CFCA RSA 数字证书申请及证书应用

CSP 接口调用规范

（版本：3.0.0）

中国金融认证中心

2012 年 10 月 19 日

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
3.0.0	初稿完成	2012/10/19	李哲	

目　录

1. 概述

1.1 文档描述

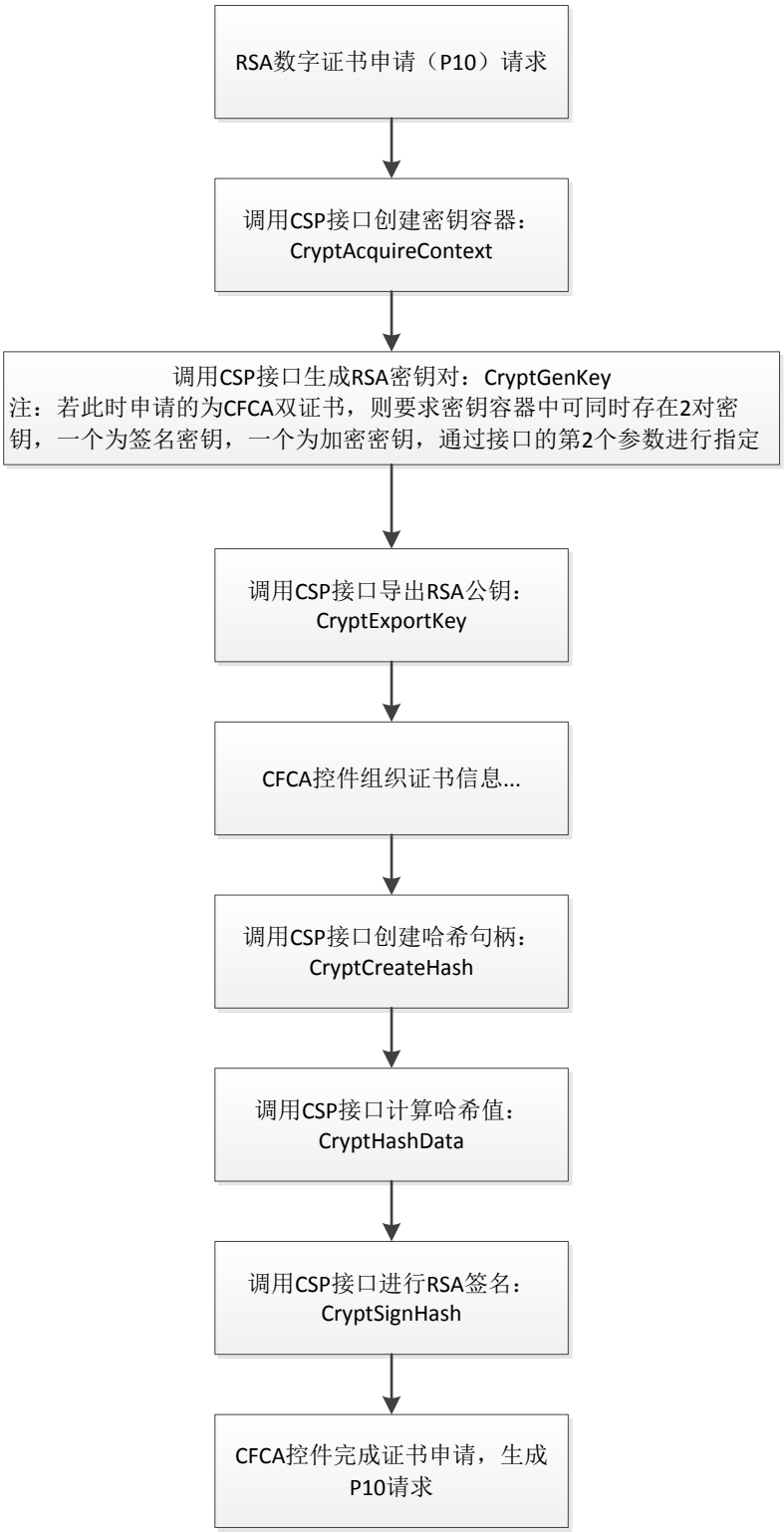
本文档描述了：

1、 申请 CFCA RSA 数字证书所需调用的 CSP 接口。

2、 导入 CFCA RSA 数字证书（包括签名证书及加密证书）所需调用的 CSP 接口。

2. RSA 数字证书申请调用 CSP 接口规范

2.1 RSA 数字证书申请流程


RSA数字证书申请（P10）请求
调用CSP接口创建密钥容器：
CryptAcquireContext
调用CSP接口生成RSA密钥对：CryptGenKey
注：若此时申请的为CFCA双证书，则要求密钥容器中可同时存在2对密钥，一个为签名密钥，一个为加密密钥，通过接口的第2个参数进行指定
调用CSP接口导出RSA公钥：
CryptExportKey
CFCA控件组织证书信息...
调用CSP接口创建哈希句柄：
CryptCreateHash
调用CSP接口计算哈希值：
CryptHashData
调用CSP接口进行RSA签名：
CryptSignHash
CFCA控件完成证书申请，生成P10请求

2.2 CSP 接口描述

2.2.1 CryptAcquireContext

```
BOOL WINAPI CryptAcquireContext(
  __out  HCRYPTPROV *phProv,
  __in   LPCTSTR pszContainer,
  __in   LPCTSTR pszProvider,
  __in   DWORD dwProvType,
  __in   DWORD dwFlags)
```

描述：

创建密钥容器。

参数取值说明：

pszContainer：待创建的密钥容器的名称

dwProvType：PROV_RSA_FULL

dwFlags：CRYPT_NEWKEYSET

2.2.2 CryptGenKey

```
BOOL WINAPI CryptGenKey(
