

# 深圳 CA WebTrust SSL 证书

## 渠道接口文档

V2.0.8

深圳市电子商务安全证书管理有限公司  
2020 年 6 月 22 日

## 目录

1. 概述.....	4
2. 接口规范说明.....	4
2.1. 接口协议.....	4
2.2. 用户协议书及认证流程.....	5
2.3. 推送信息约定.....	5
2.4. 收集 OV、EV 审核文件责任方的约定.....	6
3. 接口参数规范说明.....	6
3.1. 接口调用参数规范.....	6
3.2. 返回结果规范.....	7
3.3. 签名规范.....	8
4. API 接口.....	8
4.1. 申请证书接口.....	8
4.2. 证书续费接口.....	13
4.3. 上传申请文件接口.....	17
4.4. 查询订单接口.....	20
4.5. 取消订单接口.....	26
4.6. 重签证书接口.....	28
4.7. 获取重签证书详情接口.....	30
4.8. 撤销证书接口.....	33
4.9. 根据订单号获取订单状态.....	35
4.10. 重发域名验证邮件接口.....	37

4.11. 下载《信息确认函》模版.....	3 8
4.12. 获取域名管理员邮箱接口.....	4 1
5. 推送订单状态.....	4 3
5.1. 功能描述.....	4 3
5.2. 推送内容.....	4 3
6. 数据格式与编码表.....	4 4
6.1. 订单 JSON 格式.....	4 4
6.2. 产品编码表.....	4 7
6.3. 订单状态表.....	5 1
6.4. 订单后台状态表.....	5 1
6.5. 文档修改记录.....	5 2

## 1. 概述

深圳 CA 的 SSL 域名证书业务系统提供主要的证书管理接口，渠道合作伙伴的系统通过 SSL 证书 API 接口直接申请证书，实现全自动证书的申请、证书重发、订单取消、认证状态推送和证书的下发等功能。

API 包含以下功能：

- 申请证书接口
- 上传申请文件接口
- 查询订单接口
- 取消订单接口
- 重签证书接口
- 撤销证书接口
- 重发域名验证邮件接口
- 获取域名管理员邮箱接口
- 订单信息推送接口

## 2. 接口规范说明

### 2.1. 接口协议

接口采用 HTTPS POST 通信方式，字符编码格式采用 UTF8，接口传入参数采用 JSON POST Data(Content-Type: application/x-www-form-urlencoded;charset=UTF-8)，返回的数据报文格式采用 JSON。

API 地址：

生产环境的接口地址: <https://ssl.szca.com/api/>

测试环境的接口地址: <https://ssltest.szca.com:19443/api/>

## 2.2. 用户协议书及认证流程

在申请证书接口和上传申请文件接口中, 上传的《深圳 CA SSL 数字证书用户协议》、《深圳 CA SSL 证书申请表》文件需要从下面地址下载模板, 然后打印、填写、盖章、扫描, 再上传。

深圳 CA SSL 数字证书用户协议, 请参考:

<https://ssl.szca.com/doc/download/3.html>

深圳 CA SSL 证书申请表 (适用 OV、EV 型证书), 请参考:

<https://ssl.szca.com/doc/download/5.html>

不同类型的 SSL 域名证书的认证签发流程稍有区别, 具体以参考平台公开的流程为准:

DV 证书申请认证流程, 请参考:

<https://ssl.szca.com/content/4.html>

OV 证书申请认证流程, 请参考:

<https://ssl.szca.com/content/5.html>

EV 证书申请认证流程, 请参考:

<https://ssl.szca.com/content/6.html>

## 2.3. 推送信息约定

当 SSL 证书订单状态发生变化时, 系统将定期自动把订单数据送给合作伙伴的系统。合作伙伴系统应提供一个基于 http 或 https 的接收地址(本文中统一称此地址为 notifyUrl), 用于接收推送信息。如果使用 https 协议, 必须保证 https 使用证书是合法 (浏览器信任) 有效的证书。

若合作伙伴系统不提供 notifyUrl, 则深圳 CA 不主动推送订单数据, 由合作伙伴系统自行查询订单数

据。

## 2.4. 收集 OV、EV 审核文件责任方的约定

由于 OV、EV 类型的 SSL 证书需要收集申请组织机构的证件以及加盖公章的用户协议书和证书申请表。

这些材料用于认证申请 SSL 证书的组织机构的真实身份，以及该组织机构申请 SSL 证书的真实意图。

合作伙伴需要与我方约定由谁来负责收集上述文件。负责方需要与用户联系，引导用户按起规范提供正确的证明材料。

## 3. 接口参数规范说明

渠道商调用深圳 CA 的接口，为保证传送的数据真实可靠，不被篡改，调用前必须对传递的参数做签名。所有接口均包含下表的四个参数。在确定合同时，深圳 CA 会将合作伙伴身份 ID(partnerId)和合作伙伴密钥(accessKeySecret)这两个参数通过安全渠道分发给合作伙伴。同时，若合作伙伴系统需要接收深圳 CA 的推送信息，则需要向深圳 CA 提供 notifyUrl 的地址。

### 3.1. 接口调用参数规范

所有接口，都是需要传递 4 个参数：

参数	必填	类型	参数说明	样例
partnerId	是	String	合作伙伴身份 ID	P001
timestamp	是	String	时间戳(格式：毫秒如 java 用 System.currentTimeMillis()), 中国时区。校验时，时间差应小于 30 分钟。	1633918434258
signData	是	String	数据包签名，算法为 sha256(partnerId+timestamp+accessKeySecret+reqData)	d219a3dd8...7fe a9d7fcaca3

reqData	是	ReqData 对象	JSON 格式的请求数据包	
---------	---	------------	---------------	--

例如：

```
{
  "partnerId" : "P001" ,
  "timestamp" : "1633918434258" ,
  "signData" : "d219a3dd877f943be0ee65ef6a34129778d2fd0568ce2d9a8f37fea9d7fcaca3" ,
  "reqData" :{
    //不同的接口返回的数据结构不同，请参考下面接口的具体描述。
  }
}
```

## 3.2. 返回结果规范

所有接口的返回结果都是一个 JSON 格式的数据包。其中，每个接口的返回结果都包含了 code 与 msg，主要用于判断接口操作是否成功。主要格式如下：

返回参数名称	说明	样例
code	调用接口的错误代码，0 代表操作成功	0
msg	调用接口的错误信息	

例如：

```
{
  "code" :0,
  "msg" : "" ,
}
```

```
"retData" :{
```

```
    //不同的接口返回的数据结构不同，请参考下面接口的具体描述。
```

```
    }
```

```
}
```

