新暗号対応

クライアントソフト API 仕様書
【PKCS#11 編】

第 1.0 版

電子認証局会議

変更履歴

版数	変更日付	変更内容
1.0 版	平成 26 年 1 月　7 日	新規作成

本文に関する補足

『公的個人認証サービス利用者クライアントソフト API 仕様書【カード AP ライブラリ PKCS#11 編】第 2.5 版』との差分内容を明確化するために赤字でとしている。

― 目次 ―

第１章 はじめに

本仕様は、公的個人認証サービス 利用者クライアントソフト(以下、JPKI 利用者ソフト)におけるカード AP ライブラリのうち、PKCS#11 の API 仕様をベースに新暗号対応機能について見直しを行ったものである。最終的には各社の PKCS#11 の仕様の統一を図ることで上位アプリケーションが各社固有の実装を意識することなく API を利用できることを目標とする。

本 API 機能について２通りの実装方式がある。すなわち PKCS#11 規格バージョン 2.01 をベースに新暗号対応のために必要な修正を加えたものと、新暗号に対応できる PKCS#11 規格バージョン 2.20 をベースとするものである。

第２章 ドキュメント体系

クライアントソフトのドキュメント体系を以下に示す。

・クライアントソフト API 仕様書 【PKCS#11 編】

クライアントソフト(PKCS#11)の API 仕様について説明する。

第３章 動作環境

クライアントソフト(PKCS#11)の動作環境については、各認証局が公開する情報を参照のこと。

第４章 機能仕様

第１節 ソフトウェア構成

本仕様書では、Windows 環境において上位アプリケーションの API をクライアントソフト(PKCS#11)の仕様としてまとめる。なおクライアントソフト(PKCS#11)の下位実装については規定しない。

第２節 実現可能な機能の一覧

クライアントソフト(PKCS#11)で実現可能な機能の一覧を表 4 に示す。

表 4　実現可能な機能の一覧

NO	機能	概要
1	証明書取得	IC カードに格納された電子証明書(利用者証明書、認証局の自己署名証明書)を取得する。
2	署名生成	署名対象データからハッシュ値を計算し、IC カードに格納された利用者秘密鍵を使用して電子署名を生成する。
3	署名検証（※１）	検証対象データからハッシュ値を計算し、ハッシュ値、電子署名、公開鍵を使用して電子署名を検証する。
4	繰り返し署名生成	NO2 の処理を繰り返し実行し、複数の署名対象データに対する電子署名を生成する。
5	繰り返し署名検証（※１）	NO3 の処理を繰り返し実行し、複数の電子署名を検証する。

※１：実装は任意とする。

第５章　API 仕様

第１節　サポート API 一覧

クライアントソフト(PKCS#11)のサポート API の一覧を表 5 に示す。

表 5　サポート API 一覧

NO	API 名	概要
1	C_GetFunctionList	関数ポインタリストを取得する。
2	C_Initialize	PKCS#11 ライブラリを初期化する。
3	C_Finalize	PKCS#11 ライブラリを終了する。
4	C_GetInfo	ライブラリ情報を取得する。
5	C_GetSlotList	スロットリストを取得する。
6	C_GetSlotInfo	スロット情報を取得する。
7	C_GetTokenInfo	トークン情報を取得する。
8	C_GetMechanismList	サポートメカニズム（アルゴリズム）を取得する。
9	C_GetMechanismInfo	メカニズム（アルゴリズム）情報を返す。
10	C_OpenSession	セッションを確立する。
11	C_CloseSession	セッションを切断する。
12	C_CloseAllSessions	すべてのセッションを切断する。
13	C_GetSessionInfo	セッション状態を取得する。
14	C_Login	トークンをログイン状態にする。
15	C_Logout	トークンをログアウト状態にする。
16	C_FindObjectsInit	オブジェクトの検索を開始する。
17	C_FindObjects	オブジェクトの検索を行う。
18	C_FindObjectsFinal	オブジェクトの検索を終了する。
19	C_GetAttributeValue	オブジェクトの属性値を取得する。
20	C_SignInit	署名処理を初期化する。
21	C_Sign	データに署名を行う。
22	C_DigestInit	ダイジェスト作成を開始する。（※１）
23	C_DigestUpdate	ダイジェストを作成する。（※１）
24	C_DigestFinal	ダイジェスト作成を終了する。（※１）
25	C_VerifyInit	署名検証を開始する。（※１）
26	C_Verify	署名値を検証する。（※１）
27	C_CreateObject	公開鍵オブジェクトを作成する。（※１）
28	C_DestroyObject	公開鍵オブジェクトを破棄する。（※１）

※１：実装は任意とするが、実装する場合には次節サポート API 仕様詳細に準拠することが望ましい。

第２節　サポート API 仕様詳細

各 API の「戻り値」には、上位アプリケーションで処理することが望ましい代表的な値のみを記す。ライブラリは記載のものを全て返却するとは限らず、また、記載のもの以外を返却してもよい。各 API が戻り値として返却しうる値の完全なリストは下記 URL にある PKCS#11 Ver2.01 および Ver2.20 の正式な資料を参照のこと。

