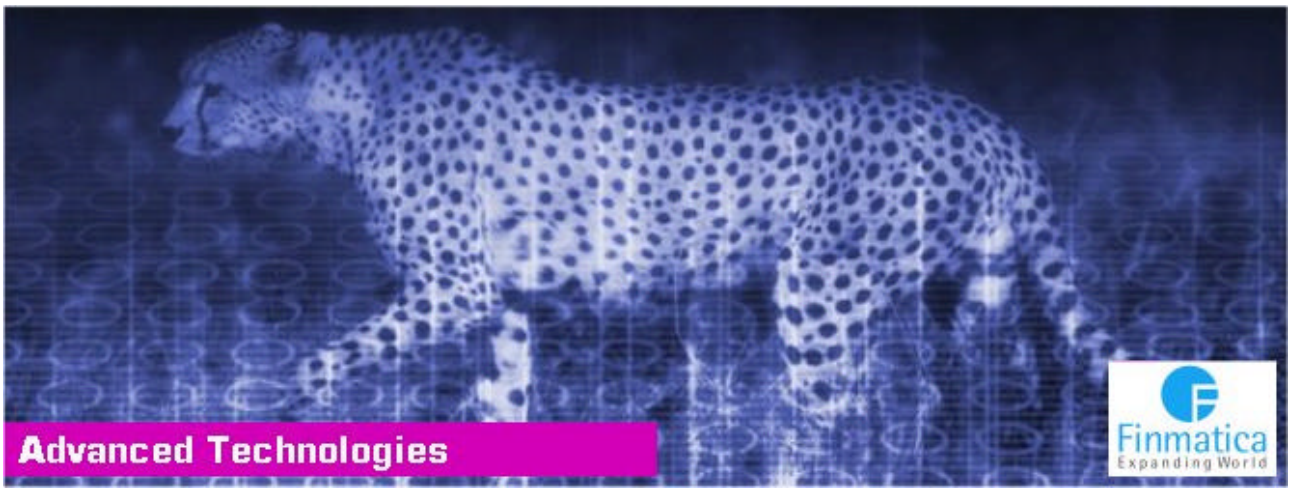
Security BOX® *Suite*

Cible de sécurité Critères Communs

Niveau EAL4+

Security BOX® *Crypto* 6.0

Version 2.6 du 09/01/2004



Méthode et Solution Informatique S.A.

7, rue Jean Mermoz - 78000 Versailles - France

www.securitybox.net

Suivi des modifications

Version	Date	Commentaires
1.0	22/10/2002	Création du document
2.0	06/01/2003	Mise à jour du périmètre d'évaluation
2.1	09/10/2003	Mise à jour de la présentation de la cible
2.2	13/10/2003	Ajout de la librairie p11msidll.lib dans la liste des binaires constituant la cible
2.3	13/11/2003	Révision des chapitres 1, 2, 5 et 6
2.4	25/11/2003	Corrections diverses - Mise en forme
2.5	28/11/2003	Modification de la version finale des binaires
2.6	09/01/2004	Modification de la version finale des binaires

Reproduction, copie, prêt, diffusion interdits sans accord préalable.

Table des matières

Reproduction, copie, prêt, diffusion interdits sans accord préalable.

1. Introduction de la cible de sécurité

1.1. Identification de la cible de sécurité

Cible d'évaluation (TOE) :	Librairie **Security BOX® *Crypto* 6.0** de la gamme de produits **Security BOX® *Suite*** pour les plates-formes PC sous Microsoft Windows 95/98/Me/NT/2000/XP
Niveau EAL :	**EAL4 augmenté de AVA_VLA.3**
Résistance des fonctions :	**SOF-élevé**
Conformité à un PP existant :	Aucune.
Référence des CC :	ISO/IEC 15408:1999(E), Parts 1 to 3 - Décembre 1999.

1.2. Vue d'ensemble de la cible de sécurité

Security BOX® *Crypto* est le module cryptographique utilisé par tous les produits de la gamme **Security BOX® *Suite***.

Il assure :

- l'exécution de tous les calculs cryptographiques nécessaires aux produits de la gamme Security BOX® (génération de clés, dérivation de clés, calcul de compressé, signature, vérification de signature, cryptage de données, décryptage de données, cryptage de clé [wrapping], décryptage de clé [unwrapping], en mettant en oeuvre des algorithmes et des mécanismes standards et publiés ;
- le stockage sécurisé d'objets appartenant à un utilisateur, ces objets pouvant être des clés (secrètes, privées ou publiques), des certificats, ou des objets de données (i.e sans caractère cryptographique particulier, mais pouvant être confidentiels ou dont il faut assurer l'intégrité).

