



万达征信系统

接口文档（外部）

版本：V1.0.0

请注意保密

2016 年 5 月

目录

1	接口列表.....	2
2	概要说明.....	2
2.1	接口：S1	2
2.2	接口：S2	2
3	请求方式.....	2
3.1	Token 接口.....	2
3.2	产品服务接口.....	2
4	接入步骤.....	3
4.1	获取授权信息.....	3
4.2	构建业务参数.....	3
4.3	构建授权参数.....	3
4.4	加密数据.....	3
4.5	发送请求.....	3
4.6	解密数据.....	3
5	代码示例.....	4
6	附录.....	5
6.1	产品编目.....	5
6.2	传入参数详解.....	5
6.3	返回数据详解.....	6
6.4	返回码(业务).....	9
6.5	返回码(系统).....	10

1 接口列表

接口编号	接口名称	接口说明
S1	Token 令牌获取接口	获取授权令牌
S2	征信服务请求接口	请求征信服务

2 概要说明

2.1 接口：S1

1. 令牌三要素：授权时效，访问次数，再生次数。
2. 令牌分为临时，短期和永久类型，分别对应不同的授权时间，再生次数和服务请求次数。
3. 对于短期令牌，授权时间和授权服务请求次数任一项满足条件即会失效。
4. 如果令牌授权再生次数未到，还可继续申请令牌。
5. 如果令牌授权再生次数已到，将不可以再次申请令牌，请及时联系快钱征信公司。

2.2 接口：S2

1. 请求此接口必须先获取到有效的令牌。
2. 根据产品 ID 不一样，返回结构会不同，具体可参考描述文档。

3 请求方式

3.1 Token 接口

请按要求提交 get 请求

3.2 产品服务接口

请按要求提交 post 请求

4 接入步骤

4.1 获取授权信息

- 需获取授权账号
- 需获取授权api-key
- 需获取授权证书

4.2 构建业务参数

- 先构建treemap对象
- 填充prod_id(授权产品id)
- 根据产品不一样，需填充其他参数，具体参考描述文档

4.3 构建授权参数

- 先构建map对象
- 填充acct_id
- 填充api_id
- 填充token_id (调用token获取接口)
- 填充signature (调用sign方法)
- 填充request_sn (唯一标识)

4.4 加密数据

- 加密业务参数 (调用encodeData方法)
- 填充requestBody (4.2加密后的业务参数)

4.5 发送请求

- 提交 get/post 请求

4.6 解密数据

- 交易成功： 部分加密(需调用decodeData方法对retdata节点进行解密)
- 交易失败： 全明文传输

(*) 如有疑问，可参照代码示例。

5 代码示例

```
Map<String, String> params = new TreeMap<String, String>();
String request_sn = new Date().getTime()+"";
// Step1. 构建业务参数
Map<String, Object> datas = new TreeMap<String, Object>();
{ //必不可少
    datas.put("prod_id", "P_C_B002");
}
{ //产品参数
    datas.put("name", "李圣杰");
    datas.put("cardNo", "310321198311164112");
    datas.put("mobileNo", "13601731122");
    datas.put("99billAcount", "1111111111");
}
// Step2. 加密业务参数
params.put("requestBody", encodeData(datas));

// Step3. 构建授权参数
params.put("acct_id", acct_id);
params.put("api_id", api_id);
params.put("token_id", access_token);
params.put("signature", sign(request_sn, acct_id, api_id, access_token));
params.put("request_sn", request_sn);

// Step4. 调用接口
String res = RequestHelper.doGet(server_url + suffix, params, true);

// Step5. 解析返回数据
System.out.println("原始返回数据: " + res);
Map<String, Object> map = new ObjectMapper().readValue(res,
    new TypeReference<Map<String, Object>>() {
    });
String retcode = map.get("retcode").toString();
if (retcode.equals("000000")) {
    //retdata 为加密节点
    String decodeResp = decodeData(map.get("retdata").toString(), acct_id);
    map.put("retdata", decodeResp);
} else
    System.out.println("解密后返回数据:" + JSONObject.toJSONString(map, true));
```

6 附录

6.1 产品编目

产品代码	名称	类型	方式
P_C_B260	产品-公积金-获取目前支持的城市信息	通用	同步
P_C_B261	产品-公积金-根据当前城市编码获取当前城市的公积金登陆方式及表单	通用	同步
P_C_B262	产品-公积金-提交公积金采集请求	通用	同步
P_C_Q010	产品-公积金-根据序列号获取公积金原始数据	通用	同步