RSA　Laboratories

「http://www.emc.com/emc-plus/rsa-labs/standards-initiatives/pkcs-11-cryptographic-token-interface-standard.htm」

（１）　C_GetFunctionList

API 名	C_GetFunctionList		
概要	関数ポインタリストを取得する。		
関数インターフェース	CK_DEFINE_FUNCTION(CK_RV,　C_GetFunctionList)(CK_FUNCTION_LIST_PTR_PTR　ppFunctionList);		
戻り値	CK_RV（CKR_OK: 成功 CKR_FUNCTION_FAILED: 失敗）（※１） ※１エラー時には本書で規定していない戻り値が返ることがある。		
	型	I/O	内容
引数	CK_FUNCTION_LIST_PTR_PTR	OUT	関数アドレスリストポインタ

（２）　C_Initialize

API 名	C_Initialize		
概要	PKCS#11 ライブラリを初期化する。		
関数インターフェース	CK_DEFINE_FUNCTION(CK_RV,　C_Initialize)(CK_VOID_PTR　pReserved);		
戻り値	CK_RV（CKR_OK: 成功 CKR_FUNCTION_FAILED: 失敗）（※１） ※１エラー時には本書で規定していない戻り値が返ることがある。		
	型	I/O	内容
引数	CK_VOID_PTR	IN	NULL ポインタを指定

（3） C_Finalize

API 名	C_Finalize		
概要	PKCS#11 ライブラリを終了する。		
関数インターフェース	CK_DEFINE_FUNCTION(CK_RV, C_Finalize)(CK_VOID_PTR pReserved);		
戻り値	CK_RV（CKR_OK: 成功 CKR_FUNCTION_FAILED: 失敗）（※１） ※１エラー時には本書で規定していない戻り値が返ることがある。		
	型	I/O	内容
引数	CK_VOID_PTR	IN	NULL ポインタを指定

（4） C_GetInfo

API 名	C_GetInfo		
概要	ライブラリ情報を取得する。		
関数インターフェース	CK_DEFINE_FUNCTION(CK_RV, C_GetInfo)(CK_INFO_PTR pInfo);		
戻り値	CK_RV（CKR_OK: 成功 CKR_FUNCTION_FAILED: 失敗）（※１） ※１エラー時には本書で規定していない戻り値が返ることがある。		
	型	I/O	内容
引数	CK_INFO_PTR	IN/OUT	ライブラリ情報ポインタ
備考	取得可能なライブラリ情報は以下の通り。 CK_INFO::cryptokiVersion: PKCS11 規格バージョン：2.01 または 2.20 CK_INFO::manufacturerID: ライブラリ製造者名：任意 CK_INFO::description: ライブラリ記述文：任意 CK_INFO::libraryVersion: 任意		

（5） C_GetSlotList

API 名	C_GetSlotList
概要	スロットリストを取得する。
関数インターフェース	CK_DEFINE_FUNCTION(CK_RV, C_GetSlotList)(CK_BBOOL tokenPresent, CK_SLOT_ID_PTR pSlotList, CK_ULONG_PTR pulCount);

戻り値	CK_RV (CKR_OK: 成功 CKR_FUNCTION_FAILED: 失敗) （※１） ※１エラー時には本書で規定していない戻り値が返ることがある。		
	型	I/O	内容
引数	CK_BBOOL	IN	TRUE: カード有りのスロットリストを返す FALSE: 接続されているすべてのスロットリストを返す
	CK_SLOT_ID_PTR	IN/OUT	スロット ID リストポインタ
	CK_ULONG_PTR	IN/OUT	スロット ID リスト件数
備考	複数台の IC カード R/W(物理)が接続されている場合、そのうちの 1 台のみをスロットとして見せる実装を許容する。		

（６） C_GetSlotInfo

API 名	C_GetSlotInfo		
概要	スロット情報を取得する。		
関数インターフェース	CK_DEFINE_FUNCTION(CK_RV, C_GetSlotInfo)(CK_SLOT_ID slotID, CK_SLOT_INFO_PTR pInfo);		
戻り値	CK_RV（CKR_OK: 成功 CKR_FUNCTION_FAILED: 失敗）（※１） ※１エラー時には本書で規定していない戻り値が返ることがある。		
	型	I/O	内容
引数	CK_SLOT_ID	IN	スロット ID
	CK_SLOT_INFO_PTR	IN/OUT	スロット情報ポインタ
備考	取得可能なスロット情報は以下の通り。 CK_SLOT_INFO::hardwareVersion: スロットハードウェアバージョン: 任意 CK_SLOT_INFO::firmwareVersion: スロットファームウェアバージョン: 任意 CK_SLOT_INFO::slotDescription: スロット記述文: 任意（※２） CK_SLOT_INFO::manufacturerID: スロット製造者名: 任意 CK_SLOT_INFO::flags: カード有無 ※２ IC カード R/W 名称が返ることがある。		

