طرح:

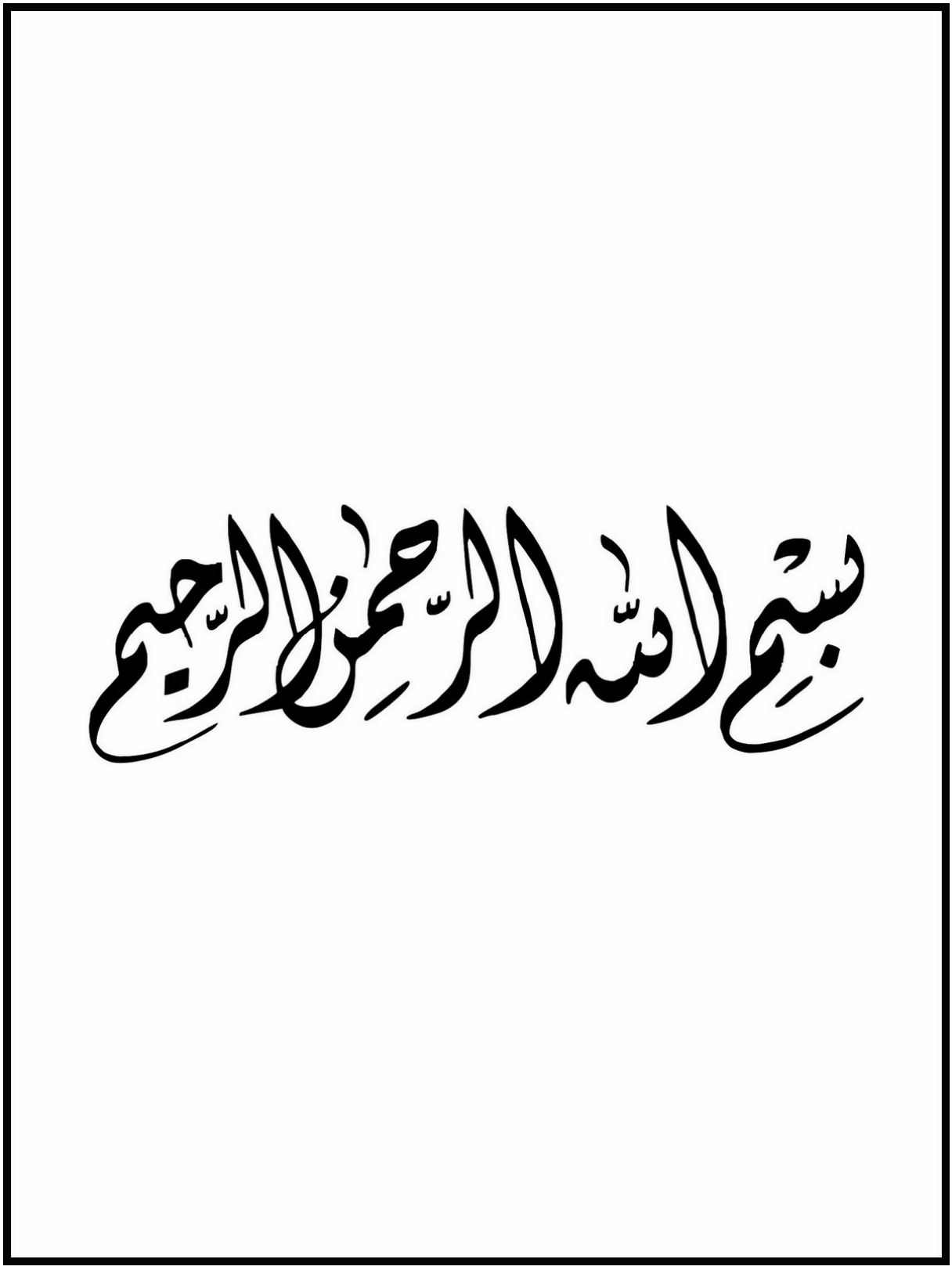
تدوین شاخص های ارزیابی عملیاتی ماژول‌های رمزنگاری بر مبنای استانداردهای روز رمزنگاری

مجری:

دکتر سید امیر اصغری

دانشگاه خوارزمی تهران

۱۳۹۹


بسم الله الرحمن الرحیم

فهرست مطالب

۱ روش تحقیق ۳
۱.۱ سوالات پژوهش ۳
۱.۲ اهداف و خروجی های پژوهش ۳
۱.۳ مراحل اجرای پژوهش ۴
۱.۴ روش های مورد استفاده در اجرای پژوهش ۴
۲ مقدمه و تعریف موضوع ۴
۲.۱ لغت نامه ۵
۲.۲ تعاریف ۵
۲.۳ مراجع اصلی ۶
۲.۴ مراجع فرعی ۶
۲.۵ توابع کتابخانه Cryptoki ۶
۲.۶ مقادیر بازگشتی توابع کتابخانه Cryptoki ۱۱
۲.۶.۱ مقادیر بازگشتی عمومی برای توابع Cryptoki ۱۱
۲.۶.۲ مقادیر بازگشتی توابعی که از شناسه نشست استفاده می کنند ۱۱
۲.۶.۳ مقادیر بازگشتی برای توابعی که از توکن استفاده می کنند ۱۲
۲.۶.۴ مقدار برگشتی ویژه ۱۲
۲.۶.۵ مقادیر برگشتی ویژه برای توابع کنترل Mutex ۱۲
۲.۶.۶ سایر مقادیر بازگشتی توابع Cryptoki ۱۳
۲.۶.۷ توابعی که یک خروجی تولید و در یک بافر با طول متغیر باز می گردانند ۱۹
۳ توابع کتابخانه Cryptoki ۲۰
توابع عام ۲۰
توابع مدیریت توکن و اسلات ۲۳
توابع مدیریت نشست ۲۹
توابع مدیریت شیء ۳۴
توابع رمزگذاری ۳۹
توابع رمزگذاری مبتنی بر پیام ۴۱
توابع رمزگشایی ۴۴
توابع رمزگشایی مبتنی بر پیام ۴۷
توابع رمزگشایی مبتنی بر پیام ۴۸
توابع چکیده سازی پیام ۵۰
توابع امضا و MAC ۵۲
توابع مبتنی بر پیام برای امضا و MAC ۵۷
توابع راستی آزمایی امضا و MAC ۶۰
توابع مبتنی بر پیام برای راستی آزمایی امضا و MAC ۶۴
توابع رمزنگاری دو عملیاتی ۶۷
توابع مدیریت کلید ۶۹
توابع مولد اعداد تصادفی ۷۳
توابع مدیریت عملکرد موازی ۷۳

۱ روش تحقیق

با توسعه روز افزون فناوری های الکترونیکی در کسب و کارها و فعالیت های اقتصادی، موضوع امنیت در این بستر از اهمیت زیادی برخوردار است. یکی از مهمترین ساز و کارهای تامین امنیت در این حوزه، زیرساخت کلید عمومی (PKI) و گواهی الکترونیکی می باشد.

ماژول های رمزنگاری نقش محوری در زیرساخت کلید عمومی ایفا می کنند. این ماژول ها دارای جنبه های سخت افزاری و نرم افزاری هستند. بنابراین اطمینان از عملکرد صحیح ماژول های رمزنگاری از جنبه سخت افزاری و نرم افزاری بسیار اهمیت دارد. بررسی و ارزیابی کارکردی و امنیتی ماژول های رمزنگاری باید مبتنی بر یک سری شاخص شفاف و قابل آزمون صورت پذیرد.

شاخص های ارزیابی ماژول های رمزنگاری بر اساس نوع داده ها، توابع عملیاتی، ساز و کارهای رمزنگاری و امنیتی فیزیکی قابل تقسیم هستند. منابع فنی و استانداردهای متنوعی به این شاخص ها پرداخته است. در هر یک از این منابع با توجه به زمینه عملیاتی ماژولها به یک جنبه از شاخص ها توجه بیشتری شده است. بنابراین تدوین مجموعه ای یکپارچه و مبتنی بر نیازمندیهای بومی برای شاخص های ارزیابی ماژول های رمزنگاری قابل استفاده در زیرساخت کلید عمومی کشور امری ضروری می نماید.

۱.۱ سوالات پژوهش

- مهمترین موارد استفاده از ماژول های امنیتی در زیرساخت کلید عمومی کدام است؟
- مهمترین چالش های عملیاتی و امنیتی ماژول های رمزنگاری کدام است؟
- شاخص های اصلی ماژول های رمزنگاری در منابع و استانداردهای علمی و مرجع چگونه معرفی شده است؟
- چگونه می توان مبتنی بر نیازمندی های بومی و ملاحظات فنی زیرساخت کلید عمومی مجموعه ای از شاخص های قابل آزمون برای ماژول های رمزنگاری تدوین کرد؟
- این شاخص های ارزیابی با چه روشی باید برای ماژول های رمزنگاری مورد آزمون قرار گیرند؟

۱.۲ اهداف و خروجی های پژوهش

- تدوین سند شاخص های ارزیابی نرم افزاری ماژول های رمزنگاری

- تدوین سند ساز و کارهای رمزنگاری مورد تایید برای اهداف امنیتی در ماژول های رمزنگاری

۱.۳ مراحل اجرای پژوهش

- بررسی شرایط و نیازمندی های ماژول های رمزنگاری برای استفاده در زیرساخت کلید عمومی کشور
- مطالعه و بررسی اسناد و استانداردهای ملی و بین المللی مرتبط با ماژول های رمزنگاری
- استخراج و دسته بندی شاخصه های عملیاتی ماژول های رمزنگاری
- استخراج و دسته بندی ساز و کارها و الگوریتم های رمزنگاری ماژول های رمزنگاری
- تدوین سند شاخصهای ارزیابی عملیاتی ماژول های رمزنگاری
- تدوین سند ساز و کارها و الگوریتم های رمزنگاری مورد تایید برای ماژول های رمزنگاری
- تدوین سند روش آزمون شاخص های ماژول های رمزنگاری

۱.۴ روش های مورد استفاده در اجرای پژوهش

- مطالعه تطبیقی اسناد و منابع کتابخانه ای و الکترونیکی
- استفاده از ابزارهای مدلسازی نرم افزاری

۲ مقدمه و تعریف موضوع

استاندارد PKCS # 11 یک رابط برنامه نویسی کاربردی به نام "Cryptoki" برای دستگاه های نگهدارنده یا پردازنده اطلاعات رمزنگاری ارائه می دهد. Cryptoki دارای رویکرد برنامه نویسی شیءگرا، مستقل از فناوری (سازگاری با هر نوع دستگاه) و اشتراک گذاری منابع (دسترسی برنامه های مختلف به چندین دستگاه) می باشد. Cryptoki یک نمای مشترک و منطقی از دستگاه را به نام "توکن رمزنگاری" به برنامه ها ارائه می دهد.

Cryptoki انواع داده ها و توابع موجود در برنامه که نیاز به خدمات رمزنگاری دارد را مشخص می کند. زبان برنامه نویسی این کتابخانه زبان ANSI C است. توسعه دهندگان کتابخانه Cryptoki معمولاً انواع داده ها و توابع را از طریق فایل های سرایند ANSI C فراهم می کنند. فایل های سرآیند عمومی برای Cryptoki از صفحه وب PKCS 11 #در دسترس هستند.

۲.۱ لغت نامه

فارسی	انگلیسی
رابط برنامه نویسی	**API**
برنامه کاربردی	**Application**
توکن	**Token**
شیء	**Object**
نشست	**Session**
ویژگی	**Attribute**
اسلات	**Slot**
کاربر	**User**
راهبر امنیتی	**Security Officer**
کلید محرمانه	**Secret key**
کلید خصوصی/عمومی	**Private/Public key**
گواهینامه	**Certificate**
امضا	**Signature**
رمزگذاری	**Encryption**
رمزگشایی	**Decryption**
چکیده سازی	**Digest**
راستی آزمایی	**Verify**
بذر اولیه	**Seed**
مکانیزم	**Mechanism**
رمزگذاری کلید	**Wrap key**
رمزگشایی کلید	**Unwrap key**
عدد تصادفی	**Random number**

۲.۲ تعاریف

عبارت	تعریف
Cryptoki	رابط توکن رمزنگاری که در این استاندارد تعریف شده است.
کتابخانه Cryptoki	کتابخانه ای که توابع مشخص شده در این استاندارد را اجرا می کند.
برنامه کاربردی	هر برنامه رایانه ای که رابط Cryptoki را فراخوانی می کند.
توکن	دستگاه رمزنگاری (سخت افزاری یا نرم افزاری) تعریف شده توسط Cryptoki.

شیء	مجموعه‌ای از اطلاعات (داده، گواهینامه یا کلید) که به عنوان واحدی منفرد با آن‌ها رفتار می‌شود.
ویژگی شیء	هر شیء دارای مجموعه ای از ویژگی ها است که شیء ها را از یکدیگر متمایز می نماید.
نشست	ارتباط متقابل بین برنامه کاربردی و توکن.
مکانیزم	فرایندی برای اجرای یک عملیات رمزنگاری.
اسلات	محلی بر روی سخت افزار رایانه که برای قرارگیری توکن سخت افزاری تعبیه شده است.
کاربر	شخصی که از برنامه کاربردی در ارتباط با Cryptoki استفاده می کند.

۲.۳ مراجع اصلی

[PKCS11-Base] PKCS #11 Cryptographic Token Interface Base Specification Version 2.40. Edited by Robert Griffin and Tim Hudson. Latest stage. http://docs.oasis-open.org/pkcs11/pkcs11-base/v2.40/pkcs11-base-v2.40.html.

۲.۴ مراجع فرعی

[PKCS11-Prof] PKCS#11 Cryptographic Token Interface Profiles Version 3.0. Edited by Tim Hudson. Latest stage. https://docs.oasis-open.org/pkcs11/pkcs11-profiles/v3.0/pkcs11-profiles-v3.0.html.

[PKCS11-Curr] PKCS #11 Cryptographic Token Interface Current Mechanisms Specification Version 3.0. Edited by Chris Zimman and Dieter Bong. Latest stage. https://docs.oasis-open.org/pkcs11/pkcs11-curr/v3.0/pkcs11-curr-v3.0.html.

[PKCS11-Hist] PKCS #11 Cryptographic Token Interface Historical Mechanisms Specification Version 3.0. Edited by Chris Zimman and Dieter Bong. Latest stage. https://docs.oasis-open.org/pkcs11/pkcs11-hist/v3.0/pkcs11-hist-v3.0.html.

۲.۵ توابع کتابخانه Cryptoki

در نسخه فعلی از این کتابخانه، توابع در دسته بندی زیر سازماندهی شده اند:

- توابع عمومی (۶ تابع)
- توابع مدیریت اسلات و توکن (۹ تابع)
- توابع مدیریت نشست (۱۰ تابع)
- توابع مدیریت شیء (۹ تابع)
- توابع رمزگذاری (۴ تابع)
- توابع رمزگذاری مبتنی بر پیام (۵ تابع)

- توابع رمزگشایی (۴ تابع)
- توابع رمزگشایی مبتنی بر پیام (۵ تابع)
- توابع چکیده سازی (۵ تابع)
- توابع امضا و MAC (۶ تابع)
- توابع امضا مبتنی بر پیام (۵ تابع)
- توابع راستی آزمایی امضا و MAC (۶ تابع)
- توابع مبتنی بر پیام برای راستی آزمایی امضا و MAC (۵ تابع)
- توابع رمزنگاری دو منظوره (۴ تابع)
- توابع مدیریت کلید (۵ تابع)
- توابع تولید اعداد تصادفی (۲ تابع)
- توابع مدیریت عملکرد موازی (۲ تابع)

لیست توابع کتابخانه به شرح جدول زیر می باشد:

دسته بندی	تابع	شرح
توابع عمومی	C_Initialize	کتابخانه Cryptoki را راه اندازی و مقداردهی اولیه می کند
	C_Finalize	منابع متفرقه مرتبط با Cryptoki را حذف کنید
	C_GetInfo	اطلاعات کلی در مورد Cryptoki بدست می آورد
	C_GetFunctionList	لیستی از توابع کتابخانه Cryptoki را بدست می آورد
	C_GetInterfaceList	لیستی از رابط های پشتیبانی شده توسط کتابخانه Cryptoki را بدست می آورد
	C_GetInterface	نقاط ورودی خاص رابط کاربری کتابخانه Cryptoki را بدست می آورد
توابع مدیریت اسلات و توکن	C_GetSlotList	لیستی از اسلات های موجود در سیستم را بدست می آورد
	C_GetSlotInfo	اطلاعات مربوط به یک اسلات خاص را بدست می آورد
	C_GetTokenInfo	اطلاعات مربوط به یک توکن خاص را بدست می آورد
	C_WaitForSlotEvent	منتظر یک رویداد اسلات (درج توکن ، حذف و غیره) است
	C_GetMechanismList	لیستی از مکانیزم های پشتیبانی شده توسط یک توکن را بدست می آورد
	C_GetMechanismInfo	اطلاعات مربوط به یک مکانیزم خاص را بدست می آورد
	C_InitToken	یک توکن را مقدار دهی اولیه می کند
	C_InitPIN	پین کاربر عادی را مقداردهی اولیه می کند

	C_SetPIN	پین کاربر معین را اصلاح می کند
توابع مدیریت نشست	C_OpenSession	یک ارتباط بین برنامه کاربردی و توکن معین باز می کند
	C_CloseSession	یک نشیت را می بندد
	C_CloseAllSessions	تمام نشست ها را با یک توکن می بندد
	C_GetSessionInfo	اطلاعات مربوط به نشست را بدست می آورد
	C_SessionCancel	عملیات مبتنی بر نشست فعال را خاتمه می دهد
	C_GetOperationState	حالت عملیات رمزنگاری یک نشست را بدست می آورد
	C_SetOperationState	حالت عملیات رمزنگاری یک نشست را تنظیم می کند
	C_Login	وارد یک توکن می شود
	C_LoginUser	وارد یک توکن با نام کاربری مشخص می شود
	C_Logout	از یک توکن خارج می شود
توابع مدیریت شیء	C_CreateObject	یک شیء ایجاد می کند
	C_CopyObject	یک کپی از یک شیء ایجاد می کند
	C_DestroyObject	شیئی را از بین می برد
	C_GetObjectSize	اندازه یک شی را در قالب بایت بدست می آورد
	C_GetAttributeValue	مقدار ویژگی یک شیء را بدست می آورد
	C_SetAttributeValue	مقدار ویژگی یک شیء را اصلاح می کند
	C_FindObjectsInit	عملیات جستجوی اشیا را مقدار دهی اولیه می کند
	C_FindObjects	عملیات جستجوی شیء را ادامه می دهد
	C_FindObjectsFinal	عملیات جستجوی شیء را به پایان می رساند
توابع رمزگذاری	C_EncryptInit	عملیات رمزگذاری را شروع می کند
	C_Encrypt	داده های تک بخشی را رمزگذاری می کند
	C_EncryptUpdate	عملیات رمزگذاری چند بخشی را ادامه می دهد
	C_EncryptFinal	عملیات رمزگذاری چند بخشی را به پایان می رساند
توابع رمزگذاری مبتنی بر پیام	C_MessageEncryptInit	عملیات رمزگذاری مبتنی بر پیام را مقداردهی اولیه می کند
	C_EncryptMessage	عملیات رمزگذاری مبتنی بر پیام را انجام می دهد
	C_EncryptMessageBegin	عملیات رمزگذاری میتنی بر پیام چند بخشی را آغاز می کند
	C_EncryptMessageNext	عملیات رمزگذاری مبتنی بر پیام چند بخشی را ادامه می دهد یا به پایان می رساند
	C_MessageEncryptFinal	عملیات رمزگذاری مبتنی بر پیام را به پایان می رساند

توابع رمزگشایی	C_DecryptInit	عملیات رمزگشایی را آغاز می کند
	C_Decrypt	عملیات رمزگشایی داده های تک بخشی را انجام می دهد
	C_DecryptUpdate	عملیات رمزگشایی چند بخشی را ادامه می دهد
	C_DecryptFinal	عملیات رمزگشایی چند بخشی را به پایان می رساند
توابع رمزگشایی مبتنی بر پیام	C_MessageDecryptInit	عملیات رمزگشایی مبتنی بر پیام را مقدار دهی اولیه می کند
	C_DecryptMessage	عملیات رمزگشایی مبتنی بر پیام داده های تک بخشی را انجام می دهد
	C_DecryptMessageBegin	عملیات رمزگشایی مبتنی بر پیام داده های چند بخشی را شروع می کند
	C_DecryptMessageNext	عملیات رمزگشایی مبتنی بر پیام داده های چند بخشی را ادامه داده و به پایان می رساند
	C_MessageDecryptFinal	عملیات رمزگشایی مبتنی بر پیام را به پایان می رساند
توابع چکیده سازی	C_DigestInit	عملیات چکیده سازی پیام را مقداردهی اولیه می کند
	C_Digest	عملیات چکیده سازی داده های تک بخشی را انجام می دهد
	C_DigestUpdate	عملیات چکیده سازی داده های چند بخشی را ادامه می دهد
	C_DigestKey	عملیات چکیده سازی کلید را انجام می دهد
	C_DigestFinal	عملیات چکیده سازی داده های چند بخشی را به پایان می رساند
توابع امضا و MAC	C_SignInit	عملیات امضا را مقداردهی اولیه می کند
	C_Sign	عملیات امضای داده های تک بخشی را انجام می دهد
	C_SignUpdate	عملیات امضا داده های چند بخشی را ادامه می دهد
	C_SignFinal	عملیات امضا داده های چند بخشی را به پایان می رساند
	C_SignRecoverInit	یک عملیات امضا را مقداردهی اولیه می کند، جایی که داده ها می توانند از امضا بازیابی شوند
	C_SignRecover	داده های تک بخشی را امضا می کند، جایی که داده ها می توانند از طریق امضا بازیابی شوند
توابع امضا مبتنی بر پیام	C_MessageSignInit	عملیات امضای مبتنی بر پیام را مقدار دهی اولیه می کند
	C_SignMessage	عملیات امضای مبتنی بر پیام را برای داده های تک بخشی انجام می دهد
	C_SignMessageBegin	عملیات امضای مبتنی بر پیام را برای داده های چند بخشی شروع می کند
	C_SignMessageNext	عملیات امضای مبتنی بر پیام را برای داده های چند بخشی ادامه داده و به پایان می رساند
	C_MessageSignFinal	عملیات امضای مبتنی بر پیام را به پایان می رساند
توابع راستی	C_VerifyInit	عملیات راستی آزمایی امضا یا MAC را مقداردهی اولیه می کند
	C_Verify	عملیات راستی آزمایی امضا یا MAC را برای داده های تک بخشی انجام می

آزمایی امضا وMAC		دهد
	C_VerifyUpdate	عملیات راستی آزمایی امضا یا MAC را برای داده های چند بخشی ادامه می دهد
	C_VerifyFinal	عملیات راستی آزمایی امضا یا MAC را برای داده های چند بخشی به پایان می رساند
	C_VerifyRecoverInit	عملیات راستی آزمایی امضا را در جایی که داده ها از امضا بازیابی می شوند، مقداردهی اولیه می کند
	C_VerifyRecover	عملیات راستی آزمایی امضا را در جایی که داده ها از امضا بازیابی می شوند، مقداردهی اولیه می کند
توابع مبتنی بر پیام برای راستی آزمایی امضا وMAC	**C_MessageVerifyInit**	عملیات راستی آزمایی مبتی بر پیام را مقدار دهی اولیه می کند
	C_VerifyMessage	عملیات راستی آزمایی مبتنی بر پیام را برای داده های تک بخشی انجام می دهد
	C_VerifyMessageBegin	عملیات راستی آزمایی مبتنی بر پیام را برای داده های چند بخشی شروع می کند
	C_VerifyMessageNext	عملیات راستی آزمایی مبتنی بر پیام را برای داده های چند بخشی ادامه داده و به پایان می رساند
	C_MessageVerifyFinal	یک عملیات راستی آزمایی مبتنی بر پیام را به پایان می رساند
توابع رمزنگاری دو منظوره	**C_DigestEncryptUpdate**	عملیات چکیده سازی و رمزگذاری همزمان چند بخشی داده را ادامه می دهد
	C_DecryptDigestUpdate	عملیات رمزگشایی و چکیده سازی همزمان چند بخشی داده را ادامه می دهد
	C_SignEncryptUpdate	عملیات امضا و رمزگذاری همزمان چند بخشی داده را ادامه می دهد
	C_DecryptVerifyUpdate	عملیات رمزگشایی و راستی آزمایی همزمان چند بخشی داده را ادامه می دهد
توابع مدیریت کلید	**C_GenerateKey**	یک کلید محرمانه تولید می کند
	C_GenerateKeyPair	یک زوج کلید عمومی/خصوصی ایجاد می کند
	C_WrapKey	یک کلید را رمزگذاری می کند
	C_UnwrapKey	یک کلید را رمزگشایی می کند
	C_DeriveKey	یک کلید را از یک کلید پایه استخراج می کند
توابع تولید اعداد تصادفی	**C_SeedRandom**	یک بذر اولیه به مولد اعداد تصادفی می دهد
	C_GenerateRandom	یک دنباله اعداد تصادفی تولید می کند

توابع مدیریت عملکرد موازی	C_GetFunctionStatus	این توابع مربوط به نسخه های قدیمی Cryptoki می باشد و در نسخه فعلی استفاده نمی شود . فراخوانی این توابع همیشه کد خطای CKR_FUNCTION_NOT_PARALLEL را برمی گرداند.
	C_CancelFunction	

۲.۶ مقادیر بازگشتی توابع کتابخانه Cryptoki

توابع این کتابخانه هم در زمان عملکرد درست و همچنین هنگام بروز خطا، مقداری را به عنوان خروجی برمی گردانند. در ادامه هر یک از این خروجی ها توضیح داده می شود.

