

ETCP 开放平台接口文档

2017-07-24

ETCP 开放平台开发组

文档版本信息

版本号	更新内容	更新时间	操作人
1.0	创建文档	2017-07-24	ETCP 开放平台开发组

目录

1. 总体概述.....	5
2. 产品接入流程.....	5
3. 登录与用户身份识别.....	5
4. 接口协议.....	5
5. 签名规则.....	6
5.1. 商户号.....	6
5.2. 时间戳.....	6
5.3. 数据参数.....	6
5.4. 签名.....	7
6. 接口版本信息.....	7
7. 访问地址.....	8
8. 接口定义.....	8
8.1. 获取手机验证码.....	8
8.2. 手机号验证码登录.....	9
8.3. 联合登录.....	11
8.4. 车牌绑定.....	12
8.5. 车牌解除绑定.....	14
8.6. 优惠券查询.....	15

8.7.	查询用户停车记录.....	17
8.8.	停车费查询.....	20
8.9.	主动支付.....	22
8.10.	用户支付完成通知.....	24
8.11.	车场剩余车位查询.....	27
8.12.	通用车场列表.....	28
9.	消息推送.....	31
9.1.	支付宝签约通知信息推送.....	32
9.2.	入场信息推送.....	33
9.3.	出场信息推送.....	36
10.	错误码.....	38

1. 总体概述

ETCP 开放平台为企业及个人开发者提供各种停车服务集成方案，通过对接开放接口，可以实现用户从注册，绑定车牌，到费用查询，在线缴费，代扣停车费等功能的全流程停车服务。实时准确的实现用户停车数据同步，与车场管理功能。

2. 产品接入流程

- 1、实现与开放平台的对接首先需要获取商户号（merchant_no）与商户密钥（merchant_key），向 ETCP 提交相应信息审核通过后方可获取。
- 2、在使用功能接口时，需要通过商户密钥实现对请求数据的安全签名，通过接口安全验证后，可以成功调取 ETCP 开放平台停车服务。

3. 登录与用户身份识别

ETCP 停车开放平台用过登录操作识别用户行为，因此使用功能前，需要通过向登录接口传递用户唯一标识进行登录，登录成功后获取身份令牌（token）。

使用其他功能接口前，需统一传递令牌（token）参数，进行用户身份识别。

4. 接口协议

接口采用 HTTP 协议，支持 GET 或 POST 请求方式（参见[接口定义](#)），POST 编码格式为 application/x-www-form-urlencoded，POST 请求示例如下：

POST 接口地址 HTTP/1.1

Content-Type:application/x-www-form-urlencoded;charset=utf-8

```
merchant_no=9827c6b5c6d94485b912b6e90c92a7ae&time_stamp=20170328  
172011&sign=A3DE1813F4DDBADFC18555E18A2E579&data=%7B%22token%  
22%3A%22d970cff4-3811-438b-ba90-50503510345b%22%7D
```

5. 签名规则

ETCP 开放平台为了保证安全性，通签名方式对所有请求进行合法性校验。

商户对所有接口的访问均需要如下四个参数：

5.1. 商户号

ETCP 开放平台会给接入的商户颁发商户号(merchant_no)和与之对应的商户密钥(merchant_key)。商户号用于标示商户，每次请求均需带上该参数，标明请求来源。商户密钥用于计算请求签名，开放平台会依据签名校验商户合法性。

5.2. 时间戳

时间戳参数 (time_stamp) 代表请求发出时的系统时间，格式为 "yyyyMMddHHmmss"，例如：2017 年 7 月 21 日 13 点 24 分 52 秒对应为 20170721132452，请求发出一定时间之后，请求将被视为过期，并标记为无效。

5.3. 数据参数

数据参数 (data) 为 json 格式，传递具体接口的数据参数，结构如 {"param1":"value1","param2":"value2","param3":"value3"}

