ICS 35.030
CCS L 80

中华人民共和国密码行业标准

GM/T 0020—2023
代替 GM/T 0020—2012

证书应用综合服务接口规范

Certificate application integrated service interface specification

2023-12-04 发布　　2024-06-01 实施

国家密码管理局　发布

目　　次

前　言

本文件按照 GB/T 1.1—2020《标准化工作导则　第 1 部分：标准化文件的结构和起草规则》的规定起草。

本文件代替 GM/T 0020—2012《证书应用综合服务接口规范》，与 GM/T 0020—2012 相比，除结构调整和编辑性改动外，主要技术变化如下：

a) 删除了术语“数字证书”（见 2012 年版的 3.1）；

b) 增加了 Base64 格式数据的更明确描述（见 6.5）；

c) 增加了接口“证书登出 SOF_Logout”（见 7.1.35）、“证书登录状态检测 SOF_IsLogin”（见 7.1.36）；

d) 增加了接口“数据摘要 SOF_HashData”（见 7.1.31、7.2.37、7.3.38）“文件摘要 SOF_HashFile”（见 7.1.32、7.2.38、7.3.39）“摘要值签名 SOF_SignHashData”（见 7.1.33、7.2.39、7.3.40）“摘要值验签 SOF_VerifySignedHashData”（见 7.1.34、7.2.40、7.3.41）；

e) 删除了接口“SOF_EncryptFile”（见 2012 年版的 7.1.23）“SOF_DecryptFile”（见 2012 年版的 7.1.24）；

f) 增加了“客户端 JavaScript 脚本接口”（见 7.4）。

请注意本文件的某些内容可能涉及专利。本文件的发布机构不承担识别专利的责任。

本文件由密码行业标准化技术委员会提出并归口。

本文件起草单位：北京数字认证股份有限公司、格尔软件股份有限公司、北京海泰方圆科技股份有限公司、上海市数字证书认证中心有限公司、无锡江南信息安全工程技术中心、中电科网络安全科技股份有限公司、长春吉大正元信息技术股份有限公司、兴唐通信科技有限公司、山东得安信息技术有限公司、北京国脉信安科技有限公司、国家密码管理局商用密码检测中心、中国电子技术标准化研究院。

本文件主要起草人：刘伟、赵永省、刘平、刘蕾、李述胜、郑强、谭武征、蒋红宇、柳增寿、许涛、寇建波、赵丽丽、王妮娜、马洪富、孔凡玉、袁峰、罗鹏、肖秋林、张绍博、上官晓丽、蔡一鸣、黄晶晶。

本文件及其所代替文件的历次版本发布情况为：

——2012 年首次发布版为 GM/T 0020—2012；

——本次是第一次修订。

引　言

本文件依托于GM/T 0019《通用密码服务接口规范》，为应用层规定了统一的高级密码服务接口。

证书应用综合服务接口为应用系统提供简洁、易用的证书应用接口，屏蔽了各类密码设备（服务器密码机和智能密码钥匙等）的设备差异性，以及各类密码设备的密码应用接口之间的差异性，实现应用与密码设备无关性，可简化应用开发的复杂性。证书应用综合服务接口分成客户端服务接口和服务器端服务接口两类，可满足B/S和C/S等多种架构的应用系统的调用需求，有利于密码服务接口产品的开发，有利于应用系统在密码服务过程中的集成和实施，有利于实现各应用系统的互联互通。

证书应用综合服务接口规范

1 范围

本文件规定了面向证书应用的综合服务接口。

本文件适用于公钥密码应用技术体系下密码应用服务产品的开发，密码应用支撑平台的研制及检测，也可用于指导直接使用密码设备和密码服务的应用系统的集成和开发。

2 规范性引用文件

下列文件中的内容通过文中的规范性引用而构成本文件必不可少的条款。其中，注日期的引用文件，仅该日期对应的版本适用于本文件；不注日期的引用文件，其最新版本(包括所有的修改单)适用于本文件。

GB/T 25061—2020 信息安全技术 XML 数字签名语法与处理规范

GM/T 0006 密码应用标识规范

GM/T 0009 SM2 密码算法使用规范

GM/T 0010 SM2 密码算法加密签名消息语法规范

GM/T 0015 基于 SM2 密码算法的数字证书格式规范

GM/T 0019 通用密码服务接口规范

GM/Z 4001 密码术语

PKCS #1 RSA 加密标准(RSA Cryptography Standard)

PKCS #7 加密消息语法标准(Cryptographic Message Syntax Standard)

IETF RFC 3275 (可扩展标记语言)XML 签名语法和处理[(Extensible Markup Language)XML-Signature Syntax and Processing]

IETF RFC 4648 Base16、Base32 和 Base64 数据编码(The Base16, Base32, and Base64 Data Encodings)

3 术语和定义

GM/Z 4001 界定的以及下列术语和定义适用于本文件。

3.1

用户密钥 user key

存储在设备内部的用于应用密码运算的非对称密钥对。

注：用户密钥包含签名密钥对和加密密钥对。

3.2

容器 container

密码设备中用于保存密钥所划分的唯一性存储空间。

4 缩略语

下列缩略语适用于本文件。

API:应用程序接口/应用接口(Application Program Interface)
CA:证书认证机构(Certification Authority)
CN:通用名(Common Name)
CRL:证书撤销列表(Certificate Revocation List)
CSP:加密服务提供者(Cryptographic Service Provider)
DER:可区分编码规则(Distinguished Encoding Rules)
DN:可识别名(Distinguished Name)
LDAP:轻量级目录访问协议(Lightweight Directory Access Protocol)
OID:对象标识符(Object Identifier)
PKCS:公钥密码标准(the Public-Key Cryptography Standard)

5 算法标识和数据结构

5.1 标识定义

本文件所使用常量、各类算法标识和证书解析标识符合 GM/T 0006。

5.2 数据结构定义

本文件对应的接口处理的数据分为两种类型。

数据类型 A:一种基于公钥的加密或签名结构,当公钥算法为 RSA 时,数据的结构应符合 PKCS#1 v1.5;当公钥算法为 SM2 时,数据的结构应符合 GM/T 0009。

数据类型 B:一种基于证书的加密或签名结构,当证书的公钥算法为 RSA 时,消息的结构应符合 PKCS#7;当证书的公钥算法为 SM2 时,消息的结构应符合 GM/T 0010。

6 证书应用综合服务接口定位、分类和要求

6.1 证书应用综合服务接口在公钥密码应用技术体系框架中的位置

公钥密码应用技术体系框架由应用层、典型密码应用支撑层、通用密码应用支撑层、基础设施安全支撑平台、密码设备服务层组成。证书应用综合服务接口属于通用密码应用支撑层,是中间件,为典型密码应用支撑层和应用层提供密码服务。其功能主要包括:负责完成与密码设备的安全连接;实现基于数字证书的身份认证,从证书中获取有关信息,实现授权管理、访问控制等安全机制;负责具体与密码设备交互实现具体的密码运算;将数据按照 GM/T 0010 格式进行封装,实现数据封装格式与应用系统无关性,实现应用系统互联互通和信息共享。证书应用综合服务接口规范在公钥密码应用技术框架内的位置参见 GM/T 0094—2020 的第 4 章。

6.2 证书应用综合服务接口分类

证书应用综合服务接口包括客户端服务接口和服务器端服务接口两类。它位于典型密码应用支撑层和密码设备服务层之间,既可以被典型密码应用支撑层调用也可以被应用层直接调用,向上层提供证书信息解析、基于数字证书身份认证、信息的机密性、完整性和不可否认性等密码服务。

6.3 客户端服务接口

客户端服务接口采用支持多种编程语言的 COM 组件和客户端常使用的 JavaScript 脚本语言为例描述,其适用于客户端程序调用,接口的形态包括动态库、控件、插件、扩展等。

客户端服务接口一般通过调用 GM/T 0016 或 GM/T 0019 来实现，主要功能包括：配置管理、证书解析、签名与验证、加密与解密数字信封、XML 数据签名与验证等。

6.4 服务器端服务接口

服务器端服务接口采用支持多种编程语言的 COM 组件和服务端常使用的 Java 语言为例描述，其适用于服务器端程序调用，接口的形态包括 COM 组件、JAR 包等，支持主流操作系统。

服务器端服务接口一般通过调用 GM/T 0018 或 GM/T 0019 来实现，功能基本与客户端服务接口相对应，主要包括：配置管理、证书解析、签名与验证、加密与解密数字信封、XML 数据签名与验证、时间戳等。

6.5 数据格式要求

本文所涉及的数字证书格式应符合 GM/T 0015。

本文件所描述的 Base64 编码格式参照 IETF RFC 4648 中 Base 64 Encoding 部分。

在没有特殊说明的情况下，本文件中字符串采用 UTF-8 编码，其格式参照 IETF RFC 3629。

本文件所定义接口返回的错误码或抛出的异常信息应符合附录 A，典型部署模型可参考附录 B，集成示例可参考附录 C。

本文件以 COM 组件为例进行接口描述时，所用到的数据类型说明见表 1。

表 1 COM 组件数据类型说明

类型	说明
BSTR	字符串类型，不同的开发语言应采取对应的类型定义，如：char*、CString、java.lang.String 等
LONG	32 位整数
SHORT	16 位整数
BOOL	布尔类型，其取值范围是 TRUE 和 FALSE，其中 TRUE 表示真值，FALSE 表示假值
空串	长度等于 0 的字符串
非空	长度大于 0 的字符串

7 证书应用综合服务接口定义

7.1 客户端 COM 组件接口

7.1.1 客户端 COM 组件接口综述

客户端 COM 组件接口提供配置管理、证书解析、签名与验证、加密与解密数字信封、XML 数据签名与验证等功能，共包含 35 个接口。表 2 列出了客户端 COM 组件接口列表，7.1.2～7.1.36 给出了接口详细定义。

表 2 客户端 COM 组件接口列表

序号	章条号	接口名称	功能
1	7.1.2	SOF_GetVersion	获取接口的版本号
2	7.1.3	SOF_SetSignMethod	设置签名算法

表 2　客户端 COM 组件接口列表(续)

序号	章条号	接口名称	功能
3	7.1.4	SOF_GetSignMethod	获得签名算法
4	7.1.5	SOF_SetEncryptMethod	设置加密算法
5	7.1.6	SOF_GetEncryptMethod	获得加密算法
6	7.1.7	SOF_GetUserList	获得证书列表
7	7.1.8	SOF_ExportUserCert	导出用户签名证书
8	7.1.9	SOF_Login	证书登录
9	7.1.10	SOF_GetPinRetryCount	获取证书口令剩余重试次数
10	7.1.11	SOF_ChangePassWd	修改证书口令
11	7.1.12	SOF_ExportExChangeUserCert	导出用户加密证书
12	7.1.13	SOF_GetCertInfo	获得证书信息
13	7.1.14	SOF_GetCertInfoByOid	获得证书扩展信息
14	7.1.15	SOF_GetDeviceInfo	获得设备信息
15	7.1.16	SOF_ValidateCert	验证证书有效性
16	7.1.17	SOF_SignData	数据签名
17	7.1.18	SOF_VerifySignedData	验证数据签名
18	7.1.19	SOF_SignFile	文件签名
19	7.1.20	SOF_VerifySignedFile	验证文件签名
20	7.1.21	SOF_EncryptData	数据加密
21	7.1.22	SOF_DecryptData	数据解密
22	7.1.23	SOF_SignMessage	消息签名
23	7.1.24	SOF_VerifySignedMessage	验证消息签名
24	7.1.25	SOF_GetInfoFromSignedMessage	解析消息签名
25	7.1.26	SOF_SignDataXML	XML数字签名
26	7.1.27	SOF_VerifySignedDataXML	验证XML数字签名
27	7.1.28	SOF_GetXMLSignatureInfo	解析XML签名数据
28	7.1.29	SOF_GenRandom	产生随机数
29	7.1.30	SOF_GetLastError	获取最新的错误信息
30	7.1.31	SOF_HashData	计算数据摘要
31	7.1.32	SOF_HashFile	计算文件摘要
32	7.1.33	SOF_SignHashData	摘要数据签名
33	7.1.34	SOF_VerifySignedHashData	验证摘要值签名
34	7.1.35	SOF_Logout	证书登出
35	7.1.36	SOF_IsLogin	证书登录状态检测

7.1.2 获取接口版本信息 SOF_GetVersion

原型： BSTR SOF_GetVersion()
描述： 获取接口的版本号。
参数： 无
返回值：

版本号	成功，返回接口版本号
空串	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.1.3 设置签名算法 SOF_SetSignMethod

原型： LONG SOF_SetSignMethod(LONG SignMethod)
描述： 设置接口在签名和验签运算时使用的签名算法。
参数：
返回值：

SignMethod		签名算法标识，应符合GM/T 0006
返回值：	SOR_OK	成功
	其他	失败，返回符合表A.1定义的错误代码

7.1.4 获得签名算法 SOF_GetSignMethod

原型： LONG SOF_GetSignMethod()
描述： 获得接口当前使用的签名算法。
参数： 无
返回值：

签名算法标识	成功，返回接口当前使用的签名算法标识
0	接口当前没有设置签名算法

7.1.5 设置加密算法 SOF_SetEncryptMethod

原型： LONG SOF_SetEncryptMethod(LONG EncryptMethod)
描述： 设置接口进行数据加密时使用的对称算法。
参数：
返回值：

EncryptMethod		对称加密算法标识，应符合GM/T 0006，本接口可支持不带附加认证数据的加解密算法
返回值：	SOR_OK	成功
	其他	失败，返回表A.1定义的错误代码

7.1.6 获得加密算法 SOF_GetEncryptMethod

原型： LONG SOF_GetEncryptMethod()
描述： 获得接口当前使用的对称加解密算法。
参数： 无
返回值：

加密算法标识	成功，返回接口当前使用的加密算法标识
0	接口当前没有设置加密算法

7.1.7 获得证书列表 SOF_GetUserList

原型： BSTR SOF_GetUserList()
描述： 取得当前已安装证书的用户列表。

参数： 无

返回值：

证书列表	成功，返回证书列表字符串数据，格式为：用户名1‖CertID1&&&用户名2‖CertID2&&&…，根据证书应用的策略不同得到不同的证书列表返回值。在证书列表中，用户名代表证书的CN项内容，CertID是证书唯一标识，其格式由实现者自定义，宜包含容器名、应用名、设备序列号等项。通过CertID可以找到唯一的签名证书、加密证书，并使用对应的用户密钥
空串	失败或当前不存在证书用户列表，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.1.8 导出用户签名证书 SOF_ExportUserCert

