

车牌识别一体机

HTTP 通讯协议说明手册

非常感谢您使用我们公司的产品，我们将竭诚为您提供最好的服务。本手册可能包含技术上不准确的地方或文字错误，欢迎您的纠正。本手册内容将做定期的更新，更新内容将在本手册的新版本中加入。我们随时会改进或更新本手册中描述的产品或程序。

版本变更记录

版本号	拟制日期	版本描述	存档编号
1.8.0	2018.08.23	1. 增加HTTP断线重传功能	20180823180

timeStamp

一、HTTP 服务器配置

1. 使用 HTTP 推送首先需要用户建立一个 HTTP 服务器，同时将这台 HTTP 服务器的地址配置给相机一体机；

2. 当一体机有识别结果后（或者其他需要推送的内容时），就会往指定的服务器地址发送 HTTP 协议消息；

3. 在一体机网页，登录后，点击菜单栏->高级设置->HTTP 推送，进入到 HTTP 推送的设置界面**(不同版本稍有区别)**；

4. 接收 HTTP 推送的服务器，配置包括地址（可以填 ip 地址或者域名），端口号，是否开启 ssl 连接，ssl 端口号，和超时时间设置。请根据架设的服务器的情况进行配置；

车牌识别一体机
实时视频
高级设置
图片预览
设备维护

中心服务器

基本配置

服务器地址	<input type="text" value="192.168.1.106"/>
备选服务器	<input type="text"/>
端口	<input type="text" value="80"/>
SSL连接	<input type="checkbox"/> 开启
验证方式	<input checked="" type="radio"/> 匿名 <input type="radio"/> CA证书
SSL端口	<input type="text" value="443"/>
超时时间(s)	<input type="text" value="5"/>

推送配置

设备注册	<input checked="" type="radio"/> 取消心跳 <input type="radio"/> 普通心跳 <input type="radio"/> comet轮询
主服务器优先	<input type="checkbox"/> 开启
地址	<input type="text" value="/devicemanagement/php/receivede"/>
推送车牌识别结果	<input type="checkbox"/> 开启 <input type="checkbox"/> 断线重传
地址	<input type="text" value="/devicemanagement/php/plateresul"/>
内容详细等级	<input type="text" value="全部"/>
发送图片	<input type="checkbox"/> 发送大图片 <input type="checkbox"/> 发送小图片
推送端口触发信息	<input type="checkbox"/> 开启
地址	<input type="text" value="/devicemanagement/php/gio.php"/>
推送串口数据	<input type="checkbox"/> 开启
地址	<input type="text" value="/devicemanagement/php/serial.php"/>
HTTP脱机检查	<input type="checkbox"/> 开启
重发次数	<input type="text" value="不启用"/>

确定

- 视频设置
- 白名单设置
- 输出联动
- 出入口配置
- 串口屏显设置
- 语音配置
- 网络设置
- HTTP推送
- 时间设置
- 存储管理

个地址用英文";"分隔

5. HTTP 服务器可配置 1 个主服务器，最多 3 个备选服务器；若勾选了主服务器优先，则相机只往主服务器推送，当且仅当主服务器断开连接时，才会往备选服务器推送；若没勾选主服务优先，则会同时往主/备服务器推送数据；

6. HTTP 推送具体配置，即 HTTP 需要推送的内容，包括车牌识别结果、IO 触发、串口 485 数据，需要配置相应推送的 url。

7. HTTP 心跳分为取消心跳、普通心跳、comet 轮询；普通心跳相机定时往主服务器推送心跳，不处理主服务器的业务回复；comet 轮询则一直与服务器推送心跳交互，并且服务器可在回复消息中携带业务处理；

8. HTTP 脱机检查，当开启脱机检查时，相机会对 HTTP 进行脱机检查；脱机检查分为心跳检查以及识别结果检查；心跳检查则为普通心跳定时检查；识别结果检查为当产生识别结果时，推送识别结果后需要在检查时间内收到服务器的回

复, 否则相机会置状态为脱机; 当相机处于脱机时, 会进行脱机相关的业务处理;

