

智能锁PasS平台接口文档

- 智能锁项目接口文档
 - 1 配置管理
 - 1.1 登录鉴权
 - 1.2 配置推送
 - 1.3 查询智能锁列表
 - 1.4 查询开锁记录
 - 1.5 重置推送错误次数
 - 2 密码管理
 - 2.1 添加永久密码
 - 2.2 修改永久密码
 - 2.3 删除永久密码
 - 2.4 添加临时密码
 - 2.5 修改临时密码
 - 2.6 删除临时密码
 - 2.7 添加蓝牙开锁权限
 - 2.8 删除蓝牙开锁权限
 - 2.9 获取蓝牙开锁密钥
 - 2.10 获取添加门卡蓝牙指令
 - 2.11 添加门卡
 - 2.12 删除门卡
 - 2.13 添加指纹
 - 2.14 删除指纹
 - 2.15 远程开锁
 - 2.16 清空指定密码
 - 2.17 删除门锁指定类型密码
 - 2.18 获取蓝牙开锁密钥(无需授权)
 - 2.19 查询指纹数据
 - 2.20 按锁编号查询临时密码
 - 2.21 按锁编号查询永久密码
 - 2.22 按锁编号查询蓝牙权限数据
 - 2.23 按锁查询临时密码使用情况
 - 2.24 按锁查询永久密码使用情况
 - 2.25 按锁查询多次输入错误密码报警日志记录
 - 2.26 按锁编号和标识查询临时密码数据
 - 2.27 拉取推送失败或没有推送的数据
 - 2.28 配置把手朝向
 - 2.29 查询门锁所有分类各自的日志记录总数
 - 2.30 查询下发密码到锁的时长
 - 2.31 添加离线密码
 - 3 电表管理
 - 3.1 查询电表信息
 - 3.2 查询电表历史信息
 - 3.3 查询电表告警信息
 - 3.4 下发电表指令
 - 4 水表管理
 - 4.1 查询水表信息
 - 4.2 查询水表历史信息
 - 4.3 查询水表告警信息
 - 4.4 下发水表指令
 - 5 加解密管理
 - 5.1 加密管理
 - 5.2 解密管理
 - 5.3 CRC8校验值
 - 6 门禁锁管理

url	String	配置要推送的URL (必填)
响应	code	状态码:0-成功,非0:失败
data	登录的用户信息	

示例请求:

```
curl -X PUT /api/configurationPush.do \
-H 'content-type: application/json' \
-d '{
  "url": "https://xxx.xxx.xxx.xxx/lock/callback"
}'
```

示例响应:

```
{
  "code": 0,
  "data": null,
  "msg": "配置成功!"
}
```

门锁上报推送数据格式:

参数	类型	描述
productType	INT	产品厂商 1: 门锁 2: 电表 3: 水表 4: 门禁锁
type	INT	数据类型 1:锁上报推送 2: 告警信息 3: 下发指令回调 4: 开锁记录 5:离线推送
deviceNum	String	设备编号
battery	INT	电池电量
signalStrength	INT	信号强度
dataType	INT	0:心跳上报 1: 门磁状态变化

```
{
  "productType":1,
  "type": 1,
  "data": {
    "deviceNum":"18100402000208",
    "battery":80,
    "dataType":0,
    "signalStrength":80
  }
}
```

参数	类型	描述
productType	INT	产品厂商 1: 门锁 2: 电表 3: 水表 4: 门禁锁
type	INT	数据类型 1:锁上报推送 2: 告警信息 3: 下发指令回调 4: 开锁记录 5:离线推送
deviceNum	String	设备编号
battery	INT	电池电量
signalStrength	INT	信号强度
dataType	INT	0:心跳上报 1: 门磁状态变化

gateState	INT	门磁标志。0位关闭，1位打开
-----------	-----	----------------

```
{
  "productType":1,
  "type": 1,
  "data": {
    "deviceNum":"181004020000208",
    "battery":80,
    "dataType":1,
    "gateState":1,
    "signalStrength":80
  }
}
```

参数	类型	描述
productType	INT	产品厂商 1: 门锁 2: 电表 3: 水表 4: 门禁锁
type	INT	数据类型 1:锁上报推送 2: 告警信息 3: 下发指令回调 4: 开锁记录 5:离线推送
deviceNum	String	设备编号
alarmType	INT	1: 撞击报警 2: 低电量 3: 撬锁报警 5: 长时间未关 6: 蓝牙cookie错误 8: 非法开门报警 9: 密码错误自锁报警

```
{
  "productType":1,
  "type": 2,
  "data": {
    "deviceNum":"181004020000208",
    "alarmType":2
  }
}
```

参数	类型	描述
productType	INT	产品厂商 1: 门锁 2: 电表 3: 水表 4: 门禁锁
type	INT	数据类型 1:锁上报推送 2: 告警信息 3: 下发指令回调 4: 开锁记录 5:离线推送
deviceNum	String	设备编号
uuid	String	uuid
callbackState	INT	0: 成功 1: 失败
uuid	String	uuid
deviceNum	String	设备编号
commandType	INT	1:同步临时密码 2: 同步成员密码 4: 同步门卡 6: 同步指纹 7:远程开锁 8:清空指令 9:清空指令(包含蓝牙密码)

```
{
  "productType":1,
  "type": 3,
  "data": {
    "uuid": "a4c89c2e-d02b-4e0c-a821-d8b840717bd9",
    "deviceNum":"181004020000208",
    "commandType":1,
    "callbackState":0
  }
}
```

参数	类型	描述
productType	INT	产品厂商 1: 门锁 2: 电表 3: 水表 4: 门禁锁
type	INT	数据类型 1:锁上报推送 2: 告警信息 3: 下发指令回调 4: 开锁记录 5:离线推送
deviceNum	String	设备编号
unlockType	INT	开锁类型:1: 永久密码 2: 指纹 3: 临时密码 4: 远程开锁 5: 蓝牙开锁 6: cookie开锁 7: 门卡开锁 10: 机械钥匙开门 11: 反锁 12 - 门内开锁 15:离线开锁
identification	String	标识
recordType	INT	反锁(1:关闭反锁 2:打开反锁) 门内开锁(1:门内开门 2:弹出方舌) 其它开锁类型这个字段无效
unlockTime	String	记录时间
batteryInfo	String	电池电量
status	String	开锁状态 0-成功 -1 失败

```
{
  "productType":1,
  "type": 4,
  "data": {
    "unlockTime": "2017-08-21 15:22:03",
    "deviceNum": "181004020000192",
    "unlockType": 1,
    "status": 0,
    "identification": "1",
    "batteryInfo": 100,
    "recordType": 1
  }
}
```

参数	类型	描述
productType	INT	产品厂商 1: 门锁 2: 电表 3: 水表 4: 门禁锁
type	INT	数据类型 1:锁上报推送 2: 告警信息 3: 下发指令回调 4: 开锁记录 5:离线推送
deviceNum	String	设备编号
dateTime	String	记录时间

```
{
  "productType":1,
  "type": 5,
  "data": {
    "dateTime": "2019-08-29 15:22:03",
    "deviceNum": "181004020000192"
  }
}
```

电表上报推送数据格式:

参数	类型	描述
productType	INT	产品厂商 1: 门锁 2: 电表 3: 水表 4: 门禁锁
type	INT	数据类型 1:电表上报推送 2: 告警信息 3: 下发指令回调
deviceNum	String	设备编号

degree	String	度数
voltage	String	电压 -1是没有值
sluiceGate	INT	开合闸状态 0:合闸 1:跳闸

```
{
  "productType":2,
  "type": 1,
  "data": {
    "deviceNum":"5130001000014001606057",
    "degree":"80.2",
    "voltage":"-1",
    "sluiceGate":0
  }
}
```

