTTP响应示例

响应200

```
{
  "code": 0,
  "data": {
    "personName": "备用员工1",
    "personType": 1,
    "personTypeMsg": "员工",
    "personUuid": "ee71f8b5442e4ea9a1eebf9d0acf8b8f",
    "plateColor": "白色",
    "plateNumber": "鄂U622VC",
    "plateType": 1,
    "plateTypeMsg": "标准民用车与军用车",
    "vehicleUuid": "41e5b09d4928432db3f5e5ae5e764a40"
  },
  "msg": "成功"
}
```


- 1. 结构化、车辆管控布控
 - 1.1. 结构化、车辆管控布控列表
 - 1.1.1. 基本信息
 - 1.1.2. 请求参数
 - 1.1.3. 返回数据
 - 1.2. 结构化、车辆管控布控删除
 - 1.2.1. 基本信息
 - 1.2.2. 请求参数
 - 1.2.3. 返回数据
 - 1.3. 结构化、车辆管控布控新增
 - 1.3.1. 基本信息
 - 1.3.2. 请求参数
 - 1.3.3. 返回数据
 - 1.4. 结构化、车辆管控布控更新
 - 1.4.1. 基本信息
 - 1.4.2. 请求参数
 - 1.4.3. 返回数据
 - 1.5. 结构化、车辆管控布控详情
 - 1.5.1. 基本信息
 - 1.5.2. 请求参数
 - 1.5.3. 返回数据

1. 结构化、车辆管控布控

1.1. 结构化、车辆管控布控列表

1.1.1. 基本信息

Path: /v1/api/vehicle/rule/list

Method: POST

接口描述: 根据查询条件分页查询结构化或车辆列表

1.1.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
ruleName	string	非必须	规则名称
deviceUuldList	string []	非必须	设备uuld集合
ruleType	integer	必须	布控类型枚举 (1:结构化布控; 2:车辆布控)
pageNum	integer	非必须	查询页码,默认第一页
pageSize	integer	非必须	页面大小,默认每页大小20

1.1.3. 返回数据

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码
pageSize	integer	非必须	分页大小
total	integer	非必须	总数
list	object []	非必须	
├-structureRuleUuld	string	非必须	规则uuld
├-ruleName	string	非必须	规则名称
├-deviceBeanListBean	object []	非必须	设备列表信息集合
├-deviceUuld	string	非必须	设备uuld
├-deviceName	string	非必须	设备名称
├-timePlanUuld	string	非必须	时间计划uuld
├-timePlanName	string	非必须	时间计划名称

HTTP请求示例

请求body

```
{
  "ruleType":1,
  "pageNum":1,
  "pageSize":15
}
```

HTTP响应示例

响应200

```
{
  "code":0,
  "data":{
    "list":[
      {
        "deviceBeanListBean":[
          {
            "deviceName":"网络摄像机-IPC-结构化",
            "deviceUuid":"14a3de958d87428dbca4df670c35167a"
          }
        ],
        "ruleName":"结构化布控备用01",
        "structureRuleUuid":"72a16e585c26451ca9f3d3b5d1734b2d",
        "timePlanName":"默认时间计划",
        "timePlanUuid":"4178cc7d6fd3449a93fd053b7fc4a065"
      }
    ],
    "pageNum":1,
    "pageSize":15,
    "total":1
  },
  "msg":"成功"
}
```

1.2. 结构化、车辆管控布控删除

1.2.1. 基本信息

Path: /v1/api/vehicle/rule/delete

Method: POST

接口描述: 根据规则唯一uuid删除结构化、车辆布控信息

1.2.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

1.2.3. 返回数据

OK

HTTP请求示例

请求body

```
{
  "structureRuleUuid":"c3b3139f8ea3404c96088a56fb78e421"
}
```

HTTP响应示例

响应200

```
{
  "code":0,
  "msg":"成功"
}
```

1.3. 结构化、车辆管控布控新增

1.3.1. 基本信息

Path: /v1/api/vehicle/rule/add

Method: POST

接口描述: 新增结构化车辆信息。

1.3.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
ruleName	string	必须	规则名称,最大50个中文/字母/数字
deviceUuldList	string []	必须	设备id列表信息集合
timePlanUuld	string	必须	时间计划uuld
ruleType	integer	必须	布控类型枚举 (1:结构化布控; 2: 车辆布控)

1.3.3. 返回数据

HTTP请求示例

请求body

```
{
  "ruleName": "结构化布控",
  "deviceUuIdList": [
    "14a3de958d87428dbca4df670c35167a"
  ],
  "timePlanUuId": "4178cc7d6fd3449a93fd053b7fc4a065",
  "ruleType": 1
}
```

HTTP响应示例

响应200

```
{
  "code": 0,
  "data": {
    "bindFailedDeviceNames": "",
    "structureRuleUuId": "93f4182665e643e89986afef5b432cc7"
  },
  "msg": "成功"
}
```

1.4. 结构化、车辆管控布控更新

1.4.1. 基本信息

Path: /v1/api/vehicle/rule/update

Method: POST

接口描述: 更新结构化、车辆布控信息

1.4.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
structureRuleUuld	string	必须	规则唯一uuld
ruleName	string	必须	规则名称,最大50个中文/字母/数字
deviceUuldList	string []	必须	设备uuld列表信息集合
timePlanUuld	string	必须	时间计划uuld
ruleType	integer	必须	布控类型枚举 (1:结构化布控; 2:车辆布控)

1.4.3. 返回数据

OK

HTTP请求示例

请求body

```
{
  "structureRuleUuid": "f56cacfe2fb94830b27f96977016ea7a",
  "ruleName": "update车辆布控",
  "deviceUuidList": [
    "c0a2388fcc444abbaa1b2b7c77a7d5d4"
  ],
  "timePlanUuid": "5513afe19244443fa9a10caf49c50a1e",
  "ruleType": 2
}
```

HTTP响应示例

响应200

```
{
  "code": 0,
  "msg": "成功"
}
```

1.5. 结构化、车辆管控布控详情

1.5.1. 基本信息

Path: /v1/api/vehicle/rule/query

Method: POST

接口描述: 根据布控规则查询结构化、车辆布控详细信息

1.5.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

1.5.3. 返回数据

名称	类型	是否必须	备注
structureRuleUuld	string	非必须	车辆&结构新增化布控唯一uuld
deviceBeanListBean	object []	非必须	设备列表信息集合
└─deviceUuld	string	非必须	设备uuld
└─deviceName	string	非必须	设备名称
ruleName	string	非必须	规则名称
timePlanUuld	string	非必须	时间计划唯一uuld
timePlanName	string	非必须	时间计划名称

HTTP请求示例

请求body

```
{
  "structureRuleUuId": "7d8a324f89454bbe9545eb901a390758"
}
```

HTTP响应示例

响应200

```
{
  "code": 0,
  "data": {
    "deviceBeanListBean": [
      {
        "deviceName": "网络摄像机-MegEye-C4R-322-结构化",
        "deviceUuId": "c0a2388fcc444abbaa1b2b7c77a7d5d4"
      }
    ],
    "ruleName": "车辆布控备用01",
    "structureRuleUuId": "7d8a324f89454bbe9545eb901a390758",
    "timePlanName": "默认时间计划",
    "timePlanUuId": "4178cc7d6fd3449a93fd053b7fc4a065"
  },
  "msg": "成功"
}
```

- 1. 车辆管控记录
 - 1.1. 车辆识别记录列表
 - 1.1.1. 基本信息
 - 1.1.2. 请求参数
 - 1.1.3. 返回数据
 - 1.2. 车辆识别记录详情
 - 1.2.1. 基本信息
 - 1.2.2. 请求参数
 - 1.2.3. 返回数据

1. 车辆管控记录

1.1. 车辆识别记录列表

1.1.1. 基本信息

Path: /v1/api/vehicle/record/list

Method: POST

接口描述: 车辆布控之后, 产生的带分页的车辆识别记录列表

1.1.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
plateNumber	string	非必须	车牌号码
vehicleType	integer	非必须	车辆类型枚举 (0:未知; 1:两厢轿车; 2:轿车; 3:轿跑; 4:小型轿车; 5:微型轿车; 6:MPV; 7:SUV; 8:大型客车; 9:中型客车; 10:面包车; 11:微型面包车; 12:货车; 13:中型货车; 14:油罐车; 15:吊车; 16:渣土车; 17:小货车; 18:皮卡; 19:微卡; 20:两轮车; 21:三轮车; 22:行人; 29:挂车; 30:小型车(大范围); 31,102:大型车(大范围))
vehicleColor	integer	非必须	车身颜色枚举 (0:白色; 1:红色; 2:黄色; 5:绿色; 7:紫色; 8:粉色; 9:黑色; 10:橙色; 11:银色; 12:香槟色; 13:灰色; 15:蓝色; 21:青色; 29:棕色; 32:深蓝; 40:彩色; 99:未知)
plateColor	integer	非必须	车牌颜色枚举 (0:未知; 1:蓝色; 2:黄色; 3:白色; 4:黑色; 5:绿色; 6:渐变绿底黑字; 7:黄绿双拼底黑字)

名称	类型	是否必须	备注
vehicleBrand	integer	非必须	车辆品牌枚举 (0:其它; 1:大众; 2:奥迪; 3:丰田; 4:本田; 5:奔驰; 6:雪佛兰; 7:奇瑞; 8:别克; 9:长城; 10:现代; 11:尼桑; 12:福特; 13:宝马; 14:雪铁龙; 15:起亚; 16:铃木; 17:马自达; 18:比亚迪; 20:标致; 21:长安; 22:雷克萨斯; 23:中华; 24:斯柯达; 25:海马; 26:夏利; 27:五菱; 28:东风; 29:哈飞; 30:一汽; 31:宝骏; 32:帝豪(吉利); 33:MG名爵; 34:东南; 35:皇冠(丰田); 36:金杯; 37:三菱; 38:荣威; 39:吉利; 40:英伦(吉利); 41:吉利全球鹰(吉利); 42:哈飞赛豹; 43:长丰; 44:北汽威旺; 45:北京汽车; 46:新凯; 47:吉奥汽车; 48:玛莎拉蒂; 49:罗孚; 50:奥斯汀; 52:道奇; 55:捷豹; 57:阿尔法; 58:兰博基尼; 59:布加迪; 60:林肯; 61:法拉利汽车; 62:昌河; 63:菲亚特; 64:福田; 65:讴歌; 66:莲花汽车; 67:华普汽车; 68:红旗; 69:瑞麟; 70:一汽奔腾(一汽); 71:威麟汽车; 72:众泰; 73:力帆; 74:JEEP(吉普); 75:中兴; 76:开瑞; 77:路虎; 78:迈巴赫; 79:雷诺; 80:欧宝; 81:野马; 82:吉普; 83:依维柯; 84:英菲尼迪; 85:斯巴鲁; 86:阿斯顿·马丁; 87:安凯客车; 88:保时捷; 89:宾利; 90:福迪; 91:福建奔驰; 92:吉姆斯; 93:观致; 94:广汽; 95:双龙; 96:海格; 97:悍马; 98:华泰; 99:黄海; 100:九龙客车; 101:理念; 102:奔驰SMART; 103:陆风; 104:纳智捷; 105:欧朗; 106:启辰; 107:哈弗HAVAL; 108:豪沃; 109:上海汇众(上汽集团); 110:金龙; 112:江淮; 113:JMC(江铃); 114:凯迪拉克; 116:宝马MINI; 117:陕汽; 118:少林; 119:沃尔沃; 120:五十铃(庆铃); 121:跃进; 122:宇通; 123:中通; 125:申沃; 126:扬子江; 128:北奔重卡; 139:红岩汽车(上汽依维柯红岩); 130:劳斯莱斯; 131:特斯拉; 132:腾势; 134:英致; 135:西雅特; 136:亚星客车; 137:大宇客车; 138:长安谛艾仕; 139:凯翼; 140:上汽大通; 141:本田思铭; 142:凯马)
vehicleFunctional	integer	非必须	功能用车枚举 (0:未知; 1:挖掘机; 2:渣土车; 3:校车; 4:救护车; 5:警车; 6:集装箱车)
specialOperation	integer	非必须	专项作业枚举 (0:未知; 7:混凝土搅拌机; 8:消防车)

名称	类型	是否必须	备注
personName	string	非必须	车主姓名
personType	integer	非必须	车主身份,1-员工, 2-访客, 3-重点人员
vehicleIdentityType	integer	非必须	车辆身份枚举 (1:重点车辆; 2:员工车辆; 3:访客车辆; 4:陌生车辆; 5:已知车辆)
deviceUuldList	string []	非必须	设备uuid集合
startTime	integer	必须	开始时间,毫秒时间戳
endTime	integer	必须	结束时间,毫秒时间戳
pageNum	integer	非必须	查询页码,默认第一页
pageSize	integer	非必须	页面大小,默认每页大小20

1.1.3. 返回数据

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码
pageSize	integer	非必须	分页大小
total	integer	非必须	总数
list	object []	非必须	
└─vehicleRecordUuld	string	非必须	记录唯一uuid
└─cropUrl	string	非必须	抓拍图
└─plateNumber	string	非必须	车牌号码
└─personName	string	非必须	车主姓名
└─personType	integer	非必须	车主身份,1-员工, 2-访客, 3-重点人员
└─personTypeMsg	string	非必须	车主身份
└─vehicleIdentityType	integer	非必须	车辆身份枚举 (1:重点车辆; 2:员工车辆; 3:访客车辆; 4:陌生车辆; 5:已知车辆)
└─vehicleIdentityMsg	string	非必须	车辆身份信息
└─deviceUuld	string	非必须	设备uuld
└─deviceName	string	非必须	设备名称
└─deviceLocation	string	非必须	设备位置

名称	类型	是否必须	备注
recognitionTime	integer	非必须	识别时间,毫秒时间戳
vehicleUuld	string	非必须	基础车辆信息uuld

HTTP请求示例

请求body

```
{
  "plateNumber": "冀FC3C10",
  "startTime": 1622523871000,
  "endTime": 1622624671000
}
```

HTTP响应示例

响应200

```
{
  "code": 0,
  "data": {
    "list": [
      {
        "cropUrl": "http://10.171.4.130/pub/_ZzEwMF8zbQ==_6fad71e532764dc6b3237",
        "deviceLocation": "OpenAPI测试",
        "deviceName": "网络摄像机-MegEye-C4R-322-结构化",
        "deviceUuId": "c0a2388fcc444abbaa1b2b7c77a7d5d4",
        "personName": "备用员工1",
        "personType": 1,
        "personTypeMsg": "员工",
        "plateNumber": "冀FC3C10",
        "recognitionTime": 1622539645125,
        "vehicleIdentityMsg": "员工车辆",
        "vehicleIdentityType": 2,
        "vehicleRecordUuId": "upbmxnkBn31txtxEXg-B",
        "vehicleUuId": "6f4ad260ab2d4d708aecdfe53a70a99"
      }
    ],
    "pageNum": 1,
    "pageSize": 20,
    "total": 1
  },
  "msg": "成功"
}
```

1.2. 车辆识别记录详情

1.2.1. 基本信息

Path: /v1/api/vehicle/record/query

Method: POST

接口描述: 根据车辆记录记录唯一请求uuid，查询车辆识别详细记录信息

1.2.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

1.2.3. 返回数据

名称	类型	是否必须	备注
vehicleRecordUuld	string	非必须	记录唯一uuid
cropUrl	string	非必须	抓拍图
fullUrl	string	非必须	全景图
plateNumber	string	非必须	车牌号码
vehicleType	integer	非必须	车辆类型枚举 (0:未知; 1:两厢轿车; 2:轿车; 3:轿跑; 4:小型轿车; 5:微型轿车; 6:MPV; 7:SUV; 8:大型客车; 9:中型客车; 10:面包车; 11:微型面包车; 12:货车; 13:中型货车; 14:油罐车; 15:吊车; 16:渣土车; 17:小货车; 18:皮卡; 19:微卡; 20:两轮车; 21:三轮车; 22:行人; 29:挂车; 30:小型车(大范围); 31,102:大型车(大范围))
vehicleTypeMsg	string	非必须	车辆类型信息
vehicleColor	integer	非必须	车身颜色枚举 (0:白色; 1:红色; 2:黄色; 5:绿色; 7:紫色; 8:粉色; 9:黑色; 10:橙色; 11:银色; 12:香槟色; 13:灰色; 15:蓝色; 21:青色; 29:棕色; 32:深蓝; 40:彩色; 99:未知)
vehicleColorMsg	string	非必须	车身颜色信息
plateColor	integer	非必须	车牌颜色枚举 (0:未知; 1:蓝色; 2:黄色; 3:白色; 4:黑色; 5:绿色; 6:渐变绿底黑字; 7:黄绿双拼底黑字)
plateColorMsg	string	非必须	车牌颜色信息

名称	类型	是否必须	备注
vehicleBrand	integer	非必须	车辆品牌枚举 (0:其它; 1:大众; 2:奥迪; 3:丰田; 4:本田; 5:奔驰; 6:雪佛兰; 7:奇瑞; 8:别克; 9:长城; 10:现代; 11:尼桑; 12:福特; 13:宝马; 14:雪铁龙; 15:起亚; 16:铃木; 17:马自达; 18:比亚迪; 20:标致; 21:长安; 22:雷克萨斯; 23:中华; 24:斯柯达; 25:海马; 26:夏利; 27:五菱; 28:东风; 29:哈飞; 30:一汽; 31:宝骏; 32:帝豪(吉利); 33:MG名爵; 34:东南; 35:皇冠(丰田); 36:金杯; 37:三菱; 38:荣威; 39:吉利; 40:英伦(吉利); 41:吉利全球鹰(吉利); 42:哈飞赛豹; 43:长丰; 44:北汽威旺; 45:北京汽车; 46:新凯; 47:吉奥汽车; 48:玛莎拉蒂; 49:罗孚; 50:奥斯汀; 52:道奇; 55:捷豹; 57:阿尔法; 58:兰博基尼; 59:布加迪; 60:林肯; 61:法拉利汽车; 62:昌河; 63:菲亚特; 64:福田; 65:讴歌; 66:莲花汽车; 67:华普汽车; 68:红旗; 69:瑞麟; 70:一汽奔腾(一汽); 71:威麟汽车; 72:众泰; 73:力帆; 74:JEEP(吉普); 75:中兴; 76:开瑞; 77:路虎; 78:迈巴赫; 79:雷诺; 80:欧宝; 81:野马; 82:吉普; 83:依维柯; 84:英菲尼迪; 85:斯巴鲁; 86:阿斯顿·马丁; 87:安凯客车; 88:保时捷; 89:宾利; 90:福迪; 91:福建奔驰; 92:吉姆士; 93:观致; 94:广汽; 95:双龙; 96:海格; 97:悍马; 98:华泰; 99:黄海; 100:九龙客车; 101:理念; 102:奔驰SMART; 103:陆风; 104:纳智捷; 105:欧朗; 106:启辰; 107:哈弗HAVAL; 108:豪沃; 109:上海汇众(上汽集团); 110:金龙; 112:江淮; 113:JMC(江铃); 114:凯迪拉克; 116:宝马MINI; 117:陕汽; 118:少林; 119:沃尔沃; 120:五十铃(庆铃); 121:跃进; 122:宇通; 123:中通; 125:申沃; 126:扬子江; 128:北奔重卡; 139:红岩汽车(上汽依维柯红岩); 130:劳斯莱斯; 131:特斯拉; 132:腾势; 134:英致; 135:西雅特; 136:亚星客车; 137:大宇客车; 138:长安谛艾仕; 139:凯翼; 140:上汽大通; 141:本田思铭; 142:凯马)
vehicleBrandMsg	string	非必须	车辆品牌信息

名称	类型	是否必须	备注
vehicleFunctional	integer	非必须	功能用车枚举 (0:未知; 1:挖掘机; 2:渣土车; 3:校车; 4:救护车; 5:警车; 6:集装箱车)
vehicleFunctionalMsg	string	非必须	功能用车信息
specialOperation	integer	非必须	专项作业枚举 (0:未知; 7:混凝土搅拌机; 8:消防车)
specialOperationMsg	string	非必须	专项作业信息
personName	string	非必须	车主姓名
personType	integer	非必须	车主身份,1-员工, 2-访客, 3-重点人员
personTypeMsg	string	非必须	车主身份信息
vehicleIdentityType	integer	非必须	车辆身份枚举 (1:重点车辆; 2:员工车辆; 3:访客车辆; 4:陌生车辆; 5:已知车辆)
vehicleIdentityMsg	string	非必须	车辆身份信息
deviceUuld	string	非必须	设备uuld
deviceName	string	非必须	设备名称
recognitionTime	integer	非必须	识别时间
deviceLocation	string	非必须	设备位置

HTTP请求示例

请求body

```
{
  "vehicleRecordUuid": "upbmxnkBn31txtxEXg-B"
}
```

HTTP响应示例

响应200

```
{
  "code": 0,
  "data": {
    "cropUrl": "http://10.171.4.130/pub/_ZzEwMF8zbQ==_6fad71e532764dc6b32376b4c6873",
    "deviceLocation": "OpenAPI测试",
    "deviceName": "网络摄像机-MegEye-C4R-322-结构化",
    "deviceUuid": "c0a2388fcc444abbaa1b2b7c77a7d5d4",
    "fullUrl": "http://10.171.4.130/pub/_ZzEwMF8zbQ==_7191969c97ad4d69b1ef82df753d2",
    "personName": "备用员工1",
    "personType": 1,
    "personTypeMsg": "员工",
    "plateColor": 1,
    "plateColorMsg": "蓝色",
    "plateNumber": "冀FC3C10",
    "recognitionTime": 1622539645125,
    "specialOperation": 0,
    "specialOperationMsg": "未知",
    "vehicleBrand": 31,
    "vehicleBrandMsg": "宝骏",
    "vehicleColor": 0,
    "vehicleColorMsg": "白色",
    "vehicleFunctional": 0,
    "vehicleFunctionalMsg": "未知",
    "vehicleIdentityMsg": "员工车辆",
    "vehicleIdentityType": 2,
    "vehicleRecordUuid": "upbmxnkBn31txtxEXg-B",
    "vehicleType": 7,
    "vehicleTypeMsg": "SUV"
  },
  "msg": "成功"
}
```