（7） C_GetTokenInfo

API 名	C_GetTokenInfo		
概要	トークン情報を取得する。		
関数インターフェース	CK_DEFINE_FUNCTION(CK_RV, C_GetTokenInfo)(CK_SLOT_ID slotID, CK_TOKEN_INFO_PTR pInfo);		
戻り値	CK_RV (CKR_OK: 成功 CKR_TOKEN_NOT_PRESENT: カードが挿入されていないまたはカードが抜かれた CKR_TOKEN_NOT_RECOGNIZED 不正な IC カードを検出した（※１） CKR_FUNCTION_FAILED: 失敗) ※１エラー時には本書で規定していない戻り値が返ることがある。 例：CKR_DEVICE_ERROR 不正な IC カードを検出したまたは IC カードアクセスでのエラー		
	型	I/O	内容
引数	CK_SLOT_ID	IN	スロット ID
	CK_TOKEN_INFO_PTR	OUT	トークン情報ポインタ

（8） C_GetMechanismList

API 名	C_GetMechanismList		
概要	サポートメカニズム（アルゴリズム）を取得する。		
関数インターフェース	CK_DEFINE_FUNCTION(CK_RV, C_GetMechanismList)(CK_SLOT_ID slotID, CK_MECHANISM_TYPE_PTR pMechanismList, CK_ULONG_PTR pulCount);		
戻り値	CK_RV（CKR_OK: 成功 CKR_FUNCTION_FAILED: 失敗）（※１） ※１エラー時には本書で規定していない戻り値が返ることがある。		
	型	I/O	内容
引数	CK_SLOT_ID	IN	スロット ID
	CK_MECHANISM_TYPE_PTR	OUT	メカニズムタイプポインタ

	CK_ULONG_PTR	IN/OUT	メカニズムタイプ件数
備考	取得可能なサポートメカニズムには以下を含むこと。 Sign、Verify 用: CKM_RSA_PKCS Digest 用: CKM_SHA_1、CKM_SHA_256 （※２） ※２ : Digest 用はその機能を実装している場合に取得可能である。		

（９） C_GetMechanismInfo

API 名	C_GetMechanismInfo		
概要	メカニズム（アルゴリズム）情報を返す。		
関数インターフェース	CK_DEFINE_FUNCTION(CK_RV, C_GetMechanismInfo)(CK_SLOT_ID slotID, CK_MECHANISM_TYPE type, CK_MECHANISM_INFO_PTR pInfo);		
戻り値	CK_RV（CKR_OK: 成功 CKR_FUNCTION_FAILED: 失敗）（※１） ※１エラー時には本書で規定していない戻り値が返ることがある。		
	型	I/O	内容
引数	CK_SLOT_ID	IN	スロット ID
	CK_MECHANISM_TYPE	IN	メカニズムタイプ
	CK_MECHANISM_INFO_PTR	IN/OUT	メカニズム情報ポインタ
備考	設定する情報は以下が可能であること。 type = CKM_RSA_PKCS の場合 CK_MECHANISM_INFO::ulMinKeySize: 1024 以下 CK_MECHANISM_INFO::ulMaxKeySize: 1024 | 2048 以上 CK_MECHANISM_INFO::flags: CKF_VERIFY（※２） | CKF_SIGN | CKF_HW ※２ : VERIFY 機能を実装している場合に設定可能である。 type = CKM_SHA_1 | CKM_SHA_256 の場合（※３） CK_MECHANISM_INFO::ulMinKeySize: 任意 CK_MECHANISM_INFO::ulMaxKeySize: 任意 CK_MECHANISM_INFO::flags: CKF_DIGEST ※３ : Digest 機能を実装している場合に設定可能である。		

（１０） C_OpenSession

API 名	C_OpenSession
概要	セッションを確立する。

関数インターフェース	CK_DEFINE_FUNCTION(CK_RV, C_OpenSession)(CK_SLOT_ID slotID, CK_FLAGS flags, CK_VOID_PTR pApplication, CK_NOTIFY Notify, CK_SESSION_HANDLE_PTR phSession);
戻り値	CK_RV（ CKR_OK: 成功 CKR_TOKEN_NOT_PRESENT: カードが挿入されていないまたはカードが抜かれた CKR_TOKEN_NOT_RECOGNIZED: 不正な IC カードを検出した（※１） CKR_FUNCTION_FAILED: 失敗 ） ※１エラー時には本書で規定していない戻り値が返ることがある。 例：CKR_DEVICE_ERROR 不正な IC カードを検出したまたは IC カードアクセスでのエラー

	型	I/O	内容
引数	CK_SLOT_ID	IN	スロット ID
	CK_FLAGS	IN	CKF_SERIAL_SESSION を指定
	CK_VOID_PTR	IN	NULL ポインタを指定
	CK_NOTIFY	IN	NULL ポインタを指定
	CK_SESSION_HANDLE_PTR	IN/OUT	セッションハンドルポインタ