参数	类型	描述
productType	INT	产品厂商 1: 门锁 2: 电表 3: 水表 4: 门禁锁
type	INT	数据类型 1:电表上报推送 2: 告警信息 3: 下发指令回调
deviceNum	String	设备编号
alarmType	INT	0:过流告警 1:开表盖告警 2: 电能表清零告警 3:其它告警

```
{
  "productType":2,
  "type": 2,
  "data": {
    "deviceNum":"5130001000014001606057",
    "alarmType":0
  }
}
```

参数	类型	描述
productType	INT	产品厂商 1: 门锁 2: 电表 3: 水表 4: 门禁锁
type	INT	数据类型 1:电表上报推送 2: 告警信息 3: 下发指令回调
deviceNum	String	设备编号
uuid	String	uuid
callbackState	INT	0: 成功 1: 失败

```
{
  "productType":2,
  "type": 3,
  "data": {
    "uuid": "a4c89c2e-d02b-4e0c-a821-d8b840717bd9",
    "deviceNum":"5130001000014001606057",
    "callbackState":0
  }
}
```

水表上报推送数据格式:

参数	类型	描述

productType	INT	产品厂商 1: 门锁 2: 电表 3: 水表 4: 门禁锁
type	INT	数据类型 1:水表上报推送 2: 告警信息 3: 下发指令回调
deviceNum	String	设备编号
accumulatedDosage	String	累计用量
voltage	String	电压 -1是没有值
sluiceGate	INT	开合闸状态 0:开闸 1:合闸

```
{
  "productType":3,
  "type": 1,
  "data": {
    "deviceNum":"5130001000014001606057",
    "accumulatedDosage":80,
    "voltage":-1,
    "sluiceGate":0
  }
}
```

参数	类型	描述
productType	INT	产品厂商 1: 门锁 2: 电表 3: 水表 4: 门禁锁
type	INT	数据类型 1:水表上报推送 2: 告警信息 3: 下发指令回调
deviceNum	String	设备编号
alarmType	INT	0:过流告警 1:开表盖告警 2: 水表清零告警 3:其它告警

```
{
  "productType":3,
  "type": 2,
  "data": {
    "deviceNum":"5130001000014001606057",
    "alarmType":0
  }
}
```

参数	类型	描述
productType	INT	产品厂商 1: 门锁 2: 电表 3: 水表 4: 门禁锁
type	INT	数据类型 1:水表上报推送 2: 告警信息 3: 下发指令回调
deviceNum	String	设备编号
uuid	String	uuid
callbackState	INT	0: 成功 1: 失败

```
{
  "productType":3,
  "type": 3,
  "data": {
    "uuid": "a4c89c2e-d02b-4e0c-a821-d8b840717bd9",
    "deviceNum":"5130001000014001606057",
    "callbackState":0
  }
}
```

门禁锁上报推送数据格式:

参数	类型	描述
productType	INT	产品厂商 1: 门锁 2: 电表 3: 水表 4: 门禁锁
type	INT	数据类型 1:锁上报推送 2: 开锁记录 3: 下发指令回调
deviceNum	String	门禁设备编号
dataType	INT	数据类型 1: 门锁状态 2: 心跳
doorMsg	INT	门锁状态 (0: 关 1: 开)

```
{
  "productType":4,
  "type": 1,
  "data": {
    "deviceNum":"5130001000014001606057",
    "dataType":1,
    "doorMsg":0
  }
}
```

参数	类型	描述
productType	INT	产品厂商 1: 门锁 2: 电表 3: 水表 4: 门禁锁
type	INT	数据类型 1:锁上报推送 2: 开锁记录 3: 下发指令回调
deviceNum	String	门禁设备编号
dataType	INT	数据类型 1: 门锁状态 2: 心跳

```
{
  "productType":4,
  "type": 1,
  "data": {
    "deviceNum":"5130001000014001606057",
    "dataType":2
  }
}
```

参数	类型	描述
productType	INT	产品厂商 1: 门锁 2: 电表 3: 水表 4: 门禁锁
type	INT	数据类型 1:锁上报推送 2: 开锁记录 3: 下发指令回调
deviceNum	String	门禁设备编号
authType	String	开锁类型 1: 远程开锁 2:临时密码开锁 3: 刷卡开锁 10:出门按钮动作 15:门异常打开 17:门异常关闭 21:未注册卡, 刷卡无效 22:过期卡, 刷卡无效 29:无效密码进入
status	INT	0:成功 1:失败

```
{
  "productType":4,
  "type": 2,
  "data": {
    "deviceNum":"5130001000014001606057",
    "authType":1,

```



```

    "status":0
  }
}

```

参数	类型	描述
productType	INT	产品厂商 1: 门锁 2: 电表 3: 水表 4: 门禁锁
type	INT	数据类型 1:锁上报推送 2: 开锁记录 3: 下发指令回调
deviceNum	String	门禁设备编号
uuid	String	uuid
status	INT	1:成功 0:失败
commandType	INT	1:远程开门 3:下发临时密码 4: 删除临时密码 5: 删除全部临时密码 6: 下发卡号 7:删除卡 9: 删除全部卡

```

{
  "productType":4,
  "type": 3,
  "data": {
    "uuid": "a4c89c2e-d02b-4e0c-a821-d8b840717bd9",
    "deviceNum":"5130001000014001606057",
    "commandType":9
    "status":0
  }
}

```

1.3 查询智能锁列表

URL	GET	/api/deviceList.do
参数	application/json	
deviceNum	设备编号	
pageNum	当前页（必填）	
pageSize	每页显示数量（必填）	
响应	智能锁基本信息	
list.deviceNum	设备号	
list.lockStatus	门锁状态, 0:关闭,1:打开,-1:异常	
list.signalStrength	信号强度	
list.connStatus	锁的连接状态, 0:离线,1:在线	
list.battery	电池电量	
list.createAt	创建时间	
list.fwVersion	软件版本号	
list.bluetoothMac	MAC地址	

示例请求:

```

curl -X GET /api/deviceList.do?pageNum=1&pageSize=10 \
-H 'content-type: application/json' \

```

```
-d ''
```

示例响应:

```
{
  "code": 0,
  "data": {
    "pageNum": 1,
    "pageSize": 10,
    "total": 1,
    "list": [
      {
        "battery": 99,
        "connStatus": 1,
        "createAt": "2019-03-14 09:59:44",
        "deviceNum": "18100402000192",
        "lockStatus": 1,
        "signalStrength": 69,
        "bluetoothMac": "cd62b44374d2",
        "fwVersion": 355
      }
    ]
  },
  "msg": "成功!"
}
```

1.4 查询开锁记录

URL	GET	/api/unlockRecord.do
参数	application/json	
deviceNum	设备编号	
authType	开锁类型:1: 永久密码 2: 指纹 3: 临时密码 4: 远程开锁 5: 蓝牙开锁 6: cookie开锁 7: 门卡开锁 8- 撤防/布防 10 – 机械钥匙开门 11 – 反锁 12 - 门内开锁	
pageNum	当前页 (必填)	
pageSize	每页显示数量 (必填)	
startTime	开始时间	
endTime	结束时间	
响应	智能锁基本信息	
deviceNum	设备号	
authType	开锁类型:1: 永久密码 2: 指纹 3: 临时密码 4: 远程开锁 5: 蓝牙开锁 6: cookie开锁 7: 门卡开锁	
identification	标识	
opTime	记录时间	
status	开锁状态 0-成功 -1 失败	
unlockDuration	开锁时长 (单位: 秒)	

示例请求:

```
curl -X GET /api/unlockRecord.do?pageNum=1&pageSize=10 \
-H 'content-type: application/json' \
```

```
-d ''
```