### 3.3. 签名规范

签名规范，具体如下：

1. 先将 reqData 数据包对象转换成**嵌套有序**(reqData json 对象的每一层都按照 json key 升序排序)的 json string --->reqDataStr
2. 然后拼接 partnerId+timestamp+accessKeySecret+reqDataStr--->signStr
3. 最后使用 org.apache.commons.codec.digest.sha256Hex(signStr)得到 signData

## 4. API 接口

### 4.1. 申请证书接口

#### 4.1.1.功能描述

该接口用于创建订单，申请 SSL 证书。用户调用一次此接口，即可生成一张 SSL 证书的订单。如果用户申请 DV 类型的 SSL 证书，若调用接口成功，则几分钟内即可签发证书。如果用户申请 OV、EV 类型的 SSL 证书，深圳 CA 收到申请请求后，再核实用户提供的资料正确无误之后，再签发 SSL 证书，若用户的足够资料完备，核实过程顺利，一般 1~3 个工作日内可以签发 OV、EV 类型的 SSL 证书。



### 4.1.2.环境地址

接口相对地址： /api/quickOrder.html

传入参数格式： POST Data。

### 4.1.3.请求参数表

参数	必填	类型	参数说明	样例
partnerId	是	String	合作伙伴身份 ID	P001
timestamp	是	String	时间戳 (中国时区)	1633918434258
signData	是	String	签名数据	d219a3dd8...7fea9 d7fcaca3
reqData	是	ReqData 对象	JSON 格式的请求数据包	

### 4.1.4.reqData 数据包结构

参数	参数说明	样例
productCode	渠道产品编号(必填), 参考 6.6.2 产品编码表章节	CertumOV
domainName	证书域名(必填)	www.szca.com
sanList	多域名证书的附加域名列表, 用逗号 ( , ) 分隔不同的域名(选填)	cps.szca.com,app.s zca.com
csr	证书请求内容(必填)。计算签名值的时候, 因为使用的是 Gson.toJson()方法, 默认转义特殊字符了 ( unicode ) , 比如 == 是转成 \u003d\u003d, 传过来的 csr 本身, 不用做转换的	
<b>sanApprover 域名验证方式(必填)</b>		
approveMethod	验证类型 (必填) EMAIL, DNS, FILE;	DNS
approverEmail	验证类型是 DNS, FILE 时必填, 接收验证码的邮箱	123@qq.com
approverEmailPrefix	验证类型是 EMAIL 时必填, 域名的管理邮箱前	ADMIN

	缀，取值范围：ADMIN,ADMINISTRATOR, POSTMASTER, HOSTMASTER, WEBMASTER	
<b>contactInfo 联系人信息(必填)</b>		
contactName	联系人的姓名(必填)	
contactPhone	联系人电话(必填)	
contactEmail	联系人 Email 地址(必填)，用于发送如到期提醒等一般性的通知(必填)	
contactAddr	联系地址(选填)	
contactPostalCode	联系邮政编码(选填)	
<b>orgInfo 机构信息(DV 选填，OV、EV 必填)</b>		
orgName	组织机构名称(必填)	
orgType	机构性质(必填选填)：国家机关事业单位   商业企业单位   私有组织   非营利性组织	orgType,orgIdType,orgIdCode 三个字段对于 szca 品牌,后续线下提供资料后,暂由 szca 后台人工补齐订单信息
orgUnit	证书使用部门(必填)	
orgAddr	机构注册地址(必填)	
orgCity	城市(必填)	
orgState	省份/直辖市(必填)	
orgCountry	国家编码(默认是 CN, 可选)	
orgIdType	组织机构的身份证件类型(必填选填)，从下面三个之中选一个：组织机构代码证   营业执照   统一社会信用代码	
orgIdCode	组织机构的身份证件号码，如统一社会信用代码号码 (选填)	
orgPhone	机构电话	
orgPostalCode	机构邮编	
<b>uploadFiles 上传文件 arraylist 格式 (可选)</b>		
content	BASE64 格式的文件内容	可选
docType	内容只能如下： operatorIdFile("经办人身份证正反两面"), applicationFile("申请表及授权书"), orgRegisterDocumentFile("营业执照"), agreementFile("用户协议"),	可选

	domainCert("域名证书"), confirmLetter("信息确认函"), --非 certum revokeLetter("证书吊销确认函"), --非 certum	
fileName	文件名称	可选
pagerType	正面, 反面, 双面	可选

注：对于 operatorIdFile、applicationFile、orgRegisterDocumentFile、domainCert、domainCertFile、confirmLetter 等几个文件参数，在本接口中，都是可选项。对于 OV、EV 类型的证书，若本接口不上传上述文件，则可以在下单成功后，再通过调用“上传申请文件接口”上传上述文件。若合作伙伴与我方约定由深圳 CA 负责收集上述文件，则深圳 CA 收到 OV、EV 申请后再与用户联系，引导用户按我方规范要求提供上述文件。

JSON 请求样例：

```
{  
  
  "partnerId": "p10030",  
  
  "timestamp": "1627742811557",  
  
  "signData":  
  
  "0c4f5a5e56d75c099706182edc80d1347d85968378f7e2faf641619c0a3d8d63",  
  
  "reqData": {  
  
    "productCode": "DigiCertOV",  
  
    "csr": "-----BEGIN CERTIFICATE REQUEST...END CERTIFICATE REQUEST-----",  
  
    "sanApprover": {  
  
      "approveMethod": "EMAIL",  
  
      "approverEmailPrefix": "ADMIN"  
  
    },  
  
  },  
  
}
```

```
"contactInfo": {  
  
    "contactName": "liangyh",  
  
    "contactAddr": "联系地址",  
  
    "contactPhone": "15678987654",  
  
    "contactPostalCode": "510000",  
  
    "contactEmail": "liangyh@szca.com"  
  
},  
  
"orgInfo": {  
  
    "orgUnit": "IT 部门",  
  
    "orgCity": "机构城市 gz",  
  
    "orgIdCode": "yyzz100",  
  
    "orgCountry": "CN",  
  
    "orgType": "商业企业单位",  
  
    "orgAddr": "机构注册地址 gdgz",  
  
    "orgIdType": "营业执照",  
  
    "orgName": "企业名称",  
  
    "orgState": "广东"  
  
},  
  
"domainName": "dcov7308.jdjj.xin"  
  
}  
  
}
```

