PHP版服务器端工具包（Linux版）

软件需求说明书

（版本：3.0）

中国金融认证中心

2017年8月3日

版权声明：本文档的版权属于中国金融认证中心，任何人或组织未经许可，

不得擅自修改、拷贝或以其它方式使用本文档中的内容

文档修订记录

本文档会随时保持更新，请与中国金融认证中心索要最新版本

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **版本** | **内容** | **日期** | **编写** | **审核** |
| 3.0 | 需求初稿 | 2017/08/03 | 胡军华 | 贺 磊  张 哲 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

注：对该文件内容增加、删除或修改须填写此修订记录，详细记载变更信息，以保证其可追溯性。

目 录

[1 项目范围 1](#_Toc490734360)

[1.1 项目描述 1](#_Toc490734361)

[1.2 项目背景 1](#_Toc490734362)

[1.3 项目目标 1](#_Toc490734363)

[2 版本需求 1](#_Toc490734364)

[2.1 操作系统版本 1](#_Toc490734365)

[2.2 PHP版本 2](#_Toc490734366)

[2.3 运行环境依赖 2](#_Toc490734367)

[3 接口列表 2](#_Toc490734368)

[3.1 cfca\_initialize 2](#_Toc490734369)

[3.2 cfca\_uninitialize 3](#_Toc490734370)

[3.3 cfca\_signData\_PKCS1 3](#_Toc490734371)

[3.4 cfca\_signData\_PKCS7Detached 5](#_Toc490734372)

[3.5 cfca\_signData\_PKCS7Attached 6](#_Toc490734373)

[3.6 cfca\_verifyDataSignature\_PKCS1 7](#_Toc490734374)

[3.7 cfca\_verifyDataSignature\_PKCS7Detached 8](#_Toc490734375)

[3.8 cfca\_verifyDataSignature\_PKCS7Attached 9](#_Toc490734376)

[3.9 cfca\_signFile\_PKCS7Detached 10](#_Toc490734377)

[3.10 cfca\_verifyFileSignature\_PKCS7Detached 11](#_Toc490734378)

[3.11 cfca\_encryptDataToCMSEnvelope 12](#_Toc490734379)

[3.12 cfca\_decryptDataFromCMSEnvelope 13](#_Toc490734380)

[3.13 cfca\_symEncryptFile 14](#_Toc490734381)

[3.14 cfca\_symDecryptFile 15](#_Toc490734382)

[3.15 cfca\_verifyCertificate 16](#_Toc490734383)

[3.16 cfca\_getCertificateInfo 17](#_Toc490734384)

[3.17 cfca\_getPublicCertFromPFX 18](#_Toc490734385)

[3.18 cfca\_calculateDataHash 19](#_Toc490734386)

[3.19 cfca\_calculateFileHash 19](#_Toc490734387)

[4 提供程序 20](#_Toc490734388)

# 项目范围

## 项目描述

该项目的主要内容为开发供PHP5.6、7.0和7.1调用的扩展库。

## 项目背景

目前，PHP程序调用Java程序完成签名、验签等证书相关业务操作。PHP与Java是两种不同的语言，二者之间的通讯通过Socket完成。在这种架构下，Socket通讯的稳定性对整个程序的运行产生影响。若Socket通讯不稳定，系统业务将无法完成。同时，Java的程序运行速度也决定了系统的性能。稳定性和效率，是PHP通过Socket调用Java进行证书相关业务操作这一方案的短板。

自PHP3开始，PHP支持直接调用C/C++语言开发的扩展库。 PHP 7.1依然支持这一动态调用。

## 项目目标

保证在系统需求中所列出的操作系统平台下的正常运行。

# 版本需求

## 操作系统版本

支持以下操作系统:

CentOS 7.0 64bit

CentOS 7.1 64bit。

## PHP版本

PHP5.6系列(NTS)；

PHP7.0系列(NTS)；

PHP7.1系列(NTS)。

## 运行环境依赖

扩展库在开发中用到了C++11的特性，其对运行时环境要求如下：CXXABI版本不低于1.3.7（GCC4.8.0）；GLIBC版本不低于2.14。

　　CentOS7.0和7.1自带环境已经满足要求。

# 接口列表

## cfca\_initialize

integer cfca\_initialize(string strConfigFilePath);