6.2 传入参数详解

产品	参数 code	参数名称	可否为空	描述&备注
P_C_B260 产品-公积金-获取目前支持的城市信息	无	无	无	获取目前公积金支持的省市信息
P_C_B261 产品-公积金-根据当前城市编码获取当前城市的公积金登陆方式及表单	regioncode	城市编码	否	城市编码该值来自接口 P_C_B260
P_C_B262 产品-公积金-提交公积金采集请求	website_en	数据源英文缩写	否	来自接口 P_C_B261
	website_sort	数据源编码	否	来自接口 P_C_B261
	submit_type	采集方式	否	来自接口 P_C_B261
	id_card_num	身份证号	是	非必要参数, 具体取值参考 P_C_B261 接口返回值 parameter_code 对应的值
	name	姓名	是	非必要参数, 具体取值参考 P_C_B261 接口返回值 parameter_code 对应的值
	account	公积金网站账号	是	非必要参数, 具体取值参考 P_C_B261 接口

				返回值 parameter_code 对应的值
	password	公积金网站密码	是	非必要参数, 具体取值参考 P_C_B261 接口 返回值 parameter_code 对应的值
	cell_phone_num	手机号	是	非必要参数, 具体取值参考 P_C_B261 接口 返回值 parameter_code 对应的值
	.			非必要参数, 具体取值参考 P_C_B261 接口 返回值 parameter_code 对应的值
P_C_Q010 产品-公积金-根据序列号 获取公积金原始数据	request_id	交易序列号	否	接口 P_C_B262 返回的序列号

6.3 返回数据详解

产品	数据项 ID	数据项名称	可否 为空	描述&备注
P_C_B260 产品-公积金-获取目 前支持的城市信息	support_province_num	支持的省(市、直辖市) 个数	是	
	province_i	省(市、直辖市)具体 信息		父节点, 有多个。 i 为数字, 取值从 1 开 始, 最大值和 support_province_num 一致
	NAME	省市名称	是	
	CODE	省市子编号	是	
	FULL_CODE	省市编码	是	该字段用于 P_C_B261 接口的入参
	LEVELS	当前省市级别	是	省为 1, 市为 2, 县为 3
P_C_B261 产品-公积金-根据当 前城市编码获取当	regioncode	省市编码	是	与传入参数一致
	data	数据节点		父节点, 可能有多个子 节点

前城市的公积金登陆方式及表单	website	数据源英文缩写	否	用于 P_C_B262 接口
	name	数据源中文名称	否	对 website 的中文描述
	tabs	表单内容	否	父节点, 但是为 data 子节点
	type	采集方式	否	用于 P_C_B262 接口
	sort	数据源编码	否	用于 P_C_B262 接口
	descript	登陆方式描述	是	
	field	登录公积金网站参数信息	否	父节点, tabs 的子节点
	parameter_name	参数描述	否	
	parameter_code	参数 id	否	用于 P_C_B262 接口
	grab_type	动态参数标识	否	用于 P_C_B262 接口取值为 0 或 1。
P_C_B262 产品-公积金-提交公积金采集请求	request_id	交易序列号	否	每次提交采集请求都会返回一个交易序列号。交易成功的交易序列号可用于抓取数据
P_C_Q010 产品-公积金-根据序列号获取公积金原始数据	house_fund_basic	公积金基本信息	否	内容为 json 格式。父节点
	update_time	数据获取时间	是	
	housing_fund_status	公积金账户状态	是	
	company	所属单位	是	
	last_fund_date	末次缴存年	是	
	id_card	身份证号	是	
	real_name	登记姓名	是	
	fund_amt	月缴存额	是	
	fund_num	公积金账号	是	
	open_date	开户日期	是	
	balance	账户余额	是	
	company_code	公司编码	是	
	employee_code	员工编码	是	
	transfer_amount	转出金额	是	
	open_bank_name	开户银行名	是	
	open_bank_account	开户银行账号	是	
	deposit_ratio_personal	个人缴纳比例	是	
	deposit_ratio_company	单位缴纳比例	是	
	pay_base_amount	缴纳基数	是	
	house_fund_details	公积金缴纳明细信息	否	内容为 json 数据, 父节点