### 4.1.5. 返回结果

返回参数名称	说明	样例
code	调用接口的错误代码，0 为成功	0
msg	调用接口的错误信息	
data	若 code!=0,则此项为空；如 code=0, 则返回订单号	21073122001

JSON 返回样例：

```
{"data":"21073122001","code":0,"msg":"成功"}
```

## 4.2. 证书续费接口

### 4.2.1. 功能描述

该接口用于续期原订单，和下单接口大致一样，新增必填字段“原订单号”

### 4.2.2. 环境地址

接口相对地址： /api/renewCert.html

传入参数格式：POST Data。

### 4.2.3. 请求参数表

参数	必填	类型	参数说明	样例
partnerId	是	String	合作伙伴身份 ID	P001
timestamp	是	String	时间戳 (中国时区)	1633918434258
signData	是	String	签名数据	d219a3dd8...7fea9d7fcaca3
reqData	是	ReqData 对象	JSON 格式的请求数据包	

#### 4.2.4.reqData 数据包结构

参数	参数说明	样例
orderId	原订单号 (必填)	
productCode	渠道产品编号(必填), 参考 6.6.2 产品编码表章节	CertumOV
domainName	证书域名(必填)	www.szca.com
sanList	多域名证书的附加域名列表, 用逗号 (, ) 分隔不同的域名(选填)	cps.szca.com,app.szca.com
csr	证书请求内容(必填)	
<b>sanApprover 域名验证方式(必填)</b>		
approveMethod	验证类型 (必填) EMAIL, DNS, FILE	DNS
approverEmail	验证类型是 DNS, FILE 时必填, 接收验证码的邮箱	123@qq.com
approverEmailPrefix	验证类型是 EMAIL 时必填, 域名的管理邮箱前缀, 取值范围: ADMIN,ADMINISTRATOR, POSTMASTER, HOSTMASTER, WEBMASTER	ADMIN
<b>contactInfo 联系人信息(必填)</b>		
contactName	联系人的姓名(必填)	
contactPhone	联系人电话(必填)	
contactEmail	联系人 Email 地址(必填), 用于发送如到期提醒等一般性的通知(必填)	
contactAddr	联系地址(选填)	
contactPostalCode	联系邮政编码(选填)	
<b>orgInfo 机构信息(DV 选填, OV、EV 必填)</b>		
orgName	组织机构名称(必填)	
orgType	机构性质(必填选填): 国家机关事业单位   商业企业单位   私有组织   非营利性组织	orgType,orgldType,orgldCode 三个字段对于 szca 品牌是必填的, 如果可以尽量传过来; 否则后续线下提供资料后, 暂由 szca 后台人工补齐订单信息会比较麻烦了

orgUnit	证书使用部门(必填)	
orgAddr	机构注册地址(必填)	
orgCity	城市(必填)	
orgState	省份/直辖市(必填)	
orgCountry	国家编码(默认是 CN, 可选)	
orgIdType	组织机构的身份证件类型(必填选填), 从下面三个之中选一个: 组织机构代码证   营业执照   统一社会信用代码	
orgIdCode	组织机构的身份证件号码, 如统一社会信用代码号码 (选填)	
orgPhone	机构电话	
orgPostalCode	机构邮编	
<b>uploadFiles 上传文件 arraylist 格式 (可选)</b>		
content	BASE64 格式的文件内容	可选
docType	内容只能如下: operatorIdFile("经办人身份证正反两面"), applicationFile("申请表及授权书"), orgRegisterDocumentFile("营业执照"), agreementFile("用户协议"), domainCert("域名证书"), confirmLetter("信息确认函"), --非 certum 的 revokeLetter("证书吊销确认函"), --非 certum 的	可选
fileName	文件名称	可选
pagerType	正面, 反面, 双面	可选

JSON 请求样例:

```
{
  "partnerId": "p10030",
  "timestamp": "1627742811557",
  "signData":
    "0c4f5a5e56d75c099706182edc80d1347d85968378f7e2faf641619c0a3d8d63",
```

```
"reqData": {  
  
  "orderId": "123",  
  
  "productCode": "DigiCertOV",  
  
  "csr": "-----BEGIN CERTIFICATE REQUEST...END CERTIFICATE REQUEST-----",  
  
  "sanApprover": {  
  
    "approveMethod": "EMAIL",  
  
    "approverEmailPrefix": "ADMIN"  
  
  },  
  
  "contactInfo": {  
  
    "contactName": "liangyh",  
  
    "contactAddr": "联系地址",  
  
    "contactPhone": "15678987654",  
  
    "contactPostalCode": "510000",  
  
    "contactEmail": "liangyh@szca.com"  
  
  },  
  
  "orgInfo": {  
  
    "orgUnit": "IT 部门",  
  
    "orgCity": "机构城市 gz",  
  
    "orgIdCode": "yyzz100",  
  
    "orgCountry": "CN",  
  
    "orgType": "商业企业单位",  
  
    "orgAddr": "机构注册地址 gdgz",
```



```
"orgIdType": "营业执照",  
  
"orgName": "企业名称",  
  
"orgState": "广东"  
  
},  
  
"domainName": "dcov7308.jdjj.xin"  
  
}  
  
}
```

#### 4.2.5. 返回结果

返回参数名称	说明	样例
code	调用接口的错误代码，0 为成功	0
msg	调用接口的错误信息	
data	若 code!=0,则此项为空；如 code=0, 则返回订单号	21073122001

JSON 返回样例：

```
{"data":"21073122001","code":0,"msg":"成功"}
```

### 4.3. 上传申请文件接口

#### 4.3.1. 功能描述

如果在调用申请证书接口时上传的文件不符合要求，可以通过此接口重新上传相关的文件。DV 型证书不能调用此接口。若合作伙伴与我方约定由深圳 CA 负责收集上述文件，则不用调用此接口。

### 4.3.2.环境地址

接口相对地址： /api/uploadFile.html

传入参数格式： POST Data。

### 4.3.3.请求参数表

参数	必填	类型	参数说明	样例
partnerId	是	String	合作伙伴身份 ID	P001
timestamp	是	String	时间戳 (中国时区)	1633918434258
signData	是	String	签名数据	d219a3dd8...7fea9d7fcaca 3
reqData	是	ReqData 对象	JSON 格式的请求数据包	

