|  |  |
| --- | --- |
| 产品名称Product name | 密级Confidentiality level |
|  |  |
| 产品版本Product version | Total XX pages共XX页 |
| 1.0 |

功能清单接口说明书

|  |  |  |  |
| --- | --- | --- | --- |
| Prepared by  拟制 |  | Date  日期 |  |
| Reviewed by  评审人 |  | Date  日期 |  |
| Approved by  批准 |  | Date  日期 |  |
| Authorized by  签发 |  | Date  日期 |  |

杭州瑞成信息技术有限公司

All rights reserved

版权所有侵权必究

Revision record 修订记录

| Date  日期 | Revision Version  修订版本 | CR ID/ Defect ID CR号 | Section Number 修改章节 | Change  Description  修改描述 | Author  作者 |
| --- | --- | --- | --- | --- | --- |
| 2022.2.22 | 1.0 |  |  | 新建 |  |
| 2022.3.18 | 2.0 |  | 1. 新增2.4.4章节 2. 新增4服务开发规范-V2.0 3. 新增5大屏统计接口 | 修改 | 张情情 |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

目录

[1. 名词描述 10](#_Toc98180677)

[2. 管理界面原型及服务调用 10](#_Toc98180678)

[2.1. 系统管理 10](#_Toc98180679)

[2.2. 应用管理 10](#_Toc98180680)

[2.2.1. 菜单路径 10](#_Toc98180681)

[2.2.2. 界面及服务标注 10](#_Toc98180682)

[2.2.3. 服务接口 10](#_Toc98180683)

[2.3. 加密机管理 10](#_Toc98180684)

[2.3.1. 菜单路径 10](#_Toc98180685)

[2.3.2. 界面及服务标注 11](#_Toc98180686)

[2.3.3. 服务接口 11](#_Toc98180687)

[2.4. 密钥管理 11](#_Toc98180688)

[2.4.1. 菜单路径 11](#_Toc98180689)

[2.4.2. 界面及服务标注 11](#_Toc98180690)

[2.4.3. 服务接口 11](#_Toc98180691)

[2.4.4. 密钥归档管理 12](#_Toc98180692)

[2.4.4.1界面及服务标注 12](#_Toc98180693)

[2.5. 日志管理 13](#_Toc98180694)

[2.5.1. 菜单路径 13](#_Toc98180695)

[2.5.2. 界面及服务标注 14](#_Toc98180696)

[2.5.3. 服务接口 14](#_Toc98180697)

[3. 服务开发规范-V1.0 14](#_Toc98180698)

[3.1. 获取密钥 15](#_Toc98180699)

[3.1.1. 接口描述 15](#_Toc98180700)

[3.1.2. 服务接口 15](#_Toc98180701)

[3.1.2.1接口定义 15](#_Toc98180702)

[3.1.2.2请求参数 15](#_Toc98180703)

[3.1.2.3响应参数： 15](#_Toc98180704)

[3.2. 数据加密接口 16](#_Toc98180705)

[3.2.1. 接口描述 16](#_Toc98180706)

[3.2.2. 接口业务流程图 16](#_Toc98180707)

[3.2.3. 服务接口 16](#_Toc98180708)

[3.2.3.1接口定义 16](#_Toc98180709)

[3.2.3.2请求参数 17](#_Toc98180710)

[3.2.3.3响应参数 17](#_Toc98180711)

[3.3. 数据解密 17](#_Toc98180712)

[3.3.1. 接口描述 17](#_Toc98180713)

[3.3.2. 接口流程图 17](#_Toc98180714)

[3.3.3. 服务接口 18](#_Toc98180715)

[3.3.3.1接口定义 18](#_Toc98180716)

[3.3.3.2请求参数 18](#_Toc98180717)

[3.3.3.3响应参数 18](#_Toc98180718)

[3.4. 批量数据加密 19](#_Toc98180719)

[3.4.1. 接口描述 19](#_Toc98180720)

[3.4.2. 服务接口 19](#_Toc98180721)

[3.4.2.1接口定义 19](#_Toc98180722)

[3.4.2.2请求参数 19](#_Toc98180723)

[3.4.2.3响应参数： 19](#_Toc98180724)

[3.5. 批量数据解密接口 20](#_Toc98180725)

[3.5.1. 接口描述 20](#_Toc98180726)

[3.5.2. 服务接口 20](#_Toc98180727)

[3.5.2.1接口定义 20](#_Toc98180728)

[3.5.2.2请求参数 20](#_Toc98180729)

[3.5.2.3响应参数 21](#_Toc98180730)

[3.6. 不可逆加密 21](#_Toc98180731)

[3.6.1. 接口描述 21](#_Toc98180732)

[3.6.2. 服务接口 21](#_Toc98180733)

[3.6.2.1接口定义 21](#_Toc98180734)

[3.6.2.2请求参数 22](#_Toc98180735)

[3.6.2.3响应参数 22](#_Toc98180736)

[3.7. 文件加密接口 22](#_Toc98180737)

[3.7.1. 接口描述 22](#_Toc98180738)

[3.7.2. 服务接口 22](#_Toc98180739)

[3.7.2.1接口定义 22](#_Toc98180740)

[3.7.2.2请求参数 23](#_Toc98180741)

[3.7.2.3响应参数： 23](#_Toc98180742)

[3.8. 文件解密接口 23](#_Toc98180743)

[3.8.1. 接口描述 23](#_Toc98180744)

[3.8.2. 服务接口 23](#_Toc98180745)

[3.8.2.1接口定义 23](#_Toc98180746)

[3.8.2.2请求参数 23](#_Toc98180747)

[3.8.2.3响应参数 24](#_Toc98180748)

[3.9. 数据签名接口 24](#_Toc98180749)

[3.9.1. 接口描述 24](#_Toc98180750)

[3.9.2. 服务接口 24](#_Toc98180751)

[3.9.2.1接口定义 24](#_Toc98180752)

[3.9.2.2请求参数 24](#_Toc98180753)

[3.9.2.3响应参数 25](#_Toc98180754)

[3.10. 数据验签接口 25](#_Toc98180755)

[3.10.1. 接口描述 25](#_Toc98180756)

[3.10.2. 服务接口 25](#_Toc98180757)

[3.10.2.1接口定义 25](#_Toc98180758)

[3.10.2.2请求参数 25](#_Toc98180759)

[3.10.2.3响应参数： 26](#_Toc98180760)

[3.11. 文件签名接口 26](#_Toc98180761)

[3.11.1. 接口描述 26](#_Toc98180762)

[3.11.2. 服务接口 26](#_Toc98180763)

[3.11.2.1接口定义 26](#_Toc98180764)

[3.11.2.2请求参数 26](#_Toc98180765)

[3.11.2.3响应参数 26](#_Toc98180766)

[3.12. 文件验签接口 27](#_Toc98180767)

[3.12.1. 接口描述 27](#_Toc98180768)

[3.12.2. 服务接口 27](#_Toc98180769)

[3.12.2.1接口定义 27](#_Toc98180770)

[3.12.2.2请求参数 27](#_Toc98180771)

[3.12.2.3响应参数 27](#_Toc98180772)

[4. 服务开发规范-V2.0 28](#_Toc98180773)

[4.1客户端模式授权码 28](#_Toc98180774)

[4.1.1接口描述 28](#_Toc98180775)

[4.1.2服务接口： 28](#_Toc98180776)

[4.1. 获取密钥 29](#_Toc98180777)

[4.1.1. 接口描述 29](#_Toc98180778)