示例响应:

```
{
  "code": 0,
  "data": {
    "pageNum": 1,
    "pageSize": 10,
    "total": 1,
    "list": [
      {
        "opTime": "2017-08-21 15:22:03",
        "deviceNum": "18100402000192",
        "authType": 1,
        "unlockDuration": 1,
        "identification": "1",
        "status": 0
      }
    ]
  },
  "msg": "成功!"
}
```

1.5 重置推送错误次数

URL	PUT	/api/resetPushErrorNum.do
参数	application/json	
响应	data	若没有记录返回空数组

示例请求:

```
curl -X PUT /api/resetPushErrorNum.do \
-H 'content-type: application/json' \
-d '{
}'
```

示例响应

```
{
  "code": 0,
  "data": "重置成功!",
  "msg": "succ"
}
```

2 密码管理

2.1 添加永久密码

URL	POST	/api/permanentPassword.do
参数	application/json	
deviceNum	设备编号	
passwordList	密码集合	
identification	密码标识 (passwordList集合的参数) 最长20	

password	密码 (passwordList集合的参数)	
响应	data	若没有记录返回空数组
UUID	生成UUID	

示例请求:

```
curl -X POST /api/permanentPassword.do \
-H 'content-type: application/json' \
-d '{
  "deviceNum": "181004020000208",
  "passwordList": [
    {
      "identification": "1",
      "password": "123456"
    },
    {
      "identification": "2",
      "password": "654321"
    }
  ]
}'
```

示例响应

```
{
  "code": 0,
  "data": {
    "UUID": "f03d93d2-a9d1-4721-aba9-95b860a35b9d"
  },
  "msg": "添加成功"
}
```

2.2 修改永久密码

URL	PUT	/api/permanentPassword.do
参数	application/json	
deviceNum	设备编号	
identification	密码标识	
password	密码	
响应	0-成功,1-失败	
UUID	生成UUID	

示例请求:

```
curl -X PUT /api/permanentPassword.do \
-H 'content-type: application/json' \
-d '{
  "deviceNum": "181004020000208",
  "identification": "1",
  "password": "123456"
}'
```

示例响应:

```
{
```

```

"code": 0,
"data": {
  "UUID": "f03d93d2-a9d1-4721-aba9-95b860a35b9d"
},
"msg": "修改成功"
}

```

2.3 删除永久密码

URL	DELETE	/api/permanentPassword.do
参数	application/json	
deviceNum	设备编号	
identifications	密码标识数组	
响应	0-成功,1-失败	
UUID	生成UUID	

示例请求:

```

curl -X DELETE /api/permanentPassword.do \
-H 'content-type: application/json' \
-d '{
  "deviceNum": "18100402000208",
  "identifications": ["1", "2"]
}'

```

示例响应:

```

{
  "code": 0,
  "data": {
    "UUID": "f03d93d2-a9d1-4721-aba9-95b860a35b9d"
  },
  "msg": "删除成功"
}

```

2.4 添加临时密码

URL	POST	/api/temporaryPassword.do
参数	application/json	
deviceNum	设备编号	
passwordList	密码集合	
identification	密码标识 (passwordList集合的参数) 最长20	
password	密码 (passwordList集合的参数)	
startTime	密码生效起始时间	
endTime	密码失效时间	
响应	data	若没有记录返回空数组
UUID	生成UUID	

示例请求:

```
curl -X POST /api/temporaryPassword.do \
-H 'content-type: application/json' \
-d '{
  "deviceNum": "181004020000208",
  "passwordList": [
    {
      "identification": "1",
      "password": "123456",
      "startTime": "2019-05-09 18:33:36",
      "endTime": "2019-05-09 20:33:36"
    },
    {
      "identification": "2",
      "password": "654321",
      "startTime": "2019-05-09 18:33:36",
      "endTime": "2019-05-09 20:33:36"
    }
  ]
}'
```

示例响应:

```
{
  "code": 0,
  "data": {
    "UUID": "f03d93d2-a9d1-4721-aba9-95b860a35b9d"
  },
  "msg": "添加成功"
}
```

2.5 修改临时密码

URL	PUT	/api/temporaryPassword.do
参数	application/json	
deviceNum	设备编号	
identification	密码标识	
password	密码	
startTime	密码生效起始时间	
endTime	密码失效时间	
响应	0-成功,1-失败	
UUID	生成UUID	

示例请求:

```
curl -X PUT /api/temporaryPassword.do \
-H 'content-type: application/json' \
-d '{
  "deviceNum": "181004020000208",
  "identification": "1",
  "password": "123456",
  "startTime": "2019-05-09 18:33:36",
  "endTime": "2019-05-09 20:33:36"
}'
```

示例响应:

```
{
  "code": 0,
  "data": {
    "UUID": "f03d93d2-a9d1-4721-aba9-95b860a35b9d"
  },
  "msg": "修改成功"
}
```

2.6 删除临时密码

URL	DELETE	/api/temporaryPassword.do
参数	application/json	
deviceNum	设备编号	
identifications	密码标识数组	
响应	0-成功,1-失败	
UUID	生成UUID	

示例请求:

```
curl -X DELETE /api/temporaryPassword.do \
-H 'content-type: application/json' \
-d '{
  "deviceNum": "18100402000208",
  "identifications": ["1", "2"]
}'
```

示例响应:

```
{
  "code": 0,
  "data": {
    "UUID": "f03d93d2-a9d1-4721-aba9-95b860a35b9d"
  },
  "msg": "删除成功"
}
```

2.7 添加蓝牙开锁权限

URL	POST	/api/bluetooth/unlockAuthority.do
参数	application/json	
deviceNum	设备编号	
passwordList	密码集合	
identification	权限标识 (passwordList集合的参数) 最长20	
startTime	开始时间	
endTime	结束时间	
响应	data	若没有记录返回空数组

示例请求:

```
curl -X POST /api/bluetooth/unlockAuthority.do \
-H 'content-type: application/json' \
-d '{
  "deviceNum": "18100402000208",
  "passwordList": [
    {
      "identification": "1",
      "startTime": "2019-05-09 18:33:36",
      "endTime": "2019-05-09 20:33:36"
    },
    {
      "identification": "2",
      "startTime": "2019-05-09 18:33:36",
      "endTime": "2019-05-09 20:33:36"
    }
  ]
}'
```

示例响应:

```
{
  "code": 0,
  "data": null,
  "msg": "添加成功"
}
```

2.8 删除蓝牙开锁权限

URL	DELETE	/api/bluetooth/unlockAuthority.do
参数	application/json	
deviceNum	设备编号	
identifications	密码标识数组	
响应	0-成功,1-失败	

示例请求:

```
curl -X DELETE /api/bluetooth/unlockAuthority.do \
-H 'content-type: application/json' \
-d '{
  "deviceNum": "18100402000208",
  "identifications": ["1", "2"]
}'
```

示例响应:

```
{
  "code": 0,
  "data": null,
  "msg": "删除成功"
}
```

2.9 获取蓝牙开锁密钥

URL	GET	/api/bluetooth/unlockKey.do
参数	application/json	
deviceNum	设备编号	

identification	密码标识
响应	0-成功,1-失败
bluetoothMac	MAC地址
bluetoothPassword	蓝牙密码, 16进制字符串。App发送给门锁前需要转成20个字节数字。
bluetoothCookie	蓝牙Cookie, 16进制字符串。App发送给门锁前需要转成20个字节数字。

示例请求:

```
curl -X GET /api/bluetooth/unlockKey.do?deviceNum=18100402000208&identification=1 \
-H 'content-type: application/json' \
-d ''
```

示例响应:

```
{
  "code": 0,
  "data": {
    "bluetoothMac": "dc9b3d0c602c",
    "bluetoothCookie": "c9b530678ed3282c35f24fbbf14c16dc00000068",
    "bluetoothPassword": "67f66233ba2bd65245c63d0f562ace10000009b"
  },
  "msg": ""
}
```