原型： BSTR SOF_ExportUserCert(BSTR CertID)

描述： 根据证书唯一标识，获取Base64编码的签名证书字符串。

参数：

	CertID	证书唯一标识

返回值：

签名证书	成功，返回Base64编码的签名证书字符串
空串	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.1.9 证书登录 SOF_Login

原型： BOOL SOF_Login(BSTR CertID, BSTR PassWd)

描述： 校验证书口令，进行证书认证。

参数：

	CertID	证书唯一标识
	PassWd	证书口令

返回值：

TRUE	成功
FALSE	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

注：证书登录成功后，表示该证书用户已拥有私钥使用权限，即登录状态。能正常调用签名、解密等需要私钥使用权限的接口，获得计算结果。

7.1.10 获取证书口令剩余重试次数 SOF_GetPinRetryCount

原型： LONG SOF_GetPinRetryCount(BSTR CertID)

描述： 获取证书口令的剩余重试次数。

参数：

	CertID	证书唯一标识

返回值： 剩余口令重试次数，当重试次数小于或等于0时表示证书口令已被锁定

7.1.11 修改证书口令 SOF_ChangePassWd

原型： BOOL SOF_ChangePassWd(BSTR CertID, BSTR OldPassWd,BSTR NewPassWd)

描述： 修改设备的用户认证口令。

参数：

	CertID	证书唯一标识
	OldPassWd	旧证书口令
	NewPassWd	新证书口令

返回值：

TRUE	修改证书口令成功

	FALSE	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.1.12 导出用户加密证书 SOF_ExportExChangeUserCert

原型：	BSTR SOF_ExportExChangeUserCert(BSTR CertID)	
描述：	根据证书唯一标识，获取Base64编码的加密(交换)证书字符串。	
参数：	CertID	证书唯一标识
返回值：	加密证书	成功，返回Base64编码的加密(交换)证书字符串
	空串	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.1.13 获得证书信息 SOF_GetCertInfo

原型：	BSTR SOF_GetCertInfo(BSTR Base64Cert, SHORT Type)	
描述：	获取证书内指定类型的信息。	
参数：	Base64Cert	Base64编码的证书字符串
	Type	证书解析标识，应符合GM/T 0006
返回值：	证书信息	成功，返回证书内指定类型的信息
	空串	失败或证书中不存在该项内容

7.1.14 获得证书扩展信息 SOF_GetCertInfoByOid

原型：	BSTR SOF_GetCertInfoByOid(BSTR Base64Cert, BSTR Oid)	
描述：	根据OID获取证书私有扩展项信息。	
参数：	Base64Cert	Base64编码的证书字符串
	Oid	私有扩展对象ID，如“1.2.156.xxx”
返回值：	证书扩展信息	成功，返回证书私有扩展项OID对应的信息
	空串	失败或证书中不存在该私有扩展项

7.1.15 获得设备信息 SOF_GetDeviceInfo

原型：	BSTR SOF_GetDeviceInfo(BSTR CertID, LONG Type)	
描述：	根据证书唯一标识和类型代码获得设备信息。	
参数：	CertID	证书唯一标识
	Type	设备信息的类型，应符合GM/T 0006
返回值：	设备信息	成功，返回Type对应的设备信息
	空串	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.1.16 验证证书有效性 SOF_ValidateCert

原型：	LONG SOF_ValidateCert(BSTR Base64Cert)	
描述：	验证证书有效性。	
参数：	Base64Cert	Base64编码的证书字符串
返回值：	SAR_OK	验证成功
	其他	验证失败，失败原因应符合附录A中表A.1中的错误代码范围0X0B000500-0X0B000505

注：基本的证书验证策略需包括：

a) 验证CA信任列表，各层都要进行签名验证；

b) 各层证书的有效期验证；

c) 各层证书的吊销状态。在特殊情况下（如：网络条件不允许），证书的吊销状态可采取灵活方式，由应用系统各层内部维护一个吊销列表，在证书登录认证时应用该吊销列表。验证证书有效性也可采取代理验证方式。

7.1.17 数据签名 SOF_SignData

原型： BSTR SOF_SignData(BSTR CertID, BSTR InData)

描述： 对字符串数据进行数字签名，返回Base64编码的数据类型A签名结果。

参数：

	CertID	证书唯一标识
	InData	签名原文

返回值：

签名值	成功，返回Base64编码的签名结果
空串	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

注：本接口在登录状态下才能成功返回签名值，否则返回空串。

7.1.18 验证数据签名 SOF_VerifySignedData

原型： BOOL SOF_VerifySignedData(BSTR Base64Cert, BSTR InData, BSTR SignValue)

描述： 验证数数据签名，签名值格式为Base64编码的数据类型A。

参数：

	Base64Cert	Base64编码的签名者证书字符串
	InData	签名原文
	SignValue	Base64编码的签名值

返回值：

TRUE	成功
FALSE	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.1.19 文件签名 SOF_SignFile

原型： BSTR SOF_SignFile(BSTR CertID, BSTR InFile)

描述： 根据文件全路径，对指定文件进行数字签名，返回Base64编码的数据类型A签名结果。

参数：

	CertID	证书唯一标识
	InFile	原文文件全路径（包含路径＋文件名）

返回值：

签名值	成功，返回Base64编码的签名值
空串	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

注：本接口在登录状态下才能成功返回签名值，否则返回空串。

7.1.20 验证文件签名 SOF_VerifySignedFile

原型： BOOL SOF_VerifySignedFile(BSTR Base64Cert, BSTR InFile, BSTR SignValue)

描述： 验证文件的数字签名，签名值格式为Base64编码的数据类型A。

参数：

	Base64Cert	Base64编码的签名者证书
	InFile	原文文件全路径（包含路径＋文件名）
	SignValue	Base64编码的签名值

返回值：

TRUE	成功

	FALSE	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.1.21 数据加密 SOF_EncryptData

原型：	BSTR SOF_EncryptData(BSTR Base64Cert, BSTR InData)	
描述：	数字信封加密，加密过程为使用临时产生的对称密钥加密数据，然后使用数字证书的公钥加密对称密钥，返回Base64编码格式的数据类型B密文数据。	
参数：	Base64Cert	Base64编码的数据接收者的加密证书，如使用多个证书对数据加密，证书之间用&&&作为分隔符连接
	InData	待加密的明文
返回值：	密文数据	成功，返回Base64编码的密文数据
	空串	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.1.22 数据解密 SOF_DecryptData

原型：	BSTR SOF_DecryptData(BSTR CertID, BSTR InData)	
描述：	使用证书对应的私钥解密数字信封，密文数据格式为Base64编码的数据类型B。	
参数：	CertID	证书唯一标识
	InData	待解密的Base64编码的密文数据
返回值：	明文数据	成功，返回解密后的明文
	空串	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

注：本接口在登录状态下才能解密得到明文，否则返回空串。

7.1.23 消息签名 SOF_SignMessage

原型：	BSTR SOF_SignMessage(SHORT Flag, BSTR CertID, BSTR InData)	
描述：	对字符串数据进行消息签名，返回Base64编码的数据类型B签名结果。	
参数：	Flag	是否为Detached的标识，取值范围： a） 1表示Detached，即不带原文； b） 0表示Attached，即带原文
	CertID	证书唯一标识
	InData	消息原文
返回值：	消息签名值	成功，返回Base64编码的签名值
	空串	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

注：签名结果包含：原文（可选）、签名者证书、签名算法和签名值。
本接口在登录状态下才能返回签名值，否则返回空串。

7.1.24 验证消息签名 SOF_VerifySignedMessage

原型：	BOOL SOF_VerifySignedMessage(BSTR SignedMessage, BSTR InData)	
描述：	验证消息签名，签名值格式为Base64编码的数据类型B。	
参数：	SignedMessage	Base64编码的签名值
	InData	消息原文，如果签名结果中包含原文，则本参数传空串
返回值：	TRUE	成功

	FALSE	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.1.25 解析消息签名 SOF_GetInfoFromSignedMessage

原型：BSTR SOF_GetInfoFromSignedMessage(BSTR SignedMessage,SHORT Type)

描述：解析消息签名值中的信息，包括：原文、签名值、签名证书等，消息签名值格式为Base64编码的数据类型B。

参数：	SignedMessage	Base64编码的签名值
	Type	获取的信息类型，取值范围： a） 1解析出原文； b） 2解析出Base64编码的签名者证书； c） 3解析出Base64编码的签名值
返回值：	解析结果	成功，返回解析出的对应信息
	空串	失败或不存在该项

7.1.26 XML数据签名 SOF_SignDataXML

原型：BSTR SOF_SignDataXML(BSTR CertID, BSTR InData)

描述：对XML数据进行数字签名，证书为RSA算法时签名结果符合RFC3275，证书为SM2算法时签名结果符合GB/T 25061。

参数：	CertID	证书唯一标识
	InData	XML格式的签名原文
返回值：	XML签名结果	成功，返回XML签名结果
	空串	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

注：本接口在登录状态下才能返回签名值，否则返回空串。

缺省配置和参数：

a） 采用封皮签名，签名前需对签名原文做规范化；

b） 规范化方法采用带注释的XML规范化1.1，证书为RSA算法时标识符为http://www.w3.org/2006/12/xml-c14n11#WithComment，证书为SM2算法时标识符为http://127.0.0.1/2006/12/xml—c14n11#WithComments。

7.1.27 验证XML数字签名 SOF_VerifySignedDataXML

原型：BOOL SOF_VerifySignedDataXML(BSTR XMLSignedData)

描述：验证XML签名。

参数：	XMLSignedData	XML签名结果
返回值：	TRUE	成功
	FALSE	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.1.28 解析XML签名数据 SOF_GetXMLSignatureInfo

原型：BSTR SOF_GetXMLSignatureInfo(BSTR XMLSignedData, SHORT Type)

描述：解析XML签名数据，获取签名值、XML原文、证书和相关算法等信息。

参数：	XMLSignedData	XML签名结果

	Type	待解析的参数类型，取值范围： a) 1解析出XML原文； b) 2解析出摘要值； c) 3解析出签名值； d) 4解析出签名证书； e) 5解析出摘要算法； f) 6解析出签名算法
返回值：	XML信息	成功，返回根据Type解析出各项对应的信息
	空串	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

注：XML签名缺省配置和参数值样例：

a) XML原文取自Object元素；
b) 摘要值取自DigestValue元素，形如http://127.0.0.1/2001/04/xmldsig-more#sm3；
c) 签名值取自SignatureValue元素，为BASE64编码；
d) 签名证书取自X509Data元素，为BASE64编码；
e) 摘要算法取自DigestMethod元素的Algorithm属性，形如http://127.0.0.1/2001/04/xmldsig-more#sm3；
f) 签名算法取自SignatureMethod元素的Algorithm属性，形如http://127.0.0.1/2001/04/xmldsig-more#sm2-sm3。

7.1.29 产生随机数 SOF_GenRandom

原型：	BSTR SOF_GenRandom(LONG RandomLen)	
描述：	产生指定长度的随机数。	
参数：	RandomLen	待产生随机数的长度(字节数)
返回值：	随机数	成功，返回Base64编码的随机数值
	空串	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.1.30 获取最新的错误信息 SOF_GetLastError

原型：	LONG SOF_GetLastError()
描述：	获取最新的错误代码。
参数：	无
返回值：	错误代码，应符合表A.1错误代码表

7.1.31 计算数据摘要 SOF_HashData

原型：	BSTR SOF_HashData(LONG HashAlg, BSTR InData,BSTR SignCert,BSTR UserID)	
描述：	计算数据摘要。如果是SM3算法，当SignCert和UsreID为空时，只计算数据的摘要值，当SignCert和UsreID值不为空时，应按照GM/T 0009规定的预处理过程计算，摘要值可以作为SM2签名的输入。	
参数：	HashAlg	摘要算法，应符合GM/T 0006
	Indata	原始数据
	SignCert	Base64编码的签名者证书，摘要算法为SM3时有效，如不需要传空串
	UserID	签名者用户ID，摘要算法为SM3时有效，如果SignCert参数为空，则本参数无意义

返回值：	摘要值	成功，返回Base64编码的数据摘要值
	空串	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.1.32 计算文件摘要 SOF_HashFile

原型： BSTR SOF_HashFile(LONG HashAlg, BSTR InFile,BSTR SignCert,BSTR UserID)

描述： 计算文件数据摘要，如果是SM3算法，当SignCert和UsreID为空时，只计算数据的摘要值，当SignCert和UsreID值不为空时，应按照GM/T 0009规定的预处理过程计算，摘要值可以作为SM2签名的输入。

参数：	HashAlg	摘要算法，应符合GM/T 0006
	InFile	文件全路径(包含路径＋文件名)
	SignCert	Base64编码的签名者证书，摘要算法为SM3时有效，如不需要传空串
	UserID	签名者用户ID，摘要算法为SM3时有效，如果SignCert参数为空，则本参数无意义
返回值：	摘要值	成功，返回Base64编码的文件摘要值
	空串	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.1.33 摘要数据签名 SOF_SignHashData

原型： BSTR SOF_SignHashData(BSTR CertID, BSTR Base64HashData, LONG HashAlg)

描述： 对数据摘要值签名，返回Base64编码的数据类型A签名结果。Base64HashData参数一般是SOF_HashData或SOF_HashFile函数的计算结果。如果是SM2签名，SM3算法的摘要值应按照GM/T 0009规定的预处理过程计算。

参数：	CertID	证书唯一标识
	Base64HashData	Base64编码的摘要值
	HashAlg	摘要算法，应符合GM/T 0006
返回值：	签名值	成功，返回Base64编码的签名值
	空串	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

注：本接口在登录状态下才能成功返回签名值，否则返回空串。

7.1.34 数据摘要签名验证 SOF_VerifySignedHashData

原型： BOOL SOF_VerifySignedHashData(BSTR Base64Cert,
BSTR Base64HashData,
BSTR SignValue, LONG HashAlg)

描述： 数据摘要签名验证，签名值格式为Base64编码的数据类型A。如果使用SM2算法验证签名，Base64HashData参数是SM3算法的摘要值，应按照GM/T 0009规定的预处理过程计算。