9. 配置重发次数, 最多可配置 4 次, 即当产生推送时, 若没推送成功, 相机会对推送数据进行重发;

二、车牌识别结果推送

开启推送车牌识别结果, 同时配置推送 url 后, 当有车牌识别结果产生时, 相机会按图中的配置会发送消息到:

<http://192.168.1.106/devicemanagement/php/plateresult.php>;

请注意: 在增加 HTTP 重传功能后, 相对于旧版本, 车牌识别结果推送消息增加了 3 个字段, 分别为: plateid、isoffline、gioouts;

数据内容: JSON 格式 (utf8 编码)

```
{
  "AlarmInfoPlate" : {
    "channel" : 0,
    "deviceName" : "IVS",
    "ipaddr" : "192.168.1.100",
    "result" : {
      "PlateResult" : {
        // 红色字体部分为扩展信息
        "plate_true_width" : 80,
        "plate_distance" : 40,
        "is_fake_plate" : 1,
        "car_location" :
        {
          "RECT" : {
            "bottom" : 545,
            "left" : 871,
            "right" : 1365,
            "top" : 391
          }
        },
        "car_brand" :
        {
          "brand" : 6
          "year" : 2018
          "type" : 1
        },
        "feature_Code" : "asf", // 扩展信息结束
        "bright" : 0,
        "carBright" : 0,
        "carColor" : 0,
        "colorType" : 0,

```

```

    "colorValue" : 0,
    "confidence" : 0,
    "direction" : 0,
    "imagePath" :
"%2Fmmc%2FVzIPCCap%2F2015_09_09%2F1714224504__%CE%DE_.jpg",
    "license" : "_无_",
    "location" : {
      "RECT" : {
        "bottom" : 0,
        "left" : 0,
        "right" : 0,
        "top" : 0
      }
    },
    "timeStamp" : {
      "Timeval" : {
        "decday" : 8,
        "dechour" : 10,
        "decmin" : 26,
        "decmon" : 6,
        "decsec" : 28,
        "decyear" : 2018,
        "sec" : 1441815171,
        "usec" : 672241
      }
    },
    "timeUsed" : 0,
    "triggerType" : 4,
    "type" : 0,
    "plateid" : 123,
    "isoffline" : 0,
    "gioouts" : [
      {"ionum" : 1, "ctrltype": 0}
      ...
    ]
  }
},
"serialno" : "eff50e18-e3d3862b"
}
}

```

根据内容详细等级，数据的内容会有所不同，“全部”将回传全部内容，简略的只包含其中部分重要内容，请根据需要选择详细度。开启报警发送图片后，json中的将包含图像数据。

包含全部(扩展)内容的 json 数据如下, 选项“全部”去掉了车头等扩展信息, “较详细”去掉了“location”, “timeUsed”, 选项“较简略”还去掉了“colorValue”, “confidence”, “bright”, “carBright”, “carColor”, 选项“简略”进一步去掉了“timeStamp”

说明:

字段名称	类型	取值范围	是否必须	描述
AlarmInfo Plate	json	无	是	推送结果为车牌识别结果
serialno	string	无	是	设备序列号, 设备唯一
channel	int	[0, 100]	是	默认通道号 (预留)
deviceName	string	无	是	设备名称
ipaddr	string	无	是	设备 ip 地址
result	json	无	是	实际数据
PlateResult	json	无	是	车牌识别结果信息
license	string	无	是	车牌号字符串, 如“京 AAAAAA”
colorValue	int	[0, 32]	是	(预留)
colorType	int	[0, 5]	是	车牌颜色 0: 未知、1: 蓝色、2: 黄色、3: 白色、4: 黑色、5: 绿色
type	int	[0, 19]	是	车牌类型 0: 未知车牌、1: 蓝牌小汽车、2: 黑牌小汽车、3: 单排黄牌、4: 双排黄牌、5: 警车车牌、6: 武警车牌、7: 个性化车牌、8: 单排军车牌、9: 双排军车牌、10: 使馆车牌、11: 香港进出中国大陆车牌、12: 农用车