[4.1.2. 服务接口 29](#_Toc98180779)

[4.1.2.1接口定义 29](#_Toc98180780)

[4.1.2.2请求参数 29](#_Toc98180781)

[4.1.2.3响应参数： 30](#_Toc98180782)

[4.2. 数据加密接口 30](#_Toc98180783)

[4.2.1. 接口描述 30](#_Toc98180784)

[4.2.2. 接口业务流程图 31](#_Toc98180785)

[4.2.3. 服务接口 31](#_Toc98180786)

[4.2.3.1请求参数 31](#_Toc98180787)

[4.2.3.2响应参数： 32](#_Toc98180788)

[4.3. 数据解密 33](#_Toc98180789)

[4.3.1. 接口描述 33](#_Toc98180790)

[4.3.2. 接口流程图 33](#_Toc98180791)

[4.3.3. 服务接口 33](#_Toc98180792)

[4.3.3.1接口定义 34](#_Toc98180793)

[4.3.3.2请求参数 34](#_Toc98180794)

[4.3.3.3响应参数 35](#_Toc98180795)

[4.4. 批量数据加密 35](#_Toc98180796)

[4.4.1. 接口描述 35](#_Toc98180797)

[4.4.2. 服务接口 35](#_Toc98180798)

[4.4.2.1接口定义 35](#_Toc98180799)

[4.4.2.2请求参数 36](#_Toc98180800)

[4.4.2.3响应参数： 37](#_Toc98180801)

[4.5. 批量数据解密接口 37](#_Toc98180802)

[4.5.1. 接口描述 37](#_Toc98180803)

[4.5.2. 服务接口 37](#_Toc98180804)

[4.5.2.1接口定义 38](#_Toc98180805)

[4.5.2.2请求参数 38](#_Toc98180806)

[4.5.2.3响应参数 39](#_Toc98180807)

[4.6. 不可逆加密 39](#_Toc98180808)

[4.6.1. 接口描述 40](#_Toc98180809)

[4.6.2. 服务接口 40](#_Toc98180810)

[4.6.2.1接口定义 40](#_Toc98180811)

[4.6.2.2请求参数 40](#_Toc98180812)

[4.6.2.3响应参数 41](#_Toc98180813)

[4.7. 文件加密接口 41](#_Toc98180814)

[4.7.1. 接口描述 41](#_Toc98180815)

[4.7.2. 服务接口 42](#_Toc98180816)

[4.7.2.1接口定义 42](#_Toc98180817)

[4.7.2.2请求参数 42](#_Toc98180818)

[4.7.2.3响应参数： 43](#_Toc98180819)

[4.8. 文件解密接口 43](#_Toc98180820)

[4.8.1. 接口描述 43](#_Toc98180821)

[4.8.2. 服务接口 43](#_Toc98180822)

[4.8.2.1接口定义 44](#_Toc98180823)

[4.8.2.2请求参数 44](#_Toc98180824)

[4.8.2.3响应参数 44](#_Toc98180825)

[4.9. 数据签名接口 45](#_Toc98180826)

[4.9.1. 接口描述 45](#_Toc98180827)

[4.9.2. 服务接口 45](#_Toc98180828)

[4.9.2.1接口定义 45](#_Toc98180829)

[4.9.2.2请求参数 45](#_Toc98180830)

[4.9.2.3响应参数 46](#_Toc98180831)

[4.10. 数据验签接口 47](#_Toc98180832)

[4.10.1. 接口描述 47](#_Toc98180833)

[4.10.2. 服务接口 47](#_Toc98180834)

[4.10.2.1接口定义 47](#_Toc98180835)

[4.10.2.2请求参数 47](#_Toc98180836)

[4.10.2.3响应参数： 48](#_Toc98180837)

[4.11. 文件签名接口 48](#_Toc98180838)

[4.11.1. 接口描述 48](#_Toc98180839)

[4.11.2. 服务接口 49](#_Toc98180840)

[4.11.2.1接口定义 49](#_Toc98180841)

[4.11.2.3响应参数 49](#_Toc98180842)

[4.1.2.3响应参数： 50](#_Toc98180843)

[4.12. 文件验签接口 50](#_Toc98180844)

[4.12.1. 接口描述 50](#_Toc98180845)

[4.12.2. 服务接口 50](#_Toc98180846)

[4.12.2.1接口定义 50](#_Toc98180847)

[4.12.2.2请求参数 51](#_Toc98180848)

[4.12.2.3响应参数 52](#_Toc98180849)

[5. 大屏统计接口 52](#_Toc98180850)

[5.1. 客户端模式授权码 52](#_Toc98180851)

[5.1.1. 接口描述 52](#_Toc98180852)

[5.1.2. 服务接口： 52](#_Toc98180853)

[5.2. 统计接口描述 53](#_Toc98180854)

[5.2.1. 接口定义 53](#_Toc98180855)

[5.2.2. 请求参数 53](#_Toc98180856)

[5.2.3. 响应参数 54](#_Toc98180857)

# 名词描述

1. 三方应用：接入到加解密服务平台的应用。

# 管理界面原型及服务调用

## 系统管理

使用平台原有的用户管理、角色管控、菜单资源管控

## 应用管理

对三方应用的添加、编辑、启用/禁用、删除、设置加密机、设置主密钥。

### 菜单路径

应用管理🡪 三方应用

### 界面及服务标注

### 服务接口

使用的是统一用户的增删改查。

* 设置加密机接口：

/risen/cipher/app/updateEncryptor.do

* 设置主密钥接口：

/risen/cipher/app/updateAppIpkIndex.do

## 加密机管理

### 菜单路径

加密机管理

### 界面及服务标注

### 服务接口

使用低代码平台进行配置出的表单设计，视图设计。

涉及到的接口是低代码内部自适应的增删改查接口。

## 密钥管理

### 菜单路径

密钥管理

### 界面及服务标注

### 服务接口

* 密钥配置：使用低代码视图设计,

调用api接口：/risen/core/uams/mag/showCoreUamsApp.do

视图配置时配置的请求参数

{

"CMD":"LIST",

"TYPE":"jsn\_grid"

}

* 密钥策略管理：密钥策略新建、编辑。使用低代码配置，

新建接口和编辑接口：使用的是低代码表单视图配置后低代码自适应接口。

查询列表接口：/risen/cipher/app/strategy/showCipherAppStrategy

* 密钥查看：

/risen/cipher/app/strategy/appStrategy/keyView.do get请求

### 密钥归档管理

密钥管理->密钥归档管理->密钥归档、密钥恢复

密钥归档时，会进行查询服务日志，如果此密钥创建时间超过了配置的月份，且服务日志记录中配置的最近月份中没有使用此密钥进行加密过。则可进行归档处理。

#### 2.4.4.1界面及服务标注

##### 2.4.4.1.1密钥归档：

* 使用低代码视图设计,调用api接口：

/risen/cipher/dek/showCipherDekForHis.do

视图配置时配置的请求参数

{

"CMD":"LIST",

"TYPE":"jsn\_grid",

"strMap.CQ":"{\"OF\":[{\"colName\":\"cdekCreateTime\", \"OP\":\"DESC\"}]}"

}

* 一键归档

/risen/cipher/dek/his/insertDekHis.do

请求参数：

strList:CDF674291E7441578ABDC0BFD5416ABB,E2964A5DB5234DD398405DFDA73591D7

##### 2.4.4.1.1密钥恢复：

* 使用低代码视图设计,调用api接口：

/risen/cipher/dek/his/showCipherDekHisForReserve.do

视图配置时配置的请求参数

{

"CMD":"LIST",

"TYPE":"jsn\_grid",

"strMap.CQ":"{\"OF\":[{\"colName\":\"cdhisCreateTime\", \"OP\":\"DESC\"}],\"CF\":[{\"colName\":\"cdhisStatus\",\"OP\":\"EQ\",\"value\":\"1\"}]}"

