

## 企业对公账户核验

贵州数据宝网络科技有限公司专有

2021年09月13日

## 更新日志

时间	说明
20191219	接口下架
20191220	接口上架

贵州数据宝网络科技有限公司专有

## 1. 接口信息

### 1.1 企业账户信息核验

URL: /government/economic/9034

REQUEST TYPE : post

REQUEST PARAM :

Param name	Param type	Required	Desc
key	string	是	您购买的API的key值
creditcode	string	是	统一社会信用代码
entname	string	是	企业名称
frname	string	是	法人名称
accountNo	string	是	银行账号
accountBank	string	是	账户开户行（部分开户行会进行参数校验）
accountProv	string	是	开户行所在省（部分开户行会进行参数校验）
accountCity	string	是	开户行所在地区（部分开户行会进行参数校验）

RESPONSE PARAM :

Param name	Param type	Desc
code	string	请求code码
message	string	code码说明
data	string	接口返回数据体
seqNo	string	调用唯一标识（如有接口问题，请提供此值）

RESPONSE PARAM data :

Param name	Param type	Desc
orderDate	string	订单日期
orderId	string	订单号

SUCCESS RESPONSE :

```
{
  "code": "10000",
  "message": "成功",
  "data": {
    "orderId": "1573197222446",
    "orderDate": "20191108"
  },
  "seqNo": "6N7247U61911081513"
}
```

ERROR CODE :

Code value	Desc
10000	成功
10001	认证已提交，正在处理中
10002	验证未成功，请联系发卡行

10003	无效的发卡行
10004	无效交易
10005	无效金额
10006	无效卡号
10007	客户取消交易
10008	无效交易响应
10009	此卡已过期
10010	密码错误
10011	余额不足
10012	未开通此功能
10013	交易异常,请联系发卡行
10014	超出金额限制
10015	此卡受限制,请联系发卡行
10016	超出取款次数限制
10017	超出最大输入密码次数,请联系发卡行
10018	交易超时,请重试
10019	交易重复,请稍后查询结果
10020	密码格式错误
10021	银行卡与姓名不符
10022	银行卡验证信息及身份信息或手机号输入不正确
10023	未查询到卡BIN
10024	未查询到归属地
10025	验证不通过
10026	无此号,请到户籍所在地进行核查
10027	卡有效,暂无相关消费信息
10028	无效证件类型
10029	无效证件号
10030	无效手机号
10031	无效姓名
10032	重复开票
10033	找不到抬头信息
10034	找不到商户信息
10035	找不到发票信息
10036	手机号、证件号、姓名不一致
10037	手机号一致,证件号和姓名不一致
10038	手机号和证件号一致,姓名不一致
10039	手机号和姓名一致,证件号不一致
10040	未查得
10041	非正常号码
10042	用户未实名
10043	查询成功且无结果
10044	用户验证失败
10045	用户过期
10046	用户禁用
10047	手机号非法或不存在
10048	活体检测通过,照片比对失败
10049	活体检测不通过

10050	转账中，请查看银行账户
10051	无数据
10052	识别失败
10053	不支持验证
10054	系统异常
10055	超出权限
10056	付款银行名称有误
10057	收款账户不支持此业务
10058	收款方账户状态异常或已注销
10059	银行查无此账号
10060	照片质量不符
10061	照片大小不符合要求
10062	交易未成功，请稍后重试
10063	请求失败
10064	未查到工商信息，请核对企业类型或上送的信息是否正确
10065	收款账户名称和账户号码不一致
10066	企业名称与法人信息不一致

## 1.2企业打款信息核验

REQUEST TYPE : postURL: /communication/personal/9140

REQUEST TYPE : post

REQUEST PARAM :

Param name	Param type	Required	Desc
key	string	是	您购买的API的key值
amount	string	是	金额(单位:分;格式:正整数;时效性:48h;如果第一次输入金额带了小数点,属于格式错误,允许再校验一次.如果是金额输错了,例如送了10,则第二次再送正确金额,也不会成功.)
orderDate	string	是	订单日期 (输入企业对公账户核验接口:获取订单日期)
orderId	string	是	订单号 (输入企业对公账户核验接口:获取订单号)

RESPONSE PARAM :

Param name	Param type	Desc
code	string	请求code码
message	string	code码说明
data	string	接口返回数据体
seqNo	string	调用唯一标识 (如有接口问题, 请提供此值)