字段名称	类型	取值范围	是否必须	描述
				牌、13：教练车牌、14：澳门进出中国大陆车牌、15：双层武警车牌、16：武警总队车牌、17：双层武警总队车牌、18：民航车牌、19：新能源车牌
confidence	int	[0, 100]	是	识别结果可信度 1-100
bright	int	无	是	亮度评价（预留）
direction	int	[0, 4]	是	车的行进方向，0：未知，1：左，2：右，3：上，4：下
location	json	无	是	车牌在图片中位置
RECT	json	无	是	位置为矩形区域；left\right\top\bottom:车牌在图片中位置
timeUsed	int	无	是	识别所用时间
carBright	int	无	是	车身亮度（预留）
carColor	int	无	是	车身颜色（预留）
timeStamp	json	无	是	识别结果对应帧的时间戳
Timeval	json	无	是	时间戳结构体类型
sec	uint32	无	是	从 1970 年 1 月 1 日到对应帧的秒
dechour	int	[0, 60]	是	时间，小时
decmin	Int	[0, 60]	是	时间，分钟

字段名称	类型	取值范围	是否必须	描述
decsec	Int	[0, 60]	是	时间，秒
decday	Int	[0, 31]	是	时间，天
decmon	Int	[0, 12]	是	时间，月
decyear	Int	[0, 2038]	是	时间，年
usec	uint32	无	是	从 1970 年 1 月 1 日到对应帧的毫秒
triggerType	int	[1, 8]	是	当前结果的触发类型：1：自动触发类型、2：外部输入触发（IO 输入）、4：软件触发（SDK）、8：虚拟线圈触发
imagePath	string	无	是	识别大图片的路径,开启推送大图片后，没有此字段
imageFile	string	无	是	识别大图片内容经过 base64 后的字符串
imageFileLen	int	无	是	识别大图片内容长度，注意不是 base64 后的长度
imageFragmentFile	string	无	是	识别车牌小图片内容经过 base64 后的字符串
imageFragmentFileLen	int	无	是	识别小图片内容长度，注意不是 base64 后的长度
plateid	unsigned int	无	否	识别结果车牌 ID

字段名称	类型	取值范围	是否必须	描述
isoffline	int	[0,1]	是	设备离线状态，0：在线，1：离线
ionum	int	[0, 3]	是	IO OUT 序号，当前最大 4 个 IOout
ctrltype	int	[0, 2]	是	开闸类型：HTTP_IO_OUT_STATUS
plate_true_width	int	[10, 1000]	是	车牌真实宽度
plate_distance	int	[0, 1000]	是	车牌距离
is_fake_plate	int	[0, 1]	是	是否伪车牌，0：真实车牌，1：伪车牌
car_location	无	无	是	车头位置
car_brand	无	无	无	车辆品牌
brand	int	[0, 128]	是	车辆品牌
year	int	[0, 65535]	是	车辆年份
type	int	[0, 128]	是	车辆类型
feature_Code	string	长度 [0, 20]	是	车辆特征码

