

# 庖丁解车 4S 查询 系统接入说明

广州越端信息技术有限公司

2016 年 10 月 12 日

# 目录

目录 .....	2
文档变更历史 .....	5
名词说明 .....	5
通讯规范 .....	5
协议 .....	5
域名 .....	6
安全策略 .....	6
概念说明 .....	7
接口版本控制（当前版本 1.0） .....	7
接口定义 .....	7
接口一：生成令牌 .....	7
接口地址 .....	7
注意 .....	7
协议 .....	7
传入参数 .....	8
请求头 .....	8
请求头示例 .....	8
返回内容 .....	8
返回示例 .....	8
接口二：刷新令牌时长 .....	9
接口地址 .....	9
注意 .....	9
协议 .....	9
传入参数 .....	9
请求头 .....	9
请求头示例 .....	10
返回内容 .....	10
返回示例 .....	10
接口三：品牌获取 .....	11
接口地址 .....	11
注意 .....	11
协议 .....	11
传入参数 .....	11
请求头 .....	11
请求头示例 .....	11
返回内容 .....	12
返回示例 .....	12

接口四：品牌厂商列表.....	13
接口地址.....	13
注意.....	13
协议.....	13
传入参数.....	13
请求头.....	13
请求头示例.....	13
返回内容.....	14
返回示例.....	14
接口五：获取单个品牌厂商.....	15
接口地址.....	15
注意.....	15
协议.....	15
传入参数.....	15
请求头.....	15
请求头示例.....	15
返回内容.....	16
返回示例.....	16
接口六：购买订单.....	17
订单处理时序图.....	17
接口地址.....	18
注意.....	18
协议.....	18
传入参数.....	18
请求头.....	18
请求头示例.....	19
返回内容.....	19
返回示例.....	19
接口七：查询订单结果.....	20
协议.....	20
注意.....	20
接口地址.....	20
传入参数.....	20
请求头.....	20
请求头示例.....	20
返回内容.....	21
返回示例.....	24
接口八：异步结果通知.....	26
协议.....	26
接口地址.....	26
传入参数.....	26
返回值.....	26
重试机制.....	26
业务状态码：.....	27
通用.....	27
品牌厂商错误状态.....	27

---

订单错误状态.....27

## 文档变更历史

版本	日期	作者	变更内容描述
V1.0	2016-09-12	王帅	发布 1.0 版本
V1.1	2016-09-30	王帅	细节点调整
V1.2	2016-10-10	王帅	请求头中的下划线改成中划线
V1.3	2016-10-12	王帅	原先的名称定义“品牌”由于含义不明确，改为“品牌厂商”；增加获取品牌接口
V1.4	2017-01-11	黄雪冬	增加订单结果内容。增加 resultType，增加 resultDetail。
V1.5	2017-04-20	黄雪冬	增加里程表是否异常、安全气囊是否异常、发动机是否异常等描述
V1.6	2017-06-23	钟翔	增加由 vin 获取的车辆信息
V1.7	2017-07-17	钟翔	查询订单结果中的事故次数字段修改，统一赋值成空(null)
V1.8	2017-08-15	钟翔	订单结果中添加出险记录