  Cette gestion implique des règles d'authentification et de droits d'accès en lecture et/ou écriture. Ce stockage est effectué dans un fichier, appelé "coffre-fort" (ou "keystore"), avec un format privé et un schéma de sécurité faisant partie intégrante de la cible évaluée.

  Ce stockage peut également être effectué en collaboration (pour partie des objets) avec un dispositif externe (carte à puce, token USB ou autre), accédé au travers d'une interface PKCS#11. Dans ce cas, ce dispositif et les logiciels associés ne font pas l'objet de l'évaluation, mais le mode d'utilisation de ce dispositif, ainsi que l'authentification conjointe en font partie.

La cible est évaluée pour une plate-forme PC sous les systèmes d'exploitation Microsoft Windows 95/98/Me/NT/2000/XP.

Reproduction, copie, prêt, diffusion interdits sans accord préalable.

1.3. Conformité aux Critères Communs

Cette cible de sécurité respecte les exigences des Critères Communs ISO/IEC 15408:1999(E) de Décembre 1999.

Tous les composants fonctionnels décrits dans cette cible de sécurité sont issus de la Partie 2 des Critères Communs ISO/IEC 15408:1999(E) de Décembre 1999.

Le niveau d'assurance **EAL4+** retenu est conforme à la Partie 3 de Critères Communs ISO/IEC 15408:1999(E) de Décembre 1999.

Aucune interprétation des Critères Communs n'est retenue.

Reproduction, copie, prêt, diffusion interdits sans accord préalable.

2. Description de la cible d'évaluation (TOE)

2.1. Présentation de la gamme de produits Security BOX®

Security BOX® ***Suite*** est une suite logicielle destinée à couvrir l'ensemble des besoins en terme de sécurité du contenu sur le poste de travail. Cette suite regroupe plusieurs modules qui peuvent être achetés et installés ensemble ou séparément. Elle comprend :

- **Security BOX®** ***File*** permet le cryptage des fichiers. Cette application peut être complétée par une ou plusieurs de ses extensions :
  - **Security BOX®** ***Disk*** pour le chiffrement de disques virtuels ;
  - **Security BOX®** ***Shredder*** pour l'effacement irréversible des données ;
  - **Security BOX®** ***Sign*** pour la signature électronique de fichiers et de dossiers ;
- **Security BOX®** ***Mail*** pour le cryptage et la signature de courriers électroniques,
- **Security BOX®** ***VPN*** permet l'échange sécurisé des informations sur un réseau public (Internet) ou un réseau d'entreprise (LAN) . Il répond par exemple aux besoins des employés travaillant à distance, des succursales en relation avec leur siège, des stations connectées à des réseaux sans fil....
- **Security BOX®** ***Card Extension*****,** permet l'utilisation de cartes à microprocesseurs et clés USB pour le stockage physique des clés publiques/privées des utilisateurs.
- **Security BOX®** ***Browser Extension*****,** permet l'authentification SSL/TLS à l'aide de la clé de signature du compte utilisateur **Security BOX®.**
- **Security BOX®** ***Manager***, permet l'administration de la gamme des produits **Security BOX®** ***Suite.*** Il assure la gestion et le paramétrage des comptes utilisateurs et de leurs clés cryptographiques.

2.2. Périmètre de la TOE

Le périmètre d'évaluation est la librairie **Security BOX®** ***Crypto***, qui est le module cryptographique utilisé par tous les produits de la gamme **Security BOX®**.

Ce module est une librairie conforme au standard Pkcs#11 qui assure :

- le tirage et le stockage des clés privées de l'utilisateur ;
- le stockage des données sensibles de l'utilisateur ;
- le tirage d'aléa (clés de session) ;
- les calculs cryptographiques (cryptage/décryptage de données ou de clé, signature, vérification de signature, compressé).

Les données sensibles de l'utilisateur sont stockées dans un fichier dit "coffre fort individuel", dont l'ouverture nécessite l'authentification préalable de l'utilisateur.

Reproduction, copie, prêt, diffusion interdits sans accord préalable.