### 4.3.4.reqData 数据包结构

参数	参数说明	样例
orderId	订单号	20033015001
<b>uploadFiles 上传文件 arraylist 格式</b>		
content	BASE64 格式的文件内容	OV/EV 必填
docType	内容只能如下： operatorIdFile("经办人身份证正反两面"), applicationFile("申请表及授权书"), orgRegisterDocumentFile("营业执照"), agreementFile("用户协议"), domainCert("域名证书"), confirmLetter("信息确认函"), --非 certum 的 revokeLetter("证书吊销确认函"), --非 certum 的	OV/EV 必填
fileName	文件名称	可选
pagerType	正面, 反面, 双面	可选

JSON 请求样例:

```
{
  "partnerId": "p10030",
  "timestamp": "1627743494904",
  "signData": "b7ebc076789d66de5e54bc8ad005e5adca1415d14f23503041ff7071f0e96bc9",
  "reqData": {
    "uploadFiles": [
      {
        "content": "0M8R4KGxGu...==",
        "docType": "confirmLetter",
        "fileName": "企业名称-GeoTrust 企业型 SSL 证书-确认函模板.doc"
      }
    ],
    "orderId": "21073011002"
  }
}
```

#### 4.3.5.返回结果

返回参数名称	说明	样例
code	调用接口的错误代码, 0 为成功	0
msg	调用接口的错误信息	

JSON 返回样例:

```
{"code":0,"msg":"成功"}
```

## 4.4. 查询订单接口

### 4.4.1. 功能描述

此接口供合作伙伴根据需要，输入相关的参数，查询订单状态或者订单数据。

### 4.4.2. 接口地址

接口地址： /api/queryOrder.html

传入参数格式：POST Data。

### 4.4.3. 请求参数

参数	必填	类型	参数说明	样例
partnerId	是	String	合作伙伴身份 ID	P001
timestamp	是	String	时间戳 (中国时区, 毫秒), 如 java 用 System.currentTimeMillis()	1630055959224
signData	是	String	签名数据	d219a3dd8...7fea9d7fcaca3
reqData	是	ReqData 对象	JSON 格式的请求数据包	

### 4.4.4. reqData 数据包结构

参数	参数说明	样例
searchType	订单查询搜索条件类型: ORDER_ID:根据订单 ID 查询 DOMAIN_NAME:根据订单域名查询	ORDER_ID
searchCon	订单查询搜索条件值, 与 searchType 组合为有效的订单的查询条件	20033015001

fuzzyMatch	是否启用模糊匹配(默认 false): true   false 若启用, 则当: searchType=ORDER_ID searchCon=2007 时, 可以搜索所有以 2007 开头的订单号	false
------------	---	-------

JSON 请求样例:

```
{
  "partnerId": "p10030",
  "timestamp": "1627743938306",
  "signData": "088c7b63657d12a1e30b684f7c83dcb04dfb6d2f92966839ac3264327958a3c5",
  "reqData": {
    "searchType": "ORDER_ID",
    "searchCon": "2007",
    "fuzzyMatch": true
  }
}
```

#### 4.4.5.返回结果

返回参数名称	说明	样例
code	调用接口的错误代码, 0 为成功	0
msg	调用接口的错误信息	
orders	订单信息, 若 errCode!=0,则此项为空	

JSON 返回样例:

```
{
  "data": {
```

```
"orderCount": 10,

"orders": [

  {

    "verifyType": "DV",

    "modTime": "null",

    "productCode": "CertumDV",

    "periodDays": 365,

    "contactInfo": {

      "contactName": "liangyh",

      "contactAddr": "联系地址",

      "contactPhone": "15678987654",

      "contactPostalCode": "510000",

      "contactEmail": "liangyh@szca.com"

    },

    "approvers": [

      {

        "approveMethod": "DNS",

        "fqdn": "schedule070201.catest.top",

        "key": "_dnsauth.schedule070201.catest.top",

        "code": "xxx",

        "expireDate": "2021-10-09 10:11:12",

      }

    ]

  }

]
```

```
],  
  
"orgInfo": {},  
  
"domainName": "schedule070201.catest.top",  
  
"certificates": [  
  
  {  
  
    "notAfter": "2021-07-02 04:07:46.0",  
  
    "crt": "-----BEGIN CERTIFICATE-----...-----END CERTIFICATE-----",  
  
    "chainCrt": "-----BEGIN CERTIFICATE----- ...",  
  
    "notBefore": "2020-07-02 04:07:46.0",  
  
    "dn": "CN=schedule070201.catest.top",  
  
    "sn": "3F9217B5184C622921C76B3CC712E9B3",  
  
    "status": "REVOKED"  
  
  }  
  
],  
  
"periodStart": "2020-07-02 04:07:46.0",  
  
"providerProductId": "601",  
  
"backendStatus": "ENROLLED",  
  
"crtTime": "2020-07-02 09:17:58.622",  
  
"payment": {  
  
  "payStatus": "已支付",  
  
  "priceDiscounted": "0.0",  
  
  "payMethod": "线下支付_代理商",
```

```
    "pricePayment": "480.0",

    "priceTotal": "480.0"

  },

  "sanList": "schedule070201.catest.top",

  "orderId": "20070209001",

  "status": "已取消"

},

{

  "verifyType": "DV",

  "modTime": "null",

  "productCode": "CertumDV",

  "periodDays": 365,

  "contactInfo": {

    "contactName": "liangyh",

    "contactAddr": "联系地址",

    "contactPhone": "15678987654",

    "contactPostalCode": "510000",

    "contactEmail": "liangyh@szca.com"

  },

  "approvers": [

    {

      "approveMethod": "邮箱验证",
```



```
        "fqdn": "schedule070202.catest.top"
      }
    ],
    "orgInfo": {},
    "domainName": "schedule070202.catest.top",
    "periodStart": "null",
    "providerProductId": "601",
    "crtTime": "2020-07-02 10:44:50.216",
    "payment": {
      "payStatus": "已支付",
      "priceDiscounted": "0.0",
      "payMethod": "线下支付_代理商",
      "pricePayment": "480.0",
      "priceTotal": "480.0"
    },
    "sanList": "schedule070202.catest.top",
    "orderId": "20070210001",
    "status": "已取消"
  },
  ...
]
},
```

```
"code": 0,  
"msg": "成功"  
}
```

## 4.5. 取消订单接口

### 4.5.1. 功能描述

根据订单编号取消订单。尚未签发证书的订单，可以直接取消订单，已经签发证书的订单，则可以在证书签发后 30 天内使用吊销证书接口（同时会取消订单）。证书签发 30 天后不能取消订单，还是可以调用撤销证书接口来实现撤销 SSL 证书，但是不作退款的。注意，本接口只是接收取消订单的请求，实际的取消订单及撤销证书的操作是否成功，以后续推送信息为准。