请注意：增加 HTTP 重传功能后，需要使用此功能，服务器的回复中，必须携带 plateid 字段，和相机推送上来的数值保持一致；

中心服务器回复内容:

```
{
  "Response_AlarmInfoPlate": {
    "info": "ok", // 回复 ok 开闸
    "plateid" : 123, // 当前车牌 id
    "channelNum" : 0, // 回复开闸端口号, 若无, 则默认为 0
    "manualTrigger" : "ok", // 回复 ok 进行手动触发
    // (可选, 不触发截图可不添加该字段)
    "TriggerImage" : {
      // 回复截图内容端口号 (可选, 不填则默认使用 http 页面配置端口)
      "port": 80,
      // 回复截图内容相对路径 (可选, 不触发截图可不添加该字段)
      "snapImageRelativeUrl" : "/devicemanagement/php/receivedeviceinfo.php",
      // 回复截图内容绝对路径 (可选, 不触发截图可不添加该字段)
      "snapImageAbsolutelyUrl" : "http://192.168.1.106/devicemanagement/php/receivedeviceinfo.php"
    },
    "is_pay": "true",
    // 回复串口数据可以发送到相应串口
    "serialData" : [
      {
        "serialChannel": 0,
        "data" : "...",
        "dataLen" : 123
      },
      { // 数据 1, 可以有或者没有, 收到后将发送到对应串口
        "serialChannel": 1,
        "data" : "...",
        "dataLen" : 123
      } // 数据 2, 可以有或者没有, 收到后将发送到对应串口
    ]
  }
}
```

2.1 断线重传

1. 当 HTTP 服务器因为某些原因, 导致相机与服务器断线以后, 相机会把推送失败的识别结果记为离线记录, 当服务器重新连接上以后, 相机根据配置判断是否需要推送离线记录, 同时发送离线记录;

2. 配置在网页配置, 开启断线重传功能, **注意当取消断线重传功能时, 会清空当前相机的离线记录;**

3. 相机推送识别结果, 相对于旧版本的推送消息, 新增三个字段: plateid, isoffline, gioouts, 脱机记录 isoffline 的值为 1;

4. 服务器回复相机识别结果时，**在线记录需要将 plateid 字段值回复到相机消息中**；
5. 服务器回复离线识别结果时，**需要回复是否继续接收离线记录以及接收到最新的 plateid**；
6. 离线脱机记录理论上最大支持 **9000 条** 离线记录的重新推送；
7. 在推送离线记录的过程中，如果发生新的识别结果，**优先推送新的识别结果**，此时未完成推送的离线记录，会直接终止处理，当新识别结果推送完成之后，方重新开始推送离线记录；

注意：当推送离线记录，相机还未收到服务器的回复时，产生了新的识别结果，相机会终止上一条离线记录的推送处理，直接推送新的识别结果，当新的识别结果推送完毕后，再推送上一条离线记录，故服务器此时有可能会收到两条一模一样的离线记录，服务器可根据 **plateid** 进行过滤；

```
{
  "AlarmInfoPlate" : {
    "channel" : 0,
    "deviceName" : "IVS",
    "ipaddr" : "192.168.1.100",
    "result" : {
      "PlateResult" : {
        "bright" : 0,
        "carBright" : 0,
        "carColor" : 0,
        "colorType" : 0,
        "colorValue" : 0,
        "confidence" : 0,
        "direction" : 0,
        "imagePath" :
"%2Fmmc%2FVzIPCCap%2F2015_09_09%2F1714224504__%CE%DE_.jpg",
        "license" : "_无_",
        "location" : {
          "RECT" : {
            "bottom" : 0,
            "left" : 0,
            "right" : 0,
            "top" : 0
          }
        }
      },
      "timeStamp" : {
        "Timeval" : {
          "decday" : 8,
          "dechour" : 10,
          "decmin" : 26,
          "decmon" : 6,
          "decsec" : 28,
          "decyear" : 2018,

```

```

        "sec" : 1441815171,
        "usec" : 672241
    }
},
"timeUsed" : 0,
"triggerType" : 4,
"type" : 0,
"plateid" : 123,
"isoffline" : 0,
"gioouts" : [
    {"ionum" : 1, "ctrltype": 0}
]
}
},
"serialno" : "eff50e18-e3d3862b"
}
}

```

字段名称	类型	取值范围	是否必须	描述
plateid	unsigned int	无	否	识别结果车牌 ID
isoffline	int	[0,1]	是	设备离线状态，0: 在线，1: 离线
ionum	int	[0, 3]	是	IO OUT 序号，当前最大 4 个 IOout
ctrltype	int	[0, 2]	是	开闸类型：HTTP_IO_OUT_STATUS