**描述:**

扩展初始化。

调用本函数库中其它函数之前调用cfca\_initialize()。如果需要在多线程环境下调用此函数库中的函数，cfca\_initialize()需要在开启多线程之前调用。

此函数只需要调用一次。

**参数**：

strConfigFilePath：[IN]配置文件路径，路径中不许有中文。

**返回值**：

0：成功；

其它：失败。

## cfca\_uninitialize

integer cfca\_uninitialize();

**描述:**

调用本函数库中其它函数之后调用cfca\_uninitialize()。如果需要在多线程环境下调用此函数库中的函数，cfca\_uninitialize()需要在多线程结束之后调用。

此函数只需要调用一次。

**返回值**：

0：成功；

其它：失败。

## cfca\_signData\_PKCS1

integer cfca\_signData\_PKCS1(string strSignAlg,

string strSourceData,

string strPfxFilePath,

string strPfxPassWord,

string strHashAlg,

string strBase64PKCS1Signature)

**描述:**

按指定的算法类型（SM2/RSA），使用软证书对数据进行PKCS#1签名。

**参数：**

string strSignAlg:[IN]算法类型，传入“SM2”或“RSA”，不区分大小写；

string strSourceData:[IN]待签名的字符串，需使用UTF8格式编码，以保持与其他系统兼容；

string strPfxFilePath:[IN]用于签名的软证书文件的路径，路径中不许有中文；

string strPfxPassWord:[IN]用于签名的软证书文件的密码；

string strHashAlg:[IN] 哈希算法，传入“SHA-1”或“SHA-256”，不区分大小写。此参数仅在RSA签名时才起作用；SM2签名默认使用SM3哈希算法，忽略此参数;

string strBase64PKCS1Signature:[OUT] Base64编码的PKCS#1签名结果；

**返回值**：

0：成功；

其它：失败。

## cfca\_signData\_PKCS7Detached

integer cfca\_signData\_PKCS7Detached(string strSignAlg,

string strSourceData,

string strPfxFilePath,

string strPfxPassWord,

string strHashAlg,

string strBase64PKCS7DetachedSignature)

**描述:**

按指定的算法类型（SM2/RSA），使用软证书对数据进行PKCS#7不带原文签名。

**参数：**

string strSignAlg:[IN]算法类型，传入“SM2”或“RSA”，不区分大小写；

string strSourceData:[IN]待签名的字符串，需使用UTF8格式编码，以保持与其他系统兼容；

string strPfxFilePath:[IN]用于签名的软证书文件的路径，路径中不许有中文；

string strPfxPassWord:[IN]用于签名的软证书文件的密码；

string strHashAlg:[IN] 哈希算法，传入“SHA-1”或“SHA-256”，不区分大小写。此参数仅在RSA签名时才起作用；SM2签名默认使用SM3哈希算法，忽略此参数;

string strBase64PKCS7DetachedSignature:[OUT] Base64编码的PKCS#7不带原文签名结果；

**返回值**：

0：成功；

其它：失败。

## cfca\_signData\_PKCS7Attached

integer cfca\_signData\_PKCS7Attached(string strSignAlg,

string strSourceData,

string strPfxFilePath,

string strPfxPassWord,

string strHashAlg,

string strBase64PKCS7AttachedSignature)

**描述:**

按指定的算法类型（SM2/RSA），使用软证书对数据进行PKCS#7带原文签名。

**参数：**

string strSignAlg:[IN]算法类型，传入“SM2”或“RSA”，不区分大小写；

string strSourceData:[IN]待签名的字符串，需使用UTF8格式编码，以保持与其他系统兼容；

string strPfxFilePath:[IN]用于签名的软证书文件的路径，路径中不许有中文；

string strPfxPassWord:[IN]用于签名的软证书文件的密码；

string strHashAlg:[IN] 哈希算法，传入“SHA-1”或“SHA-256”，不区分大小写。此参数仅在RSA签名时才起作用；SM2签名默认使用SM3哈希算法，忽略此参数;

string strBase64PKCS7AttachedSignature:[OUT] Base64编码的PKCS#7不带原文签名结果；

**返回值**：

0：成功；

其它：失败。

## cfca\_verifyDataSignature\_PKCS1

integer cfca\_verifyDataSignature\_PKCS1(string strSignAlg,

string strSourceData,

string strBase64CertContent,

string strHashAlg,

string strBase64PKCS1Signature)