（１１） C_CloseSession

API 名	C_CloseSession
概要	セッションを切断する。
関数インターフェース	CK_DEFINE_FUNCTION(CK_RV, C_CloseSession)(CK_SESSION_HANDLE hSession);
戻り値	CK_RV（CKR_OK: 成功 CKR_FUNCTION_FAILED: 失敗）（※１） ※１エラー時には本書で規定していない戻り値が返ることがある。

	型	I/O	内容
引数	CK_SESSION_HANDLE	IN	セッションハンドル

（１２） C_CloseAllSessions

API 名	C_CloseAllSessions		
概要	すべてのセッションを切断する。		
関数インターフェース	CK_DEFINE_FUNCTION(CK_RV, C_CloseAllSessions)(CK_SLOT_ID slotID);		
戻り値	CK_RV（CKR_OK: 成功 CKR_FUNCTION_FAILED: 失敗）（※１） ※１エラー時には本書で規定していない戻り値が返ることがある。		
	型	I/O	内容
引数	CK_SLOT_ID	IN	スロット ID

（１３） C_GetSessionInfo

API 名	C_GetSessionInfo		
概要	セッション状態を取得する。		
関数インターフェース	CK_DEFINE_FUNCTION(CK_RV, C_GetSessionInfo)(CK_SESSION_HANDLE hSession, CK_SESSION_INFO_PTR pInfo);		
戻り値	CK_RV（CKR_OK: 成功 CKR_FUNCTION_FAILED: 失敗）（※１） ※１エラー時には本書で規定していない戻り値が返ることがある。		
	型	I/O	内容
引数	CK_SESSION_HANDLE	IN	セッションハンドル
	CK_SESSION_INFO_PTR	OUT	セッション状態ポインタ
備考	以下の状態を返す。 ログインしていないとき: CKS_RO_PUBLIC_SESSION ログインしているとき: CKS_RO_USER_FUNCTIONS		

（１４） C_Login

API 名	C_Login
概要	トークンをログイン状態にする（証明書 DF を活性化する）。
関数インターフェース	CK_DEFINE_FUNCTION(CK_RV, C_Login)(CK_SESSION_HANDLE hSession, CK_USER_TYPE userType, CK_CHAR_PTR pPin, CK_ULONG ulPinLen

	);
戻り値	CK_RV（ CKR_OK: 成功 CKR_DEVICE_REMOVED カードが挿入されていないまたはカードが抜かれた（※１） CKR_PIN_INCORRECT: パスワード指定誤り CKR_PIN_LOCKED: パスワードがロックされている CKR_FUNCTION_FAILED: 失敗 ） ※１エラー時には本書で規定していない戻り値が返ることがある。 例：CKR_SESSION_HANDLE_INVALID API 呼び出し時に既にカードが抜けている。 CKR_DEVICE_REMOVED API 実行中にカードが抜けた。 CKR_DEVICE_ERROR IC カードアクセスでのエラー

	型	I/O	内容
引数	CK_SESSION_HANDLE	IN	セッションハンドル
	CK_USER_TYPE	IN	ユーザタイプ
	CK_CHAR_PTR	IN	パスワード文字列ポインタ
	CK_ULONG	IN	パスワード文字列長

（１５） C_Logout

API 名	C_Logout
概要	トークンをログアウト状態にする。
関数インターフェース	CK_DEFINE_FUNCTION(CK_RV, C_Logout)(CK_SESSION_HANDLE hSession);
戻り値	CK_RV（CKR_OK: 成功 CKR_FUNCTION_FAILED: 失敗）（※１） ※１エラー時には本書で規定していない戻り値が返ることがある。

	型	I/O	内容
引数	CK_SESSION_HANDLE	IN	セッションハンドル

（１６） C_FindObjectsInit

API 名	C_FindObjectsInit
概要	オブジェクトの検索を開始する。

関数インターフェース	CK_DEFINE_FUNCTION(CK_RV, C_FindObjectsInit)(CK_SESSION_HANDLE hSession, CK_ATTRIBUTE_PTR pTemplate, CK_ULONG ulCount);
戻り値	CK_RV（ CKR_OK: 成功 CKR_DEVICE_REMOVED: カードが挿入されていないまたはカードが抜かれた（※１） CKR_FUNCTION_FAILED: 失敗 ） ※１エラー時には本書で規定していない戻り値が返ることがある。 例：CKR_SESSION_HANDLE_INVALID API 呼び出し時に既にカードが抜けている。 CKR_DEVICE_REMOVED API 実行中にカードが抜けた。 CKR_DEVICE_ERROR IC カードアクセスでのエラー

	型	I/O	内容
引数	CK_SESSION_HANDLE	IN	セッションハンドル
	CK_ATTRIBUTE_PTR	IN	属性テーブルポインタ
	CK_ULONG	IN	属性テーブル数
備考	以下の属性による検索を行う。 CKA_CLASS CKO_CERTIFICATE または CKO_PRIVATE_KEY CKA_ID オブジェクトを識別するための番号またはラベル名 CKA_TOKEN 真 CKA_LABEL 証明書の名前 CKA_VALUE 証明書の値 CKA_MODULUS 利用者公開鍵の N CKA_PUBLIC_EXPONENT 利用者公開鍵の E		

