

PHP 版服务器端工具包（Windows 版）

接口使用手册

（版本：3.0）

中国金融认证中心

2017 年 8 月 17 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	建立文档	2017/08/17	胡军华	

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 文档描述.....	1
2 版本需求.....	1
2.1 操作系统版本.....	1
2.2 PHP 版本	1
3 接口列表.....	1
3.1 SignData_PKCS1.....	2
3.2 SignData_PKCS7Detached	3
3.3 SignData_PKCS7Attached	4
3.4 VerifyDataSignature_PKCS1.....	5
3.5 VerifyDataSignature_PKCS7Detached	6
3.6 VerifyDataSignature_PKCS7Attached	7
3.7 GetSignSourceData.....	7
3.8 SignFile_PKCS7Detached	8
3.9 VerifyFileSignature_PKCS7Detached	9
3.10 EncryptDataToCMSEnvelope	9
3.11 DecryptDataFromCMSEnvelope	10
3.12 SymEncryptFile	11
3.13 SymDecryptFile.....	12
3.14 VerifyCertificate.....	13
3.15 GetCertificateInfo	14
3.16 GetPublicCertFromPFX	15
3.17 CalculateDataHash	16
3.18 CalculateFileHash	17
3.19 GetLastErrorDesc.....	17
4 提供程序.....	18
5 PHP 调用控件方法	18

1 文档描述

该文档主要描述 Windows 版 PHP 服务器端工具包接口的定义以及使用，帮助使用者了解接口的调用方式。

2 版本需求

2.1 操作系统版本

支持以下操作系统：

WinServer2008 64bit、

WinServer2012 64bit。

2.2 PHP 版本

支持以下版本的 PHP：

PHP5.6 系列；

PHP7.0 系列；

PHP7.1 系列。

3 接口列表

以 ATL COM 组件形式实现。

3.1 SignData_PKCS1

HRESULT SignData_PKCS1(BSTR bstrSignAlg,

 BSTR bstrSourceData,

 BSTR bstrPfxFilePath,

 BSTR bstrPfxPassWord,

 BSTR bstrHashAlg,

 BSTR* pbstrBase64PKCS1Signature)

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:

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

BSTR bstrSourceData:[IN]待签名的字符串，签名之前控件会将其转换为 UTF-8 编码；

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

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

BSTR bstrHashAlg:[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；

BSTR* pbstrBase64PKCS1Signature:[OUT, RETVAL] Base64 编码的 PKCS#1 签名结果；

3.2 SignData_PKCS7Detached

HRESULT SignData_PKCS7Detached(BSTR bstrSignAlg,

BSTR bstrSourceData,

BSTR bstrPfxFilePath,

BSTR bstrPfxPassWord,

BSTR bstrHashAlg,

BSTR* pbstrBase64PKCS7DetachedSignature)

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:

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

BSTR bstrSourceData:[IN]待签名的字符串，签名之前控件会将其转换为 UTF-8 编码；

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

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

BSTR bstrHashAlg:[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；

BSTR* pbstrBase64PKCS7DetachedSignature:[OUT, RETVAL] Base64 编码的 PKCS#7 不带原文签名结果；

3.3 SignData_PKCS7Attached

```
HRESULT SignData_PKCS7Attached(BSTR bstrSignAlg,  
  
                                BSTR bstrSourceData,  
  
                                BSTR bstrPfxFilePath,  
  
                                BSTR bstrPfxPassWord,  
  
                                BSTR bstrHashAlg,  
  
                                BSTR* pbstrBase64PKCS7AttachedSignature)
```

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:

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

BSTR bstrSourceData:[IN] 待签名的字符串，签名之前控件会将其转换为 UTF-8 编码；

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

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

BSTR bstrHashAlg:[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；

BSTR* pbstrBase64PKCS7AttachedSignature:[OUT, RETVAL] Base64 编码的

PKCS#7 不带原文签名结果;