## 名词说明

合作商：接入该系统的客户

## 通讯规范

### 协议

https

## 域名

正式: <https://openapi4s.paodingcar.com>

测试: <http://openapi4s.beitaijf.cn>

## 安全策略

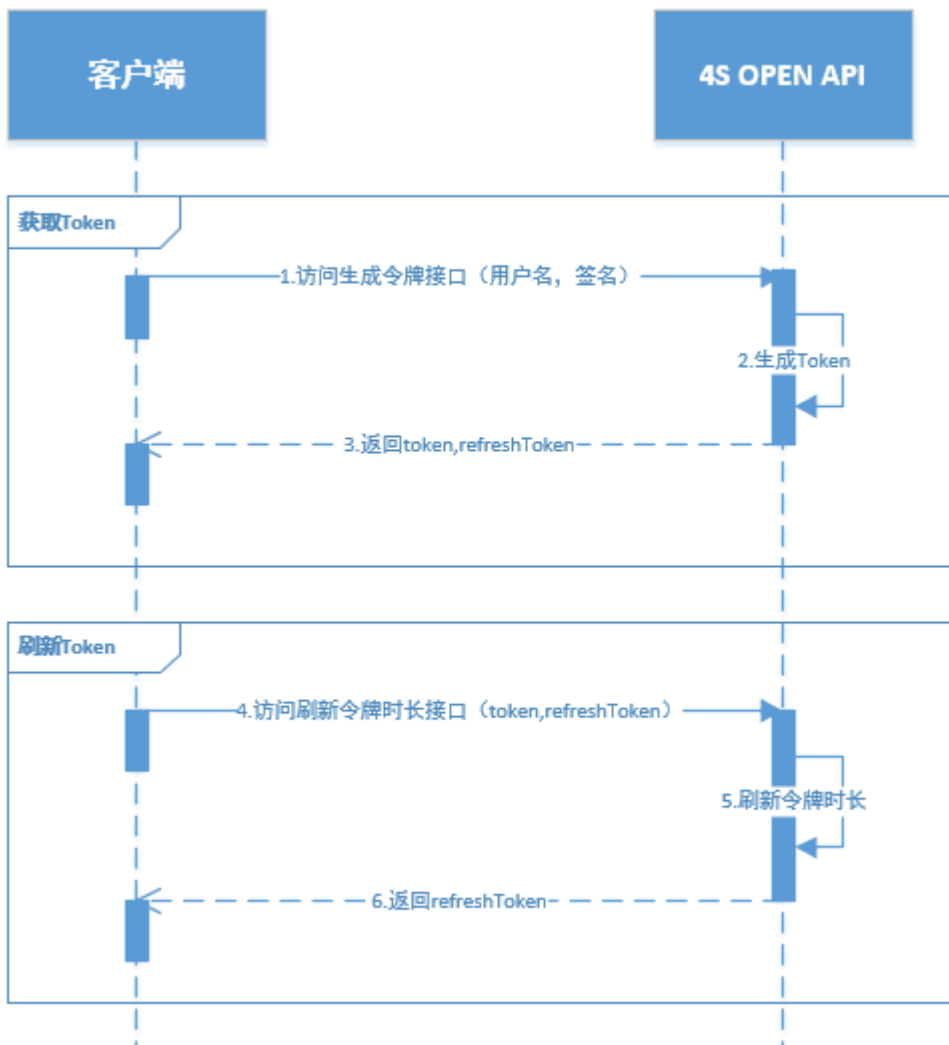
采用 OAuth 2.0 作为授权协议，授权流程可以简单归纳为：

(1) 通过"获取令牌(/4s/token/get)"接口获取 AuthorizationToken 和 RefreshToken，调用其它接口时将 AuthorizationToken 放在请求头中用于验证

(2) AuthorizationToken 有效时长 2 小时，通过调用"刷新令牌时长"，重置令牌有效时间

(3) 如果 AuthorizationToken 验证失败，需调用"获取令牌(/4s/token/get)"接口重新获取

时序图



## 概念说明

AuthorizationToken:调用业务接口时用于验证合作商信息合法性

RefreshToken 令牌:用于重置 AuthorizationToken 时长，只能使用一次

## 接口版本控制（当前版本 1.0）

调用接口时，在请求头中附加接口的版本号

注意：接口版本不等同文档的版本，两者是不同的概念

接口版本变更附录：

版本	日期	变更内容描述
1.0	2016-09-12	发布 1.0 版本

## 接口定义

### 接口一：生成令牌

#### 接口地址

/4s/token/generate

#### 注意

生产环境限制每天调用 1000 次，测试环境限制每天 5000 次

#### 协议

POST

## 传入参数

参数名称	参数类型	是否必填	参数说明
userName	字符串(最长 15 位)	是	用户名
sign	字符串(32 位)	是	签名=md5(用户名+md5(密码))