}

* 一键恢复

/risen/cipher/dek/his/reverseDekHis.do

strList:CDF674291E7441578ABDC0BFD5416ABB

* 一键销毁

/risen/cipher/dek/his/updateCipherDekHisStatusByIdList.do

strList:2098AE93AA2C4A4596DF3375A72EF875,40C14AD501B540A7B922A3EE74D85269

status:0

## 日志管理

### 菜单路径

日志管理

### 界面及服务标注

### 服务接口

使用低代码视图设计,调用api接口：

* 登录日志：

/risen/core/mag/log/findByCustomQueryCoreLog.action

视图配置时默认请求参数：

{"strMap.CQ":"{\"OF\":[],\"CF\":[{\"GROUP\":[{\"colName\":\"crlogType\",\"OP\":\"LIKE\",\"value\":\"L\"}]}]}"}

* 操作日志：

/risen/core/mag/log/findByCustomQueryCoreLog.action

视图配置时默认请求参数：

{"strMap.CQ": "{\"OF\":[],\"CF\":[{\"GROUP\":[{\"colName\":\"crlogType\",\"OP\":\"LIKE\",\"value\":\"A\"}, {\"OP\": \"OR\" },{ \"colName\": \"crlogType\", \"OP\": \"LIKE\", \"value\": \"D\"},{ \"OP\": \"OR\" },{ \"colName\": \"crlogType\", \"OP\": \"LIKE\", \"value\": \"U\" }]}]}"}

* 服务日志：

/risen/cipher/service/log/showCipherServiceLog.do

视图配置时默认请求参数：

{"CMD":"LIST","TYPE":"jsn\_grid","strMap.CQ":"{\"OF\":[],\"CF\":[]}","strMap.ECQ":"{\"OF\":[],\"CF\":[]}"}

# 服务开发规范-V1.0

## 获取密钥

### 接口描述

为了满足加密粒度的要求，应用系统需要使用不同的密钥来对不同的数据进行加密，因此应用系统在需要新密钥对新数据进行加密时，需要调用该接口来获取新密钥。

### 服务接口

#### 3.1.2.1接口定义

接口名称：/api/getDataKey通信协议：http 请求方式：post

#### 3.1.2.2请求参数

{

    "appCode": "8C09761853134B6DA11B906162253ECC",//应用唯一标识

    "size": "2"//本次生成的密钥数量

}

#### 3.1.2.3响应参数：

{

    "message": "成功",//当失败时返回原因

    "data": [

        {

            "dataKey": "517f0aa50dd0536d5f3a32774181a046"//数据密钥

        },

        {

            "dataKey": "c92473c7760a85feba00e001ddbdcba7"//数据密钥

        }

    ],

    "code": 200//根据 code 类型判断本次操作是否成功

}

## 数据加密接口

### 接口描述

对于一些重要的用户信息、组织机构信息、日志信息等可以使用该接口对相关的数据进行加密。

### 接口业务流程图



### 服务接口

#### 3.2.3.1接口定义

接口名称： /api/encryptData通信协议：http 请求方式：post

#### 3.2.3.2请求参数

{

"appCode": "DFB41EB431524D84A34F64546FF341DE", //应用唯一标识

"dataKey": "47ac46530bac8de869896c8db7de7c69", //数据密钥

"data": "13212344321" //明文数据

}

#### 3.2.3.3响应参数

{

"message": "成功", //当失败时返回原因

"data": "b6c4b63f8d3b74813d30830ca7accb3fdbe57d625ddd1615a733d6d64544eaa3", //密文数据

"code": 200 //根据 code 类型判断本次操作是否成功

}

## 数据解密

### 接口描述

用于数据的解密操作，在应用系统的数据进行页面展示时，调用此接口将数据解密后展示明文。

### 接口流程图



### 服务接口

#### 3.3.3.1接口定义

接口名称： /api/decryptData 通信协议：http 请求方式：post

#### 3.3.3.2请求参数

{

"appCode": "01bc857f5c1245708b1247236d9869fe", //应用唯一标识

"dataKey": "5fb1192655db0f620dca5cfdb2da3ed2",//数据密钥

"data": "b6c4b63f8d3b74813d30830ca7accb3fdbe57d625ddd1615a733d6d64544eaa3" //密文数据

}

#### 3.3.3.3响应参数

{

"message": "成功",//当失败时返回原因

"data": "123456789123456789",//明文数据

"code": 200//根据 code 类型判断本次操作是否成功

}

## 批量数据加密

### 接口描述

提供给三方应用，进行批量对数据进行加密操作。

### 服务接口

#### 3.4.2.1接口定义

接口名称： /api/batchEncrypt 通信协议：http 请求方式：post

#### 3.4.2.2请求参数

{

    "appCode": "01BC857F5C1245708B1247236D9869FE",//应用唯一标识

    "batchData": [

        {

            "dataKey": "9ac0979a0242c6a0c0ed1ade0e653421",//数据密钥

            "IDCard1": "123456789123456789"//明文数据

        },

        {

            "dataKey": "dfab61e346c9ba682d785cf60366b8d3",//数据密钥

            "IDCard2": "987654321987654321"//明文数据

        }

    ]

}

#### 3.4.2.3响应参数：

{

    "message": "成功",//当失败时返回原因

    "data": [

        {

            "IDCard1": "3a9574cf82aa215bc5d4e65ab1960e0e7bc59419f9951779d5e94231089d7b6f"//密文数据

        },

        {

            "IDCard2": "ec05e7341effe2aca141b44081b13e4914e63acf48f87e4fe2929872e03c292b"//密文数据

        }

    ],

    "code": 200//根据 code 类型判断本次操作是否成功

}

## 批量数据解密接口

### 接口描述

对大量密文数据进行批量解密，提高三方应用系统的解密效率。

### 服务接口

#### 3.5.2.1接口定义

接口名称： /api/batchDecrypt 通信协议：http 请求方式：post

#### 3.5.2.2请求参数

{

    "appCode": "01BC857F5C1245708B1247236D9869FE",//应用唯一标识

    "batchData": [

        {

            "dataKey": "9ac0979a0242c6a0c0ed1ade0e653421",//数据密钥

            "IDCard1": "3a9574cf82aa215bc5d4e65ab1960e0e7bc59419f9951779d5e94231089d7b6f" //密文数据

        },

        {

            "dataKey": "dfab61e346c9ba682d785cf60366b8d3",//数据密钥

            "IDCard2": "ec05e7341effe2aca141b44081b13e4914e63acf48f87e4fe2929872e03c292b"//密文数据

        }

    ]

}

#### 3.5.2.3响应参数

{

    "message": "成功",//当失败时返回原因

    "data": [

        {

            "IDCard1": "123456789123456789"//明文数据

        },

        {

            "IDCard2": "987654321987654321"//明文数据

        }

    ],

    "code": 200//根据 code 类型判断本次操作是否成功

}

## 不可逆加密

### 接口描述

不可逆加密是指数据哈希。数据哈希使用的是sm3算法，具备有不可逆性，因此使用该算法进行加密的数据是那些不需要再进行解密的数据。该方法不需要使用数据密钥。常用此接口加密的数据例如用户的登录密码。