- 1. 车辆轨迹
 - 1.1. 车辆轨迹查询
 - 1.1.1. 基本信息
 - 1.1.2. 请求参数
 - 1.1.3. 返回数据

1. 车辆轨迹

1.1. 车辆轨迹查询

1.1.1. 基本信息

Path: /v1/api/vehicle/record/track/query

Method: POST

接口描述: 记录车辆出现的轨迹

1.1.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
plateNumber	string	必须	车辆号码,精确查询
personName	string	非必须	车主姓名,精确查询
deviceUuldList	string []	非必须	设备id集合
startTime	integer	必须	开始时间,毫秒时间戳
endTime	integer	必须	结束时间,毫秒时间戳

1.1.3. 返回数据

名称	类型	是否必须	备注
queryVehicleTrackBeans	object []	非必须	轨迹记录
└─cropUrl	string	非必须	抓拍图
└─plateNumber	string	非必须	车辆号码
└─recognitionTime	integer	非必须	识别时间
└─deviceUuld	string	非必须	设备uuld
└─deviceName	string	非必须	设备名称
└─personName	string	非必须	车主姓名
└─deviceLocation	string	非必须	设备位置
total	integer	非必须	出现次数

HTTP请求示例

请求body

```
{
  "plateNumber": "冀FC3C10",
  "deviceUuIdList": [
    "c0a2388fcc444abbaa1b2b7c77a7d5d4"
  ],
  "startTime": 1622523871000,
  "endTime": 1622624671000
}
```

HTTP响应示例

响应200

```
{
  "code": 0,
  "data": {
    "queryVehicleTrackBeans": [
      {
        "cropUrl": "http://10.171.4.130/pub/_ZzEwMF8zbQ==_6fad71e532764dc6b3237",
        "deviceLocation": "OpenAPI测试",
        "deviceName": "网络摄像机-MegEye-C4R-322-结构化",
        "deviceUuId": "c0a2388fcc444abbaa1b2b7c77a7d5d4",
        "personName": "备用员工1",
        "plateNumber": "冀FC3C10",
        "recognitionTime": 1622539645125
      }
    ],
    "total": 1
  },
  "msg": "成功"
}
```

- 1. 结构化检索
 - 1.1. 人体数据检索
 - 1.1.1. 基本信息
 - 1.1.2. 请求参数
 - 1.1.3. 返回数据
 - 1.2. 人体数据详情查询
 - 1.2.1. 基本信息
 - 1.2.2. 请求参数
 - 1.2.3. 返回数据
 - 1.3. 人脸数据检索
 - 1.3.1. 基本信息
 - 1.3.2. 请求参数
 - 1.3.3. 返回数据
 - 1.4. 人脸数据详情查询
 - 1.4.1. 基本信息
 - 1.4.2. 请求参数
 - 1.4.3. 返回数据
 - 1.5. 机动车数据检索
 - 1.5.1. 基本信息
 - 1.5.2. 请求参数
 - 1.5.3. 返回数据
 - 1.6. 机动车数据详情查询
 - 1.6.1. 基本信息
 - 1.6.2. 请求参数
 - 1.6.3. 返回数据
 - 1.7. 非机动车数据检索
 - 1.7.1. 基本信息
 - 1.7.2. 请求参数
 - 1.7.3. 返回数据
 - 1.8. 非机动车数据详情查询
 - 1.8.1. 基本信息
 - 1.8.2. 请求参数
 - 1.8.3. 返回数据

1. 结构化检索

1.1. 人体数据检索

1.1.1. 基本信息

Path: /v1/api/structure/pedestrian/search

Method: POST

接口描述: 根据查询条件，分页查询人体数据

1.1.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
currPage	integer	非必须	当前页，默认是1
gender	integer	非必须	性别，枚举值有：{0=未知, 1=男性, 2=女性}
age	integer	非必须	年龄，枚举值有：{0=未知, 1=儿童, 2=少年, 3=青年, 4=中年, 5=中老年, 6=老年}
pageSize	integer	非必须	每页条数，默认是10
dressUpperColor	integer	非必须	上衣颜色，枚举值有：{0=未知, 1=白色, 10=青色, 11=紫色, 12=浅蓝色, 13=深蓝色, 14=黑色, 15=彩色, 2=灰色, 3=棕色, 4=红色, 5=蓝色, 6=黄色, 7=绿色, 8=粉色, 9=橙色}
startTime	integer	必须	开始时间，时间戳（毫秒）
dressLowerColor	integer	非必须	下衣颜色，枚举值有：{0=未知, 1=白色, 10=青色, 11=紫色, 12=浅蓝色, 13=深蓝色, 14=黑色, 15=彩色, 2=灰色, 3=棕色, 4=红色, 5=蓝色, 6=黄色, 7=绿色, 8=粉色, 9=橙色}
endTime	integer	必须	结束时间，时间戳（毫秒）
deviceUuldList	string []	非必须	设备uuid
dressUpperStyle	integer	非必须	上衣服饰，枚举值有：{0=未知, 1=长袖, 2=短袖}
dressLowerStyle	integer	非必须	下衣服饰，枚举值有：{0=未知, 1=长裤, 2=短裤, 3=长裙, 4=短裙}
searchImageUrl	string	非必须	被检索图片的URL，被图片可以通过接口/v1/api/misc/file/upload上传，将这个接口返回的uri设置到这个字段即可
wearHat	integer	非必须	是否戴帽子，枚举值有：{0=未知, 1=无帽子, 2=有帽子}

名称	类型	是否必须	备注
hatColor	integer	非必须	帽子颜色，枚举值有：{0=未知, 1=白色, 10=青色, 11=紫色, 12=浅蓝色, 13=深蓝色, 14=黑色, 15=彩色, 2=灰色, 3=棕色, 4=红色, 5=蓝色, 6=黄色, 7=绿色, 8=粉色, 9=橙色}
bagType	integer	非必须	带包类型，枚举值有：{0=未知, 1=未携带包, 2=双肩包, 3=手提包, 4=单肩包, 5=拉杆箱, 6=腰包, 7=有背包, 类型未知}
rideBike	integer	非必须	是否骑车，枚举值有：{0=未知, 1=未骑车, 2=骑车}
wearSafetycap	integer	非必须	是否戴安全帽，枚举值有：{0=未知, 1=无安全帽, 2=有安全帽}
scoreThreshold	number	非必须	相似度，默认是90
fallStatus	integer	非必须	跌倒状态，枚举值有：{0=未知, 1=未跌倒, 2=跌倒}
runStatus	integer	非必须	奔跑状态，枚举值有：{0=未知, 1=未奔跑, 2=奔跑}
smokeStatus	integer	非必须	抽烟状态，枚举值有：{0=未知, 1=未抽烟, 2=抽烟}
watchPhoneStatus	integer	非必须	看手机状态，枚举值有：{0=未知, 1=未看手机, 2=看手机}
usePhoneStatus	integer	非必须	打电话状态，枚举值有：{0=未知, 1=未打电话, 2=打电话}
umbrellaStatus	integer	非必须	打伞状态，枚举值有：{0=未知, 1=不打伞, 2=打伞}

1.1.3. 返回数据

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码
pageSize	integer	非必须	分页大小
total	integer	非必须	总数
list	object []	非必须	
└gender	integer	非必须	性别，枚举值有：{0=未知, 1=男性, 2=女性}
└id	string	非必须	数据的唯一ID
└age	integer	非必须	年龄，枚举值有：{0=未知, 1=儿童, 2=少年, 3=青年, 4=中年, 5=中老年, 6=老年}
└cropUri	string	非必须	抓拍图URL
└dressUpperColor	integer	非必须	上衣颜色，枚举值有：{0=未知, 1=白色, 10=青色, 11=紫色, 12=浅蓝色, 13=深蓝色, 14=黑色, 15=彩色, 2=灰色, 3=棕色, 4=红色, 5=蓝色, 6=黄色, 7=绿色, 8=粉色, 9=橙色}
└fullUri	string	非必须	图片全景图URL
└deviceUuld	string	非必须	设备Uuld
└dressLowerColor	integer	非必须	下衣颜色，枚举值有：{0=未知, 1=白色, 10=青色, 11=紫色, 12=浅蓝色, 13=深蓝色, 14=黑色, 15=彩色, 2=灰色, 3=棕色, 4=红色, 5=蓝色, 6=黄色, 7=绿色, 8=粉色, 9=橙色}
└deviceName	string	非必须	设备名

名称	类型	是否必须	备注
└─dressUpperStyle	integer	非必须	上衣服饰，枚举值有：{0=未知, 1=长袖, 2=短袖}
└─deviceLocation	string	非必须	设备所在位置
└─dressLowerStyle	integer	非必须	下衣服饰，枚举值有：{0=未知, 1=长裤, 2=短裤, 3=长裙, 4=短裙}
└─category	string	非必须	结构化的类别，枚举值有：{1=人脸, 2=人体, 3=机动车, 4=非机动车}
└─wearHat	integer	非必须	是否戴帽子，枚举值有：{0=未知, 1=无帽子, 2=有帽子}
└─hatColor	integer	非必须	帽子颜色，枚举值有：{0=未知, 1=白色, 10=青色, 11=紫色, 12=浅蓝色, 13=深蓝色, 14=黑色, 15=彩色, 2=灰色, 3=棕色, 4=红色, 5=蓝色, 6=黄色, 7=绿色, 8=粉色, 9=橙色}
└─recognitionTime	integer	非必须	识别时间，时间戳（毫秒）
└─bagType	integer	非必须	是否带包，枚举值有：{0=未知, 1=未携带包, 2=双肩包, 3=手提包, 4=单肩包, 5=拉杆箱, 6=腰包, 7=有背包, 类型未知}
└─cropRectLeft	integer	非必须	目标在抓拍图片中的距离，左
└─cropRectTop	integer	非必须	目标在抓拍图片中的距离，上
└─rideBike	integer	非必须	是否骑车，枚举值有：{0=未知, 1=未骑车, 2=骑车}
└─cropRectRight	integer	非必须	目标在抓拍图片中的距离，右

名称	类型	是否必须	备注
└wearSafetycap	integer	非必须	是否戴安全帽，枚举值有：{0=未知, 1=无安全帽, 2=有安全帽}
└cropRectBottom	integer	非必须	目标在抓拍图片中的距离，下
└relatedFaceId	string	非必须	关联的人脸id
└fallStatus	integer	非必须	跌倒状态，枚举值有：{0=未知, 1=未跌倒, 2=跌倒}
└fullRectLeft	integer	非必须	目标在全景图片中的距离，左
└fullRectTop	integer	非必须	目标在全景图片中的距离，上
└runStatus	integer	非必须	奔跑状态，枚举值有：{0=未知, 1=未奔跑, 2=奔跑}
└fullRectRight	integer	非必须	目标在全景图片中的距离，右
└smokeStatus	integer	非必须	抽烟状态，枚举值有：{0=未知, 1=未抽烟, 2=抽烟}
└fullRectBottom	integer	非必须	目标在全景图片中的距离，下
└watchPhoneStatus	integer	非必须	看手机状态，枚举值有：{0=未知, 1=未看手机, 2=看手机}
└score	number	非必须	特征比对时相似对得分
└usePhoneStatus	integer	非必须	打电话状态，枚举值有：{0=未知, 1=未打电话, 2=打电话}
└umbrellaStatus	integer	非必须	打伞状态，枚举值有：{0=未知, 1=不打伞, 2=打伞}

HTTP请求示例

请求body

```
{  
  "currPage":1,  
  "pageSize":10,  
  "startTime":1622523868000,  
  "endTime":1622624668000  
}
```

HTTP响应示例

响应200

```

{
  "code":0,
  "data":{
    "list":[
      {
        "age":4,
        "bagType":1,
        "category":"2",
        "cropRectBottom":170,
        "cropRectLeft":16,
        "cropRectRight":112,
        "cropRectTop":25,
        "cropUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_db85013411a141baab0c1",
        "deviceLocation":"OpenAPI测试",
        "deviceName":"网络摄像机-IPC-结构化",
        "deviceUuId":"14a3de958d87428dbca4df670c35167a",
        "dressLowerColor":14,
        "dressLowerStyle":0,
        "dressUpperColor":13,
        "dressUpperStyle":2,
        "fallStatus":1,
        "fullRectBottom":1072,
        "fullRectLeft":850,
        "fullRectRight":377,
        "fullRectTop":148,
        "fullUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_94f11b1f26d04b6faaef5",
        "gender":1,
        "hatColor":0,
        "id":"tZbPxnkBn31txtxE8g3v",
        "recognitionTime":1622538165558,
        "rideBike":1,
        "runStatus":1,
        "smokeStatus":1,
        "umbrellaStatus":1,
        "usePhoneStatus":1,
        "watchPhoneStatus":1,
        "wearHat":1,
        "wearSafetycap":1
      },
      {
        "age":4,
        "bagType":1,
        "category":"2",
        "cropRectBottom":136,
        "cropRectLeft":12,
        "cropRectRight":81,
        "cropRectTop":20,
        "cropUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_d6369274ea86456b92728",
        "deviceLocation":"OpenAPI测试",
        "deviceName":"网络摄像机-MegEye-C4R-322-结构化",
        "deviceUuId":"c0a2388fcc444abbaa1b2b7c77a7d5d4",
        "dressLowerColor":14,
        "dressLowerStyle":1,
        "dressUpperColor":14,
        "dressUpperStyle":1,
        "fallStatus":1,
        "fullRectBottom":136,
        "fullRectLeft":669,
        "fullRectRight":81,
        "fullRectTop":443,
        "fullUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_a87815fd36c0453291835",
        "gender":1,
        "hatColor":0,
        "id":"n5bPxnkBn31txtxEzw2a",
        "recognitionTime":1622538163635,

```

```

        "rideBike":2,
        "runStatus":1,
        "smokeStatus":1,
        "umbrellaStatus":1,
        "usePhoneStatus":1,
        "watchPhoneStatus":1,
        "wearHat":1,
        "wearSafetycap":1
    },
    {
        "age":4,
        "bagType":1,
        "category":"2",
        "cropRectBottom":256,
        "cropRectLeft":16,
        "cropRectRight":106,
        "cropRectTop":9,
        "cropUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_12be6e38cd2742dbae5d6",
        "deviceLocation":"OpenAPI测试",
        "deviceName":"网络摄像机-IPC-结构化",
        "deviceUuId":"14a3de958d87428dbca4df670c35167a",
        "dressLowerColor":13,
        "dressLowerStyle":1,
        "dressUpperColor":14,
        "dressUpperStyle":2,
        "fallStatus":1,
        "fullRectBottom":869,
        "fullRectLeft":182,
        "fullRectRight":290,
        "fullRectTop":23,
        "fullUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_e73aba50d96c4208986e4",
        "gender":1,
        "hatColor":0,
        "id":"uZbQxnkbn31ttxEKQ1o",
        "recognitionTime":1622538162662,
        "rideBike":1,
        "runStatus":1,
        "smokeStatus":1,
        "umbrellaStatus":1,
        "usePhoneStatus":1,
        "watchPhoneStatus":1,
        "wearHat":1,
        "wearSafetycap":1
    },
    {
        "age":4,
        "bagType":1,
        "category":"2",
        "cropRectBottom":277,
        "cropRectLeft":18,
        "cropRectRight":125,
        "cropRectTop":2,
        "cropUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_4e8929e43e1a4eb8b40aa",
        "deviceLocation":"OpenAPI测试",
        "deviceName":"网络摄像机-IPC-结构化",
        "deviceUuId":"14a3de958d87428dbca4df670c35167a",
        "dressLowerColor":2,
        "dressLowerStyle":1,
        "dressUpperColor":14,
        "dressUpperStyle":2,
        "fallStatus":1,
        "fullRectBottom":824,
        "fullRectLeft":562,
        "fullRectRight":323,
        "fullRectTop":4,
        "fullUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_54f5953e448e4b04b5c7a

```

```

        "gender":1,
        "hatColor":0,
        "id":"tpbPxnkBn31txtxE8g3z",
        "recognitionTime":1622538161196,
        "rideBike":1,
        "runStatus":1,
        "smokeStatus":1,
        "umbrellaStatus":1,
        "usePhoneStatus":1,
        "watchPhoneStatus":1,
        "wearHat":1,
        "wearSafetycap":1
    },
    {
        "age":4,
        "bagType":1,
        "category":"2",
        "cropRectBottom":307,
        "cropRectLeft":14,
        "cropRectRight":97,
        "cropRectTop":0,
        "cropUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_ff73adda1a404f788c6c7",
        "deviceLocation":"OpenAPI测试",
        "deviceName":"网络摄像机-IPC-结构化",
        "deviceUuId":"14a3de958d87428dbca4df670c35167a",
        "dressLowerColor":2,
        "dressLowerStyle":1,
        "dressUpperColor":12,
        "dressUpperStyle":2,
        "fallStatus":1,
        "fullRectBottom":666,
        "fullRectLeft":916,
        "fullRectRight":203,
        "fullRectTop":-27,
        "fullUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_78ac5cc8839441f0bcd1a",
        "gender":1,
        "hatColor":0,
        "id":"05bQxnkBn31txtxEXA0q",
        "recognitionTime":1622538160599,
        "rideBike":1,
        "runStatus":1,
        "smokeStatus":1,
        "umbrellaStatus":1,
        "usePhoneStatus":1,
        "watchPhoneStatus":1,
        "wearHat":1,
        "wearSafetycap":1
    },
    {
        "age":3,
        "bagType":1,
        "category":"2",
        "cropRectBottom":276,
        "cropRectLeft":15,
        "cropRectRight":105,
        "cropRectTop":0,
        "cropUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_3a9bec6347894245a6c9f",
        "deviceLocation":"OpenAPI测试",
        "deviceName":"网络摄像机-IPC-结构化",
        "deviceUuId":"14a3de958d87428dbca4df670c35167a",
        "dressLowerColor":14,
        "dressLowerStyle":1,
        "dressUpperColor":13,
        "dressUpperStyle":2,
        "fallStatus":1,
        "fullRectBottom":584,

```

```

        "fullRectLeft":472,
        "fullRectRight":193,
        "fullRectTop":-80,
        "fullUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_b4a0c0f2249c4687813a5",
        "gender":1,
        "hatColor":0,
        "id":"t5bPxnkBn31txtxE8g34",
        "recognitionTime":1622538159991,
        "rideBike":1,
        "runStatus":1,
        "smokeStatus":1,
        "umbrellaStatus":1,
        "usePhoneStatus":1,
        "watchPhoneStatus":1,
        "wearHat":1,
        "wearSafetycap":1
    },
    {
        "age":4,
        "bagType":1,
        "category":"2",
        "cropRectBottom":301,
        "cropRectLeft":19,
        "cropRectRight":128,
        "cropRectTop":0,
        "cropUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_ceab75c31e0c4e4a85604",
        "deviceLocation":"OpenAPI测试",
        "deviceName":"网络摄像机-IPC-结构化",
        "deviceUuId":"14a3de958d87428dbca4df670c35167a",
        "dressLowerColor":14,
        "dressLowerStyle":1,
        "dressUpperColor":14,
        "dressUpperStyle":2,
        "fallStatus":1,
        "fullRectBottom":764,
        "fullRectLeft":544,
        "fullRectRight":308,
        "fullRectTop":-17,
        "fullUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_1293724fffa74ab5b5bbc",
        "gender":1,
        "hatColor":0,
        "id":"tJbPxnkBn31txtxE8g3q",
        "recognitionTime":1622538156139,
        "rideBike":1,
        "runStatus":1,
        "smokeStatus":1,
        "umbrellaStatus":1,
        "usePhoneStatus":1,
        "watchPhoneStatus":1,
        "wearHat":1,
        "wearSafetycap":1
    },
    {
        "age":4,
        "bagType":1,
        "category":"2",
        "cropRectBottom":73,
        "cropRectLeft":5,
        "cropRectRight":39,
        "cropRectTop":11,
        "cropUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_d9ca4b566a394da9a2fcc",
        "deviceLocation":"OpenAPI测试",
        "deviceName":"网络摄像机-MegEye-C4R-322-结构化",
        "deviceUuId":"c0a2388fcc444abbaa1b2b7c77a7d5d4",
        "dressLowerColor":14,
        "dressLowerStyle":1,

```



```

        "dressUpperColor":14,
        "dressUpperStyle":1,
        "fallStatus":1,
        "fullRectBottom":73,
        "fullRectLeft":694,
        "fullRectRight":39,
        "fullRectTop":350,
        "fullUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_3a83f8648f2a4b53945e9",
        "gender":1,
        "hatColor":0,
        "id":"dZbPxnkBn31txtxEoA24",
        "recognitionTime":1622538155879,
        "rideBike":2,
        "runStatus":1,
        "smokeStatus":1,
        "umbrellaStatus":1,
        "usePhoneStatus":1,
        "watchPhoneStatus":1,
        "wearHat":1,
        "wearSafetycap":1
    },
    {
        "age":4,
        "bagType":1,
        "category":"2",
        "cropRectBottom":286,
        "cropRectLeft":16,
        "cropRectRight":107,
        "cropRectTop":0,
        "cropUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_7218be4bc05a46f098bdf",
        "deviceLocation":"OpenAPI测试",
        "deviceName":"网络摄像机-IPC-结构化",
        "deviceUuId":"14a3de958d87428dbca4df670c35167a",
        "dressLowerColor":14,
        "dressLowerStyle":2,
        "dressUpperColor":7,
        "dressUpperStyle":2,
        "fallStatus":1,
        "fullRectBottom":806,
        "fullRectLeft":925,
        "fullRectRight":269,
        "fullRectTop":-41,
        "fullUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_6a82c1379435441e8a6e8",
        "gender":1,
        "hatColor":0,
        "id":"upbQxnkBn31txtxERA28",
        "recognitionTime":1622538155867,
        "rideBike":1,
        "runStatus":1,
        "smokeStatus":1,
        "umbrellaStatus":1,
        "usePhoneStatus":1,
        "watchPhoneStatus":1,
        "wearHat":1,
        "wearSafetycap":1
    },
    {
        "age":3,
        "bagType":1,
        "category":"2",
        "cropRectBottom":301,
        "cropRectLeft":15,
        "cropRectRight":100,
        "cropRectTop":0,
        "cropUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_17f14e2fcb2f4ac4bb568",
        "deviceLocation":"OpenAPI测试",