۲.۶.۱ مقادیر بازگشتی عمومی برای توابع Cryptoki

هر تابع Cryptoki می تواند هر یک از مقادیر زیر را بازگرداند:

- CKR_GENERAL_ERROR: هنگامی که خطای بحرانی و غیرقابل بازیابی رخ داده است. این امکان وجود دارد که اجرای تابع تا حدودی موفقیت آمیز باشد، ولی کامپیوتر و/ یا توکن در یک حالت ناسازگار باشند.
- CKR_HOST_MEMORY: کامپیوتری که کتابخانه Cryptoki در آن در حال اجرا است حافظه کافی برای انجام تابع درخواستی ندارد.
- CKR_FUNCTION_FAILED: هنگامی که تابع فراخوانی شده اجرا نمی شود، اما اطلاعات دقیقی در مورد چرایی آن موجود نیست.
- CKR_OK: هنگامی که تابع با موفقیت اجرا شود. تمامی توابع کتابخانه Cryptoki، به جز توابع قدیمی C_GetFunctionStatus و C_CancelFunction، در صورت اجرای موفقیت آمیز این مقدار را برمی گردانند.

۲.۶.۲ مقادیر بازگشتی توابعی که از شناسه نشست استفاده می کنند

هر تابع Cryptoki که شناسه نشست را به عنوان یکی از آرگومان های خود در نظر می گیرد (تمام توابع Cryptoki به غیر از توابع: C_Initialize، C_Finalize، C_GetInfo، C_GetFunctionList، C_GetSlotList، C_GetSlotInfo، C_GetTokenInfo، C_WaitForStoSnews، C_SetSventSvent، C_CetSnews، C_FetSistNet، C_SetEve، C_FetListSvent، C_SetEnSial، C_FetSistNet، C_FetSistNet، C_SetEnSial، C_CetSnews، C_SetSnews، C_SetSetSloSnews، C_SetSnews، C_SetEnSlo، TheCentSet، C_SetSnoftE، C_CetSnews، C_SetSnoft، C_SNetSloSvent، C_CetSnews، C_CetSnews) ممکن است هر یک از مقادیر زیر را برگرداند:

- **CKR_SESSION_HANDLE_INVALID**: شناسه نشست مشخص شده در زمان فراخوانی تابع نامعتبر باشد.
- **CKR_DEVICE_REMOVED**: توکن هنگام اجرای تابع از اسلات کامپیوتر جدا شود.

- **CKR_SESSION_CLOSED**: نشست در هنگام اجرای تابع بسته شود.

۲.۶.۳ مقادیر بازگشتی برای توابعی که از توکن استفاده می کنند

هر تابع Cryptoki که از یک توکن مشخص استفاده می کند (تمامی توابع Cryptoki به غیر از توابع: C_Initialize، C_Finalize، C_GetInfo، C_GetFunctionList، C_GetSlotList، C_GetSlotInfo یا C_WaitForSlotEvent) ممکن است هر یک از مقادیر زیر را برگرداند::

- **CKR_DEVICE_MEMORY**: توکن برای انجام تابع درخواستی حافظه کافی ندارد.
- **CKR_DEVICE_ERROR**: مشکلی در توکن و/یا اسلات اتصال به کامپیوتر رخ داده است. به طور خاص، امکان بازگشت این خطا برای تابع C_GetSlotInfo بیشتر است.
- **CKR_TOKEN_NOT_PRESENT**: توکن در زمان فراخوانی تابع در اسلات خود وجود ندارد.
- **CKR_DEVICE_REMOVED**: توکن هنگام اجرای تابع از اسلات خود برداشته شود.

۲.۶.۴ مقدار برگشتی ویژه

- **CKR_FUNCTION_CANCELED**: این خطا توسط هیچ یک از توابع کتابخانه Cryptoki مستقیما بازگشت داده نمی شود، بلکه توسط برنامه کاربردی برگشت داده می شود. وقتی برنامه کاربردی تصمیم می گیرد برای اجرای یک عملکرد مشخص یک وقفه ایجاد نماید، مستقل از نتایج اجرای تابع فراخوانده شده این خطا را برمی گرداند.

۲.۶.۵ مقادیر برگشتی ویژه برای توابع کنترل Mutex

Mutex یک نوع خاص متغیر است که توسط سیستم عامل پشتیبانی می شود و وظیفه آن حفاظت از ناحیه بحرانی است. وقتی که داده مشترک بین چند نخ (Thread) وجود داشته باشد، باید دسترسی به این داده حفاظت شود. Mutex متغیری است که بین این نخ‌ها مشترک است. هر نخ که بخواهد از داده‌های مشترک استفاده کند، باید Mutex را قفل کند. در اینصورت هیچ نخ دیگری نمی تواند به داده مشترک دسترسی پیدا کند. وقتی که نخ اولی که Mutex را قفل کرده است، کارش با متغیر مشترک تمام شد، باید مجددا Mutex رو آزاد کند تا بقیه نخ‌ها امکان استفاده از داده مشترک را داشته باشند.

دو مقدار برگشتی خاص وجود دارد که توسط هیچ یک از توابع واقعی Cryptoki بازگردانده نمی شوند. این مقادیر توسط توابع کنترل mutex استفاده شده توسط برنامه کاربردی بازگردانده می شوند:

- CKR_MUTEX_BAD: هنگامی که آرگومان پاس شده به توابع کنترل mutex نادرست باشد، این خطا توسط توابع کنترل mutex برگشت داده می شود.

- CKR_MUTEX_NOT_LOCKED: این کد خطا توسط توابع باز کردن قفل mutex بازگشت داده می شو. این خطا نشان می دهد که mutex قفل نشده است.

۲.۶.۶ سایر مقادیر بازگشتی توابع Cryptoki

به استثنای آنچه در توصیف کد خطاهای خاص در بالا ذکر شد، سایر مقادیر بازگشتی توابع Cryptoki در ادامه شرح داده می شود. به طور کلی هیچ اولویت خاصی در میان خطاهای ذکر شده در زیر وجود ندارد.

- CKR_ACTION_PROHIBITED: این مقدار فقط توسط C_CopyObject، C_SetAttributeValue و C_DestroyObject بازگشت داده می شود. این خطا نشان می دهد که اجرای تابع مورد نظر با محدودیت های سیاستی توکن مواجه است. اگر ویژگی‌های CKA_COPYABLE، CKA_MODIFIABLE یا CKA_DESTROYABLE متعلق به یک شیء با مقدار CK_FALSE تنظیم شوند، سبب بروز این خطا می شود.
- CKR_ARGUMENTS_BAD: این یک کد خطای کاملاً عمومی است که نشان می دهد آرگومانهای ارائه شده به تابع به نوعی مناسب نیستند.
- CKR_ATTRIBUTE_READ_ONLY: اگر تابعی سعی در تنظیم یا اصلاح مقدار یک ویژگی فقط-خواندنی نماید، این خطا بازگشت داده می شود.
- CKR_ATTRIBUTE_SENSITIVE: تلاش برای بدست آوردن مقدار ویژگی یک شی که اجازه دسترسی به آن وجود ندارد زیرا این شی حساس است یا غیرقابل استخراج است.
- CKR_ATTRIBUTE_TYPE_INVALID: اگر نوع ویژگی مورد نظر در یک الگو نامعتبر باشد، این خطا برگشت داده می شود.
- CKR_ATTRIBUTE_VALUE_INVALID: اگر یک مقدار نامعتبر برای یک ویژگی معتبر در یک الگو مشخص شده باشد، این خطا برگشت داده می شود.
- CKR_BUFFER_TOO_SMALL: اندازه خروجی تابع بسیار بزرگ است و نمی تواند در بافر تعیین شده قرار گیرد.
- CKR_CANT_LOCK: این مقدار را فقط تابع C_Initialize می تواند برگرداند. این بدان معناست که نوع قفل درخواست شده توسط برنامه برای امن کردن ترد در این کتابخانه موجود نیست، بنابراین برنامه نمی تواند به روش مشخص از این کتابخانه استفاده کند.
- CKR_CRYPTOKI_ALREADY_INITIALIZED: این مقدار را فقط تابع C_Initialize می تواند برگرداند. این بدان معناست که کتابخانه Cryptoki قبلاً راه اندازی شده است (با فراخوانی قبلی با C_Initialize که فراخوانی C_Finalize را به دنبال نداشته است).
- CKR_CRYPTOKI_NOT_INITIALIZED: این مقدار توسط هر تابعی غیر از توابع C_Initialize، C_GetFunctionList، C_GetInterfaceList و C_GetInterface قابل بازگشت است. این نشان می دهد که تابع نمی تواند اجرا شود، زیرا کتابخانه Cryptoki هنوز با فراخوانی C_Initialize راه اندازی نشده است.
- CKR_CURVE_NOT_SUPPORTED: این کد خطا برای مکانیسم رمزنگاری منحنی بیضوی استفاده می شود. این کد خطا بدان معناست که منحنی بیضوی توسط این توکن پشتیبانی نمی شود.

- CKR_DATA_INVALID: داده های ورودی متن اصلی به یک تابع رمزنگاری نامعتبر است. این مقدار بازگشتی نسبت به CKR_DATA_LEN_RANGE دارای اولویت پایین تری است.
- CKR_DATA_LEN_RANGE: طول داده های ورودی متن اصلی برای یک تابع رمزنگاری نادرست است. بسته به مکانیزم رمزنگاری، این کد خطا می تواند به این معنی باشد که داده های متن اصلی خیلی کوتاه یا خیلی بلند است یا مضربی از طول بلوک معین نیست. این مقدار بازگشتی نسبت به CKR_DATA_INVALID دارای اولویت بیشتری است.
- CKR_DOMAIN_PARAMS_INVALID: هنگامی که پارامترهای دامنه نامعتبر یا پشتیبانی نشده به تابع ارائه شود.
- CKR_ENCRYPTED_DATA_INVALID: ورودی متن رمزی در یک عملیات رمزگشایی نامعتبر تشخیص داده می شود. این مقدار بازگشتی نسبت به CKR_ENCRYPTED_DATA_LEN_RANGE دارای اولویت کمتری است.
- CKR_ENCRYPTED_DATA_LEN_RANGE: طول متن رمزی ورودی به تابع رمزگشایی نامعتبر شناسایی شده است. بسته به مکانیسم رمزنگاری ، این کد خطا می تواند به این معنی باشد که متن رمز خیلی کوتاه یا خیلی بلند است یا مضربی از اندازه بلوک معین نیست. این مقدار بازگشتی دارای اولویت بیشتری نسبت به CKR_ENCRYPTED_DATA_INVALID است.
- CKR_EXCEEDED_MAX_ITERATIONS: به دلیل اینکه تعداد تکرار در الگوریتم تکرار شونده (برای تولید زوج کلید، تولید پارامتر دامنه و غیره) از حداکثر تعداد مجاز بیشتر شده است. این کد خطا نسبت به CKR_FUNCTION_FAILED دارای اولویت بالاتری است.
- CKR_FIPS_SELF_TEST_FAILED: خودآزمایی مبتنی بر FIPS 140-2 یا خودآزمایی مشروط با شکست مواجه شده و توکن وارد حالت خطا می شود. فراخوان های بعدی توابع رمزنگاری در توکن با خطای CKR_GENERAL_ERROR برگشت داده می شود. کد خطای CKR_FIPS_SELF_TEST_FAILED نسبت به CKR_GENERAL_ERROR دارای اولویت بالاتری است. این خطا ممکن است توسط C_Initialize، در صورت عدم موفقیت در تست خودآزمایی، با C_GenerateRandom یا C_SeedRandom، در صورت عدم موفقیت در تولید مولد اعداد تصادفی پیوسته، یا توسط C_GenerateKeyPair، در صورت عدم موفقیت در آزمون تناظر زوج کلید، بازگشت داده می شود.
- CKR_FUNCTION_CANCELED: هنگامی که تابع در میانه اجرا لغو شود. این کد خطا در ارتباط با کد خطای CKR_CANCEL است (به CKR_CANCEL مراجعه کنید).
- CKR_FUNCTION_NOT_PARALLEL: در نسخه فعلی کتابخانه Cryptoki هیچ تابعی به طور موازی در یک نشست معین اجرا نمی شود. این یک کد خطای قدیمی است که فقط توسط توابع قدیمی C_GetFunctionStatus و C_CancelFunction بازگردانده می شود.
- CKR_FUNCTION_NOT_SUPPORTED: تابع فراخوانی شده توسط این کتابخانه Cryptoki پشتیبانی نمی شود.
- CKR_FUNCTION_REJECTED: درخواست امضا توسط کاربر رد می شود.
- CKR_INFORMATION_SENSITIVE: امکان دستیابی به اطلاعات درخواستی، به دلیل حساس بودن یا غیرقابل آشکار شدن، وجود ندارد.
- CKR_KEY_CHANGED: این کد خطا فقط توسط تابع C_SetOperationState برگردانده می شود. این نشان می دهد که یکی از کلیدهای مشخص شده با کلید مورد استفاده در نشست ذخیره شده اصلی یکسان نیست.

- CKR_KEY_FUNCTION_NOT_PERMITTED: سعی شده از کلیدی برای مزنگاری استفاده شود که برای این هدف تنظیم نشده است. به عنوان مثال، برای استفاده از یک کلید برای انجام رمزگذاری، باید ویژگی CKA_ENCRYPT آن کلید روی CK_TRUE تنظیم شود. این مقدار بازگشتی نسبت به CKR_KEY_TYPE_INCONSISTENT دارای اولویت کمتری است.
- CKR_KEY_HANDLE_INVALID: شناسه کلید مشخص شده معتبر نیست. ممکن است این حالت وجود داشته باشد که شناسه مشخص شده یک شناسه معتبر باشد ولی شناسه یک کلید نیست. باید توجه داشت که مقدار صفر (۰) هرگز یک شناسه کلید معتبر نیست.
- CKR_KEY_INDIGESTIBLE: این کد خطا فقط توسط تابع C_DigestKey قابل بازگشت است. این نشان می دهد که به دلایلی نمی توان مقدار کلید مشخص شده را فشرده کند (شاید کلید مشخص شده یک کلید مخفی نباشد، یا شاید توکن به راحتی نتواند این نوع کلید را فشرده کند).
- CKR_KEY_NEEDED: این مقدار فقط توسط تابع C_SetOperationState برگردانده می شود. این نشان می دهد که حالت نشست نمی تواند بازیابی شود زیرا این تابع برای اجرا نیاز به ورود کلید دارد.
- CKR_KEY_NOT_NEEDED: یک کلید اضافی به C_SetOperationState ارائه شده است. به عنوان مثال، تابع چکیده ساز (Hash) نیاز به کلید ندارد، در چنین مکانیزمی اگر یک کلید به عنوان ورودی داده شود، چنین کد خطایی برگشت داده می شود.
- CKR_KEY_NOT_WRAPPABLE: اگرچه ویژگی CKA_EXTRACTABLE برای کلید خصوصی یا مخفی مشخص شده روی CK_FALSE تنظیم نشده است، اما کتابخانه Cryptoki (یا توکن) قادر نیست کلید را کپسوله کند. (با CKR_KEY_UNEXTRACTABLE مقایسه شود).
- CKR_KEY_SIZE_RANGE: اندازه کلید ارائه شده خارج از محدوده مجاز می باشد.
- CKR_KEY_TYPE_INCONSISTENT: نوع کلید ورودی برای استفاده در مکانیزم تعیین شده همخوانی ندارد. این کد خطا نسبت به کد خطای CKR_KEY_FUNCTION_NOT_PERMITTED دارای اولویت بالاتری است.
- CKR_KEY_UNEXTRACTABLE: کلید خصوصی یا کلید مخفی مشخص شده نمی تواند کپسوله شود، زیرا ویژگی CKA_EXTRACTABLE کلید روی CK_FALSE تنظیم شده است. (با CKR_KEY_NOT_WRAPPABLE مقایسه شود).
- CKR_LIBRARY_LOAD_FAILED: کتابخانه Cryptoki نمی تواند یک کتابخانه مشترک وابسته را بارگیری کند.
- CKR_MECHANISM_INVALID: مکانیزم نامعتبری برای عملیات رمزنگاری مشخص شده است. اگر مکانیزم ناشناخته ای مشخص شده باشد یا مکانیزم مشخص شده در توکن انتخاب شده قابل استفاده نباشد، این کد خطا بازگشت داده می شود.
- CKR_MECHANISM_PARAM_INVALID: پارامترهای نامعتبر به مکانیزم مشخص شده برای عملیات رمزنگاری ارائه شده است.
- CKR_NEED_TO_CREATE_THREADS: این مقدار فقط توسط C_Initialize در شرایط زیر برگشت داده می شود:

۱. برنامه ای که C_Initialize را فراخوانی می کند، به کتابخانه Cryptoki اعلام می کند که بدون ایجاد رشته (Tread) جدید قادر به اجرای فرایند نیست.

۲. کتابخانه Cryptoki بدون ایجاد رشته (Tread) جدید، نمی تواند به درستی کار کند.

- CKR_NO_EVENT: این مقدار را فقط تابع C_WaitForSlotEvent می تواند برگرداند. وقتی C_WaitForSlotEvent در حالت غیر مسدود (پرچم CKF_DONT_BLOCK تنظیم شده باشد) فراخوانی می شود و هیچ رویداد اسلات جدیدی رخ ندهد ، کد CKR_NO_EVENT برگشت داده می شود.
- CKR_OBJECT_HANDLE_INVALID: شناسه شی مشخص شده معتبر نیست. یادآوری می شود که مقدار صفر (۰) هرگز یک شناسه شی معتبر نیست.
- CKR_OPERATION_ACTIVE: هنگامی یک عملیات فعال (یا ترکیبی از عملیات فعال) وجود دارد که مانع از فعال سازی عملیات مشخص شده Cryptoki می شود. به عنوان مثال، وقتی که یک عملیات جستجوی شی فعال، از فعال کردن رمزگذاری با تابع C_EncryptInit جلوگیری می کند.
- CKR_OPERATION_NOT_INITIALIZED: در نشست مشخص شده فعالیتی از نوع مناسب وجود ندارد. به عنوان مثال، یک برنامه نمی تواند C_Encrypt را در یک نشست فراخوانی کند بدون اینکه ابتدا C_EncryptInit را برای فعال سازی عملیات رمزگذاری فراخوانی کرده باشد.
- CKR_PIN_EXPIRED: پین مشخص شده منقضی شده و عملیات درخواستی نمی تواند انجام شود، مگر اینکه C_SetPIN برای تغییر مقدار پین فراخوانی شود. اینکه پین کاربر عادی روی یک توکن منقضی شود یا خیر، از توکنی به توکن دیگر متفاوت است.
- CKR_PIN_INCORRECT: پین مشخص شده نادرست است، یعنی با پین ذخیره شده در توکن منطبق نیست. به طور کلی - هنگامی که راستی آزمایی اعتبار در توکن به غیر از پین شامل عامل دیگری باشد - تلاش برای احراز هویت کاربر ناموفق است.
- CKR_PIN_INVALID: پین مشخص شده دارای کاراکترهای نامعتبر است. این کد بازگشتی فقط برای توابعی اعمال می شود که سعی در تنظیم پین دارند.
- CKR_PIN_LEN_RANGE: اندازه پین مشخص شده خارج از محدوده است. این کد بازگشتی فقط برای عملکردهایی اعمال می شود که سعی در تنظیم PIN دارند.
- CKR_PIN_LOCKED: پین مشخص شده "قفل شده" است و قابل استفاده نیست. به این دلیل که تعداد مشخصی از تلاشهای راستی آزمایی اعتبار ناموفق حاصل شده است. بسته به سیاست امنیتی توکن، پین مشخص شده می تواند به طور نامحدود با محدود قفل شود.
- CKR_PIN_TOO_WEAK: پین مشخص شده بسیار ضعیف است به طوری که حدس زدن آن آسان است. اگر پین خیلی کوتاه باشد، باید CKR_PIN_LEN_RANGE به جای آن برگردانده شود. این کد بازگشت فقط برای عملکردهایی اعمال می شود که سعی در تنظیم PIN دارند.
- CKR_PUBLIC_KEY_INVALID: کلید عمومی مشخص شده یک کلید عمومی معتبر نیست. این کد خطا ممکن است توسط C_CreateObject، هنگام ایجاد کلید عمومی، یا توسط C_VerifyInit یا C_VerifyRecoverInit، هنگام استفاده از کلید عمومی، برگردانده شود. هنگام فراخوانی C_DeriveKey، اگر کلید عمومی طرف مقابل در پارامترهای مکانیزم رمزنگاری نامعتبر باشد، بجای کد خطای CKR_MECHANISM_PARAM_INVALID، کد خطا CKR_PUBLIC_KEY_INVALID برگشت داده می شود.

