

# POS 销售交易接口规范

版本：V3.2



安胜旗信息咨询（深圳）有限公司  
Asiatic Consulting Co., Ltd

2017 年 08 月

## 版 权 说 明

本文件中的信息和文字说明的知识产权归安胜旗信息咨询（深圳）有限公司所有。除非经过书面允许，本文件持有者必须：1) 将本文件视为机密文件，2) 保护本文件中的信息不向外界或第三方泄露。

## 使 用 说 明

本建议书仅限于在平安集团内部使用，除了对本文件内容进行评审外，不能以任何理由复制、使用或泄露建议书的全部或部分内容。本限制条款不限制安胜旗使用建议书中从其他非限制渠道获得的信息。

## 文档编号

*POS\_Interface\_001*

## 修订记录

日期	版本	修改者	描述
2017 年 08 月 18 日	V3.1	Henry	发布
2017 年 11 月 13 日	V3.2	Henry	新增应收金额

## 目 录

1. POS 交易数据接口规范 .....	4
1.1. 数据生成方式.....	4
1.2. 接入方式.....	4
1.2.1. WebServices 接入方式 .....	4
1.3. 格式规范.....	4
1.3.1. WebServices 参数 .....	4

## 1. POS 交易数据接口规范

### 1.1. 数据生成方式

- ✓ 第三方收银系统通过调用 POS 系统的 WebServices 将交易数据接入；

### 1.2. 接入方式

#### 1.2.1. WebServices 接入方式

第三方收银系统调用 MallPOS 发布到互联网的 WebServices，将每笔交易数据接入。

互联网的 WebServices 地址为：

<http://ascentis.eicp.net:2003/FTPDataServer/FTPService.asmx>

局域网的 WebServices 地址为：

接口方法：TrasAddForPay(jsonstring)

传入 jsonstring 字段要求见 1.3.1 参数部分描述；

### 1.3. 格式规范

#### 1.3.1. WebServices 参数

传入 Json 字符串参数，各自的解析如下，：

Auth:

字段	类型	举例	说明
Signature	字符	5F5024AE56CF405 AA9D8A3F9A39B0 16D	签名：客户端用 Token+ Timestamp+ Nonce 后的字符串进行字典排序； 客户端将排序后的字符串进行 MD5 加密； Token:由商管公司提供
Timestamp	长整型	1503479467649	时间戳
Nonce	字符	17	2 位随机数

Trans:

字段	说明	类型	举例	说明
StoreCode	商铺编号	字符 (10)	B1-2	后台系统中商铺管理中查询 SID
POSCode	POS 编号	字符 (10)	1089	后台系统 POS 终端管理中查询
TransID	小票号	字符(30)	5001	
Price	实收金额	货币	-8900.56	退货为负值、销售为正值
BillPrice	应收金额	货币	-8900.56	退货为负值、销售为正值
TransDate	交易日期	日期型	2006-01-05 12:30:55	格式: YYYY-MM-DD hh:mm:ss
TransDetail	交易明细	数组	包含多条	见下面详细说明
TransPay	支付明细	数组	包含多条	见下面详细说明

TransDetail:

字段	说明	类型	举例	说明
ProductCode	商品编码	字符 (30)	B1-2	
ProductName	商品名称	字符 (30)	1089	
Price	销售总价	字符(20)	88.00	单价*数量
Num	数量	字符(20)	1	退货为负数

TransPay:

字段	说明	类型	举例	说明
PayCode	支付类型	字符 (30)	101	统一约定如下 (按项目修改) PAFC : 1011 现金 3011 支票转账 4011 支付宝 4012 微信 4013 储蓄卡 4014 信用卡 6012 代金券 8011 壹钱包 9011 支付调整 9012 其它
PayAmt	支付金额	字符(20)	88.00	保留 2 位小数, 退货为

				负数
--	--	--	--	----

传入 json 参数例如:

```
{
  "Auth": {
    "Signature": "5F5024AE56CF405AA9D8A3F9A39B016D",
    "Timestamp": "1503479467649",
    "Nonce": "17"
  },
  "Trans": [
    {
      "StoreCode": "527",
      "PosCode": "1207",
      "TransID": "0100001",
      "Price": "6.00",
      "BillPrice": "6.00",
      "TransDate": "2016-09-01 10:16:54",
      "TransDetail": [
        {
          "ProductCode": "P01",
          "ProductName": "商品",
          "Price": "6.00",
          "num": "1"
        }
      ],
      "TransPay": [
        {
          "PayCode": "101",
```

```
        "PayAmt": "6.00"
      }
    ]
  },
  {
    "StoreCode": "527",
    "PosCode": "1207",
    "TransID": "0100002",
    "Price": "8.00",
    "BillPrice": "8.00",
    "TransDate": "2016-09-01 10:17:50",
    "TransDetail": [
      {
        "ProductCode": "P01",
        "ProductName": "商品",
        "Price": "8.00",
        "Num": "1"
      }
    ],
    "TransPay": [
      {
        "PayCode": "101",
        "PayAmt": "8.00"
      }
    ]
  }
]
```

返回 json 参数例如:



```
{  
  
    "HasError": false, //是否包含错误, true/false  
  
    "ErrorMsg": null //错误内容, 当HasError=true是才有值  
  
}
```

签名算法 C#示例:

```
/// <summary>  
/// 获取加密签名  
/// </summary>  
/// <param name="timestamp"></param>  
/// <param name="nonce"></param>  
/// <returns></returns>  
public static string GetSignature(long timestamp, string nonce)  
{  
    string token = "AS"; //由安胜旗提供  
    string str = string.Format("{0}{1}{2}", token, timestamp.ToString(),  
nonce);  
  
    List<char> str2 = str.ToList<char>();  
  
    str2.Sort();  
  
    string str3 = "";  
  
    foreach (var item in str2)  
    {  
        str3 = string.Format("{0}{1}", str3, item.ToString());  
    }  
    return MD5Encrypt(str3);  
}  
  
/// <summary>  
/// MD5加密  
/// </summary>  
/// <param name="strText"></param>  
/// <returns></returns>  
public static string MD5Encrypt(string strText)  
{  
    string cryptStr = "";  
    MD5CryptoServiceProvider md5 = new MD5CryptoServiceProvider();
```

```
byte[] bytes = Encoding.UTF8.GetBytes(strText);

byte[] cryptBytes = md5.ComputeHash(bytes);

for (int i = 0; i < cryptBytes.Length; i++)
{
    cryptStr += cryptBytes[i].ToString("X2");
}

return cryptStr;
}
```