

## 企业经营凭证核验

贵州数据宝网络科技有限公司专有

2020年09月28日

## 更新日志

时间	说明
----	----

贵州数据宝网络科技有限公司专有

## 1. 接口信息

### 1.1 企业经营凭证核验

URL: /government/economic/2286

REQUEST TYPE : post

REQUEST PARAM :

Param name	Param type	Required	Desc
key	string	是	您购买的API的key值
InvCode	string	是	发票代码
InvNo	string	是	发票号码
InvTime	string	是	开票日期(格式:YYYYMMDD)
InvOther	string	否	不含税金额(发票类型为增值税专用发票、货运运输业增值税专用发票、机动车销售统一发票、二手车发票时此字段不可为空, 增值税专用发票、货运运输业增值税专用发票、机动车销售统一发票填写不含税金额; 二手车发票, 填写发票车价合计, 金额格式: 90.00)
checkCode	string	否	校验码(校验码后六位(发票类型为增值税普通发票、增值税普通发票(电子)、增值税普通发票(卷式)、通行费发票时不可为空))

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
invoiceList	string	发票列表
invoiceInfo	string	发票信息
bandModel	string	厂牌型号(机动车票返回信息)
blueInvoiceCode	string	蓝票发票代码
blueInvoiceNo	string	蓝票发票号码
businessUnit	string	经营、拍卖单位(二手车发票返回信息)
businessUnitAddress	string	经营、拍卖单位地址(二手车发票返回信息)

businessUnitBankAndAccount	string	开户银行及账号(二手车发票返回信息)
businessUnitPhone	string	经营、拍卖单位电话(二手车发票返回信息)
businessUnitTaxNo	string	经营、拍卖单位纳税人识别号(二手车发票返回信息)
buyerAccount	string	购方开户行及账号
buyerAddressPhone	string	购方地址、电话
buyerName	string	购方名称
buyerPhone	string	购方电话
buyerTaxNo	string	购方税号
buyerUnitCodeOrIdNo	string	买方单位代码(二手车发票返回信息)
buyerUnitOrIndividual	string	买方单位/个人(二手车发票返回信息)
buyerUnitOrIndividualAddress	string	买方单位/个人地址(二手车发票返回信息)
cancellationMark	string	作废标志(Y:已作废;N:未作废;3:红冲)
carNumber	string	车种车号(货运发票返回信息)
carPrice	string	车价合计(二手车发票返回信息)
carrierName	string	承运人名称(货运发票返回信息)
carrierTaxNo	string	承运人识别号(货运发票返回信息)
certificateOfImport	string	进口证明书号(货运发票返回信息)
checkCode	string	校验码
checkCount	string	查验次数
commodityInspectionNo	string	商检单号(机动车票返回信息)
consignorName	string	发货人名称(货运发票返回信息)
consignorTaxNo	string	发货人识别号(货运发票返回信息)
detailList	string	发票详情
detailAmount	string	金额
detailNo	string	明细编号
expenseItem	string	费用项目(货运发票返回信息)
goodsName	string	货物名称
num	string	数量
plateNo	string	车牌号(通行费发票返回信息)
specificationModel	string	规格型号
taxAmount	string	税额
taxDetailAmount	string	含税金额
taxRate	string	税率
taxUnitPrice	string	含税单价
trafficDateEnd	string	通行日期止(通行费发票返回信息)