参数：	Base64Cert	Base64编码签名者证书
	Base64HashData	Base64编码摘要值
	SignValue	Base64编码的签名值
	HashAlg	摘要算法，应符合GM/T 0006
返回值：	TRUE	成功

FALSE　　失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.1.35 证书登出 SOF_Logout

原型：BOOL SOF_Logout(BSTR CertID)
描述：退出登录状态。
参数：CertID　　证书唯一标识
返回值：TRUE　　成功
FALSE　　失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

注：证书登出表示释放该证书用户对应的私钥使用权限。后续调用签名、解密等需要拥有私钥使用权限才能正常操作的接口前，需再次成功调用SOF_Login接口获取私钥使用权限。

7.1.36 证书登录状态检测 SOF_IsLogin

原型：BOOL SOF_IsLogin(BSTR CertID)
描述：判断证书用户是否为登录状态。
参数：CertID　　证书唯一标识
返回值：TRUE　　登录状态
FALSE　　非登录状态

7.2 服务器端COM组件接口

7.2.1 服务器端COM组件接口综述

服务器端COM组件接口提供配置管理、证书解析、签名与验证、加密与解密数字信封、XML数据签名与验证、时间戳等功能，共包含39个接口。表3列出了服务器端COM组件接口列表，7.2.2～7.2.40给出了接口详细定义。

表3 服务器端COM组件接口列表

序号	章条号	接口名称	功能
1	7.2.2	SOF_SetCertTrustList	设置证书信任列表
2	7.2.3	SOF_GetCertTrustListAltNames	查询证书信任列表别名
3	7.2.4	SOF_GetCertTrustList	查询证书信任列表
4	7.2.5	SOF_DelCertTrustList	删除证书信任列表
5	7.2.6	SOF_InitCertAppPolicy	初始化应用策略
6	7.2.7	SOF_SetSignMethod	设置签名算法
7	7.2.8	SOF_GetSignMethod	获得签名算法
8	7.2.9	SOF_SetEncryptMethod	设置加密算法
9	7.2.10	SOF_GetEncryptMethod	获得加密算法
10	7.2.11	SOF_GetServerCertificate	获得服务器证书
11	7.2.12	SOF_GenRandom	产生随机数
12	7.2.13	SOF_GetCertInfo	获得证书信息

表 3　服务器端 COM 组件接口列表(续)

序号	章条号	接口名称	功能
13	7.2.14	SOF_GetCertInfoByOid	获得证书扩展信息
14	7.2.15	SOF_ValidateCert	验证证书有效性
15	7.2.16	SOF_SignData	数据签名
16	7.2.17	SOF_VerifySignedData	验证数据签名
17	7.2.18	SOF_SignFile	文件签名
18	7.2.19	SOF_VerifySignedFile	验证文件签名
19	7.2.20	SOF_EncryptData	数据加密
20	7.2.21	SOF_DecryptData	数据解密
21	7.2.22	SOF_EncryptFile	文件加密
22	7.2.23	SOF_DecryptFile	文件解密
23	7.2.24	SOF_SignMessage	消息签名
24	7.2.25	SOF_VerifySignedMessage	验证消息签名
25	7.2.26	SOF_SignMessageDetach	不带原文的消息签名
26	7.2.27	SOF_VerifySignedMessageDetach	验证不带原文的消息签名
27	7.2.28	SOF_GetInfoFromSignedMessage	解析消息签名
28	7.2.29	SOF_SignDataXML	XML数字签名
29	7.2.30	SOF_VerifySignedDataXML	验证XML数字签名
30	7.2.31	SOF_GetXMLSignatureInfo	解析XML签名数据
31	7.2.32	SOF_CreateTimeStampRequest	创建时间戳请求
32	7.2.33	SOF_CreateTimeStampResponse	创建时间戳响应
33	7.2.34	SOF_VerifyTimeStamp	验证时间戳
34	7.2.35	SOF_GetTimeStampInfo	解析时间戳
35	7.2.36	SOF_GetLastError	获取最新的错误代码
36	7.2.37	SOF_HashData	计算数据摘要
37	7.2.38	SOF_HashFile	计算文件摘要
38	7.2.39	SOF_SignHashData	摘要数据签名
39	7.2.40	SOF_VerifySignedHashData	数据摘要签名验证

7.2.2　设置证书信任列表 SOF_SetCertTrustList

原型：　LONG SOF_SetCertTrustList(BSTR CTLAltName, BSTR CTLContent, LONG CTL-ContentLen)

描述：　设置证书信任列表。

参数：　CTLAltName　　证书信任列表别名

　　CTLContent　　Base64编码格式的证书信任列表内容

	CTLContentLen	证书信任列表长度
返回值：	SAR_OK	成功
	其他	失败，返回表A.1定义的错误代码

7.2.3 查询证书信任列表别名 SOF_GetCertTrustListAltNames

原型：	BSTR SOF_GetCertTrustListAltNames()	
描述：	查询证书信任列表别名。	
参数：	无	
返回值：	信任列表别名	成功，返回信任列表别名的字符串组合，如“CA001@CA002@CA003”
	空串	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.2.4 查询证书信任列表 SOF_GetCertTrustList

原型：	BSTR SOF_GetCertTrustList(BSTR CTLAltName)	
描述：	根据别名查询证书信任列表。	
参数：	CTLAltName	证书信任列表别名
返回值：	信任列表	成功，返回Base64编码的证书信任列表
	空串	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.2.5 删除证书信任列表 SOF_DelCertTrustList

原型：	LONG SOF_DelCertTrustList(BSTR CTLAltName)	
描述：	根据别名删除证书信任列表。	
参数：	CTLAltName	证书信任列表别名
返回值：	SAR_OK	成功
	其他	失败，返回表A.1定义的错误代码

7.2.6 初始化应用策略 SOF_InitCertAppPolicy

原型：	LONG SOF_InitCertAppPolicy(BSTR PolicyName)	
描述：	根据应用策略名称设置应用符合的证书应用策略。该名称要和服务器配置文件对应。接口从配置文件中读取应用策略信息，宜包括使用的密钥和证书、信任的根证书、证书验证的策略、验证方式。配置内容自行定义。	
参数：	PolicyName	应用策略名称
返回值：	SAR_OK	成功
	其他	失败，返回表A.1定义的错误代码

7.2.7 设置签名算法 SOF_SetSignMethod

原型：	LONG SOF_SetSignMethod(LONG SignMethod)	
描述：	设置COM组件签名运算使用的签名算法。	
参数：	SignMethod	签名算法标识，应符合GM/T 0006
返回值：	SAR_OK	成功
	其他	失败，返回表A.1定义的错误代码

7.2.8 获得签名算法 SOF_GetSignMethod

原型： LONG SOF_GetSignMethod()
描述： 获得组件当前签名和验签运算使用的签名算法。
参数： 无

返回值：	签名算法标识	成功，返回接口当前使用的签名算法标识
	0	没有设置签名算法

7.2.9 设置加密算法 SOF_SetEncryptMethod

原型： LONG SOF_SetEncryptMethod(LONG EncryptMethod)
描述： 设置组件对数据加密使用的对称算法。

参数：	EncryptMethod	对称密码算法标识，应符合GM/T 0006。本接口支持不带附加认证数据的加密算法
返回值：	SAR_OK	成功
	其他	失败，返回表A.1定义的错误代码

7.2.10 获得加密算法 SOF_GetEncryptMethod

原型： LONG SOF_GetEncryptMethod()
描述： 获得组件当前使用的对称加解密算法。
参数： 无

返回值：	加密算法标识	成功，返回接口当前使用的加密算法标识
	0	没有设置加密算法

7.2.11 获得服务器证书 SOF_GetServerCertificate

原型： BSTR SOF_GetServerCertificate(SHORT CertUsage)
描述： 读取当前应用指定的服务器证书。

参数：	CertUsage	证书用途，取值范围： a） 1加密证书； b） 2签名证书
返回值：	服务器证书	成功，返回Base64编码的服务器证书
	空串	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.2.12 产生随机数 SOF_GenRandom

原型： BSTR SOF_GenRandom(SHORT RandomLen)
描述： 产生指定长度的随机数。

参数：	RandomLen	待产生随机数的长度(字节数)
返回值：	随机数	成功，返回Base64编码的随机数值
	空串	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.2.13 获得证书信息 SOF_GetCertInfo

原型：BSTR SOF_GetCertInfo(BSTR Base64Cert, LONG Type)

描述：根据指定类型，获取证书内的相关信息。

参数：

	Base64Cert	Base64编码的数字证书
	Type	证书解析标识，应符合GM/T 0006

返回值：

证书信息	成功，返回证书内指定类型的信息
空串	失败或证书中不存在该项内容

7.2.14 获得证书扩展信息 SOF_GetCertInfoByOid

原型：BSTR SOF_GetCertInfoByOid(BSTR Base64Cert, BSTR Oid)

描述：根据OID获取证书扩展项信息。

参数：

	Base64Cert	Base64编码的证书
	Oid	私有扩展对象ID，如“1.2.156.xxx”

返回值：

证书扩展信息	成功，返回证书私有扩展项OID对应的信息
空串	失败或证书中不存在该私有扩展项

7.2.15 验证证书有效性 SOF_ValidateCert

原型：LONG SOF_ValidateCert(BSTR Base64Cert)

描述：根据应用的策略根据验证证书有效性。

参数：

	Base64Cert	待验证的Base64编码证书

返回值：

SAR_OK	验证成功
其他	验证失败，失败原因应符合表A.1中的错误代码范围0X0B000500～0X0B000505

注：基本的证书验证策略需包括：

a) 验证CA信任列表，各层都要进行签名验证；

b) 各层证书的有效期验证；

c) 各层证书的吊销状态。在特殊情况下(如：网络条件不允许)，证书的吊销状态能采取灵活方式，由应用系统内部维护一个吊销列表，在证书登录认证时应用该吊销列表。验证证书有效性也能采取代理验证方式。

7.2.16 数据签名 SOF_SignData

原型：BSTR SOF_SignData(BSTR InData)

描述：对字符串数据进行数字签名，返回Base64编码的数据类型A签名结果。

参数：

	InData	待签名的数据原文

返回值：

签名值	成功，返回Base64编码的签名值
空串	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.2.17 验证数据签名 SOF_VerifySignedData

原型：LONG SOF_VerifySignedData(BSTR Base64Cert, BSTR InData, BSTR SignValue)

描述：验证数字签名，签名值格式为Base64编码的数据类型A。

参数：

	Base64Cert	Base64编码的签名证书
	InData	待验证的原文
	SignValue	Base64编码的签名值

返回值：

SAR_OK	成功
其他	失败，返回表A.1定义的错误代码

7.2.18 文件签名 SOF_SignFile

原型：BSTR SOF_SignFile(BSTR InFile)

描述：对文件进行数字签名，返回Base64编码的数据类型A签名结果。

参数：

InFile	待签名的文件全路径(包含路径＋文件名)

返回值：

签名值	成功，返回Base64编码的签名值
空串	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

注：待签名的文件全路径指运行本接口的主机上的文件，非服务器端文件。

7.2.19 验证文件签名 SOF_VerifySignedFile

原型：LONG SOF_VerifySignedFile(BSTR Base64Cert, BSTR InFile, BSTR SignValue)

描述：验证文件数字签名，签名值格式为Base64编码的数据类型A。

参数：

	Base64Cert	Base64编码的签名证书
	InFile	待验证的文件全路径(包含路径＋文件名)
	SignValue	Base64编码的签名值

返回值：

SAR_OK	成功
其他	失败，返回表A.1定义的错误代码

注：待验证的原文路径指运行本接口的主机上的文件，非服务器端文件。

7.2.20 数据加密 SOF_EncryptData

原型：BSTR SOF_EncryptData(BSTR Cert, BSTR InData)

描述：数字信封加密，加密过程为使用临时产生的对称密钥加密数据，然后使用数字证书的公钥加密对称密钥，返回Base64编码的数据类型B密文数据。

参数：

	Cert	Base64编码的数据接收者的加密证书，如有多个接收者，多个接收者加密证书之间用&&&作为分隔符连接
	InData	待加密的明文

返回值：

密文数据	成功，返回Base64编码格式的密文数据
空串	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.2.21 数据解密 SOF_DecryptData

原型：BSTR SOF_DecryptData(BSTR CertID, BSTR InData)

描述：解密格式为Base64编码的数据类型B密文数据。

参数：

	CertID	解密密钥对应的证书唯一标识，如不需要可传空串
	InData	Base64编码的密文数据

返回值：明文数据　成功，返回解密后的明文
空串　失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.2.22 文件加密 SOF_EncryptFile

原型：LONG SOF_EncryptFile(BSTR Cert, BSTR InFile, BSTR OutFile)
描述：使用证书加密文件，密文文件结果为数据类型B。
参数：Cert　Base64编码的数据接收者的加密证书，如有多个接收者，多个接收者加密证书之间用&&&作为分隔符连接
InFile　待加密的明文文件全路径(包含路径＋文件名)
OutFile　密文文件保存全路径(包含路径＋文件名)
返回值：SAR_OK　成功
其他　失败，返回表A.1定义的错误代码

注：待加密的明文文件全路径和密文文件保存全路径，指运行本接口的主机上的文件，非服务器端文件。

7.2.23 文件解密 SOF_DecryptFile

原型：LONG SOF_DecryptFile(BSTR CertID, BSTR InFile, BSTR OutFile)
描述：使用证书对应的私钥解密文件，密文文件格式为数据类型B。
参数：CertID　解密密钥对应的证书唯一标识，如不需要可传空串
InFile　待解密的密文文件全路径(包含路径＋文件名)
OutFile　明文文件保存全路径(包含路径＋文件名)
返回值：SAR_OK　成功
其他　失败，返回表A.1定义的错误代码

注：待解密的密文文件全路径和明文文件保存全路径，指运行本接口的主机上的文件，非服务器端文件。

7.2.24 消息签名 SOF_SignMessage

原型：BSTR SOF_SignMessage(BSTR InData)
描述：对字符串数据进行消息签名，返回Base64编码的数带原文的据类型B签名结果。
参数：InData　待签名的数据原文
返回值：消息签名值　成功，返回Base64编码的签名值
空串　失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.2.25 验证消息签名 SOF_VerifySignedMessage

原型：LONG SOF_VerifySignedMessage(BSTR SignedMessage)
描述：验证消息签名值，消息签名值格式为Base64编码的带原文的数据类型B。
参数：SignedMessage　Base64编码的签名值
返回值：SAR_OK　成功
其他　失败，返回表A.1定义的错误代码