```

```
    "deviceName": "网络摄像机-IPC-结构化",
    "deviceUuId": "14a3de958d87428dbca4df670c35167a",
    "dressLowerColor": 2,
    "dressLowerStyle": 1,
    "dressUpperColor": 2,
    "dressUpperStyle": 2,
    "fallStatus": 1,
    "fullRectBottom": 727,
    "fullRectLeft": 842,
    "fullRectRight": 228,
    "fullRectTop": -42,
    "fullUri": "http://10.171.4.130/pub/_ZzEwMF8zbQ==_d290d856512d44aebc985",
    "gender": 1,
    "hatColor": 0,
    "id": "aJbPxnkBn31txtxEnA3M",
    "recognitionTime": 1622538155121,
    "rideBike": 1,
    "runStatus": 1,
    "smokeStatus": 1,
    "umbrellaStatus": 1,
    "usePhoneStatus": 1,
    "watchPhoneStatus": 1,
    "wearHat": 1,
    "wearSafetycap": 1
  }
],
"pageNum": 1,
"pageSize": 10,
"total": 61
},
"msg": "成功"
}
```

1.2. 人体数据详情查询

1.2.1. 基本信息

Path: /v1/api/structure/pedestrian/query

Method: POST

接口描述: 根据数据的唯一id, 查询人数数据详细信息

1.2.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

1.2.3. 返回数据

名称	类型	是否必须	备注
cropUri	string	非必须	抓拍图URL
gender	integer	非必须	性别，枚举值有：{0=未知, 1=男性, 2=女性}
age	integer	非必须	年龄，枚举值有：{0=未知, 1=儿童, 2=少年, 3=青年, 4=中年, 5=中老年, 6=老年}
fullUri	string	非必须	图片全景图URL
deviceUuld	string	非必须	设备Uuld
dressUpperColor	integer	非必须	上衣颜色，枚举值有：{0=未知, 1=白色, 10=青色, 11=紫色, 12=浅蓝色, 13=深蓝色, 14=黑色, 15=彩色, 2=灰色, 3=棕色, 4=红色, 5=蓝色, 6=黄色, 7=绿色, 8=粉色, 9=橙色}
deviceName	string	非必须	设备名
dressLowerColor	integer	非必须	下衣颜色，枚举值有：{0=未知, 1=白色, 10=青色, 11=紫色, 12=浅蓝色, 13=深蓝色, 14=黑色, 15=彩色, 2=灰色, 3=棕色, 4=红色, 5=蓝色, 6=黄色, 7=绿色, 8=粉色, 9=橙色}
deviceLocation	string	非必须	设备所在位置
dressUpperStyle	integer	非必须	上衣服饰，枚举值有：{0=未知, 1=长袖, 2=短袖}
category	string	非必须	结构化的类别，枚举值有：{1=人脸, 2=人体, 3=机动车, 4=非机动车}
dressLowerStyle	integer	非必须	下衣服饰，枚举值有：{0=未知, 1=长裤, 2=短裤, 3=长裙, 4=短裙}
recognitionTime	integer	非必须	识别时间，时间戳（毫秒）
wearHat	integer	非必须	是否戴帽子，枚举值有：{0=未知, 1=无帽子, 2=有帽子}

名称	类型	是否必须	备注
cropRectLeft	integer	非必须	目标在抓拍图片中的距离，左
hatColor	integer	非必须	帽子颜色，枚举值有：{0=未知, 1=白色, 10=青色, 11=紫色, 12=浅蓝色, 13=深蓝色, 14=黑色, 15=彩色, 2=灰色, 3=棕色, 4=红色, 5=蓝色, 6=黄色, 7=绿色, 8=粉色, 9=橙色}
bagType	integer	非必须	是否带包，枚举值有：{0=未知, 1=未携带包, 2=双肩包, 3=手提包, 4=单肩包, 5=拉杆箱, 6=腰包, 7=有背包, 类型未知}
cropRectTop	integer	非必须	目标在抓拍图片中的距离，上
cropRectRight	integer	非必须	目标在抓拍图片中的距离，右
rideBike	integer	非必须	是否骑车，枚举值有：{0=未知, 1=未骑车, 2=骑车}
cropRectBottom	integer	非必须	目标在抓拍图片中的距离，下
wearSafetycap	integer	非必须	是否戴安全帽，枚举值有：{0=未知, 1=无安全帽, 2=有安全帽}
fullRectLeft	integer	非必须	目标在全景图片中的距离，左
relatedFaceld	string	非必须	关联的人脸id
fallStatus	integer	非必须	跌倒状态，枚举值有：{0=未知, 1=未跌倒, 2=跌倒}
fullRectTop	integer	非必须	目标在全景图片中的距离，上
fullRectRight	integer	非必须	目标在全景图片中的距离，右

名称	类型	是否必须	备注
runStatus	integer	非必须	奔跑状态，枚举值有：{0=未知, 1=未奔跑, 2=奔跑}
fullRectBottom	integer	非必须	目标在全景图片中的距离，下
smokeStatus	integer	非必须	抽烟状态，枚举值有：{0=未知, 1=未抽烟, 2=抽烟}
watchPhoneStatus	integer	非必须	看手机状态，枚举值有：{0=未知, 1=未看手机, 2=看手机}
usePhoneStatus	integer	非必须	打电话状态，枚举值有：{0=未知, 1=未打电话, 2=打电话}
umbrellaStatus	integer	非必须	打伞状态，枚举值有：{0=未知, 1=不打伞, 2=打伞}

HTTP请求示例

请求body

```
{
  "id": "tZbPxnkBn31txtxE8g3v"
}
```

HTTP响应示例

响应200

```
{
  "code":0,
  "data":{
    "age":4,
    "bagType":1,
    "category":"2",
    "cropRectBottom":170,
    "cropRectLeft":16,
    "cropRectRight":112,
    "cropRectTop":25,
    "cropUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_db85013411a141baab0c13a9d3e18",
    "deviceLocation":"OpenAPI测试",
    "deviceName":"网络摄像机-IPC-结构化",
    "deviceUuId":"14a3de958d87428dbca4df670c35167a",
    "dressLowerColor":14,
    "dressLowerStyle":0,
    "dressUpperColor":13,
    "dressUpperStyle":2,
    "fallStatus":1,
    "fullRectBottom":1072,
    "fullRectLeft":850,
    "fullRectRight":377,
    "fullRectTop":148,
    "fullUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_94f11b1f26d04b6faaef5844addec",
    "gender":1,
    "hatColor":0,
    "recognitionTime":1622538165558,
    "rideBike":1,
    "runStatus":1,
    "smokeStatus":1,
    "umbrellaStatus":1,
    "usePhoneStatus":1,
    "watchPhoneStatus":1,
    "wearHat":1,
    "wearSafetycap":1
  },
  "msg":"成功"
}
```

1.3. 人脸数据检索

1.3.1. 基本信息

Path: /v1/api/structure/face/search

Method: POST

接口描述: 根据查询条件, 分页查询人脸数据

1.3.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
currPage	integer	非必须	当前页，默认是1
gender	integer	非必须	性别，枚举值有：{0=未知, 1=男性, 2=女性}
age	integer	非必须	年龄，枚举值有：{0=未知, 1=儿童, 2=少年, 3=青年, 4=中年, 5=中老年, 6=老年}
pageSize	integer	非必须	每页条数，默认是10
maskStatus	integer	非必须	口罩状态，枚举值有：{0=未知, 1=无口罩, 2=有口罩}
startTime	integer	必须	开始时间，时间戳（毫秒）
endTime	integer	必须	结束时间，时间戳（毫秒）
glassesStatus	integer	非必须	眼镜状态，枚举值有：{0=未知, 1=无眼镜, 2=有眼镜}
deviceUuldList	string []	非必须	设备uuid
noseOcclusion	integer	非必须	鼻子遮挡，枚举值有：{0=未知, 1=未遮挡, 2=有遮挡}
beardStatus	integer	非必须	胡子状态，枚举值有：{0=未知, 1=无胡子, 2=有胡子}
searchImageUrl	string	非必须	被检索图片的URL，被图片可以通过接口/v1/api/misc/file/upload上传，将这个接口返回的uri设置到这个字段即可
scoreThreshold	number	非必须	相似度，默认是90
hairStatus	integer	非必须	头发状态，枚举值有：{0=未知, 1=长发, 2=短发, 3=少量头发（秃顶）, 4=光头}

1.3.3. 返回数据

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码
pageSize	integer	非必须	分页大小
total	integer	非必须	总数
list	object []	非必须	
└gender	integer	非必须	性别，枚举值有：{0=未知, 1=男性, 2=女性}
└id	string	非必须	数据的唯一ID
└age	integer	非必须	年龄，枚举值有：{0=未知, 1=儿童, 2=少年, 3=青年, 4=中年, 5=中老年, 6=老年}
└cropUri	string	非必须	抓拍图URL
└fullUri	string	非必须	图片全景图URL
└maskStatus	integer	非必须	口罩状态，枚举值有：{0=未知, 1=无口罩, 2=有口罩}
└deviceUuld	string	非必须	设备Uuld
└glassesStatus	integer	非必须	眼镜状态，枚举值有：{0=未知, 1=无眼镜, 2=有眼镜}
└deviceName	string	非必须	设备名
└noseOcclusion	integer	非必须	鼻子遮挡，枚举值有：{0=未知, 1=未遮挡, 2=有遮挡}

名称	类型	是否必须	备注
└beardStatus	integer	非必须	胡子状态，枚举值有：{0=未知, 1=无胡子, 2=有胡子}
└deviceLocation	string	非必须	设备所在位置
└category	string	非必须	结构化的类别，枚举值有：{1=人脸, 2=人体, 3=机动车, 4=非机动车}
└hairStatus	integer	非必须	头发状态，枚举值有：{0=未知, 1=长发, 2=短发, 3=少量头发（秃顶），4=光头}
└recognitionTime	integer	非必须	识别时间，时间戳（毫秒）
└relatedPedestrianId	string	非必须	关联的人体id
└cropRectLeft	integer	非必须	目标在抓拍图片中的距离，左
└cropRectTop	integer	非必须	目标在抓拍图片中的距离，上
└cropRectRight	integer	非必须	目标在抓拍图片中的距离，右
└cropRectBottom	integer	非必须	目标在抓拍图片中的距离，下
└fullRectLeft	integer	非必须	目标在全景图片中的距离，左
└fullRectTop	integer	非必须	目标在全景图片中的距离，上
└fullRectRight	integer	非必须	目标在全景图片中的距离，右
└fullRectBottom	integer	非必须	目标在全景图片中的距离，下
└score	number	非必须	特征比对时相似对得分

HTTP请求示例

请求body

```
{
  "currPage":1,
  "pageSize":10,
  "startTime":1622523868000,
  "endTime":1622624668000
}
```

HTTP响应示例

响应200

```

{
  "code":0,
  "data":{
    "list":[
      {
        "age":4,
        "beardStatus":1,
        "category":"1",
        "cropRectBottom":116,
        "cropRectLeft":63,
        "cropRectRight":127,
        "cropRectTop":58,
        "cropUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_01e3ee0b32b340c4a80a8",
        "deviceLocation":"OpenAPI测试",
        "deviceName":"网络摄像机-IPC-结构化",
        "deviceUuId":"14a3de958d87428dbca4df670c35167a",
        "fullRectBottom":123,
        "fullRectLeft":1017,
        "fullRectRight":135,
        "fullRectTop":387,
        "fullUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_13e13d4399a44c27b7ea3",
        "gender":1,
        "glassesStatus":2,
        "hairStatus":2,
        "id":"HZbhxnkBn31txtxEHQ97",
        "maskStatus":1,
        "noseOcclusion":0,
        "recognitionTime":1622539291498
      },
      {
        "age":4,
        "beardStatus":2,
        "category":"1",
        "cropRectBottom":126,
        "cropRectLeft":63,
        "cropRectRight":126,
        "cropRectTop":63,
        "cropUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_e29afadd27634d4eab474",
        "deviceLocation":"OpenAPI测试",
        "deviceName":"网络摄像机-IPC-结构化",
        "deviceUuId":"14a3de958d87428dbca4df670c35167a",
        "fullRectBottom":126,
        "fullRectLeft":751,
        "fullRectRight":126,
        "fullRectTop":394,
        "fullUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_54c74976d1a84fe88f961",
        "gender":1,
        "glassesStatus":2,
        "hairStatus":2,
        "id":"HpbhxnkBn31txtxEHQ-A",
        "maskStatus":1,
        "noseOcclusion":0,
        "recognitionTime":1622539290557
      },
      {
        "age":3,
        "beardStatus":1,
        "category":"1",
        "cropRectBottom":81,
        "cropRectLeft":48,
        "cropRectRight":97,
        "cropRectTop":40,
        "cropUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_b6bc23a8482c43e098b1f",
        "deviceLocation":"OpenAPI测试",
        "deviceName":"网络摄像机-IPC-结构化",

```

```

        "deviceUuId": "14a3de958d87428dbca4df670c35167a",
        "fullRectBottom": 81,
        "fullRectLeft": 971,
        "fullRectRight": 97,
        "fullRectTop": 218,
        "fullUri": "http://10.171.4.130/pub/_ZzEwMF8zbQ==_e41c9c91a57f4ca28990c",
        "gender": 1,
        "glassesStatus": 1,
        "hairStatus": 2,
        "id": "HJbgxnkbn31ttxEzw8f",
        "maskStatus": 1,
        "noseOcclusion": 0,
        "recognitionTime": 1622539289186
    },
    {
        "age": 4,
        "beardStatus": 1,
        "category": "1",
        "cropRectBottom": 110,
        "cropRectLeft": 64,
        "cropRectRight": 128,
        "cropRectTop": 55,
        "cropUri": "http://10.171.4.130/pub/_ZzEwMF8zbQ==_0d2184586ea5405daa532",
        "deviceLocation": "OpenAPI测试",
        "deviceName": "网络摄像机-IPC-结构化",
        "deviceUuId": "14a3de958d87428dbca4df670c35167a",
        "fullRectBottom": 118,
        "fullRectLeft": 750,
        "fullRectRight": 136,
        "fullRectTop": 404,
        "fullUri": "http://10.171.4.130/pub/_ZzEwMF8zbQ==_82c78b310d3e41a4a049c",
        "gender": 1,
        "glassesStatus": 2,
        "hairStatus": 1,
        "id": "G5bgxnkbn31ttxEyw9m",
        "maskStatus": 1,
        "noseOcclusion": 0,
        "recognitionTime": 1622539289085
    },
    {
        "age": 3,
        "beardStatus": 1,
        "category": "1",
        "cropRectBottom": 81,
        "cropRectLeft": 48,
        "cropRectRight": 97,
        "cropRectTop": 40,
        "cropUri": "http://10.171.4.130/pub/_ZzEwMF8zbQ==_2473bec3340c4936bcb1c",
        "deviceLocation": "OpenAPI测试",
        "deviceName": "网络摄像机-IPC-结构化",
        "deviceUuId": "14a3de958d87428dbca4df670c35167a",
        "fullRectBottom": 81,
        "fullRectLeft": 971,
        "fullRectRight": 97,
        "fullRectTop": 218,
        "fullUri": "http://10.171.4.130/pub/_ZzEwMF8zbQ==_71640a9bfadd44d2b5a59",
        "gender": 1,
        "glassesStatus": 1,
        "hairStatus": 2,
        "id": "1ZbQxnkbn31ttxEYw36",
        "maskStatus": 1,
        "noseOcclusion": 0,
        "recognitionTime": 1622538213192
    },
    {
        "age": 4,

```

```

        "beardStatus":1,
        "category":"1",
        "cropRectBottom":110,
        "cropRectLeft":64,
        "cropRectRight":128,
        "cropRectTop":55,
        "cropUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_aa5c619484884c698febe",
        "deviceLocation":"OpenAPI测试",
        "deviceName":"网络摄像机-IPC-结构化",
        "deviceUuId":"14a3de958d87428dbca4df670c35167a",
        "fullRectBottom":118,
        "fullRectLeft":750,
        "fullRectRight":136,
        "fullRectTop":404,
        "fullUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_d55806c505574bea958f3",
        "gender":1,
        "glassesStatus":2,
        "hairStatus":1,
        "id":"1JbQxnkbn31txtxEYA0V",
        "maskStatus":1,
        "noseOcclusion":0,
        "recognitionTime":1622538213081
    },
    {
        "age":4,
        "beardStatus":1,
        "category":"1",
        "cropRectBottom":67,
        "cropRectLeft":39,
        "cropRectRight":79,
        "cropRectTop":33,
        "cropUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_772364b9e4fb4e1eab9de",
        "deviceLocation":"OpenAPI测试",
        "deviceName":"网络摄像机-IPC-结构化",
        "deviceUuId":"14a3de958d87428dbca4df670c35167a",
        "fullRectBottom":67,
        "fullRectLeft":848,
        "fullRectRight":79,
        "fullRectTop":93,
        "fullUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_ebddf16ddf534a82bdec7",
        "gender":1,
        "glassesStatus":1,
        "hairStatus":2,
        "id":"6JbHxnkbn31txtxEYwtK",
        "maskStatus":1,
        "noseOcclusion":0,
        "recognitionTime":1622537621548
    },
    {
        "age":3,
        "beardStatus":1,
        "category":"1",
        "cropRectBottom":77,
        "cropRectLeft":39,
        "cropRectRight":79,
        "cropRectTop":38,
        "cropUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_cbef739a8b34491c82a78",
        "deviceLocation":"OpenAPI测试",
        "deviceName":"网络摄像机-IPC-结构化",
        "deviceUuId":"14a3de958d87428dbca4df670c35167a",
        "fullRectBottom":77,
        "fullRectLeft":733,
        "fullRectRight":79,
        "fullRectTop":123,
        "fullUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_86bb6cf505834067857c9",
        "gender":1,

```

```

        "glassesStatus":2,
        "hairStatus":1,
        "id":"55bHxnkBn31txtxEXwta",
        "maskStatus":1,
        "noseOcclusion":0,
        "recognitionTime":1622537621003
    },
    {
        "age":4,
        "beardStatus":1,
        "category":"1",
        "cropRectBottom":88,
        "cropRectLeft":48,
        "cropRectRight":97,
        "cropRectTop":44,
        "cropUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_05ae84521ff6417db2bdf",
        "deviceLocation":"OpenAPI测试",
        "deviceName":"网络摄像机-IPC-结构化",
        "deviceUuId":"14a3de958d87428dbca4df670c35167a",
        "fullRectBottom":88,
        "fullRectLeft":660,
        "fullRectRight":97,
        "fullRectTop":159,
        "fullUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_832cc8ebc21a41cba146a",
        "gender":1,
        "glassesStatus":2,
        "hairStatus":2,
        "id":"5ZbHxnkBn31txtxEUwuh",
        "maskStatus":1,
        "noseOcclusion":0,
        "recognitionTime":1622537617672
    },
    {
        "age":4,
        "beardStatus":1,
        "category":"1",
        "cropRectBottom":86,
        "cropRectLeft":48,
        "cropRectRight":97,
        "cropRectTop":43,
        "cropUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_46f6614d743743298ee3f",
        "deviceLocation":"OpenAPI测试",
        "deviceName":"网络摄像机-IPC-结构化",
        "deviceUuId":"14a3de958d87428dbca4df670c35167a",
        "fullRectBottom":86,
        "fullRectLeft":982,
        "fullRectRight":97,
        "fullRectTop":136,
        "fullUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_b0b439c2aa2b427bb0a50",
        "gender":1,
        "glassesStatus":2,
        "hairStatus":2,
        "id":"5JbHxnkBn31txtxETwvB",
        "maskStatus":1,
        "noseOcclusion":0,
        "recognitionTime":1622537617066
    }
},
"pageNum":1,
"pageSize":10,
"total":16
},
"msg":"成功"
}

```

1.4. 人脸数据详情查询

1.4.1. 基本信息

Path: /v1/api/structure/face/query

Method: POST

接口描述: 根据数据的唯一id查询人脸数据详细信息

1.4.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

1.4.3. 返回数据

名称	类型	是否必须	备注
cropUri	string	非必须	抓拍图URL
gender	integer	非必须	性别，枚举值有：{0=未知, 1=男性, 2=女性}
age	integer	非必须	年龄，枚举值有：{0=未知, 1=儿童, 2=少年, 3=青年, 4=中年, 5=中老年, 6=老年}
fullUri	string	非必须	图片全景图URL
deviceUuld	string	非必须	设备Uuld
maskStatus	integer	非必须	口罩状态，枚举值有：{0=未知, 1=无口罩, 2=有口罩}
deviceName	string	非必须	设备名
glassesStatus	integer	非必须	眼镜状态，枚举值有：{0=未知, 1=无眼镜, 2=有眼镜}
beardStatus	integer	非必须	胡子状态，枚举值有：{0=未知, 1=无胡子, 2=有胡子}
deviceLocation	string	非必须	设备所在位置
category	string	非必须	结构化的类别，枚举值有：{1=人脸, 2=人体, 3=机动车, 4=非机动车}
noseOcclusion	integer	非必须	鼻子遮挡，枚举值有：{0=未知, 1=未遮挡, 2=有遮挡}
hairStatus	integer	非必须	头发状态，枚举值有：{0=未知, 1=长发, 2=短发, 3=少量头发(秃顶), 4=光头}
recognitionTime	integer	非必须	识别时间，时间戳（毫秒）
cropRectLeft	integer	非必须	目标在抓拍图片中的距离，左

名称	类型	是否必须	备注
relatedPedestrianId	string	非必须	关联的人体id
cropRectTop	integer	非必须	目标在抓拍图片中的距离，上
cropRectRight	integer	非必须	目标在抓拍图片中的距离，右
cropRectBottom	integer	非必须	目标在抓拍图片中的距离，下
fullRectLeft	integer	非必须	目标在全景图片中的距离，左
fullRectTop	integer	非必须	目标在全景图片中的距离，上
fullRectRight	integer	非必须	目标在全景图片中的距离，右
fullRectBottom	integer	非必须	目标在全景图片中的距离，下