3.4 VerifyDataSignature_PKCS1

HRESULT VerifyDataSignature_PKCS1(BSTR bstrSignAlg,
BSTR bstrSourceData,
BSTR bstrBase64CertContent,
BSTR bstrHashAlg,
BSTR bstrBase64PKCS1Signature,
VARIANT_BOOL* pbSuccess)

描述:

按指定的算法类型 (SM2/RSA), 验证 PKCS#1 签名。

参数:

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

BSTR bstrSourceData:[IN] 被签名的字符串, 验签之前控件会将其转换为 UTF-8 编码;

BSTR bstrBase64CertContent:[IN] 用于验签的公钥证书内容 (Base64 编码, 不带证书头、尾);

BSTR bstrHashAlg:[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;

BSTR bstrBase64PKCS1Signature:[IN] Base64 编码的 PKCS#1 签名结果;

VARIANT_BOOL* pbSuccess: [OUT, RETVAL] 验签结果, VARIANT_TRUE: 成功,
VARIANT_FALSE: 失败。

3.5 VerifyDataSignature_PKCS7Detached

HRESULT VerifyDataSignature_PKCS7Detached (BSTR bstrSignAlg,

BSTR bstrSourceData,

BSTR bstrBase64PKCS7DetachedSignature,

BSTR* pbstrBase64SignCertContent)

描述:

按指定的算法类型 (SM2/RSA), 验证 PKCS#7 不带原文签名。

参数:

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

BSTR bstrSourceData:[IN] 被签名的字符串, 验签之前控件会将其转换为 UTF-8 编码;

BSTR bstrBase64PKCS7DetachedSignature:[IN] Base64 编码的 PKCS#7 不带原文签名结果;

BSTR* pbstrBase64SignCertContent:[OUT, RETVAL] PKCS#7 中的签名证书 (Base64 编码);

3.6 VerifyDataSignature_PKCS7Attached

HRESULT VerifyDataSignature_PKCS7Attached (BSTR bstrSignAlg,
BSTR bstrBase64PKCS7AttachedSignature,
BSTR* pbstrBase64SignCertContent)

描述:

按指定的算法类型（SM2/RSA），验证 PKCS#7 带原文签名。

参数:

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

BSTR bstrBase64PKCS7AttachedSignature:[IN] Base64 编码的 PKCS#7 带原文签名结果；

BSTR* pbstrBase64SignCertContent:[OUT, RETVAL] PKCS#7 中的签名证书（Base64 编码）；

3.7 GetSignSourceData

HRESULT GetSignSourceData (BSTR bstrBase64PKCS7AttachedSignature,
BSTR* pbstrSourceData)

描述:

获取 P7 带原文签名结果中的签名原文。

参数:

BSTR bstrBase64PKCS7AttachedSignature:[IN] Base64 编码的 PKCS#7 带原文签

名结果;

BSTR* pbstrSourceData:[OUT, RETVAL] PKCS#7 中的签名原文;

3.8 SignFile_PKCS7Detached

HRESULT SignFile_PKCS7Detached (BSTR bstrSignAlg,

BSTR bstrSourceFilePath,

BSTR bstrPfxFilePath,

BSTR bstrPfxPassword,

BSTR bstrHashAlg,

BSTR* pbstrBase64PKCS7DetachedSignature)

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:

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

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

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

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

BSTR bstrHashAlg:[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;

BSTR pbstrBase64PKCS7DetachedSignature:[OUT, RETVAL] Base64 编码的 PKCS#7 不带原文签名结果;

3.9 VerifyFileSignature_PKCS7Detached

HRESULT VerifyFileSignature_PKCS7Detached(BSTR bstrSignAlg,
BSTR bstrSourceFilePath,
BSTR bstrBase64PKCS7DetachedSignature,
BSTR bstrBase64SignCertContent)

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:

BSTR bstrSignAlg:[IN] 算法类型, 传入 “SM2” 或 “RSA”, 不区分大小写;
BSTR bstrSourceFilePath:[IN] 待验签的文件原文路径, 路径中不许有中文;
BSTR bstrBase64PKCS7DetachedSignature:[IN] Base64 编码的 PKCS#7 不带原文签名结果;
BSTR bstrBase64SignCertContent:[OUT, RETVAL] 返回 PKCS#7 中的签名证书 (Base64 编码);

3.10 EncryptDataToCMSEnvelope

HRESULT EncryptDataToCMSEnvelope (BSTR bstrAlgorithm,

BSTR bstrPlainData,
 BSTR bstrBase64EncryptCert,
 BSTR bstrSymEncAlg,
 BSTR* pstrBase64CMSEnvelope)

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:

BSTR bstrAlgorithm:[IN] 算法类型，传入“SM2”或“RSA”，不区分大小写；
 BSTR bstrPlainData:[IN] 待加密的明文字符串，加密之前控件会将其转换为 UTF-8 编码；

BSTR bstrBase64EncryptCert:[IN] 用于加密的公钥证书内容（Base64 编码，不带证书头、尾）；

BSTR bstrSymEncAlg:[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；

BSTR* pstrBase64CMSEnvelope:[OUT, RETVAL] 加密后的 Base64 编码数字信封；

3.11 DecryptDataFromCMSEnvelope

HRESULT DecryptDataFromCMSEnvelope (BSTR bstrAlgorithm,
 BSTR bstrBase64CMSEnvelope,

BSTR bstrPfxFilePath,
BSTR bstrPfxPassWord,
BSTR* pbstrPlainData)

描述:

按指定的算法类型（SM2/RSA），解密 CMS 格式的数字信封。

参数:

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

BSTR bstrBase64CMSEnvelope:[IN] 待解密的 Base64 编码数字信封；

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

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

BSTR* pbstrPlainData:[OUT, RETVAL] 解密出的明文字符串；

3.12 SymEncryptFile

HRESULT SymEncryptFile (BSTR bstrSymEncAlgorithm,

BSTR bstrPlainFilePath,

BSTR bstrEncryptFilePath,

BSTR* pbstrSymKey)

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密钥。

注意：待加密的明文文件大小不能超过 1G！

参数:

BSTR bstrSymEncAlgorithm:[IN]对称加密算法,传入“3DES_CBC”或“SM4_CBC”,不区分大小写;

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

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

BSTR* pbstrSymKey:[OUT, RETVAL] 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。

3.13 SymDecryptFile

```
HRESULT SymDecryptFile (BSTR bstrSymEncAlgorithm,  
                        BSTR bstrEncryptFilePath,  
                        BSTR bstrSymKey,  
                        BSTR bstrPlainFilePath,  
                        VARIANT_BOOL* pbSuccess);
```

描述:

使用指定的算法对文件进行解密。

参数:

BSTR bstrSymEncAlgorithm:[IN]对称解密算法,传入“3DES_CBC”或“SM4_CBC”,不区分大小写;

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

BSTR bstrSymKey:[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。

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

VARIANT_BOOL* pbSuccess:[OUT, RETVAL] 验签结果，VARIANT_TRUE: 成功，VARIANT_FALSE: 失败。

3.14 VerifyCertificate

HRESULT VerifyCertificate(BSTR bstrBase64CertContent,

INT nCertVerifyFlag,

BSTR bstrTrustedCACertFilePath,

BSTR bstrCRLFilePath,

VARIANT_BOOL* pbSuccess)

描述:

验证证书有效性。

参数:

BSTR bstrBase64CertContent:[IN] 待验证的公钥证书内容（Base64 编码，不带证书头、尾）；

INT 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；

BSTR bstrTrustedCACertFilePath:[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；