服务器回复：

```

{
  "Response_AlarmInfoPlate": {
    ...// 其他数据
    "ContinuePushOffline":{
      "plateid" : 123,
      "continue": 1
    }
  }
}

```

字段名称	类型	取值范围	是否必须	描述
------	----	------	------	----

字段名称	类型	取值范围	是否必须	描述
plateid	unsigned int	无	是	推送的离线车牌记录 ID
continue	unsigned int	[0,1]	是	是否继续推送离线记录，0: 否，1: 是

注意：只有当服务器回复了离线记录消息，以及 **continue** 字段为 **1**时，才继续推送下一条离线记录；

2.2 墨系列相机

针对墨系列的相机，HTTP 推送协议增加车辆信息以及伪车牌信息：

```
{
  "AlarmInfoPlate" :
  {
    ...// 其他字段
    "result" :
    {
      "PlateResult" :
      {
        ...//
        "plate_true_width" : 80,
        "plate_distance" : 40,
        "is_fake_plate" : 1,
        "car_location" :
        {
          "RECT" : {
            "bottom" : 545,
            "left" : 871,
            "right" : 1365,
            "top" : 391
          }
        },
        "car_brand" :
        {
          "brand" : 6
          "year" : 2018
          "type" : 1
        },
        "feature_Code" : "asf"
      }
    }
  }
}
```

```
}  
}
```

字段名称	类型	取值范围	是否必须	描述
plate_true_width	int	[10, 1000]	是	车牌真实宽度
plate_distance	int	[0, 1000]	是	车牌距离
is_fake_plate	int	[0, 1]	是	是否伪车牌, 0: 真实车牌, 1: 伪车牌
car_location	无	无	是	车头位置
car_brand	无	无	无	车辆品牌
brand	int	[0, 128]	是	车辆品牌
year	int	[0, 65535]	是	车辆年份
type	int	[0, 128]	是	车辆类型
feature_Code	string	长度[0, 20]	是	车辆特征码

三、端口触发信息推送

当开启时, 如果在输入输出页面->车牌触发方式里, 开启了外部输入 1 触发或者 2 触发, 输入有变化时, 会推送 json 格式数据, 内容如下:

数据内容: JSON 格式 (utf8 编码)

```
{  
  "AlarmGioIn" :  
  {  
    "deviceName" : "IVS",  
    "ipaddr" : "192.168.109.40",  
    "result" :  
    {  
      "TriggerResult" :  
      {
```

```

        "source" : 3,
        "value" : 1
    }
},
"serialno" : "d85b1269-8f942256"
}
}

```

其中，TriggerResult 中：

source=0 代表是 IO 输入 1；

source=1 代表是 IO 输入 2；

source=2 代表是 IO 输入 3；

source=3 代表输入 TCP 触发输入；

source=4 代表是 IO 输入 4；

value 表示触发时输入的状态；

其他参数的含义和车牌识别结果推送中相同。

中心服务器回复内容：任意合法数据，相机不对回复消息做业务处理

四、串口数据推送

当开启串口数据推送时，配置了 url，在相机收到 485 数据时，会主动往服务器地址推送 485 数据；

数据内容：JSON 格式（utf8 编码）

```

{
  "SerialData":{
    "channel" : 0, // 通道号, 当前为0
    "serialno" : "cead13eb-1a198cd7", // 设备序列号
    "ipaddr" : "192.168.1.100", // 设备ip
    "deviceName" : "IVS", // 设备名称
    "serialChannel" : 0, // 串口的通道号, 通道0 为 485 口1, 通道1 根据跳线方式为 485 口2
    // 或者 232
    "data": "Y2guY29tFw==", // 串口数据, 采用 base64 编码
    "dataLen" : 7 // 串口数据实际长度
  }
}