	note	业务描述	是	
	trading_amt	转入余额(单位：元)	是	
	trading_date	交易日期	是	
	transfer_amount	转出余额(单位：元)	是	
	pay_base	缴纳基数	是	
	house_fund_loans	贷款信息	否	没有贷款信息该项返回[]
	loan_basic	贷款基本信息	是	父节点， house_fund_loans 子项
	loan_name	贷款人姓名	是	
	loan_idcard	贷款人身份证号	是	
	company_name	单位名称	是	
	bank_name	受托银行	是	
	bank_account	还款账户	是	
	loan_num	贷款合同编号	是	
	loan_amount	贷款金额	是	
	loan_length	贷款年限	是	
	loan_rate	贷款利率	是	
	loan_status	还款状态	是	
	loan_date	放款日期	是	
	house_type	购房类型	是	
	guarantee_type	担保方式	是	
	payment_type	还款方式	是	
	payment_day	月还款日期	是	
	payment_amount	月还款金额	是	
	due_principal	已还本金	是	
	overdue_times	逾期次数	是	
	overdue_amount	逾期金额	是	
	loan_payment	贷款偿还信息	是	父节点， house_fund_loans 子项
	payment_date	还款日期	是	
	payment_num	还款期次	是	
	payment_principal	还款本金	是	
	payment_interest	还款利息	是	
	payment_penalty	偿还罚息	是	
	payment_sum	本息合计	是	
	prin_balance	本金余额	是	

	payment_describe	还款描述	是	
	loan_overdue	贷款逾期信息	是	父节点， house_fund_loans 子项
	overdue_date	逾期日期	是	
	overdue_principal	逾期本金	是	
	overdue_interest	逾期利息	是	
	overdue_penalty	逾期罚息	是	
	overdue_summary	合计	是	

6.4 返回码(业务)

code	名称
warn_100	request_id 不存在,请使用正确的 request_id!
warn_103	非法采集请求!
warn_104	密码错误!
warn_108	密码过于简单无法登陆!
warn_109	运营商或电商网站异常无法获取数据!
warn_110	未完成所有采集请求, 请继续!
warn_111	身份证号码不符合规范!
warn_114	生成安全码失败, 请稍后再试!
warn_117	传入参数错误!
warn_118	缺少必要参数!
warn_121	需要再次提交采集请求!

6.5 返回码(系统)

code值	说明
000000	交易成功
err_000	交易失败
err_001	您的账户余额不足!
err_002	您的账户不存在或未被授权!
err_003	您的账户被锁定
err_004	您的账户被注销
err_005	您的账户被销户
err_006	您的账户已到期
err_007	数字签名校验不通过
err_008	客户端回调失败
err_010	您没有权限查询此产品
err_011	无合适的适配产品
err_012	传入参数不正确
err_013	对应产品已过期
err_015	数据源请求时出现异常!
err_016	模型计算请求时出现异常!
err_017	数据源请求超时!
err_018	模型计算请求超时!
err_019	您的令牌已失效或许可次数已到,请联系您的万达征信客户经理!
err_020	您的签名安全校验未通过,如有疑问,请联系您的万达征信客户经理!
err_021	您的证书安全校验未通过,如有疑问,请联系万达征信公司!
err_022	您的 IP 安全校验未通过,如有疑问,请联系万达征信公司!

7 报文样例

7.1 P_C_B260 报文样例

获取支持的城市信息返回报文
<pre>{ "retdata":{ "province_1":[</pre>

```

    {
        "NAME": "合肥",
        "CODE": "3610",
        "FULL_CODE": "00343610",
        "LEVELS": "2"
    },
    {
        "NAME": "安徽省直公积金",
        "CODE": "3401",
        "FULL_CODE": "00343401",
        "LEVELS": "2"
    },
    {
        "NAME": "安徽",
        "CODE": "0034",
        "FULL_CODE": "0034",
        "LEVELS": "1"
    }
],
"province_2": [
    {
        "NAME": "西宁",
        "CODE": "8510",
        "FULL_CODE": "00638510",
        "LEVELS": "2"
    },
    {
        "NAME": "青海省省直公积金",
        "CODE": "6301",
        "FULL_CODE": "00636301",
        "LEVELS": "2"
    },
    {
        "NAME": "青海",
        "CODE": "0063",
        "FULL_CODE": "0063",
        "LEVELS": "1"
    }
],
"support_province_num": 2
},
"retmsg": "获取支持的省市县地区成功",
"retstatus": "000000"
}