Deux modes d'authentification sont possibles :

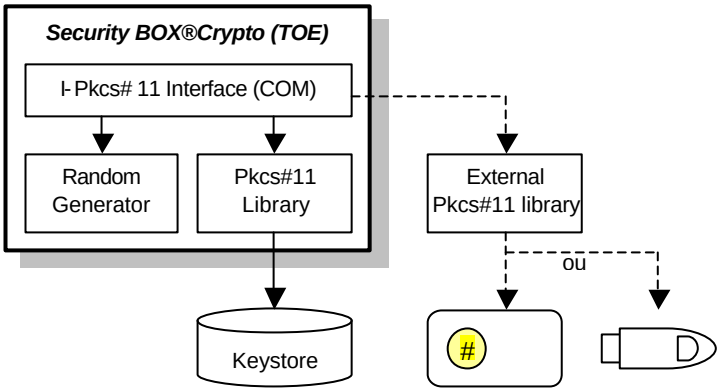
- sans dispositif matériel (mode "mot de passe") :
  - l'utilisateur est authentifié avec un identifiant et un mot de passe (login/password) ;
  - les clés privées de l'utilisateur sont stockées dans le "coffre fort" ;
  - les calculs à clé privée sont effectués de façon logicielle ;
- avec un dispositif matériel (mode "carte ou clé USB") :
  - l'utilisateur est authentifié avec une carte à puce (ou une clé USB) et un code "PIN" ;
  - les clés privées sont stockées dans la carte ;
  - les calculs à clé privée sont effectués par la carte.

L'évaluation porte sur :

- le fonctionnement complet de la librairie quand elle est utilisée en mode "mot de passe" ;
- la mise en œuvre du dispositif cryptographique matériel quand elle est utilisée en mode "carte ou clé USB" (le dispositif lui-même ne fait l'objet de cette évaluation).

Cette librairie **Security BOX®** ***Crypto*** est évaluée, en tant que produit, sur une plate-forme PC sous les systèmes d'exploitation de Microsoft suivants : Windows 95, Windows 98, Windows Millenium, Windows NT, Windows 2000 et Windows XP.

2.2.1. Architecture interne



Le composant '**Pkcs#11 Library**" implémente le standard Pkcs#11 (API, gestion de slots, de tokens, de sessions, d'objets et de mécanismes) et la gestion du keystore (il implémente donc le schéma de sécurité du keystore).

Le composant "**I-Pkcs#11 Interface**" enveloppe le précédent sous une vision "objet" (C++/COM), et prend en charge le pilotage d'un éventuel module Pkcs#11 externe (carte ou clé USB), en donnant l'impression à l'applicatif que les deux ne forment qu'un seul et même slot et token.

Le composant "**Random generator**" produit les aléas utilisés par les services cryptographiques. Il s'appuie sur plusieurs sources de bruit au sein de la plate-forme PC (clavier, souris) pour assurer une bonne qualité de ces aléas.

2.2.2. Composants Binaires

Security BOX® *Crypto* est constitué des trois binaires suivants :

Nom	Version	Description
sbp11.dll	6.0.2.0	intègre la librairie Pkcs#11 logicielle, les fonctions d'accès au keystore et le générateur d'aléa.
sbp11ka.dll	6.0.2.0	nécessaire uniquement en mode "carte ou clé USB" : assure l'interface avec la librairie Pkcs#11 de la carte utilisée.
p11msidll.lib	6.0.2.0	librairie Pkcs#11 logicielle, incluse dans **sbp11.dll** et dans certains composants de Security BOX® Suite.

2.3. Les biens sensibles

2.3.1. Biens sensibles de l'utilisateur

Tout d'abord, la librairie Security BOX® Crypto gère l'accès à un « coffre-fort » individuel logiciel par authentification de l'utilisateur (mot de passe ou code PIN lors de l'utilisation d'une carte à puce ou un jeton USB). Les biens sensibles associés sont donc le mot de passe ou le code PIN de l'utilisateur (et de l'officier de sécurité, conformément à la norme PKCS#11).

Ensuite, la librairie Security BOX® Crypto permet aux autres produits Security BOX® de protéger les données des utilisateurs, soit directement sur le support de stockage ou soit lors de leur échange via un réseau (messagerie ou tunnel chiffrant VPN). Les biens sensibles sont donc les données utilisateur manipulées par la librairie Security BOX® Crypto (fichiers, répertoires, messages électroniques…), les bi-clés de signature et de chiffrement des utilisateurs et les clés de chiffrement utilisées (clés de session) lorsqu'elles sont reçues ou envoyées par la librairie Security BOX® Crypto.