- CKR_RANDOM_NO_RNG: این مقدار توسط C_SeedRandom و C_GenerateRandom برگردانده می شود. این نشان می دهد که توکن مشخص شده داری مکانیزم مولد اعداد تصادفی نیست. این مقدار بازگشتی نسبت به CKR_RANDOM_SEED_NOT_SUPPORTED دارای اولویت بالاتری است.
- CKR_RANDOM_SEED_NOT_SUPPORTED: این مقدار فقط توسط C_SeedRandom قابل بازگشت است. این نشان می دهد که مکانیزم مولد اعداد تصادفی توکن بذر اولیه (Seed) را از بیرون توکن قبول نمی کند. این مقدار بازگشتی نسبت به CKR_RANDOM_NO_RNG دارای اولویت کمتری است.
- CKR_SAVED_STATE_INVALID: این مقدار فقط توسط C_SetOperationState می تواند برگشت داده شود. این نشان می دهد که حالت عملیات رمزنگاری ذخیره شده نامعتبر است، بنابراین نمی توان آن را در نشست مشخص شده بازیابی کرد.
- CKR_SESSION_COUNT: این مقدار فقط توسط C_OpenSession می تواند برگشت داده شود. این نشان می دهد که تلاش برای باز کردن یک نشست به نتیجه نرسیده است، به این دلیل که توکن بیش از حد نشست را باز کرده است، یا به دلیل اینکه توکن بیش از حد عملیات خواندن/نوشتن از نشست های باز شده در حال اجرا دارد.
- CKR_SESSION_EXISTS: این مقدار فقط توسط C_InitToken می تواند برگشت داده شود. این نشان می دهد که یک نشست از قبل باز است و بنابراین نمی تواند دوباره باز شود.
- CKR_SESSION_PARALLEL_NOT_SUPPORTED: توکن نشست های موازی را پشتیبانی نمی کند. این یک کد خطای قدیمی است - در Cryptoki نسخه ۲.۰۱ و بالاتر ، هیچ توکنی از نشست های موازی پشتیبانی نمی کند. CKR_SESSION_PARALLEL_NOT_SUPPORTED فقط توسط C_OpenSession قابل بازگشت است و فقط وقتی C_OpenSession به روشی خاص [منسوخ] فراخوانی می شود.
- CKR_SESSION_READ_ONLY: نشست مشخص شده نتوانست اقدام مورد نظر را انجام دهد زیرا این یک نشست فقط-خواندنی است. این مقدار بازگشتی نسبت به CKR_TOKEN_WRITE_PROTECTED دارای اولویت پایین تری است.
- CKR_SESSION_READ_ONLY_EXISTS: یک نشست فقط-خواندنی از قبل وجود دارد و بنابراین کاربر مامور امنیتی (SO) نمی تواند به توکن وارد شود (لاگین کند).
- CKR_SESSION_READ_WRITE_SO_EXISTS: یک نشست خواندن/نوشتن SO از قبل وجود دارد و بنابراین یک نشست فقط-خواندنی باز نمی شود.
- CKR_SIGNATURE_LEN_RANGE: امضا/MAC ارائه شده فقط بر اساس طول آن فاقد اعتبار است. این مقدار بازگشتی دارای اولویت بالاتری نسبت به CKR_SIGNATURE_INVALID است.
- CKR_SIGNATURE_INVALID: امضا/MAC ارائه شده نامعتبر است. این مقدار بازگشتی نسبت به CKR_SIGNATURE_LEN_RANGE دارای اولویت پایین تری است.
- CKR_SLOT_ID_INVALID: شناسه اسلات مشخص معتبر نیست.
- CKR_STATE_UNSAVEABLE: به دلایلی حالت عملیات رمزنگاری نشست مشخص را نمی توان ذخیره کرد (احتمالاً توکن نمی تواند حالت فعلی را ذخیره کند). این مقدار بازگشتی نسبت به CKR_OPERATION_NOT_INITIALIZED اولویت پایین تری دارد.

- CKR_TEMPLATE_INCOMPLETE: الگویی که برای ایجاد یک شی مشخص شده ناقص است و فاقد برخی ویژگی های لازم است.
- CKR_TEMPLATE_INCONSISTENT: الگویی که برای ایجاد یک شی مشخص شده دارای ویژگی های متناقضی است.
- CKR_TOKEN_NOT_RECOGNIZED: کتابخانه Cryptoki و/یا اسلات توکن موجود در اسلات را تشخیص نمی دهد.
- CKR_TOKEN_WRITE_PROTECTED: اقدام درخواستی انجام نمی شود زیرا توکن از نوشتن محافظت می کند، زیرا پرچم CKF_WRITE_PROTECTED با CK_TRUE مقداردهی شده است. این مقدار بازگشتی اولویت بالاتری نسبت به CKR_SESSION_READ_ONLY دارد.
- CKR_UNWRAPPING_KEY_HANDLE_INVALID: این مقدار را فقط C_UnwrapKey می تواند برگرداند. این نشان می دهد که شناسه کلید مشخص شده برای استفاده از آن برای باز کردن کلید دیگر معتبر نیست.
- CKR_UNWRAPPING_KEY_SIZE_RANGE: این مقدار را فقط C_UnwrapKey می تواند برگرداند. این نشان می دهد که اگرچه عملیات باز کردن درخواستی در اصل می تواند انجام شود، اما کتابخانه Cryptoki (یا توکن) واقعاً قادر به انجام آن نیست زیرا اندازه کلید ارائه شده خارج از محدوده اندازه کلید است.
- CKR_UNWRAPPING_KEY_TYPE_INCONSISTENT: این مقدار را فقط C_UnwrapKey می تواند برگرداند. این نشان می دهد که نوع کلید مشخص شده برای باز کردن کلید دیگر با سازوکار مشخص شده برای باز کردن بسته همخوانی ندارد.
- CKR_USER_ALREADY_LOGGED_IN: این مقدار فقط توسط C_Login قابل بازگشت است. این نشان می دهد که کاربر مشخص شده نمی تواند به نشست وارد شود، زیرا قبلاً در نشست وارد شده است.
- CKR_USER_ANOTHER_ALREADY_LOGGED_IN: این مقدار فقط توسط C_Login قابل بازگشت است. این نشان می دهد که کاربر مشخص شده نمی تواند در نشست وارد شود، زیرا کاربر دیگری قبلاً در نشست وارد شده است.
- CKR_USER_NOT_LOGGED_IN: اقدام مورد نظر نمی تواند انجام شود زیرا یک کاربر مناسب به سیستم وارد نشده است. در واقع اقداماتی که نیازمند ورود کاربر هستند را محافظت می کند.
- CKR_USER_PIN_NOT_INITIALIZED: این مقدار را فقط C_Login می تواند برگرداند. این نشان می دهد که PIN کاربر عادی هنوز با C_InitPIN تنظیم نشده است.
- CKR_USER_TOO_MANY_TYPES: سعی می شود که کاربران مجزا فراتر از مجوزهای توکن و/یا کتابخانه Cryptoki به طور همزمان به توکن وارد شوند. به عنوان مثال، اگر برخی از برنامه ها دارای یک نشست SO باز هستند، و برنامه دیگری سعی دارد کاربر عادی را به یک نشست وارد کند، ممکن است این خطا را بازگرداند. اما این یک الزام نیست. فقط اگر کاربران مجزا همزمان توسط توکن پشتیبانی نشوند، C_Login مجبور است این مقدار را برگرداند. توجه داشته باشید که این کد خطا به توکن های چند کاربره واقعی مربوط می شود.
- CKR_USER_TYPE_INVALID: یک مقدار نامعتبر برای ویژگی CK_USER_TYPE مشخص شده باشد. مقادیر معتبر عبارتند از: CKU_SO، CKU_USER و CKU_CONTEXT_SPECIFIC.
- CKR_WRAPPED_KEY_INVALID: این مقدار را فقط C_UnwrapKey می تواند برگرداند. این نشان می دهد که کلید بسته بندی شده معتبر نیست. اگر برای باز کردن نوع خاصی از کلید، تابع C_UnwrapKey فراخوانی شود (یعنی نوع

خاصی از کلید در الگوی ارائه شده به C_UnwrapKey مشخص شده باشد)، و کلید بسته بندی شده به C_UnwrapKey مشخصاً یک کلید بسته بندی شده از نوع مناسب نیست، آنگاه C_UnwrapKey باید CKR_WRAPPED_KEY_INVALID را برگرداند. این مقدار بازگشتی نسبت به CKR_WRAPPED_KEY_LEN_RANGE دارای اولویت پایین تری است.

- CKR_WRAPPED_KEY_LEN_RANGE: این مقدار را فقط C_UnwrapKey می تواند برگرداند. این نشان می دهد که کلید بسته بندی شده ارائه شده فقط بر اساس اندازه آن نامعتبر است. این مقدار بازگشتی نسبت به CKR_WRAPPED_KEY_INVALID دارای اولویت بالاتری است.
- CKR_WRAPPING_KEY_HANDLE_INVALID: این مقدار را فقط C_WrapKey می تواند برگرداند. این نشان می دهد که شناسه کلید مشخص شده برای بسته بندی کلید دیگر معتبر نیست.
- CKR_WRAPPING_KEY_SIZE_RANGE: این مقدار را فقط C_WrapKey می تواند برگرداند. این نشان می دهد که اگرچه در اصل می توان عملیات بسته بندی درخواستی را انجام داد، اما کتابخانه Cryptoki (یا توکن) واقعاً قادر به انجام آن نیست زیرا اندازه کلید بسته بندی ارائه شده خارج از محدوده مجاز است.
- CKR_WRAPPING_KEY_TYPE_INCONSISTENT: این مقدار را فقط C_WrapKey می تواند برگرداند. این نشان می دهد که نوع کلید تعیین شده برای بسته بندی کلید دیگر با سازوکار مشخص شده برای بسته بندی سازگار نیست.
- CKR_OPERATION_CANCEL_FAILED: این مقدار فقط توسط C_SessionCancel قابل بازگشت است. این بدان معنی است که یک یا چند عملیات درخواستی به دلایل پیاده سازی یا دلایل مخصوص سازنده لغو نمی شوند.

۲.۶.۷ توابعی که یک خروجی تولید و در یک بافر با طول متغیر باز می گردانند

تعدادی از توابع تعریف شده در Cryptoki، خروجی تولید شده توسط مکانیزم رمزنگاری را در یک بافر باز می گردانند. پارامتر بازگشتی توسط این توابع دارای طول متغیر است. نمونه ای از این توابع، تابع C_Encrypt است که متن ساده ای را به عنوان ورودی دریافت و یک بافر حاوی متن رمزگذاری شده تولید می کند. در این توابع دو مولفه ورودی تعریف می شود، یک اشاره گر بافر خروجی (pBuf) برای نگهداری خروجی تابع و یک اشاره گر برای اندازه خروجی (pulBufLen) که تعداد بایت های مورد نیاز برای نگهداری خروجی تابع را مشخص می کند. این دو مولفه به روش های زیر می توانند تنظیم شوند:

۱. اشاره گر pBuf دارای مقدار NULL_PTR باشد و اشاره گر pulBufLen دارای اندازه خروجی باشد. در این صورت، تابع مقدار خروجی را در بافر قرار داده و کد CKR_OK را بر می گرداند.
۲. اشاره گر pBuf دارای مقداری غیر از NULL_PTR باشد و اشاره گر pulBufLen اندازه بافر را مشخص کند. در این صورت، اگر ظرفیت بافر برای نگهداری خروجی تابع کافی باشد، مقدار خروجی در آن قرار گرفته و CKR_OK توسط تابع برگشت داده می شود. اگر ظرفیت بافر به اندازه کافی نباشد، کد خطای CKR_BUFFER_TOO_SMALL بازگردانده می شود.

در هر دو حالت، pulBufLen تعداد دقیق بایت های مورد نیاز برای نگهداری خروجی تابع را مشخص می کند.

۳ توابع کتابخانه Cryptoki

در جدول زیر تمامی توابع کتابخانه Cryptoki تشریح شده است. برای پیاده سازی و ارزیابی هر یک از این توابع باید تمامی موارد ذکر شده در ستون شواهد ارزیابی مورد توجه قرار گیرد.

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
توابع عام	P 5.4.1	C_Initialize	• این تابع کتابخانه Cryptoki را مقدار دهی اولیه و تنظیم می کند. • C_Initialize باید اولین فراخوانی Cryptoki باشد که توسط یک برنامه کاربردی اعمال می شود. البته برای توابع C_GetFunctionList، C_GetInterfaceList یا C_GetInterface این الزام وجود ندارد. • C_Initialize دارای دو پرچم است: ۱. CKF_LIBRARY_CANT_CREATE_OS_THREADS: اگر با CK_TRUE تنظیم شود به این معنی است که کتابخانه Cryptoki مجاز به استفاده از فراخوانی های محلی سیستم عامل برای ایجاد رشته های جدید نیست. اگر کتابخانه تحت این محدودیت قادر به عملکرد صحیح نباشد ، C_Initialize باید با مقدار CKR_NEED_TO_CREATE_THREADS بازگردد. ۲. CKF_OS_LOCKING_OK: اگر با CK_TRUE تنظیم شود به این معنی است که کتابخانه Cryptoki مجاز به استفاده از مدل چند رشته سازی محلی سیستم عامل برای متغیرهای Mutext می باشد. در صورتی که کتابخانه قادر به انجام این کار نباشد، C_Initialize باید مقدار CKR_CANT_LOCK بازگردد. • اگر برخی، اما نه همه، از اشاره گرهای ارائه شده به C_Initialize غیر

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
			NULL_PTR باشند، آنگاه مقدار CKR_ARGUMENTS_BAD بازگشت داده می شود. • اگر چندین برنامه از Cryptoki استفاده می کنند ، هر یک باید C_Initialize را فراخوانی کنند. هر فراخوانی C_Initialize باید (در نهایت) با یک فراخوانی C_Finalize پایان پذیرد. مقادیر بازگشتی برای فراخوانی ناموفق: • CKR_CANT_LOCK • CKR_ARGUMENTS_BAD • CKR_FUNCTION_FAILED • CKR_GENERAL_ERROR • CKR_CRYPTOKI_ALREADY_INITIALIZED مقادیر بازگشتی برای فراخوانی موفق: • CKR_OK
	P 5.4.2	**C_Finalize**	• این تابع منابع متفرقه مرتبط با کتابخانه Cryptoki را پاک می کند. • در این تابع، پارامتر pReservation باید روی NULL_PTR تنظیم شود. اگر تابع C_Finalize با مقدار غیر NULL_PTR برای pReservation فراخوانی شود، کد خطای CKR_ARGUMENTS_BAD برگشت داده می شود. • اگر قبل از این تابع، تابع C_Initialize فراخوانی نشده باشد، مقدار CKR_CRYPTOKI_NOT_INITIALIZED برگشت داده می شود. • مقادیر بازگشتی برای فراخوانی ناموفق: • CKR_ARGUMENTS_BAD • CKR_CRYPTOKI_NOT_INITIALIZED • CKR_FUNCTION_FAILED • CKR_GENERAL_ERROR • CKR_HOST_MEMORY مقادیر بازگشتی برای فراخوانی موفق: • CKR_OK
	P 5.4.3	**C_ GetInfo**	• این تابع اطلاعات کلی در مورد Cryptoki بدست می آورد. • این تابع اطلاعات زیر را درباره کتابخانه Cryptoki برمی گرداند: – cryptokiVersion: شماره نسخه رابط Cryptoki، – ManufacturerID: شناسه سازنده کتابخانه Cryptoki. – flags: پرچم های بیتی مختص نسخه های بعدی. برای این نسخه باید صفر باشد. – LibraryDescription: توصیف کتابخانه Cryptoki. – libraryVersion: شماره نسخه کتابخانه Cryptoki. • قبل از فراخوانی این تابع، باید تابع C_Initialize فراخوانی شود. در غیر

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
			اینصورت، کد خطای CKR_CRYPTOKI_NOT_INITIALIZED برگشت داده می شود. • مقادیر بازگشتی برای فراخوانی ناموفق: • CKR_ARGUMENTS_BAD • CKR_CRYPTOKI_NOT_INITIALIZED • CKR_FUNCTION_FAILED • CKR_GENERAL_ERROR • CKR_HOST_MEMORY • مقادیر بازگشتی برای فراخوانی موفق: • CKR_OK
	P 5.4.4	**C_ GetFunctionList**	• این تابع لیستی از توابع کتابخانه Cryptoki را بدست می آورد. • این تابع از معدود توابعی است که قبل از تابع C_Initialize می تواند فراخوانی شود. • لیست توابع برگشتی باید با لیست توابع نسخه ۳ کتابخانه Cryptoki مطابقت داشته باشد (پیوست الف). • مقادیر بازگشتی برای فراخوانی ناموفق: • CKR_ARGUMENTS_BAD • CKR_FUNCTION_FAILED • CKR_GENERAL_ERROR • CKR_HOST_MEMORY • مقادیر بازگشتی برای فراخوانی موفق: • CKR_OK
	P 5.4.5	**C_GetInterfaceList**	• این تابع لیستی از رابط های پشتیبانی شده توسط کتابخانه Cryptoki را بدست می آورد. • این تابع از معدود توابعی است که قبل از تابع C_Initialize می تواند فراخوانی شود. • اگر اندازه بافر خروجی برای مقدار خروجی تابع کافی نباشد، کد خطای CKR_BUFFER_TOO_SMALL برگشت داده می شود. • مقادیر بازگشتی برای فراخوانی ناموفق: • CKR_BUFFER_TOO_SMALL • CKR_ARGUMENTS_BAD • CKR_FUNCTION_FAILED • CKR_GENERAL_ERROR • CKR_HOST_MEMORY • مقادیر بازگشتی برای فراخوانی موفق: • CKR_OK
	P 5.4.6	**C_GetInterface**	• این تابع رابط کاربری خاص نقاط ورودی کتابخانه Cryptoki را بدست می آورد. • این تابع از معدود توابعی است که قبل از تابع C_Initialize می تواند

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
			فراخوانی شود. • اگر اندازه بافر خروجی برای مقدار خروجی تابع کافی نباشد، کد خطای CKR_BUFFER_TOO_SMALL برگشت داده می شود. • مقادیر بازگشتی برای فراخوانی ناموفق: • CKR_BUFFER_TOO_SMALL • CKR_ARGUMENTS_BAD • CKR_FUNCTION_FAILED • CKR_GENERAL_ERROR • CKR_HOST_MEMORY • مقادیر بازگشتی برای فراخوانی موفق: • CKR_OK
توابع مدیریت توکن و اسلات	P 5.5.1	C_GetSlotList	• این تابع لیستی از اسلات های موجود در سیستم را بدست می آورد. • اگر اندازه بافر خروجی برای مقدار خروجی تابع کافی نباشد، کد خطای CKR_BUFFER_TOO_SMALL برگشت داده می شود. • مقادیر بازگشتی برای فراخوانی ناموفق: • CKR_ARGUMENTS_BAD • CKR_BUFFER_TOO_SMALL • CKR_CRYPTOKI_NOT_INITIALIZED • CKR_FUNCTION_FAILED • CKR_GENERAL_ERROR • CKR_HOST_MEMORY • مقادیر بازگشتی برای فراخوانی موفق: • CKR_OK
	P 5.5.2	C_GetSlotInfo	• این تابع اطلاعات مربوط به یک اسلات خاص را بدست می آورد. • اگر شناسه اسلات ورودی نادرست باشد، کد خطای CKR_SLOT_ID_INVALID برگشت داده می شود. • اگر مشکلی در توکن و/یا اسلات اتصال به کامپیوتر رخ دهد، کد خطای CKR_DEVICE_ERROR برگشت داده می شود. • مقادیر بازگشتی برای فراخوانی ناموفق: • CKR_ARGUMENTS_BAD • CKR_CRYPTOKI_NOT_INITIALIZED • CKR_DEVICE_ERROR • CKR_FUNCTION_FAILED • CKR_GENERAL_ERROR • CKR_HOST_MEMORY • CKR_SLOT_ID_INVALID

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
			• مقادیر بازگشتی برای فراخوانی موفق: • CKR_OK
	P 5.5.3	C_GetTokenInfo	• این تابع اطلاعات مربوط به یک توکن خاص را بدست می آورد. • اگر توکن برای انجام تابع درخواستی حافظه کافی نداشته باشد، کد خطای CKR_DEVICE_MEMORY برگشت داده می شود. • اگر مشکلی در توکن و/یا اسلات اتصال به کامپیوتر رخ دهد، کد خطای CKR_DEVICE_ERROR برگشت داده می شود. • اگر توکن در زمان فراخوانی تابع در اسلات خود وجود نداشته باشد، کد خطای CKR_TOKEN_NOT_PRESENT برگشت داده می شود. • اگر توکن هنگام اجرای تابع از اسلات خود برداشته شود، کد خطای CKR_DEVICE_REMOVED برگشت داده می شود. • اگر کتابخانه Cryptoki و/یا اسلات، توکن موجود در اسلات را تشخیص ندهد کد خطای CKR_TOKEN_NOT_RECOGNIZED رخ می دهد. • مقادیر بازگشتی برای فراخوانی ناموفق: • CKR_CRYPTOKI_NOT_INITIALIZED, • CKR_DEVICE_ERROR, • CKR_DEVICE_MEMORY, • CKR_DEVICE_REMOVED, • CKR_FUNCTION_FAILED, • CKR_GENERAL_ERROR, • CKR_HOST_MEMORY, • CKR_SLOT_ID_INVALID, • CKR_TOKEN_NOT_PRESENT, • CKR_TOKEN_NOT_RECOGNIZED, • CKR_ARGUMENTS_BAD • مقادیر بازگشتی برای فراخوانی موفق: • CKR_OK
	P 5.5.4	C_WaitForSlotEvent	• این تابع وقوع یک رویداد اسلات (درج توکن، حذف و غیره) را تشخیص می دهد.. • این تابع دارای پرچم CKF_DONT_BLOCK است. در صورتی که این تابع با تنظیم این پرچم با CK_TRUE فراخوانی شود، تابع بررسی می کند اگر رویداد اسلاتی رخ داده، شناسه آن اسلات را برمی گرداند، در غیراینصورت، کد CKR_NO_EVENT را برمی گرداند. • اگر در حین انتظار تابع C_WaitForSlotEvent برای یافتن رویداد اسلات،