  __in   HCRYPTPROV hProv,
  __in   ALG_ID Algid,
  __in   DWORD dwFlags,
  __out  HCRYPTKEY *phKey)
```

描述：

生成 RSA 密钥对。

参数取值说明：

Algid：AT_SIGNATURE　签名密钥；AT_KEYEXCHANGE 加密密钥，

2.2.3 CryptExportKey

```
BOOL WINAPI CryptExportKey(
  __in     HCRYPTKEY hKey,
  __in     HCRYPTKEY hExpKey,
  __in     DWORD dwBlobType,
  __in     DWORD dwFlags,
  __out    BYTE *pbData,
  __inout  DWORD *pdwDataLen)
```

描述：

导出 RSA 公钥。

参数取值说明：

dwBlobType：PUBLICKEYBLOB

pbData：公钥数据

2.2.4 CryptCreateHash

```
BOOL WINAPI CryptCreateHash(
  __in   HCRYPTPROV hProv,
  __in   ALG_ID Algid,
  __in   HCRYPTKEY hKey,
  __in   DWORD dwFlags,
  __out  HCRYPTHASH *phHash)
```

描述：

创建哈希句柄。

参数取值说明：

Algid：表示哈希算法

2.2.5 CryptHashData

```
BOOL WINAPI CryptHashData(
  __in  HCRYPTHASH hHash,
```

```
  __in  BYTE *pbData,
  __in  DWORD dwDataLen,
  __in  DWORD dwFlags)
```

描述：

对数据进行哈希运算。

参数取值说明：

无

2.2.6 CryptSignHash

```
BOOL WINAPI CryptSignHash(
  __in     HCRYPTHASH hHash,
  __in     DWORD dwKeySpec,
  __in     LPCTSTR sDescription,
  __in     DWORD dwFlags,
  __out    BYTE *pbSignature,
  __inout  DWORD *pdwSigLen)
```