2.3.2. Biens sensibles de la TOE

Les fichiers informatiques constituant la librairie Security BOX® Crypto, ainsi que les fichiers contenant un « coffre-fort » individuel, sont considérés comme sensibles.

Pour le contrôle d'accès aux «coffres-forts » individuels logiciels, la librairie Security BOX® Crypto utilise des clés de chiffrement et de scellement internes propres à un « coffre-fort » individuel et des attributs PKCS#11 associés à chacun des objets stockés dans un « coffre-fort » individuel.

Pour le besoin des autres produits Security BOX®, la librairie Security BOX® Crypto gère les attributs PKCS#11 associés aux bi-clés de signature et de chiffrement des utilisateurs.

Reproduction, copie, prêt, diffusion interdits sans accord préalable.

2.3.3. Synthèse des biens sensibles

Le tableau ci-dessous résume la liste des biens sensibles protégés par la librairie Security BOX® Crypto et indique la nature de la sensibilité associée.

Biens sensibles	Confidentialité	Intégrité
Biens sensibles de l'utilisateur		
Mot de passe ou code PIN de l'utilisateur	Forte	Forte
Mot de passe ou code PIN de l'officier de sécurité	Forte	Forte
Données utilisateur manipulées	Forte	Forte
Clés publiques de signature et de chiffrement de l'utilisateur	*Aucune*	Forte
Clés privées de signature et de chiffrement de l'utilisateur	Forte	Forte
Clés de chiffrement (ou de session)	Forte	Forte
Biens sensibles de la TOE		
Fichier contenant le « coffre-fort » individuel de l'utilisateur	*Faible*	Forte
Fichiers de la librairie Security BOX® Crypto	*Faible*	Forte
Les clés de chiffrement et de scellement internes propres à un « coffre-fort » individuel	*Faible*	Forte
Attributs PKCS#11 associés aux objets du « coffre-fort » individuel	*Faible*	Forte

Reproduction, copie, prêt, diffusion interdits sans accord préalable.

3. Environnement de sécurité de la TOE

3.1. Hypothèses

Pour la librairie Security BOX® Crypto, nommée la TOE dans les paragraphes suivants, les hypothèses suivantes sur l'environnement d'utilisation seront prises en compte pour l'évaluation du niveau de confiance offert aux utilisateurs :

H.NON_OBSERV	L'environnement physique de la TOE permet aux utilisateurs d'entrer leur mot de passe (ou code PIN) sans être observable directement ou interceptable (clavier sans fil…) par d'autres utilisateurs ou attaquants potentiels.
H.NON_PIEG	Le PC de l'utilisateur supportant la TOE est considéré comme sain, c'est-à-dire qu'il ne comporte aucun dispositif matériel ou logiciel (sonde, piège, espion de clavier…) permettant d'accéder aux biens sensibles protégés par la TOE lorsqu'une session utilisateur est ouverte, en contournant les fonctions de sécurité de la TOE (lecture directe de la mémoire, interruption des processus de la TOE,…).

3.2. Menaces

La librairie Security BOX® Crypto, de part sa diversité de services offerts aux autres produits Security BOX®, permet de couvrir des menaces très variées. L'attaquant considéré a un profil stratégique, c'est-à-dire qu'il dispose de bonnes compétences (en informatique et en cryptographie), de ressources modérées (plusieurs hommes.mois) et d'une bonne motivation, Ses méthodes d'attaque vont de l'exploitation d'une vulnérabilité connue d'un algorithme cryptographique aux attaques statistiques, « force brute » (essais exhaustifs) et « clair connu » (rejeu ou substitution).

M.ACCES	Un attaquant consulte de manière non autorisée les données sensibles d'un utilisateur protégées par la TOE, en exécutant la TOE directement (appel à la librairie constituant la TOE).
M.USURP_LOGIN	Un attaquant désirant accéder aux données sensibles d'un utilisateur tente d'ouvrir une session en essayant systématiquement tous les mots de passe possibles (attaque par force brute ou par dictionnaire).
M.VOL_COFFRE	Un attaquant récupère sur le PC d'un utilisateur le fichier contenant son « coffre-fort » individuel (fichier .usr) et essaie toutes les attaques cryptographiques connues (attaque par force brute, clair connu, clair choisi ou par dictionnaire) pour accéder aux données sensibles de cet utilisateur.

Reproduction, copie, prêt, diffusion interdits sans accord préalable.