### 服务接口

#### 3.6.2.1接口定义

接口名称： /api/hashData 通信协议：http 请求方式：post

#### 3.6.2.2请求参数

{

    "appCode": "01bc857f5c1245708b1247236d9869fe", //应用唯一标识

"data":"123456789123456789" //明文数据

}

#### 3.6.2.3响应参数

{ "message": "成功",//当失败时返回原因

"data":"f72a78ff84bd13e6a011f5427e53fa278989e0afe9d5f2dd811ffa 58d6c290b2", //密文数据

"code": 200//根据 code 类型判断本次操作是否成功

}

## 文件加密接口

### 接口描述

为了保证存储安全，服务器内部存放的文件应该都是密文，因此在文件的收发文过程产生的文件，都应该进行加密，使文件以密文的形式在服务器上存储。将文件BASE64编码后调用接口服务进行加密操作,文件超过4MB建议使用本地API方法。

### 服务接口

#### 3.7.2.1接口定义

接口名称： /api/encryptFile 通信协议：http 请求方式：post

#### 3.7.2.2请求参数

{

    "appCode": "01BC857F5C1245708B1247236D9869FE",//应用唯一标识

    "dataKey": "b04d4bd00a04832290f9d17f07379bfe",//数据密钥

    "fileContent": "BASE64 编码文件内容"//明文 BASE64 编码

}

#### 3.7.2.3响应参数：

{

    "message": "成功",//当失败时返回原因

    "data": " BASE64 编码文件内容",//密文 BASE64 编码

    "code": 200//根据 code 类型判断本次操作是否成功

}

## 文件解密接口

### 接口描述

三方应用在服务器上存储的是加密后的文件，当三方应用需要查看文件内容时，必须调用文件的解密接口将文件解密。

### 服务接口

#### 3.8.2.1接口定义

接口名称： /api/decryptFile 通信协议：http 请求方式：post

#### 3.8.2.2请求参数

{

    "appCode": "01BC857F5C1245708B1247236D9869FE",//应用唯一标识

    "dataKey": "b04d4bd00a04832290f9d17f07379bfe",//数据密钥

    "fileContent": "BASE64 文件编码内容"//密文 BASE64 编码

}

#### 3.8.2.3响应参数

{

    "message": "成功",//当失败时返回原因

    "data": " BASE64 文件编码",//明文 BASE64 编码

    "code": 200//根据 code 类型判断本次操作是否成功

}

## 数据签名接口

### 接口描述

三方应用系统调用此服务对数据进行签名操作。签名操作可用来验证数据是否被篡改。

### 服务接口

#### 3.9.2.1接口定义

接口名称： /api/signData 通信协议：http 请求方式：post

#### 3.9.2.2请求参数

{

    "appCode":"540EA9EFDAB94DA583E323BDE1703A92",//应用唯一标识

    "data":"2233399877"//签名数据

}

#### 3.9.2.3响应参数

{

    "message": "成功",//当失败时返回原因

    "data": "bfa0cdf00fda8a9a1ca0fbb2e1c2c816b1ade49a4fb0a20af7139eeb08c039c041d3d275e468217db3329552faf17f17860d60350e8b557d0c7d84321fcd7a4f",//签名值

    "code": 200//根据 code 类型判断本次操作是否成功

}

## 数据验签接口

### 接口描述

提供给三方应用系统，用来验证签名数据。验证数据在流转和存储过程中的真实性。

### 服务接口

#### 3.10.2.1接口定义

接口名称： /api/verifyData 通信协议：http 请求方式：post

#### 3.10.2.2请求参数

{

    "appCode":"540EA9EFDAB94DA583E323BDE1703A92",//应用唯一标识

    "data":"2233399877",//签名数据

    "signature":"bfa0cdf00fda8a9a1ca0fbb2e1c2c816b1ade49a4fb0a20af7139eeb08c039c041d3d275e468217db3329552faf17f17860d60350e8b557d0c7d84321fcd7a4f"//签名值

}

#### 3.10.2.3响应参数：

{

    "message": "成功",//当失败时返回原因

    "data": "true",//成功后返回标识

    "code": 200//根据 code 类型判断本次操作是否成功

}

## 文件签名接口

### 接口描述

应用系统调用此服务对文件签名用于对文件生成的base64编码数据签名操作。此签名数据用来验证文件是否被篡改。

### 服务接口

#### 3.11.2.1接口定义

接口名称： /api/signFile 通信协议：http请求方式：post

#### 3.11.2.2请求参数

{

    "appCode": "540EA9EFDAB94DA583E323BDE1703A92",//应用唯一标识

    "fileContent":"BASE64 文件编码内容",//BASE64 编码

}

#### 3.11.2.3响应参数

{

    "message": "成功",//当失败时返回原因

    "data": "7b183e3bd1a7d68bd3a26cf201a22298a6e4dd34cb9787550e1d3e856abc0441c78b04dc28318186fcdf6bc861560dc05d37f9561407b9e7d54675b825235de5",//签名值

    "code": 200//根据 code 类型判断本次操作是否成功

}

## 文件验签接口

### 接口描述

用于三方应用系统对文件进行验签操作，验证文件在流转和存储过程中的真实性。

### 服务接口

#### 3.12.2.1接口定义

接口名称： /api/verifyFile 通信协议：http 请求方式：post

#### 3.12.2.2请求参数

{

    "appCode": "540EA9EFDAB94DA583E323BDE1703A92",//应用唯一标识

    "fileContent":"BASE64 文件编码内容",//BASE64 编码

    "signature":"9621278774caaf0f72f9d2eae01620cc5d6884c14d03582b37b33792eca896a23f3a9c45ab052ab7258c884187097a0b5979c558eaefcad222711c0c37ec198e"//签名值

}

#### 3.12.2.3响应参数

{

    "message": "成功",//当失败时返回原因

    "data": "true",//成功后返回标识

    "code": 200//根据 code 类型判断本次操作是否成功

}

# 服务开发规范-V2.0

## 4.1客户端模式授权码

### 4.1.1接口描述

应用接入时会分配客户端id:client\_id和客户端secret:client\_secret。且给应用进行API授权操作。此时应用调用API下各个服务时，需要先获取token进行授权操作，然后根据在有效期限内的token去鉴权获取各服务接口。

### 4.1.2服务接口：

[http://[*ip*]:[*port*]/[*projectname*]](http://[ip]:[port]/%5bservername%5d/risen/core/mag/org/loadCmmTreeCoreOrgTree.action?TYPE=jsn_grid)/public/oauth2/token

* 请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 值 | 类型 | 说明 |
| grant\_type | client\_credentials | 字符串 | 说明授权类型为客户端类型（固定值） |
| scope | cipher | 字符串 | 访问授权的范围（固定值） |
| client\_id | risen\_wUceuIuyj1Cof8pQy2yclA | 字符串 | 客户端id |
| client\_secret | 2cUFnYoaRxdre1e88SyUJ2txBp1TuccHQz0HCA0xTPniChUAcq32wL4YlTFQlBsg | 字符串 | 客户端secret |

* 返回结果：

|  |
| --- |
| {  "access\_token":"l3DbofLaXyiZJ60gp2n-**iQiGmEgQND1V**", //授权码token  "expires\_in":3600 // 有效期限  } |