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
			تابع C_Finalize فراخوانی شود، تابع C_WaitForSlotEvent مقدار CKR_CRYPTOKI_NOT_INITIALIZED را برمی گرداند. • مقادیر بازگشتی برای فراخوانی ناموفق: • CKR_ARGUMENTS_BAD, • CKR_CRYPTOKI_NOT_INITIALIZED, • CKR_FUNCTION_FAILED, • CKR_GENERAL_ERROR, • CKR_HOST_MEMORY, • CKR_NO_EVENT, • مقادیر بازگشتی برای فراخوانی موفق: • CKR_OK
	P 5.5.5	C_GetMechanismList	• این تابع لیست مکانیزم های قابل پشتیبانی توسط توکن را برمی گرداند. • برای فراخوانی C_GetMechanismList دو روش وجود دارد: ۱. اگر pMechanismList دارای مقدار NULL_PTR است، تعداد مکانیزم ها، نه لیست آنها، در pulCount، به همراه CKR_OK باید برگرداند. ۲. اگر pMechanismList دارای مقداری غیر از NULL_PTR است، آنگاه pulCount باید اندازه بافری را که pMechanismList به آن اشاره کرده است، داشته باشد. اگر آن بافر به اندازه کافی برای نگهداری لیست مکانیزم ها جا داشته باشد، لیست مکانیزم ها در آن به همراه CKR_OK بازگردانده می شود. در غیر این صورت یعنی ناکافی بودن فضای بافر، مقدار CKR_BUFFER_TOO_SMALL برگشت داده می شود. در هر صورت مقدار pulCount برای نگهداری تعداد مکانیزم ها تنظیم شده است. • مقادیر بازگشتی برای فراخوانی ناموفق: • CKR_BUFFER_TOO_SMALL, • CKR_CRYPTOKI_NOT_INITIALIZED, • CKR_DEVICE_ERROR, • CKR_DEVICE_MEMORY, • CKR_DEVICE_REMOVED, • CKR_FUNCTION_FAILED, • CKR_GENERAL_ERROR, • CKR_HOST_MEMORY, • CKR_SLOT_ID_INVALID, • CKR_TOKEN_NOT_PRESENT,

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
			• CKR_TOKEN_NOT_RECOGNIZED, • CKR_ARGUMENTS_BAD. • مقادیر بازگشتی برای فراخوانی موفق: • CKR_OK
	P 5.5.6	C_GetMechanismInfo	• این تابع اطلاعات مربوط به یک مکانیزم خاص را بدست می آورد. • اگر مکانیزم نامعتبری برای عملیات رمزنگاری مشخص شده باشد، کد خطای CKR_MECHANISM_INVALID برگشت داده می شود. اگر مکانیزم ناشناخته ای مشخص شده باشد یا مکانیزم مشخص شده در توکن انتخاب شده قابل استفاده نباشد، این کد خطا بازگشت داده می شود. • مقادیر بازگشتی برای فراخوانی ناموفق: • CKR_CRYPTOKI_NOT_INITIALIZED, • CKR_DEVICE_ERROR, • CKR_DEVICE_MEMORY, • CKR_DEVICE_REMOVED, • CKR_FUNCTION_FAILED, • CKR_GENERAL_ERROR, • CKR_HOST_MEMORY, • CKR_MECHANISM_INVALID, • CKR_SLOT_ID_INVALID, • CKR_TOKEN_NOT_PRESENT, • CKR_TOKEN_NOT_RECOGNIZED, • CKR_ARGUMENTS_BAD. • مقادیر بازگشتی برای فراخوانی موفق: • CKR_OK

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
	P 5.5.7	C_InitToken	• این تابع یک توکن را مقدار دهی اولیه می کند. • اگر Cryptoki تشخیص دهد که برنامه یک نشست باز با آن دارد، توکن نمی تواند مقداردهی اولیه شود. در چنین شرایطی فراخوانی تابع C_InitToken با خطای CKR_SESSION_EXISTS متوقف می شود. • اگر پرچم CKF_WRITE_PROTECTED از توکن با CK_TRUE مقداردهی شده باشد، آنگاه کد خطای CKR_TOKEN_WRITE_PROTECTED برگشت داده می شود. • مقادیر بازگشتی برای فراخوانی ناموفق: • CKR_CRYPTOKI_NOT_INITIALIZED, • CKR_DEVICE_ERROR, • CKR_DEVICE_MEMORY, • CKR_DEVICE_REMOVED, • CKR_FUNCTION_CANCELED, • CKR_FUNCTION_FAILED, • CKR_GENERAL_ERROR, • CKR_HOST_MEMORY, • CKR_PIN_INCORRECT, • CKR_PIN_LOCKED, • CKR_SESSION_EXISTS, • CKR_SLOT_ID_INVALID, • CKR_TOKEN_NOT_PRESENT, • CKR_TOKEN_NOT_RECOGNIZED, • CKR_TOKEN_WRITE_PROTECTED, • CKR_ARGUMENTS_BAD. • مقادیر بازگشتی برای فراخوانی موفق: • CKR_OK
	P 5.5.8	C_InitPIN	• این تابع پین کاربر عادی را مقداردهی اولیه می کند. • C_InitPIN را فقط می توان در حالت "R/W SO Functions" فراخوانی کرد، یعنی اجرای این تابع نیارمند ورود کاربر در حالت خواندنی/نوشتنی به توکن است. تلاش برای فراخوانی آن از یک نشست در هر حالت دیگر با خطای CKR_USER_NOT_LOGGED_IN اجرا نمی شود. • اگر پرچم CKF_WRITE_PROTECTED از توکن با CK_TRUE مقداردهی شده باشد، آنگاه کد خطای CKR_TOKEN_WRITE_PROTECTED برگشت داده می شود. • CKR_SESSION_READ_ONLY: این تابع در یک نشست فقط-خواندنی قابل اجرا نیست. در صورت فراخوانی این تابع در یک نشست فقط خواندنی، کد خطای CKR_SESSION_READ_ONLY برگشت داده می شود. این کد خطا نسبت به CKR_TOKEN_WRITE_PROTECTED دارای اولویت پایین تری

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
			است. • مقادیر بازگشتی برای فراخوانی ناموفق: • CKR_CRYPTOKI_NOT_INITIALIZED, • CKR_DEVICE_ERROR, • CKR_DEVICE_MEMORY, • CKR_DEVICE_REMOVED, • CKR_FUNCTION_CANCELED, • CKR_FUNCTION_FAILED, • CKR_GENERAL_ERROR, • CKR_HOST_MEMORY, • CKR_PIN_INVALID, • CKR_PIN_LEN_RANGE, • CKR_SESSION_CLOSED, • CKR_SESSION_READ_ONLY, • CKR_SESSION_HANDLE_INVALID, • CKR_TOKEN_WRITE_PROTECTED, • CKR_USER_NOT_LOGGED_IN, • CKR_ARGUMENTS_BAD. • مقادیر بازگشتی برای فراخوانی موفق: • CKR_OK
	P 5.5.9	**C_SetPIN**	• این تابع پین کاربر عادی را اصلاح می کند. • C_SetPIN را فقط می توان در حالت "R/W Public Session"، "R/W SO Functions" یا "R/W User Functions" فراخوانی کرد. تلاش برای فراخوانی آن از یک نشست در هر حالت دیگر با خطای CKR_SESSION_READ_ONLY انجام نمی شود. • C_InitPIN را فقط می توان در حالت "R/W SO Functions" فراخوانی کرد، یعنی اجرای این تابع نیارمند ورود کاربر در حالت خواندنی/نوشتنی به توکن است. تلاش برای فراخوانی آن از یک نشست در هر حالت دیگر با خطای CKR_USER_NOT_LOGGED_IN اجرا نمی شود. • اگر پرچم CKF_WRITE_PROTECTED از توکن با CK_TRUE مقداردهی شده باشد، آنگاه کد خطای CKR_TOKEN_WRITE_PROTECTED برگشت داده می شود. • مقادیر بازگشتی برای فراخوانی ناموفق: • CKR_CRYPTOKI_NOT_INITIALIZED, • CKR_DEVICE_ERROR, • CKR_DEVICE_MEMORY, • CKR_DEVICE_REMOVED, • CKR_FUNCTION_CANCELED, • CKR_FUNCTION_FAILED, • CKR_GENERAL_ERROR, • CKR_HOST_MEMORY,

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
			• CKR_PIN_INCORRECT, • CKR_PIN_INVALID, • CKR_PIN_LEN_RANGE, • CKR_PIN_LOCKED, • CKR_SESSION_CLOSED, • CKR_SESSION_HANDLE_INVALID, • CKR_SESSION_READ_ONLY, • CKR_TOKEN_WRITE_PROTECTED, • CKR_ARGUMENTS_BAD. • مقادیر بازگشتی برای فراخوانی موفق: • CKR_OK
توابع مدیریت نشست	P 5.6.1	C_OpenSession	• این تابع ارتباط بین برنامه و یک توکن خاص باز می کند یا پاسخ برنامه در مقابل درج توکن را تنظیم می کند. • این تابع دارای پرچم CKF_SERIAL_SESSION می باشد. هنگام فراخوانی تابع C_OpenSession، پرچم CKF_SERIAL_SESSION همیشه باید با مقدار CK_TRUE تنظیم شود، در غیر اینصورت، کد خطای CKR_SESSION_PARALLEL_NOT_SUPPORTED برگشت داده می شود. • ممکن است محدودیتی در تعداد نشست های همزمان یک برنامه با توکن وجود داشته باشد. در صورتی که به دلیل محدودیت تعداد زیاد نشست ها امکان باز کردن نشست جدید وجود نداشته باشد، کد خطای CKR_SESSION_COUNT برگشت داده می شود. • اگر توکن در حالت محافظت از نوشتن باشد (یعنی پرچم CKF_WRITE_PROTECTED با مقدار CK_TRUE تنظیم شده است)، نشست های فقط-خواندنی با آن می تواند باز شوند. در صورت فراخوانی نشست هایی غیر از فقط-خواندنی، کد خطای CKR_SESSION_READ_ONLY برگشت داده می شود. • اگر برنامه ای که C_OpenSession را فراخوانی می کند، از قبل دارای یک نشست R/W SO با توکن است، بنابراین هرگونه تلاش برای باز کردن یک نشست R/O با توکن با کد خطای CKR_SESSION_READ_WRITE_SO_EXISTS برگشت داده می شود. • مقادیر بازگشتی برای فراخوانی ناموفق: • CKR_CRYPTOKI_NOT_INITIALIZED, • CKR_DEVICE_ERROR, • CKR_DEVICE_MEMORY, • CKR_DEVICE_REMOVED, • CKR_FUNCTION_FAILED, • CKR_GENERAL_ERROR,

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
			• CKR_HOST_MEMORY, • CKR_SESSION_COUNT, • CKR_SESSION_PARALLEL_NOT_SUPPORTED, • CKR_SESSION_READ_WRITE_SO_EXISTS, • CKR_SLOT_ID_INVALID, • CKR_TOKEN_NOT_PRESENT, • CKR_TOKEN_NOT_RECOGNIZED, • CKR_TOKEN_WRITE_PROTECTED, • CKR_ARGUMENTS_BAD. • مقادیر بازگشتی برای فراخوانی موفق: • CKR_OK
	P 5.6.2	**C_CloseSession**	• این تابع یک نشست معین را می بندد. • هنگامی که یک نشست بسته می شود، تمام شیءهای ایجاد شده توسط آن نشست به طور خودکار از بین می روند، حتی اگر نشست های دیگری از آن شیءها استفاده می کنند. اگر یک نشست که قبلا بسته شده است، به این تابع داده شود، کد خطای CKR_SESSION_CLOSED را برگشت می دهد. • مقادیر بازگشتی برای فراخوانی ناموفق: • CKR_CRYPTOKI_NOT_INITIALIZED, • CKR_DEVICE_ERROR, • CKR_DEVICE_MEMORY, • CKR_DEVICE_REMOVED, • CKR_FUNCTION_FAILED, • CKR_GENERAL_ERROR, • CKR_HOST_MEMORY, • CKR_OK, CKR_SESSION_CLOSED, • CKR_SESSION_HANDLE_INVALID • مقادیر بازگشتی برای فراخوانی موفق: • CKR_OK

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
	P 5.6.3	C_CloseAllSessions	• این تابع تمامی نشست های توکن را می بندد. • مقادیر بازگشتی برای فراخوانی ناموفق: • CKR_CRYPTOKI_NOT_INITIALIZED, • CKR_DEVICE_ERROR, • CKR_DEVICE_MEMORY, • CKR_DEVICE_REMOVED, • CKR_FUNCTION_FAILED, • CKR_GENERAL_ERROR, • CKR_HOST_MEMORY, • CKR_SLOT_ID_INVALID, • CKR_TOKEN_NOT_PRESENT. • مقادیر بازگشتی برای فراخوانی موفق: • CKR_OK
	P 5.6.4	C_GetSessionInfo	• این تابع اطلاعات مربوط به یک نشست معین را بدست می آورد. • اگر شناسه نشست مشخص شده نادرست باشد، کد خطای CKR_SESSION_HANDLE_INVALID برگشت داده می شود. • اگر نشست قبلا بسته شده باشد، کد خطای CKR_SESSION_CLOSED برگشت داده می شود. • مقادیر بازگشتی برای فراخوانی ناموفق: • CKR_CRYPTOKI_NOT_INITIALIZED, • CKR_DEVICE_ERROR, • CKR_DEVICE_MEMORY, • CKR_DEVICE_REMOVED, • CKR_FUNCTION_FAILED, • CKR_GENERAL_ERROR, • CKR_HOST_MEMORY, • CKR_SESSION_CLOSED, • CKR_SESSION_HANDLE_INVALID, • CKR_ARGUMENTS_BAD. • مقادیر بازگشتی برای فراخوانی موفق: • CKR_OK
	P 5.6.5	C_SessionCancel	• این تابع عملیات مبتنی بر نشست فعال را لغو می کند.. • اگر امکان لغو کردن نشست از طرف توکن وجود نداشته باشد، کد خطای CKR_OPERATION_CANCEL_FAILED برگشب داده می شود. • مقدار بازگشتی برای فراخوانی موفق: • CKR_OK
	P 5.6.6	C_GetOperationState	• این تابع وضعیت عملیات رمزنگاری یک نشست را بدست می آورد. • تلاش برای ذخیره وضعیت عملیات رمزنگاری نشستی که دارای وضعیت عملیات رمزنگاری فعال (از قبیل، رمزگذاری، رمزگشایی، چکیده سازی، امضای بدون بازیابی پیام، راستی آزمایی بدون بازیابی پیام یا ترکیبی از این

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
			موارد) نیست. باید با خطای CKR_OPERATION_NOT_INITIALIZED برگشت داده شود. • تلاشی برای ذخیره وضعیت عملیات رمزنگاری نشستی که در حال انجام یک عملیات رمزنگاری مناسب است، اما به دلایل مختلف نمی تواند آنها را ذخیره کرد، باید با خطای CKR_STATE_UNSAVEABLE برگشت داده شود. • مقدار بازگشتی برای فراخوانی موفق: • CKR_OK
	P 5.6.7	**C_SetOperationState**	• حالت های نشست در کتابخانه Cryptoki عبارتند از: ۱. CKS_RO_PUBLIC_SESSION ۲. CKS_RO_USER_FUNCTIONS ۳. CKS_RW_PUBLIC_SESSION ۴. CKS_RW_USER_FUNCTIONS ۵. CKS_RW_SO_FUNCTIONS • با استفاده از این تابع، می توان حالت عملیات رمزنگاری بدست آمده در تابع C_GetOperationState، را بازیابی کرد. لزومی ندارد نشست مبدا (نشستی که حالت از آن بدست می آید) و نشست مقصد (نشستی که حالت در آن بازیابی می شود) یکسان باشد. بلکه این دو نشست باید متعلق به یک توکن بوده و دارای حالت نشست یکسان باشند. تابع C_SetOperationState در شرایط زیر خطای CKR_SAVED_STATE_INVALID را برمی گرداند: • نشست های مبدا و مقصد متعلق به یک توکن نباشند. • نشست های مبدا و مقصد دارای حالت نشست یکسان نباشند. • اگر حالت عملیات رمزنگاری مشخص شده دارای کلید رمزنگاری باشد (به عنوان مثال عملیات رمزگذاری و رمزگشایی و توابع MAC)، باید کلیدهای صحیح به عنوان ورودی به تابع داده شود: • در صورتی که برای در چنین عملیاتی، کلید رمزنگاری وارد نشود، کد خطای CKR_KEY_NEEDED برگشت داده می شود. • اگر کلید وارد شود ولی صحیح نباشد، کد خطای CKR_KEY_CHANGED برگشت داده می شود. • اگر حالت عملیات رمزنگاری مشخص شده نیازمند کلید رمزنگاری نباشد (به عنوان مثال عملیات چکیده سازی پیام، Hash)، نباید کلیدی به عنوان ورودی به تابع داده شود. در صورتی که برای چنین عملیاتی، کلید رمزنگاری وارد شود، کد خطای CKR_KEY_NOT_NEEDED برگشت داده می شود. • در بخش پرچم های توکن (CK_TOKEN_INFO)، پرچم CKF_RESTORE_KEY_NOT_NEEDED اگر با مقدار CK_TRUE

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
			تنظیم شود، تابع C_SetOperationState برای هیچ یک از عملیات رمزنگاری نیازمند ورود کلید نیست. اما اگر این پرچم با مقدار CK_FALS تنظیم شود، بسته به نوع عملیات رمزنگاری ممکن است کلید نیاز داشته باشد با نیاز نداشته باشد. • مقدار بازگشتی برای فراخوانی موفق: • CKR_OK
	P 5.6.8	**C_Login**	• این تابع برای ورود به توکن مورد استفاده قرار می گیرد. • در کتابخانه Cryptoki سه نوع کاربر توکن وجود دارد، (۱) CKU_SO، (۲) CKU_USER و (۳) CKU_CONTEXT_SPECIFIC. • در صورت فراخوانی موفق تابع C_Login برای کاربر نوع CKU_SO، نشست های برنامه وارد حالت "CKS_RW_SO_FUNCTIONS" می شوند. • در صورت فراخوانی موفق تابع C_Login برای کاربر نوع CKU_USER، نشست های برنامه وارد حالت "CKS_RW_USER_FUNCTIONS" یا "CKS_RO_USER_FUNCTIONS"می شوند. • اگر نوع کاربر CKU_CONTEXT_SPECIFIC باشد، رفتار تابع C_Login بستگی به زمینه ای دارد که در آن فراخوانی می شود. استفاده نادرست از این نوع کاربر منجر به بازگشت کد خطای CKR_OPERATION_NOT_INITIALIZED خواهد شد. • اگر توکن دارای "مسیر احراز هویت محافظت شده" باشد، یعنی پرچم CKF_PROTECTED_AUTHENTICATION_PATH در CK_TOKEN_INFO با مقدار CK_TRUE تنظیم شده تنظیم شده باشد. این بدان معناست که برای احراز هویت کاربر به توکن راهی غیر از ارسال پین از طریق کتابخانه Cryptoki وجود دارد. به عنوان مثال احراز هویت کاربر از طریق اثر انگشت با ورود پین از روی خود توکن سخت افزاری انجام می شود. برای ورود به توکن با مسیر احراز هویت محافظت شده، پارامتر pPin برای C_Login باید NULL_PTR باشد. • وقتی C_Login، به هر روش احراز هویتی که توسط توکن پشتیبانی می شود، انجام شود، مقدار برگشتی CKR_OK به این معنی است که کاربر با موفقیت احراز هویت شده است و مقدار برگشتی CKR_PIN_INCORRECT به معنای عدم موفقیت احراز هویت کاربر است.