（１７） C_FindObjects

API 名	C_FindObjects
概要	オブジェクトの検索を行う。
関数インターフェース	CK_DEFINE_FUNCTION(CK_RV, C_FindObjects)(CK_SESSION_HANDLE hSession,

	CK_OBJECT_HANDLE_PTR phObject, CK_ULONG ulMaxObjectCount, CK_ULONG_PTR pulObjectCount);
戻り値	CK_RV（ CKR_OK: 成功 CKR_DEVICE_REMOVED: カードが挿入されていないまたはカードが抜かれた（※１） CKR_FUNCTION_FAILED: 失敗 ） ※１エラー時には本書で規定していない戻り値が返ることがある。 例：CKR_SESSION_HANDLE_INVALID API 呼び出し時に既にカードが抜けている。 CKR_DEVICE_REMOVED API 実行中にカードが抜けた。 CKR_DEVICE_ERROR IC カードアクセスでのエラー

	型	I/O	内容
引数	CK_SESSION_HANDLE	IN	セッションハンドル
	CK_OBJECT_HANDLE_PTR	IN/OUT	オブジェクトハンドルポインタ
	CK_ULONG	IN	最大オブジェクト数
	CK_ULONG_PTR	IN/OUT	オブジェクト数ポインタ

（１８） C_FindObjectsFinal

API 名	C_FindObjectsFinal
概要	オブジェクトの検索を終了する。
関数インターフェース	CK_DEFINE_FUNCTION(CK_RV, C_FindObjectsFinal)(CK_SESSION_HANDLE hSession);
戻り値	CK_RV（ CKR_OK: 成功 CKR_DEVICE_REMOVED: カードが挿入されていないまたはカードが抜かれた（※１） CKR_FUNCTION_FAILED: 失敗 ）

	※１エラー時には本書で規定していない戻り値が返ることがある。 例：CKR_SESSION_HANDLE_INVALID API 呼び出し時に既にカードが抜けている。 CKR_DEVICE_REMOVED API 実行中にカードが抜けた。 CKR_DEVICE_ERROR IC カードアクセスでのエラー

	型	I/O	内容
引数	CK_SESSION_HANDLE	IN	セッションハンドル

（１９） C_GetAttributeValue

API 名	C_GetAttributeValue
概要	オブジェクトの属性値を取得する。
関数インターフェース	CK_DEFINE_FUNCTION(CK_RV, C_GetAttributeValue)(CK_SESSION_HANDLE hSession, CK_OBJECT_HANDLE hObject, CK_ATTRIBUTE_PTR pTemplate, CK_ULONG ulCount);
戻り値	CK_RV（ CKR_OK: 成功 CKR_DEVICE_REMOVED: カードが挿入されていないまたはカードが抜かれた（※１） CKR_FUNCTION_FAILED: 失敗 ） ※１エラー時には本書で規定していない戻り値が返ることがある。 例：CKR_SESSION_HANDLE_INVALID API 呼び出し時に既にカードが抜けている。 CKR_DEVICE_REMOVED API 実行中にカードが抜けた。 CKR_DEVICE_ERROR IC カードアクセスでのエラー

	型	I/O	内容
引数	CK_SESSION_HANDLE	IN	セッションハンドル
	CK_OBJECT_HANDLE	IN	オブジェクトハンドル
	CK_ATTRIBUTE_PTR	OUT	属性テーブルポインタ

	CK_ULONG	OUT	属性テーブル数
備考	以下の属性に対する値が取得可能である。 CKA_CLASS　CKO_CERTIFICATE または CKO_PUBLIC_KEY または CKO_PRIVATE_KEY CKA_LABEL　表 6 オブジェクト情報一覧参照。 CKA_VALUE　表 6 オブジェクト情報一覧参照。 CKA_ISSUER　表 6 オブジェクト情報一覧参照。 CKA_SERIAL_NUMBER　表 6 オブジェクト情報一覧参照。 CKA_SUBJECT　表 6 オブジェクト情報一覧参照。 CKA_ID　表 6 オブジェクト情報一覧参照。 CKA_TOKEN　True CKA_PRIVATE　True CKA_CERTFICATE_TYPE　CKC_X_509 CKA_MODULUS　表 6 オブジェクト情報一覧参照。 CKA_MODULUS_BITS　表 6 オブジェクト情報一覧参照。 CKA_PUBLIC_EXPONENT　表 6 オブジェクト情報一覧参照。 CKA_KEY_TYPE　CKK_RSA CKA_SENSITIVE　True CKA_SIGN　True		

表 6　オブジェクト属性一覧

NO	オブジェクト	属性名	属性値
1	利用者鍵 （公開鍵および秘密鍵）	CKA_LABEL	USERKEY またはオブジェクト生成時に指定された値
		CKA_ID	N の SHA1 ハッシュ（※１）またはオブジェクト生成時に指定された値もしくは内部生成した値
		CKA_PUBLIC_EXPONENT	利用者証明書から取得した値
		CKA_MODULUS	利用者証明書から取得した値
		CKA_MODULUS_BITS	利用者証明書から取得した値
2	利用者証明書	CKA_LABEL	USERCERT またはオブジェクト生成時に指定された値
		CKA_ID	N の SHA1 ハッシュ（※１）またはオブジェクト生成時に指定