```

中心服务器回复内容：任意合法数据，相机不对回复消息做业务处理

五、截图数据

用户在 comet 轮询或者收到识别结果的回复字段有获取截图时，设备会进行当前视频截图并上传，imageFile 字段为图片 base64 后的编码，imageFileLen 为编码前的图片长度

数据内容：JSON 格式（utf8 编码）

```
//推送截图数据
{
  "ipaddr" : "192.168.1.100",
  "TriggerImage":
  {
    "imageFile": "Y2guY29tFw==", //图片数据（base64 编码）
    "imageFileLen": 7 //图片数据实际长度
  }
}
```

中心服务器回复内容：任意合法数据，相机不对回复消息做业务处理

六、设备注册

6.1 普通心跳

1. 当相机网页配置设备注册状态为普通心跳时，相机会定时往主服务器推送心跳消息；
2. 当主服务连接正常，开启脱机检查的情况下，相机每隔 5S 左右推送一次心跳；
3. 当主服务连接正常，同时没开启脱机检查的情况下，是 30S 推送一次心跳消息；
4. 当主服务心跳丢失以后，相机每隔 1S 尝试连接一次；
5. 心跳推送使用 HTTP 长连接；

数据内容，使用 formpost 的格式：

```
192.168.109.40-----caa771fe4a61f3d9Content-Disposition:
form-data;
  name="device_name"IVS-----caa771fe4a61f3d9Content-Disposition:
form-data;

name="ipaddr"192.168.109.40-----caa771fe4a61f3d9Content-Dispos
ition: form-data;
  name="port"80-----caa771fe4a61f3d9Content-Disposition:
form-data;
  name="user_name"admin-----caa771fe4a61f3d9Content-Disposition:
form-data;
```

```
name="pass_wd"admin-----caa771fe4a61f3d9Content-Disposition:
form-data;

name="serialno"d85b1269-8f942256-----caa771fe4a61f3d9Content-D
isposition: form-data;

name="channel_num"1-----caa771fe4a61f3d9--
```

6.2 comet 轮询

1. 当开启 comet 轮询之后，相机一直与 HTTP 服务器进行交互，保持连接请求，相机主动发送设备注册消息，内容与普通心跳内容一致，收到回复时，立即发送下一条消息；
2. 发送设备注册消息，与普通心跳消息保持一致；
3. comet 轮询会根据服务器回复做相应处理；

七、业务处理

1. 相机根据服务器的回复消息，进行相应的业务处理；
2. 当前仅支持**车牌识别结果的推送**回复，以及 **comet 轮询**的消息回复，相机会根据回复做业务处理；

7.1 控制 IO 开闸

服务器在收到车牌识别结果推送、或者 comet 轮询时，回复以下结构的消息，可触发开闸

```
{
  "Response_AlarmInfoPlate":
  {
    "info":"ok",//回复ok 开闸
    // ....其他数据
  }
}
```

7.2 控制串口推送 485 数据

服务器在收到车牌识别结果推送、或者 comet 轮询时，回复以下结构的消息，可发送 485 数据

```
{
  "Response_AlarmInfoPlate":
  {
    "serialData" :[
```

```

    {
        "serialChannel":0,
        "data" : "...",
        "dataLen" : 123
    },
    { //数据1, 可以有或者没有, 收到后将发送到对应串口
        "serialChannel":1,
        "data" : "....",
        "dataLen" : 123
    } //数据2, 可以有或者没有, 收到后将发送到对应串口
    ]
    // ....其他数据
}
}

```

7.3 截图

服务器在收到车牌识别结果推送、或者 comet 轮询时，回复以下结构的消息，可触发截图：

相机会触发当前的视频截图，然后将截图数据推送到 `snapImageAbsolutelyUrl` 字段指定的服务器地址；

```

{
    "Response_AlarmInfoPlate":
    {
        "TriggerImage" :
        {
            //回复截图内容端口号（可选，不填则默认使用http 页面配置端口）
            "port":80,
            //回复截图内容相对路径（可选，不触发截图可不添加该字段）
            "snapImageRelativeUrl" : "/devicemanagement/php/receivedeviceinfo.php",
            //回复截图内容绝对路径（可选，不触发截图可不添加该字段）
            "snapImageAbsolutelyUrl" : "http://192.168.1.106/devicemanagement/php/receivedeviceinfo.php"
        }
        // ....其他数据
    }
}