HTTP请求示例

请求body

```
{
  "id": "e5bjxnkBn31txtxE_w-f"
}
```

HTTP响应示例

响应200

```
{
  "code":0,
  "data":{
    "age":4,
    "beardStatus":1,
    "category":"1",
    "cropRectBottom":116,
    "cropRectLeft":63,
    "cropRectRight":127,
    "cropRectTop":58,
    "cropUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_b3758b62730642b1b10fd62886a69",
    "deviceLocation":"OpenAPI测试",
    "deviceName":"网络摄像机-IPC-结构化",
    "deviceUuId":"14a3de958d87428dbca4df670c35167a",
    "fullRectBottom":123,
    "fullRectLeft":1017,
    "fullRectRight":135,
    "fullRectTop":387,
    "fullUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_8b020aee0f0a4fd28690dfa955645",
    "gender":1,
    "glassesStatus":2,
    "hairStatus":2,
    "maskStatus":1,
    "noseOcclusion":0,
    "recognitionTime":1622539480505
  },
  "msg":"成功"
}
```

1.5. 机动车数据检索

1.5.1. 基本信息

Path: /v1/api/structure/motorVehicle/search

Method: POST

接口描述: 根据查询条件，分页查询机动车数据

1.5.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
currPage	integer	非必须	当前页，默认是1
plateNumber	string	非必须	车牌号
pageSize	integer	非必须	每页条数，默认是10
vehicleType	integer	非必须	车辆类型，枚举值有：{0=未知, 1=两厢轿车, 2=轿车, 3=轿跑, 4=小型轿车, 5=微型轿车, 6=MPV, 7=SUV, 8=大型客车, 9=中型客车, 10=面包车, 11=微型面包车, 12=货车, 13=中型货车, 14=油罐车, 15=吊车, 16=渣土车, 17=小货车, 18=皮卡, 19=微卡, 22=行人, 29=挂车, 30=小型车(大范围), 31=大型车(大范围)}
startTime	integer	必须	开始时间，时间戳（毫秒）
vehicleColor	integer	非必须	车身颜色，枚举值有：{0=白色, 1=红色, 2=黄色, 5=绿色, 7=紫色, 8=粉色, 9=黑色, 10=橙色, 11=银色, 12=香槟色, 13=灰色, 15=蓝色, 21=青色, 29=棕色, 32=深蓝, 40=彩色, 99=未知}
endTime	integer	必须	结束时间，时间戳（毫秒）

名称	类型	是否必须	备注
vehicleBrand	integer	非必须	车辆品牌，枚举值有：{0=其它，1=大众，2=奥迪，3=丰田，4=本田，5=奔驰，6=雪佛兰，7=奇瑞，8=别克，9=长城，10=现代，11=尼桑，12=福特，13=宝马，14=雪铁龙，15=起亚，16=铃木，17=马自达，18=比亚迪，20=标致，21=长安，22=雷克萨斯，23=中华，24=斯柯达，25=海马，26=夏利，27=五菱，28=东风，29=哈飞，30=一汽，31=宝骏，32=帝豪(吉利)，33=MG名爵，34=东南，35=皇冠(丰田)，36=金杯，37=三菱，38=荣威，39=吉利，40=英伦(吉利)，41=吉利全球鹰(吉利)，42=哈飞赛豹，43=长丰，44=北汽威旺，45=北京汽车，46=新凯，47=吉奥汽车，48=玛莎拉蒂，49=罗孚，50=奥斯汀，52=道奇，55=捷豹，57=阿尔法，58=兰博基尼，59=布加迪，60=林肯，61=法拉利汽车，62=昌河，63=菲亚特，64=福田，65=讴歌，66=莲花汽车，67=华普汽车，68=红旗，69=瑞麟，70=一汽奔腾(一汽)，71=威麟汽车，72=众泰，73=力帆，74=JEEP(吉普)，75=中兴，76=开瑞，77=路虎，78=迈巴赫，79=雷诺，80=欧宝，81=野马，82=吉普，83=依维柯，84=英菲尼迪，85=斯巴鲁，86=阿斯顿·马丁，87=安凯客车，88=保时捷，89=宾利，90=福迪，91=福建奔驰，92=吉姆斯，93=观致，94=广汽，95=双龙，96=海格，97=悍马，98=华泰，99=黄海，100=九龙客车，101=理念，102=奔驰SMART，103=陆风，104=纳智捷，105=欧朗，106=启辰，107=哈弗HAVAL，108=豪沃，109=上海汇众(上汽集团)，110=金龙，112=江淮，113=JMC(江铃)，114=凯迪拉克，116=宝马MINI，117=陕汽，118=少林，119=沃尔沃，120=五十铃(庆铃)，121=跃进，122=宇通，123=中通，125=申沃，126=扬子江，128=北奔重卡，130=劳斯莱斯，131=特斯拉，132=腾势，134=英致，135=西雅特，136=亚星客车，137=大宇客车，138=长安谛艾仕，139=凯翼，140=上汽大通，141=本田思铭，142=凯马}
deviceUuldList	string []	非必须	设备uuid
vehicleFunctional	integer	非必须	功能用车，枚举值有：{0=未知，1=挖掘机，2=渣土车，3=校车，4=救护车，5=警车，6=集装箱车}

名称	类型	是否必须	备注
searchImageUrl	string	非必须	被检索图片的URL，被图片可以通过接口/v1/api/misc/file/upload上传，将这个接口返回的uri设置到这个字段即可
specialOperation	integer	非必须	专项作业，枚举值有：{0=未知, 7=混凝土搅拌机, 8=消防车}
scoreThreshold	number	非必须	相似度，默认是70
plateColor	integer	非必须	车牌颜色，枚举值有：{0=未知, 1=蓝色, 2=黄色, 3=白色, 4=黑色, 5=绿色, 6=渐变绿底黑字, 7=黄绿双拼底黑字}

1.5.3. 返回数据

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码
pageSize	integer	非必须	分页大小
total	integer	非必须	总数
list	object []	非必须	
└─id	string	非必须	数据的唯一ID
└─plateNumber	string	非必须	车牌号码
└─cropUri	string	非必须	抓拍图URL
└─vehicleType	integer	非必须	车辆类型，枚举值有：{0=未知, 1=两厢轿车, 2=轿车, 3=轿跑, 4=小型轿车, 5=微型轿车, 6=MPV, 7=SUV, 8=大型客车, 9=中型客车, 10=面包车, 11=微型面包车, 12=货车, 13=中型货车, 14=油罐车, 15=吊车, 16=渣土车, 17=小货车, 18=皮卡, 19=微卡, 22=行人, 29=挂车, 30=小型车(大范围), 31=大型车(大范围)}
└─fullUri	string	非必须	图片全景图URL
└─vehicleColor	integer	非必须	车身颜色，枚举值有：{0=白色, 1=红色, 2=黄色, 5=绿色, 7=紫色, 8=粉色, 9=黑色, 10=橙色, 11=银色, 12=香槟色, 13=灰色, 15=蓝色, 21=青色, 29=棕色, 32=深蓝, 40=彩色, 99=未知}
└─deviceUuld	string	非必须	设备Uuld

名称	类型	是否必须	备注
plateColor	integer	非必须	车牌颜色，枚举值有：{0=未知, 1=蓝色, 2=黄色, 3=白色, 4=黑色, 5=绿色, 6=渐变绿底黑字, 7=黄绿双拼底黑字}
deviceName	string	非必须	设备名

名称	类型	是否必须	备注
└─vehicleBrand	integer	非必须	车辆品牌，枚举值有：{0=其它, 1=大众, 2=奥迪, 3=丰田, 4=本田, 5=奔驰, 6=雪佛兰, 7=奇瑞, 8=别克, 9=长城, 10=现代, 11=尼桑, 12=福特, 13=宝马, 14=雪铁龙, 15=起亚, 16=铃木, 17=马自达, 18=比亚迪, 20=标致, 21=长安, 22=雷克萨斯, 23=中华, 24=斯柯达, 25=海马, 26=夏利, 27=五菱, 28=东风, 29=哈飞, 30=一汽, 31=宝骏, 32=帝豪(吉利), 33=MG名爵, 34=东南, 35=皇冠(丰田), 36=金杯, 37=三菱, 38=荣威, 39=吉利, 40=英伦(吉利), 41=吉利全球鹰(吉利), 42=哈飞赛豹, 43=长丰, 44=北汽威旺, 45=北京汽车, 46=新凯, 47=吉奥汽车, 48=玛莎拉蒂, 49=罗孚, 50=奥斯汀, 52=道奇, 55=捷豹, 57=阿尔法, 58=兰博基尼, 59=布加迪, 60=林肯, 61=法拉利汽车, 62=昌河, 63=菲亚特, 64=福田, 65=讴歌, 66=莲花汽车, 67=华普汽车, 68=红旗, 69=瑞麟, 70=一汽奔腾(一汽), 71=威麟汽车, 72=众泰, 73=力帆, 74=JEEP(吉普), 75=中兴, 76=开瑞, 77=路虎, 78=迈巴赫, 79=雷诺, 80=欧宝, 81=野马, 82=吉普, 83=依维柯, 84=英菲尼迪, 85=斯巴鲁, 86=阿斯顿·马丁, 87=安凯客车, 88=保时捷, 89=宾利, 90=福迪, 91=福建奔驰, 92=吉姆斯, 93=观致, 94=广汽, 95=双龙, 96=海格, 97=悍马, 98=华泰, 99=黄海, 100=九龙客车, 101=理念, 102=奔驰SMART, 103=陆风, 104=纳智捷, 105=欧朗, 106=启辰, 107=哈弗HAVAL, 108=豪沃, 109=上海汇众(上汽集团), 110=金龙, 112=江淮, 113=JMC(江铃), 114=凯迪拉克, 116=宝马MINI, 117=陕汽, 118=少林, 119=沃尔沃, 120=五十铃(庆铃), 121=跃进, 122=宇通, 123=中通, 125=申沃, 126=扬子江, 128=北奔重卡, 130=劳斯莱斯, 131=特斯拉, 132=腾势, 134=英致, 135=西雅特, 136=亚星客车, 137=大宇客车, 138=长安谛艾仕, 139=凯翼, 140=上汽大通, 141=本田思铭, 142=凯马}
└─deviceLocation	string	非必须	设备所在位置

名称	类型	是否必须	备注
└─vehicleFunctional	integer	非必须	功能用车，枚举值有：{0=未知, 1=挖掘机, 2=渣土车, 3=校车, 4=救护车, 5=警车, 6=集装箱车}
└─category	string	非必须	结构化的类别，枚举值有：{1=人脸, 2=人体, 3=机动车, 4=非机动车}
└─specialOperation	integer	非必须	专项作业，枚举值有：{0=未知, 7=混凝土搅拌机, 8=消防车}
└─recognitionTime	integer	非必须	识别时间，时间戳（毫秒）
└─cropRectLeft	integer	非必须	目标在抓拍图片中的距离，左
└─cropRectTop	integer	非必须	目标在抓拍图片中的距离，上
└─cropRectRight	integer	非必须	目标在抓拍图片中的距离，右
└─cropRectBottom	integer	非必须	目标在抓拍图片中的距离，下
└─fullRectLeft	integer	非必须	目标在全景图片中的距离，左
└─fullRectTop	integer	非必须	目标在全景图片中的距离，上
└─fullRectRight	integer	非必须	目标在全景图片中的距离，右
└─fullRectBottom	integer	非必须	目标在全景图片中的距离，下
└─score	number	非必须	特征比对时相似对得分

HTTP请求示例

请求body

```
{  
  "currPage":1,  
  "pageSize":10,  
  "startTime":1622523870000,  
  "endTime":1622624670000  
}
```

HTTP响应示例

响应200

```

{
  "code":0,
  "data":{
    "list":[
      {
        "category":"3",
        "cropRectBottom":98,
        "cropRectLeft":35,
        "cropRectRight":237,
        "cropRectTop":14,
        "cropUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_2f0f90ddb2bd46d4b555e",
        "deviceLocation":"OpenAPI测试",
        "deviceName":"网络摄像机-MegEye-C4R-322-结构化",
        "deviceUuId":"c0a2388fcc444abbaa1b2b7c77a7d5d4",
        "fullRectBottom":98,
        "fullRectLeft":869,
        "fullRectRight":237,
        "fullRectTop":306,
        "fullUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_de7346f72521444fa6ab7",
        "id":"jpbkxnkBn31txtxEdg88",
        "plateColor":0,
        "plateNumber":"",
        "recognitionTime":1622539524615,
        "specialOperation":0,
        "vehicleBrand":0,
        "vehicleColor":11,
        "vehicleFunctional":0,
        "vehicleType":10
      },
      {
        "category":"3",
        "cropRectBottom":150,
        "cropRectLeft":27,
        "cropRectRight":184,
        "cropRectTop":22,
        "cropUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_d33e2e9fdcde441bbd282",
        "deviceLocation":"OpenAPI测试",
        "deviceName":"网络摄像机-MegEye-C4R-322-结构化",
        "deviceUuId":"c0a2388fcc444abbaa1b2b7c77a7d5d4",
        "fullRectBottom":150,
        "fullRectLeft":1720,
        "fullRectRight":184,
        "fullRectTop":450,
        "fullUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_bef30dba60fd4dccb446",
        "id":"ipbkxnkBn31txtxEZg-N",
        "plateColor":1,
        "plateNumber":"青FC3C10",
        "recognitionTime":1622539510458,
        "specialOperation":0,
        "vehicleBrand":31,
        "vehicleColor":0,
        "vehicleFunctional":0,
        "vehicleType":7
      },
      {
        "category":"3",
        "cropRectBottom":244,
        "cropRectLeft":44,
        "cropRectRight":297,
        "cropRectTop":36,
        "cropUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_26eaa20b3df47babff88",
        "deviceLocation":"OpenAPI测试",
        "deviceName":"网络摄像机-MegEye-C4R-322-结构化",
        "deviceUuId":"c0a2388fcc444abbaa1b2b7c77a7d5d4",

```

```

        "fullRectBottom":244,
        "fullRectLeft":1354,
        "fullRectRight":297,
        "fullRectTop":503,
        "fullUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_93e73ae87b1c472bbf50a",
        "id":"fZbkxnkBn31txtxEBQ8x",
        "plateColor":1,
        "plateNumber":"京N629V0",
        "recognitionTime":1622539499647,
        "specialOperation":0,
        "vehicleBrand":5,
        "vehicleColor":0,
        "vehicleFunctional":0,
        "vehicleType":2
    },
    {
        "category":"3",
        "cropRectBottom":63,
        "cropRectLeft":18,
        "cropRectRight":123,
        "cropRectTop":9,
        "cropUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_6ace4a5ee5cb431f9c8c0",
        "deviceLocation":"OpenAPI测试",
        "deviceName":"网络摄像机-MegEye-C4R-322-结构化",
        "deviceUuId":"c0a2388fcc444abbaa1b2b7c77a7d5d4",
        "fullRectBottom":63,
        "fullRectLeft":1178,
        "fullRectRight":123,
        "fullRectTop":320,
        "fullUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_9e43f8a3a2604632b4830",
        "id":"f5bkxnkBn31txtxEBQ86",
        "plateColor":0,
        "plateNumber":"",
        "recognitionTime":1622539494916,
        "specialOperation":0,
        "vehicleBrand":0,
        "vehicleColor":0,
        "vehicleFunctional":0,
        "vehicleType":2
    },
    {
        "category":"3",
        "cropRectBottom":70,
        "cropRectLeft":16,
        "cropRectRight":112,
        "cropRectTop":10,
        "cropUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_167d3edb26a044beaaa74",
        "deviceLocation":"OpenAPI测试",
        "deviceName":"网络摄像机-MegEye-C4R-322-结构化",
        "deviceUuId":"c0a2388fcc444abbaa1b2b7c77a7d5d4",
        "fullRectBottom":70,
        "fullRectLeft":1454,
        "fullRectRight":112,
        "fullRectTop":326,
        "fullUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_29bad7870b1b4e79a65f1",
        "id":"gpbkxnkBn31txtxEBg8m",
        "plateColor":0,
        "plateNumber":"",
        "recognitionTime":1622539490523,
        "specialOperation":0,
        "vehicleBrand":3,
        "vehicleColor":0,
        "vehicleFunctional":0,
        "vehicleType":18
    },
    {

```

```

        "category": "3",
        "cropRectBottom": 74,
        "cropRectLeft": 15,
        "cropRectRight": 104,
        "cropRectTop": 11,
        "cropUri": "http://10.171.4.130/pub/_ZzEwMF8zbQ==_428e087321dc40cdbf1dc",
        "deviceLocation": "OpenAPI测试",
        "deviceName": "网络摄像机-MegEye-C4R-322-结构化",
        "deviceUuId": "c0a2388fcc444abbaa1b2b7c77a7d5d4",
        "fullRectBottom": 74,
        "fullRectLeft": 1325,
        "fullRectRight": 104,
        "fullRectTop": 265,
        "fullUri": "http://10.171.4.130/pub/_ZzEwMF8zbQ==_189c8f1bdcd54d51b21ac",
        "id": "g5bkxnkBn31txtxEBg8q",
        "plateColor": 0,
        "plateNumber": "",
        "recognitionTime": 1622539483066,
        "specialOperation": 0,
        "vehicleBrand": 0,
        "vehicleColor": 10,
        "vehicleFunctional": 0,
        "vehicleType": 0
    },
    {
        "category": "3",
        "cropRectBottom": 213,
        "cropRectLeft": 30,
        "cropRectRight": 200,
        "cropRectTop": 32,
        "cropUri": "http://10.171.4.130/pub/_ZzEwMF8zbQ==_5ac141c59c664aa692e5c",
        "deviceLocation": "OpenAPI测试",
        "deviceName": "网络摄像机-MegEye-C4R-322-结构化",
        "deviceUuId": "c0a2388fcc444abbaa1b2b7c77a7d5d4",
        "fullRectBottom": 213,
        "fullRectLeft": 490,
        "fullRectRight": 200,
        "fullRectTop": 386,
        "fullUri": "http://10.171.4.130/pub/_ZzEwMF8zbQ==_11b199be30554e339025b",
        "id": "fpgkxnkBn31txtxEBQ82",
        "plateColor": 1,
        "plateNumber": "京QX5B65",
        "recognitionTime": 1622539482611,
        "specialOperation": 0,
        "vehicleBrand": 21,
        "vehicleColor": 11,
        "vehicleFunctional": 0,
        "vehicleType": 10
    },
    {
        "category": "3",
        "cropRectBottom": 57,
        "cropRectLeft": 18,
        "cropRectRight": 124,
        "cropRectTop": 8,
        "cropUri": "http://10.171.4.130/pub/_ZzEwMF8zbQ==_4a31bbbf0f2542d599e96",
        "deviceLocation": "OpenAPI测试",
        "deviceName": "网络摄像机-MegEye-C4R-322-结构化",
        "deviceUuId": "c0a2388fcc444abbaa1b2b7c77a7d5d4",
        "fullRectBottom": 57,
        "fullRectLeft": 1173,
        "fullRectRight": 124,
        "fullRectTop": 323,
        "fullUri": "http://10.171.4.130/pub/_ZzEwMF8zbQ==_1c9759cad16b4ec9b2edc",
        "id": "QZbjxnkBn31txtxEtg_1",
        "plateColor": 0,

```

```

        "plateNumber": "",
        "recognitionTime": 1622539471849,
        "specialOperation": 0,
        "vehicleBrand": 0,
        "vehicleColor": 15,
        "vehicleFunctional": 0,
        "vehicleType": 7
    },
    {
        "category": "3",
        "cropRectBottom": 200,
        "cropRectLeft": 37,
        "cropRectRight": 251,
        "cropRectTop": 30,
        "cropUri": "http://10.171.4.130/pub/_ZzEwMF8zbQ==_36e6643508d14427b4658",
        "deviceLocation": "OpenAPI测试",
        "deviceName": "网络摄像机-MegEye-C4R-322-结构化",
        "deviceUuId": "c0a2388fcc444abbaa1b2b7c77a7d5d4",
        "fullRectBottom": 200,
        "fullRectLeft": 657,
        "fullRectRight": 251,
        "fullRectTop": 476,
        "fullUri": "http://10.171.4.130/pub/_ZzEwMF8zbQ==_246bfe75ac08467f927b1",
        "id": "QJbjxnkbn31txtxEtg_d",
        "plateColor": 1,
        "plateNumber": "苏N629V0",
        "recognitionTime": 1622539469423,
        "specialOperation": 0,
        "vehicleBrand": 0,
        "vehicleColor": 15,
        "vehicleFunctional": 0,
        "vehicleType": 2
    },
    {
        "category": "3",
        "cropRectBottom": 70,
        "cropRectLeft": 16,
        "cropRectRight": 110,
        "cropRectTop": 10,
        "cropUri": "http://10.171.4.130/pub/_ZzEwMF8zbQ==_24d7d6778f644469adf8f",
        "deviceLocation": "OpenAPI测试",
        "deviceName": "网络摄像机-MegEye-C4R-322-结构化",
        "deviceUuId": "c0a2388fcc444abbaa1b2b7c77a7d5d4",
        "fullRectBottom": 70,
        "fullRectLeft": 1458,
        "fullRectRight": 110,
        "fullRectTop": 326,
        "fullUri": "http://10.171.4.130/pub/_ZzEwMF8zbQ==_1afa7ef802e644689058a",
        "id": "Ipbhxnkbn31txtxEtw-X",
        "plateColor": 0,
        "plateNumber": "",
        "recognitionTime": 1622539314214,
        "specialOperation": 0,
        "vehicleBrand": 3,
        "vehicleColor": 0,
        "vehicleFunctional": 0,
        "vehicleType": 18
    }
],
"pageNum": 1,
"pageSize": 10,
"total": 37
},
"msg": "成功"
}