2.10 获取添加门卡蓝牙指令

App将该指令后发送到门锁, 门锁授予30秒内识别到的第一张门卡开锁权限。

URL	POST	/api/bluetooth/cardAuthority.do
参数	application/json	
deviceNum	设备编号	
identification	门卡标识 最长20	
响应	0-成功,1-失败	
bluetoothMac	MAC地址	
addCardCmd	蓝牙启动本地授权门卡指令, 16进制字符串。App发送给门锁前需要转成20个字节数字。	

示例请求:

```
curl -X POST /api/bluetooth/cardAuthority.do \
-H 'content-type: application/json' \
-d '{
  "deviceNum": "18100402000208",
  "identification": "1",
}'
```

示例响应:

```
{
  "code": 0,
  "data": {
    "bluetoothMac": "dc9b3d0c602c",
    "addCardCmd": "c3b630978fd3282d35e24faaf14c16dc00000068"
  }
}
```

```

    },
    "msg": ""
}

```

2.11 添加门卡

URL	POST	/api/cardInfo.do
参数	application/json	
deviceNum	设备编号	
passwordList	权限标识 (passwordList集合的参数)	
cardType	卡类型 0: 普通门卡 1: 身份证	
cardNum	门卡卡号	
identification	标识 最长20	
响应	data	若没有记录返回空数组
UUID	生成UUID	

示例请求:

```

curl -X POST /api/cardInfo.do \
-H 'content-type: application/json' \
-d '{
    "deviceNum": "181004020000208",
    "passwordList": [
        {
            "identification": "2",
            "cardType": 2,
            "cardNum": "123456"
        },
        {
            "identification": "2",
            "cardType": 1,
            "cardNum": "123132"
        }
    ]
}'

```

示例响应:

```

{
  "code": 0,
  "data": {
    "UUID": "f03d93d2-a9d1-4721-aba9-95b860a35b9d"
  },
  "msg": "添加成功"
}

```

2.12 删除门卡

URL	DELETE	/api/cardInfo.do
参数	application/json	
deviceNum	设备编号	
identifications	标识数组	

响应	0-成功,1-失败
UUID	生成UUID

示例请求:

```
curl -X DELETE /api/cardInfo.do \
-H 'content-type: application/json' \
-d '{
  "deviceNum": "181004020000208",
  "identifications": ["1", "2"]
}'
```

示例响应:

```
{
  "code": 0,
  "data": {
    "UUID": "f03d93d2-a9d1-4721-aba9-95b860a35b9d"
  },
  "msg": "删除成功"
}
```

2.13 添加指纹

URL	POST	/api/fingerprintInfo.do
参数	application/json	
deviceNum	设备编号	
fpNum	指纹编号	
identification	标识 最长20	
响应	data	若没有记录返回空数组

示例请求:

```
curl -X POST /api/fingerprintInfo.do \
-H 'content-type: application/json' \
-d '{
  "deviceNum": "181004020000208",
  "passwordList": [
    {
      "identification": "2",
      "fpNum": 1
    },
    {
      "identification": "2",
      "fpNum": 2
    }
  ]
}'
```

示例响应:

```
{
  "code": 0,
  "data": null,
  "msg": "添加成功"
}
```

2.14 删除指纹

URL	DELETE	/api/fingerprintInfo.do
参数	application/json	
deviceNum	设备编号	
identifications	指纹标识数组	
响应	0-成功,1-失败	

示例请求:

```
curl -X DELETE /api/fingerprintInfo.do \  
-H 'content-type: application/json' \  
-d '{  
    "deviceNum": "181004020000208",  
    "identifications": ["1", "2"]  
}'
```

示例响应:

```
{  
  "code": 0,  
  "data": null,  
  "msg": "删除成功"  
}
```

2.15 远程开锁(远程开锁要先有蓝牙权限)

URL	POST	/api/remoteUnlock.do
参数	application/json	
deviceNum	设备编号	
identification	标识	
响应	0-成功,1-失败	
UUID	生成UUID	

示例请求:

```
curl -X POST /api/remoteUnlock.do \  
-H 'content-type: application/json' \  
-d '{  
    "deviceNum": "181004020000208",  
    "identification": "1"  
}'
```

实例

```
{  
  "code": 0,  
  "data": {  
    "UUID": "f03d93d2-a9d1-4721-aba9-95b860a35b9d"  
  },  
  "msg": "下发成功"  
}
```

2.16 清空指定密码

URL	POST	/api/clearPassword.do
参数	application/json	
deviceNum	设备编号	
authType	0: 恢复出厂, 清除所有开锁方式的权限; 1: 永久密码; 2: 临时密码; 4: 蓝牙扫码权限; 8: 门卡; 16: 指纹; 32:自定义删除密码(蓝牙密码除外)	
cardPassword	门卡 0: 不清除 1: 清除(authType为32时传)	
fingerprintPassword	指纹 0: 不清除 1: 清除(authType为32时传)	
permanentPassword	永久密码 0: 不清除 1: 清除(authType为32时传)	
temporaryPassword	临时密码 0: 不清除 1: 清除(authType为32时传)	
响应	0-成功,1-失败	
UUID	生成UUID	

示例请求:

```
curl -X POST /api/clearPassword.do \
-H 'content-type: application/json' \
-d '{
    "deviceNum": "181004020000208",
    "authType": 0
}'
```

实例

```
{
  "code": 0,
  "data": {
    "UUID": "f03d93d2-a9d1-4721-aba9-95b860a35b9d"
  },
  "msg": "成功"
}
```

2.17 删除门锁指定类型密码

URL	DELETE	/api/deletePasswordInfo.do
参数	application/json	
deviceNum	设备编号	
identifications	数组	
identification	标识	
type	类型: 1: 永久密码; 2: 临时密码; 4: 蓝牙扫码权限; 8: 门卡; 16: 指纹	
响应	0-成功,1-失败	
deviceNum	设备编号	
isSucc	下发同步密码指令操作是否成功 0: 成功 1: 失败	
type	类型: 1: 永久密码; 2: 临时密码; 4: 蓝牙扫码权限; 8: 门卡; 16: 指纹	

UUID	生成UUID (蓝牙权限不用同步UUID默认为空, 其它失败的为空)
------	------------------------------------

示例请求:

```
curl -X DELETE /api/deletePasswordInfo.do \
-H 'content-type: application/json' \
-d '{
  "deviceNum": "18100402000208",
  "identifications": [
    {"identification": "123", type: 1},
    {"identification": "124", type: 2}
  ]
}'
```

实例

```
{
  "code": 0,
  "data": [
    {
      "deviceNum": "867900040227764",
      "isSucc": 0,
      "type": 1,
      "UUID": "7f6905d4-8eb4-45ff-8ddc-a4819bbbed626"
    },
    {
      "deviceNum": "867900040227764",
      "isSucc": 0,
      "type": 2,
      "UUID": "94ccde0f-c259-4983-b5a9-dd68689904b8"
    },
    {
      "deviceNum": "867900040227764",
      "isSucc": 0,
      "type": 4,
      "UUID": ""
    },
    {
      "deviceNum": "867900040227764",
      "isSucc": 0,
      "type": 8,
      "UUID": "8294d1e5-1312-4ff9-a323-ef7997a0a678"
    },
    {
      "deviceNum": "867900040227764",
      "isSucc": 0,
      "type": 16,
      "UUID": "36139361-a56b-4c3a-98e1-ebf59a49eac4"
    }
  ],
  "msg": "删除权限成功!"
}
```

2.18 获取蓝牙开锁密钥(无需授权)

URL	GET	/api/bluetooth/unAuthorizeUnlockKey.do
参数	application/json	
deviceNum	设备编号	
响应	0-成功,1-失败	
bluetoothMac	MAC地址	

bluetoothPassword	蓝牙密码，16进制字符串。App发送给门锁前需要转成20个字节数字。
bluetoothCookie	蓝牙Cookie，16进制字符串。App发送给门锁前需要转成20个字节数字。