描述：

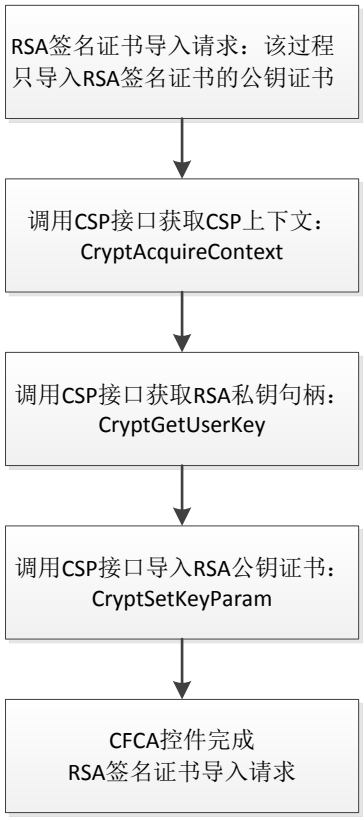
对哈希值进行 RSA 签名。

参数取值说明：

dwKeySpec：AT_KEYEXCHANGE 或 AT_SIGNATURE

3. RSA 数字证书导入调用 CSP 接口规范

3.1 RSA 签名证书导入流程



3.1.1 CSP 接口描述

3.1.1.1 CryptAcquireContext

```
BOOL WINAPI CryptAcquireContext(
  __out  HCRYPTPROV *phProv,
  __in   LPCTSTR pszContainer,
  __in   LPCTSTR pszProvider,
  __in   DWORD dwProvType,
  __in   DWORD dwFlags)
```

描述：

获取 CSP 上下文。

参数取值说明：

pszContainer：待获取的密钥容器的名称

dwProvType：PROV_RSA_FULL

dwFlags：CRYPT_VERIFYCONTEXT

3.1.1.2 CryptGetUserKey

```
BOOL WINAPI CryptGetUserKey(
  __in   HCRYPTPROV hProv,
  __in   DWORD dwKeySpec,
  __out  HCRYPTKEY *phUserKey)
```

描述：

获得私钥句柄。

参数取值说明：

dwKeySpec：AT_KEYEXCHANGE 或 AT_SIGNATURE

3.1.1.3 CryptSetKeyParam

```
BOOL WINAPI CryptSetKeyParam(
  __in  HCRYPTKEY hKey,
  __in  DWORD dwParam,
  __in  const BYTE *pbData,
  __in  DWORD dwFlags)
```

描述：

导入 RSA 公钥证书。

参数取值说明：

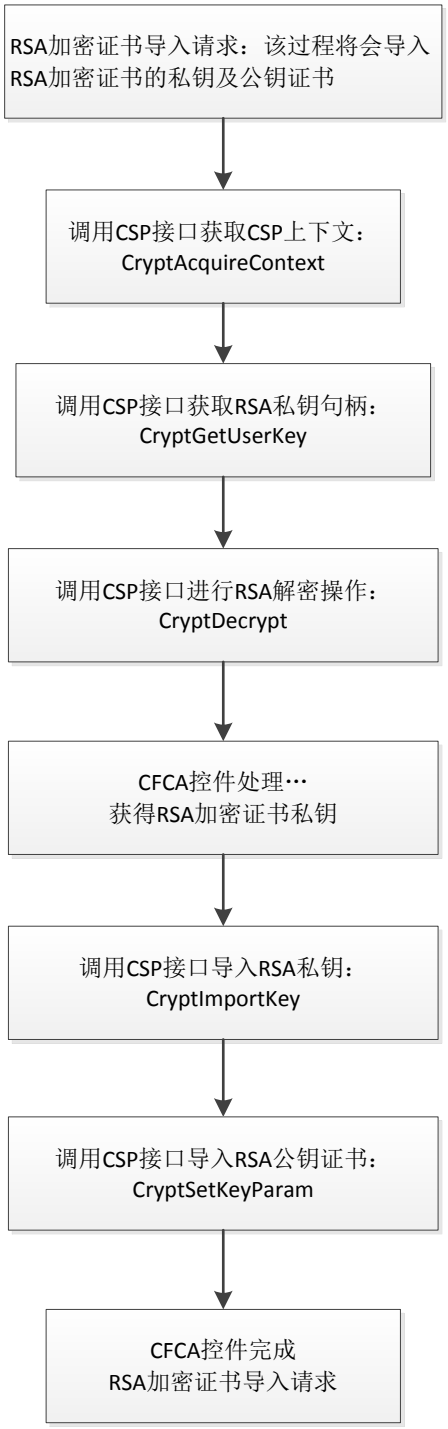
hKey：RSA私钥句柄

dwParam：KP_CERTIFICATE

3.2 RSA 加密证书导入流程

1、安装 1024bit 的 RSA 加密证书时，RSA 私钥以明文方式导入 CSP。
2、安装 2048/4096bit 的 RSA 加密证书时，RSA 私钥以密文的方式导入 CSP。