```

1.6. 机动车数据详情查询

1.6.1. 基本信息

Path: /v1/api/structure/motorVehicle/query

Method: POST

接口描述：根据数据唯一id查询机动车数据详情

1.6.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

1.6.3. 返回数据

名称	类型	是否必须	备注
cropUri	string	非必须	抓拍图URL
plateNumber	string	非必须	车牌号码
fullUri	string	非必须	图片全景图URL
vehicleType	integer	非必须	车辆类型，枚举值有：{0=未知, 1=两厢轿车, 2=轿车, 3=轿跑, 4=小型轿车, 5=微型轿车, 6=MPV, 7=SUV, 8=大型客车, 9=中型客车, 10=面包车, 11=微型面包车, 12=货车, 13=中型货车, 14=油罐车, 15=吊车, 16=渣土车, 17=小货车, 18=皮卡, 19=微卡, 22=行人, 29=挂车, 30=小型车(大范围), 31=大型车(大范围)}
deviceUuld	string	非必须	设备Uuld
vehicleColor	integer	非必须	车身颜色，枚举值有：{0=白色, 1=红色, 2=黄色, 5=绿色, 7=紫色, 8=粉色, 9=黑色, 10=橙色, 11=银色, 12=香槟色, 13=灰色, 15=蓝色, 21=青色, 29=棕色, 32=深蓝, 40=彩色, 99=未知}
deviceName	string	非必须	设备名
plateColor	integer	非必须	车牌颜色，枚举值有：{0=未知, 1=蓝色, 2=黄色, 3=白色, 4=黑色, 5=绿色, 6=渐变绿底黑字, 7=黄绿双拼底黑字}
deviceLocation	string	非必须	设备所在位置

名称	类型	是否必须	备注
vehicleBrand	integer	非必须	车辆品牌，枚举值有：{0=其它, 1=大众, 2=奥迪, 3=丰田, 4=本田, 5=奔驰, 6=雪佛兰, 7=奇瑞, 8=别克, 9=长城, 10=现代, 11=尼桑, 12=福特, 13=宝马, 14=雪铁龙, 15=起亚, 16=铃木, 17=马自达, 18=比亚迪, 20=标致, 21=长安, 22=雷克萨斯, 23=中华, 24=斯柯达, 25=海马, 26=夏利, 27=五菱, 28=东风, 29=哈飞, 30=一汽, 31=宝骏, 32=帝豪(吉利), 33=MG名爵, 34=东南, 35=皇冠(丰田), 36=金杯, 37=三菱, 38=荣威, 39=吉利, 40=英伦(吉利), 41=吉利全球鹰(吉利), 42=哈飞赛豹, 43=长丰, 44=北汽威旺, 45=北汽汽车, 46=新凯, 47=吉奥汽车, 48=玛莎拉蒂, 49=罗孚, 50=奥斯汀, 52=道奇, 55=捷豹, 57=阿尔法, 58=兰博基尼, 59=布加迪, 60=林肯, 61=法拉利汽车, 62=昌河, 63=菲亚特, 64=福田, 65=讴歌, 66=莲花汽车, 67=华普汽车, 68=红旗, 69=瑞麟, 70=一汽奔腾(一汽), 71=威麟汽车, 72=众泰, 73=力帆, 74=JEEP(吉普), 75=中兴, 76=开瑞, 77=路虎, 78=迈巴赫, 79=雷诺, 80=欧宝, 81=野马, 82=吉普, 83=依维柯, 84=英菲尼迪, 85=斯巴鲁, 86=阿斯顿·马丁, 87=安凯客车, 88=保时捷, 89=宾利, 90=福迪, 91=福建奔驰, 92=吉姆士, 93=观致, 94=广汽, 95=双龙, 96=海格, 97=悍马, 98=华泰, 99=黄海, 100=九龙客车, 101=理念, 102=奔驰SMART, 103=陆风, 104=纳智捷, 105=欧朗, 106=启辰, 107=哈弗HAVAL, 108=豪沃, 109=上海汇众(上汽集团), 110=金龙, 112=江淮, 113=JMC(江铃), 114=凯迪拉克, 116=宝马MINI, 117=陕汽, 118=少林, 119=沃尔沃, 120=五十铃(庆铃), 121=跃进, 122=宇通, 123=中通, 125=申沃, 126=扬子江, 128=北奔重卡, 130=劳斯莱斯, 131=特斯拉, 132=腾势, 134=英致, 135=西雅特, 136=亚星客车, 137=大宇客车, 138=长安谛艾仕, 139=凯翼, 140=上汽大通, 141=本田思铭, 142=凯马}
category	string	非必须	结构化的类别，枚举值有：{1=人脸, 2=人体, 3=机动车, 4=非机动车}
vehicleFunctional	integer	非必须	功能用车，枚举值有：{0=未知, 1=挖掘机, 2=渣土车, 3=校车, 4=救护车, 5=警车, 6=集装箱车}

名称	类型	是否必须	备注
recognitionTime	integer	非必须	识别时间，时间戳（毫秒）
specialOperation	integer	非必须	专项作业，枚举值有：{0=未知, 7=混凝土搅拌机, 8=消防车}
cropRectLeft	integer	非必须	目标在抓拍图片中的距离，左
cropRectTop	integer	非必须	目标在抓拍图片中的距离，上
cropRectRight	integer	非必须	目标在抓拍图片中的距离，右
cropRectBottom	integer	非必须	目标在抓拍图片中的距离，下
fullRectLeft	integer	非必须	目标在全景图片中的距离，左
fullRectTop	integer	非必须	目标在全景图片中的距离，上
fullRectRight	integer	非必须	目标在全景图片中的距离，右
fullRectBottom	integer	非必须	目标在全景图片中的距离，下

HTTP请求示例

请求body

```
{
  "id": "jpbkxnk8n31txtxEg88"
}
```

HTTP响应示例

响应200

```
{
  "code":0,
  "data":{
    "category":"3",
    "cropRectBottom":98,
    "cropRectLeft":35,
    "cropRectRight":237,
    "cropRectTop":14,
    "cropUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_2f0f90ddb2bd46d4b555e557056ca",
    "deviceLocation":"OpenAPI测试",
    "deviceName":"网络摄像机-MegEye-C4R-322-结构化",
    "deviceUuid":"c0a2388fcc444abbaa1b2b7c77a7d5d4",
    "fullRectBottom":98,
    "fullRectLeft":869,
    "fullRectRight":237,
    "fullRectTop":306,
    "fullUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_de7346f72521444fa6ab725c70243",
    "plateColor":0,
    "plateNumber":"",
    "recognitionTime":1622539524615,
    "specialOperation":0,
    "vehicleBrand":0,
    "vehicleColor":11,
    "vehicleFunctional":0,
    "vehicleType":10
  },
  "msg":"成功"
}
```

1.7. 非机动车数据检索

1.7.1. 基本信息

Path: /v1/api/structure/nonMotorVehicle/search

Method: POST

接口描述: 根据查询条件分页查询非机动车数据

1.7.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
currPage	integer	非必须	当前页，默认是1
vehicleType	integer	非必须	车辆类型，枚举值有：{20=两轮车, 21=三轮车, 100=自行车, 101=电动车, 102=摩托车}
pageSize	integer	非必须	每页条数，默认是10
scoreThreshold	number	非必须	相似度，默认是60
startTime	integer	必须	开始时间，时间戳（毫秒）
endTime	integer	必须	结束时间，时间戳（毫秒）
deviceUuldList	string []	非必须	设备uuid
searchImageUrl	string	非必须	被检索图片的URL，被图片可以通过接口/v1/api/misc/file/upload上传，将这个接口返回的uri设置到这个字段即可

1.7.3. 返回数据

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码
pageSize	integer	非必须	分页大小
total	integer	非必须	总数
list	object []	非必须	
└─id	string	非必须	数据的唯一ID
└─vehicleType	string	非必须	车辆类型，枚举值有：{20=两轮车, 21=三轮车, 100=自行车, 101=电动车, 102=摩托车}
└─cropUri	string	非必须	抓拍图URL
└─fullUri	string	非必须	图片全景图URL
└─deviceUuld	string	非必须	设备Uuld
└─deviceName	string	非必须	设备名
└─deviceLocation	string	非必须	设备所在位置
└─category	string	非必须	结构化的类别，枚举值有：{1=人脸, 2=人体, 3=机动车, 4=非机动车}
└─recognitionTime	integer	非必须	识别时间，时间戳（毫秒）
└─cropRectLeft	integer	非必须	目标在抓拍图片中的距离，左
└─cropRectTop	integer	非必须	目标在抓拍图片中的距离，上

名称	类型	是否必须	备注
└─cropRectRight	integer	非必须	目标在抓拍图片中的距离，右
└─cropRectBottom	integer	非必须	目标在抓拍图片中的距离，下
└─fullRectLeft	integer	非必须	目标在全景图片中的距离，左
└─fullRectTop	integer	非必须	目标在全景图片中的距离，上
└─fullRectRight	integer	非必须	目标在全景图片中的距离，右
└─fullRectBottom	integer	非必须	目标在全景图片中的距离，下
└─score	number	非必须	特征比对时相似对得分

HTTP请求示例

请求body

```
{
  "currPage":1,
  "pageSize":10,
  "startTime":1622523870000,
  "endTime":1622624670000
}
```

HTTP响应示例

响应200

```

{
  "code":0,
  "data":{
    "list":[
      {
        "category":"4",
        "cropRectBottom":66,
        "cropRectLeft":5,
        "cropRectRight":35,
        "cropRectTop":9,
        "cropUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_dfc883e8bf8a4749b8343",
        "deviceLocation":"OpenAPI测试",
        "deviceName":"网络摄像机-MegEye-C4R-322-结构化",
        "deviceUuId":"c0a2388fcc444abbaa1b2b7c77a7d5d4",
        "fullRectBottom":66,
        "fullRectLeft":1880,
        "fullRectRight":35,
        "fullRectTop":419,
        "fullUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_bdd7dbed83484205919a9",
        "id":"hpbkxnkBn31txtxERQ9v",
        "recognitionTime":1622539509722,
        "vehicleType":"100"
      },
      {
        "category":"4",
        "cropRectBottom":105,
        "cropRectLeft":23,
        "cropRectRight":158,
        "cropRectTop":15,
        "cropUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_e87b271ef5924023a019a9",
        "deviceLocation":"OpenAPI测试",
        "deviceName":"网络摄像机-MegEye-C4R-322-结构化",
        "deviceUuId":"c0a2388fcc444abbaa1b2b7c77a7d5d4",
        "fullRectBottom":105,
        "fullRectLeft":43,
        "fullRectRight":158,
        "fullRectTop":507,
        "fullUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_2b06eb07cbb04cd8b244e",
        "id":"hJbkxnkBn31txtxEBg8t",
        "recognitionTime":1622539493755,
        "vehicleType":"100"
      },
      {
        "category":"4",
        "cropRectBottom":97,
        "cropRectLeft":8,
        "cropRectRight":57,
        "cropRectTop":14,
        "cropUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_1bb67e6680034cb2bbdcf",
        "deviceLocation":"OpenAPI测试",
        "deviceName":"网络摄像机-MegEye-C4R-322-结构化",
        "deviceUuId":"c0a2388fcc444abbaa1b2b7c77a7d5d4",
        "fullRectBottom":97,
        "fullRectLeft":235,
        "fullRectRight":57,
        "fullRectTop":493,
        "fullUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_d5fbdc15f00443ec8a38e",
        "id":"gJbkxnkBn31txtxEBQ_B",
        "recognitionTime":1622539491901,
        "vehicleType":"100"
      },
      {
        "category":"4",
        "cropRectBottom":127,
        "cropRectLeft":12,

```

```

        "cropRectRight":81,
        "cropRectTop":19,
        "cropUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_904c5d82f6074f9ba7536",
        "deviceLocation":"OpenAPI测试",
        "deviceName":"网络摄像机-MegEye-C4R-322-结构化",
        "deviceUuId":"c0a2388fcc444abbaa1b2b7c77a7d5d4",
        "fullRectBottom":127,
        "fullRectLeft":322,
        "fullRectRight":81,
        "fullRectTop":517,
        "fullUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_3ddfebef486a402892264",
        "id":"gZbkxnkBn31txtxEBQ_M",
        "recognitionTime":1622539480933,
        "vehicleType":"101"
    },
    {
        "category":"4",
        "cropRectBottom":296,
        "cropRectLeft":29,
        "cropRectRight":194,
        "cropRectTop":44,
        "cropUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_fcf1416da245424babfd",
        "deviceLocation":"OpenAPI测试",
        "deviceName":"网络摄像机-MegEye-C4R-322-结构化",
        "deviceUuId":"c0a2388fcc444abbaa1b2b7c77a7d5d4",
        "fullRectBottom":296,
        "fullRectLeft":156,
        "fullRectRight":194,
        "fullRectTop":657,
        "fullUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_dcd34908d25f45568808",
        "id":"fJbkxnkBn31txtxEBA-c",
        "recognitionTime":1622539480290,
        "vehicleType":"101"
    },
    {
        "category":"4",
        "cropRectBottom":58,
        "cropRectLeft":10,
        "cropRectRight":73,
        "cropRectTop":8,
        "cropUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_1795cba115d94b2a8fea7",
        "deviceLocation":"OpenAPI测试",
        "deviceName":"网络摄像机-MegEye-C4R-322-结构化",
        "deviceUuId":"c0a2388fcc444abbaa1b2b7c77a7d5d4",
        "fullRectBottom":58,
        "fullRectLeft":836,
        "fullRectRight":73,
        "fullRectTop":335,
        "fullUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_8445886bafc24e91b5db2",
        "id":"QpbjxnkBn31txtxEug_P",
        "recognitionTime":1622539475668,
        "vehicleType":"21"
    },
    {
        "category":"4",
        "cropRectBottom":127,
        "cropRectLeft":11,
        "cropRectRight":79,
        "cropRectTop":19,
        "cropUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_6e79106b964342be9433b",
        "deviceLocation":"OpenAPI测试",
        "deviceName":"网络摄像机-MegEye-C4R-322-结构化",
        "deviceUuId":"c0a2388fcc444abbaa1b2b7c77a7d5d4",
        "fullRectBottom":127,
        "fullRectLeft":323,
        "fullRectRight":79,

```



```

        "fullRectTop":517,
        "fullUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_6c890a4602a8444995e9a",
        "id":"JZbhxnkbn31txtxETw-l",
        "recognitionTime":1622539315900,
        "vehicleType":"101"
    },
    {
        "category":"4",
        "cropRectBottom":113,
        "cropRectLeft":21,
        "cropRectRight":146,
        "cropRectTop":16,
        "cropUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_77b7efe4718741f592385",
        "deviceLocation":"OpenAPI测试",
        "deviceName":"网络摄像机-MegEye-C4R-322-结构化",
        "deviceUuId":"c0a2388fcc444abbaa1b2b7c77a7d5d4",
        "fullRectBottom":113,
        "fullRectLeft":50,
        "fullRectRight":146,
        "fullRectTop":498,
        "fullUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_ebff1fe35a694218ad70b",
        "id":"JJbhxnkbn31txtxETw-g",
        "recognitionTime":1622539315796,
        "vehicleType":"100"
    },
    {
        "category":"4",
        "cropRectBottom":97,
        "cropRectLeft":8,
        "cropRectRight":57,
        "cropRectTop":14,
        "cropUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_f339c8bc458c4cfa9e5c1",
        "deviceLocation":"OpenAPI测试",
        "deviceName":"网络摄像机-MegEye-C4R-322-结构化",
        "deviceUuId":"c0a2388fcc444abbaa1b2b7c77a7d5d4",
        "fullRectBottom":97,
        "fullRectLeft":235,
        "fullRectRight":57,
        "fullRectTop":493,
        "fullUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_33ab38f83d8c4d5ea313f",
        "id":"Jpbhxnkbn31txtxEXg9m",
        "recognitionTime":1622539302904,
        "vehicleType":"100"
    },
    {
        "category":"4",
        "cropRectBottom":70,
        "cropRectLeft":5,
        "cropRectRight":39,
        "cropRectTop":10,
        "cropUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_24e9e9132d6e4e1ca6595",
        "deviceLocation":"OpenAPI测试",
        "deviceName":"网络摄像机-MegEye-C4R-322-结构化",
        "deviceUuId":"c0a2388fcc444abbaa1b2b7c77a7d5d4",
        "fullRectBottom":70,
        "fullRectLeft":1877,
        "fullRectRight":39,
        "fullRectTop":417,
        "fullUri":"http://10.171.4.130/pub/_ZzEwMF8zbQ==_a8d0dd9112d743d8b07e3",
        "id":"u5bQxnkbn31txtxERA28",
        "recognitionTime":1622538195850,
        "vehicleType":"100"
    }
},
"pageNum":1,
"pageSize":10,

```

```
    "total":30
  },
  "msg":"成功"
}
```

1.8. 非机动车数据详情查询

1.8.1. 基本信息

Path: /v1/api/structure/nonMotorVehicle/query

Method: POST

接口描述: 根据数据唯一id，分页查询非机动车数据详细信息

1.8.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

1.8.3. 返回数据

名称	类型	是否必须	备注
cropUri	string	非必须	抓拍图URL
vehicleType	string	非必须	车辆类型，枚举值有：{20=两轮车, 21=三轮车, 100=自行车, 101=电动车, 102=摩托车}
fullUri	string	非必须	图片全景图URL
deviceUuld	string	非必须	设备Uuld
deviceName	string	非必须	设备名
deviceLocation	string	非必须	设备所在位置
category	string	非必须	结构化的类别，枚举值有：{1=人脸, 2=人体, 3=机动车, 4=非机动车}
recognitionTime	integer	非必须	识别时间，时间戳（毫秒）
cropRectLeft	integer	非必须	目标在抓拍图片中的距离，左
cropRectTop	integer	非必须	目标在抓拍图片中的距离，上
cropRectRight	integer	非必须	目标在抓拍图片中的距离，右
cropRectBottom	integer	非必须	目标在抓拍图片中的距离，下
fullRectLeft	integer	非必须	目标在全景图片中的距离，左
fullRectTop	integer	非必须	目标在全景图片中的距离，上
fullRectRight	integer	非必须	目标在全景图片中的距离，右

名称	类型	是否必须	备注
fullRectBottom	integer	非必须	目标在全景图片中的距离，下

HTTP请求示例

请求body

```
{
  "id": "hpbkxnkBn31txtxERQ9v"
}
```

HTTP响应示例

响应200

```
{
  "code": 0,
  "data": {
    "category": "4",
    "cropRectBottom": 66,
    "cropRectLeft": 5,
    "cropRectRight": 35,
    "cropRectTop": 9,
    "cropUri": "http://10.171.4.130/pub/_ZzEwMF8zbQ==_dfc883e8bf8a4749b8343fd1af63a",
    "deviceLocation": "OpenAPI测试",
    "deviceName": "网络摄像机-MegEye-C4R-322-结构化",
    "deviceUuId": "c0a2388fcc444abbaa1b2b7c77a7d5d4",
    "fullRectBottom": 66,
    "fullRectLeft": 1880,
    "fullRectRight": 35,
    "fullRectTop": 419,
    "fullUri": "http://10.171.4.130/pub/_ZzEwMF8zbQ==_bdd7dbed83484205919a9e7f073d2",
    "recognitionTime": 1622539509722,
    "vehicleType": "100"
  },
  "msg": "成功"
}
```

- 1. 警戒记录
 - 1.1. 警戒记录列表查询
 - 1.1.1. 基本信息
 - 1.1.2. 请求参数
 - 1.1.3. 返回数据
 - 1.1.4. HTTP请求示例
 - 1.1.5. HTTP响应示例

1. 警戒记录

1.1. 警戒记录列表查询

1.1.1. 基本信息

Path: /v1/api/areaAlert/record/list

Method: POST

接口描述:

首先在盘古页面进行警戒布控，才能产生警戒记录。然后根据条件(警戒设备uuid、警戒类型、警戒开始时间、警戒结束时间、分页信息)来查询出警戒记录。

1.1.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
deviceUuldList	string []	非必须	设备id列表
type	integer	非必须	警戒类型，1：人员越界，2：人员入侵，3：车辆越界，31：机动车越界，32：非机动车越界，4：车辆禁停，41：机动车禁停，42：非机动车禁停，5：人员越界-翻墙检测，6：人员徘徊，7：机动车离开，8：非机动车离开，10：人员值岗/离岗-离岗，11：人员值岗/离岗-超员，12：人员值岗/离岗-少员，13：人员奔跑，14：摔倒检测，15：人员扭打，16：抽烟检测，17：看手机，18：接打电话，19：人员聚众，20：防碰撞预警，21：车辆逆行，22：睡岗检测
startTime	integer	必须	开始时间,毫秒时间戳
endTime	integer	必须	结束时间,毫秒时间戳
currPage	integer	非必须	当前页，默认为1
pageSize	integer	非必须	每页条数，默认为10

1.1.3. 返回数据

名称	类型	是否必须	备注
pageNum	integer	非必须	当前页码
pageSize	integer	非必须	分页大小
total	integer	非必须	总数
list	object []	非必须	
alarmRecordUuld	string	非必须	警戒记录uuid
alarmControlType	integer	非必须	1-警戒机, 2-警戒算法仓
wholeImageUrl	string	非必须	全景图
alarmType	integer	非必须	警戒类型, 1: 人员越界, 2: 人员入侵, 3: 车辆越界, 31: 机动车越界, 32: 非机动车越界, 4: 车辆禁停, 41: 机动车禁停, 42: 非机动车禁停, 5: 人员越界-翻墙检测, 6: 人员徘徊, 7: 机动车离开, 8: 非机动车离开, 10: 人员值岗/离岗-离岗, 11: 人员值岗/离岗-超员, 12: 人员值岗/离岗-少员, 13: 人员奔跑, 14: 摔倒检测, 15: 人员扭打, 16: 抽烟检测, 17: 看手机, 18: 接打电话, 19: 人员聚众, 20: 防碰撞预警, 21: 车辆逆行, 22: 睡岗检测
deviceUuld	string	非必须	设备uuid
deviceName	string	非必须	设备名称
alarmTime	integer	非必须	警戒时间,毫秒时间戳,算法仓设备为警戒开始时间