示例请求:

```
curl -X GET /api/bluetooth/unAuthorizeUnlockKey.do?deviceNum=181004020000208 \
-H 'content-type: application/json' \
-d ''
```

示例响应:

```
{
  "code":0,
  "data":{
    "bluetoothMac":"dc9b3d0c602c",
    "bluetoothCookie":"c9b530678ed3282c35f24fbbf14c16dc0000068",
    "bluetoothPassword":"67f66233ba2bd65245c63d0f562ace10000009b"
  },
  "msg":""
}
```

2.19 查询指纹数据

URL	GET	/api/fingerprintInfoList.do
参数	application/json	
deviceNum	设备编号(必填)	
响应	0-成功,1-失败	
mobile	标示	

示例请求:

```
curl -X GET /api/fingerprintInfoList.do?deviceNum=960010102201000008 \
-H 'content-type: application/json' \
-d ''
```

示例响应:

```
{
  "code": 0,
  "data": [
    {
      "mobile": "963f454211e54ac5ae2"
    },
    {
      "mobile": "a7af7e6f22e34426b15"
    },
    {
      "mobile": "8046a7e30c76432b881"
    }
  ],
  "msg": "succ"
}
```

2.20 按锁编号查询临时密码

URL	GET	/api/temporaryPassword.do
参数	application/json	
deviceNum	设备编号(必填)	
响应	0-成功,1-失败	
createAt	创建时间	
deviceNum	设备编号	
mobile	标示	
id	锁内用户ID	
validTimeStart	有效期开始时间	
validTimeEnd	有效期结束时间	

示例请求:

```
curl -X GET /api/temporaryPassword.do?deviceNum=181001020000042 \
-H 'content-type: application/json' \
-d ''
```

示例响应:

```
{
  "code": 0,
  "data": [
    {
      "createAt": "2020-07-06 15:23:50",
      "deviceNum": "181001020000042",
      "id": 1552,
      "mobile": "16471110",
      "validTimeEnd": "2020-09-30 09:34:59",
      "validTimeStart": "2020-08-25 09:34:59"
    },
    {
      "createAt": "2020-07-06 14:56:20",
      "deviceNum": "181001020000042",
      "id": 1548,
      "mobile": "16470584",
      "validTimeEnd": "2020-09-30 09:34:59",
      "validTimeStart": "2020-08-25 09:34:59"
    },
    {
      "createAt": "2020-07-06 14:55:47",
      "deviceNum": "181001020000042",
      "id": 1547,
      "mobile": "16470556",
      "validTimeEnd": "2020-09-30 09:34:59",
      "validTimeStart": "2020-08-25 09:34:59"
    }
  ],
  "msg": "succ"
}
```

2.21 按锁编号查询永久密码

URL	GET	/api/permanentPassword.do
参数	application/json	

deviceNum	设备编号(必填)
响应	0-成功,1-失败
deviceNum	设备编号
mobile	标示
id	锁内用户ID
createAt	创建时间

示例请求:

```
curl -X GET /api/permanentPassword.do?deviceNum=181001020000042 \
-H 'content-type: application/json' \
-d ''
```

示例响应:

```
{
  "code": 0,
  "data": [
    {
      "createAt": "2020-07-06 10:55:26",
      "deviceNum": "181001020000042",
      "id": 1079,
      "mobile": "13528714370"
    }
  ],
  "msg": "succ"
}
```

2.22 按锁编号查询蓝牙权限数据

URL	GET	/api/bluetooth/unlockAuthority.do
参数	application/json	
deviceNum	设备编号(必填)	
响应	0-成功,1-失败	
deviceNum	设备编号	
mobile	标示	
createAt	创建时间	
validStart	有效期开始时间	
validEnd	有效期结束时间	

示例请求:

```
curl -X GET /api/bluetooth/unlockAuthority.do?deviceNum=181001020000042 \
-H 'content-type: application/json' \
-d ''
```

示例响应:

```
{
  "code": 0,
```

```

    "data": [
      {
        "createAt": "2020-08-03 09:56:06",
        "deviceNum": "18100102000042",
        "mobile": "13528714371",
        "validEnd": "2020-08-11 05:25:17",
        "validStart": "2020-08-03 09:56:00"
      },
      {
        "createAt": "2020-07-31 16:20:17",
        "deviceNum": "18100102000042",
        "mobile": "13528714370",
        "validEnd": "2020-08-04 21:45:00",
        "validStart": "2020-07-31 16:20:10"
      }
    ],
    "msg": "succ"
  }

```

2.23 按锁查询临时密码使用情况

URL	GET	/api/temporaryPasswordCount.do
参数	application/json	
deviceNum	设备编号(必填)	
响应	0-成功,1-失败	
totalQuantity	总数量	
usedQuantity	已使用数量	
availableQuantity	可用数量	

示例请求:

```

curl -X GET /api/temporaryPasswordCount.do?deviceNum=18100102000042 \
-H 'content-type: application/json' \
-d ''

```

示例响应:

```

{
  "code": 0,
  "data": {
    "availableQuantity": 7,
    "totalQuantity": 10,
    "usedQuantity": 3
  },
  "msg": "succ"
}

```

2.24 按锁查询永久密码使用情况

URL	GET	/api/permanentPasswordCount.do
参数	application/json	
deviceNum	设备编号(必填)	
响应	0-成功,1-失败	

totalQuantity	总数量
usedQuantity	已使用数量
availableQuantity	可用数量

示例请求:

```
curl -X GET /api/permanentPasswordCount.do?deviceNum=181001020000042 \
-H 'content-type: application/json' \
-d ''
```

示例响应:

```
{
  "code": 0,
  "data": {
    "availableQuantity": 15,
    "totalQuantity": 16,
    "usedQuantity": 1
  },
  "msg": "succ"
}
```

2.25 按锁查询多次输入错误密码报警日志记录

URL	GET	/api/wrongPasswordAlarmLog.do
参数	application/json	
deviceNum	设备编号(必填)	
pageNum	当前页(默认为第1页)	
pageSize	每页显示数量(默认10条)	
startTime	开始时间	
endTime	结束时间	
响应	0-成功,1-失败	
alertTime	报警时间	
createAt	保存时间	

示例请求:

```
curl -X GET /api/wrongPasswordAlarmLog.do?deviceNum=181001020000042 \
-H 'content-type: application/json' \
-d ''
```

示例响应:

```
{
  "code": 0,
  "data": {
    "pageNum": 1,
    "pageSize": 10,
    "total": 2,
    "list": [
      {
        "alertTime": "2020-08-03 17:56:16",
```

```

        "createAt": "2020-08-03 17:56:19"
      },
      {
        "alertTime": "2020-08-03 14:47:40",
        "createAt": "2020-08-03 14:47:44"
      }
    ]
  },
  "msg": "succ"
}

```

2.26 按锁编号和标识查询临时密码数据

URL	GET	/api/findTemporaryPasswordData.do
参数	application/json	
deviceNum	设备编号(必填)	
identification	标识(必填)	
响应	0-成功,1-失败	
startTime	开始时间	
endTime	结束时间	
identification	标识	
optId	临时密码ID	

示例请求:

```

curl -X GET /api/findTemporaryPasswordData.do?deviceNum=181001020000042&identification=16470584 \
-H 'content-type: application/json' \
-d ''

```

示例响应:

```

{
  "code": 0,
  "data": {
    "identification": "16470584",
    "startTime": "2020-08-25 09:34:59",
    "endTime": "2020-09-30 09:34:59",
    "optId": 1548
  },
  "msg": "succ"
}