5.4. 签名

签名值 (sign) 根据统一签名规则计算得出，用于验证调用方身份，确保保证请求来源合法且信息未被篡改。

签名规则：

签名计算通过对待签名数据进行 HASH 计算得出。

HASH 算法采用 MD5 算法，计算结果用 16 进制大写表示：

规则可以表示为：

$sign = \text{uppercase} (\text{Hex} (\text{MD5} (\text{待签名数据})))$

待签名数据 = data + merchant_key + time_stamp (顺序固定)

例如，调用某请求时，信息如下：

参数	{"param1":"value1","param2":"value2","param3":"value3"}
商户密钥	noxxxxx
时间戳	20170721132452

则生成的对应的待签名数据为：

{"param1":"value1","param2":"value2","param3":"value3"}noxxxxx20170721132452

签名生成的 sign 为：A3DE1813F4DDBADFC18555E18A2E579D

6. 接口版本信息

接口中 version 参数统一为：1.0.0

7. 访问地址

测试域名：http://mapi.test.etcp.cn

生产域名：https://mapi.etcp.cn

8. 接口定义

8.1. 获取手机验证码

接口调用方通过调用此接口，获取手机短信验证码。注：1 分钟之内不允许重复获取验证码。

8.1.1. 接口定义

URL	/merchant/open/{version}/openapi/uservalidatecode			
请求方式	GET			
请求参数	参数名称	是否必须	类型	描述
	mobilePhone	必须	String	用户手机号
返回结果	名称	必选	类型及范围	说明
	code	必选	Int	0:成功 1:失败
	message	必选	String	失败原因描述

8.1.2. 请求参数示例

```
"data":{
```

```

"mobilePhone": "13800000001"
}
    
```

8.1.3. 应结果示例

```

{
  "code": 0,
  "message": "ok"
}
    
```

8.2. 手机号验证码登录

接口调用方传入用户手机号、手机验证码及应用标识 (APPID) 调用此接口完成用户的登录，获取登录令牌 (token)。

8.2.1. 接口定义

URL	/merchant/open/{version}/openapi/userlogin			
请求方式	POST			
请求参数	参数名称	是否必须	类型	描述
	mobilePhone	必须	String	用户手机号
	code	必须	String	手机验证码
	appld	必须	String	客户端设备 ID
返回结果	名称	必选	类型及范	说明

			围	
	code	必选	Int	0:成功 1:失败
	message	必选	String	失败原因描述
	data.token	非必选	json	token 用户登录令牌

8.2.2. 请求参数示例

```

"data":{
  "mobilePhone": "13800000001",
  "code": "562371",
  "appld": "ETCP"
},
    
```

8.2.3. 响应结果示例

```

{
  "code": 0,
  "data": {
    "token": "0b59582d-ea6f-4bf1-a570-e803cc24ebe0"
  },
  "message": "ok"
}
    
```

8.3. 联合登录

接口调用方传入用户输入的手机号、车牌号及应用标识 (APPID) 为用户分配登录 token。token 的默认有效期为 1 天，如果用户在一天之内多次调用此接口，则 token 以最后一次生成的为准。

8.3.1. 接口定义

URL	/merchant/open/{version}/openapi/usersignin			
请求方式	POST			
请求参数	参数名称	是否必须	类型	描述
	mobilePhone	必须	String	用户手机号
	carNumber	非必须	String	车牌号
	appld	必须	String	客户端设备 ID
返回结果	名称	必选	类型及范围	说明
	code	必选	Int	0:成功 1:失败
	message	必选	String	失败原因描述
	data.token	非必选	json	token 用户登录令牌

8.3.2. 请求参数示例

```
"data":{
    "appld": "ETCP",
```

```
"carNumber": "渝 AJS537",  
"mobilePhone": "13508381199"  
}
```

8.3.3. 响应结果示例

```
{  
  "code": 0,  
  "data": {  
    "token": "0b59582d-ea6f-4bf1-a570-e803cc24ebe0"  
  },  
  "message": "ok"  
}
```