名称	类型	是否必须	备注
alarmEndTime	integer	非必须	警戒结束时间,毫秒时间戳;警戒类型为人员入侵、人员离岗时存在
continueTime	number	非必须	警戒持续时间, 单位: 秒;警戒类型为人员入侵、人员离岗时存在
drawLineDetail	object []	非必须	划线坐标信息
firstNodeDetail	string	非必须	画线详情,第一个节点详情
secondNodeDetail	string	非必须	画线详情,第二个节点详情,双拌线并且是预警规则时存在
inTheArea	boolean	非必须	第一个画线详情,true表示在此区域, false表示不在此区域
targetId	string	非必须	报警对象的id, 这个对于单个视频是唯一的, 更新事件结束时间时用到
areaId	integer	非必须	布控画线所属区域id, 同一个场景下的画线规则不允许重复, -2代表全局
sceneId	integer	非必须	场景id, 0-场景1, 1-场景2, 2-场景3, 3-场景4
targetRect	object []	非必须	目标框在全景图中的位置
ruleConfig	object	非必须	布控内容string,例如: { "stayTime":1, "delayAlarmTime":60, "personNumber":1 }
count	integer	非必须	离岗超员/少员、聚众人数
shieldDrawLineDetail	object []	非必须	屏蔽区域画线坐标
shieldDrawLineDetail	string	非必须	
areaMsg	string	非必须	区域画线名称

1.1.4. HTTP请求示例

请求 **path**

```
/v1/api/areaAlert/record/list
```

请求 **body**

```
{
  "deviceUuIdList": ["62fd1d5c4e7a4f99848f1443ab7a5810"],
  "type": 1,
  "startTime": 1622523870000,
  "endTime": 1622624670000,
  "currPage": 1,
  "pageSize": 10
}
```

1.1.5. HTTP响应示例

响应 **200**

```

{
  "code": 0,
  "data": {
    "list": [
      {
        "alarmControlType": 2,
        "alarmRecordUuid": "d551fc9e11d9447192e1fb0fbdeaaa77",
        "alarmTime": 1624589365000,
        "alarmType": 2,
        "areaId": 0,
        "areaMsg": "区域1",
        "count": 1,
        "deviceName": "警戒算法虚拟流-人员",
        "deviceUuid": "62fd1d5c4e7a4f99848f1443ab7a5810",
        "drawLineDetail": [
          {
            "firstNodeDetail": "[{\\"x\\":0.18600682593856654,\\"y\\":0.257868",
            "inTheArea": true
          }
        ],
        "ruleConfig": {
          "stayTime": 1,
          "drawType": 1
        },
        "sceneId": 1,
        "shieldDrawLineDetail": [],
        "targetId": "[\\"person:f425e948-d55f-11eb-9a8f-ac1f6b9868ee:6\\"]",
        "targetRect": [
          {
            "top": -4.9274793,
            "left": 53.597534,
            "bottom": 41.688705,
            "right": 65.65099
          }
        ],
        "wholeImageUrl": "http://10.122.101.86/pub/_ZzEwMF8zbQ==_ab4bc842b0f24
      },
      {
        "alarmControlType": 2,
        "alarmRecordUuid": "5a914fc998814f22b03655ee696155f2",
        "alarmTime": 1624589278000,
        "alarmType": 2,
        "areaId": 0,
        "areaMsg": "区域1",
        "count": 1,
        "deviceName": "警戒算法虚拟流-人员",
        "deviceUuid": "62fd1d5c4e7a4f99848f1443ab7a5810",
        "drawLineDetail": [
          {
            "firstNodeDetail": "[{\\"x\\":0.18600682593856654,\\"y\\":0.257868",
            "inTheArea": true
          }
        ],
        "ruleConfig": {
          "stayTime": 1,
          "drawType": 1
        },
        "sceneId": 1,
        "shieldDrawLineDetail": [],
        "targetId": "[\\"person:c0478064-d55f-11eb-9d5e-ac1f6b9868ee:22\\"]",
        "targetRect": [
          {
            "top": -1.422417,
            "left": 76.50203,
            "bottom": 47.683025,

```

```

        "right": 88.56769
    }
},
"wholeImageUrl": "http://10.122.101.86/pub/_ZzEwMF8zbQ==_eba635d44d494
},
{
    "alarmControlType": 2,
    "alarmRecordUuId": "c50b053d37cd475d83150caada36abfa",
    "alarmTime": 1624589158000,
    "alarmType": 1,
    "areaId": 0,
    "areaMsg": "拌线1",
    "count": 1,
    "deviceName": "警戒算法虚拟流-人员",
    "deviceUuId": "62fd1d5c4e7a4f99848f1443ab7a5810",
    "drawLineDetail": [
        {
            "firstNodeDetail": "[{\\"x\\":0.8959044368600683,\\"y\\":0.5582106
            "inTheArea": true
        }
    ],
    "ruleConfig": {
        "drawType": 3,
        "delayAlarmTime": 3
    },
    "sceneId": 0,
    "shieldDrawLineDetail": [],
    "targetId": "[\\"person:75299c66-d55f-11eb-8234-ac1f6b9868ee:5\\"]",
    "targetRect": [
        {
            "top": -1.9168109,
            "left": 63.24581,
            "bottom": 45.980465,
            "right": 75.35093
        }
    ],
    "wholeImageUrl": "http://10.122.101.86/pub/_ZzEwMF8zbQ==_9cc2fb1d23294
},
{
    "alarmControlType": 2,
    "alarmRecordUuId": "c7b27bf2724e40898115212da8649a4c",
    "alarmTime": 1624589152000,
    "alarmType": 2,
    "areaId": 0,
    "areaMsg": "区域1",
    "count": 1,
    "deviceName": "警戒算法虚拟流-人员",
    "deviceUuId": "62fd1d5c4e7a4f99848f1443ab7a5810",
    "drawLineDetail": [
        {
            "firstNodeDetail": "[{\\"x\\":0.18600682593856654,\\"y\\":0.257868
            "inTheArea": true
        }
    ],
    "ruleConfig": {
        "stayTime": 1,
        "drawType": 1
    },
    "sceneId": 1,
    "shieldDrawLineDetail": [],
    "targetId": "[\\"person:75299c66-d55f-11eb-8234-ac1f6b9868ee:5\\"]",
    "targetRect": [
        {
            "top": -6.8047338,
            "left": 55.314774,
            "bottom": 42.76008,

```

```

        "right": 66.946556
    }
},
"wholeImageUrl": "http://10.122.101.86/pub/_ZzEwMF8zbQ==_ddd3d60863814
},
{
    "alarmControlType": 2,
    "alarmRecordUuId": "4fc42248d07a433787fe398a439ea7ed",
    "alarmTime": 1624589062000,
    "alarmType": 1,
    "areaId": 0,
    "areaMsg": "拌线1",
    "count": 1,
    "deviceName": "警戒算法虚拟流-人员",
    "deviceUuId": "62fd1d5c4e7a4f99848f1443ab7a5810",
    "drawLineDetail": [
        {
            "firstNodeDetail": "[{\\"x\\":0.8959044368600683,\\"y\\":0.5582106
            "inTheArea": true
        }
    ],
    "ruleConfig": {
        "drawType": 3,
        "delayAlarmTime": 3
    },
    "sceneId": 0,
    "shieldDrawLineDetail": [],
    "targetId": "[\\"person:3b5a544e-d55f-11eb-9f86-ac1f6b9868ee:6\\"]",
    "targetRect": [
        {
            "top": -4.5526495,
            "left": 67.47632,
            "bottom": 39.630642,
            "right": 77.71192
        }
    ],
    "wholeImageUrl": "http://10.122.101.86/pub/_ZzEwMF8zbQ==_0ed3bc5e50a34
},
{
    "alarmControlType": 2,
    "alarmRecordUuId": "144c3e17cb394e5bbc52528937b8dc8f",
    "alarmTime": 1624589058000,
    "alarmType": 2,
    "areaId": 0,
    "areaMsg": "区域1",
    "count": 1,
    "deviceName": "警戒算法虚拟流-人员",
    "deviceUuId": "62fd1d5c4e7a4f99848f1443ab7a5810",
    "drawLineDetail": [
        {
            "firstNodeDetail": "[{\\"x\\":0.18600682593856654,\\"y\\":0.257868
            "inTheArea": true
        }
    ],
    "ruleConfig": {
        "stayTime": 1,
        "drawType": 1
    },
    "sceneId": 1,
    "shieldDrawLineDetail": [],
    "targetId": "[\\"person:3c895c02-d55f-11eb-9f86-ac1f6b9868ee:7\\"]",
    "targetRect": [
        {
            "top": -4.046484,
            "left": 59.998413,
            "bottom": 41.05018,

```

```

        "right": 69.54478
    }
},
"wholeImageUrl": "http://10.122.101.86/pub/_ZzEwMF8zbQ==_1ce7552a3d144
},
{
    "alarmControlType": 2,
    "alarmRecordUuId": "4306a80481fc4d93bca19de368ac9ffb",
    "alarmTime": 1624588956000,
    "alarmType": 1,
    "areaId": 0,
    "areaMsg": "拌线1",
    "count": 1,
    "deviceName": "警戒算法虚拟流-人员",
    "deviceUuId": "62fd1d5c4e7a4f99848f1443ab7a5810",
    "drawLineDetail": [
        {
            "firstNodeDetail": "[{\\"x\\":0.8959044368600683,\\"y\\":0.5582100
            "inTheArea": true
        }
    ],
    "ruleConfig": {
        "drawType": 3,
        "delayAlarmTime": 3
    },
    "sceneId": 0,
    "shieldDrawLineDetail": [],
    "targetId": "[\\"person:fb8f01c0-d55e-11eb-9a8f-ac1f6b9868ee:4\\"]",
    "targetRect": [
        {
            "top": -5.5814805,
            "left": 56.620174,
            "bottom": 40.145218,
            "right": 69.16712
        }
    ],
    "wholeImageUrl": "http://10.122.101.86/pub/_ZzEwMF8zbQ==_2da0f15b89b44
},
{
    "alarmControlType": 2,
    "alarmRecordUuId": "8a3953a7be174aba97f94f368d475034",
    "alarmTime": 1624588949000,
    "alarmType": 2,
    "areaId": 0,
    "areaMsg": "区域1",
    "count": 1,
    "deviceName": "警戒算法虚拟流-人员",
    "deviceUuId": "62fd1d5c4e7a4f99848f1443ab7a5810",
    "drawLineDetail": [
        {
            "firstNodeDetail": "[{\\"x\\":0.18600682593856654,\\"y\\":0.257868
            "inTheArea": true
        }
    ],
    "ruleConfig": {
        "stayTime": 1,
        "drawType": 1
    },
    "sceneId": 1,
    "shieldDrawLineDetail": [],
    "targetId": "[\\"person:fb8f01c0-d55e-11eb-9a8f-ac1f6b9868ee:4\\"]",
    "targetRect": [
        {
            "top": -6.8047338,
            "left": 54.91622,
            "bottom": 42.55575,

```

```

        "right": 67.64477
    }
},
"wholeImageUrl": "http://10.122.101.86/pub/_ZzEwMF8zbQ==_5a7043049cbc4
},
{
    "alarmControlType": 2,
    "alarmRecordUuId": "c1c92773d49745f9a4364c24e48f4be0",
    "alarmTime": 1624561301000,
    "alarmType": 16,
    "areaId": -2,
    "areaMsg": "全局",
    "count": 1,
    "deviceName": "警戒算法虚拟流-人员",
    "deviceUuId": "62fd1d5c4e7a4f99848f1443ab7a5810",
    "drawLineDetail": [
        {
            "firstNodeDetail": "[{\\"x\\":0,\\"y\\":0},{\\"x\\":0,\\"y\\":1},{\\"x\\":1,\\"y\\":0},{\\"x\\":1,\\"y\\":1}]",
            "inTheArea": true
        }
    ],
    "ruleConfig": {
        "drawType": 6,
        "delayAlarmTime": 3
    },
    "sceneId": 0,
    "shieldDrawLineDetail": [
        {
            "shieldDrawLineDetail": "[{\\"x\\":0.1286438109580779,\\"y\\":0.1111111111111111}]",
        },
        {
            "shieldDrawLineDetail": "[{\\"x\\":0.6586563121053588,\\"y\\":0.2727272727272727}]",
        }
    ],
    "targetId": "[\\"person:99848ef4-d51e-11eb-9f86-ac1f6b9868ee:10\\"]",
    "targetRect": [
        {
            "top": 4.477406,
            "left": 60.013153,
            "bottom": 103.47008,
            "right": 78.71064
        }
    ],
    "wholeImageUrl": "http://10.122.101.86/pub/_ZzEwMF8zbQ==_c1900a60e2f84
    }
},
"pageNum": 1,
"pageSize": 10,
"total": 9
},
"msg": "成功"
}

```

- 1. 视频监控
 - 1.1. 云台控制
 - 1.1.1. 基本信息
 - 1.1.2. 请求参数
 - 1.1.3. 返回数据
 - 1.2. 视频实时预览（不建议使用）
 - 1.2.1. 基本信息
 - 1.2.2. 请求参数
 - 1.2.3. 返回数据
 - 1.3. 视频抓图
 - 1.3.1. 基本信息
 - 1.3.2. 请求参数
 - 1.3.3. 返回数据

1. 视频监控

1.1. 云台控制

1.1.1. 基本信息

Path: /v1/api/video/realtime/ptzControl

Method: POST

接口描述: 可以通过传入设备uuid以及控制命令来控制具有云台控制功能的设备进行移动、变焦、光圈变化等操作

1.1.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
deviceId	string	必须	设备uuid
ptzCmd	integer	必须	云台控制命令:1<上;2<下;3<左;4<右;5<左上;6<右上;7<左下;8<右下;9<焦距变大（变倍+）;10<焦距变小（变倍-）;11<焦点前调（调焦+）;12<焦点后调（调焦-）;13<光圈扩大;14<光圈缩小< span="">
speed	integer	必须	速度:控制云台、镜头的速度，范围为[1-8]
stop	integer	必须	停止或者开始动作:云台停止动作或开始动作：0-开始，非0-停止

1.1.3. 返回数据

OK

HTTP请求示例

请求body

```
{
  "deviceId": "c0a2388fcc444abbaa1b2b7c77a7d5d4",
  "ptzCmd": 1,
  "speed": 3,
  "stop": 0
}
```

HTTP响应示例

响应200

```
{
  "code": 0,
  "data": {
  },
  "msg": "成功"
}
```

1.2. 视频实时预览（不建议使用）

1.2.1. 基本信息

Path: /v1/api/video/realtime/preview

Method: POST

接口描述: 通过传入设备uuid获取到设备实时预览的流url, 此url客户端通过ws连接GMP服务获取视频流的二进制数据, 通过mediaSource API 转换为浏览器video 标签可播放的视频信息

1.2.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
deviceUuid	string	必须	设备deviceUuid

1.2.3. 返回数据

名称	类型	是否必须	备注
url	string	非必须	视频预览地址url:客户端通过ws连接GMP服务获取视频流的二进制数据，通过mediaSource API 转换为浏览器video标签可播放的视频信息

HTTP请求示例

请求body

```
{
  "deviceId": "c0a238fcc44abbaa1b2b7c77a7d5d4"
}
```

HTTP响应示例

响应200

```
{
    "code":0,
    "data":{"ur1":"/gmp/stream/eyJyZXNvbHV0aw9uIjoizGVmYXVsdCIiInR5cGUiOiJpb3QiLCJ1cmwiOiJ"}},
    "msg":"成功"
```

1.3. 视频抓图

1.3.1. 基本信息

Path: /v1/api/video/realtime/capture

Method: POST

接口描述: 通过此接口，传入设备的uuid，就可以抓取到该设备此时的所播放视频的截图

1.3.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
deviceUuid	string	必须	设备deviceUuid

1.3.3. 返回数据

名称	类型	是否必须	备注
captureUrl	string	非必须	抓取的图片url,url有效期默认24小时

HTTP请求示例

请求body

```
{
  "deviceUuid": "c0a2388fcc444abbaa1b2b7c77a7d5d4"
}
```

HTTP响应示例

响应200

```
{
  "code": 0,
  "data": {
    "captureUrl": "http://10.171.4.130/pub/_ZzEwMF8zbQ==_6fad71e532764dc6b32376b4c6"
  },
  "msg": "成功"
}
```

- 1. 事件记录回调
 - 1.1. 事件记录回调说明
 - 1.2. 事件记录回调数据结构
 - 1.3. 回调数据示例
 - 1.3.1. 车辆事件记录回调数据示例
 - 1.3.2. 警戒事件记录回调数据示例
 - 1.3.3. 通行记录回调数据示例
 - 1.3.4. 人员回调数据示例

1. 事件记录回调

1.1. 事件记录回调说明

在【系统管理】-【功能配置】-【事件配置】页面，填入事件订阅服务器地址后，配置事件规则-联动反馈时，选择联动事件订阅，产生的事件记录就会推送到该服务器地址。

1.2. 事件记录回调数据结构

字段	字段类型	是否必须	默认值	备注
id	string	非必须		事件记录id
trackId	string	非必须		事件记录的trackId
uuid	string	非必须		事件记录的uuid
captureImg	string	非必须		抓拍图
groupImg	string	非必须		底库图
fullImg	string	非必须		全景图
eventName	string	非必须		事件类型名
eventId	integer	非必须		事件类型id
deviceId	integer	非必须		设备id
deviceName	string	非必须		设备名
deviceUuid	string	非必须		设备uuid
zoneId	integer	非必须		区域ID
zoneUuid	String	非必须		区域uuid
zoneName	String	非必须		区域名
timestamp	integer	非必须		事件时间
eventLevelId	integer	非必须		事件等级ID: 1:高、2:中、3:低
eventLevelColor	string	非必须		事件等级颜色
eventLevelName	string	非必须		事件等级名
status	integer	非必须		事件记录状态: 0-未处理,1-已处理
nvrId	integer	非必须		nvr设备id
nvrChannelNo	integer	非必须		nvr channel 号

字段	字段类型	是否必须	默认值	备注
gmtCreate	integer	非必须		创建时间
gmtModified	integer	非必须		修改时间
ext	string	非必须		业务扩展数据

ext 车辆事件

字段	字段类型	是否必须	默认值	备注
plateNumber	String	非必须		车牌号
plateColor	String	非必须		车牌颜色code
plateColorMsg	String	非必须		车牌颜色描述
personName	String	非必须		车主姓名

ext 警戒事件

字段	字段类型	是否必须	默认值	备注
drawLineDetail	String	非必须		画线详情: firstNodeDetail --画线详情, 第一个节点详情, secondNodeDetail --第二个节点详情,双拌线并且是预警规则时存在, inTheArea --目标是否在此区域: true 表示在此区域, false 表示不在此区域
targetRect	String	非必须		目标框在全景图中的位置: top : 目标距离上方间距, left : 目标距离左方间距, bottom : 目标距离下方间距, right : 目标距离右方间距
alarmType	Integer	非必须		警戒类型, 1: 人员越界, 2: 人员入侵, 3: 车辆越界, 31: 机动车越界, 32: 非机动车越界, 33: 机动车和非机动车越界, 4: 车辆禁停, 41: 机动车禁停, 42: 非机动车禁停, 43: 机动车和非机动车禁停, 5: 人员越界-翻墙检测, 6: 人员徘徊, 7: 机动车离开, 8: 非机动车离开, 9: 机动车和非机动车离开, 10: 人员值岗/离岗-离岗, 11: 人员值岗/离岗-超员, 12: 人员值岗/离岗-少员, 13: 人员奔跑, 14: 摔倒检测, 15: 人员扭打, 16: 抽烟检测, 17: 看手机, 18: 接打电话, 19: 人员聚众, 20: 防碰撞预警, 21: 车辆逆行, 22: 睡岗
alarmControlType	Integer	非必须		警戒相机类型: 1-警戒机, 2-警戒算法仓
ruleConfig	String	非必须		布控jsonString, 其中 stayTime 表示停留时间, delayAlarmTime 表示延迟报警时间, personNumber 表示值岗离岗-人员数量, 包括人员上限和人员下限; 人员聚众, 聚众人数例如: {\"stayTime\":1, \\\"delayAlarmTime\\\":60, \\\"personNumber\\\":1}
shieldLineDetail	String	非必须		屏蔽区域画线,里面的 firstNodeDetail 代码屏蔽区域画线

ext 通行记录

字段	字段类型	是否必须	默认值	备注
personName	String	非必须		用户名

1.3. 回调数据示例

1.3.1. 车辆事件记录回调数据示例

响应 200

```
{
  "captureImg": "https://10.122.101.7/pub/_ZzEwMF9mb3JldmVyQnVja2V0_3a978ff8f86a43b",
  "deviceId": 288,
  "deviceName": "core结构化车辆",
  "eventLevelColor": "#E1E600",
  "eventLevelId": 2,
  "eventLevelName": "中",
  "eventType": 202,
  "eventName": "陌生车辆",
  "ext": {
    "plateColorMsg": "蓝色",
    "plateNumber": "桂C6R325",
    "plateColor": 1
  },
  "fullImg": "https://10.122.101.7/pub/_ZzEwMF9mb3JldmVyQnVja2V0_845129a659644d438f",
  "gmtCreate": 1625111168049,
  "gmtModified": 1625111168049,
  "id": "KgYsYHoBdNc5wFDjhEE4",
  "nvrChannelNo": 1,
  "nvrId": 374,
  "status": 0,
  "timestamp": 1625111160504,
  "uuid": "3a3f0567528547eca7b613fa7b6909a8",
  "zoneId": 1,
  "zoneName": "一级区域"
}
```

1.3.2. 警戒事件记录回调数据示例

响应 200

```

{
  "deviceId": 322,
  "deviceName": "人员入侵-人员奔跑",
  "eventLevelColor": "#D9001B",
  "eventLevelId": 1,
  "eventLevelName": "高",
  "eventTypeId": 301,
  "eventTypeName": "人员入侵",
  "ext": {
    "alarmType": 2,
    "targetRect": [{"bottom":49.641,"left":50.726166,"right":53.478657,"top":36.95},
    "drawLineDetail": [{"firstNodeDetail":[{"x":0.032423208191126277,"y":0.1820256},
    "ruleConfig": {
      "stayTime":1,
      "drawType":1
    },
    },
    "alarmControlType": 2
  },
  "fullImg": "https://10.122.101.7/pub/_ZzEwMF9mb3JldmVyQnVja2V0_88f6595f2f9246ba8c",
  "gmtCreate": 1625111313716,
  "gmtModified": 1625111313716,
  "id": "kQYuYHoBdNc5wFDjvUE3",
  "status": 0,
  "timestamp": 1625111312000,
  "uuid": "6136b525e35a4141b40286042ee8597c",
  "zoneId": 1,
  "zoneName": "一级区域"
}