```

2.27 拉取推送失败或没有推送的数据

URL	GET	/api/pullPushData.do
参数	application/json	
deviceNum	设备编号(必填)	
pageNum	当前页 (默认第一页)	
pageSize	每页显示数量 (默认10条)	
alreadyPushNum	推送次数 0: 推送次过多后没有推送 其它(如 1): 有推送过, 都失败了	

startTime	开始时间(必填)
endTime	结束时间(必填)
pullType	拉取状态 0:未拉取 1: 已拉取(必填)
响应	0-成功,1-失败
deviceNum	设备编号
isSucc	推送成功 0: 成功 1: 失败
createTime	创建时间
alreadyPushNum	已经推送次数
content	消息内容
url	URL

示例请求:

```
curl -X GET /api/pullPushData.do?deviceNum=181001020000042&startTime=2020-08-01 12:12:36&endTime=2020-08-18 12:12:36&pageNum=1&pageNum=1&pullType=1
-H 'content-type: application/json' \
-d ''
```

示例响应:

```
{
  "code":0,
  "data":{
    "pageSize":10,
    "pageNum":1,
    "total":4,
    "list":[
      {
        "deviceNum":"181001020000042",
        "isSucc":1,
        "createTime":"2020-08-03 10:30:04",
        "alreadyPushNum":3,
        "content":{"data":{"deviceNum":"181001020000042","signalStrength":48,"dataType":"0","battery":100}},
        "url":"http://speedcn.vicp.cc:8085/sl/device/receive"
      },
      {
        "deviceNum":"181001020000042",
        "isSucc":1,
        "createTime":"2020-08-03 10:30:04",
        "alreadyPushNum":3,
        "content":{"data":{"deviceNum":"181001020000042","signalStrength":48,"dataType":"0","battery":100}},
        "url":"http://speedcn.vicp.cc:8085/sl/device/receive"
      },
      {
        "deviceNum":"181001020000042",
        "isSucc":1,
        "createTime":"2020-08-03 10:30:05",
        "alreadyPushNum":3,
        "content":{"data":{"deviceNum":"181001020000042","signalStrength":51,"dataType":"0","battery":100}},
        "url":"http://speedcn.vicp.cc:8085/sl/device/receive"
      },
      {
        "deviceNum":"181001020000042",
        "isSucc":1,
        "createTime":"2020-08-03 10:30:06",
        "alreadyPushNum":3,

```

```

        "content": "{ \"data\": { \"deviceNum\": \"18100102000042\", \"signalStrength\": 48, \"dataType\": \"\", \"battery\": 100 },
        \"url\": \"http://speedcn.vicp.cc:8085/sl/device/receive\"
    }
}
},
\"msg\": \"succ\"
}

```

2.28 配置把手朝向

URL	PUT	/api/configurationHandle.do
参数	application/json	
deviceNum	设备编号(必填)	
doorknob	门把手是左还是右 false是右, true 是左(必填)	
响应	0-成功,1-失败	

示例请求:

```

curl -X PUT /api/pullPushData.do \
-H 'content-type: application/json' \
-d '{
    "deviceNum": "930010112202400003",
    "doorknob": "true"
}'

```

示例响应:

```

{
  "code": 0,
  "data": null,
  "msg": "配置成功!"
}

```

2.29 查询门锁所有分类各自的日志记录总数

URL	GET	/api/lockClassificationCount.do
参数	application/json	
deviceNum	设备编号(必填)	
alertType	1: 查询报警分类数据 2: 查询心跳分类数据	
dataType	102:低电量 103:撬锁 105:长时间未关 106:蓝牙密码错误报警 108:非法开门报警 109:密码错误自锁报警 110:蓝牙损坏 201:启动后第一次上报 202:心跳上报 203:按键唤醒 204:用于时间校准 205:门磁状态变化	
startTime	开始时间	
endTime	结束时间	
响应	0-成功,1-失败	

示例请求:

```

curl -X GET /api/lockClassificationCount.do?deviceNum=18100102000042&alertType=2&dataType=201&startTime=2020-08-05 13:23:25&endTi
-H 'content-type: application/json' \

```

```
-d ''
```

示例响应:

```
{
  "code":0,
  "data":{
    "deviceNum":"181001020000042",
    "total":3,
    "dataType":201
  },
  "msg":"succ"
}
```

2.30 查询下发密码到锁的时长

URL	GET	/api/sendDataDuration.do
参数	application/json	
deviceNum	设备编号(必填)	
pageNum	当前页 (默认第一页)	
pageSize	每页显示数量 (默认10条)	
dataType	命令类型 1-密码 2-指纹 3-临时密码 7-卡开锁	
startTime	开始时间	
endTime	结束时间	
uuid	UUID	
响应	0-成功,1-失败	
deviceNum	设备编号	
commandType	命令类型 1-密码 2-指纹 3-临时密码 7-卡开锁	
sendStatus	发送状态 0: 待发送 1: 已下发 2: 取消发送 3: 发送成功 4:下发失败 5:发送成功并有回应 6: 取消失败 7:取消并有回应	
updateTime	成功时间	
createAt	下发时间	
pushUuid	推送UUID	

示例请求:

```
curl -X GET /api/sendDataDuration.do?deviceNum=940020113193400004&startTime=2020-08-27 12:12:36&endTime=2020-08-28 18:12:36&pageNum=1
-H 'content-type: application/json' \
-d ''
```

示例响应:

```
{
  "code":0,
  "data":{
    "pageNum":1,
    "pageSize":10,
  }
}
```

```

"total":5,
"list":[
  {
    "deviceNum":"940020113193400004",
    "commandType":1,
    "sendStatus":5,
    "updateTime":"2020-08-27 16:00:20",
    "createAt":"2020-08-27 15:59:50",
    "pushUuid":"dbc1bed1-7ca0-4761-90c5-39b183df0dda"
  },
  {
    "deviceNum":"940020113193400004",
    "commandType":1,
    "sendStatus":7,
    "updateTime":"2020-08-27 16:00:20",
    "createAt":"2020-08-27 15:59:50",
    "pushUuid":"71f1ee90-f559-4487-8b40-acda66d6e95c"
  },
  {
    "deviceNum":"940020113193400004",
    "commandType":1,
    "sendStatus":7,
    "updateTime":"2020-08-27 16:00:20",
    "createAt":"2020-08-27 15:59:37",
    "pushUuid":"0a9246b9-f225-49af-a7e6-c82be965c66b"
  },
  {
    "deviceNum":"940020113193400004",
    "commandType":1,
    "sendStatus":7,
    "updateTime":"2020-08-27 16:00:20",
    "createAt":"2020-08-27 15:59:06",
    "pushUuid":"e8230cb8-f97a-474b-b3f7-0e6352225877"
  },
  {
    "deviceNum":"940020113193400004",
    "commandType":1,
    "sendStatus":7,
    "updateTime":"2020-08-27 16:00:20",
    "createAt":"2020-08-27 15:58:59",
    "pushUuid":"9b10434d-dc35-4d99-8d49-b2c871bcb4cb"
  }
]
},
"msg":"succ"
}

```

2.31 添加离线密码

URL	POST	/api/offlinePassword.do
参数	application/json	
deviceNum	设备编号(必填)	
identification	用户标示	
passTime	计算好的有效时间 范围：1到999(第一位为5以下则代表分钟数，第一位为5或以上，则应减去5后代表小时数)	
响应	0-成功,1-失败	

示例请求:

```
curl -X POST /api/offlinePassword.do \
```



```
-H 'content-type: application/json' \  
-d '{  
  "deviceNum": "960610132203000030",  
  "identification": "123465",  
  "passTime": 020  
}'
```

示例响应:

```
{  
  "code": 0,  
  "data": {  
    "password": "514548020"  
  },  
  "msg": "添加成功!"  
}
```