			された値もしくは内部生成した値
		CKA_VALUE	証明書自体
		CKA_SUBJECT	利用者証明書から取得した値
		CKA_ISSUER	利用者証明書から取得した値
		CKA_SERIAL_NUMBER	利用者証明書から取得した値
3	都道府県知事の自己署名証明書（※１）	CKA_LABEL	CACERT
		CKA_ID	N の SHA1 ハッシュ
		CKA_VALUE	証明書自体
		CKA_SUBJECT	都道府県知事の自己署名証明書から取得した値
		CKA_ISSUER	都道府県知事の自己署名証明書から取得した値
		CKA_SERIAL_NUMBER	都道府県知事の自己署名証明書から取得した値

※１：公的個人認証サービスのみ

（２０） C_SignInit

API 名	C_SignInit
概要	署名処理を初期化する。
関数インターフェース	CK_DEFINE_FUNCTION(CK_RV, C_SignInit)(CK_SESSION_HANDLE hSession, CK_MECHANISM_PTR pMechanism, CK_OBJECT_HANDLE hKey);
戻り値	CK_RV（ CKR_OK: 成功 CKR_DEVICE_REMOVED: カードが挿入されていないまたはカードが抜かれた（※１） CKR_FUNCTION_FAILED: 失敗 ） ※１エラー時には本書で規定していない戻り値が返ることがある。 例：CKR_SESSION_HANDLE_INVALID API 呼び出し時に既にカードが抜けている。 CKR_DEVICE_REMOVED API 実行中にカードが抜けた。

	CKR_DEVICE_ERROR IC カードアクセスでのエラー

	型	I/O	内容
引数	CK_SESSION_HANDLE	IN	セッションハンドル
	CK_MECHANISM_PTR	IN	メカニズム情報ポインタ mechanism: CKM_RSA_PKCS が指定可能
	CK_OBJECT_HANDLE	IN	オブジェクトハンドル

（２１） C_Sign

API 名	C_Sign
概要	データに署名を行う。
関数インターフェース	CK_DEFINE_FUNCTION(CK_RV, C_Sign)(CK_SESSION_HANDLE hSession, CK_BYTE_PTR pData, CK_ULONG ulDataLen, CK_BYTE_PTR pSignature, CK_ULONG_PTR pulSignatureLen);
戻り値	CK_RV（ CKR_OK: 成功 CKR_DEVICE_REMOVED: カードが挿入されていないまたはカードが抜かれた（※１） CKR_FUNCTION_FAILED: 失敗 ） ※１エラー時には本書で規定していない戻り値が返ることがある。 例：CKR_SESSION_HANDLE_INVALID API 呼び出し時に既にカードが抜けている。 CKR_DEVICE_REMOVED API 実行中にカードが抜けた。 CKR_DEVICE_ERROR IC カードアクセスでのエラー

	型	I/O	内容
引数	CK_SESSION_HANDLE	IN	セッションハンドル
	CK_BYTE_PTR	IN	データポインタ

	CK_ULONG	IN	データ長
	CK_BYTE_PTR	IN/OUT	署名データポインタ
	CK_ULONG_PTR	IN/OUT	署名データ長ポインタ

（２２） C_DigestInit

API 名	C_DigestInit		
概要	ダイジェスト作成を開始する。		
関数インターフェース	CK_DEFINE_FUNCTION(CK_RV, C_DigestInit)(CK_SESSION_HANDLE hSession, CK_MECHANISM_PTR pMechanism);		
戻り値	CK_RV（CKR_OK: 成功 CKR_FUNCTION_FAILED: 失敗）（※１） ※１エラー時には本書で規定していない戻り値が返ることがある。		
	型	I/O	内容
引数	CK_SESSION_HANDLE	IN	セッションハンドル
	CK_MECHANISM_PTR	IN	メカニズム情報ポインタ mechanism: CKM_SHA_1 |CKM_SHA_256 が指定可能

（２３） C_DigestUpdate

API 名	C_DigestUpdate		
概要	ダイジェストを作成する。		
関数インターフェース	CK_DEFINE_FUNCTION(CK_RV, C_DigestUpdate)(CK_SESSION_HANDLE hSession, CK_BYTE_PTR pPart, CK_ULONG ulPartLen);		
戻り値	CK_RV（CKR_OK: 成功 CKR_FUNCTION_FAILED: 失敗）（※１） ※１エラー時には本書で規定していない戻り値が返ることがある。		
	型	I/O	内容
引数	CK_SESSION_HANDLE	IN	セッションハンドル
	CK_BYTE_PTR	IN	ハッシュするデータポインタ
	CK_ULONG	IN	ハッシュするデータ長