8.4. 车牌绑定

接口调用方传入登录时获取的 token 及要绑定的车牌号，通过调用此接口完成车牌绑定。注：新能源车牌号有效位数 8 位，非新能源车牌号为 7 位；一个手机号最多只能绑定 3 辆车。

8.4.1. 接口定义

URL	/merchant/open/{version}/openapi/bindcar
请求方式	POST

请求参数	参数名称	是否必须	类型	描述
	token	必须	String	用户 token
	plateNumber	必须	String	车牌号码
	plateColor	非必须	String	车牌颜色(1 蓝 2 黑 3 黄 4 白)
返回结果	名称	必选	类型及范围	说明
	code	必选	Int	0:成功 其他代码:失败
	message	必选	String	失败原因描述
	data	非必选	json	返回数据 (如果存在)

8.4.2. 请求参数示例

```
"data": {  
  "plateColor": "2",  
  "plateNumber": "渝 B757D0",  
  "token": "ee8b9712-fc20-4b9d-96eb-cd1fdacd3c1a"  
}
```

8.4.3. 响应结果示例

```
{  
  "code": 0,  
  "message": "ok"
```

```
}

```

8.5. 车牌解除绑定

接口调用方传入登录时获取的 token 及要解绑的车牌号，通过调用此接口完成车牌解绑。注：新能源车牌号有效位数 8 位，非新能源车牌号为 7 位；一个车牌号一天内最多只能被解绑 4 次。

8.5.1. 接口定义

URL	/merchant/open/{version}/openapi/unbindcar			
请求方式	POST			
请求参数	参数名称	是否必须	类型	描述
	token	必须	String	用户 token
	plateNumber	必须	String	车牌号码
返回结果	名称	必选	类型及范围	说明
	code	必选	Int	0:成功 其他代码:失败
	message	必选	String	失败原因描述

8.5.2. 请求参数示例

```
"data":{
  "token": "36a575cf-b3ea-4591-a1f4-9977dd755e0b",

```

```

    "plateNumber": "京 F43264"
}
    
```

8.5.3. 响应结果示例

```

{
    "code": 0,
    "message": "ok"
}
    
```

8.6. 优惠券查询

接口调用方传入登录时获取的 token 及分页信息,通过调用此接口完成用户优惠券信息查询。

8.6.1. 接口定义

URL	/merchant/open/{version}/openapi/couponlist			
请求方式	GET			
请求参数	参数名称	是否必须	类型	描述
	token	必须	String	用户 token
	lastCode	非必须	String	上一分页代码
	pageSize	必须	Integer	1-200 之间的分页每页数量
返回结果	名称	必选	类型及范	说明

			围	
	code	必选	Int	0:成功 1:失败
	message	必选	String	失败原因描述
	data	非必选	jsonArray	absolute_expire_time: 优惠券过期时间, bindTime: 优惠券发放时间, code: 优惠券代码, coupon_value: 优惠券金额 (单位元), name: 优惠券名称,

8.6.2. 请求参数示例

```

"data":{
  "token": "36a575cf-b3ea-4591-a1f4-9977dd755e0b",
  "lastCode": "",
  "pageSize": "10"
}
    
```

8.6.3. 响应结果示例

```

{
  "code": 0,
}
    
```

```
"message": "ok",
"data": [
  {
    "absolute_expire_time": "2017-06-11 16:27:30",
    "bindTime": "2017-06-10 16:27:30",
    "code": "1136",
    "coupon_value": "1",
    "name": "现金优惠券"
  },
  {
    "absolute_expire_time": "2017-06-12 16:27:30",
    "bindTime": "2017-06-10 16:27:30",
    "code": "1126",
    "coupon_value": "2",
    "name": "现金优惠券"
  }
]
}
```