RESPONSE PARAM data :

Param name	Param type	Desc
------------	------------	------

SUCCESS RESPONSE :

```
{
  "code": "10000",
  "message": "成功",
  "data": null,
  "seqNo": "45YC5E7U1911081539"
}
```

ERROR CODE :

Code value	Desc
10000	成功
10001	认证已提交，正在处理中
10002	验证未成功,请联系发卡行
10003	无效的发卡行
10004	无效交易
10005	无效金额
10006	无效卡号
10007	客户取消交易
10008	无效交易响应
10009	此卡已过期
10010	密码错误
10011	余额不足
10012	未开通此功能
10013	交易异常,请联系发卡行
10014	超出金额限制
10015	此卡受限制,请联系发卡行
10016	超出取款次数限制
10017	超出最大输入密码次数,请联系发卡行
10018	交易超时,请重试
10019	交易重复,请稍后查询结果
10020	密码格式错误
10021	银行卡与姓名不符
10022	银行卡验证信息及身份信息或手机号输入不正确
10023	未查询到卡BIN
10024	未查询到归属地
10025	验证不通过
10026	无此号，请到户籍所在地进行核查
10027	卡有效，暂无相关消费信息
10028	无效证件类型
10029	无效证件号
10030	无效手机号
10031	无效姓名
10032	重复开票
10033	找不到抬头信息
10034	找不到商户信息
10035	找不到发票信息
10036	手机号、证件号、姓名不一致
10037	手机号一致，证件号和姓名不一致
10038	手机号和证件号一致，姓名不一致

10039	手机号和姓名一致，证件号不一致
10040	未查得
10041	非正常号码
10042	用户未实名
10043	查询成功且无结果
10044	用户验证失败
10045	用户过期
10046	用户禁用
10047	手机号非法或不存在
10048	活体检测通过，照片比对失败
10049	活体检测不通过
10050	转账中，请查看银行账户
10051	无数据
10052	识别失败
10053	不支持验证
10054	系统异常
10055	超出权限
10056	付款银行名称有误
10057	收款账户不支持此业务
10058	收款方账户状态异常或已注销
10059	银行查无此账号
10060	照片质量不符
10061	照片大小不符合要求
10062	交易未成功，请稍后重试
10063	请求失败

## 2. 全系统错误码

Code value	Desc
SYSTEM_900	IP 不合法
SYSTEM_999	接口处理异常
SYSTEM_000	key 参数不能为空
SYSTEM_001	找不到这个 key
SYSTEM_002	调用次数已用完
SYSTEM_003	用户该接口状态为不可用
SYSTEM_004	接口信息不存在
SYSTEM_005	你没有认证信息
SYSTEM_008	当前接口只允许“企业认证”通过的账户进行调用，请在数据宝官网个人中心进行企业认证后再进行调用，谢谢！
SYSTEM_009	必须认证审核通过才可以使用
SYSTEM_011	接口缺少参数
SYSTEM_012	没有 ip 访问权限
SYSTEM_013	接口模板不存在
SYSTEM_014	接口模板没开启
SYSTEM_015	该接口已下架
SYSTEM_017	模板配置的平台参数与请求的参数不一致
SYSTEM_019	调用第三方协议配置错误
SYSTEM_020	调用第三方产生异常
SYSTEM_022	调用第三方返回的数据格式错误
SYSTEM_025	你没有购买此接口
SYSTEM_026	用户信息不存在
SYSTEM_027	请求第三方地址超时，请稍后再试
SYSTEM_028	请求第三方地址被拒绝，请稍后再试
SYSTEM_029	返回示例错误
SYSTEM_034	签名不合法
SYSTEM_035	请求参数加密有误
SYSTEM_036	验签失败
SYSTEM_037	timestamp 不能为空



SYSTEM_038	请求繁忙，请稍候再试
SYSTEM_039	请在个人中心接口设置加密状态
SYSTEM_040	timestamp 不合法
SYSTEM_041	timestamp 过期
SYSTEM_042	身份证手机号等不符合规则
SYSTEM_043	该号段不支持验证
SYSTEM_047	请在个人中心获取密钥
SYSTEM_048	找不到这个 secretKey