3.2.1 1024bit RSA 加密证书导入流程及 CSP 接口描述



3.2.1.1 CryptAcquireContext

```
BOOL WINAPI CryptAcquireContext(
  __out  HCRYPTPROV *phProv,
```

```
  __in    LPCTSTR pszContainer,
  __in    LPCTSTR pszProvider,
  __in    DWORD dwProvType,
  __in    DWORD dwFlags)
```

描述：

获取 CSP 上下文。

参数取值说明：

pszContainer：待获取的密钥容器的名称

dwProvType：PROV_RSA_FULL

dwFlags：CRYPT_VERIFYCONTEXT

3.2.1.2 CryptGetUserKey

```
BOOL WINAPI CryptGetUserKey(
  __in    HCRYPTPROV hProv,
  __in    DWORD dwKeySpec,
  __out   HCRYPTKEY *phUserKey)
```

描述：

获得 RSA 私钥句柄。

参数取值说明：

dwKeySpec：AT_KEYEXCHANGE 或 AT_SIGNATURE

3.2.1.3 CryptDecrypt

```
BOOL WINAPI CryptDecrypt(
  __in      HCRYPTKEY hKey,
  __in      HCRYPTHASH hHash,
  __in      BOOL Final,
  __in      DWORD dwFlags,
  __inout   BYTE *pbData,
  __inout   DWORD *pdwDataLen)
```

描述：

RSA 解密。

参数取值说明：

hKey：RSA私钥句柄

dwFlags：0

3.2.1.4 CryptImportKey

```
BOOL WINAPI CryptImportKey(
  __in    HCRYPTPROV hProv,
  __in    BYTE *pbData,
  __in    DWORD dwDataLen,
  __in    HCRYPTKEY hPubKey,
  __in    DWORD dwFlags,
  __out   HCRYPTKEY *phKey)
```

描述：

导入 RSA 私钥（以明文方式）。

参数取值说明：

pbData：RSA私钥数据。

导入的私钥结构为微软标准的Private key BLOBs，可参考：

http://msdn.microsoft.com/en-us/library/windows/desktop/aa375601(v=vs.85).aspx

3.2.1.5 CryptSetKeyParam

```
BOOL WINAPI CryptSetKeyParam(
  __in  HCRYPTKEY hKey,
  __in  DWORD dwParam,
  __in  const BYTE *pbData,
  __in  DWORD dwFlags)
```

描述：

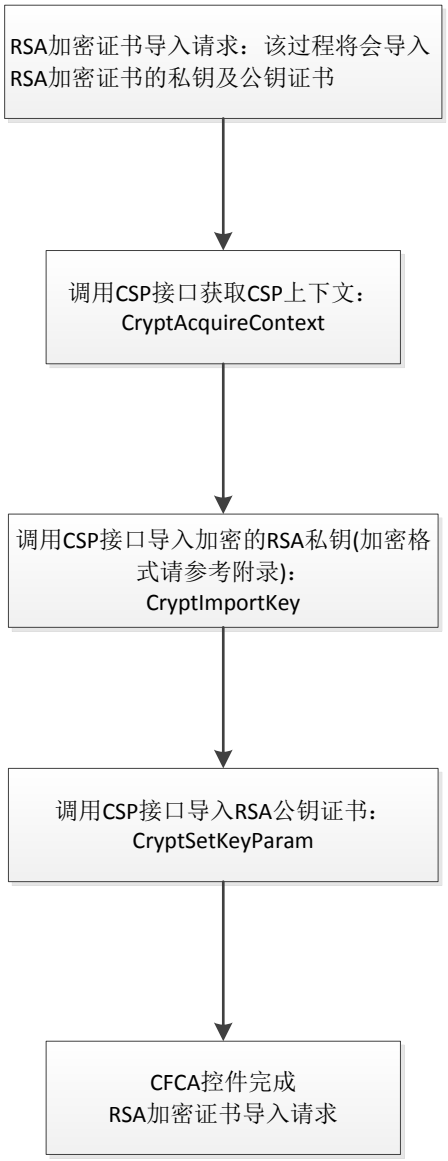
导入 RSA 公钥证书。

参数取值说明：

hKey：RSA私钥句柄

dwParam：KP_CERTIFICATE

3.2.2 2048bit/4096bit RSA 加密证书导入流程及 CSP 接口描述



3.2.2.1 CryptAcquireContext

```
BOOL WINAPI CryptAcquireContext(
  __out  HCRYPTPROV *phProv,
  __in   LPCTSTR pszContainer,
  __in   LPCTSTR pszProvider,
  __in   DWORD dwProvType,
  __in   DWORD dwFlags)
```