（２４） C_DigestFinal

API 名	C_DigestFinal		
概要	ダイジェスト作成を終了する。		
関数インターフェース	CK_DEFINE_FUNCTION(CK_RV, C_DigestFinal)(CK_SESSION_HANDLE hSession, CK_BYTE_PTR pDigest, CK_ULONG_PTR pulDigestLen);		
戻り値	CK_RV（CKR_OK: 成功 CKR_FUNCTION_FAILED: 失敗）（※１） ※１エラー時には本書で規定していない戻り値が返ることがある。		
	型	I/O	内容
引数	CK_SESSION_HANDLE	IN	セッションハンドル
	CK_BYTE_PTR	IN/OUT	ダイジェストデータポインタ
	CK_ULONG_PTR	IN/OUT	ダイジェストデータ長ポインタ

（２５） C_VerifyInit

API 名	C_VerifyInit		
概要	署名検証を開始する。		
関数インターフェース	CK_DEFINE_FUNCTION(CK_RV, C_VerifyInit)(CK_SESSION_HANDLE hSession, CK_MECHANISM_PTR pMechanism CK_OBJECT_HANDLE hKey);		
戻り値	CK_RV（CKR_OK: 成功 CKR_FUNCTION_FAILED: 失敗）（※１） ※１エラー時には本書で規定していない戻り値が返ることがある。		
	型	I/O	内容
引数	CK_SESSION_HANDLE	IN	セッションハンドル
	CK_MECHANISM_PTR	IN	メカニズム情報ポインタ mechanism: CKM_RSA_PKCS が指定可能
	CK_OBJECT_HANDLE	IN	RSA 公開鍵オブジェクト

（２６） C_Verify

API 名	C_Verify
概要	署名値を検証する。
関数インターフェース	CK_DEFINE_FUNCTION(CK_RV, C_Verify)(CK_SESSION_HANDLE hSession, CK_BYTE_PTR pData, CK_ULONG ulDataLen, CK_BYTE_PTR pSignature, CK_ULONG ulSignatureLen);
戻り値	CK_RV（ CKR_OK: 成功 CKR_SIGNATURE_INVALID: 署名データ不正（※１） CKR_FUNCTION_FAILED: 失敗 ） ※１エラー時には本書で規定していない戻り値が返ることがある。 例：CKR_DEVICE_ERROR カードが挿入されていないまたはカードが抜かれたまたは IC カードアクセスでのエラー

	型		内容
引数	CK_SESSION_HANDLE	IN	セッションハンドル
	CK_BYTE_PTR	IN	検証するデータポインタ
	CK_ULONG	IN	検証するデータ長
	CK_BYTE_PTR	IN	署名データポインタ
	CK_ULONG	IN	署名データ長

備考

pData と pSigneture の OID の関係は以下の通り。
署名データに OID をつける場合は、上位 UP にて検証するデータに OID を付加したデータを入力しなければならない。

		検証するデータ(pData)	
		OID あり	OID なし
署名データ (pSigneture)	OID あり	OK	NG
	OID なし	NG	OK

（２７） C_CreateObject

API 名	C_CreateObject		
概要	公開鍵オブジェクトを作成する。		
関数インターフェース	CK_DEFINE_FUNCTION(CK_RV, C_CreateObject)(CK_SESSION_HANDLE hSession, CK_ATTRIBUTE_PTR pTemplate, CK_ULONG ulCount, CK_OBJECT_HANDLE_PTR phObject);		
戻り値	CK_RV（CKR_OK: 成功 CKR_FUNCTION_FAILED: 失敗）（※１） ※１エラー時には本書で規定していない戻り値が返ることがある。		
	型	I/O	内容
引数	CK_SESSION_HANDLE	IN	セッションハンドル
	CK_ATTRIBUTE_PTR	IN	属性テーブルポインタ
	CK_ULONG	IN	属性テーブル数
	CK_OBJECT_HANDLE_PTR	IN/OUT	オブジェクトハンドルポインタ
備考	本 API では RSA 公開鍵セッションオブジェクトのみ作成できる。 以下の属性を pTemplate で指定する。 CKA_CLASS: CKO_PUBLIC_KEY（※２） CKA_KEY_TYPE: CKK_RSA CKA_PUBLIC_EXPONENT CKA_MODULUS ※２ :CKA_CLASS の値は CKO_PUBLIC_KEY 固定とする。その他の属性は使用不可とする。		

（２８） C_DestroyObject

API 名	C_DestroyObject		
概要	公開鍵オブジェクトを破棄する。		
関数インターフェース	CK_DEFINE_FUNCTION(CK_RV, C_DestroyObject)(CK_SESSION_HANDLE hSession, CK_OBJECT_HANDLE hObject);		
戻り値	CK_RV（CKR_OK: 成功 CKR_FUNCTION_FAILED: 失敗）（※１） ※１エラー時には本書で規定していない戻り値が返ることがある。		
	型	I/O	内容

引数	CK_SESSION_HANDLE	IN	**セッションハンドル**
	CK_OBJECT_HANDLE	IN	**オブジェクトハンドル**
備考	**本 API では RSA 公開鍵セッションオブジェクトのみ破棄できる。**		

第３節 構造体仕様

構造体は下記 URL にある PKCS#11 Ver2.01 および Ver2.20 の正式な資料を参照のこと。

RSA Laboratories

「http://www.emc.com/emc-plus/rsa-labs/standards-initiatives/pkcs-11-cryptographic-token-interface-standard.htm」© Copyright 2013 EMC Corporation. all rights reserved.