3 电表管理

3.1 查询电表信息

URL	GET	/api/meterList.do
参数	application/json	
meterNum	电表编号	
pageNum	当前页 (必填)	
pageSize	每页显示数量 (必填)	
响应	智能锁基本信息	
meterNum	电表编号	
degree	度数	
sluiceGate	开合闸状态 0为合,1为跳	
meterType	电表厂商 1:移动 2:北电	

示例请求:

```
curl -X GET /api/meterList.do?pageNum=1&pageSize=10 \  
-H 'content-type: application/json' \  
-d ''
```

实例

```
{  
  "code": 0,  
  "data": {  
    "pageNum": 1,  
    "pageSize": 10,  
    "total": 1,  
    "list": [  
      {  
        "sluiceGate": 0,  
        "meterType": 1,  
        "degree": 0,  
        "meterNum": "5130001000014001606057"  
      }  
    ]  
  }  
}
```

```
},
  "msg": "succ"
}
```

3.2 查询电表历史信息

URL	GET	/api/meterHistoryInfo.do
参数	application/json	
meterNum	电表编号	
pageNum	当前页（必填）	
pageSize	每页显示数量（必填）	
响应	智能锁基本信息	
meterNum	电表编号	
kwh	有功总电能	
sluiceGate	开合闸状态 0为合,1为跳	
createTime	创建时间	

示例请求:

```
curl -X GET /api/meterHistoryInfo.do?pageNum=1&pageSize=10 \
-H 'content-type: application/json' \
-d ''
```

实例

```
{
  "code": 0,
  "data": {
    "pageNum": 1,
    "pageSize": 10,
    "total": 1,
    "list": [
      {
        "sluiceGate": 0,
        "createTime": "2019-03-28 11:24:36",
        "kwh": 0,
        "meterNum": "5130001000014001606057"
      }
    ]
  },
  "msg": "succ"
}
```

3.3 查询电表告警信息

URL	GET	/api/meterAlarmList.do
参数	application/json	
meterNum	电表编号	
pageNum	当前页（必填）	
pageSize	每页显示数量（必填）	

响应	智能锁基本信息
meterNum	电表编号
alarmType	0:过流告警 1:开表盖告警 2: 电能表清零告警 3:其它告警
alarmInfo	告警内容
createTime	创建时间

示例请求:

```
curl -X GET /api/meterAlarmList.do?pageNum=1&pageSize=10 \
-H 'content-type: application/json' \
-d ''
```

实例

```
{
  "code":0,
  "data":{
    "pageNum":1,
    "pageSize":10,
    "total":1,
    "list":[
      {
        "alarmType":0,
        "alarmInfo":"123",
        "createTime":"2019-03-28 11:24:36",
        "meterNum":"5130001000014001606057"
      }
    ]
  },
  "msg":"succ"
}
```

3.4 下发电表指令

URL	POST	/api/sendMeterCommand.do
参数	application/json	
meterNum	电表编号	
commandType	类型 0: 合闸 1: 跳闸	
响应	0-成功,1-失败	
UUID	生成UUID	

示例请求:

```
curl -X POST /api/sendMeterCommand.do \
-H 'content-type: application/json' \
-d '{
  "meterNum":"181004020000208",
  "commandType":0
}'
```

实例

```
{
  "code": 0,
```

```

"data": {
  "UUID": "f03d93d2-a9d1-4721-aba9-95b860a35b9d"
},
"msg": "成功"
}

```

4 水表管理

4.1 查询水表信息

URL	GET	/api/waterMeterList.do
参数	application/json	
waterMeterNum	水表编号	
pageNum	当前页（必填）	
pageSize	每页显示数量（必填）	
响应	智能锁基本信息	
waterMeterNum	水表编号	
accumulatedDosage	累计用量	
sluiceGate	开合闸状态 0:开闸 1:合闸	

示例请求:

```

curl -X GET /api/waterMeterList.do?pageNum=1&pageSize=10 \
-H 'content-type: application/json' \
-d ''

```

实例

```

{
  "code": 0,
  "data": {
    "pageNum": 1,
    "pageSize": 10,
    "total": 1,
    "list": [
      {
        "sluiceGate": 0,
        "accumulatedDosage": 0,
        "waterMeterNum": "5130001000014001606057"
      }
    ]
  },
  "msg": "succ"
}

```

4.2 查询水表历史信息

URL	GET	/api/waterMeterHistoryInfo.do
参数	application/json	
waterMeterNum	水表编号	
pageNum	当前页（必填）	

pageSize	每页显示数量（必填）
响应	智能锁基本信息
waterMeterNum	水表编号
accumulatedDosage	累计用量
sluiceGate	开合闸状态 0:开闸 1:合闸
createTime	创建时间

示例请求:

```
curl -X GET /api/waterMeterHistoryInfo.do?pageNum=1&pageSize=10 \
-H 'content-type: application/json' \
-d ''
```

实例

```
{
  "code":0,
  "data":{
    "pageNum":1,
    "pageSize":10,
    "total":1,
    "list":[
      {
        "sluiceGate":0,
        "createTime":"2019-03-28 11:24:36",
        "accumulatedDosage":0,
        "waterMeterNum":"5130001000014001606057"
      }
    ]
  },
  "msg":"succ"
}
```

4.3 查询水表告警信息

URL	GET	/api/waterMeterAlarmList.do
参数	application/json	
waterMeterNum	水表编号	
pageNum	当前页（必填）	
pageSize	每页显示数量（必填）	
响应	智能锁基本信息	
waterMeterNum	水表编号	
alarmType	告警类型	
alarmInfo	告警内容	
createTime	创建时间	

示例请求:

```
curl -X GET /api/waterMeterAlarmList.do?pageNum=1&pageSize=10 \
```

```
-H 'content-type: application/json' \  
-d ''
```

实例

```
{  
  "code":0,  
  "data":{  
    "pageNum":1,  
    "pageSize":10,  
    "total":1,  
    "list":[  
      {  
        "alarmType":0,  
        "alarmInfo":"123",  
        "createTime":"2019-03-28 11:24:36",  
        "waterMeterNum":"5130001000014001606057"  
      }  
    ]  
  },  
  "msg":"succ"  
}
```

4.4 下发水表指令

URL	POST	/api/sendWaterMeterCommand.do
参数	application/json	
waterMeterNum	水表编号	
commandType	类型 0: 开闸 1: 合闸	
响应	0-成功,1-失败	
UUID	生成UUID	

示例请求:

```
curl -X POST /api/sendWaterMeterCommand.do \  
-H 'content-type: application/json' \  
-d '{  
  "waterMeterNum":"181004020000208",  
  "commandType":0  
}'
```

实例

```
{  
  "code": 0,  
  "data": {  
    "UUID": "f03d93d2-a9d1-4721-aba9-95b860a35b9d"  
  },  
  "msg": "成功"  
}
```

5 加解密管理

5.1 加密管理

URL	POST	/api/aesEncryption.do
-----	------	-----------------------

参数	application/json
encryptionInfo	加密信息
响应	信息
data	加密后信息

示例请求:

```
curl -X POST /api/aesEncryption.do \
-H 'content-type: application/json' \
-d '{
  "encryptionInfo": "1,2,1,50,48,48,51,48,49,136,17,0,0,0,0"
}'
```

实例

```
{
  "code": 0,
  "data": "c9,42,52,4f,f7,ab,dc,12,9a,4b,f4,79,dd,74,16,9a",
  "msg": "succ"
}
```

5.2 解密管理

URL	POST	/api/aesDecrypt.do
参数	application/json	
encryptionInfo	加密信息	
响应	信息	
data	加密后信息	

示例请求:

```
curl -X POST /api/aesDecrypt.do \
-H 'content-type: application/json' \
-d '{
  "decryptInfo": "3,11,41,55,215,26,207,154,30,173,106,39,11,176,152,171"
}'
```