## 获取密钥

### 接口描述

为了满足加密粒度的要求，应用系统需要使用不同的密钥来对不同的数据进行加密，因此应用系统在需要新密钥对新数据进行加密时，需要调用该接口来获取新密钥。

### 服务接口

#### 4.1.2.1接口定义

[http://[*ip*]:[*port*]/[*projectname*]](http://[ip]:[port]/%5bservername%5d/risen/core/mag/org/loadCmmTreeCoreOrgTree.action?TYPE=jsn_grid)*/public/service/endPoint.do*

通信协议：http 请求方式：post

#### 4.1.2.2请求参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 参数 | | 值 | 类型 | 必填 | 说明 |
| meta | serviceCode | commercialCipherService#getDataKey | String | 是 | 获取密钥接口【固定值】 |
| ver | V2.0 | String | 是 | 版本【固定值】 |
| token | 7Zz4xnthZolqdcnkfr25SYDRlnwjSSP8L4YlTFQlBsg | String | 是 | 获取的授权码token |
| data | appCode | 8C09761853134B6DA11B906162253ECC | String | 是 | 应用标识 |
| **size** | 1 | String | 是 | 获取密钥个数 |

{

    "meta": {

        "serviceCode": "commercialCipherService#getDataKey", //获取密钥接口

        "ver": "V2.0",

        "token": "vYCKDyDJGav3VON5aQJW7et4asAd5Ja7"

    },

    "data": {

        "appCode": "64AB10FA0F644E8C802E78A7F39F1C9A",//应用唯一标识

     "size": 2 //本次生成的密钥数量

    }

}

#### 4.1.2.3响应参数：

{

"meta": {

"respMsg": "0000",

"serviceCode": "commercialCipherService#getDataKey",

"ver": "V2.0"

}，

"data": {

"message": "成功",

"data": [

{

"dataKey": "517f0aa50dd0536d5f3a32774181a046"//数据密钥

},

{

"dataKey": "c92473c7760a85feba00e001ddbdcba7"//数据密钥

}

],

"code": 200//根据 code 类型判断本次操作是否成功

}

}

## 数据加密接口

### 接口描述

对于一些重要的用户信息、组织机构信息、日志信息等可以使用该接口对相关的数据进行加密。

### 接口业务流程图



### 服务接口

[http://[*ip*]:[*port*]/[*projectname*]](http://[ip]:[port]/%5bservername%5d/risen/core/mag/org/loadCmmTreeCoreOrgTree.action?TYPE=jsn_grid)*/public/service/endPoint.do*

通信协议：http 请求方式：post

#### 4.2.3.1请求参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 参数 | | 值 | 类型 | 必填 | 说明 |
| meta | serviceCode | commercialCipherService#encryptData | String | 是 | 获取加密接口【固定值】 |
| ver | V2.0 | String | 是 | 版本【固定值】 |
| token | 7Zz4xnthZolqdcnkfr25SYDRlnwjSSP8L4YlTFQlBsg | String | 是 | 获取的授权码token |
| data | appCode | 8C09761853134B6DA11B906162253ECC | String | 是 | 应用标识 |
| dataKey | 47ac46530bac8de869896c8db7de7c69 | String | 是 | 数据密钥 |
| **data** | 176324 | String | 是 | 明文数据 |

{

    "meta": {

        "serviceCode": "commercialCipherService#encryptData", //加密接口

        "ver": "V2.0",

        "token": "vYCKDyDJGav3VON5aQJW7et4asAd5Ja7"

    },

    "data": {

"appCode": "DFB41EB431524D84A34F64546FF341DE", //应用唯一标识

"dataKey": "47ac46530bac8de869896c8db7de7c69", //数据密钥

"data": "13212344321" //明文数据

}

}

#### 4.2.3.2响应参数：

{

"meta": {

"respMsg": "0000",

"serviceCode": "commercialCipherService#encryptData ",

"ver": "V2.0"

},

"data": {

"message": "成功", //当失败时返回原因

"data": "b6c4b63f8d3b74813d30830ca7accb3fdbe57d625ddd1615a733d6d64544eaa3", //密文数据

"code": 200 //根据 code 类型判断本次操作是否成功

}

}

## 数据解密

### 接口描述

用于数据的解密操作，在应用系统的数据进行页面展示时，调用此接口将数据解密后展示明文。

### 接口流程图



### 服务接口

#### 4.3.3.1接口定义

[http://[*ip*]:[*port*]/[*projectname*]](http://[ip]:[port]/%5bservername%5d/risen/core/mag/org/loadCmmTreeCoreOrgTree.action?TYPE=jsn_grid)*/public/service/endPoint.do*

通信协议：http 请求方式：post

#### 4.3.3.2请求参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 参数 | | 值 | 类型 | 必填 | 说明 |
| meta | serviceCode | commercialCipherService#decryptData | String | 是 | 获取解密接口【固定值】 |
| ver | V2.0 | String | 是 | 版本【固定值】 |
| token | 7Zz4xnthZolqdcnkfr25SYDRlnwjSSP8L4YlTFQlBsg | String | 是 | 获取的授权码token |
| data | appCode | 8C09761853134B6DA11B906162253ECC | String | 是 | 应用标识 |
| **dataKey** | 5fb1192655db0f620dca5cfdb2da3ed2 | String | 是 | 数据密钥 |
| data | b6c4b63f8d3b74813d30830ca7accb3fdbe57d625ddd1615a733d6d64544eaa3 | String | 是 | 密文数据 |

{

    "meta": {

        "serviceCode": "commercialCipherService#decryptData", //解密接口

        "ver": "V2.0",

        "token": "vYCKDyDJGav3VON5aQJW7et4asAd5Ja7"

    },

    "data": {

"appCode": "01bc857f5c1245708b1247236d9869fe", //应用唯一标识

"dataKey": "5fb1192655db0f620dca5cfdb2da3ed2",//数据密钥

"data": "b6c4b63f8d3b74813d30830ca7accb3fdbe57d625ddd1615a733d6d64544eaa3" //密文数据

}

}

#### 4.3.3.3响应参数

{

"meta": {

"respMsg": "0000",

"serviceCode": "commercialCipherService#decryptData",

"ver": "V2.0"

}，

"data": {

"message": "成功",//当失败时返回原因

"data": "123456789123456789",//明文数据

"code": 200//根据 code 类型判断本次操作是否成功

}

}

## 批量数据加密

### 接口描述

提供给三方应用，进行批量对数据进行加密操作。

### 服务接口

#### 4.4.2.1接口定义

[http://[*ip*]:[*port*]/[*projectname*]](http://[ip]:[port]/%5bservername%5d/risen/core/mag/org/loadCmmTreeCoreOrgTree.action?TYPE=jsn_grid)*/public/service/endPoint.do*

通信协议：http 请求方式：post

#### 4.4.2.2请求参数

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 参数 | | | 值 | 类型 | 必填 | 说明 |
| meta | serviceCode | | commercialCipherService#batchEncrypt | String | 是 | 获取批量加密接口【固定值】 |
| ver | | V2.0 | String | 是 | 版本【固定值】 |
| token | | 7Zz4xnthZolqdcnkfr25SYDRlnwjSSP8L4YlTFQlBsg | String | 是 | 获取的授权码token |
| data | appCode | | 8C09761853134B6DA11B906162253ECC | String | 是 | 应用标识 |
| batchData | dataKey | 9ac0979a0242c6a0c0ed1ade0e653421 | String | 是 | 数据密钥 |
| IDCard1 | 123456789123456789 | String | 是 | 明文数据 |

{

    "meta": {

        "serviceCode": "commercialCipherService#batchEncrypt", //批量加密接口

        "ver": "V2.0",

        "token": "vYCKDyDJGav3VON5aQJW7et4asAd5Ja7"