```

1.3.3. 通行记录回调数据示例

响应 200

```

{
  "captureImg": "http://10.122.94.244/pub/_ZzEwMF9mb3JldmVyQnVja2V0_132e9973444141c",
  "deviceId": 351,
  "deviceName": "C3S123-视频流-B3R1012-126-通",
  "eventLevelColor": "#D9001B",
  "eventLevelId": 1,
  "eventLevelName": "高",
  "eventTypeId": 501,
  "eventTypeName": "重点人员出入事件",
  "ext": {
    "personName": "杨亮"
  },
  "gmtCreate": 1625123076976,
  "gmtModified": 1625123076976,
  "groupImg": "http://10.122.94.244/pub/_ZzEwMF9mb3JldmVyQnVja2V0_7568ee9cf70c476f8",
  "id": "8-LiYHoBG4rJ_r3IO1B0",
  "status": 0,
  "timestamp": 1625123075478,
  "uuid": "64bfa27726e44db181c22dc4e7c97ab4",
  "zoneId": 1,
  "zoneName": "一级区域"
}

```

1.3.4. 人员回调数据示例

响应 200

```
{
  "captureImg": "http://10.122.94.244/pub/_ZzEwMF9mb3JldmVyQnVja2V0_e23fff4beb154841b",
  "deviceId": 285,
  "deviceName": "C4S322-视频流-core-133-重+陌",
  "eventLevelColor": "#D9001B",
  "eventLevelId": 1,
  "eventLevelName": "高",
  "eventType": "101",
  "eventName": "重点人员",
  "ext": {
    "personName": "杨志超"
  },
  "fullImg": "http://10.122.94.244/pub/_ZzEwMF9mb3JldmVyQnVja2V0_9f48ed7c5d2d40e497fa",
  "gmtCreate": 1625123387841,
  "gmtModified": 1625123387841,
  "groupImg": "http://10.122.94.244/pub/_ZzEwMF9mb3JldmVyQnVja2V0_4fd6ba3d2c194e6a966",
  "id": "A-LmYHoBG4rJ_r3I-VzD",
  "nvrChannelNo": 5,
  "nvrId": 415,
  "status": 0,
  "timestamp": 1625123384049,
  "uuid": "430553d0038a41e6a25084b6470d96ee",
  "zoneId": 1,
  "zoneName": "一级区域"
}
```

- 1.1. 通行记录回调数据结构
 - 1.1.1. 回调数据示例

1.1. 通行记录回调数据结构

字段	字段类型	是否必须	备注
ecordType	integer	非必须	记录类型。1 - 通行; 2 - 安防
personId	long	非必须	人员id
personUuid	string	非必须	人员UUID
personName	string	非必须	人员姓名
personType	integer	非必须	人员类型。1 - 员工; 2 - 访客; 3 - 重点人员（黑名单）
personCode	string	非必须	人员编码
personPhone	string	非必须	员工手机
personIdentifyNum	string	非必须	员工身份证
personEmail	string	非必须	员工邮箱
personUniquelyIdentify	string	非必须	员工唯一标识
personExt	string	非必须	员工备注
groupDtoList	string	非必须	重点人员组的id, uuid, name
visitType	integer	非必须	访客类型。1 - 普通访客, 2 - VIP访客
visitPersonId	long	非必须	访问人员的id
visitPeronUuid	string	非必须	访客人员UUID

字段	字段类型	是否必须	备注
visitPersonType	integer	非必须	访问人员的类型。1 - 员工; 2 - 访客; 3 - 重点人员（黑名单）
visitPersonCode	string	非必须	访问人员的员工编码
visitPersonName	string	非必须	访问人员的姓名
visitReason	string	非必须	来访原因
visitFirm	string	非必须	访客单位
visitNumPlate	string	非必须	访客车牌
orgId	long	非必须	组织id
orgUuid	string	非必须	组织uuid
orgType	integer	非必须	组织类型。1 - 非顶层组织; 2 - 区域顶层组织
orgFullName	string	非必须	组织名称
orgNickName	string	非必须	组织别名
orgParentId	long	非必须	父组织id
orgParentFullName	string	非必须	组织全路径
zoneId	long	非必须	所属区域id
zoneUuid	string	非必须	所属区域uuid

字段	字段类型	是否必须	备注
zoneName	string	非必须	所属区域名称
deviceSetId	long	非必须	device set的id
deviceSetUuid	long	非必须	device set的uuid
deviceId	long	非必须	抓拍设备的id
deviceUuid	string	非必须	抓拍设备的uuid
deviceName	string	非必须	抓拍设备的名称
deviceLocation	string	非必须	抓拍设备的位置
deviceType	integer	非必须	抓拍设备的类型。1 - 智能摄像头; 2 - 智能面板机; 3 - 智能服务器; 4 - 网络摄像机; 5 - 存储服务器
deviceModel	integer	非必须	抓拍设备的型号
deviceStatus	integer	非必须	抓拍设备在抓拍时的状态。0 - 未激活; 1 - 在线未绑定; 2 - 离线未绑定; 3 - 在线; 4 - 离线; 5 - 解绑中
recognitionType	integer	非必须	识别人员类型。1 - 员工; 2 - 访客; 3 - 重点人员（黑名单）; 4 - 陌生人; 5 - 未识别; 6 - VIP访客
verificationMode	integer	非必须	验证类型。0 - 人脸; 1 - 人脸或刷卡; 2 - 人脸及刷卡; 3 - 人脸及密码; 4 - 远程开门; 5 - 二维码
vistorRecordType	integer	非必须	验证类型。0 - 人脸; 1 - 人脸或刷卡; 2 - 人脸及刷卡; 3 - 人脸及密码

字段	字段类型	是否必须	备注
passType	integer	非必须	通行类型。0 - 未通行; 1 - 通行; 2 - 复合认证未通过; 3 - 未通行（非活体攻击）; 4 - 未通行（高温）; 5 - 未通行（未佩戴口罩） 6 - 未通行（健康码检测未通过）
vistorType	integer	非必须	通行类型。0 - 未通行; 1 - 通行;
livenessType	string	非必须	活体结果。0 - 非活体; 1 - 活体; 2 - 未知
recognitionScore	number	非必须	识别分数
livenessScore	number	非必须	活体分数
passTime	integer	非必须	抓拍时间。ISO 8601的精确到秒的格式化时间。形如 2020-04-15T17:16:20+08:00
playbackSupport	boolean	非必须	是否支持录像回放
nvrUuid	string	非必须	录像回放NVR id
channelNo	integer	非必须	录像回放channel number
personImageUrl	string	非必须	人员底库图片url
snapshotUrl	string	非必须	抓拍图url
passTimestamp	integer	非必须	抓拍时间。unix时间戳，毫秒
birthdayString	string	非必须	生日祝福
entryTimeCount	string	非必须	入职周年祝福

字段	字段类型	是否必须	备注
healthyState	integer	非必须	健康码状态 0绿码、1黄码、2红码、3未知、4超时
healthkitPasstype	integer	非必须	0、未通过 1、通过 2、未使用
temperature	number	非必须	人体温度
highThreshold	number	非必须	高温阈值
maskType	integer	非必须	口罩状态，0：未知，1：戴口罩，2：未戴口罩
temperatureType	integer	非必须	人体高温状态，0：未检测，1：正常，2：高温

1.1.1. 回调数据示例

响应 200

```

{
  "deviceId":281,
  "deviceLocation":"dd",
  "deviceModel":21,
  "deviceName":"core人脸识别",
  "deviceSetId":111,
  "deviceSetUuid":"aeb1b26388b444429fc4868fb4d4b7a6",
  "deviceStatus":3,
  "deviceType":4,
  "deviceUuid":"9a5abfbcd8348c4a34975fd3b25e2a5",
  "groupDtoList":[{"groupName\\":"\\员工默认组\\",\\"groupUuid\\":"\\976055aae31445c5a583
  "highTemperatureOff":false,
  "highThreshold":37.2,
  "id":"66806a6bbe5944b5bcf19f5a2039b3a0",
  "isAttendance":true,
  "livenessScore":94.84798,
  "livenessType":1,
  "maskType":0,
  "maskTypeDesc":"未检测",
  "orgFullName":"一级区域",
  "orgId":1,
  "orgParentId":0,
  "orgType":2,
  "orgUuid":"7ca8da7d8f424fabb891c99e20251404",
  "passTime":"2021-06-30T11:28:27.362+08:00",
  "passTimeFormat":"2021-06-30 11:28:27",
  "passTimestamp":1625023707362,
  "passType":1,
  "passTypeDesc":"通行",
  "personCode":"69713819",
  "personExt":"","
  "personId":275,
  "personImageUri":"_ZzEwMF9mb3JldmVyQnVja2V0_aa5faa19712348b7944269a4df9f53c8",
  "personImageUri1":"https://10.122.101.7/pub/_ZzEwMF9mb3JldmVyQnVja2V0_aa5faa1971234
  "personName":"py",
  "personType":1,
  "personUuid":"86ccd9d26a1e4ec2905f756d8e58daa8",
  "playbackSupport":false,
  "recognitionScore":92.662994,
  "recognitionType":1,
  "recognitionTypeDesc":"员工",
  "recordSource":1,
  "recordType":1,
  "snapshotUri":"_ZzEwMF82bQ==_03bfe3fdc3644b6a902532f3a5c06a9d",
  "snapshotUri1":"https://10.122.101.7/pub/_ZzEwMF82bQ==_03bfe3fdc3644b6a902532f3a5c0
  "temperature":"0.0",
  "temperatureType":0,
  "temperatureTypeDesc":"未检测",
  "verificationMode":0,
  "verificationModeDesc":"人脸",
  "vistorType":0,
  "zoneId":1,
  "zoneName":"一级区域",
  "zoneUuid":"7ca8da7d8f424fabb891c99e20251404"
}

```


- o 1.1. 考勤记录回调数据结构
 - 1.1.1. 回调数据示例

1.1. 考勤记录回调数据结构

名称	类型	是否必须	备注
uuid	string	非必须	考勤记录uuid
personName	string	非必须	员工姓名
personCode	string	非必须	员工编号
attendanceDate	string	非必须	考勤日期，时间戳
attTimeResult	object []	非必须	考勤时段结果：打卡时间&考勤结果
└─ order	integer	非必须	时段编号：1-时段一，2-时段二，3-时段三
└─ inPunch	string	非必须	上班打卡时间eg:09:00:00，需要精确到秒
└─ outPunch	string	非必须	下班打卡时间eg:11:00:00，需要精确到秒
└─ result	integer	非必须	考勤结果：0-暂无结果（'-'表示）、1-正常、2-缺勤、3-迟到、4-早退、5-迟到早退、6-漏打卡
status	integer	非必须	考勤状态，注：：0-暂无结果（'-'表示），1-正常，2-旷工，3-出勤调整，4-异常

1.1.1. 回调数据示例

响应 200

```
{
  "attTimeResult": [{
    "adjust": -1,
    "inPunch": "09:00:00",
    "order": 1,
    "outPunch": "10:00:00",
    "period": 2,
    "result": 4
  }
],
  "attendanceDate": 1624896000000,
  "personCode": "5677978",
  "personName": "彭斯亮",
  "personUuid": "2a12603fc2db4ebebe6dd8d6004e0c93",
  "status": 4,
  "uuid": "26862b1abeb64c4d869add1d225e5ebe"
}
```

- 1. 业务数据订阅
 - 1.1. 业务数据记录订阅
 - 1.1.1. 基本信息
 - 1.1.2. 请求参数
 - 1.1.3. 返回数据
 - 1.1.4. HTTP请求示例
 - 1.1.5. HTTP响应示例
 - 1.2. 查询业务数据记录订阅
 - 1.2.1. 基本信息
 - 1.2.2. 请求参数
 - 1.2.3. 返回数据
 - 1.2.4. HTTP请求示例
 - 1.2.5. HTTP响应示例
 - 1.3. 解除业务数据记录订阅
 - 1.3.1. 基本信息
 - 1.3.2. 请求参数
 - 1.3.3. 返回数据
 - 1.3.4. HTTP请求示例
 - 1.3.5. HTTP响应示例

1. 业务数据订阅

1.1. 业务数据记录订阅

1.1.1. 基本信息

Path: /v1/api/pubsub/busi/record/add

Method: POST

接口描述: 第三方订阅不同类型的业务数据

1.1.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
pushUrl	string	必须	用户接受数据的url，产生的业务数据会推送到这个地址
busiTypes	integer []	必须	要订阅的业务类型，可多选:1-安防识别记录;2-车辆识别记录;3-警戒记录;5-通行记录;12-结构化全属性识别记录

1.1.3. 返回数据

OK

1.1.4. HTTP请求示例

请求path

```
/v1/api/pubsub/busi/record/add
```

请求body

```
{
  "busiTypes": [
    2,3,5
  ],
  "pushUrl": "http://10.170.160.136:8989/signin"
}
```

1.1.5. HTTP响应示例

响应 200

```
{
  "code": 0,
  "msg": "成功"
}
```

1.2. 查询业务数据记录订阅

1.2.1. 基本信息

Path: /v1/api/pubsub/busi/record/query

Method: POST

接口描述: 查询订阅的信息

1.2.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
appKey	string	非必须	用户appKey，优先从请求头取当前请求携带的cappKey，如果请求头没有，则必传

1.2.3. 返回数据

名称	类型	是否必须	备注
appKey	string	非必须	用户appKey
pushUrl	string	非必须	用户接受数据的url，产生的业务数据会推送到这个地址
busiTypes	integer []	非必须	要订阅的业务类型:1-安防识别记录;2-车辆识别记录;3-警戒记录;5-通行记录;12-结构化全属性识别记录

1.2.4. HTTP请求示例

请求path

```
/v1/api/pubsub/busi/record/query
```

请求body

```
{
  "appKey": "sdfajk3242324fa!d"
}
```

1.2.5. HTTP响应示例

响应 200

```
{
  "code": 0,
  "data": {
    "appKey": "sdfajk3242324fa!d",
    "busiTypes": [
      2,
      3,
      5
    ],
    "pushUrl": "http://10.170.160.136:8989/signin"
  },
  "msg": "成功"
}
```

1.3. 解除业务数据记录订阅

1.3.1. 基本信息

Path: /v1/api/pubsub/busi/record/unbind

Method: POST

接口描述: 解除订阅

1.3.2. 请求参数

Headers

参数名称	参数值	是否必须	示例	备注
Content-Type	application/json	是		

Body

名称	类型	是否必须	备注
appKey	string	非必须	用户appKey, 优先从请求头取当前请求携带的cappKey, 如果请求头没有, 则必传

1.3.3. 返回数据

OK

1.3.4. HTTP请求示例

请求path

/v1/api/pubsub/busi/record/unbind

请求body

```
{  
  "appKey": "sdfajk3242324fa!d"  
}
```

1.3.5. HTTP响应示例

响应 200

```
{  
  "code": 0,  
  "msg": "成功"  
}
```

- 1. 业务推送数据结构
 - 1.1. 通行识别记录推送
 - 1.1.1. 回调数据示例

1. 业务推送数据结构

字段	字段类型	是否必须	备注
busiType	string	非必须	业务类型: 1-安防人员识别记录;2-车辆识别记录;3-警戒数据记录; 5-通行记录;12-结构化全属性识别记录
data	object	非必须	对应业务类型的数据，不同的业务类型产生的数据各不相同，下面会一一说明

1.1. 通行识别记录推送

当busiType=5为通行识别记录推送

字段	字段类型	是否必须	备注
ecordType	integer	非必须	记录类型。1 - 通行; 2 - 安防
personUuid	string	非必须	人员UUID
personName	string	非必须	人员姓名
personType	integer	非必须	人员类型。1 - 员工; 2 - 访客; 3 - 重点人员（黑名单）
personCode	string	非必须	人员编码
visitType	integer	非必须	访客类型。1 - 普通访客, 2 - VIP 访客
visitPeronUuid	string	非必须	访客人员UUID
visitPersonType	integer	非必须	访问人员的类型。1 - 员工; 2 - 访客; 3 - 重点人员（黑名单）
visitPersonCode	string	非必须	问人员的员工编码
visitPersonName	string	非必须	访问人员的姓名
visitReason	string	非必须	来访原因
visitFirm	string	非必须	访客单位
deviceUuid	string	非必须	抓拍设备的uuid
deviceName	string	非必须	抓拍设备的名称
deviceLocation	string	非必须	抓拍设备的位置

字段	字段类型	是否必须	备注
deviceType	integer	非必须	抓拍设备的类型。1 - 智能摄像头; 2 - 智能面板机; 3 - 智能服务器; 4 - 网络摄像机; 5 - 存储服务器
deviceModel	integer	非必须	抓拍设备的型号
deviceStatus	integer	非必须	抓拍设备在抓拍时的状态。0 - 未激活; 1 - 在线未绑定; 2 - 离线未绑定; 3 - 在线; 4 - 离线; 5 - 解绑中
recognitionType	integer	非必须	识别人员类型。1 - 员工; 2 - 访客; 3 - 重点人员（黑名单）; 4 - 陌生人; 5 - 未识别; 6 - VIP访客
verificationMode	integer	非必须	验证类型。0 - 人脸; 1 - 人脸或刷卡; 2 - 人脸及刷卡; 3 - 人脸及密码; 4 - 远程开门; 5 - 二维码
vistorRecordType	integer	非必须	验证类型。0 - 人脸; 1 - 人脸或刷卡; 2 - 人脸及刷卡; 3 - 人脸及密码
passType	integer	非必须	通行类型。0 - 未通行; 1 - 通行; 2 - 复合认证未通过; 3 - 未通行（非活体攻击）; 4 - 未通行（高温）; 5 - 未通行（未佩戴口罩）; 6 - 未通行（健康码检测未通过）
vistorType	integer	非必须	通行类型。0 - 未通行; 1 - 通行;
livenessType	string	非必须	活体结果。0 - 非活体; 1 - 活体; 2 - 未知
recognitionScore	number	非必须	识别分数
livenessScore	number	非必须	活体分数
passTime	integer	非必须	抓拍时间。ISO 8601的精确到秒的格式化时间。形如2020-04-15T17:16:20+08:00
playbackSupport	boolean	非必须	是否支持录像回放
nvrUuid	string	非必须	录像回放NVR id

字段	字段类型	是否必须	备注
channelNo	integer	非必须	录像回放channel number
personImageUrl	string	非必须	人员底库图片url
snapshotUrl	string	非必须	抓拍图url
passTimestamp	integer	非必须	抓拍时间。unix时间戳，毫秒
birthdayString	string	非必须	生日祝福
entryTimeCount	string	非必须	入职周年祝福
healthyState	integer	非必须	健康码状态 0绿码、1黄码、2红码、3未知、4超时
healthkitPasstype	integer	非必填	0、未通过 1、通过 2、未使用

1.1.1. 回调数据示例

响应 200

```
{
  "captureImg": "http: //10.122.94.244/pub/_ZzEwMF9mb3JldmVyQnVja2V0_132e9973444141c",
  "deviceId": 351,
  "deviceName": "C3S123-视频流-B3R1012-126-通",
  "eventLevelColor": "#D9001B",
  "eventLevelId": 1,
  "eventLevelName": "高",
  "eventTypeId": 501,
  "eventTypeName": "重点人员出入事件",
  "ext": {
    "personName": "杨亮"
  },
  "gmtCreate": 1625123076976,
  "gmtModified": 1625123076976,
  "groupImg": "http: //10.122.94.244/pub/_ZzEwMF9mb3JldmVyQnVja2V0_7568ee9cf70c476f8",
  "id": "8-LiYHoBG4rJ_r3IO1B0",
  "status": 0,
  "timestamp": 1625123075478,
  "uuid": "64bfa27726e44db181c22dc4e7c97ab4",
  "zoneId": 1,
  "zoneName": "一级区域"
}
```

- o 1.1. 安防识别记录推送

1.1. 安防识别记录推送

字段	字段类型	是否必须	备注
personUuid	string	非必须	人员UUID
personName	string	非必须	人员名称
groupImageUrl	string	非必须	底库图
captureImageUrl	string	非必须	抓拍图
deviceUuid	string	非必须	设备uuid
deviceName	string	非必须	设备名
captureTime	long	非必须	抓拍时间
location	string	非必须	设备位置
score	double	非必须	识别分数
frameUrl	string	非必须	全景图
personType	string	非必须	识别人员类型：员工、访客、重点人员、全局陌生人、通道陌生人、高频重点人员

示例：

◀ [REDACTED] ▶

- 1.1. 结构化记录推送
 - 1.1.1. category=1 人脸结构化数据
 - 1.1.2. category=2 人体结构化数据
 - 1.1.3. category=3 机动车结构化数据
 - 1.1.4. category=4 非机动车结构化数据

1.1. 结构化记录推送

1.1.1. category=1 人脸结构化数据

字段	字段类型	是否必须	备注
age	Ingeter	非必须	年龄段 0:未知 1:儿童 2:少年 3:青年 4:中年 5:中老年 6:老年
beardStatus	Integer	非必须	胡子状态 0: 未知; 1:无胡子 2:有胡子
category	Integer	非必须	结构化的类别 1: 人脸 2: 人体 3: 机动车 4: 非机动车
cropRectBottom	Integer	非必须	目标在抓拍图中的底部边距
cropRectLeft	Integer	非必须	目标在抓拍图中的左部边距
cropRectRight	Integer	非必须	目标在抓拍图中的右部边距
cropRectTop	Integer	非必须	目标在抓拍图中的顶部边距
cropUri	String	非必须	抓拍图
dataFromCore	Boolean	非必须	判断数据是否来自Core true 是 false 否（已废弃）
deviceLocation	String	非必须	设备所在位置
deviceName	String	非必须	设备名
deviceUuld	String	非必须	设备Uuld
fullRectBottom	Integer	非必须	目标在全景图图中的边距，包括上下左右四个距离
fullRectLeft	Integer	非必须	目标在全景图图中的边距，包括上下左右四个距离