描述：

获取 CSP 上下文。

参数取值说明：

pszContainer：待获取的密钥容器的名称

dwProvType：PROV_RSA_FULL

dwFlags：CRYPT_VERIFYCONTEXT

3.2.2.2 CryptImportKey

```
BOOL WINAPI CryptImportKey(
  __in   HCRYPTPROV hProv,
  __in   BYTE *pbData,
  __in   DWORD dwDataLen,
  __in   HCRYPTKEY hPubKey,
  __in   DWORD dwFlags,
  __out  HCRYPTKEY *phKey)
```

描述：

导入 RSA 私钥（以密文形式导入）。

参数取值说明：

pbData：加密的RSA私钥数据，密文的数据结构定义详见章节5

hPubKey:此参数为NULL（倒入的公钥存在于pbData参数中）

3.2.2.3 CryptSetKeyParam

```
BOOL WINAPI CryptSetKeyParam(
```

```
  __in  HCRYPTKEY hKey,
  __in  DWORD dwParam,
  __in  const BYTE *pbData,
  __in  DWORD dwFlags)
```

描述：

导入 RSA 公钥证书。

参数取值说明：

hKey：RSA私钥句柄（此处的句柄是3.2.1.2小节中CryptImportKey函数传出的*phKey）

dwParam：KP_CERTIFICATE

5. RSA 私钥结构定义

5.1 RSA 加密证书导入私钥密文格式

加密后的 RSA 私钥格式如下：

BLOBHEADER;

RSAPRIVATEKEYBLOB

其中 BLOBHEADER 为微软标准定义；RSAPRIVATEKEYBLOB 为自定义数据结构

①BLOBHEADER 结构取值如下：

```
typedef struct _PUBLICKEYSTRUC
{
  BYTE   bType;    //取值为：PRIVATEKEYBLOB (0x7)
  BYTE   bVersion; //取值为：CUR_BLOB_VERSION (0x2)
  WORD   reserved; //取值为：0x1—代表私钥是加密的格式（RSA2048、4096
                     使用加密方式）
  ALG_ID aiKeyAlg; //取值为：CALG_RSA_KEYX
} BLOBHEADER, PUBLICKEYSTRUC;
```

②RSAPRIVATEKEYBLOB 结构取值如下：

```
typedef struct _RSAPRIVATEKEYBLOB
```

```
{
    ULONG AlgID;    //取值为：CALG_RSA_KEYX
    ULONG BitLen;   //RSA 加密证书私钥的实际位长度
    ULONG EVPPrivateKeyBitLen; //EVPPrivateKey 数据的实际位(bit)长度
    BYTE  *EVPPrivateKey;     //加密私钥或明文私钥数据
}RSAPRIVATEKEYBLOB, *PRSAPRIVATEKEYBLOB;
```

其中 EVPPrivateKey 为自定义的 ASN.1 格式（DER 编码）如下：

```
EVPPrivateKey ::= SEQUENCE {

Version INTEGER,                    --版本号

AsymAlgID OBJECT IDENTIFIER,        --非对称加密算法标识符

SymAlgID OBJECT IDENTIFIER,         --对称加密算法标识符

EncryptedSymKey OCTET STRING,       --被加密的对称密钥

EncryptedPrivateKey OCTET STRING     --用对称密钥加密过的私钥
}
```

使用 RSA 加密算法加密的 EncryptedSymKey 的格式请参见 PKCS#1 的 RSA 加密结果。

RSA 加密私钥的取值如下：

```
Version 1,                          --版本号
AsymAlgID 1.2.840.113549.1.1.1,     --RSA 加密算法
SymAlgID 1.3.6.1.4.1.4929.1.7       --3DES ECB 对称加密算法
```