M.MODIF_SBOX	Un attaquant va modifier la TOE pour y insérer une fonction cachée de recouvrement, ce qui lui permettra d'accéder aux données sensibles d'un utilisateur.
M.ABSENCE_TEMP	Profitant de l'absence temporaire de l'utilisateur ayant ouvert une session avec la TOE, un attaquant accède à son PC et récupère les données sensibles de ce dernier protégées par la TOE (exportation des bi-clés, déchiffrement de fichiers protégés,…).
M.ARRET_PC	Un arrêt brutal du PC (coupure d'alimentation, plantage système...) empêche la fermeture « propre » de la session de l'utilisateur connecté, ce qui permet, après le redémarrage du PC, l'accès à certaines données sensibles qui ne sont plus protégées par la TOE.
M.OUBLI_LOGIN	Un utilisateur autorisé oublie son mot de passe (ou son code PIN), ce qui lui interdit l'accès à ses données sensibles protégées par la TOE.
M.AUTRE_UTIL	Un utilisateur malveillant possédant également un « coffre-fort » individuel sur le même PC ouvre une session avec la TOE pour accéder à son coffre-fort et profite de cette authentification réussie pour accéder aux données sensibles du « coffre-fort » individuel d'un autre utilisateur de ce PC.
M.RESIDUS	Un attaquant désirant accéder aux données sensibles d'un utilisateur contenues dans son « coffre-fort » individuel récupère, sur le PC de celui-ci et après la fermeture de la session par la TOE, des traces, dans la mémoire (RAM) ou sur le disque dur, contenant des fragments de données sensibles protégées par la TOE.
M.PLANTAGE	Un arrêt brutal du PC (coupure d'alimentation, plantage système...) pendant le chiffrement ou le déchiffrement d'un fichier d'un utilisateur connecté rend ce fichier inutilisable et irrécupérable par la TOE.
M.DESTRUCTION	Un attaquant désirant empêcher l'accès d'un utilisateur à ses données sensibles accède à son PC et détruit ou corrompt le fichier contenant son « coffre-fort » individuel (fichier .usr) utilisé par la TOE.
M.MODIF_COFFRE	Un attaquant réussit de manière non autorisée à ajouter ou modifier un objet, ou l'attribut PKCS#11 d'un objet, contenu dans le « coffre-fort » individuel d'un utilisateur et protégé par la TOE.
M.ALEAS	La génération des aléas réalisée par la TOE n'est pas assez aléatoire et peut être prévisible lors de la création des clés. Un attaquant connaissant cette faiblesse accède aux données sensibles des autres utilisateurs protégés par la TOE.

3.3. Politiques de sécurité organisationnelles

Aucune règle relative à la politique de sécurité organisationnelle à laquelle la TOE ne doive se conformer n'est décrite.

Reproduction, copie, prêt, diffusion interdits sans accord préalable.

4. Objectifs de sécurité

4.1. Objectifs de sécurité pour la TOE

O.CTL_ACCES	La TOE ne doit autoriser l'accès (lecture, écriture, suppression…) d'une donnée sensible contenue dans le « coffre-fort » individuel d'un utilisateur qu'après présentation du mot de passe associé (identification et authentification réussies) et vérification des attributs PKCS#11 associés à cette donnée.
O.ALGO_STD	La TOE doit fournir un choix d'algorithmes cryptographiques et de tailles de clés conformes à l'état de l'art et aux standards de ce domaine.
O.INTERF_TOKEN	La TOE doit autoriser l'identification et l'authentification d'un utilisateur par la présentation d'une carte à puce ou d'un jeton USB et la saisie d'un code PIN.
O.VERROUILLAGE	La TOE doit permettre le verrouillage du « coffre-fort » individuel d'un utilisateur connecté.
O.CONFIDENTIALITE	La TOE doit assurer la confidentialité des données sensibles des utilisateurs contenues dans leur « coffre-fort » individuel respectif, même suite à un arrêt brutal du PC (coupure d'alimentation, plantage système...) qui a empêché la fermeture « propre » de la session des utilisateurs connectés.
O.ROLES	La TOE doit gérer deux profils d'utilisateur : l'utilisateur du « coffre-fort » individuel et l'administrateur local (ou officier de sécurité).
O.REINIT_LOGIN	La TOE doit offrir un mécanisme de secours qui permet à un utilisateur ayant oublier son mot de passe (ou son code PIN) de réinitialiser celui-ci en s'authentifiant en tant qu'officier de sécurité associé à son "coffre-fort individuel".
O.GEST_SECRETS	La TOE doit utiliser des secrets différents pour protéger les "coffres-forts individuels" qu'elle gère.
O.EFF_RESIDUS	La TOE doit assurer le nettoyage par réécritures successives (et non pas seulement leur suppression) des fichiers temporaires sur le disque dur du PC ou des zones de mémoires allouées, dès la fin de traitement de données sensibles.
O.INTEGRITE	La TOE doit assurer l'intégrité des données sensibles des utilisateurs connectés lors des traitements qu'elle réalise (par exemple le chiffrement ou le déchiffrement d'un fichier), même suite à un arrêt brutal du PC (coupure d'alimentation, plantage système...).