8.7. 查询用户停车记录

接口调用方传入登录时获取的 token ,通过调用此接口完成用户历史订单信息查询。注：

订单信息按入场时间倒序排列且只能返回近 7 天的订单信息。

8.7.1. 接口定义

URL	/merchant/open/{version}/openapi/orderhistory			
请求方式	GET			
请求参数	参数名称	是否必须	类型	描述
	token	必须	String	用户 token
返回结果	名称	必选	类型及范围	说明
	code	必选	Int	0:成功 1:失败
	message	必选	String	失败原因描述
	data	非必选	jsonArray	parkingName:车场名称 entranceTime:入场时间 exitTime:出场时间 parkingFee:停车费 actual:现金金额 (现金出场) parkingId : 车场 ID orderId : 停车记录订单号 carNumber : 车牌号

8.7.2. 请求参数示例

```
"data":{  
  "token": "36a575cf-b3ea-4591-a1f4-9977dd755e0b"  
}
```

8.7.3. 响应结果示例

```
{  
  "code": 0,  
  "message": "ok",  
  "data": [  
    {  
      "parkingName": "测试停车场",  
      "entranceTime": "2017-06-10 16: 27: 30",  
      "exitTime": "2017-06-10 19: 27: 30",  
      "parkingFee": "5.5",  
      "actual": "0.0",  
      "parkingId": "10361002",  
      "orderId": "B6269DE8-03A0-4560-8EAF-000000C5FF6A",  
      "carNumber": "京 MZ1173"  
    }  
  ]  
}
```

```
}

```

8.8. 停车费查询

接口调用方传入登录时获取的 token 及车牌号，通过调用此接口完成停车费查询。注：

该接口在调用时必须保证车辆已经入场，否则查询时返回的数据将会为空。

8.8.1. 接口定义

URL	/merchant/open/{version}/openapi/orderunpay			
请求方式	GET			
请求参数	参数名称	是否必须	类型	描述
	token	必须	String	用户 token
	plateNumber	必须	String	车牌号
返回结果	名称	必选	类型及范围	说明
	code	必选	Int	0:成功 1:失败
	message	必选	String	失败原因描述
	data	非必选	jsonArray	entranceTime:入场时间 exitTime:出场时间 parkingFee:总停车费 remainingFee: 剩余代缴 停车费

				parkingId : 车场 ID orderId : 停车记录订单号 carNumber : 车牌号
--	--	--	--	---

8.8.2. 请求参数示例

```

"data": {
  "token": "36a575cf-b3ea-4591-a1f4-9977dd755e0b",
  "plateNumber": "渝 DE3787"
}
    
```

8.8.3. 响应结果示例

```

{
  "code": 0,
  "data": [
    {
      "carNumber": "渝****87",
      "entranceTime": "2017-06-11 11:19:27",
      "exitTime": "2017-06-11 11:59:47",
      "orderId": "B9466E26-7420-4BDF-958D-E90106AB5F20",
      "parkingFee": 2,
      "parkingId": 3126,
      "parkingName": "ETCP 停车场",
    }
  ]
}
    
```

```

    "remainingFee": 2
  }
],
  "message": "ok"
}

```

8.9. 主动支付

接口调用方传入登录时获取的 token 及其他必传信息，通过调用此接口完成停车费支付。

8.9.1. 接口定义

URL	/merchant/open/{version}/openapi/orderpay			
请求方式	POST			
请求参数	参数名称	是否必须	类型	描述
	token	必须	String	用户 token
	source	必须	Integer	订单来源 1 微信 2 阿里 web
	synId	必须	String	停车记录标示
	openId	必须	String	支付方 openId
	returnUrl	非必须	String	阿里 web 支付回调 url
	useCoupon	必须	boolean	使用有免券

	couponCode	非必须	String	优惠券代码
	couponFree	非必须	double	优惠券金额
返回结果	名称	必选	类型及范围	说明
	code	必选	Int	0:成功 1:失败
	message	必选	String	失败原因描述
	data	非必选	json	返回数据 (如果存在)