7.2.26 不带原文的消息签名 SOF_SignMessageDetach

原型：BSTR SOF_SignMessageDetach(BSTR InData)

描述： 对字符串数据进行数字签名，返回Base64编码的不带原文的据类型B签名结果。

参数： InData 待签名的数据原文

返回值： 消息签名值 成功，返回Base64编码的签名值

空串 失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.2.27 验证不带原文的消息签名 SOF_VerifySignedMessageDetach

原型： LONG SOF_VerifySignedMessageDetach(BSTR InData,BSTR SignedMessage)

描述： 验证消息签名值，签名值格式为Base64编码的不带原文的数据类型B。

参数： InData 原文数据

SignedMessage Base64编码的签名值

返回值： SAR_OK 成功

其他 失败，返回表A.1定义的错误代码

7.2.28 解析消息签名 SOF_GetInfoFromSignedMessage

原型： BSTR SOF_GetInfoFromSignedMessage(BSTR SignedMessage,SHORT Type)

描述： 解析消息签名值中的信息，包括：原文、签名值、签名证书等，消息签名值格式为Base64编码的数据类型B。

参数： SignedMessage Base64编码的签名值

Type 类型，取值范围：

a) 1解析出原文；

b) 2解析出Base64编码的签名者证书；

c) 3解析出Base64编码的签名值

返回值： 解析结果 成功，返回解析出的对应信息

空串 失败或不存在该项

7.2.29 XML数字签名 SOF_SignDataXML

原型： BSTR SOF_SignDataXML(BSTR InData)

描述： 对XML数据进行数字签名，证书为RSA算法时签名结果符合RFC3275，证书为SM2算法时签名结果符合GB/T 25061。

参数： InData XML格式的签名原文

返回值： XML签名结果 成功，返回XML签名结果

空串 失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

注：缺省配置和参数：

a) 采用封皮签名，签名前需对签名原文做规范化；

b) 规范化方法采用带注释的XML规范化1.1，证书为RSA算法时标识符为http://www.w3.org/2006/12/xml-c14n11#WithComment，证书为SM2算法时标识符为http://127.0.0.1/2006/12/xml-c14n11#WithComments。

7.2.30 验证XML数字签名 SOF_VerifySignedDataXML

原型： LONG SOF_VerifySignedDataXML(BSTR XMLSignedData)

描述： 验证XML签名。

参数： XMLSignedData XML签名结果

返回值： SAR_OK 成功

其他 失败，返回表A.1定义的错误代码

7.2.31 解析XML签名数据 SOF_GetXMLSignatureInfo

原型： BSTR SOF_GetXMLSignatureInfo(BSTR XMLSignedData, SHORT Type)

描述： 解析XML签名数据，获取签名值、XML原文、证书等信息。

参数： XMLSignedData XML签名结果

Type 待解析的参数类型，取值范围：

a) 1解析出XML原文；

b) 2解析出摘要值；

c) 3解析出签名值；

d) 4解析出签名证书；

e) 5解析出摘要算法；

f) 6解析出签名算法

返回值： XML信息 成功，返回根据Type解析出各项对应的信息

空串 失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

注：XML签名缺省配置和参数值样例：

a) XML原文取自Object元素；

b) 摘要值取自DigestValue元素，形如http://127.0.0.1/2001/04/xmldsig-more#sm3；

c) 签名值取自SignatureValue元素，为BASE64编码；

d) 签名证书取自X509Data元素，为BASE64编码；

e) 摘要算法取自DigestMethod元素的Algorithm属性，形如http://127.0.0.1/2001/04/xmldsig-more#sm3；

f) 签名算法取自SignatureMethod元素的Algorithm属性，形如http://127.0.0.1/2001/04/xmldsig-more#sm2-sm3。

7.2.32 创建时间戳请求 SOF_CreateTimeStampRequest

原型： BSTR SOF_CreateTimeStampRequest(BSTR InData, LONG HashAlg, SHORT ReqType, BSTR Base64Ext)

描述： 创建时间戳请求。

参数： InData 待创建时间戳请求的原文

HashAlg 摘要算法，应符合GM/T 0006

ReqType 请求的时间戳服务类型，取值范围：

a) 0表示时间戳响应应包含时间戳服务器证书

b) 1表示时间戳响应不包含时间戳服务器证书

Base64Ext Base64编码格式的扩展项

返回值： 时间戳请求 成功，返回Base64编码格式的时间戳请求

空串 失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.2.33 创建时间戳响应 SOF_CreateTimeStampResponse

原型：BSTR SOF_CreateTimeStampResponse(BSTR TSRequest)

描述：创建时间戳响应，即签发时间戳。

参数：

TSRequest	Base64编码格式的时间戳请求

返回值：

时间戳响应	成功，返回Base64编码格式的时间戳响应
空串	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.2.34 验证时间戳 SOF_VerifyTimeStamp

原型：LONG SOF_VerifyTimeStamp(BSTR InData,BSTR TSResData, BSTR Base64Cert)

描述：验证时间戳。

参数：

InData	待验证的原文
TSResData	Base64编码格式的时间戳响应
Base64Cert	Base64编码格式的时间戳服务器证书，在时间戳响应中不包含时间戳服务器证书时应传入本参数

返回值：

SAR_OK	成功
其他	失败，返回表A.1定义的错误代码

7.2.35 解析时间戳 SOF_GetTimeStampInfo

原型：BSTR SOF_GetTimeStampInfo(BSTR TSResData, SHORT Type)

描述：解析时间戳，获得时间戳的信息，包括时间、时间戳服务器证书等。

参数：

TSResData	时间戳
Type	信息类型，取值范围： a) 1返回时间； b) 3返回时间戳服务器签名证书

返回值：

时间戳信息	成功，返回解析出对应type的信息
空串	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.2.36 获取最新的错误代码 SOF_GetLastError

原型：LONG SOF_GetLastError()

描述：获取接口最新的错误代码。

参数：无

返回值：错误代码，应符合表A.1错误代码表

7.2.37 计算数据摘要 SOF_HashData

原型：BSTR SOF_HashData(LONG HashAlg, BSTR InData,BSTR SignCert,BSTR UserID)

描述：计算数据摘要，如果是SM3算法，当SignCert和UsreID为空时，只计算数据的摘要值，当SignCert和UsreID值不为空时，应按照GM/T 0009规定的预处理过程计算，摘要值可以作为SM2签名的输入。

参数：

HashAlg	摘要算法，应符合GM/T 0006

	Indata	原始数据
	SignCert	Base64编码的签名者证书，摘要算法为SM3时有效，如不需要传空串
	UserID	签名者用户ID，摘要算法为SM3时有效，如果SignCert参数为空，则本参数无意义
返回值：	摘要值	成功，返回Base64编码的数据摘要值
	空串	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.2.38 计算文件摘要 SOF_HashFile

原型： BSTR SOF_HashFile(LONG HashAlg, BSTR InFile,BSTR SignCert,BSTR UserID)

描述： 计算文件数据摘要，如果是SM3算法，当SignCert和UsreID为空时，只计算数据的摘要值，当SignCert和UsreID值不为空时，应按照GM/T 0009规定的预处理过程计算，摘要值可以作为SM2签名的输入。

参数：	HashAlg	摘要算法，应符合GM/T 0006
	InFile	原文文件全路径(包含路径＋文件名)
	SignCert	Base64编码的签名者证书，摘要算法为SM3时有效，如不需要传空串
	UserID	签名者用户ID，摘要算法为SM3时有效，如果SignCert参数为空，则本参数无意义
返回值：	摘要值	成功，返回Base64编码的数据摘要值
	空串	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

注：原文文件全路径，指运行本接口的主机上的文件，非服务器端文件。

7.2.39 摘要数据签名 SOF_SignHashData

原型： BSTR SOF_SignHashData(BSTR Base64HashData, LONG HashAlg)

描述： 对数据摘要签名，返回Base64编码的数据类型A签名结果。Base64HashData值一般是SOF_HashData或SOF_HashFile函数的计算结果。如果是SM2签名，SM3算法的摘要值应按照GM/T 0009规定的预处理过程计算。

参数：	Base64HashData	Base64编码的摘要值
	HashAlg	摘要算法，应符合GM/T 0006
返回值：	签名值	Base64编码的签名值
	空串	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.2.40 数据摘要签名验证 SOF_VerifySignedHashData

原型： BOOL SOF_VerifySignedHashData(BSTR Base64Cert, BSTR Base64HashData,BSTR SignValue, LONG HashAlg)

描述： 数据摘要签名验证，签名值为Base64编码的数据类型A。如果使用SM2算法验证签名，Base64HashData参数是SM3算法的摘要值，应按照GM/T 0009规定的预处理过程计算。

参数：	Base64Cert	Base64编码签名者证书
	Base64HashData	Base64编码摘要值
	SignValue	Base64编码的签名值

返回值：HashAlg　　摘要算法，应符合GM/T 0006
TRUE　　成功
FALSE　　失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.3 服务器端Java组件接口

7.3.1 服务器端Java组件接口综述

服务器端Java组件接口提供配置管理、证书解析、签名与验证、加密与解密数字信封、XML数据签名与验证、时间戳等功能，共包含40个接口。表4列出了服务器端COM组件接口列表，7.3.2～7.3.41给出了接口详细定义。

服务器端Java组件接口可通过两种方式获得错误信息，一种是通过SOF_getLastError获取错误代码，另一种是通过Java捕获异常方式获取错误信息。本文件仅对使用SOF_getLastError方式进行说明。

表4　服务器端Java组件接口列表

序号	章条号	接口名称	功能
1	7.3.2	SOF_setCertTrustList	设置证书信任列表
2	7.3.3	SOF_getCertTrustListAltNames	查询证书信任列表别名
3	7.3.4	SOF_getCertTrustList	根据别名查询证书信任列表
4	7.3.5	SOF_delCertTrustList	删除证书信任列表
5	7.3.6	SOF_getInstance	获取指定应用的实例
6	7.3.7	SOF_setSignMethod	设置签名算法
7	7.3.8	SOF_getSignMethod	获得签名算法
8	7.3.9	SOF_setEncryptMethod	设置加密算法
9	7.3.10	SOF_getEncryptMethod	获得加密算法
10	7.3.11	SOF_getServerCertificate	获得服务器证书
11	7.3.12	SOF_getServerCertificateByUsage	获得指定密钥用途的服务器证书
12	7.3.13	SOF_genRandom	产生随机数
13	7.3.14	SOF_getCertInfo	获得证书信息
14	7.3.15	SOF_getCertInfoByOid	获得证书扩展信息
15	7.3.16	SOF_validateCert	验证证书有效性
16	7.3.17	SOF_signData	数据签名
17	7.3.18	SOF_verifySignedData	验证数据签名
18	7.3.19	SOF_signFile	文件签名
19	7.3.20	SOF_verifySignedFile	验证文件签名
20	7.3.21	SOF_encryptData	数据加密
21	7.3.22	SOF_decryptData	数据解密
22	7.3.23	SOF_encryptFile	文件加密
23	7.3.24	SOF_decryptFile	文件解密

表 4　服务器端 Java 组件接口列表（续）

序号	章条号	接口名称	功能
24	7.3.25	SOF_signMessage	消息签名
25	7.3.26	SOF_verifySignedMessage	验证消息签名
26	7.3.27	SOF_getInfoFromSignedMessage	解析消息签名
27	7.3.28	SOF_signMessageDetach	不带原文的消息签名
28	7.3.29	SOF_verifySignedMessageDetach	验证不带原文的消息签名
29	7.3.30	SOF_signDataXML	XML 数字签名
30	7.3.31	SOF_verifySignedDataXML	验证 XML 数字签名
31	7.3.32	SOF_getXMLSignatureInfo	解析 XML 签名数据
32	7.3.33	SOF_createTimeStampRequest	创建时间戳请求
33	7.3.34	SOF_createTimeStampResponse	创建时间戳响应
34	7.3.35	SOF_verifyTimeStamp	验证时间戳
35	7.3.36	SOF_getTimeStampInfo	解析时间戳
36	7.3.37	SOF_getLastError	获取最新的错误代码
37	7.3.38	SOF_hashData	计算数据摘要
38	7.3.39	SOF_hashFile	计算文件摘要
39	7.3.40	SOF_signHashData	摘要数据签名
40	7.3.41	SOF_verifySignedHashData	数据摘要签名验证

7.3.2　设置证书信任列表 SOF_setCertTrustList

原型：　boolean SOF_setCertTrustList(java.lang.String ctlAltName,
java.lang.String ctlContent)

描述：　设置证书信任列表。

参数：　ctlAltName　　证书信任列表别名
ctlContent　　Base64 编码格式的证书信任列表内容

返回值：　true　　成功
false　　失败，可通过 SOF_getLastError 获取符合表 A.1 定义的错误代码

7.3.3　查询证书信任列表别名 SOF_getCertTrustListAltNames

原型：　java.lang.String SOF_getCertTrustListAltNames()

描述：　查询证书信任列表别名。

参数：　无

返回值：　信任列表别名　　成功，返回信任列表别名的字符串组合，如“CA001@CA002@CA003”
空串　　失败，可通过 SOF_getLastError 获取符合表 A.1 定义的错误代码

7.3.4 根据别名查询证书信任列表 SOF_getCertTrustList

原型：java.lang.String SOF_getCertTrustList(java.lang.String ctlAltName)
描述：根据别名查询证书信任列表。
参数：ctlAltName　证书信任列表别名
返回值：信任列表　成功，返回Base64编码的证书信任列表
空串　失败，可通过SOF_getLastError获取符合表A.1定义的错误代码

7.3.5 删除证书信任列表 SOF_delCertTrustList

原型：boolean SOF_delCertTrustList(java.lang.String ctlAltName)
描述：根据别名删除证书信任列表。
参数：ctlAltName　证书信任列表别名
返回值：true　成功
false　失败，可通过SOF_getLastError获取符合表A.1定义的错误代码

7.3.6 获取指定应用的实例 SOF_getInstance

原型：static java.lang.Object SOF_getInstance(java.lang.String policyName)
描述：初始化接口，通过应用别名获取实例，应用别名关联所配置的证书、密钥、信任证书链、算法类型、CRL及证书验证策略等。用户如果有多应用需求，可同时获取多个实例对象满足不同的使用需求，不同的实例在调用方法时会有不同效果（如：不同密钥的签名，不同算法的加密，不同的证书验证策略）。
参数：policyName　应用策略名称
返回值：应用示例对象　成功，返回此应用策略名称所对应的实例
空　失败，可通过SOF_getLastError获取符合表A.1定义的错误代码