### 3. 接口对接示例代码

#### 3.1 sample code

```
import org.apache.http.HttpResponse;
import org.apache.http.NameValuePair;
import org.apache.http.client.config.RequestConfig;
import org.apache.http.entity.UrlEncodedFormEntity;
import org.apache.http.client.methods.HttpPost;
import org.apache.http.impl.client.CloseableHttpClient;
import org.apache.http.impl.client.HttpClients;
import org.apache.http.message.BasicNameValuePair;
import org.apache.http.util.EntityUtils;
import java.util.ArrayList;
import java.util.HashMap;
import java.util.List;
import java.util.Map;

public class HttpUtil {
    public static void main(String[] args) {
        //接口地址
        String url = "http://api.chinadatapay.com/trade/user/1985";
        //请求参数
        Map<String, Object> params = new HashMap<>();
        //输入数据宝提供的 key
        params.put("key", "");
        //输入局被查询手机号码
        params.put("mobile", "");
        String result = null;
        try {
            result = post(url, params);
        } catch (Exception e) {
            e.printStackTrace();
        }
        System.out.println("result:\n" + result);
    }
}
```

```

public static String post(String url, Map<String, Object> params) throws Exception
{
    ArrayList<NameValuePair> pairs = covertParams2NVPS(params);
    return PostHttpRequest(url, pairs);
}

public static String PostHttpRequest(String Url, List<NameValuePair> params) throws
Exception {
    CloseableHttpClient client = HttpClients.createDefault();
    //超时时间
    RequestConfig requestConfig = RequestConfig.custom()
        .setSocketTimeout(300000)
        .setConnectTimeout(300000)
        .build();
    String result = null;
    try {
        HttpPost request = new HttpPost(Url);
        request.setConfig(requestConfig);
        request.setEntity(new UrlEncodedFormEntity(params, "UTF-8"));
        HttpResponse responses = client.execute(request);
        if (responses.getStatusLine().getStatusCode() == 200) {
            result = EntityUtils.toString(responses.getEntity(), "UTF-8");
        }
    } catch (Exception e) {
        e.printStackTrace();
    } finally {
        client.close();
    }
    return result;
}

private static ArrayList<NameValuePair> covertParams2NVPS(Map<String, Object>
params) {
    ArrayList<NameValuePair> pairs = new ArrayList<>();
    if (params == null || params.size() == 0) {
        return pairs;
    }
    for (Map.Entry<String, Object> param : params.entrySet()) {
        Object value = param.getValue();
        if (value instanceof String[]) {
            String[] values = (String[]) value;
            for (String v : values) {
                pairs.add(new BasicNameValuePair(param.getKey(), v));
            }
        } else {
            if (value instanceof Integer) {
                value = Integer.toString((Integer) value);
            } else if (value instanceof Long) {
                value = Long.toString((Long) value);
            }
            pairs.add(new BasicNameValuePair(param.getKey(), (String) value));
        }
    }
    return pairs;
}
}

```

## 4. 加密对接说明

我们的加密方式使用 MD5 进行 sign 签名验证，以此来杜绝信息篡改的发生，同时针对入参和出参内容进行对称加密（base64 编码）。以下具体讲解对接方式。

对接步骤如下：

### 4.1 设置加密

第一步：登陆 <https://www.chinadatapay.com>, 首页 > 个人中心 > 基础服务 > 我的数据宝 点击”获取密钥”，如下图，举例 secretKey:IgJKiiakupIMXy4s



第二步：首页 > 个人中心 > 基础服务 > 我的 API 选择某个接口操作栏中的”数据服务”按钮，弹框中选择”加密对接”



### 4.2 技术对接

接口的入参和出参参数名无需使用 base64 编码，只是针对入参值和出参值使用密码加解密。案例如下：

Url : <http://api.chinadatapay.com/communication/personal/1896>

RequestWay : Post

**SecretKey : IgJKiiakupIMXy4s**

Request Param:

名称	类型	必填	说明
key	string	是	您购买的 API 的 key 值
name	string	是	姓名
idcard	string	是	身份证号
mobile	string	是	手机号

e. g

key=您购买的 API 的 key 值

name=张三&idcard=342623199801011234&mobile=15058571234&timestamp=1505352152882

开始处理如下：

第一步、BASE64AES 加密入参：**备注：key 字段不参与加密**

name=a69V/TNqx/9cPTSjWq26QQ==&idcard=GxQ8ooj8Vhn7X3qDHNe2DxfSKoIMhpeszFLfoi3Sgek=&mobile=P5U2chrL933LhMIxuVg5Aw==

timestamp=1D5VSE5xkjgH88Sj13FQSw==

第二步、sign 令牌获取：

规则为入参按照 ACS 码排序，结果如下：

idcard=GxQ8ooj8Vhn7X3qDHNe2DxfSKoIMhpeszFLfoi3Sgek=&mobile=P5U2chrL933LhMIxuVg5Aw==&name=a69V/TNqx/9cPTSjWq26QQ==&timestamp=1D5VSE5xkjgH88Sj13FQSw==