```

7.4 手动触发识别

服务器在收到车牌识别结果推送、或者 comet 轮询时，回复以下结构的消息，可触发手动识别：

```
{
  "Response_AlarmInfoPlate":
  {
    "manualTrigger" : "ok", //回复 ok 进行手动触发
    // ....其他数据
  }
}
```

或者仅回复以下数据：

```
{
  "type": "AVS_TRIGGER",
}
```

此时会触发相机的手动识别，设备会将识别结果数据推送至 http 服务端，前提是服务端配置了识别数据的推送；

7.5 白名单操作

中心服务器在收到 HTTP 推送的识别结果时，在回复的识别结果消息中，携带白名单操作信息

注意：单条回复消息，最多操作 5 条白名单数据；

1. 回复消息使用 JSON 格式，格式如下

```
{
  "Response_AlarmInfoPlate": {
    ...// 其他数据
    "white_list_operate":{
      "operate_type" : 0,
      "white_list_data":
      [
        {
          "plate": "京 A12345",
```

```

        "enable": 1,
        "need_alarm": 1,
        "enable_time": "2018-01-01 11:11:11",
        "overdue_time": "2018-01-01 11:11:11"
    },
    {
        "plate": "川A12345",
        "enable": 1,
        "need_alarm": 1,
        "enable_time": "2018-01-01 11:11:11",
        "overdue_time": "2018-01-01 11:11:11"
    }
]
}
}
}
}

```

2. 字段含义如下:

字段名称	类型	取值范围	是否必须	描述
operate_type	int	[0,1]	是	操作类型(0:增加, 1: 删除)
white_list_data	无	个数[1, 5]	是	白名单数组: 单次操作, 最大支持 5 条
plate	string	长度[7, 15]	是	车牌 (GB2312)
enable	int	[0, 1]	是	当前名单是否有效 (0: 无效, 1: 有效)
need_alarm	int	[0, 1]	是	当前名单是否为黑名单 (0: 否, 1: 黑名单)

字段名称	类型	取值范围	是否必须	描述
enable_time	string	固定格式、 固定长度： 19	否	当前名单生效时间，如：2018-01-01 11:11:11
overdue_time	string	固定格式、 固定长度： 19	否	当前名单过期时间，如：2018-01-01 11:11:11

3.消息示例:

同时增加两条白名单数据（多条数据，格式一致）：

第一条没有白名单有效时间，则默认白名单永久有效；

第二条有白名单有效时间，表示当前白名单在这个时间段内有效；

```
{
  "Response_AlarmInfoPlate": {
    ...// 其他数据
    "white_list_operate":{
      "operate_type" : 0,
      "white_list_data":
      [
        {
          "plate": "京 A12345",
          "enable": 1,
          "need_alarm": 0
        },
      ],
    }
  }
}
```

```

        {
            "plate": "川 A12345",
            "enable": 1,
            "need_alarm": 0,
            "enable_time": "2018-01-01 11:11:11",
            "overdue_time": "2019-01-01 11:11:11"
        }
    ]
}
}
}

```

4. 删除两条白名单的回复消息示例:

```

{
    "Response_AlarmInfoPlate": {
        ...// 其他数据
        "white_list_operate":{
            "operate_type" : 1,
            "white_list_data":
            [
                {
                    "plate": "京 A12345"
                },
                {
                    "plate": "川 A12345"
                }
            ]
        }
    }
}
}

```

5. 删除全部白名单的回复消息示例:

```

{
    "Response_AlarmInfoPlate": {
        ...// 其他数据
        "white_list_operate":{
            "operate_type" : 1,
            "white_list_data":
            [
                {
                    "plate": ""
                }
            ]
        }
    }
}

```

```
    }
  ]
}
}
```

八、常见问题

Q: 设备注册是什么?