**描述:**

按指定的算法类型（SM2/RSA），验证PKCS#1签名。

**参数：**

string strSignAlg:[IN]算法类型，传入“SM2”或“RSA”，不区分大小写；

string strSourceData:[IN]签名原文字符串，需使用UTF8格式编码，以保持与其他系统兼容；

string strBase64CertContent:[IN] 用于验签的公钥证书内容（Base64编码，不带证书头、尾）；

string strHashAlg:[IN] 哈希算法，传入“SHA-1”或“SHA-256”，不区分大小写。此参数仅在RSA签名时才起作用；SM2签名默认使用SM3哈希算法，忽略此参数;

string strBase64PKCS1Signature:[IN] Base64编码的PKCS#1签名结果；

**返回值**：

0：成功；

其它：失败。

## cfca\_verifyDataSignature\_PKCS7Detached

integer cfca\_verifyDataSignature\_PKCS7Detached (string strSignAlg,

string strSourceData,

string strBase64PKCS7DetachedSignature,

string strBase64SignCertContent)

**描述:**

按指定的算法类型（SM2/RSA），验证PKCS#7不带原文签名。

**参数：**

string strSignAlg:[IN]算法类型，传入“SM2”或“RSA”，不区分大小写；

string strSourceData:[IN]签名原文字符串，需使用UTF8格式编码，以保持与其他系统兼容；

string strBase64PKCS7DetachedSignature:[IN] Base64编码的PKCS#7不带原文签名结果；

string strBase64SignCertContent:[OUT] PKCS#7中的签名证书（Base64编码;

**返回值**：

0：成功；

其它：失败。

## cfca\_verifyDataSignature\_PKCS7Attached

integer cfca\_verifyDataSignature\_PKCS7Attached (string strSignAlg,

string strBase64PKCS7AttachedSignature,

string strBase64SignCertContent,

string strSourceData)

**描述:**

按指定的算法类型（SM2/RSA），验证PKCS#7带原文签名。

**参数：**

string strSignAlg:[IN]算法类型，传入“SM2”或“RSA”，不区分大小写；

string strBase64PKCS7AttachedSignature:[IN] Base64编码的PKCS#7带原文签名结果；

string strBase64SignCertContent:[OUT] PKCS#7中的签名证书（Base64编码）;

string strSourceData:[OUT] PKCS#7签名原文字符串；

**返回值**：

0：成功；

其它：失败

## cfca\_signFile\_PKCS7Detached

integer cfca\_signFile\_PKCS7Detached (string strSignAlg,

string strSourceFilePath,

string strPfxFilePath,

string strPfxPassWord,

string strHashAlg,

string strBase64PKCS7DetachedSignature)

**描述:**

按指定的算法类型（SM2/RSA），使用软证书对文件进行PKCS#7不带原文签名。

**参数：**

string strSignAlg:[IN]算法类型，传入“SM2”或“RSA”，不区分大小写；

string strSourceFilePath:[IN] 待签名的文件原文路径，路径中不许有中文；

string strPfxFilePath:[IN]用于签名的软证书文件的路径，路径中不许有中文；

string strPfxPassWord:[IN]用于签名的软证书文件的密码；

string strHashAlg:[IN] 哈希算法，传入“SHA-1”或“SHA-256”，不区分大小写。此参数仅在RSA签名时才起作用；SM2签名默认使用SM3哈希算法，忽略此参数;

string strBase64PKCS7DetachedSignature:[OUT] Base64编码的PKCS#7不带原文签名结果；

**返回值**：

0：成功；

其它：失败。

## cfca\_verifyFileSignature\_PKCS7Detached

integer cfca\_verifyFileSignature\_PKCS7Detached(string strSignAlg,

string strSourceFilePath,

string strBase64PKCS7DetachedSignature,

string strBase64SignCertContent)

**描述:**

按指定的算法类型（SM2/RSA），验证文件PKCS#7不带原文签名。

**参数：**

string strSignAlg:[IN]算法类型，传入“SM2”或“RSA”，不区分大小写；

string strSourceFilePath:[IN] 待验签的文件原文路径，路径中不许有中文；

string strBase64PKCS7DetachedSignature:[IN] Base64编码的PKCS#7不带原文签名结果；

string strBase64SignCertContent:[OUT] 返回PKCS#7中的签名证书（Base64编码;