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
			• اگر برنامه ای که C_Login را فراخوانی می کند، یک نشست باز R/O با توکن داشته باشد، دیگر نمی تواند به عنوان کاربر نوع CKU_SO در یک نشست ورود کند. تلاش برای انجام این کار، منجر به کد خطای CKR_SESSION_READ_ONLY_EXISTS می شود. • در صورتی که یک کلید دارای مشخصه CKA_ALWAYS_AUTHENTICATE تنظیم شده با CK_TRUE وجود داشته باشد، برای هر بار استفاده از این کلید در عملیات رمزنگاری، بدون اینکه تابع C_Logout جهت خروج از توکن فراخوانی شده باشد، نیاز به فراخوانی دوباره تابع C_Login است. به عبارت دیگر، برای هر باز استفاده از چنین کلیدی باید فرایند احراز هویت کاربر تکرار شود. • مقدار بازگشتی برای فراخوانی موفق: • CKR_OK
	P 5.6.9	C_LoginUser	• این تابع مشابه تابع C_Login است، با این تفاوت که علاوه در ورودی علاوه بر پین، یک نام کاربری نیز دریافت می کند. • مقدار بازگشتی برای فراخوانی موفق: • CKR_OK
	P 5.6.10	C_Logout	• این تابع عملیات خروج از توکن را انجام می دهد. • بسته به نوع کاربر فعلی، در صورت موفقیت آمیز بودن فراخوانی این تابع، نشست های برنامه وارد حالت CKS_RO_PUBLIC_SESSION یا حالت CKS_RW_PUBLIC_SESSION می شوند. • وقتی C_Logout با موفقیت اجرا می شود، شناسه شیءهای خصوصی نامعتبر می شوند (حتی اگر بعداً کاربر دوباره به توکن ورود کند، این شناسه ها نامعتبر هستند). علاوه بر این، تمام شیءهای خصوصی نشست از بین می روند. • مقدار بازگشتی برای فراخوانی موفق: • CKR_OK
توابع مدیریت شیء	P 5.7.1	C_CreateObject	• این تابع برای ایجاد یک شیء مورد استفاده قرار می گیرد. • مقدار بازگشتی به ازای مقدار ورودی معتبر برای یک ویژگی نامعتبر: • CKR_ATTRIBUTE_TYPE_INVALID • مقدار بازگشتی به ازای مقدار ورودی نامعتبر برای یک ویژگی معتبر: • CKR_ATTRIBUTE_VALUE_INVALID • مقدار بازگشتی برای ایجاد یک شیء با یک ویژگی فقط-خواندنی: • CKR_ATTRIBUTE_READ_ONLY • مقدار بازگشتی برای موقعی که مقادیر ویژگی ارائه شده به همراه مقادیر پیش فرض، برای ایجاد شیء کافی نباشد: • CKR_TEMPLATE_INCOMPLETE • مقدار بازگشتی برای موقعی که مقادیر ویژگی ارائه شده به همراه مقادیر پیش

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
			فرض ، برای ایجاد شیء متناقض باشند: • CKR_TEMPLATE_INCONSISTENT • مقدار بازگشتی برای موقعی که یک ویژگی معین با مقادیر مشابه بیش از یکبار فراخوانی شود: • CKR_TEMPLATE_INCONSISTENT, • در صورتی که برای یک شیء، پرچم CKF_WRITE_PROTECTED با مقدار CK_TRUE تنظیم شده باشد، اعمال این تابع بر روی چنین شیئی با کد خطای CKR_TOKEN_WRITE_PROTECTED مواجه خواهد شد. • در یک نشست فقط-خواندنی، فقط شیءهای نشست را می توان تولید کرد. بنابراین هر تلاشی برای ایجاد سایر شی‌ها با کد خطای CKR_SESSION_READ_ONLY مواجه می شود. • در نشست عمومی فقط می توان شیءهای عمومی را ایجاد کرد. برای ایجاد شیءهای نشست های کاربر، نیاز به ورود کاربر است. در صورت تلاش برای ایجاد شیءهای نشست های کاربر در نشست عمومی کد خطای CKR_USER_NOT_LOGGED_IN برگشت داده می شود. • مقدار بازگشتی برای فراخوانی موفق: • CKR_OK
	P 5.7.2	**C_CopyObject**	• این تابع بک کپی از شیء ایجاد می کند. • مقدار بازگشتی به ازای مقدار ورودی معتبر برای یک ویژگی نامعتبر: • CKR_ATTRIBUTE_TYPE_INVALID • مقدار بازگشتی به ازای مقدار ورودی نامعتبر برای یک ویژگی معتبر: • CKR_ATTRIBUTE_VALUE_INVALID • مقدار بازگشتی برای موقعی که مقادیر ویژگی ارائه شده به همراه مقادیر پیش فرض، برای ایجاد شیء کافی نباشد: • CKR_TEMPLATE_INCOMPLETE • مقدار بازگشتی برای موقعی که مقادیر ویژگی ارائه شده به همراه مقادیر پیش فرض، برای ایجاد شیء متناقض باشند: • CKR_TEMPLATE_INCONSISTENT • مقدار بازگشتی برای موقعی که یک ویژگی معین با مقادیر مشابه بیش از یکبار فراخوانی شود: • CKR_TEMPLATE_INCONSISTENT, • در یک نشست فقط-خواندنی، فقط شیءهای نشست را می توان تولید کرد. بنابراین هر تلاشی برای ایجاد سایر شی‌ها با کد خطای CKR_SESSION_READ_ONLY مواجه می شود. • در نشست عمومی فقط می توان شیءهای عمومی را ایجاد کرد. برای ایجاد شیءهای نشست های کاربر، نیاز به ورود کاربر است. در صورت تلاش برای ایجاد شیءهای نشست های کاربر در نشست عمومی کد خطای

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
			CKR_USER_NOT_LOGGED_IN برگشت داده می شود. • در صورتی که برای یک شیء، پرچم CKF_WRITE_PROTECTED با مقدار CK_TRUE تنظیم شده باشد، اعمال این تابع بر روی چنین شیئی با کد خطای CKR_TOKEN_WRITE_PROTECTED مواجه خواهد شد. • در این تابع، تمامی ویژگی های قابل تنظیم را در شی کپی می توان تنظیم کرد. علاوه بر موارد قابل تنظیم، ویژگی های خاص CKA_TOKEN، CKA_PRIVATE، CKA_MODIFIABLE و CKA_DESTROYABLE در فراخوانی این تابع قابل تغییر هستند. • برای کپی یک شیء کلید محرمانه، ویژگی CKA_EXTRACTABLE را از مقدار CK_TRUE به CK_FALSE می توان تغییر داد، ولی برعکس آن امکان پذیر نیست. در صورت تلاش برای تغییر از CK_FALSE به CK_TRUE، کد خطای CKR_ATTRIBUTE_READ_ONLY برگشت داده می شود. • برای کپی یک شیء کلید محرمانه، می توان ویژگی CKA_SENSITIVE را از CK_FALSE به CK_TRUE تغییر داد، ولی برعکس آن امکان پذیر نیست. در صورت تلاش برای تغییر از CK_TRUE به CK_FALSE، کد خطای CKR_ATTRIBUTE_READ_ONLY برگشت داده می شود. • در صورتی که ویژگی CKA_COPYABLE با مقدار CK_FALSE تنظیم شده باشد، تابع مقدار زیر را برگشت می دهد: • CKR_ACTION_PROHIBITED, • در صورتی که برای یک شیء، پرچم CKF_WRITE_PROTECTED با مقدار CK_TRUE تنظیم شده باشد، اعمال این تابع بر روی چنین شیئی با کد خطای CKR_TOKEN_WRITE_PROTECTED مواجه خواهد شد. • مقدار بازگشتی برای فراخوانی موفق: • CKR_OK
	P 5.7.3	**C_DestroyObject**	• این تابع برای از بین بردن شیء مورد استفاده قرار می گیرد. • در یک نشست فقط-خواندنی، فقط شیءهای نشست را می توان از بین برد. بنابراین هر تلاشی برای از بین بردن سایر شیءها با کد خطای CKR_SESSION_READ_ONLY مواجه می شود. • در نشست عمومی فقط می توان شیءهای عمومی را از بین برد. برای از بین بردن شیءهای نشست های کاربر نیاز به ورود کاربر است. • در صورتی که برای یک شیء، پرچم CKF_WRITE_PROTECTED با مقدار CK_TRUE تنظیم شده باشد، اعمال این تابع بر روی چنین شیئی با کد خطای CKR_TOKEN_WRITE_PROTECTED مواجه خواهد شد. • اگر برای یک شیء، ویژگی CKA_DESTROYABLE با مقدار

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
			CK_TRUE تنظیم شده باشد، قابل از بین بردن نیست. در صورت فراخوانی این تابع بر روی چنین شیئی، مقدار بازگشتی برابر است با: • CKR_ACTION_PROHIBITED,
	P 5.7.4	C_GetObjectSize	• این تابع اندازه یک شیء را به بایت بدست می آورد. • اگر ویژگی CKA_SENSITIVE با مقدار CK_TRUE، یا ویژگی CKA_EXTRACTABLE با مقدار CK_FALSE تنظیم شود، در اینصورت امکان آشکار شدن اندازه شیء وجود ندارد. در صورت تلاش برای بدست آوردن این مقدار، کد خطای CKR_INFORMATION_SENSITIVE برگشت داده می شود.
	P 5.7.5	C_GetAttributeValue	• این تابع مقدار ویژگی یک شیء را بدست می آورد. • اگر ویژگی CKA_SENSITIVE با مقدار CK_TRUE، یا ویژگی CKA_EXTRACTABLE با مقدار CK_FALSE تنظیم شود، در اینصورت امکان آشکار شده مقدار ویژگی وجود ندارد. در صورت تلاش برای بدست آوردن مقدار چنین ویژگیی، کد خطای زیر برگشت داده می شود: • CKR_ATTRIBUTE_SENSITIVE • مقدار بازگشتی در صورتی که ویژگی مشخص شده برای شی نامعتبر باشد (شیء چنین ویژگیی ندارد): • CKR_ATTRIBUTE_TYPE_INVALID, • مقدار بازگشتی در صورتی که اندازه خروجی از اندازه بافر بزرگتر باشد: • CKR_BUFFER_TOO_SMALL,
	P 5.7.6	C_SetAttributeValue	این تابع مقدار ویژگی یک شیء را تنظیم می کند. مقدار بازگشتی به ازای مقدار ورودی معتبر برای یک ویژگی نامعتبر: • CKR_ATTRIBUTE_TYPE_INVALID مقدار بازگشتی به ازای مقدار ورودی نامعتبر برای یک ویژگی معتبر: • CKR_ATTRIBUTE_VALUE_INVALID مقدار بازگشتی برای یک ویژگی فقط-خواندنی: • CKR_ATTRIBUTE_READ_ONLY مقدار بازگشتی برای موقعی که مقادیر ویژگی ارائه شده به همراه مقادیر پیش فرض ویژگی و هر مقداری از ویژگی که توسط خود تابع تولید می شود، برای تنظیم شیء متناقض باشند: • CKR_TEMPLATE_INCONSISTENT مقدار بازگشتی برای موقعی که یک ویژگی معین با مقادیر مشابه بیش از یکبار

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
			فراخوانی شود: • CKR_TEMPLATE_INCONSISTENT, در صورتی که ویژگی CKA_SENSITIVE با مقدار CK_TRUE مقدار دهی شود، شیء غیرقابل تغییر می شود. در صورت فراخوانی این تابع بر روی چنین شیئی، مقدار بازگشتی برابر است با: • CKR_ACTION_PROHIBITED, اگر مقدار تعیین شده برای ویژگی مورد نظر با سایر ویژگیهای شیء ناسازگار باشد، مقدار زیر برگشت داده می شود: • CKR_TEMPLATE_INCONSISTENT در یک نشست فقط-خواندنی، فقط شیءهای نشست را می توان اصلاح کرد. بنابراین هر تلاشی برای اصلاح سایر شیءها با کد خطای CKR_SESSION_READ_ONLY مواجه می شود. در نشست عمومی فقط می توان شیءهای عمومی را اصلاح کرد. برای اصلاح شیءهای نشست های کاربر، نیاز به ورود کاربر است. در صورت تلاش برای اصلاح شیءهای نشست های کاربر در نشست عمومی کد خطای CKR_USER_NOT_LOGGED_IN برگشت داده می شود. در صورتی که برای یک شیء، پرچم CKF_WRITE_PROTECTED با مقدار CK_TRUE تنظیم شده باشد، اعمال این تابع بر روی چنین شیئی با کد خطای CKR_TOKEN_WRITE_PROTECTED مواجه خواهد شد.
	P 5.7.7	**C_FindObjectsInit**	• عملیات جستجوی اشیا را مقدار دهی اولیه می کند • حداکثر یک عملیات جستجو ممکن است در یک زمان مشخص در یک نشست معین فعال باشد. در صورت فراخوانی بیش از یک عملیات جستجو، کد خطای زیر برگشت داده می شود: • CKR_OPERATION_ACTIVE, • مقدار بازگشتی به ازای مقدار ورودی معتبر برای یک ویژگی نامعتبر: • CKR_ATTRIBUTE_TYPE_INVALID • مقدار بازگشتی به ازای مقدار ورودی نامعتبر برای یک ویژگی معتبر: • CKR_ATTRIBUTE_VALUE_INVALID
	P 5.7.8	**C_FindObjects**	این تابع عملیات جستجوی شیء را انجام می دهد. قبل از از این تابع، ابتدا باید تابع C_FindObjectsInit فراخوانی شود. در غیر اینصورت، کد خطای زیر برگشت داده می شود: • CKR_OPERATION_NOT_INITIALIZED,

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
	P 5.7.9	C_FindObjectsFinal	این تابع یک عملیات جستجوی شیء را به پایان می رساند. قبل از این تابع، ابتدا باید تابع های C_FindObjectsInit و C_FindObjects فراخوانی شود. در غیر اینصورت، کد خطای زیر برگشت داده می شود: • CKR_OPERATION_NOT_INITIALIZED, مقادیر بازگشتی برای فراخوانی ناموفق: • CKR_CRYPTOKI_NOT_INITIALIZED, • CKR_DEVICE_ERROR, • CKR_DEVICE_MEMORY, • CKR_DEVICE_REMOVED, • CKR_FUNCTION_FAILED, • CKR_GENERAL_ERROR, • CKR_HOST_MEMORY, • CKR_OPERATION_NOT_INITIALIZED, • CKR_SESSION_CLOSED, • CKR_SESSION_HANDLE_INVALID مقادیر بازگشتی برای فراخوانی موفق: • CKR_OK
توابع رمزگذاری	P 5.8.1	C_EncryptInit	• این تابع عملیات رمزگذاری را مقدار دهی اولیه می کند. • پس از فراخوانی این تابع، برای رمزگذاری یکپارچه داده تابع C_Encrypt و برای رمزگذاری چند بخشی داده تابع C_EncryptUpdate به تعداد صفر یا چند بار و تابع C_EncryptFinal یک بار فراخوانی می شود. با فراخوانی تابع C_EncryptInit عملیات رمزگذاری فعال می شود تا زمانی که فراخوانی موفق C_Encrypt یا C_EncryptFinal رخ دهد. • یکی از ورودی های این تابع، کلید رمزگذاری است. ویژگی CKA_ENCRYPT برای این کلید، باید CK_TRUE باشد (که نشان می دهد این کلید از عملیات رمزگذاری پشتیبانی می کند)، در غیر اینصورت، کد خطای CKR_KEY_FUNCTION_NOT_PERMITTED برگشت داده می شود. • اگر شناسه کلید مشخص شده معتبر نباشد، کد خطای CKR_KEY_HANDLE_INVALID برگشت داده می شود. • اگر اندازه کلید ارائه شده خارج از محدوده مجاز باشد، کد خطای CKR_KEY_SIZE_RANGE برگشت داده می شود. • اگر نوع کلید ورودی برای استفاده در مکانیزم تعیین شده همخوانی نداشته باشد، کد خطای CKR_KEY_TYPE_INCONSISTENT برگشت داده می شود. • اگر مکانیزم ناشناخته ای مشخص شده باشد یا مکانیزم مشخص شده در توکن انتخاب شده قابل استفاده نباشد، کد خطای

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
			CKR_MECHANISM_INVALID بازگشت داده می شود. • اگر پارامترهای نامعتبر به مکانیزم مشخص شده برای عملیات رمزنگاری ارائه شود، کد خطای CKR_MECHANISM_PARAM_INVALID برگشت داده می شود. • وقتی که یک عملیات جستجوی شیء فعال وجود داشته باشد، باید از فعال کردن رمزگذاری با تابع C_EncryptInit جلوگیری شده و کد خطای CKR_OPERATION_ACTIVE برگشت داده شود. • برای خاتمه دادن به یک عملیات رمزگذاری فعال، تابع C_EncryptInit را می توان با pMechanism تنظیم شده روی NULL_PTR فراخوانی کرد. اگر عملیات رمزگذاری لغو نشود ، کد خطای CKR_OPERATION_CANCEL_FAILED باید برگردانده شود.
	P 5.8.2	**C_Encrypt**	• این تابع عملیات رمزگذاری تک بخشی داده را انجام می دهد. • قبل از فراخوانی این تابع، باید تابع C_EncryptInit فراخوانی شده باشد، در غیر ایصورت کد خطای CKR_OPERATION_NOT_INITIALIZED برگشت داده می شود. • CKR_DATA_LEN_RANGE: اگر طول داده های ورودی متن اصلی برای این تابع نادرست باشد. بسته به مکانیزم رمزنگاری، این کد خطا می تواند به این معنی باشد که داده های متن اصلی خیلی کوتاه یا خیلی بلند است یا مضربی از طول بلوک معین نیست. • CKR_DATA_INVALID: اگر داده های ورودی متن اصلی به تابع نامعتبر باشد این کد خطا برگشت داده می شود. این مقدار بازگشتی نسبت به CKR_DATA_LEN_RANGE دارای اولویت پایین تری است. • در صورتی که اندازه متن رمزی خروجی از اندازه بافر تعیین شده بزرگتر باشد، کد خطای CKR_BUFFER_TOO_SMALL برگشت داده می شود.
	P 5.8.3	**C_EncryptUpdate**	• این تابع عملیات رمزگذاری چند بخشی داده را انجام می دهد. • برای رمزگذاری چند بخشی یک داده، تابع C_EncryptUpdate به تعداد صفر یا چند بار فراخوانی شده و در پایان تابع C_EncryptFinal برای پایان دادن به فرایند رمزگذاری تمام بخشها فراخوانی می شود. • قبل از فراخوانی این تابع، باید تابع C_EncryptInit فراخوانی شده باشد، در غیر ایصورت کد خطای CKR_OPERATION_NOT_INITIALIZED برگشت داده می شود. • CKR_DATA_LEN_RANGE: اگر طول داده های ورودی متن اصلی برای این تابع نادرست باشد. بسته به مکانیزم رمزنگاری، این کد خطا می تواند به این معنی باشد که داده های متن اصلی خیلی کوتاه یا خیلی بلند

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
			است یا مضربی از طول بلوک معین نیست. • CKR_DATA_INVALID: اگر داده های ورودی متن اصلی به تابع نامعتبر باشد این کد خطا برگشت داده می شود. این مقدار بازگشتی نسبت به CKR_DATA_LEN_RANGE دارای اولویت پایین تری است. • در صورتی که اندازه متن رمزی خروجی از اندازه بافر تعیین شده بزرگتر باشد، کد خطای CKR_BUFFER_TOO_SMALL برگشت داده می شود.
	P 5.8.4	**C_EncryptFinal**	• این تابع عملیات رمزگذاری چند بخشی داده را خاتمه می دهد. • قبل از فراخوانی این تابع، باید تابع C_EncryptInit فراخوانی شده باشد، در غیر ایصورت کد خطای CKR_OPERATION_NOT_INITIALIZED برگشت داده می شود. • در صورتی که اندازه متن رمزی خروجی از اندازه بافر تعیین شده بزرگتر باشد، کد خطای CKR_BUFFER_TOO_SMALL برگشت داده می شود. • CKR_DATA_LEN_RANGE: اگر طول داده های ورودی متن اصلی برای این تابع نادرست باشد. بسته به مکانیزم رمزنگاری، این کد خطا می تواند به این معنی باشد که داده های متن اصلی خیلی کوتاه یا خیلی بلند است یا مضربی از طول بلوک معین نیست.
توابع رمزگذاری مبتنی بر پیام	**P 5.9.1**	**C_MessageEncryptInit**	رمزگذاری مبتنی بر پیام به فرآیند رمزگذاری چندین پیام با استفاده از مکانیسم رمزگذاری و کلید رمزگذاری یکسان اشاره دارد. مکانیسم رمزگذاری می تواند یک الگوریتم رمزگذاری تصدیق شده با داده های مرتبط[1] (AEAD) یا یک الگوریتم رمزگذاری محض باشد. • این تابع عملیات رمزگذاری مبتنی بر پیام را مقدار دهی اولیه می کند. • یکی از ورودی های این تابع، کلید رمزگذاری است. ویژگی CKA_ENCRYPT برای این کلید، باید CK_TRUE باشد (که نشان می دهد این کلید از عملیات رمزگذاری پشتیبانی می کند)، در غیر اینصورت، کد خطای CKR_KEY_FUNCTION_NOT_PERMITTED برگشت داده می شود. • اگر شناسه کلید مشخص شده معتبر نباشد، کد خطای CKR_KEY_HANDLE_INVALID برگشت داده می شود. • اگر اندازه کلید ارائه شده خارج از محدوده مجاز باشد، کد خطای CKR_KEY_SIZE_RANGE برگشت داده می شود. • اگر نوع کلید ورودی برای استفاده در مکانیزم تعیین شده همخوانی نداشته باشد، کد خطای CKR_KEY_TYPE_INCONSISTENT برگشت داده می شود.

[1] Authenticated Encryption with Associated Data (AEAD)

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
			• اگر مکانیزم ناشناخته ای مشخص شده باشد یا مکانیزم مشخص شده در توکن انتخاب شده قابل استفاده نباشد، کد خطای CKR_MECHANISM_INVALID بازگشت داده می شود. • اگر پارامترهای نامعتبر به مکانیزم مشخص شده برای عملیات رمزنگاری ارائه شود، کد خطای CKR_MECHANISM_PARAM_INVALID برگشت داده می شود. • وقتی که یک عملیات جستجوی شیء فعال وجود داشته باشد، باید از فعال کردن رمزگذاری با تابع C_MessageEncryptInit جلوگیری شده و کد خطای CKR_OPERATION_ACTIVE برگشت داده شود. • برای خاتمه دادن به یک عملیات رمزگذاری فعال، تابع C_MessageEncryptInit را می توان با pMechanism تنظیم شده روی NULL_PTR فراخوانی کرد. اگر عملیات رمزگذاری لغو نشود ، کد خطای CKR_OPERATION_CANCEL_FAILED باید برگردانده شود.
	P 5.9.2	**C_EncryptMessage**	• عملیات رمزگذاری مبتنی بر پیام تک بخشی داده را انجام می دهد. • این تابع هم کارکرد تابع C_Encryp را برای رمزگذاری داده دارد و هم کارکرد رمزگذاری تصدیق شده با داده مرتبط (AEAD) را دارا می باشد. اگر ورودی های pAssociatedData و ulAssociatedDataLen با مقادیر (NULL, 0) تعیین شوند، آنگاه تابع C_EncryptMessage دارای کارکرد رمزگذاری محض است. اما اگر این مقادیر ورودی غیرتهی باشند، آنگاه تابع دارای کارکرد AEAD است. • قبل از فراخوانی این تابع، باید تابع C_MessageEncryptInit فراخوانی شده باشد، در غیر ایصورت کد خطای CKR_OPERATION_NOT_INITIALIZED برگشت داده می شود. • CKR_DATA_LEN_RANGE: اگر طول داده های ورودی متن اصلی برای این تابع نادرست باشد. بسته به مکانیزم رمزنگاری، این کد خطا می تواند به این معنی باشد که داده های متن اصلی خیلی کوتاه یا خیلی بلند است یا مضربی از طول بلوک معین نیست. • CKR_DATA_INVALID: اگر داده های ورودی متن اصلی به تابع نامعتبر باشد این کد خطا برگشت داده می شود. این مقدار بازگشتی نسبت به CKR_DATA_LEN_RANGE دارای اولویت پایین تری است. • در صورتی که اندازه متن رمزی خروجی از اندازه بافر تعیین شده بزرگتر باشد، کد خطای CKR_BUFFER_TOO_SMALL برگشت داده می شود.