trafficDateStart	string	通行日期起(通行费发票返回信息)
type	string	类型
unit	string	计量单位
unitPrice	string	不含税单价
draweeName	string	受票方名称
draweeTaxNo	string	受票方识别号
drawer	string	开票人
engineNo	string	发动机号
idNo	string	购方身份证号/组织机构代码
invoiceAmount	string	发票金额
invoiceCode	string	发票代码
invoiceDate	string	开票日期
invoiceNo	string	发票号码
invoiceType	string	发票种类
invoiceTypeCode	string	发票种类代码
invoiceTypeNo	string	发票种类号码
lemonMarket	string	二手车市场(二手车发票返回信息)
lemonMarketAddress	string	二手车市场地址(二手车发票返回信息)
lemonMarketBankAndAccount	string	二手车市场开户银行及账号(二手车发票返回信息)
lemonMarketPhone	string	二手车市场电话(二手车发票返回信息)
lemonMarketTaxNo	string	二手车市场纳税人识别号(二手车发票返回信息)
licensePlate	string	车牌照号(二手车发票返回信息)
limitedPeopleCount	string	限乘人数(机动车票返回信息)
machineNo	string	机器编号
payee	string	收款人
produceArea	string	产地(机动车票返回信息)
qualifiedNo	string	合格证号(机动车票返回信息)
receiveName	string	收货人名称(机动车票返回信息)
receiveTaxNo	string	收货人识别号(机动车票返回信息)
registrationNo	string	登记证号(二手车发票返回信息)
remark	string	备注
resultCode	string	发票验证结果代码
resultTip	string	发票验证结果描述
reviewer	string	复核人
salerAccount	string	销方开户行及账号
salerAddress	string	销方地址
salerAddressPhone	string	销方地址/电话
salerBankAccount	string	销方开户账号
salerBankName	string	销方开户银行

salerName	string	销方名称
salerPhone	string	销方电话
salerTaxNo	string	销方税号
sellerPhone	string	卖方电话
sellerUnitCodeOrIdNo	string	卖方单位代码
sellerUnitOrIndividual	string	卖方单位/个人
sellerUnitOrIndividualAddress	string	卖方单位地址
taxAmount	string	发票税额
taxAuthorityCode	string	主管税务机关代码
taxAuthorityName	string	主管税务机关名称
taxDiskNumber	string	税控盘号
taxPaymentCertificateNo	string	完税凭证号码
taxRate	string	税率
throughAddress	string	起运地/经由/到达地(货运发票返回信息)
tonnage	string	吨位(动车票返回信息)
totalAmount	string	合计金额
trafficFeeFlag	string	通行费标志(Y-可抵扣通行费, N-不可抵扣通行费)
transferredVehicleOffice	string	转入地车辆车管所名称(二手车发票返回信息)
transportGoodsInfo	string	运输货物信息(货运发票返回信息)
vehicleIdentificationNo	string	车辆识别代号/车架号码(机动车票、二手车票返回信息)
vehicleTonnage	string	车船吨位(货运发票返回信息)
vehicleType	string	车辆类型(机动车票、二手车票返回信息)
zeroTaxRateFlag	string	零税率标志(通行费票返回信息: 空-非零税率, 1- 税率栏位显示)
resultCode	string	发票查验结果代码
resultTip	string	发票查验结果描述

SUCCESS RESPONSE :

```
{
  "code": "10013",
  "message": "查验成功发票一致",
  "data": {
    "invoiceList": [
      {
        "invoiceInfo": {
          "invoiceDate": "20180929",
          "vehicleTonnage": "",
          "carrierTaxNo": "",
          "salerAddressPhone": "南京市****6699",
          "remark": "标书费",
          "buyerUnitOrIndividualAddress": "",
          "registrationNo": "",
          "invoiceNo": "08091783",
          "checkCount": "1",