## 请求头

参数名称	参数类型	参数说明
Api-4s-Version	字符串(最长 5 位)	接口版本

## 请求头示例

Api-4s-Version:1.0

## 返回内容

参数名称	参数类型	参数说明
authorizationToken	字符串(32 位)	授权令牌
refreshToken	字符串(32 位)	刷新令牌
effectiveDuration	整数	授权令牌有效时长，单位(分钟)

## 返回示例

成功示例

```
{
  "code": "10000",
  "remark": "成功",
  "data": {
    "token": "ABC",
    "refreshToken": "DEF",
    "effectiveDuration": "120"
  }
}
```

失败示例，如果用户名或者签名错误，返回的是



```
{  
  "code": "10020",  
  "remark": "账户错误",  
  "data":  
    {  
    }  
}
```

## 接口二：刷新令牌时长

### 接口地址

/4s/token/refresh

### 注意

生产环境限制每天调用 1000 次，测试环境限制每天 5000 次

### 协议

POST

### 传入参数

参数名称	参数类型	是否必填	参数说明
refreshToken	字符串(32 位)	是	刷新令牌

### 请求头

参数名称	参数类型	参数说明
Authorization-Token	字符串(32 位)	授权令牌
Api-4s-Version	字符串(最长 5 位)	接口版本

## 请求头示例

Api-4s-Version:1.0

Authorization-Token:XX

## 返回内容

参数名称	参数类型	参数说明
refreshToken	字符串	新的刷新令牌
effectiveDuration	整数	授权令牌有效时长，单位(分钟)

## 返回示例

成功示例

```
{
  "code": "10000",
  "remark": "成功",
  "data": {
    "refreshToken": "DEF",
    "effectiveDuration": "120"
  }
}
```

失败示例，如果 token 校验失败，返回的是

```
{
  "code": "10030",
  "remark": "未授权",
  "data": {
  }
}
```

失败示例，如果 refreshToken 校验失败，返回的是

```
{
  "code": "10010",
  "remark": "参数错误",
  "data": {
  }
}
```

```

    }
}

```

## 接口三：品牌列表

### 接口地址

[/4s/brand/list](#)

### 注意

生产环境限制每天调用 100 次，测试环境限制每天 500 次

### 协议

GET

### 传入参数

参数名称	参数类型	是否必填	参数说明
无			

### 请求头

参数名称	参数类型	参数说明
Authorization-Token	字符串(32 位)	授权令牌
Api-4s-Version	字符串(最长 5 位)	接口版本

### 请求头示例

Api-4s-Version:1.0

Authorization-Token:XX

## 返回内容

参数名称	参数类型	参数说明
brandId	int	品牌厂商 Id
brandName	字符串	品牌厂商名称
lastModifyTime	字符串	yyyy-MM-dd HH:mm:ss
remark	字符串	备注

## 返回示例

成功示例

```
{
  "code": "10000",
  "remark": "成功",
  "data": [
    {
      "brandId": "120",
      "brandName": "宝骏",
      "remark": "",
      "lastModifyTime": "2016-01-01 01:01:01"
    }
    .....
  ]
}
```

失败示例，如果 token 校验失败，返回的是

```
{
  "code": "10030",
  "remark": "未授权",
  "data":
    {
    }
}
```

## 接口四：品牌厂商列表

### 接口地址

[/4s/brandFactory/list](#)

### 注意

生产环境限制每天调用 100 次，测试环境限制每天 500 次  
为保证业务的可用性，请缓存品牌厂商列表

### 协议

GET

### 传入参数

参数名称	参数类型	是否必填	参数说明
lastModifyTime	字符串（ yyyy-MM-dd HH:mm:ss）	否	如果为空，返回全部品牌厂商 如果不为空，返回时点之后更新过的品牌厂商

### 请求头

参数名称	参数类型	参数说明
Authorization-Token	字符串(32 位)	授权令牌
Api-4s-Version	字符串(最长 5 位)	接口版本