A: 当开启时, 每隔一段时间, 一体机会自动发送设备信息到中心服务器, 包括设备 ip, 端口, 序列号等信息。

Q: 设置好了, 请求收不到, 什么问题?

A: 请确保一体机可以访问中心服务器的相应地址。常见的问题如, 局域网内, 网线是否接好, ip 地址是否冲突, 是否在可以访问的网段; 中心服务器如果在公网, 请确保一体机可以访问公网, 需要设置好一体机的网关和 dns 地址。检查中心服务器是否运行。

Q: 请求收到了, 但没有数据 (数据格式不对)?

A: 车牌识别结果推送的请求发送的是 json 数据, http 的 body 内容如, {"AlarmInfoPlate":{...}}

接收方法例如:

php:

```
$doc = file_get_contents("php://input");
```

java:

```
StringBuffer jb = new StringBuffer();
String line = null;
try {
    BufferedReader reader = request.getReader();
    while ((line = reader.readLine()) != null)
        jb.append(line);
} catch (Exception e) { /*report an error*/ }
```

详细参考: <http://stackoverflow.com/questions/3831680/httpservletrequest-get-post-data>

Q: 设备注册又是什么格式?

A: 设备注册请求发送的数据内容如下:

```
-----cd9a1a32759bContent-Disposition: form-data; name="device_name"
IVS-----cd9a1a32759bContent-Disposition: form-data; name="ipaddr"19
2.168.0.100-----cd9a1a32759bContent-Disposition: form-data; name="po
rt"80-----cd9a1a32759bContent-Disposition: form-data; name="user_na
me"admin-----cd9a1a32759bContent-Disposition: form-data; name="pas
s_wd"admin-----cd9a1a32759bContent-Disposition: form-data; name="se
```

rialno"fc68a83-ee8409dd-----cd9a1a32759bContent-Disposition: form-data; name="channel_num"1-----cd9a1a32759b--

如所见是 formpost 的格式，接收方法例如：java 使用 request.getQueryString 接收，php 使用 \$_POST 变量接收

Q: 如何回复请求开闸?

A: 回复{"Response_AlarmInfoPlate":{"info":"ok","content":"...", "is_pay":"true"}}
info 如果是 ok 表示开闸

Q: 回复中 content 能不能是中文?

A: 所有请求都用 utf8 进行编码，回复也用 utf8 即可。

Q: 能否使用 ssl 连接发送，我们的中心服务器是 ssl 的?

A: 在设置中设置 ssl 端口（一般是 443），然后选上开启，设置就可以了，注意如果中心服务器不支持 ssl 连接，请不要选择开启该项。

Q: 怎么获取截图?

A: 推送的结果中有"imagePath": "/snapshot/lpr/tri_snap_24.jpg"，后面是访问截图的 http 路径，前面加上一体机的网址，就可以得到截图的地址如

http://192.168.1.100:8080/snapshot/lpr/tri_snap_24.jpg

Q: 为什么相同车牌返回了两次结果?

A: 推送的结果中有一项触发类型 triggerType，可以根据触发类型来过滤结果。

Q: 中心服务器，收到推送结果，回复给一体机，但看不到返回的具体内容，如何调试中心服务器?

A: 在网页上可以查看访问中心服务器的日志，可以看到中心服务器回复给一体机的内容。

Q: 多台相机同时使用 comet 轮询时，中心服务器压力过大，导致服务器响应不过来?

A: comet 轮询机制，为相机循环往中心服务器推送心跳包的数据，当服务器收到推送，回复响应以后，相机立马启动下一心跳包的推送，中间没有定时器等时延；

故当相机数量较大时，同时使用 comet 方式往一个服务器推送，请注意服务器性能；