字段	字段类型	是否必须	备注
fullRectRight	Integer	非必须	目标在全景图图中的边距，包括上下左右四个距离 右
fullRectTop	Integer	非必须	目标在全景图图中的边距，包括上下左右四个距离
fullUri	String	非必须	全景图
gender	Integer	非必须	性别 0:未知; 1:男; 2:女
glassesStatus	Integer	非必须	眼镜状态 0: 未知 1: 无眼镜 2: 有眼镜
hairStatus	Integer	非必须	头发状态 0:未知 1:长发 2:短发 3: 少量头发（秃顶）4:光头
maskStatus	Integer	非必须	口罩状态 0: 未知 1: 无 2: 有
noseOcclusion	Integer	非必须	鼻子遮挡 0 未知 1 未遮挡 2 有遮挡
recognitionTime	Long	非必须	识别时间
ruleType	Integer	非必须	布控规则类型 1: 结构化布控 2: 车辆布控 此场景下该值为1
serverTime	Long	非必须	服务器时间（已废弃）

示例：

```
{
  "deviceUuid": "620ce82d023e4fb09984f152598b18ab",
  "cropRectLeft": 61,
  "gender": 1,
  "recognitionTime": 1621588692991,
  "fullRectRight": 122,
  "noseOcclusion": 0,
  "cropRectBottom": 126,
  "deviceName": "结构化人",
  "fullRectTop": 401,
  "fullRectBottom": 126,
  "ruleType": 1,
  "beardStatus": 1,
  "fullUri": "[http://10.122.101.86:12029/pub/_ZzEwMF8zbQ==_eb42ece7e894443eba0c",
  "cropUri": "[http://10.122.101.86:12029/pub/_ZzEwMF8zbQ==_eb42ece7e894443eba0c",
  "cropRectRight": 122,
  "hairStatus": 2,
  "fullRectLeft": 757,
  "cropRectTop": 63,
  "maskStatus": 1,
  "category": 1,
  "age": 4,
  "glassesStatus": 2,
  "deviceLocation": "结构化人"
}
```

1.1.2. category=2 人体结构化数据

字段	字段类型	是否必须	备注
age	Integer	非必须	年龄段 0:未知 1:儿童 2:少年 3:青年 4:中年 5:中老年 6:老年
bagType	Integer	非必须	是否带包 0: 未知 1: 未携带包 2: 双肩包 3: 手提包 4: 单肩包 5: 拉杆箱 6: 腰包 7: 有背包, 类型未知
category	Integer	非必须	结构化的类别1: 人脸 2: 人体 3: 机动车 4: 非机动车
cropRectBottom	Integer	非必须	目标在抓拍图中的底部边距
cropRectLeft	Integer	非必须	目标在抓拍图中的左边距
cropRectRight	Integer	非必须	目标在抓拍图中的右边距
cropRectTop	Integer	非必须	目标在抓拍图中的顶部边距
cropUri	String	非必须	抓拍图
dataFromCore	Boolean	非必须	判断数据是否来自Core true 是 false 否 (已废弃)
deviceLocation	String	非必须	设备所在位置
deviceName	String	非必须	设备名
deviceUuld	String	非必须	设备Uuld
dressLowerColor	Integer	非必须	下衣颜色, 同帽子颜色
dressLowerStyle	Integer	非必须	下衣颜色 0: 未知 1: 长裤 2: 短裤 3: 长裙 4: 短裙
dressUpperColor	Integer	非必须	上衣颜色 同帽子颜色

字段	字段类型	是否必须	备注
dressUpperStyle	Integer	非必须	上衣尺寸 0: 未知 1: 长袖 2: 短袖
fallStatus	Integer	非必须	跌倒状态 0: 未知 1: 未跌倒 2: 跌倒
fullRectBottom	Integer	非必须	目标在全景图图中的底部边距
fullRectLeft	Integer	非必须	目标在全景图图中的左边距
fullRectRight	Integer	非必须	目标在全景图图中的右边距
fullRectTop	Integer	非必须	目标在全景图图中的顶部边距
fullUri	String	非必须	全景图
gender	Integer	非必须	性别 0: 未知 1: 男性 2: 女性
hatColor	Integer	非必须	帽子颜色 0: 未知 1: 白色 2: 灰色 3: 棕色 4: 红色 5: 蓝色 6: 黄色 7: 绿色 8: 粉色 9: 橙色 10: 青色 11: 紫色 12: 浅蓝色 13: 深蓝色 14: 黑色 15: 彩色
recognitionTime	Long	非必须	识别时间，毫秒时间戳
rideBike	Integer	非必须	是非必须骑车 0: 未知 1: 未骑车 2: 骑车
ruleType	Integer	非必须	布控规则类型 1: 结构化布控 2: 车辆布控 此场景下该值为1
runStatus	Integer	非必须	奔跑状态 0: 未知 1: 未奔跑 2: 奔跑
serverTime	long	非必须	服务器时间（已废弃）

字段	字段类型	是否必须	备注
smokeStatus	Integer	非必须	抽烟状态 0: 未知 1: 未抽烟 2: 抽烟
usePhoneStatus	Integer	非必须	打电话状态 0:未知 1:未打电话 2:打电话
watchPhoneStatus	Integer	非必须	看手机状态 0:未知 1:未看手机 2:看手机
wearHat	Integer	非必须	是否带帽子 0: 未知 1: 无帽子 2: 有帽子
wearSafetycap	Integer	非必须	是否带安全帽 0: 未知 1: 无安全帽 2: 有安全帽
umbrellaStatus	Integer	非必须	打伞状态 0: 未知 1: 不打伞 2: 打伞

```
{
  "cropRectLeft":14,
  "hatColor":0,
  "recognitionTime":1621232788253,
  "fullRectRight":97,
  "deviceName":"结构化车2",
  "fullRectTop":206,
  "wearSafetycap":1,
  "smokeStatus":1,
  "fullRectBottom":186,
  "bagType":1,
  "fullRectLeft":779,
  "dressUpperColor":14,
  "cropRectTop":27,
  "watchPhoneStatus":1,
  "deviceLocation":"结构化车2",
  "deviceUuId":"5e5176fc11ee4ad6bd4675dd2376960d",
  "gender":1,
  "dressLowerStyle":1,
  "cropRectBottom":186,
  "fallStatus":1,
  "ruleType":1,
  "fullUri":"[http://10\122\101\86:12029/pub/_ZzEwMF8zbQ==\_eb42ece7e894443eba0c
  "cropUri":"[http://10\122\101\86:12029/pub/_ZzEwMF8zbQ==\_eb42ece7e894443eba0c
  "runStatus":1,
  "dressLowerColor":14,
  "usePhoneStatus":1,
  "cropRectRight":97,
  "wearHat":1,
  "dressUpperStyle":1,
  "category":2,
  "age":4,
  "rideBike":2
}
```

1.1.3. category=3 机动车结构化数据

字段	字段类型	是否必须	备注
deviceUuld	String	非必须	设备Uuld
deviceLocation	String	非必须	设备所在位置
deviceName	String	非必须	设备名
cropUri	String	非必须	抓拍图
cropRectLeft	Integer	非必须	目标在抓拍图中的左边距
cropRectBottom	Integer	非必须	目标在抓拍图中的底部边距
cropRectRight	Integer	非必须	目标在抓拍图中的右边距
cropRectTop	Integer	非必须	目标在抓拍图中的顶部边距
fullUri	String	非必须	全景图
fullRectRight	Integer	非必须	目标在全景图图中的右边距
fullRectTop	Integer	非必须	目标在全景图图中的顶部边距
fullRectLeft	Integer	非必须	目标在全景图图中的左边距
fullRectBottom	Integer	非必须	目标在全景图图中的底部边距
recognitionTime	Long	非必须	识别时间
serverTime	Long	非必须	服务器时间（已废弃）

字段	字段类型	是否必须	备注
dataFromCore	Boolean	非必须	判断数据是否来自Core true 是 1
ruleType	Integer	非必须	布控规则类型 1: 结构化布控 2: 为1
category	Integer	非必须	结构化的类别 1: 人脸 2: 人体
plateColor	Integer	非必须	车牌颜色 0: 未知 1: 蓝色 2: 黄绿色 6: 渐变绿底黑字 7: 黄绿
vehicleFunctional	Integer	非必须	功能用车 0: 未知 1: 挖掘机 2: 车 5: 警车 6: 集装箱车
vehicleType	Integer	非必须	车辆类型 0: 未知 1: 两厢轿车 轿车 5: 微型轿车 6: MPV 7: 9 型客车 10: 面包车 11: 微型面 货车 14: 油罐车 15: 吊车 16: 皮卡 19: 微卡 20: 两轮车 21: 车 30: 小型车(大范围) 31: 大型 101: 电动车 102: 摩托车 机动 1,2,3,4,5,6,7,8,9,10,11,12,13,14
vehicleTypeplateType	Integer	否	车牌类型 0: 未知 1: 普通小(色) 2: 普通大车、卡车牌 (黄 4: 006车牌 5: 港澳车牌 6: 双 馆牌 9: 电动自行车车牌 10: 叉 12: 摩托车牌 13: 临时车牌 14 挂车车牌 16: 新能源牌 17: 运 练车 20: 政府车 21: 出租车

字段	字段类型	是否必须	备注
vehicleBrand	Integer	否	0=其它, 1=大众, 2=奥迪, 3=丰田, 4=本田, 5=日产, 6=福特, 7=奇瑞, 8=别克, 9=长城, 10=吉利, 11=比亚迪, 12=华晨, 13=宝马, 14=雪铁龙, 15=起亚, 16=现代, 17=江淮, 18=比亚迪, 20=标致, 21=长安, 22=奇瑞, 23=吉利, 24=斯柯达, 25=海马, 26=夏利, 27=奇瑞, 28=奇瑞, 29=奇瑞, 30=一汽, 31=宝骏, 32=帝豪, 33=吉利, 34=吉利, 35=皇冠(丰田), 36=金杯, 37=吉利, 38=吉利, 39=吉利, 40=英伦(吉利), 41=吉利全球鹰, 42=吉利, 43=长丰, 44=北汽威旺, 45=北京汽车, 46=北汽威旺, 47=北汽威旺, 48=玛莎拉蒂, 49=罗孚, 50=捷豹, 51=捷豹, 52=捷豹, 53=捷豹, 54=捷豹, 55=捷豹, 56=捷豹, 57=阿尔法, 58=兰博基尼, 59=兰博基尼, 60=兰博基尼, 61=法拉利汽车, 62=昌河, 63=菲亚特, 64=菲亚特, 65=菲亚特, 66=莲花汽车, 67=华普汽车, 68=奔腾(一汽), 69=奔腾(一汽), 70=奔腾(一汽), 71=威麟汽车, 72=众泰, 73=众泰, 74=JEEP(吉普), 75=中兴, 76=中兴, 77=中兴, 78=中兴, 79=雷诺, 80=欧宝, 81=野马, 82=福特, 83=福特, 84=福特, 85=斯巴鲁, 86=阿斯顿·马丁, 87=阿斯顿·马丁, 88=阿斯顿·马丁, 89=宾利, 90=福特, 91=福特, 92=福特, 93=观致, 94=广汽, 95=双龙, 96=双龙, 97=双龙, 98=双龙, 99=黄海, 100=九龙客车, 101=江淮, 102=江淮, 103=陆风, 104=纳智捷, 105=欧尚, 106=欧尚, 107=欧尚, 108=豪沃, 109=上海汇众, 110=上海汇众, 111=上海汇众, 112=江淮, 113=JMC(江铃), 114=江铃, 115=江铃, 116=江铃, 117=陕汽, 118=少林, 119=陕汽, 120=陕汽, 121=跃进, 122=宇通, 123=宇通, 124=宇通, 125=宇通, 126=宇通, 127=宇通, 128=北奔重卡, 129=北奔重卡, 130=北奔重卡, 131=北奔重卡, 132=腾势, 133=腾势, 134=英致, 135=英致, 136=英致, 137=大宇客车, 138=长安谛艾仕, 139=长安谛艾仕, 140=长安谛艾仕, 141=长安谛艾仕, 142=凯马
vehicleColor	Integer	否	车身颜色 0:白色; 1:红色; 2:黄色; 3:蓝色; 4:绿色; 5:黑色; 6:灰色; 7:白色; 8:白色; 9:黑色; 10:黑色; 11:黑色; 12:黑色; 13:灰色; 14:灰色; 15:蓝色; 16:蓝色; 17:蓝色; 18:蓝色; 19:蓝色; 20:蓝色; 21:青色; 22:青色; 23:青色; 24:青色; 25:青色; 26:青色; 27:青色; 28:青色; 29:青色; 30:青色; 31:青色; 32:青色; 33:青色; 34:青色; 35:青色; 36:青色; 37:青色; 38:青色; 39:青色; 40:青色; 41:青色; 42:青色; 43:青色; 44:青色; 45:青色; 46:青色; 47:青色; 48:青色; 49:青色; 50:青色; 51:青色; 52:青色; 53:青色; 54:青色; 55:青色; 56:青色; 57:青色; 58:青色; 59:青色; 60:青色; 61:青色; 62:青色; 63:青色; 64:青色; 65:青色; 66:青色; 67:青色; 68:青色; 69:青色; 70:青色; 71:青色; 72:青色; 73:青色; 74:青色; 75:青色; 76:青色; 77:青色; 78:青色; 79:青色; 80:青色; 81:青色; 82:青色; 83:青色; 84:青色; 85:青色; 86:青色; 87:青色; 88:青色; 89:青色; 90:青色; 91:青色; 92:青色; 93:青色; 94:青色; 95:青色; 96:青色; 97:青色; 98:青色; 99:未知;
plateNumber	String	否	车牌号码

```

{
  "cropRectLeft":44,
  "recognitionTime":1621232785854,
  "fullRectRight":297,
  "deviceName":"结构化车2",
  "fullRectTop":109,
  "plateColor":1,
  "vehicleFunctional":0,
  "fullRectBottom":294,
  "vehicleType":2,
  "plateType":1,
  "fullRectLeft":1133,
  "cropRectTop":44,
  "deviceLocation":"结构化车2",
  "deviceUuId":"5e5176fc11ee4ad6bd4675dd2376960d",
  "cropRectBottom":294,
  "vehicleBrand":3,
  "ruleType":1,
  "fullUri":"[http://10\122\101\86:12029/pub/_ZzEwMF8zbQ==\_eb42ece7e894443eba0c
  "cropUri":"[http://10\122\101\86:12029/pub/_ZzEwMF8zbQ==\_eb42ece7e894443eba0c
  "vehicleColor":0,
  "cropRectRight":297,
  "plateNumber":"桂CSV865",
  "category":3
}

```

1.1.4. category=4 非机动车结构化数据

字段	字段类型	是否必须	
deviceUuld	String	非必须	设备Uuld
deviceLocation	String	非必须	设备所在位置
deviceName	String	非必须	设备名
cropUri	String	非必须	抓拍图
cropRectLeft	Integer	非必须	目标在抓拍图中的左边距
cropRectBottom	Integer	非必须	目标在抓拍图中的底部边距
cropRectRight	Integer	非必须	目标在抓拍图中的右边距
cropRectTop	Integer	非必须	目标在抓拍图中的顶部边距
fullUri	String	非必须	全景图
fullRectRight	Integer	非必须	目标在全景图图中的右边距
fullRectTop	Integer	非必须	目标在全景图图中的顶部边距
fullRectLeft	Integer	非必须	目标在全景图图中的左边距
fullRectBottom	Integer	非必须	目标在全景图图中的底部边距
recognitionTime	Long	非必须	识别时间
serverTime	Long	非必须	服务器时间（已废弃）

字段	字段类型	是否必须	
dataFromCore	Boolean	非必须	判断数据是否来自Core true 是 false 否（已废弃）
ruleType	Integer	非必须	布控规则类型 1：结构化布控 2：车辆布控 此场景下该值为1
category	Integer	非必须	结构化的类别 1：人脸 2：人体 3：机动车 4：非机动车
vehicleType	Integer	非必须	车辆类型 0：未知 1：两厢轿车 2：轿车 3：轿跑 4：小型轿车 5：微型轿车 6：MPV 7：SUV 8：大型客车 9：中型客车 10：面包车 11：微型面包车 12：货车 13：中型货车 14：油罐车 15：吊车 16：渣土车 17：小货车 18：皮卡 19：微卡 20：两轮车 21：三轮车 22：行人 29：挂车 30：小型车(大范围) 31：大型车(大范围) 100：自行车 101：电动车 102：摩托车。非机动车对应：20、21、100、101、102

```
{
  "cropRectLeft":18,
  "recognitionTime":1621232788453,
  "fullRectRight":121,
  "deviceName":"结构化车2",
  "fullRectTop":287,
  "fullRectBottom":209,
  "vehicleType":102,
  "fullRectLeft":763,
  "cropRectTop":31,
  "deviceLocation":"结构化车2",
  "deviceUuId":"5e5176fc11ee4ad6bd4675dd2376960d",
  "cropRectBottom":209,
  "ruleType":1,
  "fullUri":"[http://10\122\101\86:12029/pub/_ZzEwMF8zbQ==\_eb42ece7e894443eba0c
  "cropUri":"[http://10\122\101\86:12029/pub/_ZzEwMF8zbQ==\_eb42ece7e894443eba0c
  "cropRectRight":121,
  "category":4
}
```

- 1.1. 车辆记录推送

1.1. 车辆记录推送

字段	字段类型	是否必须	备注
recognitionTime	long	非必须	识别时间，毫秒级时间戳
deviceName	string	非必须	识别的设备名字
basicsVehicleType	integer	非必须	车辆类型，1-小型车，2-大型车，3-摩托车，4-其他车
basicVehicleUuld	string	非必须	基础车辆的uuid，若识别的车辆在系统添加过，则会返回这个uuid
basicsPlateType	integer	非必须	车牌类型，1-标准民用与军用车，2-006车，3-警车，4-军用车，5-使馆车，6-农用车，7-摩托车，8-新能源车
basicsVehicleColor	string	非必须	车辆颜色
basicsPlateColor	string	非必须	车牌颜色
uniqueIdentify	string	非必须	用户主动上传的唯一标识
mapId	long	非必须	设备所在的地图id，若设备上图则会有此id（已废弃）
xoffset	double	非必须	设备所在地图的x坐标（已废弃）
yoffset	double	非必须	设备所在地图的y坐标（已废弃）
personId	long	非必须	人员id(已废弃)
deviceLocation	string	非必须	抓拍设备所在位置
deviceUuld	string	非必须	抓拍设备Uuid
vehicleIdentityType	integer	非必须	车辆身份 1-重点车辆；2-员工车辆；3-访客车辆；4-陌生车辆；5-已知车辆；

字段	字段类型	是否必须	备注
personTelephone	string	非必须	车主电话号码
fullUri	string	非必须	图片全景url
personType	integer	非必须	车主身份类型 1-员工， 2- 访客， 3- 重点人员
cropUri	string	非必须	抓拍图url
basicVehicleId	long	非必须	基础车辆信息id（已废弃）
plateNumber	string	非必须	车牌号码
personName	string	非必须	车主姓名
describe	string	非必须	描述说明

示例：

```
{
  "recognitionTime":1621582175463,
  "deviceName":"结构化车1",
  "basicsVehicleType":1,
  "basicVehicleUuId":"89496a7bb9004c83a595f253c67cdfc6",
  "basicsPlateType":3,
  "basicsVehicleColor":"红色",
  "basicsPlateColor":"黑色",
  "uniqueIdentify":"1111111111",
  "personId":276,
  "deviceLocation":"结构化车1",
  "deviceUuId":"3a4bddc14c80407183767a7595c45e82",
  "vehicleIdentityType":2,
  "personTelephone":"1111111111",
  "fullUri":"http://10.122.101.86:12029/pub/_ZzEwMF8zbQ==_eb42ece7e894443eba0d8fb6fe",
  "personType":1,
  "cropUri":"[http://10.122.101.86:12029/pub/_ZzEwMF8zbQ==_eb42ece7e894443eba0c",
  "basicVehicleId":136,
  "plateNumber":"湘HX144学",
  "personName":"lixiang",
  "describe":"啥啥啥"
}
```

- 1.1. 警戒记录推送

1.1. 警戒记录推送

字段	字段类型	是否必须	备注
alarmRecordUuld	string	非必须	警戒记录id
alarmControlType	integer	非必须	警戒布控相机类型：1-警戒机，2-警戒算法仓
wholeImageUrl	string	非必须	全景图url
alarmType	integer	非必须	警戒类型，警戒类型，1：人员越界，2：人员入侵，3：车辆越界，31：机动车越界，32：非机动车越界，33：机动车和非机动车越界，4：车辆禁停，41：机动车禁停，42：非机动车禁停，43：机动车和非机动车禁停，5：人员越界-翻墙检测，6：人员徘徊，7：机动车离开，8：非机动车离开，9：机动车和非机动车离开，10：人员值岗/离岗-离岗，11：人员值岗/离岗-超员，12：人员值岗/离岗-少员，13：人员奔跑，14：摔倒检测，15：人员扭打，16：抽烟检测，17：看手机，18：接打电话，19：人员聚众，20：防碰撞预警，21：车辆逆行，22：睡岗
deviceUuid	string	非必须	设备Uuid
deviceName	string	非必须	设备名
drawLineDetail	string	非必须	画线详情： firstNodeDetail --画线详情,第一个节点详情， secondNodeDetail --第二个节点详情,双拌线并且是预警规则时存在， inTheArea --目标是否在此区域： true 表示在此区域， false 表示不在此区域
targetRect	string	非必须	目标框在全景图中的位置： top ：目标距离上方间距， left ：目标距离左方间距， bottom ：目标距离下方间距， right ：目标距离右方间距
alarmTime	date	非必须	警戒时间,算法仓设备为警戒开始时间
alarmEndTime	date	非必须	警戒结束时间