**备注：key 字段不参与签名**

sign=new BASE64Encoder().encode(md5(入参按照 ACS 码排序结果)) 获得入参

sign=7c20PORFWywW61Z6oYwAXQ==

第三步、开始发送请求：

url: http://api.chinadatipay.com/communication/personal/1896

requestWay:post

Request:

key=您购买的 API 的 key 值

name=a69V/TNqx/9cPTSjWq26QQ==&idcard=GxQ8ooj8Vhn7X3qDHNe2DxfSKoIMhpeszFLfoi3Sgek=&mobile=P5U2chrL933LhMIxuVg5Aw==&timestamp=1D5VSE5xkjgH88Sj13FQSw==

sign=7c20PORFWywW61Z6oYwAXQ==

Response:

```
{
  "code": "10000",
  "message": "成功",
  "data": "L58pwLVQ7L8ZarUV0gpCLA==",
  "seqNo": "C03SX59Z1709071444"
}
```

data 值秘钥进行 BASE64AES 解密，解密结果如下：{"state": "1"}

### 4.3 java 代码示例

引入 cdp-common-security-3.5.0.jar(请联系客服人员获取)



cdp-common-security-3.5.0.jar

BASE64AES 加密工具类和方法:

com.cdp.product.security.encode.CdpEncryptUtil.aesEncrypt(明文,秘钥)

BASE64AES 解密工具类和方法:

com.cdp.product.security.decode.CdpDecryptUtil.aesDecrypt(密文,秘钥) Sign

签名工具类和方法: CdpSignUtil.sign(Map<String, String> param) 加密解密代

码示例如下:

```
package com.cdp.product.security.test;

import com.cdp.product.security.decode.CdpDecryptUtil;
import com.cdp.product.security.encode.CdpEncryptUtil;
import com.cdp.product.security.exception.DecryptFailureException;
import com.cdp.product.security.exception.EncryptFailureException;
import com.cdp.product.security.exception.SignFailureException;
import com.cdp.product.security.sign.CdpSignUtil;

import java.util.HashMap;
import java.util.Map;

/**
 * 测试加解密以及签名
 */
public class CommonTest {
    public static void main(String[] args) throws EncryptFailureException,
        SignFailureException, DecryptFailureException {
        //秘钥
        String secretKey = "lgJKiiakuplMXy4s";
        //入参集合 针对入参 value 值进行加密
        Map<String, String> param = new HashMap<>();
        param.put("name", CdpEncryptUtil.aesEncrypt("张三", secretKey));
        param.put("idcard", CdpEncryptUtil.aesEncrypt("342623199801011234", secretKey));
        param.put("mobile", CdpEncryptUtil.aesEncrypt("15058571234", secretKey));
        param.put("timestamp", CdpEncryptUtil.aesEncrypt(System.currentTimeMillis() +
            "", secretKey));
        //获取 sign 签名
        String sign = CdpSignUtil.sign(param);
        //返回各种加解密签名结果
        System.out.println(String.format("入参集合:\n%s", param));
        System.out.println(String.format("获取 sign 值:\n%s", sign));
        System.out.print(String.format("返回结果解密:\n%s",
            CdpDecryptUtil.aesDecrypt("81hwJ3Fzj4De9fNjccustQ==", secretKey)));
    }
}
```

结果示例:

入参集合:

```
{timestamp=ID5VSE5xkjgH88Sjl3FQSw==,name=a69V/TNqx/9cPTSjWq26QQ=  
=, idcard=GxQ8ooj8Vhn7X3qDHNe2DxfSKoIMhpeszFLfoi3Sgek=, mobile=  
P5U2chrL933LhMlxuVg5Aw==}
```

获取 sign 值:

```
7c20PORFWywW6lZ6oYwAXQ==
```

返回结果解密:

```
{"state": "1"}
```

贵州数据宝网络科技有限公司专有