7.3.7 设置签名算法 SOF_setSignMethod

原型：void SOF_setSignMethod(long signMethod)
描述：设置Java组件签名运算使用的签名算法。
参数：signMethod　签名算法标识，应符合GM/T 0006
返回值：无

7.3.8 获得签名算法 SOF_getSignMethod

原型：long SOF_getSignMethod()
描述：获得Java组件当前签名和验签运算使用的签名算法标识。
参数：无
返回值：签名算法标识　成功，返回接口当前使用的签名算法标识
0　没有设置签名算法

7.3.9 设置加密算法 SOF_setEncryptMethod

原型：void SOF_setEncryptMethod(long encryptMethod)

描述： 设置组件对数据加密使用的对称算法标识。

参数： encryptMethod 对称算法标识，应符合GM/T 0006。本接口支持不带附加认证数据的加密算法

返回值： 无

7.3.10 获得加密算法 SOF_getEncryptMethod

原型： long SOF_getEncryptMethod()

描述： 获得组件当前使用的对称算法标识。

参数： 无

返回值： 加密算法标识 成功，返回接口当前使用的加密算法标识

0 没有设置加密算法

7.3.11 获得服务器证书 SOF_getServerCertificate

原型： java.lang.String SOF_getServerCertificate()

描述： 读取当前应用的服务器的签名证书。如果有签名证书则得到签名证书，否则得到加密证书。

参数： 无

返回值： 服务器证书 成功，返回Base64编码的服务器证书

空串 失败，可通过SOF_getLastError获取符合表A.1定义的错误代码

7.3.12 获得指定密钥用途的服务器证书 SOF_getServerCertificateByUsage

原型： java.lang.String SOF_getServerCertificateByUsage(short certUsage)

描述： 根据密钥用途，读取当前应用指定的服务器证书。

参数： certUsage 证书用途，取值范围：

a） 1加密证书；

b） 2签名证书

返回值： 服务器证书 成功，返回Base64编码的服务器证书

空串 失败，可通过SOF_getLastError获取符合表A.1定义的错误代码

7.3.13 产生随机数 SOF_genRandom

原型： java.lang.String SOF_genRandom(short randomLen)

描述： 产生指定长度的随机数。

参数： randomLen 待产生的随机数字节长度(字节数)

返回值： 随机数 成功，返回Base64编码的随机数值

空串 失败，可通过SOF_getLastError获取符合表A.1定义的错误代码

7.3.14 获得证书信息 SOF_getCertInfo

原型： java.lang.String SOF_getCertInfo(java.lang.String base64Cert, int type)

描述： 根据type解析证书内的相关信息。

参数： base64Cert Base64编码的证书

	type	证书解析标识，应符合GM/T 0006
返回值：	证书信息	成功，返回证书内指定类型的信息
	空串	失败或证书中不存在该项内容

7.3.15 获得证书扩展信息 SOF_getCertInfoByOid

原型： java.lang.String SOF_getCertInfoByOid(java.lang.String base64Cert,
java.lang.String oid)

描述： 根据OID获取证书私有扩展项信息。

参数：	base64Cert	Base64编码的证书
	oid	私有扩展对象ID，如“1.2.156.xxx”
返回值：	证书扩展信息	成功，返回证书私有扩展项OID对应的信息
	空串	失败或证书中不存在该私有扩展项

7.3.16 验证证书有效性 SOF_validateCert

原型： int SOF_validateCert(java.lang.String base64Cert)

描述： 根据应用的策略验证证书有效性。

参数：	base64Cert	待验证的Base64编码证书
返回值：	SAR_OK	验证成功，证书有效
	其他	验证失败，失败原因应符合表A.1中的错误代码范围0X0B000500～0X0B000505

7.3.17 数据签名 SOF_signData

原型： java.lang.String SOF_signData(byte[] inData)

描述： 对字符串数据进行数字签名，返回Base64编码的数据类型A签名结果。

参数：	inData	待签名的数据原文
返回值：	签名值	成功，返回Base64编码的签名值
	空串	失败，可通过SOF_getLastError获取符合表A.1定义的错误代码

7.3.18 验证数据签名 SOF_verifySignedData

原型： boolean SOF_verifySignedData(java.lang.String base64Cert, byte[] inData,
java.lang.String signValue)

描述： 验证数字签名，签名值格式为Base64编码的数据类型A。

参数：	base64Cert	Base64编码的签名证书
	inData	待验证的原文
	signValue	Base64编码的签名值
返回值：	true	验证成功
	false	失败，可通过SOF_getLastError获取符合表A.1定义的错误代码

7.3.19 文件签名 SOF_signFile

原型： java.lang.String SOF_signFile(java.lang.String inFile)

描述： 对文件数字签名，返回Base64编码的数据类型A签名结果。

参数： inFile 待签名的文件全路径(包含路径＋文件名)

返回值： 签名值 成功，返回Base64编码的签名值

空串 失败，可通过SOF_getLastError获取符合表A.1定义的错误代码

注：待签名的文件全路径，指运行本接口的主机上的文件，非服务器端文件。

7.3.20 验证文件签名 SOF_verifySignedFile

原型： boolean SOF_verifySignedFile(java.lang.String base64Cert,
java.lang.String inFile,
java.lang.String signValue)

描述： 验证文件数字签名，签名值格式为Base64编码的数据类型A。

参数： base64Cert Base64编码的签名证书

inFile 待验证的文件路径

signValue Base64编码的签名值

返回值： true 验证成功

false 失败，可通过SOF_getLastError获取符合表A.1定义的错误代码

注：待签名的文件全路径，指运行本接口的主机上的文件，非服务器端文件。

7.3.21 数据加密 SOF_encryptData

原型： java.lang.String SOF_encryptData(java.lang.String baseCert, byte[] inData)

描述： 数字信封加密，加密过程为使用临时产生的对称密钥加密数据，然后使用数字证书的公钥加密对称密钥，返回Base64编码的数据类型B的密文数据。

参数： baseCert Base64编码的数据接收者的加密证书，如有多个接收者，多个接收者证书之间用&&&作为分隔符连接

inData 待加密的明文

返回值： 密文数据 成功，返回Base64编码格式的密文数据

空串 失败，可通过SOF_getLastError获取符合表A.1定义的错误代码

7.3.22 数据解密 SOF_decryptData

原型： byte[] SOF_decryptData(java.lang.String certId, java.lang.String inData)

描述： 解密Base64编码的数据类型B的密文数据。

参数： certId 解密密钥对应的证书唯一标识，如不需要可传空串

inData Base64编码的待解密的密文数据

返回值： 明文数据 成功，返回解密后的明文

空串 失败，可通过SOF_getLastError获取符合表A.1定义的错误代码

7.3.23 文件加密 SOF_encryptFile

原型： boolean SOF_encryptFile(java.lang.String base64Cert,
java.lang.String inFile,
java.lang.String outFile)

描述： 加密文件，得到数据类型B的密文文件。

参数：

base64Cert	Base64编码的数据接收者的加密证书，如有多个接收者，多个接收者证书之间用&&&作为分隔符连接
inFile	待加密的明文文件全路径
outFile	密文文件保存全路径

返回值：

true	成功
false	失败，可通过SOF_getLastError获取符合表A.1定义的错误代码

注：待加密的明文文件全路径和密文文件保存全路径，指运行本接口的主机上的文件，非服务器端文件。

7.3.24 文件解密 SOF_decryptFile

原型： boolean SOF_decryptFile(java.lang.String certId,
java.lang.String inFile,
java.lang.String outFile)

描述： 解密密文文件，密文文件格式为数据类型B。

参数：

certId	解密密钥对应的证书唯一标识，如不需要可传空串
inFile	待解密的密文文件路径
outFile	明文文件保存路径

返回值：

true	成功
false	失败，可通过SOF_getLastError获取符合表A.1定义的错误代码

注：待解密的密文文件全路径和明文文件保存全路径，指运行本接口的主机上的文件，非服务器端文件。

7.3.25 消息签名 SOF_signMessage

原型： java.lang.String SOF_signMessage(byte[] inData)

描述： 对字符串数据进行消息签名，返回Base64编码的带原文的数据类型B签名结果。

参数：

inData	待签名的数据原文

返回值：

消息签名值	成功，返回Base64编码的数据类型B
空串	失败，可通过SOF_getLastError获取符合表A.1定义的错误代码

7.3.26 验证消息签名 SOF_verifySignedMessage

原型： boolean SOF_verifySignedMessage(java.lang.String signedMessage)

描述： 验证消息签名，签名格式为Base64编码的带原文的数据类型B。

参数：

signedMessage	Base64编码的签名值

返回值：

true	成功
false	失败，可通过SOF_getLastError获取符合表A.1定义的错误代码

7.3.27 解析消息签名 SOF_getInfoFromSignedMessage

原型： byte[] SOF_getInfoFromSignedMessage(java.lang.String signedMessage,
short type)

描述： 解析Base64编码的带原文的数据类型B签名值的信息，可获得原文、签名值、签名证书等信息。

参数：	signedMessage	Base64编码的签名值
	type	类型，取值范围： a） 1解析出原文； b） 2解析出签名者证书； c） 3解析出签名值
返回值：	解析结果	成功，返回type对应的值
	空串	失败或不存在该项

7.3.28 不带原文的消息签名 SOF_signMessageDetach

原型： java.lang.String SOF_signMessageDetach(byte[] inData)

描述： 对字符串数据进行消息签名，返回Base64编码的不带原文的数据类型B的签名结果。

参数：	inData	待签名的数据原文
返回值：	消息签名值	成功，返回Base64编码的签名值
	空串	失败，可通过SOF_getLastError获取符合表A.1定义的错误代码

7.3.29 验证不带原文的消息签名 SOF_verifySignedMessageDetach

原型： boolean SOF_verifySignedMessageDetach(byte[] inData,
java.lang.String signedMessage)

描述： 验证签名格式为Base64编码的不带原文数据类型B的数字签名。

参数：	inData	原文
	signedMessage	Base64编码的签名值
返回值：	true	成功
	false	失败，可通过SOF_getLastError获取符合表A.1定义的错误代码

7.3.30 XML数字签名 SOF_signDataXML

原型： java.lang.String SOF_signDataXML(java.lang.String inData)

描述： 对XML数据进行数字签名，证书为RSA算法时签名结果符合RFC3275，证书为SM2算法时签名结果符合GB/T 25061。

参数：	inData	签名原文，XML格式
返回值：	XML签名结果	成功，返回XML签名结果
	空串	失败，可通过SOF_getLastError获取符合表A.1定义的错误代码

注：缺省配置和参数：

a） 采用封皮签名，签名前需对签名原文做规范化；

b） 规范化方法采用带注释的XML规范化1.1，证书为RSA算法时标识符为http://www.w3.org/2006/12/xml-c14n11#WithComment，证书为SM2算法时标识符为http://127.0.0.1/2006/12/xml-c14n11#WithComments。

7.3.31 验证XML数字签名 SOF_verifySignedDataXML

原型： boolean SOF_verifySignedDataXML(java.lang.String xmlSignedData)

描述： 验证XML签名。

参数： xmlSignedData XML签名结果

返回值：true　　成功
false　　失败，可通过SOF_getLastError获取符合表A.1定义的错误代码

7.3.32 解析XML签名数据 SOF_getXMLSignatureInfo

原型：java.lang.String SOF_getXMLSignatureInfo(
java.lang.String xmlSignedData, short type)
描述：解析XML签名数据，获取签名值、XML原文、证书等信息。
参数：xmlSignedData　　XML签名结果
type　　待解析的参数类型，取值范围：
a) 1解析出XML原文；
b) 2解析出摘要值；
c) 3解析出签名值；
d) 4解析出签名证书；
e) 5解析出摘要算法；
f) 6解析出签名算法
返回值：XML信息　　成功，返回根据type解析出各项对应的信息
空　　失败，可通过SOF_getLastError获取符合表A.1定义的错误代码

注：XML签名缺省配置和参数值样例：

a) XML原文取自Object元素；
b) 摘要值取自DigestValue元素，形如http://127.0.0.1/2001/04/xmldsig-more#sm3；
c) 签名值取自SignatureValue元素，为BASE64编码；
d) 签名证书取自X509Data元素，为BASE64编码；
e) 摘要算法取自DigestMethod元素的Algorithm属性，形如http://127.0.0.1/2001/04/xmldsig-more#sm3；
f) 签名算法取自SignatureMethod元素的Algorithm属性，形如http://127.0.0.1/2001/04/xmldsig-more#sm2-sm3。

7.3.33 创建时间戳请求 SOF_createTimeStampRequest

原型：java.lang.String SOF_createTimeStampRequest(byte[] inData,
long hashAlg,
short reqType,
byte[] extension)
描述：创建时间戳请求。
参数：inData　　待创建时间戳请求的原文
hashAlg　　摘要算法，应符合GM/T 0006
reqType　　请求的时间戳服务类型：
a) 0表示时间戳响应应包含时间戳服务器证书
b) 1表示时间戳响应不包含时间戳服务器证书
extension　　扩展项
返回值：时间戳请求　　成功，返回Base64编码格式的时间戳请求
空串　　失败，可通过SOF_getLastError获取符合表A.1定义的错误代码

7.3.34 创建时间戳响应 SOF_createTimeStampResponse

原型：java.lang.String SOF_createTimeStampResponse(java.lang.String tsRequest)

描述：创建时间戳响应，即签发时间戳。

参数：

tsRequest	Base64编码的时间戳请求

返回值：

时间戳响应	成功，返回Base64编码格式的时间戳响应
空串	失败，可通过SOF_getLastError获取符合表A.1定义的错误代码

7.3.35 验证时间戳 SOF_verifyTimeStamp

原型：boolean SOF_verifyTimeStamp(byte[] inData,
java.lang.String tsResData,
java.lang.String base64Cert)

描述：验证时间戳。

参数：

inData	待验证的原文
tsResData	Base64编码格式的时间戳响应
base64Cert	Base64编码格式的时间戳服务器证书，在时间戳响应中不包含时间戳服务器证书时应传入本参数

返回值：

true	成功
false	失败，可通过SOF_getLastError获取符合表A.1定义的错误代码

7.3.36 解析时间戳 SOF_getTimeStampInfo

原型：java.lang.String SOF_getTimeStampInfo(java.lang.String tsResData, short type)