### 请求头示例

Api-4s-Version:1.0

Authorization-Token:XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

## 返回内容

参数名称	参数类型	参数说明
brandFactoryId	int	品牌厂商品牌厂商 Id
brandFactoryName	字符串	品牌厂商品牌厂商名称
brandId	int	品牌厂商 Id
brandName	字符串	品牌厂商名称
price	字符串	价格
needEngineNum	true/false	是否需要发动机号
usable	true/false	是否可用
lastModifyTime	字符串	yyyy-MM-dd HH:mm:ss
remark	字符串	备注

## 返回示例

成功示例

```
{
  "code": "10000",
  "remark": "成功",
  "data": [
    {
      "brandFactoryId": "20",
      "brandFactoryName": "上汽通用五菱",
      "brandId": "120",
      "brandName": "宝骏",
      "price": "29",
      "needEngineNum": false,
      "usable": true,
      "remark": "",
      "lastModifyTime": "2016-01-01 01:01:01"
    }
    .....
  ]
}
```

失败示例，如果 token 校验失败，返回的是

```
{
  "code": "10030",
  "remark": "未授权",
  "data":
  {
```

```
}
}
```

## 接口五：获取单个品牌厂商

### 接口地址

[/4s/brandFactory/get](#)

### 注意

每天限制调用 20000 次

### 协议

GET

### 传入参数

参数名称	参数类型	是否必填	参数说明
brandFactoryId	int	是	品牌厂商 id, 由接口”品牌厂商列表”获取

### 请求头

参数名称	参数类型	参数说明
Authorization-Token	字符串(32 位)	授权令牌
Api-4s-Version	字符串(最长 5 位)	接口版本

### 请求头示例

```
Api-4s-Version:1.0
Authorization-Token:XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

## 返回内容

参数名称	参数类型	参数说明
brandFactoryId	int	品牌厂商品牌厂商 Id
brandFactoryName	字符串	品牌厂商品牌厂商名称
brandId	int	品牌厂商 Id
brandName	字符串	品牌厂商名称
price	字符串	价格
needEngineNum	true/false	是否需要发动机号
usable	true/false	是否可用
lastModifyTime	字符串	yyyy-MM-dd HH:mm:ss
remark	字符串	备注

## 返回示例

成功示例

```
{
  "code": "10000",
  "remark": "成功",
  "data": {
    "brandFactoryId": "20",
    "brandFactoryName": "上汽通用五菱",
    "brandId": "120",
    "brandName": "宝骏",
    "price": "29",
    "needEngineNum": false,
    "usable": true,
    "lastModifyTime": "2016-09-12 10:11:12",
    "remark": ""
  }
}
```

失败示例，如果 token 校验失败，返回的是