    },

    "data": {

     "appCode": "01BC857F5C1245708B1247236D9869FE",//应用唯一标识

     "batchData": [

        {

            "dataKey": "9ac0979a0242c6a0c0ed1ade0e653421",//数据密钥

            "IDCard1": "123456789123456789"//明文数据

        },

        {

            "dataKey": "dfab61e346c9ba682d785cf60366b8d3",//数据密钥

            "IDCard2": "987654321987654321"//明文数据

        }

    ]

}

}

#### 4.4.2.3响应参数：

{

"meta": {

"respMsg": "0000",

"serviceCode": "commercialCipherService#batchEncrypt",

"ver": "V2.0"

}，

"data": {

    "message": "成功",//当失败时返回原因

    "data": [

        {

            "IDCard1": "3a9574cf82aa215bc5d4e65ab1960e0e7bc59419f9951779d5e94231089d7b6f"//密文数据

        },

        {

            "IDCard2": "ec05e7341effe2aca141b44081b13e4914e63acf48f87e4fe2929872e03c292b"//密文数据

        }

    ],

    "code": 200//根据 code 类型判断本次操作是否成功

}

}

## 批量数据解密接口

### 接口描述

对大量密文数据进行批量解密，提高三方应用系统的解密效率。

### 服务接口

#### 4.5.2.1接口定义

[http://[*ip*]:[*port*]/[*projectname*]](http://[ip]:[port]/%5bservername%5d/risen/core/mag/org/loadCmmTreeCoreOrgTree.action?TYPE=jsn_grid)*/public/service/endPoint.do*

通信协议：http 请求方式：post

#### 4.5.2.2请求参数

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 参数 | | | 值 | 类型 | 必填 | 说明 |
| meta | serviceCode | | commercialCipherService#batchDecrypt | String | 是 | 获取批量解密接口【固定值】 |
| ver | | V2.0 | String | 是 | 版本【固定值】 |
| token | | 7Zz4xnthZolqdcnkfr25SYDRlnwjSSP8L4YlTFQlBsg | String | 是 | 获取的授权码token |
| data | appCode | | 8C09761853134B6DA11B906162253ECC | String | 是 | 应用标识 |
| batchData | dataKey | 9ac0979a0242c6a0c0ed1ade0e653421 | String | 是 | 数据密钥 |
| IDCard1 | 3a9574cf82aa215bc5d4e65ab1960e0e7bc59419f9951779d5e94231089d7b6f | String | 是 | 密文数据 |

{

    "meta": {

        "serviceCode": "commercialCipherService#batchDecrypt",//批量解密接口

        "ver": "V2.0",

        "token": "vYCKDyDJGav3VON5aQJW7et4asAd5Ja7"

    },

    "data": {

     "appCode": "01BC857F5C1245708B1247236D9869FE",//应用唯一标识

     "batchData": [

        {

            "dataKey": "9ac0979a0242c6a0c0ed1ade0e653421",//数据密钥

            "IDCard1": "3a9574cf82aa215bc5d4e65ab1960e0e7bc59419f9951779d5e94231089d7b6f" //密文数据

        },

        {

            "dataKey": "dfab61e346c9ba682d785cf60366b8d3",//数据密钥

            "IDCard2": "ec05e7341effe2aca141b44081b13e4914e63acf48f87e4fe2929872e03c292b"//密文数据

        }

     ]

}

}

#### 4.5.2.3响应参数

{

"meta": {

"respMsg": "0000",

"serviceCode": "commercialCipherService#batchDecrypt",

"ver": "V2.0"

},

"data": {

     "message": "成功",//当失败时返回原因

     "data": [

         {

            "IDCard1": "123456789123456789"//明文数据

         },

         {

            "IDCard2": "987654321987654321"//明文数据

         }

     ],

     "code": 200//根据 code 类型判断本次操作是否成功

}

}

## 不可逆加密

### 接口描述

不可逆加密是指数据哈希。数据哈希使用的是sm3算法，具备有不可逆性，因此使用该算法进行加密的数据是那些不需要再进行解密的数据。该方法不需要使用数据密钥。常用此接口加密的数据例如用户的登录密码。

### 服务接口

#### 4.6.2.1接口定义

[http://[*ip*]:[*port*]/[*projectname*]](http://[ip]:[port]/%5bservername%5d/risen/core/mag/org/loadCmmTreeCoreOrgTree.action?TYPE=jsn_grid)*/public/service/endPoint.do*

通信协议：http 请求方式：post

#### 4.6.2.2请求参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 参数 | | 值 | 类型 | 必填 | 说明 |
| meta | serviceCode | commercialCipherService#hashData | String | 是 | 获取不可逆加密接口【固定值】 |
| ver | V2.0 | String | 是 | 版本【固定值】 |
| token | 7Zz4xnthZolqdcnkfr25SYDRlnwjSSP8L4YlTFQlBsg | String | 是 | 获取的授权码token |
| data | appCode | 8C09761853134B6DA11B906162253ECC | String | 是 | 应用标识 |
| data | 13333 | String | 是 | 明文数据 |

{

    "meta": {

        "serviceCode": "commercialCipherService#hashData", //不可逆加密接口

        "ver": "V2.0",

        "token": "vYCKDyDJGav3VON5aQJW7et4asAd5Ja7"

    },

    "data": {

     "appCode": "01bc857f5c1245708b1247236d9869fe", //应用唯一标识

"data":"123456789123456789" //明文数据

}

}

#### 4.6.2.3响应参数

{

"meta": {

"respMsg": "0000",

"serviceCode": "commercialCipherService#getDataKey",

"ver": "V2.0"

},

"data": {

"message": "成功",//当失败时返回原因

"data":"f72a78ff84bd13e6a011f5427e53fa278989e0afe9d5f2dd811ffa 58d6c290b2", //密文数据

"code": 200//根据 code 类型判断本次操作是否成功

}

}

## 文件加密接口

### 接口描述

为了保证存储安全，服务器内部存放的文件应该都是密文，因此在文件的收发文过程产生的文件，都应该进行加密，使文件以密文的形式在服务器上存储。将文件BASE64编码后调用接口服务进行加密操作,文件超过4MB建议使用本地API方法。

### 服务接口

#### 4.7.2.1接口定义

[http://[*ip*]:[*port*]/[*projectname*]](http://[ip]:[port]/%5bservername%5d/risen/core/mag/org/loadCmmTreeCoreOrgTree.action?TYPE=jsn_grid)*/public/service/endPoint.do*

通信协议：http 请求方式：post

#### 4.7.2.2请求参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 参数 | | 值 | 类型 | 必填 | 说明 |
| meta | serviceCode | commercialCipherService#encryptFile | String | 是 | 获取文件加密接口【固定值】 |
| ver | V2.0 | String | 是 | 版本【固定值】 |
| token | 7Zz4xnthZolqdcnkfr25SYDRlnwjSSP8L4YlTFQlBsg | String | 是 | 获取的授权码token |
| data | appCode | 8C09761853134B6DA11B906162253ECC | String | 是 | 应用标识 |
| dataKey | b04d4bd00a04832290f9d17f07379bfe | String | 是 | 获取密钥个数 |
| fileContent | BASE64 编码文件内容 | String | 是 | 明文 BASE64 编码 |

{

    "meta": {

        "serviceCode": "commercialCipherService#encryptFile", //文件加密接口

        "ver": "V2.0",

        "token": "vYCKDyDJGav3VON5aQJW7et4asAd5Ja7"