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
	P 5.9.3	C_EncryptMessageBegin	• تابع C_EncryptMessageBegin به همراه تابع C_EncryptMessageNext نقش تابع C_EncryptMessage را ایفا می کنند. به این معنی که تابع C_EncryptMessage عملیات را بر روی داده یکپارچه انجام می دهد، اما اگر نیاز باشد داده به چند بخش تبدیل شده و روی آنها عملیات انجام شود، از توابع C_EncryptMessageBegin به همراه تابع C_EncryptMessageNext استفاده می شود. • پس از فراخوانی تابع C_EncryptMessageBegin، باید تابع C_EncryptMessageNext یک یا چند مرتبه فراخوانی شود. عملیات رمزگذاری مبتنی بر پیام فعال خواهد بود تا زمانی که تابع C_EncryptMessageNext با پرچم CKF_END_OF_MESSAGE فراخوانی شود، در اینصورت عملیات با ارائه متن رمزی پایان می پذیرد. پس از پایان عملیات فعال، دوباره می توان یک عملیات جدید را با فراخوانی تابع C_EncryptMessage یا C_EncryptMessageBegin آغاز نمود. • قبل از فراخوانی این تابع، باید تابع C_MessageEncryptInit فراخوانی شده باشد، در غیر ایصورت کد خطای CKR_OPERATION_NOT_INITIALIZED برگشت داده می شود. • در صورتی که اندازه متن رمزی خروجی از اندازه بافر تعیین شده بزرگتر باشد، کد خطای CKR_BUFFER_TOO_SMALL برگشت داده می شود.
	P 5.9.4	C_EncryptMessageNext	• این تابع پس از تابع C_EncryptMessageBegin به تعداد یک یا چند مرتبه قابل فراخوانی است. تابع C_EncryptMessageNext دارای یک پرچم است که اگر با مقدار 0 تنظیم شود به این معناست که عملیات ادامه دارد، اما اگر با مقدار CKF_END_OF_MESSAGE تنظیم شود به این معناست که عملیات بر روی بخش پایانی داده است. • عملیات رمزگذاری چند بخشی فعال خواهد بود تا زمانی که تابع C_EncryptMessageNext با پرچم CKF_END_OF_MESSAGE فراخوانی شود، در اینصورت عملیات با ارائه متن رمزی پایان می پذیرد. پس از پایان عملیات فعال، دوباره می توان یک عملیات جدید را با فراخوانی تابع C_EncryptMessage یا C_EncryptMessageBegin آغاز نمود. • قبل از فراخوانی این تابع، باید تابع C_MessageEncryptInit فراخوانی شده باشد، در غیر ایصورت کد خطای CKR_OPERATION_NOT_INITIALIZED برگشت داده می شود. • در صورتی که اندازه متن رمزی خروجی از اندازه بافر تعیین شده بزرگتر باشد، کد خطای CKR_BUFFER_TOO_SMALL برگشت داده می شود. • CKR_DATA_LEN_RANGE: اگر طول داده های ورودی متن اصلی

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
			برای این تابع نادرست باشد. بسته به مکانیزم رمزنگاری، این کد خطا می تواند به این معنی باشد که داده های متن اصلی خیلی کوتاه با خیلی بلند است یا مضربی از طول بلوک معین نیست.
	P 5.9.5	**C_MessageEncryptFinal**	• این تابع عملیات رمزگذاری فعال شده با C_MessageEncryptInit را به پایان می رساند. اگر قبل از فراخوانی این تابع، تابع C_MessageEncryptInit فراخوانی نشده باشد، کد خطای CKR_OPERATION_NOT_INITIALIZED برگشت داده می شود.
توابع رمزگشایی	**P5.10.1**	**C_DecryptInit**	• پس از فراخوانی این تابع، برای رمزگشایی یکپارچه داده تابع C_Decrypt و برای رمزگشایی چند بخشی داده تابع C_DecryptUpdate به تعداد صفر یا چند بار و تابع C_DecryptFinal یک بار فراخوانی می شود. با فراخوانی تابع C_DecryptInit عملیات رمزگشایی فعال می شود تا زمانی که فراخوانی موفق C_Decrypt یا C_DecryptFinal رخ دهد. • یکی از ورودی های این تابع، کلید رمزگشایی است. ویژگی CKA_DECRYPT برای این کلید، باید CK_TRUE باشد (که نشان می دهد این کلید از عملیات رمزگشایی پشتیبانی می کند)، در غیر اینصورت، کد خطای CKR_KEY_FUNCTION_NOT_PERMITTED برگشت داده می شود. • اگر شناسه کلید مشخص شده معتبر نباشد، کد خطای CKR_KEY_HANDLE_INVALID برگشت داده می شود. • اگر اندازه کلید ارائه شده خارج از محدوده مجاز باشد، کد خطای CKR_KEY_SIZE_RANGE برگشت داده می شود. • اگر نوع کلید ورودی برای استفاده در مکانیزم تعیین شده همخوانی نداشته باشد، کد خطای CKR_KEY_TYPE_INCONSISTENT برگشت داده می شود. • اگر مکانیزم ناشناخته ای مشخص شده باشد یا مکانیزم مشخص شده در توکن انتخاب شده قابل استفاده نباشد، کد خطای CKR_MECHANISM_INVALID بازگشت داده می شود. • اگر پارامترهای نامعتبر به مکانیزم مشخص شده برای عملیات رمزنگاری ارائه شود، کد خطای CKR_MECHANISM_PARAM_INVALID برگشت داده می شود. • وقتی که یک عملیات جستجوی شیء فعال وجود داشته باشد، باید از فعال کردن رمزگذاری با تابع C_DecryptInit جلوگیری شده و کد خطای CKR_OPERATION_ACTIVE برگشت داده شود. • برای خاتمه دادن به یک عملیات رمزگشایی فعال، تابع C_DecryptInit را

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
			می توان با pMechanism تنظیم شده روی NULL_PTR فراخوانی کرد. اگر عملیات رمزگشایی لغو نشود ، کد خطای CKR_OPERATION_CANCEL_FAILED باید برگردانده شود.
	P **5.10.2**	**C_Decrypt**	• این تابع عملیات رمزگشایی داده یکپارچه را انجام می دهد. • قبل از فراخوانی این تابع، باید تابع C_DecryptInit فراخوانی شده باشد، در غیر ایصورت کد خطای CKR_OPERATION_NOT_INITIALIZED برگشت داده می شود. • CKR_ENCRYPTED_DATA_LEN_RANGE: طول داده های ورودی متن رمزی برای این تابع نادرست باشد. بسته به مکانیزم رمزنگاری، این کد خطا می تواند به این معنی باشد که داده های متن رمزی خیلی کوتاه با خیلی بلند است یا مضربی از طول بلوک معین نیست. • CKR_ENCRYPTED_DATA_INVALID: اگر داده های ورودی متن رمزی به تابع نامعتبر باشد این کد خطا برگشت داده می شود. این مقدار بازگشتی نسبت به CKR_ENCRYPTED_DATA_LEN_RANGE دارای اولویت پایین تری است. • در صورتی که اندازه متن رمزگشایی شده خروجی از اندازه بافر تعیین شده بزرگتر باشد، کد خطای CKR_BUFFER_TOO_SMALL برگشت داده می شود.
	P **5.10.3**	**C_DecryptUpdate**	• این تابع به همراه تابع C_DecryptFinal برای رمزگشایی یک داده طی چند بخش مورد استفاده قرار می گیرد. • قبل از فراخوانی این تابع، باید تابع C_DecryptInit فراخوانی شده باشد، در غیر ایصورت کد خطای CKR_OPERATION_NOT_INITIALIZED برگشت داده می شود. • CKR_ENCRYPTED_DATA_LEN_RANGE: طول داده های ورودی متن رمزی برای این تابع نادرست باشد. بسته به مکانیزم رمزنگاری، این کد خطا می تواند به این معنی باشد که داده های متن رمزی خیلی کوتاه با خیلی بلند است یا مضربی از طول بلوک معین نیست. • CKR_ENCRYPTED_DATA_INVALID: اگر داده های ورودی متن رمزی به تابع نامعتبر باشد این کد خطا برگشت داده می شود. این مقدار بازگشتی نسبت به CKR_ENCRYPTED_DATA_LEN_RANGE دارای اولویت پایین تری است. • در صورتی که اندازه متن رمزگشایی شده خروجی از اندازه بافر تعیین شده بزرگتر باشد، کد خطای CKR_BUFFER_TOO_SMALL برگشت داده می شود.

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
	P **5.10.4**	**C_DecryptFinal**	• این تابع عملیات رمزگشایی چند بخشی داده را به پایان می رساند. • قبل از فراخوانی این تابع، باید تابع C_DecryptInit فراخوانی شده باشد، در غیر ایصورت کد خطای CKR_OPERATION_NOT_INITIALIZED برگشت داده می شود. • در صورتی که اندازه متن رمزگشایی شده خروجی از اندازه بافر تعیین شده بزرگتر باشد، کد خطای CKR_BUFFER_TOO_SMALL برگشت داده می شود. • CKR_ENCRYPTED_DATA_LEN_RANGE: طول داده های ورودی متن رمزی برای این تابع نادرست باشد. بسته به مکانیزم رمزنگاری، این کد خطا می تواند به این معنی باشد که داده های متن رمزی خیلی کوتاه با خیلی بلند است یا مضربی از طول بلوک معین نیست. • CKR_ENCRYPTED_DATA_INVALID: اگر داده های ورودی متن رمزی به تابع نامعتبر باشد این کد خطا برگشت داده می شود. این مقدار بازگشتی نسبت به CKR_ENCRYPTED_DATA_LEN_RANGE دارای اولویت پایین تری است.

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
توابع رمزگشایی مبتنی بر پیام	P 5.11.1	C_MessageDecryptInit	• این تابع عملیات رمزگشایی مبتنی بر پیام داده را مقدار دهی اولیه می کند. • پس از فراخوانی این تابع، برای رمزگشایی مبتنی بر پیام یکپارچه داده تابع C_MessageDecrypt و برای رمزگشایی مبتنی بر پیام چند بخشی داده تابع C_MessageDecryptBegin و تابع C_MessageDecryptNext یک یا چند بار فراخوانی می شود. با فراخوانی تابع C_MessageDecryptInit عملیات رمزگشایی مبتنی بر پیام فعال می شود تا زمانی که فراخوانی موفق تابع C_MessageDecryptFinal رخ دهد. • یکی از ورودی های این تابع، کلید رمزگشایی است. ویژگی CKA_DECRYPT برای این کلید، باید CK_TRUE باشد (که نشان می دهد این کلید از عملیات رمزگشایی پشتیبانی می کند)، در غیر اینصورت، کد خطای CKR_KEY_FUNCTION_NOT_PERMITTED برگشت داده می شود. • اگر شناسه کلید مشخص شده معتبر نباشد، کد خطای CKR_KEY_HANDLE_INVALID برگشت داده می شود. • اگر اندازه کلید ارائه شده خارج از محدوده مجاز باشد، کد خطای CKR_KEY_SIZE_RANGE برگشت داده می شود. • اگر نوع کلید ورودی برای استفاده در مکانیزم تعیین شده همخوانی نداشته باشد، کد خطای CKR_KEY_TYPE_INCONSISTENT برگشت داده می شود. • اگر مکانیزم ناشناخته ای مشخص شده باشد یا مکانیزم مشخص شده در توکن انتخاب شده قابل استفاده نباشد، کد خطای CKR_MECHANISM_INVALID بازگشت داده می شود. • اگر پارامترهای نامعتبر به مکانیزم مشخص شده برای عملیات رمزنگاری ارائه شود، کد خطای CKR_MECHANISM_PARAM_INVALID برگشت داده می شود. • وقتی که یک عملیات جستجوی شیء فعال وجود داشته باشد، باید از فعال کردن رمزگشایی با تابع C_MessageDecryptInit جلوگیری شده و کد خطای CKR_OPERATION_ACTIVE برگشت داده شود. • برای خاتمه دادن به یک عملیات رمزگشایی فعال، تابع C_MessageDecryptInit را می توان با pMechanism تنظیم شده روی NULL_PTR فراخوانی کرد. اگر عملیات رمزگشایی لغو نشود ، کد خطای CKR_OPERATION_CANCEL_FAILED باید برگردانده شود.

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
توابع رمزگشایی مبتنی بر پیام	P 5.11.2	C_DecryptMessage	• این تابع هم کارکرد تابع C_Decryp را برای رمزگشایی داده دارد و هم کارکرد رمزنگاری تصدیق شده با داده مرتبط (AEAD) را دارا می باشد. اگر ورودی های pAssociatedData و ulAssociatedDataLen با مقادیر (NULL, 0) تعیین شوند، آنگاه تابع C_DecryptMessage دارای کارکرد رمزگشایی محض مشابه C_Decryp است. اما اگر این مقادیر ورودی غیرتهی باشند، آنگاه تابع دارای کارکرد AEAD است. • تابع C_DecryptMessage هم آغازگر عملیات رمزگشایی و هم خاتمه دهنده آن است. • قبل از فراخوانی این تابع، باید تابع C_MessageDecryptInit فراخوانی شده باشد، در غیر ایصورت کد خطای CKR_OPERATION_NOT_INITIALIZED برگشت داده می شود. • CKR_ENCRYPTED_DATA_LEN_RANGE: طول داده های ورودی متن رمزی برای این تابع نادرست باشد. بسته به مکانیزم رمزنگاری، این کد خطا می تواند به این معنی باشد که داده های متن رمزی خیلی کوتاه یا خیلی بلند است یا مضربی از طول بلوک معین نیست. • CKR_ENCRYPTED_DATA_INVALID: اگر داده های ورودی متن رمزی به تابع نامعتبر باشد این کد خطا برگشت داده می شود. این مقدار بازگشتی نسبت به CKR_ENCRYPTED_DATA_LEN_RANGE دارای اولویت پایین تری است. • در صورتی که اندازه متن رمزگشایی شده خروجی از اندازه بافر تعیین شده بزرگتر باشد، کد خطای CKR_BUFFER_TOO_SMALL برگشت داده می شود. • اگر مکانیزم رمزگشایی یک الگوریتم AEAD باشد و صحت داده های مرتبط یا متن رمزی را نمی توان راستی آزمایی کرد، کد خطای CKR_AEAD_DECRYPT_FAILED بازگردانده می شود.
	P 5.11.3	C_DecryptMessageBegin	• این تابع عملیات رمزگشایی مبتنی بر پیام برای داده چند بخشی را آغاز می کند. • اگر ورودی های pAssociatedData و ulAssociatedDataLen با مقادیر (NULL, 0) تعیین شوند، آنگاه تابع C_DecryptMessageBegin دارای کارکرد رمزگشایی محض مشابه C_Decryp است. اما اگر این مقادیر ورودی غیرتهی باشند، آنگاه تابع دارای کارکرد AEAD است. • پس از فراخوانی C_DecryptMessageBegin، برنامه باید یک یا چند بار تابع C_DecryptMessageNext را فراخوانی کند تا پیام رمزگذاری شده را در چندین بخش رمزگشایی کند. عملیات رمزگشایی پیام تا زمانی فعال است

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
			که برنامه از فراخوانی C_DecryptMessageNext با پرچم CKF_END_OF_MESSAGE استفاده کرده و متن نهایی را بدست آورد. • در صورتی که یک فراخوانی C_DecryptMessageBegin فعال بدون فراخوانی C_DecryptMessageNext وجود داشته باشد، نمی توان تابع C_DecryptMessageBegin را دوباره فراخوانی کرد، در صورت فراخوانی دوباره تابع در چنین شرایطی کد خطای CKR_OPERATION_ACTIVE برگشت داده می شود.
	P 5.11.4	**C_DecryptMessageNext**	• این تابع عملیات رمزگشایی مبتنی بر پیام برای داده چند بخشی را ادامه و به پایان می رساند. • قبل از فراخوانی این تابع باید تابع C_DecryptMessageBegin فراخوانی شده باشد. تابع C_DecryptMessageNext را می توان هر چند بار متوالی فراخوانی کرد. فراخوانی این تابع با مقدار پرچم 0، که منجر به خطایی غیر از CKR_BUFFER_TOO_SMALL شود عملیات رمزگشایی پیام فعلی را خاتمه می دهد. • فراخوانی C_DecryptMessageNext با پرچم CKF_END_OF_MESSAGE همیشه عملیات رمزگشایی فعال را خاتمه می دهد مگر اینکه CKR_BUFFER_TOO_SMALL را برگرداند. • اگرچه آخرین فراخوان C_DecryptMessageNext، رمزگشایی پیام را به پایان می رساند، اما فرآیند رمزگشایی مبتنی بر پیام پایان نمی یابد، بلکه فراخوانی های C_DecryptMessage یا C_DecryptMessageBegin به همراه C_DecryptMessageNext ممکن است در نشست دوباره انجام شود. • قبل از فراخوانی این تابع، باید تابع C_MessageDecryptInit فراخوانی شده باشد، در غیر ایصورت کد خطای CKR_OPERATION_NOT_INITIALIZED برگشت داده می شود. • CKR_ENCRYPTED_DATA_LEN_RANGE: طول داده های ورودی متن رمزی برای این تابع نادرست باشد. بسته به مکانیزم رمزنگاری، این کد خطا می تواند به این معنی باشد که داده های متن رمزی خیلی کوتاه یا خیلی بلند است یا مضربی از طول بلوک معین نیست. • CKR_ENCRYPTED_DATA_INVALID: اگر داده های ورودی متن رمزی به تابع نامعتبر باشد این کد خطا برگشت داده می شود. این مقدار بازگشتی نسبت به CKR_ENCRYPTED_DATA_LEN_RANGE دارای اولویت پایین تری است. • در صورتی که اندازه متن رمزگشایی شده خروجی از اندازه بافر تعیین شده بزرگتر باشد، کد خطای CKR_BUFFER_TOO_SMALL برگشت داده می شود.

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
			• اگر مکانیزم رمزگشایی یک الگوریتم AEAD باشد و صحت داده های مرتبط یا متن رمزی را نمی توان راستی آزمایی کرد، کد خطای CKR_AEAD_DECRYPT_FAILED بازگردانده می شود.
	P 5.11.5	**C_MessageDecryptFinal**	• این تابع عملیات رمزگشایی فعال شده با C_MessageDecryptInit را به پایان می رساند. اگر قبل از فراخوانی این تابع، تابع C_MessageDeccryptInit فراخوانی نشده باشد، کد خطای CKR_OPERATION_NOT_INITIALIZED برگشت داده می شود.
توابع چکیده سازی پیام	**P 5.12.1**	**C_DigestInit**	• این تابع عملیات چکیده سازی پیام را مقدار دهی اولیه می کند. • پس از فراخوانی این تابع، برای چکیده سازی یکپارچه داده تابع C_Digest و برای چکیده سازی چند بخشی داده تابع C_DigestUpdate و به دنبال آن تابع C_DigestFinal فراخوانی می شود. با فراخوانی تابع C_DigestInit عملیات چکیده سازی داده فعال می شود تا زمانی که فراخوانی موفق تابع C_DigestFinal رخ دهد. • اگر مکانیزم ناشناخته ای مشخص شده باشد یا مکانیزم مشخص شده در توکن انتخاب شده قابل استفاده نباشد، کد خطای CKR_MECHANISM_INVALID بازگشت داده می شود. • اگر پارامترهای نامعتبر به مکانیزم مشخص شده برای عملیات چکیده سازی ارائه شود، کد خطای CKR_MECHANISM_PARAM_INVALID برگشت داده می شود. • وقتی که یک عملیات چکیده سازی فعال وجود داشته باشد، باید از فراخوانی دوباره تابع C_DigestInit جلوگیری شده و کد خطای CKR_OPERATION_ACTIVE برگشت داده شود. • برای خاتمه دادن به یک عملیات چکیده سازی فعال، تابع C_DigestInit را می توان با pMechanism تنظیم شده روی NULL_PTR فراخوانی کرد. اگر عملیات چکیده سازی لغو نشود، کد خطای CKR_OPERATION_CANCEL_FAILED باید برگردانده شود.
	P 5.12.2	**C_Digest**	• این تابع عملیات چکیده سازی داده را به صورت یکپارچه انجام می دهد. • قبل از فراخوانی این تابع، باید تابع C_DigestInit فراخوانی شده باشد، در غیر ایصورت کد خطای CKR_OPERATION_NOT_INITIALIZED برگشت داده می شود. • در صورتی که اندازه خروجی از اندازه بافر تعیین شده بزرگتر باشد، کد خطای CKR_BUFFER_TOO_SMALL برگشت داده می شود.

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
	P 5.12.3	C_DigestUpdate	• این تابع عملیات چکیده سازی داده را به صورت چند بخشی انجام می دهد. • قبل از فراخوانی این تابع، باید تابع C_DigestInit فراخوانی شده باشد، در غیر ایصورت کد خطای CKR_OPERATION_NOT_INITIALIZED برگشت داده می شود.
	P 5.12.4	C_DigestKey	• این تابع عملیات چکیده سازی چند بخشی را برای یک کلید محرمانه انجام می دهد. • قبل از فراخوانی این تابع، باید تابع C_DigestInit فراخوانی شده باشد، در غیر اینصورت کد خطای CKR_OPERATION_NOT_INITIALIZED برگشت داده می شود. • اگر امکان چکیده سازی کلید مشخص شده وجود نداشته باشد کد خطای CKR_KEY_INDIGESTIBLE برگشت داده می شود. این امر ممکن است به این دلیل باشد که کلید مشخص شده یک کلید مخفی نیست، یا شاید توکن به راحتی نمی تواند این نوع کلید را چکیده کند. • اگر شناسه کلید مشخص شده معتبر نباشد، کد خطای CKR_KEY_HANDLE_INVALID برگشت داده می شود. باید توجه داشت که مقدار صفر (۰) هرگز یک شناسه کلید معتبر نیست. • اگر اندازه کلید ارائه شده خارج از محدوده مجاز باشد، کد خطای CKR_KEY_SIZE_RANGE برگشت داده می شود.
	P 5.12.5	C_DigestFinal	• این تابع عملیات چکیده سازی داده را به صورت چند بخشی به پایان می رساند و مقدار چکیده پیام را برمی گرداند. • قبل از فراخوانی این تابع، باید تابع C_DigestInit فراخوانی شده باشد، در غیر اینصورت کد خطای CKR_OPERATION_NOT_INITIALIZED برگشت داده می شود. • در صورتی که اندازه خروجی از اندازه بافر تعیین شده بزرگتر باشد، کد خطای CKR_BUFFER_TOO_SMALL برگشت داده می شود.