描述：解析时间戳，获得时间戳的信息，包括时间、时间戳服务器证书等。

参数：

tsResData	Base64编码的时间戳
type	信息类型，type取值范围： a) 1返回时间； b) 3返回时间戳服务器签名证书

返回值：

时间戳信息	成功，返回解析出对应type的信息
空串	失败，可通过SOF_getLastError获取符合表A.1定义的错误代码

7.3.37 获取最新的错误代码 SOF_getLastError

原型：long SOF_getLastError()

描述：获取接口最新的错误代码。

参数：无

返回值：错误代码，应符合附录A表A.1错误代码表

7.3.38 计算数据摘要 SOF_hashData

原型：java.lang.String SOF_hashData(long hashAlg,
byte[] inData, byte[] signCert, byte[] userID)

描述：计算数据摘要，如果是SM3算法，当signCert和userID为空时，只计算数据的摘要值，当signCert和userID值不为空时，应按照GM/T 0009规定的预处理过程计算，摘要值可以作为SM2签名的输入。

参数：

hashAlg	摘要算法，应符合GM/T 0006
inData	原始数据
signCert	签名者证书，摘要算法为SM3时有效，如不需要传null
userID	签名者用户ID，摘要算法为SM3时有效，如signCert为null，本参数无意义

返回值：

摘要值	成功，返回Base64编码的数据摘要值
空串	失败，可通过SOF_getLastError获取符合表A.1定义的错误代码

7.3.39 计算文件摘要 SOF_hashFile

原型：java.lang.String SOF_hashFile(long hashAlg,
java.lang.String inFile, byte[] signCert, byte[] userID)

描述：计算文件数据摘要，如果是SM3算法，当signCert和userID为空时，只计算数据的摘要值，当signCert和userID值不为空时，应按照GM/T 0009规定的预处理过程计算，摘要值可以作为SM2签名的输入。

参数：

hashAlg	摘要算法，应符合GM/T 0006
inFile	原文文件全路径(包含路径＋文件名)
signCert	签名者证书，摘要算法为SM3时有效，如不需要传null
userID	签名者用户ID，摘要算法为SM3时有效，如不需要传null

返回值：

摘要值	成功，返回Base64编码的数据摘要值
空串	失败，可通过SOF_getLastError获取符合表A.1定义的错误代码

注：原文文件全路径，指运行本接口的主机上的文件，非服务器端文件。

7.3.40 摘要数据签名 SOF_signHashData

原型：java.lang.String SOF_signHashData(java.lang.String base64HashData,
long hashAlg)

描述：对数据摘要签名，返回Base64编码的数据类型A的签名结果。base64HashData参数一般是SOF_HashData或SOF_HashFile函数的计算结果。如果是SM2签名，SM3算法的摘要值应按照GM/T 0009规定的预处理过程计算。

参数：

base64HashData	Base64编码的签名摘要值
hashAlg	摘要算法，应符合GM/T 0006

返回值：

签名值	Base64编码的签名值
空串	失败，可通过SOF_getLastError获取符合表A.1定义的错误代码

7.3.41 数据摘要签名验证 SOF_verifySignedHashData

原型：boolean SOF_verifySignedHashData(java.lang.String base64Cert,
java.lang.String base64HashData,
java.lang.String signValue, long hashAlg)

描述：　数据摘要签名验证，签名值为Base64编码的数据类型A。
如果使用SM2算法验证签名，base64HashData参数是SM3算法的摘要值，应按照GM/T 0009规定的预处理过程计算。

参数：

base64Cert	Base64编码的签名者证书
base64HashData	Base64编码的摘要值
signValue	Base64编码的签名值
hashAlg	摘要算法，应符合GM/T 0006

返回值：

true	成功
false	失败，可通过SOF_getLastError获取符合表A.1定义的错误代码

7.4　客户端JavaScript脚本接口

7.4.1　客户端JavaScript脚本接口综述

客户端JavaScript脚本接口采用异步方式定义，所有接口的返回值都通过回调函数的第一个参数返回，提供配置管理、证书解析、签名与验证、加密与解密数字信封、XML数据签名与验证等功能，共包含32个接口。表5列出了客户端JavaScript脚本接口列表，7.4.2～7.4.33给出了接口详细定义，其调用示例说明见附录D。

表5　客户端JavaScript脚本接口列表

序号	章条号	接口名称	功能
1	7.4.2	SOF_GetVersion	获取接口的版本号
2	7.4.3	SOF_SetSignMethod	设置签名算法
3	7.4.4	SOF_GetSignMethod	获得签名算法
4	7.4.5	SOF_SetEncryptMethod	设置加密算法
5	7.4.6	SOF_GetEncryptMethod	获得加密算法
6	7.4.7	SOF_GetUserList	获得证书用户列表
7	7.4.8	SOF_ExportUserCert	导出用户签名证书
8	7.4.9	SOF_Login	证书登录
9	7.4.10	SOF_GetPinRetryCount	获取证书口令剩余重试次数
10	7.4.11	SOF_ChangePassWd	修改证书口令
11	7.4.12	SOF_ExportExChangeUserCert	导出用户加密证书
12	7.4.13	SOF_GetCertInfo	获得证书基本信息
13	7.4.14	SOF_GetCertInfoByOid	获得证书扩展信息
14	7.4.15	SOF_GetDeviceInfo	获得设备详细信息
15	7.4.16	SOF_ValidateCert	验证证书有效性
16	7.4.17	SOF_SignData	数据签名
17	7.4.18	SOF_VerifySignedData	验证数据签名
18	7.4.19	SOF_EncryptData	数据加密
19	7.4.20	SOF_DecryptData	数据解密

表 5　客户端 JavaScript 脚本接口列表（续）

序号	章条号	接口名称	功能
20	7.4.21	SOF_SignMessage	消息签名
21	7.4.22	SOF_VerifySignedMessage	验证消息签名
22	7.4.23	SOF_GetInfoFromSignedMessage	解析消息签名
23	7.4.24	SOF_SignDataXML	XML 数字签名
24	7.4.25	SOF_VerifySignedDataXML	验证 XML 数字签名
25	7.4.26	SOF_GetXMLSignatureInfo	解析 XML 签名数据
26	7.4.27	SOF_GenRandom	产生随机数
27	7.4.28	SOF_GetLastError	获取最新的错误信息
28	7.4.29	SOF_HashData	计算数据摘要
29	7.4.30	SOF_SignHashData	摘要数据签名
30	7.4.31	SOF_VerifySignedHashData	数据摘要签名验证
31	7.4.32	SOF_Logout	证书登出
32	7.4.33	SOF_IsLogin	证书登录状态检测

7.4.2　获取接口版本信息 SOF_GetVersion

原型：　function SOF_GetVersion(cb, ctx)
描述：　获取接口的版本号。
参数：　cb　　回调函数
　　ctx　　回调函数所需参数
返回值：　版本号　　成功，返回接口版本号
　　空串　　失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.4.3　设置签名算法 SOF_SetSignMethod

原型：　function SOF_SetSignMethod(SignMethod, cb, ctx)
描述：　设置接口在签名和验签运算时使用的签名算法。
参数：　SignMethod　　签名算法标识，应符合GM/T 0006
　　cb　　回调函数
　　ctx　　回调函数所需参数
返回值：　SAR_OK　　成功
　　其他　　失败，返回表A.1定义的错误代码

7.4.4　获得签名算法 SOF_GetSignMethod

原型：　function SOF_GetSignMethod(cb, ctx)
描述：　获得接口当前使用的签名算法。
参数：　cb　　回调函数
　　ctx　　回调函数所需参数

返回值：	签名算法标识	成功，返回接口当前使用的签名算法标识
	0	当前没有设置签名算法

7.4.5 设置加密算法 SOF_SetEncryptMethod

原型：	function SOF_SetEncryptMethod(EncryptMethod, cb, ctx)	
描述：	设置接口进行数据加密时使用的对称算法。	
参数：	EncryptMethod	对称加解算法标识，应符合GM/T 0006。本接口支持不带附加认证数据的加密算法
	cb	回调函数
	ctx	回调函数所需参数
返回值：	SOR_OK	成功
	其他	失败，返回符合表A.1定义的错误代码

7.4.6 获得加密算法 SOF_GetEncryptMethod

原型：	function SOF_GetEncryptMethod(cb, ctx)	
描述：	获得接口当前使用的对称加密算法。	
参数：	cb	回调函数
	ctx	回调函数所需参数
返回值：	加密算法标识	成功，返回接口当前使用的加密算法标识
	0	当前没有设置加密算法

7.4.7 获得证书用户列表 SOF_GetUserList

原型：	function SOF_GetUserList(cb, ctx)	
描述：	获取证书用户列表。	
参数：	cb	回调函数
	ctx	回调函数所需参数
返回值：	证书列表	成功，返回证书列表，格式和7.1.7相同
	空串	失败或当前不存在证书用户列表，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.4.8 导出用户签名证书 SOF_ExportUserCert

原型：	function SOF_ExportUserCert(CertID, cb, ctx)	
描述：	根据证书唯一标识，获取Base64编码的签名证书字符串。	
参数：	CertID	证书唯一标识
	cb	回调函数
	ctx	回调函数所需参数
返回值：	用户签名证书	成功，返回Base64编码的签名证书字符串
	空串	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.4.9 证书登录 SOF_Login

原型： function SOF_Login(CertID, PassWd, cb, ctx)
描述： 校验证书口令,进行用户认证。
参数：

	CertID	证书唯一标识
	PassWd	证书口令
	cb	回调函数
	ctx	回调函数所需参数
返回值：	true	成功
	false	失败,可通过SOF_GetLastError获取符合表A.1定义的错误代码

注：证书登录成功后,表示该证书用户已拥有私钥使用权限,即登录状态。可正常调用签名、解密等需要私钥使用权限的接口,获得计算结果。

7.4.10 获取证书口令剩余重试次数 SOF_GetPinRetryCount

原型： function SOF_GetPinRetryCount(CertID, cb, ctx)
描述： 获取证书口令剩余重试次数。
参数：

	CertID	证书唯一标识
	cb	回调函数
	ctx	回调函数所需参数
返回值：	剩余口令重试次数,当重试次数小于或等于0时表示证书口令已被锁定	

7.4.11 修改证书口令 SOF_ChangePassWd

原型： function ChangeUserPassword(CertID, OldPassWd, NewPassWd, cb, ctx)
描述： 修改证书口令。
参数：

	CertID	证书唯一标识
	OldPassWd	旧证书口令
	NewPassWd	新证书口令
	cb	回调函数
	ctx	回调函数所需参数
返回值：	true	修改口令成功
	false	失败,可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.4.12 导出用户加密证书 SOF_ExportExChangeUserCert

原型： function SOF_ExportExChangeUserCert(CertID, cb, ctx)
描述： 根据证书唯一标识,获取Base64编码的加密(交换)证书字符串。
参数：

	CertID	证书唯一标识
	cb	回调函数
	ctx	回调函数所需参数
返回值：	用户加密证书	成功,返回Base64编码的加密(交换)证书字符串
	空串	失败,可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.4.13 获得证书基本信息 SOF_GetCertInfo

原型：function SOF_GetCertInfo(Base64Cert, Type, cb, ctx)

描述：获取证书基本信息。

参数：

Base64Cert	Base64编码的证书
Type	证书解析标识，应符合GM/T 0006
cb	回调函数
ctx	回调函数所需参数

返回值：

证书信息	成功，返回证书内指定类型的信息
空串	失败或证书中不存在该项内容

7.4.14 获得证书扩展信息 SOF_GetCertInfoByOid

原型：function SOF_GetCertInfoByOid(Base64Cert, Oid, cb, ctx)

描述：根据OID获取证书扩展项信息。

参数：

Base64Cert	Base64编码的证书
Oid	私有扩展对象ID，如“1.2.156.xxx”
cb	回调函数
ctx	回调函数所需参数

返回值：

证书扩展信息	成功，返回证书私有扩展项OID对应的信息
空串	失败或证书中不存在该私有扩展项

7.4.15 获得设备详细信息 SOF_GetDeviceInfo

原型：function SOF_GetDeviceInfo(CertID, Type, cb, ctx)

描述：根据证书唯一标识和类型代码获得设备信息。

参数：

CertID	证书唯一标识
Type	设备信息的类型，应符合GM/T 0006
cb	回调函数
ctx	回调函数所需参数

返回值：

设备信息	成功，返回Type对应的设备信息
空串	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.4.16 验证证书有效性 SOF_ValidateCert

原型：function SOF_ValidateCert(Base64Cert, cb, ctx)

描述：验证证书有效性。

参数：

Base64Cert	Base64编码格式的证书
cb	回调函数
ctx	回调函数所需参数

返回值：

SAR_OK	验证成功
其他	验证失败，失败原因应符合附录A中表A.1中的错误代码范围0X0B000500～0X0B000505

注：本接口传入的证书必须是Base64格式的串，一般通过获取签名（或加密）证书来取得。验证证书有效性在本地完成，可信根证书列表由实现者管理。

基本的证书验证策略包括：

a) 验证CA信任列表，各层都要进行签名验证；

b) 各层证书的有效期验证；

c) 各层证书的吊销状态。在特殊情况下（如：网络条件不允许），证书的吊销状态可采取灵活方式，由应用系统各层内部维护一个吊销列表，在证书登录认证时应用该吊销列表。验证证书有效性也可采取代理验证方式。

7.4.17 数据签名 SOF_SignData

原型： function SOF_SignData(CertID, InData, cb, ctx)

描述： 对字符串数据进行数字签名，返回Base64编码的数据类型A签名结果。

参数：	CertID	证书唯一标识
	InData	签名原文
	cb	回调函数
	ctx	回调函数所需参数
返回值：	签名值	成功，返回Base64编码的签名值
	空串	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

注：本接口需在登录状态下才能成功返回签名值，否则返回空串。

7.4.18 验证数据签名 SOF_VerifySignedData

原型： function SOF_VerifySignedData(Base64Cert, InData, SignValue, cb, ctx)

描述： 验证数据签名，签名值为Base64编码的数据类型A。

参数：	Base64Cert	Base64编码的签名者证书
	InData	待验证的原文
	SignValue	Base64编码的签名值
	cb	回调函数
	ctx	回调函数所需参数
返回值：	true	成功
	false	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.4.19 数据加密 SOF_EncryptData