```
{
  "code": "10030",
  "remark": "未授权",
  "data": {
  }
}
```

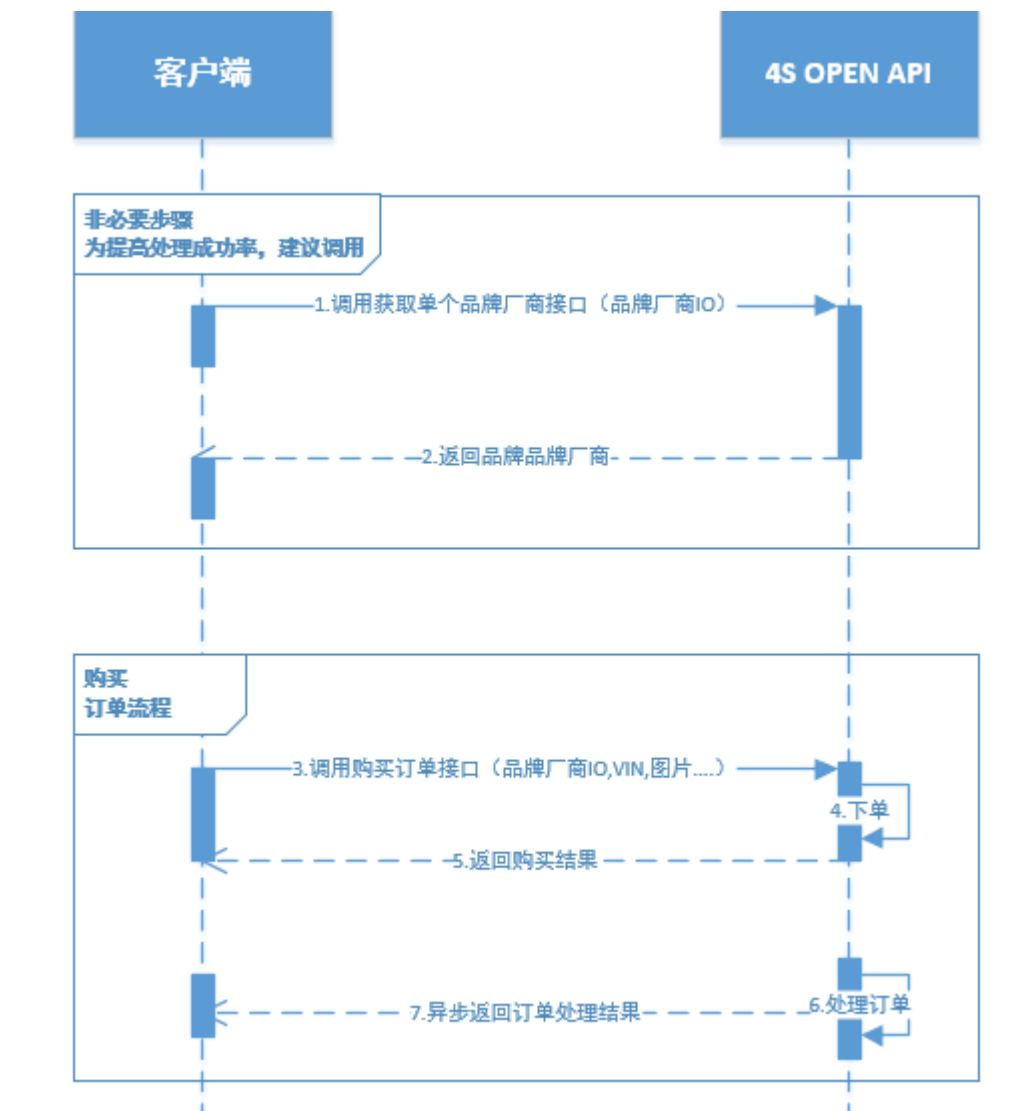


失败示例，如果品牌厂商不存在，返回的是

```
{
  "code": "20010",
  "remark": "品牌厂商不存在",
  "data":
    {
    }
}
```

## 接口六：购买订单

### 订单处理时序图



## 接口地址

/4s/report/buy

## 注意

每天限制调用 20000 次

## 协议

POST

## 传入参数

参数名称	参数类型	是否必填	参数说明
brandFactoryId	int	vin 必填，图片非必填	品牌厂商 id
vin	字符串（和 image 二选一）（17 位）	vin 和 image 二选一	车辆识别代码
engineNum	字符串（和 image 二选一）参照配置判断是否必填（最大 15 位）	如果选择的是 vin，且品牌厂商配置中的”是否需要发动机号”属性是 true，则必填	发动机号
image	字符串（和 vin 码二选一）（最大 100 位）	vin 和 image 二选一	行驶证照片 url
orderId	字符串（最大 50 位）	是	订单 id，合作商给订单生成的唯一标识
notifyUrl	字符串（最大 100 位）	是	回调地址，用于订单产生结果时回调

## 请求头

参数名称	参数类型	参数说明
Authorization-Token	字符串(32 位)	授权令牌
Api-4s-Version	字符串(最长 5 位)	接口版本

## 请求头示例

Api-4s-Version:1.0

Authorization-Token:XX

## 返回内容

参数名称	参数类型	参数说明
status	字符串	购买状态(success\failed)
remark	字符串	结果说明:
platformOrderNum	字符串	平台订单号

## 返回示例

成功示例

```
{
  "code": "10000",
  "remark": "成功",
  "data":
    {
      "status": "success",
      "remark": "购买成功",
      "platformOrderNum": "20160912-0001"
    }
}
```

失败示例：如果传入重复的订单号，返回