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
توابع امضا و MAC	P 5.13.1	C_SignInit	• این تابع عملیات امضای داده را مقدار دهی اولیه می کند. • پس از فراخوانی این تابع، برای امضای یکپارچه داده تابع C_Sign و برای امضای چند بخشی داده تابع C_SignUpdate به تعداد صفر یا چند بار و تابع C_SignFinal یک بار فراخوانی می شود. با فراخوانی تابع C_SignInit عملیات امضای فعال می شود تا زمانی که فراخوانی موفق C_Sign یا C_SignFinal رخ دهد. • یکی از ورودی های این تابع، کلید امضا است. ویژگی CKA_SIGN برای این کلید، باید CK_TRUE باشد (که نشان می دهد این کلید از عملیات امضا پشتیبانی می کند)، در غیر اینصورت، کد خطای CKR_KEY_FUNCTION_NOT_PERMITTED برگشت داده می شود. • اگر شناسه کلید مشخص شده معتبر نباشد، کد خطای CKR_KEY_HANDLE_INVALID برگشت داده می شود. • اگر اندازه کلید ارائه شده خارج از محدوده مجاز باشد، کد خطای CKR_KEY_SIZE_RANGE برگشت داده می شود. • اگر نوع کلید ورودی برای استفاده در مکانیزم تعیین شده همخوانی نداشته باشد، کد خطای CKR_KEY_TYPE_INCONSISTENT برگشت داده می شود. • اگر مکانیزم ناشناخته ای مشخص شده باشد یا مکانیزم مشخص شده در توکن انتخاب شده قابل استفاده نباشد، کد خطای CKR_MECHANISM_INVALID بازگشت داده می شود. • اگر پارامترهای نامعتبر به مکانیزم مشخص شده برای عملیات امضا ارائه شود، کد خطای CKR_MECHANISM_PARAM_INVALID برگشت داده می شود. • وقتی که یک عملیات امضای فعال وجود داشته باشد، باید از فعال کردن امضا با تابع C_SignInit جلوگیری شده و کد خطای CKR_OPERATION_ACTIVE برگشت داده شود. • برای خاتمه دادن به یک عملیات امضای فعال، تابع C_SignInit را می توان با pMechanism تنظیم شده روی NULL_PTR فراخوانی کرد. اگر عملیات امضا لغو نشود، کد خطای CKR_OPERATION_CANCEL_FAILED باید برگردانده شود.

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
	P 5.13.2	**C_Sign**	• این تابع عملیات امضای داده را به صورت یکپارچه انجام می دهد. • قبل از فراخوانی این تابع، باید تابع C_SignInit فراخوانی شده باشد، در غیر ایصورت کد خطای CKR_OPERATION_NOT_INITIALIZED برگشت داده می شود. • اگر طول داده های ورودی تابع نادرست باشد کد خطای CKR_DATA_LEN_RANGE برگشت داده می شود. بسته به مکانیزم رمزنگاری، این کد خطا می تواند به این معنی باشد که داده های متن اصلی خیلی کوتاه با خیلی بلند است یا مضربی از طول بلوک معین نیست. • اگر داده های ورودی به تابع نامعتبر باشد این کد خطای CKR_DATA_INVALID برگشت داده می شود. • در صورتی که اندازه خروجی از اندازه بافر تعیین شده بزرگتر باشد، کد خطای CKR_BUFFER_TOO_SMALL برگشت داده می شود. • اگر درخواست امضا توسط کاربر لغو شود کد خطای CKR_FUNCTION_REJECTED برگشت داده می شود.
	P 5.13.3	**C_SignUpdate**	• این تابع عملیات امضای داده را به صورت چند بخشی انجام می دهد. • قبل از فراخوانی این تابع، باید تابع C_DigestInit فراخوانی شده باشد، در غیر ایصورت کد خطای CKR_OPERATION_NOT_INITIALIZED برگشت داده می شود. • اگر طول داده های ورودی تابع نادرست باشد کد خطای CKR_DATA_LEN_RANGE برگشت داده می شود. بسته به مکانیزم رمزنگاری، این کد خطا می تواند به این معنی باشد که داده های متن اصلی خیلی کوتاه با خیلی بلند است یا مضربی از طول بلوک معین نیست.

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
	P 5.13.4	**C_SignFinal**	• این تابع عملیات امضای داده را به صورت چند بخشی به پایان می رساند و مقدار امضا را برمی گرداند. • قبل از فراخوانی این تابع، باید تابع C_SignInit فراخوانی شده باشد، در غیر اینصورت کد خطای CKR_OPERATION_NOT_INITIALIZED برگشت داده می شود. • در صورتی که اندازه خروجی از اندازه بافر تعیین شده بزرگتر باشد، کد خطای CKR_BUFFER_TOO_SMALL برگشت داده می شود. • اگر درخواست امضا توسط کاربر لغو شود کد خطای CKR_FUNCTION_REJECTED برگشت داده می شود. • اگر طول داده های ورودی تابع نادرست باشد کد خطای CKR_DATA_LEN_RANGE برگشت داده می شود. بسته به مکانیزم رمزنگاری، این کد خطا می تواند به این معنی باشد که داده های متن اصلی خیلی کوتاه با خیلی بلند است یا مضربی از طول بلوک معین نیست.

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
	P 5.13.5	C_SignRecoverInit	• این تابع عملیات بازیابی داده از امضا را مقدار دهی اولیه می کند. • پس از فراخوانی این تابع، برای بازیابی داده از تابع C_SignRecover فراخوانی می شود. با فراخوانی تابع C_SignRecoverInit عملیات بازیابی داده از امضای فعال می شود تا زمانی که فراخوانی موفق C_SignRecover رخ دهد. • یکی از ورودی های این تابع، کلید امضا است. ویژگی CKA_SIGN_RECOVER برای این کلید، باید CK_TRUE باشد (که نشان می دهد این کلید از عملیات بازیابی داده از امضا پشتیبانی می کند)، در غیر اینصورت، کد خطای CKR_KEY_FUNCTION_NOT_PERMITTED برگشت داده می شود. • اگر شناسه کلید مشخص شده معتبر نباشد، کد خطای CKR_KEY_HANDLE_INVALID برگشت داده می شود. • اگر اندازه کلید ارائه شده خارج از محدوده مجاز باشد، کد خطای CKR_KEY_SIZE_RANGE برگشت داده می شود. • اگر نوع کلید ورودی برای استفاده در مکانیزم تعیین شده همخوانی نداشته باشد، کد خطای CKR_KEY_TYPE_INCONSISTENT برگشت داده می شود. • اگر مکانیزم ناشناخته ای مشخص شده باشد یا مکانیزم مشخص شده در توکن انتخاب شده قابل استفاده نباشد، کد خطای CKR_MECHANISM_INVALID بازگشت داده می شود. • اگر پارامترهای نامعتبر به مکانیزم مشخص شده برای عملیات امضا ارائه شود، کد خطای CKR_MECHANISM_PARAM_INVALID برگشت داده می شود. • وقتی که یک عملیات بازیابی داده از امضای فعال وجود داشته باشد، باید از فعال کردن دوباره امضا با تابع C_SignRecoverInit جلوگیری شده و کد خطای CKR_OPERATION_ACTIVE برگشت داده شود. • برای خاتمه دادن به یک عملیات بازیابی داده از امضای فعال، تابع C_SignRecoverInit را می توان با pMechanism تنظیم شده روی NULL_PTR فراخوانی کرد. اگر عملیات امضا لغو نشود، کد خطای CKR_OPERATION_CANCEL_FAILED باید برگردانده شود.

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
	P 5.13.6	**C_SignRecover**	• این تابع عملیات بازیابی داده از امضا را انجام می دهد. • قبل از فراخوانی این تابع، باید تابع C_SignRecoverInit فراخوانی شده باشد، در غیر ایصورت کد خطای CKR_OPERATION_NOT_INITIALIZED برگشت داده می شود. • اگر طول داده های ورودی تابع نادرست باشد کد خطای CKR_DATA_LEN_RANGE برگشت داده می شود. بسته به مکانیزم رمزنگاری، این کد خطا می تواند به این معنی باشد که داده های متن اصلی خیلی کوتاه یا خیلی بلند است یا مضربی از طول بلوک معین نیست. • اگر داده های ورودی به تابع نامعتبر باشد این کد خطای CKR_DATA_INVALID برگشت داده می شود. • در صورتی که اندازه خروجی از اندازه بافر تعیین شده بزرگتر باشد، کد خطای CKR_BUFFER_TOO_SMALL برگشت داده می شود.

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
توابع مبتنی بر پیام برای امضا و MAC	P 5.14.1	C_MessageSignInit	• این تابع مقدار دهی اولیه فرآیند امضای مبتنی بر پیام، آماده سازی نشست برای یک یا چند عملیات امضا که با استفاده از مکانیسم امضا و کلید امضا مشابهی استفاده می کنند. • پس از فراخوانی این تابع، برای امضای مبتنی بر پیام یکپارچه داده تابع C_MessageSign و برای امضای مبتنی بر پیام چند بخشی داده تابع C_MessageSignIBegin به همراه تابع C_MessageSignNext به تعداد یک یا چند بار فراخوانی می شود. با فراخوانی تابع C_MessageSignInit عملیات امضای مبتنی بر پیام فعال می شود تا زمانی که فراخوانی موفق C_MessageSignFinal رخ دهد. • یکی از ورودی های این تابع، کلید امضا است. ویژگی CKA_SIGN برای این کلید، باید CK_TRUE باشد (که نشان می دهد این کلید از عملیات امضا پشتیبانی می کند)، در غیر اینصورت، کد خطای CKR_KEY_FUNCTION_NOT_PERMITTED برگشت داده می شود. • اگر شناسه کلید مشخص شده معتبر نباشد، کد خطای CKR_KEY_HANDLE_INVALID برگشت داده می شود. • اگر اندازه کلید ارائه شده خارج از محدوده مجاز باشد، کد خطای CKR_KEY_SIZE_RANGE برگشت داده می شود. • اگر نوع کلید ورودی برای استفاده در مکانیزم تعیین شده همخوانی نداشته باشد، کد خطای CKR_KEY_TYPE_INCONSISTENT برگشت داده می شود. • اگر مکانیزم ناشناخته ای مشخص شده باشد یا مکانیزم مشخص شده در توکن انتخاب شده قابل استفاده نباشد، کد خطای CKR_MECHANISM_INVALID بازگشت داده می شود. • اگر پارامترهای نامعتبر به مکانیزم مشخص شده برای عملیات امضا ارائه شود، کد خطای CKR_MECHANISM_PARAM_INVALID برگشت داده می شود. • وقتی که یک عملیات امضای فعال وجود داشته باشد، باید از فعال کردن امضا با تابع C_MessageSignInit جلوگیری شده و کد خطای CKR_OPERATION_ACTIVE برگشت داده شود. • برای خاتمه دادن به یک عملیات امضای فعال، تابع C_MessageSignInit را می توان با pMechanism تنظیم شده روی NULL_PTR فراخوانی کرد. اگر عملیات امضا لغو نشود، کد خطای CKR_OPERATION_CANCEL_FAILED باید برگردانده شود.

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
	P 5.14.2	**C_SignMessage**	• این تابع عملیات امضای مبتنی بر پیام یکپارچه داده را انجام می دهد. • قبل از فراخوانی این تابع، باید تابع C_MessageSignInit فراخوانی شده باشد، در غیر ایصورت کد خطای CKR_OPERATION_NOT_INITIALIZED برگشت داده می شود. • اگر طول داده های ورودی تابع نادرست باشد کد خطای CKR_DATA_LEN_RANGE برگشت داده می شود. بسته به مکانیزم رمزنگاری، این کد خطا می تواند به این معنی باشد که داده های متن اصلی خیلی کوتاه یا خیلی بلند است یا مضربی از طول بلوک معین نیست. • اگر داده های ورودی به تابع نامعتبر باشد این کد خطای CKR_DATA_INVALID برگشت داده می شود. • در صورتی که اندازه خروجی از اندازه بافر تعیین شده بزرگتر باشد، کد خطای CKR_BUFFER_TOO_SMALL برگشت داده می شود. • اگر درخواست امضا توسط کاربر لغو شود کد خطای CKR_FUNCTION_REJECTED برگشت داده می شود.
	P 5.14.3	**C_SignMessageBegin**	• این تابع عملیات امضای مبتنی بر پیام چند بخشی داده را آغاز می کند. • قبل از فراخوانی این تابع، باید تابع C_MessageSignInit فراخوانی شده باشد، در غیر ایصورت کد خطای CKR_OPERATION_NOT_INITIALIZED برگشت داده می شود.

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
	P 5.14.4	**C_SignMessageNext**	• این تابع عملیات امضای مبتنی بر پیام چند بخشی داده را ادامه و به پایان می رساند. • قبل از فراخوانی این تابع باید تابع C_SignMessageBegin فراخوانی شده باشد. تابع C_SignMessageNext را می توان هر چند بار متوالی فراخوانی کرد. فراخوانی این تابع با مقدار غیر NULL برای پارامتر ورودی *pulSignatureLen*، که منجر به خطایی غیر از CKR_BUFFER_TOO_SMALL شود، عملیات امضای پیام فعلی را خاتمه می دهد. • فراخوانی C_SignMessageNext با مقدار NULL برای پارامتر ورودی *pulSignatureLen*، همیشه عملیات امضای فعال را خاتمه می دهد مگر اینکه CKR_BUFFER_TOO_SMALL را برگرداند. • اگرچه آخرین فراخوان C_SignMessageNext، امضای پیام را به پایان می رساند، اما فرآیند امضا مبتنی بر پیام پایان نمی یابد، بلکه فراخوانی های C_SignMessage یا C_SignMessageBegin به همراه C_SignMessageNextمی تواند در نشست دوباره انجام شود. • قبل از فراخوانی این تابع، باید تابع C_SignMessageInit فراخوانی شده باشد، در غیر اینصورت کد خطای CKR_OPERATION_NOT_INITIALIZED برگشت داده می شود. • اگر طول داده های ورودی برای این تابع نادرست باشد این کد خطا CKR_DATA_LEN_RANGE برگشت داده می شود. بسته به مکانیزم رمزنگاری، این کد خطا می تواند به این معنی باشد که داده های متن رمزی خیلی کوتاه یا خیلی بلند است یا مضربی از طول بلوک معین نیست. • در صورتی که اندازه متن رمزگشایی شده خروجی از اندازه بافر تعیین شده بزرگتر باشد، کد خطای CKR_BUFFER_TOO_SMALL برگشت داده می شود.
	P 5.14.5	**C_MessageSignFinal**	• این تابع عملیات امضای مبتنی بر پیام را خاتمه می دهد. • قبل از فراخوانی این تابع، باید تابع C_MessageSignInit فراخوانی شده باشد، در غیر ایصورت کد خطای CKR_OPERATION_NOT_INITIALIZED برگشت داده می شود. • اگر درخواست امضا توسط کاربر لغو شود کد خطای CKR_FUNCTION_REJECTED برگشت داده می شود.

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
توابع راستی آزمایی امضا و MAC	P 5.15.1	C_VerifyInit	• این تابع عملیات راستی آزمایی امضای داده را مقدار دهی اولیه می کند. • پس از فراخوانی این تابع، برای راستی آزمایی امضای یکپارچه داده تابع C_Verify و برای راستی آزمایی امضای چند بخشی داده تابع C_VerifyUpdate به همراه تابع C_VerifyFinal به تعداد یک یا چند بار فراخوانی می شود. با فراخوانی تابع C_VerifyInit عملیات راستی آزمایی امضا فعال می شود تا زمانی که فراخوانی موفق C_Verify یا C_VerifyFinal رخ دهد. • یکی از ورودی های این تابع، کلید راستی آزمایی امضا است. ویژگی CKA_VERIFY برای این کلید، باید CK_TRUE باشد (که نشان می دهد این کلید از عملیات راستی آزمایی امضا پشتیبانی می کند)، در غیر اینصورت، کد خطای CKR_KEY_FUNCTION_NOT_PERMITTED برگشت داده می شود. • اگر شناسه کلید مشخص شده معتبر نباشد، کد خطای CKR_KEY_HANDLE_INVALID برگشت داده می شود. • اگر اندازه کلید ارائه شده خارج از محدوده مجاز باشد، کد خطای CKR_KEY_SIZE_RANGE برگشت داده می شود. • اگر نوع کلید ورودی برای استفاده در مکانیزم تعیین شده همخوانی نداشته باشد، کد خطای CKR_KEY_TYPE_INCONSISTENT برگشت داده می شود. • اگر مکانیزم ناشناخته ای مشخص شده باشد یا مکانیزم مشخص شده در توکن انتخاب شده قابل استفاده نباشد، کد خطای CKR_MECHANISM_INVALID بازگشت داده می شود. • اگر پارامترهای نامعتبر به مکانیزم مشخص شده برای عملیات راستی آزمایی امضا ارائه شود، کد خطای CKR_MECHANISM_PARAM_INVALID برگشت داده می شود. • وقتی که یک عملیات راستی آزمایی امضای فعال وجود داشته باشد، باید از فعال کردن عملیات راستی آزمایی امضا با تابع C_VerifyInit جلوگیری شده و کد خطای CKR_OPERATION_ACTIVE برگشت داده شود. • برای خاتمه دادن به یک عملیات راستی آزمایی امضای فعال، تابع C_VerifyInit را می توان با pMechanism تنظیم شده روی NULL_PTR فراخوانی کرد. اگر عملیات راستی آزمایی امضا لغو نشود، کد خطای CKR_OPERATION_CANCEL_FAILED باید برگردانده شود.

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
	P 5.15.2	**C_Verify**	• این تابع عملیات راستی آزمایی امضای یکپارچه داده را انجام می دهد. • قبل از فراخوانی این تابع، باید تابع C_VerifyInit فراخوانی شده باشد، در غیر ایصورت کد خطای CKR_OPERATION_NOT_INITIALIZED برگشت داده می شود. • اگر طول امضای ورودی تابع نادرست باشد کد خطای CKR_SIGNATURE_LEN_RANGE برگشت داده می شود. بسته به مکانیزم رمزنگاری، این کد خطا می تواند به این معنی باشد که داده های متن اصلی خیلی کوتاه با خیلی بلند است یا مضربی از طول بلوک معین نیست. • اگر امضای ورودی به تابع نامعتبر باشد این کد خطای CKR_SIGNATURE_INVALID برگشت داده می شود.
	P 5.15.3	**C_VerifyUpdate**	• این تابع عملیات راستی آزمایی امضای چند بخشی داده را انجام می دهد. • قبل از فراخوانی این تابع، باید تابع C_VerifyInit فراخوانی شده باشد، در غیر ایصورت کد خطای CKR_OPERATION_NOT_INITIALIZED برگشت داده می شود. • اگر طول داده های ورودی تابع نادرست باشد کد خطای CKR_DATA_LEN_RANGE برگشت داده می شود. بسته به مکانیزم رمزنگاری، این کد خطا می تواند به این معنی باشد که داده های متن اصلی خیلی کوتاه با خیلی بلند است یا مضربی از طول بلوک معین نیست.
	P 5.15.4	**C_VerifyFinal**	• این تابع عملیات راستی آزمایی امضای چند بخشی داده را ادامه و به پایان می رساند. • قبل از فراخوانی این تابع، باید تابع C_VerifyInit فراخوانی شده باشد، در غیر ایصورت کد خطای CKR_OPERATION_NOT_INITIALIZED برگشت داده می شود. • اگر طول امضای ورودی تابع نادرست باشد کد خطای CKR_SIGNATURE_LEN_RANGE برگشت داده می شود. بسته به مکانیزم رمزنگاری، این کد خطا می تواند به این معنی باشد که داده های متن اصلی خیلی کوتاه با خیلی بلند است یا مضربی از طول بلوک معین نیست. • اگر امضای ورودی به تابع نامعتبر باشد این کد خطای CKR_SIGNATURE_INVALID برگشت داده می شود.

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
	P 5.15.5	C_VerifyRecoverInit	• این تابع عملیات راستی آزمایی امضا را برای داده های بازیابی شده از امضا مقدار دهی اولیه می کند. • پس از فراخوانی این تابع، برای راستی آزمایی امضای یکپارچه داده تابع C_VerifyRecover فراخوانی می شود. با فراخوانی تابع C_VerifyRecoverInit عملیات راستی آزمایی امضا فعال می شود تا زمانی که فراخوانی موفق C_VerifyRecover رخ دهد. • یکی از ورودی های این تابع، کلید امضا است. ویژگی CKA_VERIFY_RECOVER برای این کلید، باید CK_TRUE باشد (که نشان می دهد این کلید از عملیات راستی آزمایی امضا پشتیبانی می کند)، در غیر اینصورت، کد خطای CKR_KEY_FUNCTION_NOT_PERMITTED برگشت داده می شود. • اگر شناسه کلید مشخص شده معتبر نباشد، کد خطای CKR_KEY_HANDLE_INVALID برگشت داده می شود. • اگر اندازه کلید ارائه شده خارج از محدوده مجاز باشد، کد خطای CKR_KEY_SIZE_RANGE برگشت داده می شود. • اگر نوع کلید ورودی برای استفاده در مکانیزم تعیین شده همخوانی نداشته باشد، کد خطای CKR_KEY_TYPE_INCONSISTENT برگشت داده می شود. • اگر مکانیزم ناشناخته ای مشخص شده باشد یا مکانیزم مشخص شده در توکن انتخاب شده قابل استفاده نباشد، کد خطای CKR_MECHANISM_INVALID بازگشت داده می شود. • اگر پارامترهای نامعتبر به مکانیزم مشخص شده برای عملیات راستی آزمایی امضا ارائه شود، کد خطای CKR_MECHANISM_PARAM_INVALID برگشت داده می شود. • وقتی که یک عملیات راستی آزمایی امضای فعال وجود داشته باشد، باید از فعال کردن عملیات راستی آزمایی امضا با تابع C_VerifyRecoverInit جلوگیری شده و کد خطای CKR_OPERATION_ACTIVE برگشت داده شود. • برای خاتمه دادن به یک عملیات راستی آزمایی امضای فعال، تابع C_VerifyRecoverInit را می توان با pMechanism تنظیم شده روی NULL_PTR فراخوانی کرد. اگر عملیات راستی آزمایی امضا لغو نشود، کد خطای CKR_OPERATION_CANCEL_FAILED باید برگردانده شود.

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
	P 5.15.6	**C_VerifyRecover**	• این تابع عملیات راستی آزمایی امضا را بر روی داده های یکپارچه انجام می دهد. در این تابع داده ها ابتدا از روی امضا بازیابی شده و سپس راستی آزمایی امضا انجام می شود. • قبل از فراخوانی این تابع، باید تابع C_VerifyRecoverInit فراخوانی شده باشد، در غیر ایصورت کد خطای CKR_OPERATION_NOT_INITIALIZED برگشت داده می شود. • اگر طول امضای ورودی تابع نادرست باشد کد خطای CKR_SIGNATURE_LEN_RANGE برگشت داده می شود. بسته به مکانیزم رمزنگاری، این کد خطا می تواند به این معنی باشد که داده های متن اصلی خیلی کوتاه با خیلی بلند است یا مضربی از طول بلوک معین نیست. • اگر امضای ورودی به تابع نامعتبر باشد این کد خطای CKR_SIGNATURE_INVALID برگشت داده می شود. • در صورتی که اندازه خروجی از اندازه بافر تعیین شده بزرگتر باشد، کد خطای CKR_BUFFER_TOO_SMALL برگشت داده می شود.