**返回值**：

0：成功；

其它：失败。

## cfca\_encryptDataToCMSEnvelope

integer cfca\_encryptDataToCMSEnvelope (string strAlgorithm,

string strPlainData,

string strBase64EncryptCert,

string strSymEncAlg,

string strBase64CMSEnvelope)

**描述:**

按指定的算法类型（SM2/RSA），把数据加密成CMS格式的数字信封。

**参数：**

string strAlgorithm:[IN]算法类型，传入“SM2”或“RSA”，不区分大小写；

string strPlainData:[IN] 待加密的明文字符串，需使用UTF8格式编码，以保持与其他系统兼容；

string strBase64EncryptCert:[IN] 用于加密的公钥证书内容（Base64编码，不带证书头、尾）；

string strSymEncAlg:[IN] 对称加密算法，传入“RC4”或“3DES”，不区分大小写。此参数仅在加密RSA数字信封时才起作用；SM2类型的数字信封对称加密默认使用SM4，忽略此参数；

string strBase64CMSEnvelope:[OUT] 加密后的Base64编码数字信封；

**返回值**：

0：成功；

其它：失败

## cfca\_decryptDataFromCMSEnvelope

integer cfca\_decryptDataFromCMSEnvelope (string strAlgorithm,

string strBase64CMSEnvelope,

string strPfxFilePath,

string strPfxPassWord,

string strPlainData)

**描述:**

按指定的算法类型（SM2/RSA），解密CMS格式的数字信封。

**参数：**

string strAlgorithm:[IN]算法类型，传入“SM2”或“RSA”，不区分大小写；

string strBase64CMSEnvelope:[IN] 待解密的Base64编码数字信封；

string strPfxFilePath:[IN]用于解密的软证书文件的路径，路径中不许有中文；

string strPfxPassWord:[IN]用于解密的软证书文件的密码；

string strPlainData:[OUT] 解密出的明文字符串；

**返回值**：

0：成功；

其它：失败

## cfca\_symEncryptFile

integer cfca\_symEncryptFile(string strSymEncAlgorithm,

string strPlainFilePath,

string strEncryptFilePath,

string strSymKey)

**描述**:

使用指定的加密算法对文件进行对称加密，并输出随机产生的对称秘钥。

注意：待加密的明文文件大小不能超过1G!

**参数**：

string strSymEncAlgorithm:[IN] 对称加密算法，传入“3DES\_CBC”或“SM4\_CBC”，不区分大小写；

string strPlainFilePath:[IN] 待加密的明文文件路径，路径中不许有中文；

string strEncryptFilePath:[IN] 加密后生成的密文路径，路径中不许有中文；

string strSymKey:[OUT] 十六进制编码的对称密钥；使用3DES\_CBC对称加密算法时，十六进制编码的对称密钥前16个字符为IV，后48个字符为key；使用SM4\_CBC对称加密算法时，十六进制编码的对称密钥前32个字符为IV，后32个字符为key;

**返回值**：

0：成功；

其它：失败

## cfca\_symDecryptFile

integer cfca\_symDecryptFile(string strSymEncAlgorithm,

string strEncryptFilePath,

string strSymKey,

string strPlainFilePath)

**描述**:

使用指定的算法对文件进行解密。

**参数**：

string strSymEncAlgorithm:[IN] 对称解密算法，传入“3DES\_CBC”或“SM4\_CBC”，不区分大小写；

string strEncryptFilePath:[IN] 待解密的密文路径，路径中不许有中文；

string strSymKey:[IN] 十六进制编码的对称密钥；使用3DES\_CBC对称加密算法时，十六进制编码的对称密钥前16个字符为IV，后48个字符为key；使用SM4\_CBC对称加密算法时，十六进制编码的对称密钥前32个字符为IV，后32个字符为key;

string strPlainFilePath:[IN] 解密后生成的明文文件路径，路径中不许有中文；

**返回值**：

0：成功；

其它：失败

## cfca\_verifyCertificate

integer cfca\_verifyCertificate(string strBase64CertContent,

integer nCertVerifyFlag,

string strTrustedCACertFilePath,

string strCRLFilePath)