实例

```
{
  "code": 0,
  "data": "01,c2,00,00,00,00,00,00,00,00,00,00,00,00,00",
  "msg": "succ"
}
```

5.3 CRC8校验值

URL	POST	/api/crcCheck.do
参数	application/json	
crcValue	校验信息	
响应	智能锁基本信息	

data	校验值
------	-----

示例请求:

```
curl -X POST /api/crcCheck.do \
-H 'content-type: application/json' \
-d '{
  "crcValue ": "1,2,0,54,48,50,50,51,52,136,17,0,0,0,0"
}'
```

实例

```
{
  "code":0,
  "data":"23",
  "msg":"succ"
}
```

6 门禁锁管理

6.1 查询门禁锁信息

URL	GET	/api/guardList.do
参数	application/json	
guardNum	门禁锁编号	
pageNum	当前页 (必填)	
pageSize	每页显示数量 (必填)	
响应	智能锁基本信息	
guardNum	门禁锁编号	
guardStatus	门禁锁状态 0: 处于关闭状态 1: 处于打开状态	
createTime	创建时间	

示例请求:

```
curl -X GET /api/guardList.do?pageNum=1&pageSize=10 \
-H 'content-type: application/json' \
-d ''
```

实例

```
{
  "code":0,
  "data":{
    "pageNum":1,
    "pageSize":10,
    "total":1,
    "list":[
      {
        "createTime":"2019-08-12 14:07:05.0",
        "guardId":1,
        "guardMac":"01aabc21",
        "adminId":null,
        "guardNum":"aa010a0b00ff",
        "guardStatus":1
      }
    ]
  }
}
```



```

    }
  ]
},
"msg": "succ"
}

```

6.2 下发门禁锁指令

URL	POST	/api/sendGuardCommand.do
参数	application/json	
guardNum	门禁锁编号	
identification	用户标识(标识最长为30个字节)	
commandType	类型 1:远程开锁 2:设置门锁模式	
lockMode	设置门锁模式 0: 常关 1: 常开 2: 恢复正常使用(在设置门锁模式时才传)	
响应	0-成功,1-失败	
UUID	生成UUID	

示例请求:

```

curl -X POST /api/sendGuardCommand.do \
-H 'content-type: application/json' \
-d '{
    "guardNum": "186921440182527",
    "commandType": 2,
    "lockMode": 2,
    "identification": "userName"
}'

```

实例

```

{
  "code": 0,
  "data": {
    "UUID": "f03d93d2-a9d1-4721-aba9-95b860a35b9d"
  },
  "msg": "成功"
}

```

6.3 添加门禁卡

URL	POST	/api/guardCard.do
参数	application/json	
guardNum	门禁锁编号	
identification	用户(门卡)标识(标识最长为30个字节)	
cardNum	门卡卡号	
cardRemarks	门卡备注	
startTime	生效时间(云丹门禁默认添加即生效)	
endTime	失效时间	

响应	0-成功,1-失败
UUID	生成UUID

示例请求:

```
curl -X POST /api/guardCard.do \
-H 'content-type: application/json' \
-d '{
  "guardNum": "003e05006189",
  "identification": "2345622",
  "cardNum": "123453",
  "cardRemarks": "备注",
  "startTime": "2020-08-12 12:33:36",
  "endTime": "2020-08-15 20:33:36"
}'
```

实例

```
{
  "code": 0,
  "data": {
    "UUID": "f03d93d2-a9d1-4721-aba9-95b860a35b9d"
  },
  "msg": "添加门禁卡成功!"
}
```

6.4 删除门禁卡

URL	DELETE	/api/guardCard.do
参数	application/json	
guardNum	门禁锁编号	
identification	用户(门卡)标识(标识最长为30个字节)	
响应	0-成功,1-失败	
UUID	生成UUID	

示例请求:

```
curl -X DELETE /api/guardCard.do \
-H 'content-type: application/json' \
-d '{
  "guardNum": "003e05006189",
  "identification": "2345621"
}'
```

实例

```
{
  "code": 0,
  "data": {
    "UUID": "f03d93d2-a9d1-4721-aba9-95b860a35b9d"
  },
  "msg": "删除门禁卡成功!"
}
```

6.5 删除全部门禁卡

--	--	--

URL	DELETE	/api/de1AllGuardCard.do
参数	application/json	
guardNum	门禁锁编号	
identification	用户(门卡)标识(标识最长为30个字节)	
cardNum	门卡卡号	
cardRemarks	门卡备注	
startTime	生效时间(云丹门禁默认添加即生效)	
endTime	失败时间	
响应	0-成功,1-失败	
UUID	生成UUID	

示例请求:

```
curl -X DELETE /api/de1AllGuardCard.do \
-H 'content-type: application/json' \
-d '{
  "guardNum": "003e05006189"
}'
```

实例

```
{
  "code": 0,
  "data": {
    "UUID": "f03d93d2-a9d1-4721-aba9-95b860a35b9d"
  },
  "msg": "删除全部门禁卡成功!"
}
```

6.6 添加门禁临时密码

URL	POST	/api/guard0tp.do
参数	application/json	
guardNum	门禁锁编号	
identification	用户标识(标识最长为30个字节)	
password	临时密码(长度为6位)	
startTime	生效时间(云丹门禁默认添加即生效)	
endTime	失败时间	
响应	0-成功,1-失败	
UUID	生成UUID	

示例请求:

```
curl -X POST /api/guard0tp.do \
-H 'content-type: application/json' \
-d '{
  "guardNum": "003e05006189",
```

```
"identification": "2345621",
"password": "123452",
"startTime": "2020-08-12 09:33:36",
"endTime": "2020-08-15 20:33:36"
}'
```

实例

```
{
  "code":0,
  "data":{
    "UUID":"f03d93d2-a9d1-4721-aba9-95b860a35b9d"
  },
  "msg":"添加门禁临时密码成功!"
}
```

6.7 删除门禁临时密码

URL	DELETE	/api/guardOtp.do
参数	application/json	
guardNum	门禁锁编号	
identification	用户标识(标识最长为30个字节)	
响应	0-成功,1-失败	
UUID	生成UUID	

示例请求:

```
curl -X DELETE /api/guardOtp.do \
-H 'content-type: application/json' \
-d '{
  "guardNum":"003e05006189",
  "identification": "2345621"
}'
```

实例

```
{
  "code":0,
  "data":{
    "UUID":"f03d93d2-a9d1-4721-aba9-95b860a35b9d"
  },
  "msg":"删除门禁临时密码成功!"
}
```

6.8 删除全部临时密码

URL	DELETE	/api/de1AllGuardOtp.do
参数	application/json	
guardNum	门禁锁编号	
identification	用户(门卡)标识(标识最长为30个字节)	
cardNum	门卡卡号	
cardRemarks	门卡备注	

startTime	生效时间(云丹门禁默认添加即生效)
endTime	失败时间
响应	0-成功,1-失败
UUID	生成UUID

示例请求:

```
curl -X DELETE /api/delAllGuardOtp.do \
-H 'content-type: application/json' \
-d '{
  "guardNum": "003e05006189"
}'
```

实例

```
{
  "code": 0,
  "data": {
    "UUID": "f03d93d2-a9d1-4721-aba9-95b860a35b9d"
  },
  "msg": "删除全部门禁临时密码成功!"
}
```

7 状态说明

7.1 状态说明

Code	对应说明
0	成功
1	参数错误
2	设备不存在
3	设备没有注册
4	该用户没有此设备操作权限
5	已经添加了该用户密码
6	权限还未到有效使用时间
7	操作失败
8	已有权限
9	下发指令失败
10	账户或密码错误
11	用户标识重复
12	您输入的密码错误次数过多
13	蓝牙密码为空, 请新重启设备后在试
15	参数格式错误
20	服务器内部错误