### 4.5.2. 接口地址

接口地址： /api/cancelOrder.html

传入参数格式：POST Data。

### 4.5.3. 请求参数

参数	必填	类型	参数说明	样例
partnerId	是	String	合作伙伴身份 ID	P001
timestamp	是	String	时间戳 (中国时区)	1633918434258
signData	是	String	签名数据	d219a3dd8...7fea9d7fcaca3
reqData	是	ReqData 对象	JSON 格式的请求数据包	

#### 4.5.4.reqData 数据包结构

参数	参数说明	样例
orderId	订单号	21072719003

JSON 请求样例:

```
{
  "partnerId": "p10030",
  "timestamp": "1627744751133",
  "signData": "e9b9790fca288a5312bb92325e346e95ec2c794276a5e6392510dd7053501a64",
  "reqData": {
    "orderId": "21072719003"
  }
}
```

#### 4.5.5.返回结果

返回参数名称	说明	样例
code	调用接口的错误代码, 0 为取消申请操作成功	0
msg	调用接口的错误信息	

本接口只是接收取消订单的请求, 返回 code=0, 只意味着系统已经收到取消订单的请求。实际的取消订单及撤销证书的操作是否成功, 要以系统推送信息为准。

JSON 返回样例:

```
{"code":0,"msg":"成功"}
```

## 4.6. 重签证书接口

### 4.6.1. 功能描述

重新签发指定订单的证书。

### 4.6.2. 接口地址

接口地址： /api/reissueCert.html

传入参数格式：POST Data。

### 4.6.3. 请求参数

参数	必填	类型	参数说明	样例
partnerId	是	String	合作伙伴身份 ID	P001
timestamp	是	String	时间戳（中国时区）	1633918434258
signData	是	String	签名数据	d219a3dd8...7fea9d7fcaca3
reqData	是	ReqData 对象	JSON 格式的请求数据包	

### 4.6.4. reqData 数据包结构

参数	参数说明	样例
orderId	订单号（必填）	20033015001
certSn	证书序列号（必填）	ae19ad92d0d10e8745b84075b612fd1
csr	新证书的 csr（必填）	

JSON 请求样例：

```
{
```

```
"partnerId": "p10030",

"timestamp": "1627745106553",

"signData": "5202af58603f26b28c75cbf7e841cc3897008f9f36d347f7a4b919c80551f2df",

"reqData": {

  "csr": "-----BEGIN CERTIFICATE REQUEST...END CERTIFICATE REQUEST-----",

  "orderId": "21073013002",

  "certSn": "4E04C63F767245894F3F135AF05D7B5C"

}

}
```

#### 4.6.5.返回结果

返回参数名称	说明	样例
code	调用接口的错误代码，0 为成功	0
msg	调用接口的错误信息	

本接口只是接收重签证书的请求，返回 code=0，只意味着系统已经收到重签证书的请求。实际的重签证书的操作是否成功，要以系统推送信息为准。

JSON 返回样例：

```
{"code":0,"msg":"成功"}
```

## 4.7. 获取重签证书详情接口

### 4.7.1. 功能描述

通过本接口，能够获取对应订单号的最新重颁发详情。仅仅当证书品牌是非 certum 的产品才能够使用。

### 4.7.2. 接口地址

接口地址： /api/getReissueById.html

传入参数格式：POST Data。

### 4.7.3. 请求参数

参数	必填	类型	参数说明	样例
partnerId	是	String	合作伙伴身份 ID	P001
timestamp	是	String	时间戳（中国时区）	1633918434258
signData	是	String	签名数据	d219a3dd8...7fea9d7fcaca3
reqData	是	ReqData 对象	JSON 格式的请求数据包	

### 4.7.4. reqData 数据包结构

参数	参数说明	样例
orderId	订单号（必填）	20033015001

JSON 请求样例：

```
{
```

```
"partnerId": "p10030",

"timestamp": "1628142517707",

"signData": "9aac124646660ddf55d444088b0b69cbe4085044b469378d1392206c7a343e35",

"reqData": {

    "orderId": "21072719004"

}

}
```

#### 4.7.5.返回结果

返回参数名称	说明	样例
code	调用接口的错误代码，0 为成功	0
msg	调用接口的错误信息	

JSON 返回样例:

```
{
  "data": {
    "orderType": 0,
    "auths": [
      {
        "authKey": "dv7274.jdjj.xin",
        "authValue": "201907011329592a589zuu3ff7q14dbhy4ledauxnbwp6fre1hw99mkz52hpzwy1",
        "domain": "dv7274.jdjj.xin"
      }
    ],
    "pki": {
      "csrCommonName": "dv7274.jdjj.xin",
      "csr": "-----BEGIN CERTIFICATE REQUEST-----...-----END CERTIFICATE REQUEST-----",
      "certStatus": "issued",
      "certCommonName": "dv7274.jdjj.xin",
      "certKeyParameter": "2048",
      "cert": "-----BEGIN CERTIFICATE-----...-----END CERTIFICATE-----",
      "csrAlgorithm": "SHA256withRSA",
      "csrCountry": "CN",
    }
  }
}
```

```
"certCa": "-----BEGIN CERTIFICATE-----...-----END CERTIFICATE-----",
"id": 108014,
"certEndTime": 1631088000000,
"csrLocal": "gz",
"certAlgorithm": "SHA256withECDSA",
"csrOrganizationUnit": "it",
"csrKeyParameter": "2048",
"csrOrganization": "szca",
"certBeginTime": 1628150400000,
"certDnsNames": "",
"createTime": 1628158407000,
"csrState": "gd"
},
"validityPeriod": 0,
"dvAuthKey": "dv7274.jdjj.xin",
"dvAuthValue": "201907011329592a589zuu3ff7q14dbhy4ledauxnbwp6fre1hw99mkz52hpzwy1",
"partnerOrderId": "GDNnats17x_03",
"updateTime": 1628158414000,
"dvAuthMethod": "EMAIL",
"serverCount": 0,
"createTime": 1628158407000,
"allowGetSealSite": false,
"user": "lyh2021",
"encrypt": false,
"returnCount": 0,
"orderState": {
  "stateCode": "COMPLETED",
  "stateName": "订单完成"
}
},
"code": 0,
"msg": "成功"
}
```