```
{
  "code": "30020",
  "remark": "订单号重复",
  "data":
    {
    }
}
```

更多失败场景参见业务状态码

## 接口七：查询订单结果

### 协议

GET

### 注意

每天限制调用 20000 次

### 接口地址

[/4s/report/get](#)

### 传入参数

参数名称	参数类型	是否必填	参数说明
orderId	字符串	是	订单 id,合作商给订单生成的唯一标识

### 请求头

参数名称	参数类型	参数说明
Authorization-Token	字符串(32 位)	授权令牌
Api-4s-Version	字符串(最长 5 位)	接口版本

### 请求头示例

Api-4s-Version:1.0

Authorization-Token:XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

## 返回内容

参数	参数名称	类型	参数说明	样例
status	订单状态	整型	1: 生成报告中 2: 查询成功 3: 查询失败 4: 查无记录	
orderId	订单号	字符串	合作商给订单生成的唯一标识	123
platformOrderId	平台订单号	字符串	平台声称的订单号	123
lastTimeToShop	最后到店时间	字符串	最后到店时间。 格式为 yyyy-MM-dd	2015-03-24
totalMileage	总公里数	小数	行驶的总公里数。单位为公里。	10000.00
numberOfAccidents	事故次数	整型	定义为异常记录的条数(现暂时不用， 统一是 null)，单位为条。	
brandFactoryId	车型品牌厂商ID	整型	请参见“接口一”	88
brandFactory	车型品牌厂商名称	字符串	请参见“接口一”	宝马
resultDescription	结果描述	字符串	对查询结果的一段文字描述	此车共 2 次事故，1: 后保更换 2: 尾灯，前部大灯更换，前保，后保油漆，详情见图片
resultImages	结果图片	字符串	4s 店拍摄的原始图片，查询成功时为不可空（数组）	'图片 URL_1','图片 URL_2'
createTime	交易创建时间	字符串	该笔交易创建的时间。格式为 yyyy-MM-dd HH:mm:ss。	2015-03-24 12:08:08
finishTime	交易完成时间	字符串	该笔交易完成的时间。格式为 yyyy-MM-dd HH:mm:ss。	2015-03-24 12:08:08
vin	车辆识别代码	字符串	车架号	
resultType	结果类型	整数	0: 图片 1: 文字 2: 图文	
resultDetails	结果详情	List	日期: date(String) 公里数: kilometers(Double) 维修类型: type(String) 项目: project(String) 材料: meterial(String) 是否异常: isAbnormal(Integer(1: 异常,	

			0 正常)); 图片: detailImages(List<String>)	
kilometerIsNormal	里程数是否正常	整型	1: 正常 0: 异常	
kilometerRemark	里程数备注	字符串	里程数异常说明	
airbagIsNormal	安全气囊是否正常	整型	1: 正常 0: 异常	
airbagRemark	安全气囊备注	字符串	安全气囊异常说明	
engineIsNormal	发动机是否正常	整型	1: 正常 0: 异常	
engineRemark	发动机备注	字符串	发动机异常说明	
structureIsNormal	结构部件是否正常	整型	1: 正常 0: 异常	
structureRemark	结构部件备注	字符串	结构部件异常说明	
gearBoxIsNormal	变速箱是否正常	整型	1: 正常 0: 异常	
gearBoxRemark	变速箱备注	字符串	变速箱异常说明	
electricSafetyEquipmentIsNormal	电气及安全设备是否正常	整型	1: 正常 0: 异常	
electricSafetyEquipmentRemark	电气及安全设备备注	字符串	电气及安全设备异常说明	
appearanceSheetMetalIsNormal	外观及钣金是否正常	整型	1: 正常 0: 异常	
appearanceSheetMetalRemark	外观及钣金备注	字符串	外观及钣金异常说明	
bigAccidentIsNormal	大事故记录是否正常	整型	1: 正常 0: 异常	
bigAccidentRemark	大事故记录备注	字符串	大事故记录异常说明	
burnIsNormal	火烧记录是否正常	整型	1: 正常 0: 异常	
burnRemark	火烧记录备注	字符串	火烧记录异常说明	
floodIsNormal	水淹记录是否正常	整型	1: 正常 0: 异常	
floodRemark	水淹记录备注	字符串	水淹记录异常说明	
resultDetails	结果详情	List	日期: date(String) 公里数: kilometers(Double) 类型: type(String) 项目: project(String) 材料: meterial(String) 异常标志: isAbnormal(Boolean) 详情图片表: detailImages(List<String>)	

			是否网络数据: isWebData(Boolean)	
carInfo	车辆信息	List	查询信息描述: message(String) 车架号: vin(String) 生产年份: productYear(String) 厂家名称: subBrand(String) 品牌: brand(String) 车系: series(String) 车款: model(String) 年款: years(String) 排量: enginesML(String) 发动机型号: enginesMode(String) 变速器类型: transmissionType(String) 挡位数: numbersOfShift(String) 排放标准: emission(String) 车型代码: carCode(String) 停产年份: productStop(String) 指导价格: price(String) 销售名称: salesName(String) 车辆类型: carType(String) 车辆级别: level(String) 车身形式: carBody(String) 车门数: doors(String) 座位数: seats(String) 发动机最大功率: enginesKW(String) 燃油类型: fuelType(String) 变速箱描述: transmissionDes(String) 驱动方式: driverMode(String) 匹配车型: matchModels(List) 匹配车型字段如下: 车型 ID: modelId(String) 车型名称: model(String) 车系 ID: seriesId(String) 车系名称: series(String) 车型图: picture(String) 指导价格: price(String)	
claimReportStatus	出险查询状态	整型	0: 查询成功 1: 查询无记录 2: 查询失败	
claimReportRemark	出险查询备注信息	字符串	出险查询的备注说明	
claimReport	出险记录	对象	claimCount 字符串 查询出险次数 claimMoney 字符串 理赔总金额 (单位: 本文出现的金额单位全部为“分”) repairMoney 字符串 维修总金额	

			repairCount 字符串 维修总件数 renewMoney 字符串 换件总金额 renewCount 字符串 换件总件数 carClaimRecords List 车辆出险记录列表 otherAmount 字符串 其他金额 dangerTime 字符串 出险时间； 格式: yyyy-MM-dd HH:mm:ss licenseNo 字符串 理赔车牌号 vehicleModel 字符串 理赔车型 frameNo 字符串 理赔车架号 damageMoney 字符串 理赔金额 repairAmount 字符串 维修金额 renewalAmount 字符串 换件金额 claimDetails List 车辆出险记录 明细列表 itemName 字符串 理赔项名称 itemType 字符串 理赔项类型 (维修, 换件, 人伤, 财产损失, 其他) itemAmount 字符串 理赔项金额	
--	--	--	---	--

## 返回示例

成功示例