8.9.2. 请求参数示例

```

"data": {
    "token": "cc9ca4ac-2fb9-435b-bfcb-a77ccc59bccb",
    "synId": "023212A5-E68C-4AFA-9164-2907256188FF",
    "openId": "oPwpNuGvl8Jlco50VbAGtP1wJ73o",
    "payType": 1,
    "returnUrl": "http://www.dsf.com/clickwifi/paySuccess",
    "useCoupon": false,
    "couponCode": "",
    "couponFree": null
}
    
```

8.9.3. 响应结果示例

```
{
```

```

"code": 0,
"message": "ok",
"data": {
    "paySign": "98ECBE95A7BAAC774E26D53AD466E45C",
    "appid": "wxd4a86d0d56b87fb4",
    "signType": "MD5",
    "packages":
"prepay_id=wx201706151746334caf88f2c10382919637",
    "nonceStr": "1746335779",
    "timestamp": "1497519993"
}
}
    
```

8.10. 用户支付完成通知

用户在接口调用方平台上完成支付后,接口调用方传入登录时获取的 token 及其他必传信息,通过调用此接口将用户支付信息同步给 ETCP 开放平台。注:接口调用方生成的外部订单号(externalOrderId)必须保证全局唯一。

8.10.1. 接口定义

URL	/merchant/open/{version}/openapi/orderpaid			
请求方式	POST			
请求参数	参数名称	是否必须	类型	描述

	token	必须	String	用户 token
	orderId	必须	String	停车记录订单号
	externalOrderId	必须	String	外部停车支付单 ID
	payment	必须	decimal	支付金额 (例如:10.34)
	payTime	必须	datetime	支付时间 (例如 : 2015-09-10 00:00:00)
返回结果	名称	必选	类型及范围	说明
	code	必选	Int	0:成功 1:失败
	message	必选	String	失败原因描述
	data	非必选	json	{ orderId : 停车记录订单号 externalOrderId : 外部支付订单号 payment : 支付金额(元) payTime : 支付时间 payOrderId:支付订单号 }

8.10.2. 求参数示例

```
"data":{
```

```
"externalOrderId": "20170612095857258937",  
"orderId": "58231364-19F8-4EF1-8FC1-5FC9F4C635EA",  
"payTime": "2017-06-12 09:58:58",  
"payment": "8.0",  
"token": "09f680e9-2b73-42b4-aa36-4618a8d508f0"  
}
```

8.10.3. 响应结果示例

```
{  
  "code": 0,  
  "data": {  
    "externalOrderId": "20170612095857258937",  
    "orderId": "58231364-19F8-4EF1-8FC1-5FC9F4C635EA",  
    "payOrderId": "no201707120958582728978",  
    "payTime": "2017-06-12 09:58:58",  
    "payment": 8  
  },  
  "message": "支付通知成功"  
}
```

8.11. 车场剩余车位查询

接口调用方传入车场 ID，通过调用此接口可以获取车场的车位信息。注：车场 ID 可以通过查费接口获取。

8.11.1. 接口定义

URL	/merchant/open/{version}/common/parkpoiavailable			
请求方式	POST			
请求参数	参数名称	是否必须	类型	描述
	parkId	必须	jsonArray	车场 ID jsonarray 字符串
返回结果	名称	必选	类型及范围	说明
	code	必选	Int	0:成功 1:失败
	message	必选	String	成功或者失败描述
	data	非必选	json	返回数据(如果存在) availableLotsTotal 剩余车位 publicLotsTotal 总车位 parkId 车场 ID

8.11.2. 请求参数示例

```
"data": {
```

```
"parkId": [  
  "1967"  
]  
}
```

8.11.3. 响应结果示例

```
{  
  "code": 0,  
  "data": [  
    {  
      "availableLotsTotal": 2901,  
      "parkId": "1967",  
      "publicLotsTotal": 3004  
    }  
  ],  
  "message": "查询成功"  
}
```