表 7 に本ライブラリで使用する構造体の一覧を示す。

表 7 PKCS#11 構造体一覧

NO	API 名	概要
1	CK_INFO CK_INFO_PTR	PKCS ライブラリ情報
2	CK_SLOT_ID CK_SLOT_ID_PTR	スロット ID
3	CK_SLOT_INFO CK_SLOT_INFO_PTR	スロット情報
4	CK_TOKEN_INFO CK_TOKEN_INFO_PTR	トークン情報
5	CK_SESSION_HANDLE CK_SESSION_HANDLE_PTR	セッションハンドル
6	CK_USER_TYPE	ユーザタイプ
7	CK_SESSION_INFO CK_SESSION_INFO_PTR	セッション情報
8	CK_OBJECT_HANDLE CK_OBJECT_HANDLE_PTR	オブジェクトハンドル
9	CK_OBJECT_CLASS CK_OBJECT_CLASS_PTR	オブジェクトクラス
10	CK_ATTRIBUTE CK_ATTRIBUTE_PTR	属性タイプ、値、長さを含む構造体
11	CK_MECHANISM_TYPE CK_MECHANISM_TYPE_PTR	メカニズムタイプ
12	CK_MECHANISM CK_MECHANISM_PTR	メカニズムタイプを含む、メカニズムを示す構造体
13	CK_MECHANISM_INFO CK_MECHANISM_INFO_PTR	メカニズム情報
14	CK_RV	ライブラリの戻り値
15	CK_NOTIFY	コールバック情報
16	CK_FUNCTION_LIST	PKCS ライブラリ関数

	CK_FUNCTION_LIST_PTR CK_FUNCTION_LIST_PTR_PTR	

第４節 コーリングシーケンス

各サポートAPIの実装に際しては、公的個人認証サービス利用者クライアントソフトAPI仕様書【カードAP ライブラリPKCS#11 編】に準じたコーリングシーケンスを上位アプリケーションが実現できるよう実装を行うものとする。

また 繰り返し署名生成処理および 繰り返し署名検証処理にも対応することとする。

なおPKCS11規格バージョンやライブラリ製造者名によって固有の処理を必要とする場合には、C_GetInfoによってライブラリ情報を取得すること。

総務省電子調達システムにおいては特に

(1)初期処理

C_OpenSession(アプリケーションとトークン間のセッションの確立)と

C_Login(トークンへのログイン)の間で、C_GetSessionInfo(セッション情報の取得)をコールする場合がある。

(4)署名生成処理

C_FindObjectsInit(証明書検索操作の初期設定)からC_FindObjectsFinal(証明書検索操作の終了処理)は行わない場合がある。

詳細は電子調達システム【別紙1 補足1】を参考のこと。

第６章 その他

第１節 ライブラリのロード方法

汎用受付システム等の上位アプリケーションでは、ロードすべき PKCS#11 ライブラリのパス名を固定のファイル（ロード情報定義ファイル）から取得し、ロードする方式を採用している。

このロード情報定義ファイルのパス名およびファイル名およびロード情報定義ファイル内のフォーマットには公的個人認証のカード AP ライブラリ PKCS#11 編で先約されているものがあるため、干渉しないように設定を行うものとする。

また総務省電子調達システムのロード情報定義ファイルについては、以下のいずれかのパスに配置するものとする。

「C:¥Program Files¥Common Files¥e-gov_app¥load_path¥default.dat」

「C:¥Program Files(x86)¥Common Files¥e-gov_app¥load_path¥default.dat」

「ロードすべき PKCS#11 ライブラリのパス名」のキーは、以下のいずれかとする。

「path64」

「path」

詳細は電子調達システム【別紙 1 補足 2】を参照すること。

（注意事項）

※本仕様は公的個人認証サービス利用者クライアントソフトの改変・改造を目的としたものではありません。

※本仕様に示される製品の著作権はそれぞれの製造者に属します。

※参考資料

・公的個人認証サービス利用者クライアントソフト API 仕様書【カード AP ライブラリ PKCS#11 編】第 2.5 版

・汎用受付等システムの構築・運用に関する共通事項
（http://www.soumu.go.jp/main_sosiki/gyoukan/kanri/020329_1.htm）

・電子調達システム（閲覧要望は関係機関の当該部門に問い合わせのこと）
【別紙 1 補足 1】総務省向け署名値取得コマンドの操作フロー（非公開）
【別紙 1 補足 2】64bitOS の場合の PKCS#11 ライブラリのロードについて（非公開）

・RSA　Laboratories
（http://www.emc.com/emc-plus/rsa-labs/standards-initiatives/pkcs-11-cryptographic-token-interface-standard.htm）PKCS#11 Ver2.01、Ver2.20