## 4.8. 撤销证书接口

### 4.8.1. 功能描述

撤销指定的 SSL 证书，证书签发 30 天内，可以调用此接口来撤销证书，同时也会取消订单的；30 天后再调用此接口，只能撤销证书，不能取消订单，也就是不做退款的了。撤销证书是 DV 的时候也需要验证域名随机数，需要将新接收到的撤销证书的域名随机数更新到记录值里才能判断是否主动撤销，才能撤销成功。对于 OV/EV 的，不需要验证域名了，但是需要用域名相关的工作邮箱发送邮件到指定的邮箱，又或者提供撤销确认函，审核通过之后，才能撤销成功。

### 4.8.2. 接口地址

接口地址： /api/revokeCert.html

传入参数格式：POST Data。

### 4.8.3. 请求参数

参数	必填	类型	参数说明	样例
partnerId	是	String	合作伙伴身份 ID	P001
timestamp	是	String	时间戳 (中国时区)	1633918434258
signData	是	String	签名数据	d219a3dd8...7fea9d7fcaca3
reqData	是	ReqData 对象	JSON 格式的请求数据包	

### 4.8.4. reqData 数据包结构

参数	参数说明	样例
orderId	订单号 (必填)	20033015001
certSn	要撤销证书的序列号 (必填)	ae19ad92d0d10e87

		45b84075b612fd1
--	--	-----------------

JSON 请求样例:

```
{
  "partnerId": "p10030",
  "timestamp": "1627745404181",
  "signData": "b8dff34cd6770435f4ad1f2434548273761f823ca95e86aa7daeeaf54c1600f9",
  "reqData": {
    "orderId": "21072818003",
    "certSn": "6251512004544283640"
  }
}
```

#### 4.8.5.返回结果

返回参数名称	说明	样例
code	调用接口的错误代码, 0 为成功	0
msg	调用接口的错误信息	

本接口只是接收撤销证书的请求, 返回 code=0, 只意味着系统已经收到撤销证书的请求。实际的撤销操作是否成功, 要以系统推送信息为准。

JSON 返回样例:

```
{"code":0,"msg":"成功"}
```

## 4.9. 根据订单号获取订单状态

### 4.9.1. 功能描述

通过本接口，能够获取对应订单号的状态信息。非 szca 品牌可用

### 4.9.2. 接口地址

接口地址： /api/getOrderStateById.html

传入参数格式：POST Data。

### 4.9.3. 请求参数

参数	必填	类型	参数说明	样例
partnerId	是	String	合作伙伴身份 ID	P001
timestamp	是	String	时间戳（中国时区）	1633918434258
signData	是	String	签名数据	d219a3dd8...7fea9d7fcaca3
reqData	是	ReqData 对象	JSON 格式的请求数据包	

### 4.9.4. reqData 数据包结构

参数	参数说明	样例
orderId	订单号（必填）	20033015001

JSON 请求样例：

```
{  
  
  "partnerId": "p10030",
```

```
"timestamp": "1628142517707",

"signData": "9aac124646660ddf55d444088b0b69cbe4085044b469378d1392206c7a343e35",

"reqData": {

    "orderId": "21072719004"

}

}
```

#### 4.9.5.返回结果

返回参数名称	说明	样例
code	调用接口的错误代码, 0 为成功	0
msg	调用接口的错误信息	

JSON 返回样例:

```
{
  "data": {
    "orderType": 0,
    "validityPeriod": 0,
    "updateTime": 0,
    "serverCount": 0,
    "createTime": 0,
    "allowGetSealSite": false,
    "encrypt": false,
    "returnCount": 1,
    "detail": [
      {
        "partnerOrderId": "GOnm777pl",
        "stateCode": "WF_CANCEL",
        "stateName": "待确认取消"
      }
    ]
  },
  "code": 0,
  "msg": "成功"
```

```
}

```

## 4.10. 重发域名验证邮件接口

### 4.10.1. 功能描述

调用此接口，重新发送域名验证邮件。订单申请的时候，不需要调用此接口，系统会自动发送一次验证邮件的；只有当验证邮件太久都没有收到的情况下，可以调用此接口重新发送验证邮件

### 4.10.2. 接口地址

接口地址： /api/sendApproverEmail.html

传入参数格式：POST Data

### 4.10.3. 请求参数

参数	必填	类型	参数说明	样例
partnerId	是	String	合作伙伴身份 ID	P001
timestamp	是	String	时间戳 (中国时区)	1633918434258
signData	是	String	签名数据	d219a3dd8...7fea9d7fcaca3
reqData	是	ReqData 对象	JSON 格式的请求数据包	

### 4.10.4. reqData 数据包结构

参数	参数说明	样例
orderId	订单号 (必填)	20033015001

JSON 请求样例：

```
{

```

```
"partnerId": "p10030",

"timestamp": "1627745510664",

"signData": "75edff0952722cd822d7339f6dd38b4a40aa746eacddb1dffc3a9c5daa3fa73",

"reqData": {

    "orderId": "21032500001"

}

}
```

#### 4.10.5. 返回结果

返回参数名称	说明	样例
code	调用接口的错误代码，0 为成功	0
msg	调用接口的错误信息	

本接口接收重发邮件的请求，返回 code=0，只意味着系统已经收到重发邮件的请求，并成功触发发送邮件的操作。由于网络及用户的邮件服务器配置等原因，不能确保邮件正确送达。

JSON 返回样例：

```
{"code":0,"msg":"成功"}
```

### 4.11. 下载《信息确认函》模版

#### 4.11.1. 功能描述

对于 OV/EV 证书，用户申请后，需要下载《信息确认函》完善确认函里信息后，盖章，扫描并上传，通过此接口可获取到订单对应的信息确认函

### 4.11.2. 接口地址

接口地址： /api/downloadConfirmLetterTemp.html

传入参数格式：POST Data

### 4.11.3. 请求参数

参数	必填	类型	参数说明	样例
partnerId	是	String	合作伙伴身份 ID	P001
timestamp	是	String	时间戳 (中国时区)	1633918434258
signData	是	String	签名数据	d219a3dd8...7fea9d7fcac a3
reqData	是	ReqData 对象	JSON 格式的请求数据包	

### 4.11.4. reqData 数据包结构

参数	参数说明	样例
orderId	订单号 (必填)	20033015001

JSON 请求样例：

```
{  
  "partnerId": "p10030",  
  "timestamp": "1627745641423",  
  "signData": "4d6272096ffa019566fc144311c32b7ac1f2f4a2dff2584249ed5d2d1c5a902c",  
  "reqData": {  
    "orderId": "21072915002"  
  }  
}
```

```
}
```