8.12. 通用停车场列表

接口调用方传入分页信息及电子支付标记，通过调用此接口可以获取停车场列表信息。电子支付标记字段传入 1 时，获取的停车场为支持电子支付的停车场列表信息，反之获取到的为不支持电子支付的停车场列表信息。注：停车场 ID 可以通过查费接口获取；停车场 ID 最后

一位为车场属性 1：闸机车场；2：路边车场，车场真实 ID 不包含最后一位。

8.12.1. 接口定义

URL	/merchant/open/{version}/common/parkpoisearch			
请求方式	POST			
请求参数	参数名称	是否必须	类型	描述
	startIndex	非必须	String	起始记录数 默认 0
	pageSize	非必须	String	每页数据量 默认 10,最大为 200
	isOnlinePay	非必须	String	电子支付 0 否 1 是
返回结果	名称	必选	类型	说明
	code	必选	Int	0:成功 1:失败
	message	必选	String	成功或者失败描述
	data	非必选	json	返回数据(如果存在)

8.12.2. 请求参数示例

```
"data":{
```

```
  "pageSize": 1,
```

```
  "startIndex": 1
```

```
}
```

8.12.3. 响应结果示例

```
{
```

```
"code": 0,
"data": {
  "count": 1,
  "parks": [
    {
      "aliasName": "观湖公寓", //停车场简称
      "areaRoad": [], //区域道口 暂无
      "availableLotsTotal": 0, //空余车位数
      "businessHours": "00:00--24:00", //营业时间
      "carPositionStatus": 0,
      "distance": 0, //与目标点距离
      "etcpParkId": "1139", //ETCP 停车场 ID
      "feeDescr": "",
      "feeHour": "5",
      "feeType": 2,
      "isOnlinePay": 1,
      "mapRateDesc": "5 元/小时", //地图费率
      "maxDayMoney": 0, //最高封顶值
      "minFreeTimes": 60, //最低首免时长
      "operateState": "2", //停车场状态
      "parkAddr": "长沙市岳麓区咸嘉湖路 690 号", //停车场地址
      "parkCity": "长沙市", //停车场所在城市
    }
  ]
}
```

```
"parkDistrict": "岳麓区", //车场所所在区
"parkId": "11391", //车场 ID
"parkLat": "28.217296", //车场纬度
"parkLon": "112.943688", //车场经度
"parkName": "观湖公寓",
"parkType": "146", //车场类型
"publicLotsTotal": 265, //总车位数
"publishState": 1, //发布状态
"rateDesc": "白天 7:00-21:00#免费时间|30 分钟%封顶|35
元%小型车|5 元/小时%大型车|5 元/小时%&夜间 21:00-7:00#小型车|5 元/小
时%大型车|5 元/小时" //车场费率描述
    }
  ]
},
"message": "查询成功"
}
```

9. 消息推送

消息推送需要商户按照 ETCP 开放平台标准提供消息接收接口，由 ETCP 开放平台开通相关推送功能方可接收消息。

9.1. 支付宝签约通知信息推送

商户调用签约代扣接口绑定签约代扣成功之后，会收到相应信息推送。

9.1.1. 接口定义

URL	商户提供，用于数据对接			
请求方式	POST			
请求参数	参数名称	是否必须	类型	描述
	mobilePhone	必须	String	手机号
	agreementNo	必须	String	签约号
	signTime	必须	String	签约时间
	signStatus	必须	String	签约状态 TEMP：暂存，协议未生效过； NORMAL：正常； STOP：暂停。
	signValidTime	必须	String	签约生效时间
	signInvalidTime	非必须	String	签约失效时间
返回结果	名称	必选	类型及范围	说明
	code	必选	Int	0:成功 非 0:失败

	message	必选	String	成功或者失败描述
--	---------	----	--------	----------

9.1.2. 请求参数示例

```
{  
  "mobilePhone": "15901167943",  
  "signTime": "2017-06-23 17:35:07",  
  "agreementNo": "20170623965754761881",  
  "signStatus": "NORMAL",  
  "signInvalidTime": "2115-02-01 00:00:00",  
  "signValidTime": "2017-06-23 17:35:07"  
}
```