原型： function SOF_EncryptData(Base64Cert, InData, cb, ctx)

描述： 数字信封加密，加密过程为使用临时产生的对称密钥加密数据，然后使用数字证书的公钥加密对称密钥，返回Base64编码的数据类型B密文数据。

参数：	Base64Cert	Base64编码的数据接收者的加密证书，如使用多个证书对数据加密，证书之间用&&&作为分隔符连接
	InData	待加密的数据
	cb	回调函数
	ctx	回调函数所需参数
返回值：	密文数据	成功，返回Base64编码格式的密文数据

	空串	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.4.20 数据解密 SOF_DecryptData

原型：function SOF_DecryptData(CertID, InData, cb, ctx)

描述：使用证书对应的私钥解密数字信封，密文数据格式为Base64编码的数据类型B。

参数：	CertID	证书唯一标识
	Indata	待解密的Base64编码格式的密文数据
	cb	回调函数
	ctx	回调函数所需参数
返回值：	明文数据	成功，返回解密后的明文
	空串	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

注：本接口需在登录状态下才能解密得到明文，否则返回空串。

7.4.21 消息签名 SOF_SignMessage

原型：function SOF_SignMessage(Flag, CertID, InData, cb, ctx)

描述：对字符串数据进行消息签名，返回Base64编码的数据类型B签名结果。

参数：	Flag	是否为Detached的标识，取值范围： a） 1，表示Detached，即不带原文； b） 0，表示Attached，即带原文
	CertID	证书唯一标识
	InData	待签名的数据
	cb	回调函数
	ctx	回调函数所需参数
返回值：	消息签名值	成功，返回Base64编码的签名值
	空串	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

注：本接口需在登录状态下才能成功返回签名值，否则返回空串。

7.4.22 验证消息签名 SOF_VerifySignedMessage

原型：function SOF_VerifySignedMessage(SignedMessage, InData, cb, ctx)

描述：验证消息签名，签名值的格式为Base64编码的数据类型B。

参数：	SignedMessage	Base64编码的签名值
	InData	待验证的原文，如果签名中包含原文，忽略本参数
	cb	回调函数
	ctx	回调函数所需参数
返回值：	true	成功
	false	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.4.23 解析消息签名 SOF_GetInfoFromSignedMessage

原型：function SOF_GetInfoFromSignedMessage(SignedMessage, Type, cb, ctx)

描述： 解析消息签名值中的信息，包括：原文、签名值、签名证书等，消息签名值格式为Base64编码的数据类型B。

参数：

	SignedMessage	Base64编码的签名值
	Type	类型，取值范围： a） 1解析出原文； b） 2解析出Base64编码的签名者证书； c） 3解析出Base64编码的签名值
	cb	回调函数
	ctx	回调函数所需参数

返回值：

解析结果	成功，返回解析出的对应信息
空串	失败或不存在该项

7.4.24 XML数字签名 SOF_SignDataXML

原型： function SOF_SignDataXML(CertID, InXMLData, cb, ctx)

描述： 对XML数据进行数字签名，证书为RSA算法时签名结果符合RFC3275，证书为SM2算法时签名结果符合GB/T 25061。

参数：

CertID	证书唯一标识
InXMLData	待签名的XML数据
cb	回调函数
ctx	回调函数所需参数

返回值：

XML签名结果	成功，返回XML签名结果
空串	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

注：本接口需在登录状态下才能返回签名值，否则返回空串。

缺省配置和参数：

a） 采用封皮签名，签名前应对签名原文做规范化；

b） 规范化方法采用带注释的XML规范化1.1，证书为RSA算法时标识符为http://www.w3.org/2006/12/xml-c14n11#WithComment，证书为SM2算法时标识符为http://127.0.0.1/2006/12/xml-c14n11#WithComments。

7.4.25 验证XML数字签名 SOF_VerifySignedDataXML

原型： function SOF_VerifySignedDataXML(SignedXMLData, cb, ctx)

描述： 验证XML签名。

参数：

SignedXMLData	XML签名结果
cb	回调函数
ctx	回调函数所需参数

返回值：

true	成功
false	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.4.26 解析XML签名数据 SOF_GetXMLSignatureInfo

原型： function SOF_GetXMLSignatureInfo(XMLSignedData, Type, cb, ctx)

描述： 解析XML签名数据，获取签名值、XML原文、证书和相关算法等信息。

参数：	XMLSignedData	XML签名结果
	Type	待解析的参数类型，取值范围： a） 1解析出XML原文； b） 2解析出摘要值； c） 3解析出签名值； d） 4解析出签名证书； e） 5解析出摘要算法； f） 6解析出签名算法
	cb	回调函数
	ctx	回调函数所需参数
返回值：	XML信息	成功，返回根据Type解析出各项对应的信息
	空串	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

注：XML签名缺省配置和参数值样例：

a） XML原文取自Object元素；

b） 摘要值取自DigestValue元素，形如http://127.0.0.1/2001/04/xmldsig-more#sm3；

c） 签名值取自SignatureValue元素，为BASE64编码；

d） 签名证书取自X509Data元素，为BASE64编码；

e） 摘要算法取自DigestMethod元素的Algorithm属性，形如http://127.0.0.1/2001/04/xmldsig-more#sm3；

f） 签名算法取自SignatureMethod元素的Algorithm属性，形如http://127.0.0.1/2001/04/xmldsig-more#sm2-sm3。

7.4.27 产生随机数 SOF_GenRandom

原型：	function SOF_GenRandom(RandomLen, cb, ctx)	
描述：	产生指定长度的随机数。	
参数：	RandomLen	待产生随机数的长度(字节数)
	cb	回调函数
	ctx	回调函数所需参数
返回值：	随机数	成功，返回Base64编码的随机数值
	空串	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.4.28 获取最新的错误信息 SOF_GetLastError

原型：	function SOF_GetLaseError(cb, ctx)	
描述：	获取最新的错误码。	
参数：	cb	回调函数
	ctx	回调函数所需参数
返回值：	错误代码，应符合表A.1错误代码表	

7.4.29 计算数据摘要 SOF_HashData

原型： function SOF_HashData(HashAlg, InData, SignCert, UserID, cb, ctx)

描述：计算数据摘要，如果是SM3算法，当SignCert和UsreID为空时，只计算数据的摘要值，当SignCert和UsreID值不为空时，应按照GM/T 0009规定的预处理过程计算，摘要值可以作为SM2签名的输入。

参数：

HashAlg	摘要算法，应符合GM/T 0006
InData	原始数据
SignCert	Base64编码的签名者证书，摘要算法为SM3时有效，如不需要传空串
UserID	签名者用户ID，摘要算法为SM3时有效，如果SignCert参数为空，则本参数无意义
cb	回调函数
ctx	回调函数所需参数

返回值：

摘要值	成功，返回Base64编码的数据摘要值
空串	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.4.30 摘要数据签名 SOF_SignHashData

原型：function SOF_SignHashData(CertID, Base64HashData, HashAlg, cb, ctx)

对数据摘要签名，返回Base64编码的数据类型A签名结果。

描述：Base64HashData值一般是SOF_HashData或SOF_HashFile函数的计算结果。如果是SM2签名，SM3算法的摘要值应按照GM/T 0009规定的预处理过程计算。

参数：

CertID	证书唯一标识
Base64HashData	Base64编码的摘要值
HashAlg	摘要算法，应符合GM/T 0006
cb	回调函数
ctx	回调函数所需参数

返回值：

签名值	成功，返回Base64编码的数据类型A
空串	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

注：本接口需在登录状态下才能成功返回签名值，否则返回空串。

7.4.31 数据摘要签名验证 SOF_VerifySignedHashData

原型：function SOF_VerifySignedHashData(
Base64Cert, Base64HashData, SignValue, HashAlg, cb, ctx)

数据摘要签名验证，签名值为Base64编码的数据类型A。

描述：如果使用SM2算法验证签名，Base64HashData参数是SM3算法的摘要值，应按照GM/T 0009规定的预处理过程计算。

参数：

Base64Cert	Base64编码签名者证书
Base64HashData	Base64编码摘要值
SignValue	Base64编码的签名值
HashAlg	摘要算法，应符合GM/T 0006
cb	回调函数
ctx	回调函数所需参数

返回值：

true	成功
false	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

7.4.32 证书登出 SOF_Logout

原型： function SOF_Logout(CertID, cb, ctx)

描述： 退出登录状态。

参数：

	CertID	证书唯一标识
	cb	回调函数
	ctx	回调函数所需参数

返回值：

true	成功
false	失败，可通过SOF_GetLastError获取符合表A.1定义的错误代码

注： 证书登出表示释放该证书用户对应的私钥使用权限。证书登出后再调用签名、解密等需要拥有私钥使用权限才能正常操作的接口时，需再次成功调用SOF_Login接口获取私钥使用权限。

7.4.33 用户登录状态检测 SOF_IsLogin

原型： function SOF_IsLogin(CertID, cb, ctx)

描述： 判断证书用户是否为登录状态。

参数：

	CertID	证书唯一标识
	cb	回调函数
	ctx	回调函数所需参数

返回值：

true	登录状态
false	非登录状态

附 录 A
（规范性）
证书应用综合服务接口错误代码定义

本文件所定义的错误代码符合 GM/T 0094—2020 错误代码区间划分要求。COM 组件错误代码定义见表 A.1，JAVA 组件异常信息定义见表 A.2。

表 A.1 COM 组件的错误代码表

宏描述	预定义值	说明
SOR_OK	0	成功
SOR_UnknownErr	0X0B000001	异常错误
SOR_NotSupportYetErr	0X0B000002	不支持的服务
SOR_FileErr	0X0B000003	文件操作错误
SOR_ProviderTypeErr	0X0B000004	服务提供者参数类型错误
SOR_LoadProviderErr	0X0B000005	导入服务提供者接口错误
SOR_LoadDevMngApiErr	0X0B000006	导入设备管理接口错误
SOR_AlgoTypeErr	0X0B000007	算法类型错误
SOR_NameLenErr	0X0B000008	名称长度错误
SOR_KeyUsageErr	0X0B000009	密钥用途错误
SOR_ModulusLenErr	0X0B000010	模的长度错误
SOR_NotInitializeErr	0X0B000011	未初始化
SOR_ObjErr	0X0B000012	对象错误
SOR_MemoryErr	0X0B000100	内存错误
SOR_TimeoutErr	0X0B000101	服务超时
SOR_IndataLenErr	0X0B000200	输入数据长度错误
SOR_IndataErr	0X0B000201	输入数据错误
SOR_GenRandErr	0X0B000300	生成随机数错误
SOR_HashObjErr	0X0B000301	HASH对象错
SOR_HashErr	0X0B000302	HASH运算错误
SOR_GenRsaKeyErr	0X0B000303	产生RSA密钥错
SOR_RsaModulusLenErr	0X0B000304	RSA密钥模长错误
SOR_CspImprtPubKeyErr	0X0B000305	CSP服务导入公钥错误
SOR_RsaEncErr	0X0B000306	RSA加密错误
SOR_RSGDecErr	0X0B000307	RSA解密错误
SOR_HashNotEqualErr	0X0B000308	HASH值不相等
SOR_KeyNotFountErr	0X0B000309	密钥未发现
SOR_CertNotFountErr	0X0B000310	证书未发现

表 A.1 COM 组件的错误代码表(续)

宏描述	预定义值	说明
SOR_NotExportErr	0X0B000311	对象未导出
SOR_VeryPolicyErr	0X0B000312	未能完全按照策略验证成功
SOR_DecryptPadErr	0X0B000400	解密时做补丁错误
SOR_MacLenErr	0X0B000401	MAC长度错误
SOR_KeyInfoTypeErr	0X0B000402	密钥类型错误
SOR_NULLPointerErr	0X0B000403	某一个参数为空指针
SOR_APPNotFoundErr	0X0B000404	没有找到该应用
SOR_CERTENCODEErr	0X0B000405	证书编码格式错误
SOR_CERTINVALIDErr	0X0B000406	证书无效,不是可信CA颁发的证书
SOR_CERTHASEXPIREDErr	0X0B000407	证书已过期
SOR_CERTREVOKEDErr	0X0B000408	证书已经被吊销
SOR_SIGNDATAErr	0X0B000409	签名失败
SOR_VERIFYSIGNDATAErr	0X0B000410	验证签名失败
SOR_READFILEErr	0X0B000411	读文件异常,可能文件不存在或没有读取权限等
SOR_WRITEFILEErr	0X0B000412	写文件异常,可能文件不存在或没有写权限等
SOR_SECRETSEGMENTErr	0X0B000413	门限算法密钥分割失败
SOR_SECERTRECOVERYErr	0X0B000414	门限恢复失败
SOR_ENCRYPTDATAErr	0X0B000415	对数据的对称加密失败
SOR_DECRYPTDATAErr	0X0B000416	对称算法的数据解密失败
SOR_PKCS7ENCODEErr	0X0B000417	PKCS7编码格式错误
SOR_XMLENCODEErr	0X0B000418	不是合法的XML编码数据
SOR_PARAMETERNOTSUPPORTErr	0X0B000419	不支持的参数
SOR_CTLNOTFOUND	0X0B000420	没有发现信任列表
SOR_APPNOTFOUND	0X0B000421	设置的应用名称没发现
SOR_CERTNOTTRUST	0X0B000500	验证证书失败,证书不被信任
SOR_CERTHASEXPIRED	0X0B000501	验证证书失败,证书超过有效期范围
SOR_CERTHASREVOKED	0X0B000502	验证证书失败,证书已作废
SOR_CERTHASHOLDED	0X0B000503	验证证书失败,证书已冻结
SOR_CERTNOTVALIDYET	0X0B000504	验证证书失败,证书未生效
SOR_CERTVERIFYOTHERERR	0X0B000505	验证证书失败,其他错误