#### 4.11.5. 返回结果

返回参数名称	说明	样例
code	调用接口的错误代码, 0 为成功	0
msg	调用接口的错误信息	
<b>data 返回结果信息:</b>		
content	文件 base64 string	
contentType	文件类型	application/msword
fileName	文件名称	GeoTrust 企业型 SSL 证书-确认函模板.doc
fileType	文件后缀	doc
orderId	订单号	21072909001

JSON 返回样例:

```
{
  "data": {
    "content": "0M8R4KGxGuE...AA=",
    "contentType": "application/msword",
    "fileName": "企业名称-GeoTrust 企业型 SSL 证书-确认函模板.doc",
    "fileType": "doc",
    "orderId": "21072915002"
  },
  "code": 0,
  "msg": "成功"
}
```



## 4.12. 获取域名管理员邮箱接口

### 4.12.1. 功能描述

根据域名返回合法的域名管理员邮箱。

### 4.12.2. 接口地址

接口地址： /api/getApproverEmails.html

传入参数格式：POST Data。

### 4.12.3. 请求参数

参数	必填	类型	参数说明	样例
partnerId	是	String	合作伙伴身份 ID	P001
timestamp	是	String	时间戳 (中国时区)	1633918434258
signData	是	String	签名数据	d219a3dd8...7fea9d7fcaca3
reqData	是	ReqData 对象	JSON 格式的请求数据包	

### 4.12.4. reqData 数据包结构

参数表：

参数	必填	类型	参数说明	样例
fqdn	是	String	要查询的域名	www.szca.com

JSON 请求样例：

```
{  
  
  "partnerId": "p10030",  
}
```

```
"timestamp": "1627745756366",  
  
"signData": "4cb1705c77cc459d33c0bf4225eea0341509f5c4909813fb224f07879cd94bea",  
  
"reqData": {  
  
    "fqdn": "d.win.jdjj.xin"  
  
}  
  
}
```

#### 4.12.5. 返回结果

返回参数名称	说明	样例
code	调用接口的错误代码, 0 为成功	0
msg	调用接口的错误信息	
emailList	邮箱列表	

JSON 返回样例:

```
{  
  
    "data": [  
  
        "admin@jdjj.xin",  
  
        "administrator@jdjj.xin",  
  
        "hostmaster@jdjj.xin",  
  
        "webmaster@jdjj.xin",  
  
        "postmaster@jdjj.xin",  
  
        "admin@d.win.jdjj.xin",  
  
        "administrator@d.win.jdjj.xin",  
  
        "hostmaster@d.win.jdjj.xin",  
  
    ]  
  
}
```

```
"webmaster@d.win.jdjj.xin",  
  
"postmaster@d.win.jdjj.xin"  
  
],  
  
"code": 0,  
  
"msg": "查询成功"  
  
}
```

## 5. 推送订单状态

### 5.1. 功能描述

此功能是需要合作伙伴的系统开放一个接口给深圳 CA 的系统调用。如果订单状态发生变化，深圳 CA 将订单信息推送到 notifyUrl 地址。使用 POST Data 模式。

### 5.2. 推送内容

参数	必填	类型	参数说明	样例
partnerId	是	String	合作伙伴身份 ID	P001
timestamp	是	String	时间戳，中国时区。	1633918434258
signData	是	String	数据包签名，算法为 sha256(partnerId + timestamp + accessKeySecret + reqData)	d219a3dd8...7fea9d7fca3
reqData	是	ReqData 对象	JSON 格式的请求数据包	

reqData 的数据包格式：

```
"reqData": {
```

```
"orderCount": 2,  
  
"orders": [{订单信息 1},{订单信息 2}...]  
  
}
```

注：订单信息的 JSON 格式，请参见“6.6.1 订单 JSON 格式”。如果接收成功，请返回 SUCCESS 字符串，即 `response.write("SUCCESS")`。若返回的内容不包含“SUCCESS”字符，深圳 CA 会在 24 小时内，多次推送，24 小时后不再主动推送，用户可以自行调用订单查询接口获得订单的信息。

## 6. 数据格式与编码表

### 6.1. 订单 JSON 格式

订单信息的 JSON 格式的说明如下：

```
{  
  
  "partnerId": "p10030",  
  
  "timestamp": "1629362450081",  
  
  "signData": "87511d9206276b3441a2f9ebf90926b2db77e54b4579ae9112e6598cae60690e",  
  
  "reqData": {  
  
    "orderCount": 1,  
  
    "orders": [  
  
      {  
  
        "verifyType": "DV",  
  
        "modTime": "null",  
  
        "csr": "-----BEGIN CERTIFICATE REQUEST-----\n...\n-----END CERTIFICATE"
```

```
REQUEST-----\n",

    "productCode": "GeoTrustDV",

    "remark": "",

    "periodDays": 365,

    "contactInfo": {

        "contactName": "liangyh",

        "contactAddr": "联系地址",

        "contactPhone": "15678987654",

        "contactPostalCode": "510000",

        "contactEmail": "liangyh@szca.com"

    },

    "verifications": [

        {

            "FQDN": "dv8191.jdjj.xin",

            "approveMethod": "DNS",

            "key": "_dnsauth.dv8191.jdjj.xin",

            "code":

"202108181536184zm2tqqjeg4x0lcu79kcupxmir6wkaaaeha0mctf8wae52gfmy",

            "verifyCodeExpire": "2021-08-26 15:36:18.0"

        }

    ],

    "orgInfo": {
```

```
"orgUnit": "IT 部门",

"orgIdCode": "yyzz100",

"orgType": "商业企业单位",

"orgAddr": "机构注册地址 gdgz",

"orgName": "企业名称"

},

"domainName": "dv8191.jdjj.xin",

"certificates": [

{

  "notAfter": "2021-09-17 16:00:00.0",

  "crt": "-----BEGIN CERTIFICATE-----\n...\n-----END CERTIFICATE-----",

  "chainCrt": "-----BEGIN CERTIFICATE----- ...",

  "notBefore": "2021-08-19 16:00:00.0",

  "sn": "8902260687180057076",

  "status": "VALID"

}

],

"periodStart": "2021-08-19 16:00:00.0",

"backendStatus": "ENROLLED",

"crtTime": "2021-08-19 15:36:18.321",

"payment": {

  "payStatus": "已支付",
```

```
"priceDiscounted": "0.0",

"payMethod": "线下支付_代理商",

"pricePayment": "400.0",

"priceTotal": "400.0"

},

"orderId": "21081900004",

"status": "完成",

"applyType": "NEW"

}

]

}

}
```