**描述:**

验证证书有效性。

**参数：**

string strBase64CertContent:[IN] 待验证的公钥证书内容（Base64编码，不带证书头、尾）；

integer nCertVerifyFlag:[IN] 验证证书标识位。1:验证证书时间有效性；2:验证证书是否被吊销；4:验证证书链完整性。以上标识位可以组合使用，例如传入7就是全部都验证；

string strTrustedCACertFilePath:[IN] 可信根证书或中级证书的文件路径，路径中不许有中文。如果有多个证书文件，需要把多个文件路径以“|”为分隔，组合成一个字符串传入，例如:“D:/RootCert.cer|D:/ IntermediateCert.cer”。如果nCertVerifyFlag不包含4，此参数可以传入NULL；

string strCRLFilePath:[IN] 证书吊销列表CRL文件路径，路径中不许有中文。如果nCertVerifyFlag不包含2，此参数可以传入NULL；

**返回值**：

0：成功；

其它：失败

## cfca\_getCertificateInfo

integer cfca\_getCertificateInfo (string strBase64CertContent,

string strInfoType,

string strInfoContent)

**描述:**

根据传入的标识，获取证书的相关信息。

**参数：**

string strBase64CertContent:[IN] 公钥证书内容（Base64编码，不带证书头、尾）；

string strInfoType:[IN] 要获取的信息类型标识（不区分大小写）

“CertType”： 证书类型，返回“SM2”或“RSA”。

“SubjectDN”: 证书主题DN；

“SubjectCN”: 证书主题CN；

“IssuerDN”: 颁发者主题DN；

“SerialNumber”: 证书序列号；

“ValidFrom”: 有效起始日期；

“ValidTo”: 有效截止日期；

string strInfoContent:[OUT] 返回获取到的证书信息；

**返回值**：

0：成功；

其它：失败

## cfca\_getPublicCertFromPFX

integer cfca\_getPublicCertFromPFX (string strAlgorithm,

string strPfxFilePath,

string strPfxPassWord,

string strBase64CertContent)

**描述:**

从指定的软证书文件中，获取公钥证书。

**参数：**

string strAlgorithm:[IN] 算法标识，传入“SM2”或“RSA”，不区分大小写；

string strPfxFilePath:[IN] 用于提取公钥证书的软证书文件路径，路径中不许有中文;

如果pszAlgorithm指定了SM2算法，此处传入Base64编码的“.SM2”文件；如果pszAlgorithm指定了RSA算法，此处传入DER编码的“.pfx”文件；

string strPfxPassWord:[IN] 用于提取公钥证书的软证书文件的密码;

string strBase64CertContent:[OUT] 获取到的Base64编码公钥证书内容；

**返回值**：

0：成功；

其它：失败

## cfca\_calculateDataHash

integer cfca\_calculateDataHash (string strSourceData,

string strHashAlg,

string strHexHashData)

**描述:**

计算传入数据的哈希值，传出十六进制编码的哈希计算结果。

**参数：**

string strSourceData:[IN]待计算哈希的字符串，需使用UTF8格式编码，以保持与其他系统兼容；

string strHashAlg:[IN] 哈希算法，传入“MD5”、“SHA-1”、“SHA-256”、“SM3”，不区分大小写;

string strHexHashData:[OUT] 十六进制编码的哈希计算结果。

**返回值**：

0：成功；

其它：失败

## cfca\_calculateFileHash

integer cfca\_calculateFileHash (string strSourceFilePath,

string strHashAlg,

string strHexHashData)

**描述:**

计算传入文件的哈希值，传出十六进制编码的哈希计算结果。

**参数：**

string strSourceFilePath:[IN]待计算哈希的文件路径，路径中不许有中文；

string strHashAlg:[IN] 哈希算法，传入“MD5”、“SHA-1”、“SHA-256”、“SM3”，不区分大小写;

string strHexHashData:[OUT] 十六进制编码的哈希计算结果。

**返回值**：

0：成功；

其它：失败

# 提供程序

提供PHP调用的扩展库及PHP调用Demo.

PHP5.6库: libSADKExtension.so.3.4.0.1

PHP7.0库: libSADKExtension.so.3.4.0.1

PHP7.1库: libSADKExtension.so.3.4.0.1