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
توابع مبتنی بر پیام برای راستی آزمایی امضا و MAC	P 5.16.1	C_MessageVerifyInit	• این تابع عملیات راستی آزمایی امضای مبتنی بر پیام داده را مقداردهی اولیه می کند. • پس از فراخوانی این تابع، برای راستی آزمایی امضای مبتنی بر پیام یکپارچه داده تابع C_MessageVerify و برای راستی آزمایی امضای چند بخشی داده تابع C_MessageVerifyBegin به همراه تابع C_MessageVerifyNext به تعداد یک یا چند بار فراخوانی می شود. با فراخوانی تابع C_MessageVerifyInit عملیات راستی آزمایی امضا فعال می شود تا زمانی که فراخوانی موفق C_MessageVerifyFinal رخ دهد. • یکی از ورودی های این تابع، کلید راستی آزمایی امضا است. ویژگی CKA_VERIFY برای این کلید، باید CK_TRUE باشد (که نشان می دهد این کلید از عملیات راستی آزمایی امضا پشتیبانی می کند)، در غیر اینصورت، کد خطای CKR_KEY_FUNCTION_NOT_PERMITTED برگشت داده می شود. • اگر شناسه کلید مشخص شده معتبر نباشد، کد خطای CKR_KEY_HANDLE_INVALID برگشت داده می شود. • اگر اندازه کلید ارائه شده خارج از محدوده مجاز باشد، کد خطای CKR_KEY_SIZE_RANGE برگشت داده می شود. • اگر نوع کلید ورودی برای استفاده در مکانیزم تعیین شده همخوانی نداشته باشد، کد خطای CKR_KEY_TYPE_INCONSISTENT برگشت داده می شود. • اگر مکانیزم ناشناخته ای مشخص شده باشد یا مکانیزم مشخص شده در توکن انتخاب شده قابل استفاده نباشد، کد خطای CKR_MECHANISM_INVALID بازگشت داده می شود. • اگر پارامترهای نامعتبر به مکانیزم مشخص شده برای عملیات راستی آزمایی امضا ارائه شود، کد خطای CKR_MECHANISM_PARAM_INVALID برگشت داده می شود. • وقتی که یک عملیات راستی آزمایی امضای فعال وجود داشته باشد، باید از فعال کردن عملیات راستی آزمایی امضا با تابع C_VerifyInit جلوگیری شده و کد خطای CKR_OPERATION_ACTIVE برگشت داده شود.

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
	P 5.16.2	C_MessageVerify	• این تابع عملیات راستی آزمایی امضای مبتنی بر پیام داده را به صورت یکپارچه انجام می دهد. • قبل از فراخوانی این تابع، باید تابع C_MessageVerifyInit فراخوانی شده باشد، در غیر ایصورت کد خطای CKR_OPERATION_NOT_INITIALIZED برگشت داده می شود. • اگر طول امضای ورودی تابع نادرست باشد کد خطای CKR_SIGNATURE_LEN_RANGE برگشت داده می شود. بسته به مکانیزم رمزنگاری، این کد خطا می تواند به این معنی باشد که داده های متن اصلی خیلی کوتاه با خیلی بلند است یا مضربی از طول بلوک معین نیست. • اگر امضای ورودی به تابع نامعتبر باشد این کد خطای CKR_SIGNATURE_INVALID برگشت داده می شود.
	P 5.16.3	C_VerifyMessageBegin	• این تابع عملیات راستی آزمایی امضای مبتنی بر پیام چند بخشی داده را آغاز می کند. • قبل از فراخوانی این تابع، باید تابع C_MessageVerifyInit فراخوانی شده باشد، در غیر ایصورت کد خطای CKR_OPERATION_NOT_INITIALIZED برگشت داده می شود.

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
	P 5.16.4	**C_VerifyMessageNext**	• این تابع عملیات راستی آزمایی امضای مبتنی بر پیام چند بخشی داده را ادامه و پایان می دهد. • قبل از فراخوانی این تابع باید تابع C_SignMessageBegin فراخوانی شده باشد. تابع C_VerifyMessageNext را می توان هر چند بار متوالی فراخوانی کرد. فراخوانی این تابع با مقدار غیر NULL برای پارامتر ورودی *pulSignatureLen*، که منجر به خطایی غیر از CKR_BUFFER_TOO_SMALL شود، عملیات راستی آزمایی امضای پیام فعلی را خاتمه می دهد. • فراخوانی C_VerifyMessageNext با مقدار NULL برای پارامتر ورودی pSignature، به این معناست که بخش بعدی داده برای راستی آزمایی وجود دارد، اگر این پارامتر دارای مقدار غیر NULL باشد به این معنی است که بخش پایانی داده است. • اگرچه آخرین فراخوان C_VerifyMessageNext، راستی آزمایی امضای پیام را به پایان می رساند، اما فرآیند راستی آزمایی امضای مبتنی بر پیام پایان نمی یابد، بلکه فراخوانی های C_VerifyMessage یا C_VerifyMessageBegin به همراه C_VerifyMessageNext می تواند در نشست دوباره انجام شود. • قبل از فراخوانی این تابع، باید تابع C_VerifyMessageInit فراخوانی شده باشد، در غیر اینصورت کد خطای CKR_OPERATION_NOT_INITIALIZED برگشت داده می شود. • اگر طول داده ورودی برای این تابع نادرست باشد این کد خطا CKR_DATA_LEN_RANGE برگشت داده می شود. • اگر طول امضای ورودی برای این تابع نادرست باشد این کد خطا CKR_SIGNATURE_LEN_RANGE برگشت داده می شود. بسته به مکانیزم رمزنگاری، این کد خطا می تواند به این معنی باشد که داده های متن رمزی خیلی کوتاه یا خیلی بلند است یا مضربی از طول بلوک معین نیست. • اگر امضای ورودی به تابع نامعتبر باشد این کد خطای CKR_SIGNATURE_INVALID برگشت داده می شود.
	P 5.16.5	**C_MessageVerifyFinal**	• این تابع عملیات راستی آزمایی امضای مبتنی بر پیام را خاتمه می دهد. • قبل از فراخوانی این تابع، باید تابع C_VerifyMessageInit فراخوانی شده باشد، در غیر اینصورت کد خطای CKR_OPERATION_NOT_INITIALIZED برگشت داده می شود.

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
توابع رمزنگاری دو عملیاتی	P 5.17.1	C_DigestEncryptUpdate	• این تابع عملیات چکیده سازی و رمزگذاری چند بخشی داده را بطور همزمان انجام می دهد. • این تابع کارکرد توابع C_EncryptUpdate و C_DigestUpdate را بطور همزمان دارا می باشد. بنابراین باید قبل از فراخوانی این تابع، باید عملیات چکیده سازی و رمزگذاری پیام با اسنفاده از توابع C_DigestInit و C_EncryptInit فعالسازی شده باشند. در غیر اینصورت کد خطای CKR_OPERATION_NOT_INITIALIZED برگشت داده می شود. • CKR_DATA_LEN_RANGE: اگر طول داده های ورودی متن اصلی برای این تابع نادرست باشد. بسته به مکانیزم رمزنگاری، این کد خطا می تواند به این معنی باشد که داده های متن اصلی خیلی کوتاه با خیلی بلند است یا مضربی از طول بلوک معین نیست. • در صورتی که اندازه متن رمزی خروجی از اندازه بافر تعیین شده بزرگتر باشد، کد خطای CKR_BUFFER_TOO_SMALL برگشت داده می شود.
	P 5.17.2	C_DecryptDigestUpdate	• این تابع عملیات چکیده سازی و رمزگشایی چند بخشی داده را بطور همزمان انجام می دهد. • این تابع کارکرد توابع C_DecryptUpdate و C_DigestUpdate را بطور همزمان دارا می باشد. بنابراین باید قبل از فراخوانی این تابع، باید عملیات چکیده سازی و رمزگذاری پیام با اسنفاده از توابع C_DigestInit و C_DecryptInit فعالسازی شده باشند. در غیر اینصورت کد خطای CKR_OPERATION_NOT_INITIALIZED برگشت داده می شود. • اگر طول داده های ورودی متن رمزی برای این تابع نادرست باشد، کد خطای CKR_ENCRYPTED_DATA_LEN_RANGE برگشت داده می شود. بسته به مکانیزم رمزنگاری، این کد خطا می تواند به این معنی باشد که داده های متن رمزی خیلی کوتاه با خیلی بلند است یا مضربی از طول بلوک معین نیست. • اگر داده های ورودی متن رمزی به تابع نامعتبر باشد کد خطای CKR_ENCRYPTED_DATA_INVALID برگشت داده می شود. • در صورتی که اندازه متن رمزگشایی شده خروجی از اندازه بافر تعیین شده بزرگتر باشد، کد خطای CKR_BUFFER_TOO_SMALL برگشت داده می شود.

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
	P 5.17.3	**C_SignEncryptUpdate**	• این تابع عملیات امضا و رمزگذاری چند بخشی داده را بطور همزمان انجام می دهد. • این تابع کارکرد توابع C_EncryptUpdate و C_SignUpdate را بطور همزمان دارا می باشد. بنابراین باید قبل از فراخوانی این تابع، باید عملیات چکیده سازی و رمزگذاری پیام با استفاده از توابع C_SignInit و C_EncryptInit فعالسازی شده باشند. در غیر اینصورت کد خطای CKR_OPERATION_NOT_INITIALIZED برگشت داده می شود. • CKR_DATA_LEN_RANGE: اگر طول داده های ورودی متن اصلی برای این تابع نادرست باشد. بسته به مکانیزم رمزنگاری، این کد خطا می تواند به این معنی باشد که داده های متن اصلی خیلی کوتاه یا خیلی بلند است یا مضربی از طول بلوک معین نیست. • در صورتی که اندازه متن رمزی خروجی از اندازه بافر تعیین شده بزرگتر باشد، کد خطای CKR_BUFFER_TOO_SMALL برگشت داده می شود.
	P 5.17.4	**C_DecryptVerifyUpdate**	• این تابع عملیات راستی آزمایی امضا و رمزگشایی چند بخشی داده را بطور همزمان انجام می دهد. • این تابع کارکرد توابع C_DecryptUpdate و C_VerifyUpdate را بطور همزمان دارا می باشد. بنابراین باید قبل از فراخوانی این تابع، باید عملیات چکیده سازی و رمزگذاری پیام با استفاده از توابع C_VerifyInit و C_DecryptInit فعالسازی شده باشند. در غیر اینصورت کد خطای CKR_OPERATION_NOT_INITIALIZED برگشت داده می شود. • اگر طول داده های ورودی متن رمزی برای این تابع نادرست باشد، کد خطای CKR_ENCRYPTED_DATA_LEN_RANGE برگشت داده می شود. بسته به مکانیزم رمزنگاری، این کد خطا می تواند به این معنی باشد که داده های متن رمزی خیلی کوتاه یا خیلی بلند است یا مضربی از طول بلوک معین نیست. • اگر داده های ورودی متن رمزی به تابع نامعتبر باشد کد خطای CKR_ENCRYPTED_DATA_INVALID برگشت داده می شود. • در صورتی که اندازه متن رمزگشایی شده خروجی از اندازه بافر تعیین شده بزرگتر باشد، کد خطای CKR_BUFFER_TOO_SMALL برگشت داده می شود.

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
توابع مدیریت کلید	P 5.18.1	C_GenerateKey	• این تابع یک کلید محرمانه یا مجموعه ای از پارامترهای دامنه را تولید می کند. • این تابع یک شیء از جنس کلید تولید می کند. این کلید یا یک کلید محرمانه است یا یک مجموعه از پارامترهای دامنه است. هر شیء کلید دارای یک ویژگی عمومی CKA_CLASS است. اگر خروجی تابع C_GenerateKey یک مجموعه از پارامترهای دامنه باشد، ویژگی CKA_CLASS دارای مقدار CKO_DOMAIN_PARAMETERS خواهد بود، در غیر این صورت، مقدار آن CKO_SECRET_KEY خواهد بود. • شی ایجاد شده توسط فراخوانی موفقیت آمیز C_GenerateKey ویژگی CKA_LOCALخود را روی CK_TRUE تنظیم خواهد کرد. علاوه بر این، شی ایجاد شده یک شناسه یکتا برای CKA_UNIQUE_ID تولید و تخصیص می دهد. --- • مقدار بازگشتی به ازای مقدار ورودی معتبر برای یک ویژگی نامعتبر: • CKR_ATTRIBUTE_TYPE_INVALID • مقدار بازگشتی به ازای مقدار ورودی نامعتبر برای یک ویژگی معتبر: • CKR_ATTRIBUTE_VALUE_INVALID • مقدار بازگشتی برای موقعی که مقادیر ویژگی ارائه شده به همراه مقادیر پیش فرض، برای ایجاد شیء کافی نباشد: • CKR_TEMPLATE_INCOMPLETE • مقدار بازگشتی برای موقعی که مقادیر ویژگی ارائه شده به همراه مقادیر پیش فرض، برای ایجاد شیء متناقض باشند: • CKR_TEMPLATE_INCONSISTENT • مقدار بازگشتی برای موقعی که یک ویژگی معین با مقادیر مشابه بیش از یکبار فراخوانی شود: • CKR_TEMPLATE_INCONSISTENT, • در یک نشست فقط-خواندنی، فقط شیءهای نشست را می توان تولید کرد. بنابراین هر تلاشی برای ایجاد سایر شیءها با کد خطای CKR_SESSION_READ_ONLY مواجه می شود. • در نشست عمومی فقط می توان شیءهای عمومی را ایجاد کرد. برای ایجاد شیءهای نشست های کاربر، نیاز به ورود کاربر است. در صورت تلاش برای ایجاد شیءهای نشست های کاربر در نشست عمومی کد خطای CKR_USER_NOT_LOGGED_IN برگشت داده می شود. • در صورتی که برای یک شیء، پرچم CKF_WRITE_PROTECTED با مقدار CK_TRUE تنظیم شده باشد، اعمال این تابع بر روی چنین شیئی با کد خطای CKR_TOKEN_WRITE_PROTECTED مواجه خواهد شد. • در این تابع، تمامی ویژگی های قابل تنظیم را در شی کپی می توان تنظیم کرد. علاوه بر موارد قابل تنظیم، ویژگی های خاص CKA_TOKEN، CKA_PRIVATE، CKA_MODIFIABLE و CKA_DESTROYABLE در فراخوانی این تابع قابل تغییر هستند. • برای کپی یک شیء کلید محرمانه، ویژگی CKA_EXTRACTABLE را از مقدار CK_TRUE به CK_FALSE می توان تغییر داد، ولی برعکس آن

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
	P 5.18.2	**C_GenerateKeyPair**	• این تابع یک شی از جنس زوج کلید عمومی/خصوصی تولید می کند. • از آنجایی که نوع کلیدها در مکانیسم تولید زوج کلید ضمنی هستند، در پارامترهای ورودی های تابع C_GenerateKeyPair نیازی به تعیین نوع کلیدها نیست. اگر نوع کلید تعیین شده در ورودی تابع با مکانیسم تولید کلید ناهماهنگ باشد،فراخوانی تابع C_GenerateKeyPair با کد خطا CKR_TEMPLATE_INCONSISTENT برگشت داده می شود. • فراخوانی C_GenerateKeyPair هرگز یک کلید تنها (کلید عمومی یا خصوصی تنها) ایجاد و بازگشت نمی دهد، بلکه یا هیچ کلیدی ایجاد نمی کند یا یک زوج کلید عمومی/خصوصی معتبر ایجاد و برگشت می دهد. • برای شیءهای کلید ایجاد شده توسط تابع C_GenerateKeyPair، ویژگی CKA_LOCAL روی مقدار CK_TRUE تنظیم خواهد شد. علاوه بر این، شی ایجاد شده یک شناسه یکتا برای CKA_UNIQUE_ID تولید و تخصیص می دهد.

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
	P 5.18.3	C_WrapKey	• این تابع عملیات رمزگذاری یک کلید محرمانه یا خصوصی را انجام می دهد. • این تابع دو کلید را در ورودی دریافت می کند، یکی کلید رمزکننده و دیگری کلید رمز شونده است. کلید رمزکننده باید دارای ویژگی CKA_WRAP تنظیم شده با مقدار CK_TRUE باشد و کلید رمز شونده باید دارای ویژگی CKA_EXTRACTABLE تنظیم شده با مقدار CK_TRUE باشد. • اگر کلید رمز شونده دارای ویژگی CKA_EXTRACTABLE تنظیم شده روی CK_FALSE باشد، آنگاه امکان رمز کردن آن وجود ندارد و کد خطای CKR_KEY_UNEXTRACTABLE برگشت داده می شود. • اگر کلید رمز شونده به دلایل خاص توکن، با وجود تنظیم ویژگی CKA_EXTRACTABLE روی CK_TRUE، رمز نشود، تابع C_WrapKey با کد خطای CKR_KEY_NOT_WRAPPABLE متوقف می شود. • اگر به دلیل طول کلید رمز کننده امکان رمز کردن کلید وجود نداشته باشد، تابع C_WrapKey با کد خطای CKR_KEY_SIZE_RANGE متوقف می شود. • اگر عدم همخوانی بین هر یک از مولفه های ورودی رخ دهد، آنگاه کد خطای CKR_KEY_HANDLE_INVALID برگشت داده می شود. • اگر مکانیزم ناشناخته ای مشخص شده باشد یا مکانیزم مشخص شده در توکن انتخاب شده قابل استفاده نباشد، کد خطای CKR_MECHANISM_INVALID بازگشت داده می شود. • اگر پارامترهای نامعتبر به مکانیزم مشخص شده برای عملیات راستی آزمایی امضا ارائه شود، کد خطای CKR_MECHANISM_PARAM_INVALID برگشت داده می شود. • CKR_WRAPPED_KEY_INVALID: این کد خطا نشان می دهد که کلید رمزشونده معتبر نیست. • اگر شناسه کلید رمزکننده معتبر نباشد کد خطای CKR_WRAPPING_KEY_HANDLE_INVALID برگشت داده می شود. • اگر اندازه کلید رمز کننده خارج از محدوده مجاز باشد کد خطای CKR_WRAPPING_KEY_SIZE_RANGE برگشت داده می شود. • اگر نوع کلید رمزکننده با مکانیسم رمزگذاری سازگار نباشد کد خطای CKR_WRAPPING_KEY_TYPE_INCONSISTENT برگشت داده می شود.

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
	P 5.18.4	C_UnwrapKey	• این تابع عملیات رمزگشایی کلید رمزشده را انجام داده و یک کلید خصوصی یا کلید محرمانه جدید تولید می کند. • ویژگی CKA_UNWRAP برای کلید رمزگشایی کننده باید CK_TRUE باشد، که نشان می دهد این کلید از عملیات رمزگشایی کلید رمزشده پشتیبانی می کند. • کلید جدید تولید شده توسط این تابع دارای ویژگی CKA_ALWAYS_SENSITIVE تنظیم شده روی CK_FALSE و ویژگی CKA_NEVER_EXTRACTABLE تنظیم شده روی CK_FALSE می باشد. مشخصه CKA_EXTRACTABLE به طور پیش فرض روی CK_TRUE تنظیم شده است. • شی کلید ایجاد شده توسط این تابع، ویژگی CKA_LOCAL خود را روی CK_FALSE تنظیم خواهد کرد. علاوه بر این ، شی ایجاد شده یک شناسه یکتا برای CKA_UNIQUE_ID تولید و تخصیص می دهد. • اگر عدم همخوانی در هر یک از مولفه های ورودی رخ دهد، آنگاه کد خطای CKR_TEMPLATE_INCONSISTENT برگشت داده می شود. • اگر شناسه کلید رمزگشایی کننده معتبر نباشد کد خطای CKR_UNWRAPPING_KEY_HANDLE_INVALID برگشت داده می شود. • اگر اندازه کلید رمزگشایی کننده خارج از محدوده مجاز باشد کد خطای CKR_UNWRAPPING_KEY_SIZE_RANGE برگشت داده می شود. • اگر نوع کلید رمزگشایی کننده با مکانیسم رمزگشایی سازگار نباشد کد خطای CKR_UNWRAPPING_KEY_TYPE_INCONSISTENT برگشت داده می شود. • این کد خطا نشان می دهد که اگر کلید رمزشده معتبر نباشد کد خطای CKR_WRAPPED_KEY_INVALID برگشت داده می شود. • اگر اندازه کلید رمزشده خارج از محدوده مجاز باشد کد خطای CKR_WRAPPED_KEY_SIZE_RANGE برگشت داده می شود.

مولفه های مورد ارزیابی			شواهد ارزیابی
دسته	شماره	تابع	
	P 5.18.5	C_DeriveKey	• این تابع عملیات استخراج یک کلید با استفاده از یک کلید پایه را انجام داده و یک شی کلید جدید ایجاد می کند. • مقادیر ویژگی های CKA_SENSITIVE، CKA_ALWAYS_SENSITIVE، CKA_EXTRACTABLE و CKA_NEVER_EXTRACTABLEدر کلید جدید اجاد شده از کلید پایه تاصیر می پذیرند. • شی کلید ایجاد شده توسط این تابع، ویژگی CKA_LOCAL خود را روی CK_FALSE تنظیم خواهد کرد. علاوه بر این ، شی ایجاد شده یک شناسه یکتا برای CKA_UNIQUE_ID تولید و تخصیص می دهد.
توابع مولد اعداد تصادفی	P 5.19.1	C_SeedRandom	• این تابع یک بذر اولیه را با مولد اعداد تصادفی ترکیب می کند. • اگر توکن داری مکانیزم مولد اعداد تصادفی نباشد، کد خطای CKR_RANDOM_NO_RNG برگشت داده می شود. • اگر مکانیزم مولد اعداد تصادفی توکن بذر اولیه (Seed) را از بیرون توکن قبول نکند، کد خطای CKR_RANDOM_SEED_NOT_SUPPORTED برگشت داده می شود. این مقدار بازگشتی نسبت به CKR_RANDOM_NO_RNG دارای اولویت کمتری است.
	P 5.19.2	C_GenerateRandom	• این تابع عملیات تولید داده تصادفی یا شبه تصادفی را انجام می دهد. • اگر توکن داری مکانیزم مولد اعداد تصادفی نباشد، کد خطای CKR_RANDOM_NO_RNG برگشت داده می شود.
توابع مدیریت عملکرد موازی	P 5.20.1	C_GetFunctionStatus	• این توابع مربوط به نسخه های قدیمی Cryptoki می باشد و در نسخه فعلی استفاده نمی شود . فراخوانی این توابع همیشه کد خطای CKR_FUNCTION_NOT_PARALLEL را برمی گرداند.
	P 5.20.2	C_CancelFunction	