## 6.2. 产品编码表

产品编码列表会不定期更新，可以从 api 中获得最新的产品列表：</api/getProductList.html>

下表是目前的产品列表：

证书品牌	产品编码 1	产品编码 2	类型	多域名	通配符	产品描述
Certum	CertumDV		DV	否	否	深安单域名证书

	CertumDVSANS		DV	是	是	深安多域名证书
	CertumDVWildcard		DV	否	是	深安单域名通配符证书
	CertumEV		EV	否	否	深誉单域名证书
	CertumEVSANS		EV	是	否	深誉多域名证书
	CertumOV		OV	否	否	深信单域名证书
	CertumOVSANS		OV	是	是	深信多域名证书
	CertumOVWildcard		OV	否	是	深信单域名通配符证书
DigiCert	DigiCertEV	DCOV01	EV	否	否	DigiCert 增强型 (EV) SSL 证书
	DigiCertEVSANS	DCOV02	EV	是	否	DigiCert 增强型多域名 (EV) SSL 证书
	DigiCertOV	DCOV01	OV	否	否	DigiCert 企业型 (OV) SSL 证书
	DigiCertOVSANS	DCOV02	OV	是	是	DigiCert 企业型多域名 (OV) SSL 证书
	DigiCertOVWildcard	DCOV03	OV	否	是	DigiCert 企业型通配符 (OV) SSL 证书
Geotrust	GeoTrustDV	GEODV03	DV	否	否	GeoTrust 域名型 SSL 证书
	GeoTrustDVSANS	GEODV03	DV	是	是	GeoTrust 域名型多域名 SSL 证书



	GeoTrustDVWildcard	GEODV04	DV	否	是	GeoTrust 域名型通配符 SSL 证书
	GeoTrustEV	GEOEV01	EV	否	否	GeoTrust 增强型 SSL 证 书
	GeoTrustEVSANs	GEOEV01	EV	是	否	GeoTrust 增强型多域名 SSL 证书
	GeoTrustOV	GEOOV01	OV	否	否	GeoTrust 企业型 SSL 证 书
	GeoTrustOVSANs	GEOOV01	OV	是	是	GeoTrust 企业型多域名 SSL 证书
	GeoTrustOVWildcard	GEOOV02	OV	否	是	GeoTrust 企业型通配符 SSL 证书
SecureSite 原 Symantec	SecureSiteEVPro	SYMCEV02	EV	否	否	SecureSite 增强型 SSL 证 书专业版
	SecureSiteEVSANs	SYMCEV01	EV	否	否	SecureSite 增强型多域名 SSL 证书
	SecureSiteEVSANsPro	SYMCEV02	EV	否	否	SecureSite 增强型多域名 SSL 证书专业版
	SecureSiteNEV	SYMCEV01	EV	否	否	SecureSite 增强型 SSL 证 书

	SecureSiteOV	SYMC0V01	OV	否	否	SecureSite 企业型 SSL 证书
	SecureSiteOVPro	SYMC0V02	OV	否	否	SecureSite 企业型 SSL 证书专业版
	SecureSiteOVSANs	SYMC0V01	OV	是	是	SecureSite 企业型多域名 SSL 证书
	SecureSiteOVSANsPro	SYMC0V02	OV	是	是	SecureSite 企业型多域名 SSL 证书专业版
	SecureSiteOVWildcard	SYMC0V03	OV	否	是	SecureSite 企业型通配符 SSL 证书
	SecureSiteOVWildcard Pro	SYMC0V04	OV	否	是	SecureSite 企业型通配符 SSL 证书专业版
GlobalSign	GlobalSignDV		DV	否	否	域名型 SSL 证书
	GlobalSignDVWildcard		DV	否	是	域名型通配符 SSL 证书
	GlobalSignOV		OV	是	是	企业型一个或多个域名
	GlobalSignOVWildcard		OV	否	是	企业型通配符
	GlobalSignEV		EV	是	否	增强型一个或多个域名
CFCA	CFCAOV		OV	是	是	企业型一个或多个域名
	CFCAOVWildcard		OV	否	是	企业型通配符
	CFCAEV		EV	是	否	增强型一个或多个域名

## 6.3. 订单状态表

订单状态是指订单目前所处的处理状态。订单状态表可能会偶尔不定期更新，可以从 api 中获得最新的订单状态表：

/api/getOrderStatus.html

下表是目前的订单状态表：

订单状态编码	订单状态描述
新建	尚未支付的订单
验证中	已经支付，处于域名所有权验证中的订单
取消中	用户已经申请取消，但是系统尚未正式取消的订单
已取消	用户申请且系统正式确认取消的订单
撤销中	用户已经申请撤销订单的证书，但是系统尚未成功撤销证书的订单
已撤销	已经撤销证书的订单
完成	已经正式签发证书的订单
归档	签发的证书已经过期的订单

## 6.4. 订单后台状态表

订单后台状态就批订单进入 CA 处理阶段之后，在 CA 系统中所处的状态。可以从 api 中获得最新的订单后台状态表：

/api/getOrderBackendStatus.html

状态编码	状态说明
------	------

AWAITING	待验证
VERIFICATION	核验中（一般只有 OV、EV 才有这个状态）
REJECTED	拒绝（核验不通过、用户取消订单等）
ACCEPTED	通过核验（一般只有 OV、EV 才有这个状态）
ENROLLED	已经签发

## 6.5. 文档修改记录

2021.8.4	V2.0.2	1. 增加续费接口 renewCert; 2..修改免费取消期限为 30 天内; 3.查询接口补上 验证值, 验证值过期时间, 等内容
2021.8.6	V2.0.3	1. 增加 获取订单重颁发详情 getReissueById 接口 2. 更正产品列表错误描述
2021.8.8	V2.0.4	1.修改字段 partnerProductId->productCode; 2.增加章节 3.3.签名规范; 3.增加章节 4.9. 根据订单号获取订单状态
2021.8.10	V2.05	1.修改参数描述 reqData String ->reqData 对象; 2.删除订单查询返回的字段 providerProductId
2021.11.4	V2.06	1.产品编码表, 新增一列 “产品编码 2”, 如果这个有内容, 后续将用 “产品编码 2” 替换 “产品编码 1”
2021.11.20	V2.07	1. 废弃 V2.06 新增的产品代码 2 的改动, 修改为自动 mapping, 合作伙伴无感, 不需要作任何修改

		2. 通配符域名，不再支持 FILE 文件验证方式
2021.12.3	V2.0.8	1. 增加 Globalsign 产品代码 2. 增加 CFCA 产品代码