9.1.3. 响应结果示例

```
{  
  "code": 0,  
  "message": "ok"  
}
```

9.2. 入场信息推送

车辆入场时，商户会收到相应信息推送。

9.2.1. 接口定义

URL	商户提供，用于数据对接			
请求方式	POST			
请求参数	参数名称	是否必须	类型	描述
	parkId	必须	String	车场 ID
	plateNumber	必须	String	车牌号码
	parkName	必须	String	车场名称
	synId	必须	String	停车记录 ID
	userType	必须	String	用户类型 76:长租用户； 77：临停用户
	fixParkingId	非必须	String	长租车位 ID
	remainingDays	非必须	String	长租剩余天数(如果是临停用户此字段为空)
	entranceTime	必须	String	入场时间 (yyyy-MM-dd HH:mm:ss)
	pushTime	必须	String	推送时间 (yyyy-MM-dd HH:mm:ss)
返回结果	名称	必选	类型及范围	说明

	code	必选	Int	0:成功 非 0:失败
	message	必选	String	成功或者失败描述

9.2.2. 请求参数示例

```
{  
  "entranceTime": "2017-06-12 03:17:15",  
  "fixParkingId": "J402",  
  "parkName": "ETCP 智慧停车场",  
  "remainingDays": "18",  
  "userType": "76",  
  "plateNumber": "渝 A073G8",  
  "pushTime": "2017-06-12 03:17:16",  
  "synId": "30bb750e-e055-4d5b-9c54-727c844c393f",  
  "parkId": "1"  
}
```

9.2.3. 响应结果示例

```
{  
  "code": 0,  
  "message": "ok"  
}
```

9.3. 出场信息推送

车辆出场时，商户会收到相应信息推送。

9.3.1. 接口定义

URL	商户提供，用于数据对接			
请求方式	POST			
请求参数	参数名称	是否必须	类型	描述
	parkId	必须	String	车场 ID
	plateNumber	必须	String	车牌号码
	parkName	必须	String	车场名称
	synId	必须	String	停车记录 ID
	userType	必须	String	用户类型 76:长租用户； 77：临停用户
	fixParkingId	非必须	String	长租车位 ID
	remainingDays	非必须	String	长租剩余天数(如果是临停 用户此字段为空)
	entranceTime	必须	String	入场时间 (yyyy-MM-dd HH:mm:ss)
	exitTime	必须	String	出场时间

				(yyyy-MM-dd HH:mm:ss)
	stayedTime	必须	String	停留时间(以秒为单位计算)
	receivableFee	必须	String	应收金额(以‘分’为单位)
	pushTime	必须	String	推送时间 (yyyy-MM-dd HH:mm:ss)
返回结果	名称	必选	类型及范围	说明
	code	必选	Int	0:成功 非 0:失败
	message	必选	String	成功或者失败描述

9.3.2. 请求参数示例

```

{
    "entranceTime": "2017-06-12 03:19:13",
    "fixParkingId": "",
    "exitTime": "2017-06-12 03:22:46",
    "stayedTime": 213,
    "parkName": "ETCP 智慧停车场",
}
    
```

```
"receivableFee": "0.00",  
"remainingDays": "",  
"userType": "77",  
"plateNumber": "渝 A8T769",  
"pushTime": "2017-06-12 03:22:48",  
"synId": "35959b19-f983-4d65-a1e5-041874c0074e",  
"parkId": "1"  
}
```

9.3.3. 响应结果示例

```
{  
  "code": 0,  
  "message": "ok"  
}
```

10. 错误码

code	message
------	---------

0	成功
1	失败
2	服务器异常
3	时间戳超时
4	签名校验非法
5	没有找到商户密钥
6	参数错误
7	更新订单失败
8	生产订单失败
20100	操作失败
20101	无效 token
20102	缺少必要参数
20201	手机号错误
20202	获取验证码失败
20203	手机验证码错误