```

7.2 P_C_B261 报文样例

根据当前城市编码获取当前城市的公积金登陆方式及表单返回报文

```
{
  "retdata": [{
    "name": "广州公积金",
    "tabs": [
      {
        "descript": "身份证登录",
        "field": [
          {
            "grab_type": 0,
            "parameter_code": "name",
            "parameter_name": "姓名"
          },
          {
            "grab_type": 0,
            "parameter_code": "id_card_num",
            "parameter_name": "身份证"
          },
          {
            "grab_type": 0,
            "parameter_code": "cell_phone_num",
            "parameter_name": "手机号"
          },
          {
            "grab_type": 0,
            "parameter_code": "password",
            "parameter_name": "密码"
          }
        ],
        "sort": "004458100000-1203",
        "type": "auth_auto"
      },
      {
        "descript": "公积金账号登录",
        "field": [
          {
            "grab_type": 0,
            "parameter_code": "name",
            "parameter_name": "姓名"
          },
          {
            "grab_type": 0,
```

```

        "parameter_code": "id_card_num",
        "parameter_name": "身份证"
    },
    {
        "grab_type": 0,
        "parameter_code": "cell_phone_num",
        "parameter_name": "手机号"
    },
    {
        "grab_type": 0,
        "parameter_code": "fund_account",
        "parameter_name": "公积金账号"
    },
    {
        "grab_type": 0,
        "parameter_code": "password",
        "parameter_name": "密码"
    }
],
"sort": "004458100000-1204",
"type": "auth_auto"
}
},
"website": "housingfund_guangzhou"
}],
"retmsg": "获取公积金登陆方式及表单成功",
"retstatus": "000000"
}

```

7.3 P_C_Q010 报文样例

P_C_Q006 返回结果报文样例
<pre> { "retdata": { "house_fund_basic": { "balance": "96837.28", "company": "*****有限责任公司", "company_code": "1388", "deposit_ratio_company": "0.2", "deposit_ratio_personal": "0.2", "employee_code": "00172", "fund_amt": "2222", "fund_num": "130240105748776", </pre>

```
"housing_fund_status": "正常",
"id_card": "340104000000000011",
"last_fund_date": "2015-06-01 00:00:00",
"open_bank_account": "1302000040105740000",
"open_bank_name": "***银行**支行",
"open_date": "2008-08-01 00:00:00",
"pay_base_amount": "amount",
"real_name": "王二小",
"transfer_amount": "103800",
"update_time": "2015-08-11 10:43:25"
```

```
}",
```

```
"house_fund_details": "[
```

```
{
  "balance": "96837.28",
  "company": "*****有限责任公司",
  "note": "结息",
  "pay_base": "baseee",
  "trading_amt": "1136.94",
  "trading_date": "2015-07-03 00:00:00",
  "transfer_amount": "0"
}
```

```
]",
```

```
"house_fund_loans": [{
```

```
  "loan_basic": {
    "bank_account": "1234567899654",
    "bank_name": "***银行**支行",
    "company_name": "*****有限责任公司",
    "due_principal": "3564.56",
    "guarantee_type": "公司担保",
    "house_type": "二手房",
    "loan_amount": "100000",
    "loan_date": "2012-08-21",
    "loan_idcard": "3401040000000000110011",
    "loan_length": "10",
    "loan_name": "全字段",
    "loan_num": "201206000639",
    "loan_rate": "3.25",
    "loan_status": "正常还款",
    "overdue_amount": "2500",
    "overdue_times": "2",
    "payment_amount": "784.5",
    "payment_day": "20",
    "payment_type": "等额本息"
```

```
  },"
```

```
        "loan_overdue": "[
            {
                "overdue_date": "2015-01-20",
                "overdue_interest": "384.6",
                "overdue_penalty": "0",
                "overdue_principal": "784.5",
                "overdue_summary": "1169.1"
            }
        ]",
        "loan_payment": "[
            {
                "payment_date": "2015-01-20",
                "payment_describe": "测试 describe",
                "payment_interest": "384.6",
                "payment_num": "201501",
                "payment_penalty": "0",
                "payment_principal": "784.5",
                "payment_sum": "1169.1",
                "prin_balance": "balance"
            }
        ]"
    },
    "retmsg": "获取数据成功",
    "retstatus": "000000"
}
```

8 其他

8.1 工具类



SecurityUtils.java



MD5.java

版权说明：

此文档的版权归万达征信服务有限公司所有，作为本系统的最终用户，可以拥有该份文档的使用权，但未征得万达征信服务有限公司的书面批准，不得修改、公布本文档，不得向第三方借阅、出让、出版本文档。