表 A.2 JAVA 组件的异常信息表

异常描述	说明
java.lang.PointerException	某一个参数为空指针
SOR_InitException	初始化环境失败
SOR_AppNotfoundException	没有找的该应用
SOR_CertEncodeException	证书编码格式错误
SOR_CertInvalidException	证书无效,不是可信CA颁发的证书
SOR_CertNotYetValidException	证书未生效
SOR_CertHasExpiredException	证书已过期
SOR_CertRevokedException	证书已经被吊销
SOR_SignDataException	签名失败
SOR_VerifySignDataException	验证签名失败
SOR_ReadFileException	读文件异常,可能文件不存在或没有读取权限等
SOR_WriteFileException	写文件异常,可能文件不存在或没有写权限等
SOR_SecretSegmentException	门限分割算法失败
SOR_SecertRecoveryException	门限恢复失败
SOR_EncryptDataException	数据加密失败
SOR_DecryptDataException	数据解密失败
SOR_Pkcs7EncodeException	PKCS#7编码格式错误
SOR_XmlEncodeException	不是合法的XML编码数据
SOR_ParameterNotSupportException	不支持的参数
SOR_EncryptDataException	数据加密失败
SOR_DecryptDataException	数据解密失败
SOR_MessageEncodeException	消息编码格式错误
SOR_XmlEncodeException	不是合法的XML编码数据

附 录 B
（资料性）
证书应用综合服务接口典型部署模型

基于 B/S 架构应用系统的证书应用综合服务接口的部署示意图见图 B.1。

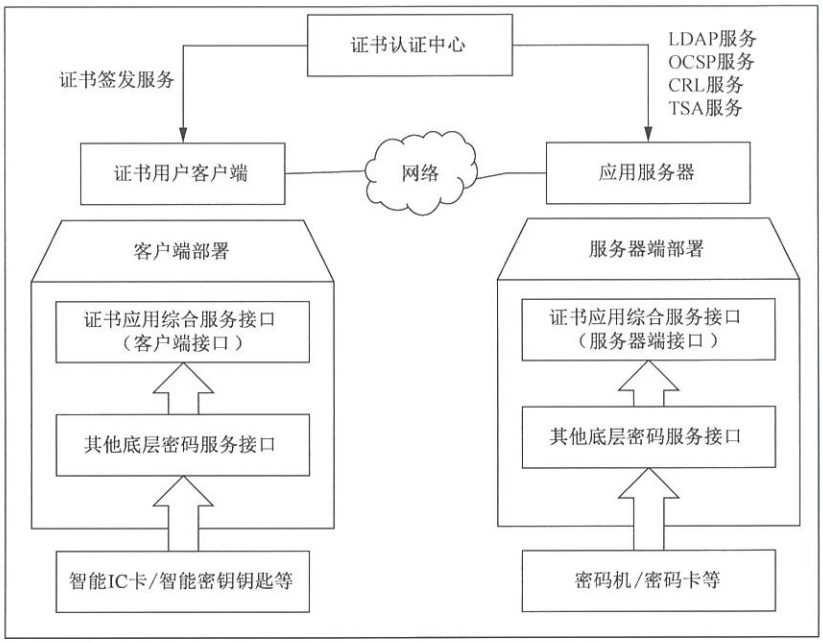

图 B.1 B/S 结构应用系统的典型证书应用接口部署示意图

在应用服务器端应部署的软件包括：证书应用综合服务接口（服务器端接口，接口形态一般为 COM 组件或 JAVA 组件两类）和其他底层密码服务接口（包括：密码设备接口和通用密码服务接口等）。应部署的硬件是密码设备，如密码机或密码卡，用于服务器端的签名、验证、加密、解密等密码运算。

证书用户客户端可以是 PC 机、手机或其他智能移动终端，部署的软件包括：证书应用综合服务接口（客户端接口，接口形态一般为 COM 组件、动态库或 JavaScript 等形态，随着技术的发展接口形态可以进行扩展）和其他底层密码服务接口（主要包括智能 IC 卡/智能密码钥匙应用接口和通用密码服务接口、证书载体驱动程序等）。应部署的硬件是密码设备，如智能 IC 卡或智能密码钥匙（USBKey），用于客户端的签名、验证、加密、解密等密码运算。

附 录 C
（资料性）
证书应用综合服务接口集成示例

典型证书登录认证流程图见图 C.1。

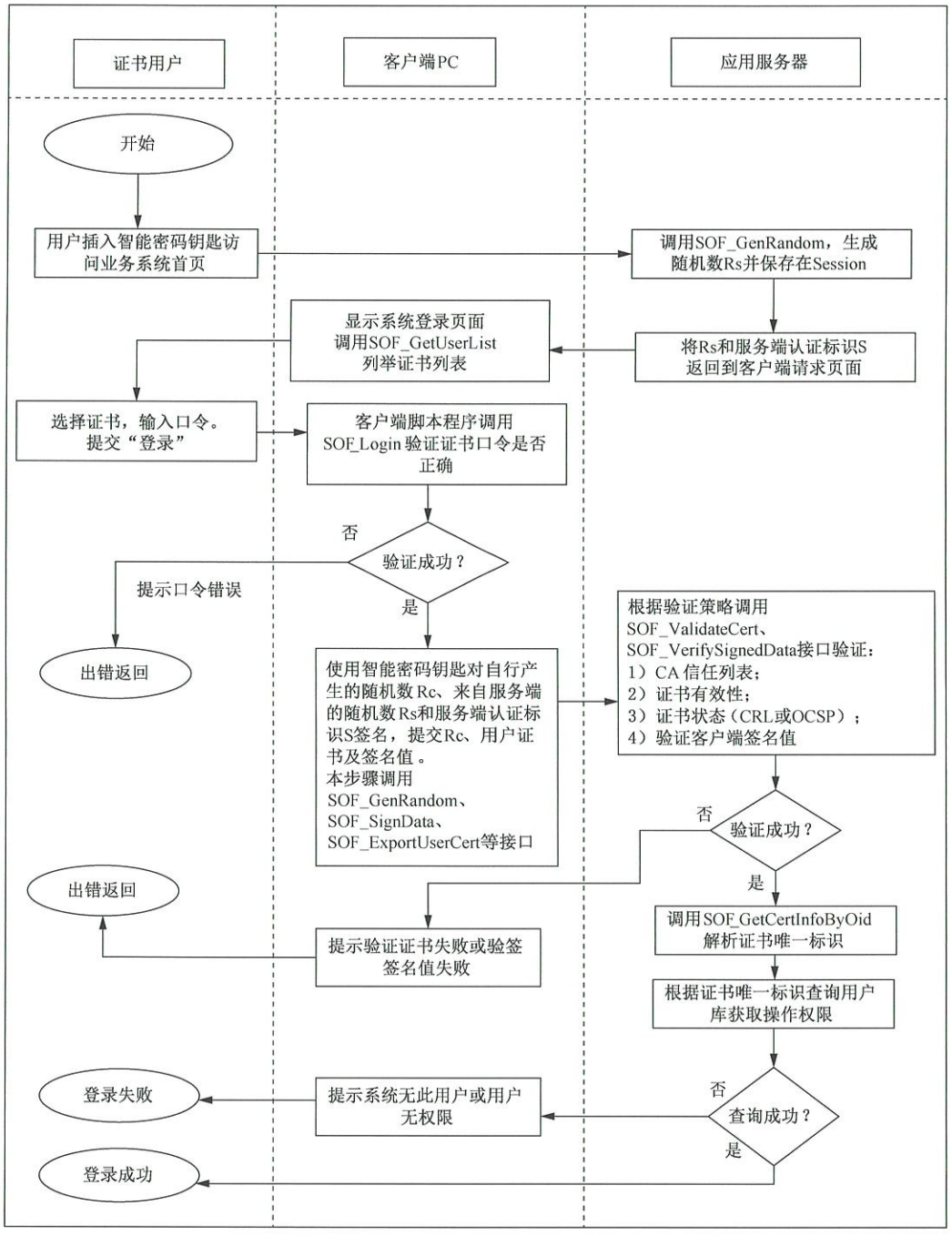

图 C.1 证书登录认证流程示意图

基于表单业务数据的签名和验签流程见图 C.2。

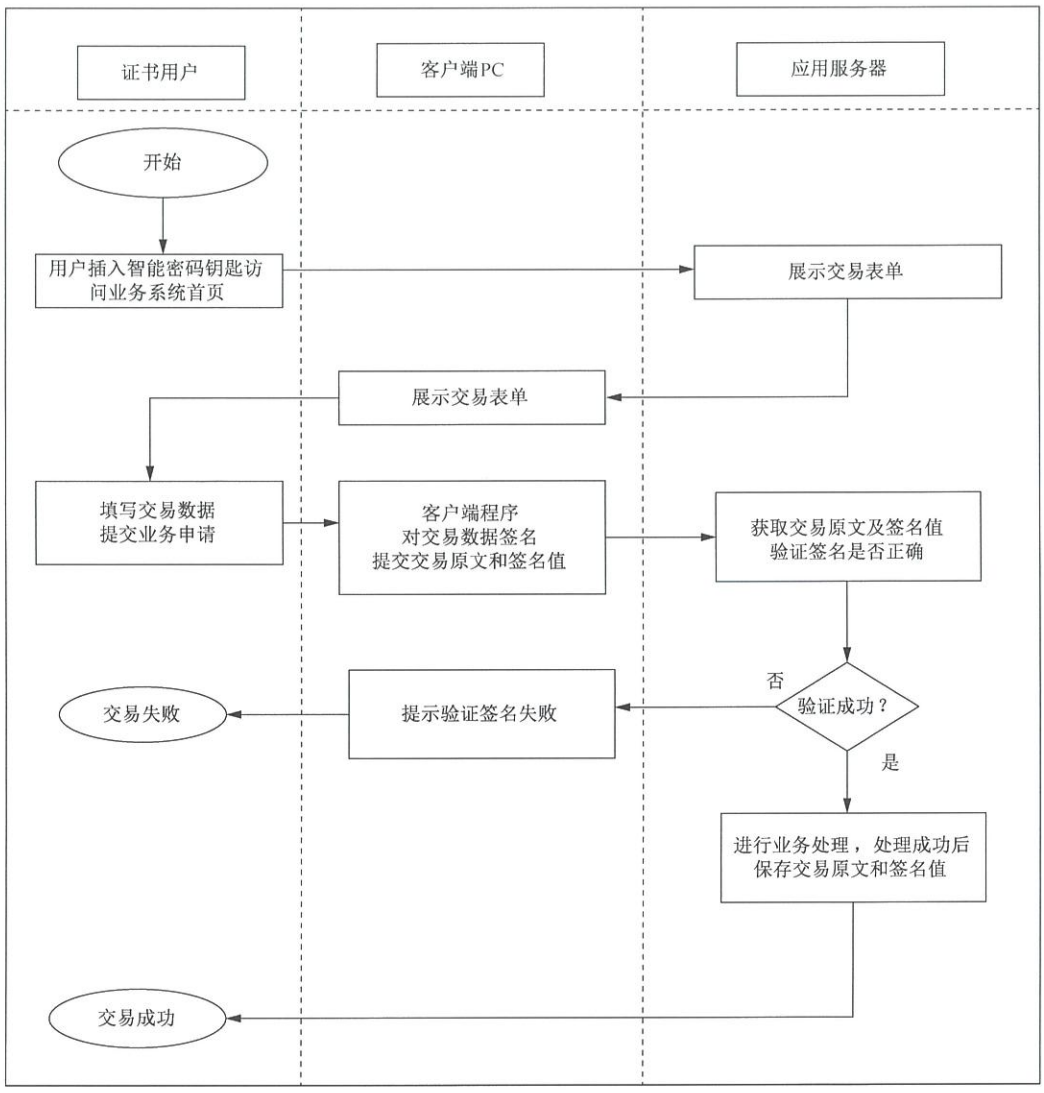

图 C.2　业务数字签名验证流程图

基于表单业务数据的签名和验签流程，需首先按照图 C.1 完成证书登录认证流程。登录认证成功后，业务系统展示交易表单，证书用户根据业务系统展示的表单填写业务交易数据，检查无误后提交业务申请。提交业务申请时需对业务交易数据做数字签名，根据业务数据不同可选择调用不同的签名接口：

- 如果业务数据是普通数据，调用 SOF_SignData 接口；
- 如果业务数据存放在文件中，调用 SOF_SignFile 接口；
- 如果业务数据是 XML 格式，调用 SOF_SignDataXML 接口；
- 如果业务数据是消息，调用 SOF_SignMessage 接口；
- 如果业务数据仅以 Hash 方式提交，先调用 SOF_HashData 或 SOF_HashFile，再调用 SOF_SignHashData 接口。

业务交易数据和签名值一起提交到业务系统服务器。业务系统服务器调用对应的验证接口，验证正确后进行业务处理，完成后提示用户交易成功。

附 录 D
（资料性）
客户端JavaScript脚本接口异步调用示例说明

为保证多种类型浏览器的兼容性，7.2 客户端脚本接口采用异步调用方式定义。每个功能接口都对应一个回调函数和一个回调参数对象，回调函数的原型为：

```
function callback(result, ctx)
```

功能接口的返回值通过 result 参数传递给回调函数，对功能接口返回值的业务处理代码在回调函数内完成。如果回调函数不需要 ctx 参数，调用功能接口时可传递空值或忽略。

使用示例：

```
function SOF_GetVersion_cb(result, ctx) {
    alert("接口版本号为:" + result);
    alert("调用者传入的回调参数" + ctx);
}
SOF_GetVersion(SOF_GetVersion_cb);
```

所有功能接口的返回值都通过回调函数的 result 返回，功能接口本身没有返回值。如上述调用，在获取版本号时，应在回调函数 SOF_GetVersion_cb 中接收 SOF_GetVersion 接口的返回值并做相关业务端处理代码，如下是不正确的使用方式：

```
var strVersion = SOF_GetVersion();
```

因功能接口采用异步调用方式定义，strVersion 变量并不保证能正确获取到版本号。

参 考 文 献

[1] GB/T 19713—2005 信息技术 安全技术 公钥基础设施 在线证书状态协议

[2] GM/T 0016 智能密码钥匙密码应用接口规范

[3] GM/T 0094—2020 公钥密码应用技术体系框架规范

[4] RSA Security:Public-Key Cryptography Standards (PKCS)

[5] PKCS #11 Cryptographic Token Interface Standard

[6] IETF RFC 2459 Internet X.509 Public Key Infrastructure Certificate and CRL Profile

[7] IETF RFC 2560 X.509 Internet Public Key Infrastructure Online Certificate Status Protocol

[8] IETF RFC 1777 Lightweight Directory Access Protocol

[9] IETF RFC 2587 Internet X.509 Public Key Infrastructure LDAPv2 Schema

[10] IETF RFC 3647 Internet X.509 Public Key Infrastructure Certificate Policy and Certification Practices Framework

[11] ISO/IEC 8825-1:2021 Information technology—ASN.1 encoding rules—Part 1: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)