```
{
  "code": "10000",
  "remark": "成功",
  "data":
    {
      "status": "30",
      "orderId": "...",
      ....
    }
}
```

失败示例，输入的订单号不存在时，返回

```
{
  "code": "30080",
  "remark": "订单号不存在",
  "data":
    {
    }
}
```





## 接口八：异步结果通知

### 协议

POST

### 接口地址

购买订单时填写的参数

### 传入参数

参数名称	参数类型	是否必填	参数说明
status	字符串	是	订单状态，同接口“查询订单结果”中的订单状态
orderId	字符串	是	合作商给订单生成的唯一标识
platformOrderId	字符串	是	平台给订单生成的唯一标识

### 返回值

收到通知，请返回 success

### 重试机制

生成报告时请求一次，如果返回值不为 success,每 5 分钟重试一次,最多重试 2 次

## 业务状态码：

### 通用

- 10000：成功
- 10010：参数错误
- 10020：账户错误
- 10030：未授权
- 10040：访问次数超限
- 10099：其它错误

### 品牌厂商错误状态

- 20010：品牌厂商不存在

### 订单错误状态

- 30010：账户余额不足
- 30020：订单号重复
- 30030：不支持此品牌厂商
- 30040：品牌厂商数据维护中
- 30050：图像模糊或者无法识别
- 30060：车架号错误
- 30070：缺少发动机号
- 30080：订单号不存在