    },

    "data": {

     "appCode": "01BC857F5C1245708B1247236D9869FE",//应用唯一标识

     "dataKey": "b04d4bd00a04832290f9d17f07379bfe",//数据密钥

     "fileContent": "BASE64 编码文件内容"//明文 BASE64 编码

}

}

#### 4.7.2.3响应参数：

{

"meta": {

"respMsg": "0000",

"serviceCode": "commercialCipherService#encryptFile",

"ver": "V2.0"

},

"data": {

     "message": "成功",//当失败时返回原因

     "data": " BASE64 编码文件内容",//密文 BASE64 编码

     "code": 200//根据 code 类型判断本次操作是否成功

}

}

## 文件解密接口

### 接口描述

三方应用在服务器上存储的是加密后的文件，当三方应用需要查看文件内容时，必须调用文件的解密接口将文件解密。

### 服务接口

#### 4.8.2.1接口定义

[http://[*ip*]:[*port*]/[*projectname*]](http://[ip]:[port]/%5bservername%5d/risen/core/mag/org/loadCmmTreeCoreOrgTree.action?TYPE=jsn_grid)*/public/service/endPoint.do*

通信协议：http 请求方式：post

#### 4.8.2.2请求参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 参数 | | 值 | 类型 | 必填 | 说明 |
| meta | serviceCode | commercialCipherService#decryptFile | String | 是 | 获取文件解密接口【固定值】 |
| ver | V2.0 | String | 是 | 版本【固定值】 |
| token | 7Zz4xnthZolqdcnkfr25SYDRlnwjSSP8L4YlTFQlBsg | String | 是 | 获取的授权码token |
| data | appCode | 8C09761853134B6DA11B906162253ECC | String | 是 | 应用标识 |
| **size** | 1 | String | 是 | 获取密钥个数 |

{

    "meta": {

        "serviceCode": "commercialCipherService#decryptFile", //文件解密接口

        "ver": "V2.0",

        "token": "vYCKDyDJGav3VON5aQJW7et4asAd5Ja7"

    },

    "data": {

     "appCode": "01BC857F5C1245708B1247236D9869FE",//应用唯一标识

     "dataKey": "b04d4bd00a04832290f9d17f07379bfe",//数据密钥

     "fileContent": "BASE64 文件编码内容"//密文 BASE64 编码

}

}

#### 4.8.2.3响应参数

{

"meta": {

"respMsg": "0000",

"serviceCode": "commercialCipherService#decryptFile",

"ver": "V2.0"

},

"data": {

     "message": "成功",//当失败时返回原因

     "data": " BASE64 文件编码",//明文 BASE64 编码

     "code": 200//根据 code 类型判断本次操作是否成功

}

}

## 数据签名接口

### 接口描述

三方应用系统调用此服务对数据进行签名操作。签名操作可用来验证数据是否被篡改。

### 服务接口

#### 4.9.2.1接口定义

[http://[*ip*]:[*port*]/[*projectname*]](http://[ip]:[port]/%5bservername%5d/risen/core/mag/org/loadCmmTreeCoreOrgTree.action?TYPE=jsn_grid)*/public/service/endPoint.do*

通信协议：http 请求方式：post

#### 4.9.2.2请求参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 参数 | | 值 | 类型 | 必填 | 说明 |
| meta | serviceCode | commercialCipherService#signData | String | 是 | 获取数据签名接口【固定值】 |
| ver | V2.0 | String | 是 | 版本【固定值】 |
| token | 7Zz4xnthZolqdcnkfr25SYDRlnwjSSP8L4YlTFQlBsg | String | 是 | 获取的授权码token |
| data | appCode | 8C09761853134B6DA11B906162253ECC | String | 是 | 应用标识 |
| data | 2233399877 | String | 是 | 签名数据 |

{

    "meta": {

        "serviceCode": "commercialCipherService#signData", //数据签名接口

        "ver": "V2.0",

        "token": "vYCKDyDJGav3VON5aQJW7et4asAd5Ja7"

    },

    "data": {

     "appCode":"540EA9EFDAB94DA583E323BDE1703A92",//应用唯一标识

     "data":"2233399877"//签名数据

}

}

#### 4.9.2.3响应参数

{

"meta": {

"respMsg": "0000",

"serviceCode": "commercialCipherService#signData",

"ver": "V2.0"

},

"data": {

     "message": "成功",//当失败时返回原因

     "data": "bfa0cdf00fda8a9a1ca0fbb2e1c2c816b1ade49a4fb0a20af7139eeb08c039c041d3d275e468217db3329552faf17f17860d60350e8b557d0c7d84321fcd7a4f",//签名值

     "code": 200//根据 code 类型判断本次操作是否成功

}

}

## 数据验签接口

### 接口描述

提供给三方应用系统，用来验证签名数据。验证数据在流转和存储过程中的真实性。

### 服务接口

#### 4.10.2.1接口定义

[http://[*ip*]:[*port*]/[*projectname*]](http://[ip]:[port]/%5bservername%5d/risen/core/mag/org/loadCmmTreeCoreOrgTree.action?TYPE=jsn_grid)*/public/service/endPoint.do*

通信协议：http 请求方式：post

#### 4.10.2.2请求参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 参数 | | 值 | 类型 | 必填 | 说明 |
| meta | serviceCode | commercialCipherService#verifyData | String | 是 | 获取数据验签接口【固定值】 |
| ver | V2.0 | String | 是 | 版本【固定值】 |
| token | 7Zz4xnthZolqdcnkfr25SYDRlnwjSSP8L4YlTFQlBsg | String | 是 | 获取的授权码token |
| data | appCode | 8C09761853134B6DA11B906162253ECC | String | 是 | 应用标识 |
| data | 2233399877 | String | 是 | 签名数据 |
| signature | bfa0cdf00fda8a9a1ca0fbb2e1c2c816b1ade49a4fb0a20af7139eeb08c039c041d3d275e468217db3329552faf17f17860d60350e8b557d0c7d84321fcd7a4f | String | 是 | 签名值 |

{

    "meta": {

        "serviceCode": "commercialCipherService#verifyData", //数据验签接口

        "ver": "V2.0",

        "token": "vYCKDyDJGav3VON5aQJW7et4asAd5Ja7"

    },

    "data": {

     "appCode":"540EA9EFDAB94DA583E323BDE1703A92",//应用唯一标识

     "data":"2233399877",//签名数据

"signature":"bfa0cdf00fda8a9a1ca0fbb2e1c2c816b1ade49a4fb0a20af7139eeb08c039c041d3d275e468217db3329552faf17f17860d60350e8b557d0c7d84321fcd7a4f"//签名值

}

}

#### 4.10.2.3响应参数：

{

"meta": {

"respMsg": "0000",

"serviceCode": "commercialCipherService#verifyData",

"ver": "V2.0"

},

"data": {

     "message": "成功",//当失败时返回原因

     "data": "true",//成功后返回标识

     "code": 200//根据 code 类型判断本次操作是否成功

}

}

## 文件签名接口

### 接口描述

应用系统调用此服务对文件签名用于对文件生成的base64编码数据签名操作。此签名数据用来验证文件是否被篡改。

### 服务接口

#### 4.11.2.1接口定义

[http://[*ip*]:[*port*]/[*projectname*]](http://[ip]:[port]/%5bservername%5d/risen/core/mag/org/loadCmmTreeCoreOrgTree.action?TYPE=jsn_grid)*/public/service/endPoint.do*