BSTR bstrCRLFilePath:[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；

VARIANT_BOOL* pbSuccess:[OUT, RETVAL] 验证结果，VARIANT_TRUE: 成功，VARIANT_FALSE: 失败。

3.15 GetCertificateInfo

HRESULT GetCertificateInfo (BSTR bstrBase64CertContent,
BSTR bstrInfoType,
BSTR* pbstrInfoContent)

描述:

根据传入的标识，获取证书的相关信息。

参数:

BSTR bstrBase64CertContent:[IN] 公钥证书内容 (Base64 编码，不带证书头、尾)；

BSTR bstrInfoType:[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;

BSTR* pbstrInfoContent:[OUT, RETVAL] 返回获取到的证书信息;

3.16 GetPublicCertFromPFX

HRESULT GetPublicCertFromPFX (BSTR bstrAlgorithm,
BSTR bstrPfxFilePath,
BSTR bstrPfxPassWord,
BSTR* pbstrBase64CertContent)

描述:

从指定的软证书文件中，获取公钥证书。

参数:

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

BSTR bstrPfxFilePath:[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；
BSTR bstrPfxPassWord:[IN] 用于提取公钥证书的软证书文件的密码；
BSTR* pbstrBase64CertContent:[OUT, RETVAL] 获取到的 Base64 编码公钥证书内容。

3.17 CalculateDataHash

HRESULT CalculateDataHash (BSTR bstrSourceData,
BSTR bstrHashAlg,
BSTR* pbstrHexHashData)

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。计算 SM3 哈希时，不带 Z 值。

参数:

BSTR bstrSourceData:[IN]待计算哈希的字符串，计算哈希之前控件会将其转换为 UTF-8 编码；

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

BSTR* pbstrHexHashData:[OUT, RETVAL] 十六进制编码的哈希计算结果。

3.18 CalculateFileHash

HRESULT CalculateFileHash (BSTR bstrSourceFilePath,
BSTR bstrHashAlg,
BSTR* pbstrHexHashData)

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。计算 SM3 哈希时，不带 Z 值。

参数:

BSTR bstrSourceFilePath:[IN]待计算哈希的文件路径，路径中不许有中文；
BSTR bstrHashAlg:[IN] 哈希算法，传入“MD5”、“SHA-1”、“SHA-256”、“SM3”，不区分大小写；
BSTR* pbstrHexHashData:[OUT,RETVAL] 十六进制编码的哈希计算结果。

3.19 GetLastErrorDesc

HRESULT GetLastErrorDesc(BSTR* pbstrErrorDesc)

描述:

获得最近一次调用接口导致发生错误的描述信息。此函数会根据不同的操作系统语言（简体中文/美国英语）来本地化错误描述。

参数:

BSTR* pbstrErrorDesc: [OUT, RETVAL]错误描述信息

4 提供程序

提供 DLL 程序、PHP 调用 Demo、C#调用 Demo。

DLL 名称: CryptoKit.Standard.x86.dll

CryptoKit.Standard.x64.dll

5 PHP 调用控件方法

1、 下载 PHP 并修改配置文件

从 PHP 官网下载所需的 Windows 版 PHP 程序，下载完成之后解压。将 PHP 文件夹下的 php.ini-production 文件重命名为 php.ini。

将重命名后的 php.ini 文件内的 “; extension_dir = "ext"” 行，修改为 “extension_dir = "./ext"”。然后，在 php.ini 文件末尾，添加以下两行内容：

```
[COM_DOT_NET]
```

```
extension=php_com_dotnet.dll
```

2、注册 COM 控件

根据所用 PHP 版本（x86 或者 x64）注册对应版本的 COM 组件。x86 版 PHP 使用 x86 版 COM，x64 版 PHP 使用 x64 版 COM 组件。

COM 组件注册命令: regsvr32 CryptoKit.Standard.x86.dll 或

regsvr32 CryptoKit.Standard.x64.dll

3、运行 Demo

将 Demo 拷贝到 PHP 程序所在路径下，然后运行以下命令执行。

```
Php SADKTest.php
```