```

```

"draweeTaxNo": "",
"drawer": "",
"licensePlate": "",
"consignorTaxNo": "",
"blueInvoiceNo": "",
"certificateOfImport": "",
"salerBankAccount": "",
"buyerAddressPhone": "贵州省贵安新区大学城****95",
"reviewer": "",
"invoiceTypeNo": "",
"taxAuthorityName": "",
"draweeName": "",
"salerAddress": "",
"idNo": "",
"taxRate": "",
"checkCode": "574999*****491",
"buyerTaxNo": "91520900MA6DL4YR2P",
"machineNo": "661538009321",
"invoiceType": "04",
"sellerPhone": "",
"trafficFeeFlag": "",
"throughAddress": "",
"taxAmount": "45.28",
"businessUnit": "",
"salerPhone": "",
"invoiceCode": "3200173320",
"businessUnitPhone": "",
"vehicleType": "",
"vehicleIdentificationNo": "",
"produceArea": "",
"buyerUnitOrIndividual": "",
"sellerUnitOrIndividualAddress": "",
"receiveTaxNo": "",
"invoiceAmount": "754.72",
"limitedPeopleCount": "",
"cancellationMark": "N",
"lemonMarketAddress": "",
"lemonMarketBankAndAccount": "",
"businessUnitBankAndAccount": "",
"transportGoodsInfo": "",
"salerTaxNo": "91320000796544096U",
"sellerUnitCodeOrIdNo": "",
"qualifiedNo": "",
"receiveName": "",
"resultCode": "0001",
"totalAmount": "800.00",
"buyerUnitCodeOrIdNo": "",
"salerName": "江苏苏宁贸易有限公司",
"taxPaymentCertificateNo": "",
"invoiceTypeCode": "",
"buyerAccount": "中国工商银行股份有限公司贵阳花溪支行2402****590",
"carPrice": "",
"lemonMarketTaxNo": "",
"consignorName": "",
"businessUnitAddress": "",
"lemonMarket": "",
"blueInvoiceCode": "",
"payee": "",
"commodityInspectionNo": "",
"resultTip": "查验成功发票一致",
"detailList":
[{
  "expenseItem": "",

```

```

        "taxUnitPrice": "",
        "taxRate": "6%",
        "specificationModel": "",
        "unitPrice": "754.7169811320755",
        "detailNo": "1",
        "type": "",
        "unit": "",
        "trafficDateStart": "",
        "num": "1",
        "taxAmount": "45.28",
        "trafficDateEnd": "",
        "taxDetailAmount": "",
        "detailAmount": "754.72",
        "plateNo": "",
        "goodsName": "*企业管理服务*标书费"
    },
    "salerAccount": "中国建设银行南京徐庄软件园支行3200***5976",
    "buyerName": "贵州数据宝网络科技有限公司",
    "bandModel": "",
    "salerBankName": "",
    "zeroTaxRateFlag": "",
    "engineNo": "",
    "taxAuthorityCode": "",
    "lemonMarketPhone": "",
    "transferredVehicleOffice": "",
    "buyerPhone": "",
    "carNumber": "",
    "sellerUnitOrIndividual": "",
    "businessUnitTaxNo": "",
    "carrierName": "",
    "tonnage": "",
    "taxDiskNumber": ""
    }
    },
    "seqNo": "4N5PW8S61812191020"
}

```

ERROR CODE :

Code value	Desc
10000	成功
10001	无权限
10002	发票代码格式有误
10003	校验码不能为空！
10004	校验码格式有误
10005	金额不可以为空
10006	发票号码格式有误
10007	开票日期格式有误
10008	接口查验数据返回为空，请检查！
10009	请求数据为空！请检查
10010	请求数据格式有误！请检查
10011	接口返回异常！
10012	请求参数有误，请检查！
10013	查验成功发票一致
10014	超过该张票当天查验次数
10015	发票查验请求太频繁，请稍后再试
10016	超过服务器最大请求数，请稍后访问
10017	请求不合法



10018	发票信息不一致
10019	所查发票不存在
10020	已超过最大查验量
10021	查询发票不规范
10022	查验异常
10023	网络超时，请重试
10024	内部错误
10025	日期当天不能查验
10026	超过一年不能查验
10027	没有查验权限

贵州数据宝网络科技有限公司专有

## 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	当前您的接口覆盖范围不能满足本次数据验证，请升级 API 接口套餐获取更丰富的服务
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.client.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:lgJKiiakuplMXy4s



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 : lgJKiiakuplMXy4s**

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