通信协议：http 请求方式：post

#### 4.11.2.3响应参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 参数 | | 值 | 类型 | 必填 | 说明 |
| meta | serviceCode | commercialCipherService#signFile | String | 是 | 获取文件签名接口【固定值】 |
| ver | V2.0 | String | 是 | 版本【固定值】 |
| token | 7Zz4xnthZolqdcnkfr25SYDRlnwjSSP8L4YlTFQlBsg | String | 是 | 获取的授权码token |
| data | appCode | 8C09761853134B6DA11B906162253ECC | String | 是 | 应用标识 |
| fileContent | BASE64 文件编码内容 | String | 是 | BASE64 编码 |

{

    "meta": {

        "serviceCode": "commercialCipherService#signFile", //文件签名接口

        "ver": "V2.0",

        "token": "vYCKDyDJGav3VON5aQJW7et4asAd5Ja7"

    },

    "data": {

     "appCode": "540EA9EFDAB94DA583E323BDE1703A92",//应用唯一标识

     "fileContent":"BASE64 文件编码内容",//BASE64 编码

}

}

#### 4.1.2.3响应参数：

{

"meta": {

"respMsg": "0000",

"serviceCode": "commercialCipherService#signFile",

"ver": "V2.0"

},

"data": {

     "message": "成功",//当失败时返回原因

"data": "7b183e3bd1a7d68bd3a26cf201a22298a6e4dd34cb9787550e1d3e856abc0441c78b04dc28318186fcdf6bc861560dc05d37f9561407b9e7d54675b825235de5",//签名值

     "code": 200//根据 code 类型判断本次操作是否成功

}

}

## 文件验签接口

### 接口描述

用于三方应用系统对文件进行验签操作，验证文件在流转和存储过程中的真实性。

### 服务接口

#### 4.12.2.1接口定义

[http://[*ip*]:[*port*]/[*projectname*]](http://[ip]:[port]/%5bservername%5d/risen/core/mag/org/loadCmmTreeCoreOrgTree.action?TYPE=jsn_grid)*/public/service/endPoint.do*

通信协议：http 请求方式：post

#### 4.12.2.2请求参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 参数 | | 值 | 类型 | 必填 | 说明 |
| meta | serviceCode | commercialCipherService#verifyFile | String | 是 | 获取文件验签接口【固定值】 |
| ver | V2.0 | String | 是 | 版本【固定值】 |
| token | 7Zz4xnthZolqdcnkfr25SYDRlnwjSSP8L4YlTFQlBsg | String | 是 | 获取的授权码token |
| data | appCode | 8C09761853134B6DA11B906162253ECC | String | 是 | 应用标识 |
| fileContent | BASE64 文件编码内容 | String | 是 | BASE64 编码 |
| signature | 9621278774caaf0f72f9d2eae01620cc5d6884c14d03582b37b33792eca896a23f3a9c45ab052ab7258c884187097a0b5979c558eaefcad222711c0c37ec198e | String | 是 | 签名值 |

{

    "meta": {

        "serviceCode": "commercialCipherService#verifyFile", //文件验签接口

        "ver": "V2.0",

        "token": "vYCKDyDJGav3VON5aQJW7et4asAd5Ja7"

    },

    "data": {

     "appCode": "540EA9EFDAB94DA583E323BDE1703A92",//应用唯一标识

     "fileContent":"BASE64 文件编码内容",//BASE64 编码

"signature":"9621278774caaf0f72f9d2eae01620cc5d6884c14d03582b37b33792eca896a23f3a9c45ab052ab7258c884187097a0b5979c558eaefcad222711c0c37ec198e"//签名值

}

}

#### 4.12.2.3响应参数

{

"meta": {

"respMsg": "0000",

"serviceCode": "commercialCipherService#verifyFile",

"ver": "V2.0"

},

"data": {

     "message": "成功",//当失败时返回原因

     "data": "true",//成功后返回标识

     "code": 200//根据 code 类型判断本次操作是否成功

}

}

# 大屏统计接口

## 客户端模式授权码

### 接口描述

应用接入时会分配客户端id:client\_id和客户端secret:client\_secret。且给应用进行API授权操作。此时应用调用API下各个服务时，需要先获取token进行授权操作，然后根据在有效期限内的token去鉴权获取各服务接口。

### 服务接口：

[http://[*ip*]:[*port*]/[*projectname*]](http://[ip]:[port]/%5bservername%5d/risen/core/mag/org/loadCmmTreeCoreOrgTree.action?TYPE=jsn_grid)/public/oauth2/token

* 请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 值 | 类型 | 说明 |
| grant\_type | client\_credentials | 字符串 | 说明授权类型为客户端类型（固定值） |
| scope | statistics | 字符串 | 访问授权的范围（固定值） |
| client\_id | risen\_wUceuIuyj1Cof8pQy2yclA | 字符串 | 客户端id |
| client\_secret | 2cUFnYoaRxdre1e88SyUJ2txBp1TuccHQz0HCA0xTPniChUAcq32wL4YlTFQlBsg | 字符串 | 客户端secret |

* 返回结果：

|  |
| --- |
| {  "access\_token":"l3DbofLaXyiZJ60gp2n-**iQiGmEgQND1V**", //授权码token  "expires\_in":3600 // 有效期限  } |

## 统计接口描述

用于大屏获取相关统计接口。

### 接口定义

[http://[*ip*]:[*port*]/[*projectname*]](http://[ip]:[port]/%5bservername%5d/risen/core/mag/org/loadCmmTreeCoreOrgTree.action?TYPE=jsn_grid)*/public/service/endPoint.do*

通信协议：http 请求方式：post

### 请求参数

统计接口，请求参数和响应参数是一致的。其中请求参数不上送时，则响应的total是总数量，然后是以应用分组后的个应用下的数量。若上送了应用标识，则total是应用下的总数。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 参数 | | 值 | 类型 | 必填 | 说明 |
| meta | serviceCode | commercialCipherStatisticsService#xxxxxx | String | 是 | 获取对应接口【参考下表中接口定义】 |
| ver | V2.0 | String | 是 | 版本【固定值】 |
| token | 7Zz4xnthZolqdcnkfr25SYDRlnwjSSP8L4YlTFQlBsg | String | 是 | 获取的授权码token |
| data | cslogAppUuid | 5FFCE546C588426B83479B02375BFFF1 | String | 否 | 应用标识；不上送，则获取全部总数量。 |

meta.serviceCode值取以下参数。根据对应接口进行获取。

|  |  |
| --- | --- |
| **接口定义** | **说明** |
| commercialCipherStatisticsService#dekCount | 密钥生成数量 |
| commercialCipherStatisticsService#destroyedDekCount | 密钥销毁数量 |
| commercialCipherStatisticsService#archiveDekCount | 密钥存档数量 |
| commercialCipherStatisticsService#useDekCount | 密钥使用次数 |
| commercialCipherStatisticsService#encryptCount | 加密次数 |
| commercialCipherStatisticsService#decryptCount | 解密次数 |
| commercialCipherStatisticsService#signCount | 签名次数 |
| commercialCipherStatisticsService#verifyCount | 验签次数 |

### 响应参数

{

    "meta": {

        "respCode": "0000",

        "respMsg": "操作成功",

        "serviceCode": "commercialCipherStatisticsService#XXXXXXX",

        "ver": "V2.0"

    },

    "data": {

        "strMap": {

            "total": "1",//总数（若上送了应用编码，则是该应用下的总数）

            "2098AE93AA2C4A4596DF3375A72EF875": "{商密应用:1}"// 应用分组下的总数

        }

    }

}