Reproduction, copie, prêt, diffusion interdits sans accord préalable.

O.INTEG_COFFRE	La TOE doit assurer l'intégrité des « coffres-forts » individuel des utilisateurs.
O.ALEAS	La TOE doit offrir un mécanisme de génération de pseudo-aléas ou d'aléas vrais.

4.2. Objectifs de sécurité pour l'environnement

OE.NON_OBSERV	L'environnement physique de la TOE doit permettre aux utilisateurs d'entrer leur mot de passe (ou code PIN) sans être observable directement ou interceptable (clavier sans fil,…) par d'autres utilisateurs ou attaquants potentiels.
OE.NON_PIEG	Le PC de l'utilisateur doit être périodiquement inspecté physiquement et doit posséder un logiciel anti-virus à jour.
OE.ALGO_UTILISES	Les applications s'appuyant sur la TOE doivent sélectionner des algorithmes cryptographiques pour lesquels aucune vulnérabilité n'est connue et des tailles de clés suffisamment longues pour rendre impossible toute attaque par force brute (essais exhaustifs de toutes les clés possibles).
OE.ROBUST_MDP	Les applications s'appuyant sur la TOE doivent proposer à l'utilisateur une aide pour la création de mots de passe, lui permettant d'estimer la robustesse de son mot de passe se basant sur sa longueur, sa non trivialité, le nombre de caractères alphanumériques et spéciaux qu'il contient.
OE.RENOUV_MDP	L'utilisateur doit s'assurer de la non divulgation de son mot de passe (ou code PIN) et de son renouvellement périodique.
OE.LOGIN_OFFICIER	L'utilisateur doit conserver le mot de passe de l'officier de sécurité associé à son "coffre-fort individuel" dans une enveloppe scellée placée dans une armoire fermant à clé.
OE.VERIF_INTEGRITE	Les applications s'appuyant sur la TOE doivent proposer à l'utilisateur la possibilité de vérifier l'intégrité de la TOE, par exemple par l'utilisation d'un outil de contrôle de scellement (type MD5).
OE.VERR_AUTO	Les applications s'appuyant sur la TOE doivent fournir un mécanisme permettant de déclencher le verrouillage par la TOE du « coffre-fort » individuel d'un utilisateur connecté, en cas d'inactivité prolongée dont la durée est paramétrable par l'utilisateur et/ou en cas de lancement du verrouillage d'écran du PC.
OE.SAUVEGARDE	L'utilisateur doit sauvegarder à chaque modification le fichier contenant son "coffre-fort individuel" (fichier .usr) et placer le support de sauvegarde dans une armoire fermant à clé.
OE.USAGE_ALEAS	L'utilisateur doit générer suffisamment de bruit lors de la création des aléas (mouvement de la souris, frappe de touches au clavier) pour assurer une bonne qualité d'aléas.

5. Exigences de sécurité des Technologies de l'Information

5.1. Exigences de sécurité pour la TOE

5.1.1. Exigences de sécurité fonctionnelles pour la TOE

Les composants fonctionnels CC sélectionnés pour répondre aux objectifs de sécurité de la TOE sont les suivants :

Composants CC retenus	
FCS_CKM.1	Génération de clés cryptographiques
FCS_CKM.3	Accès aux clés cryptographiques
FCS_CKM.4	Destruction de clés cryptographiques
FCS_COP.1	Opération cryptographique
FDP_ACC.2	Contrôle d'accès complet
FDP_ACF.1	Contrôle d'accès basé sur les attributs de sécurité
FDP_ETC.1	Exportation de données de l'utilisateur sans attributs de sécurité
FDP_ITC.1	Importation de données de l'utilisateur sans attributs de sécurité
FDP_RIP.2	Protection totale des informations résiduelles
FDP_SDI.2	Contrôle de l'intégrité des données stockées et action à entreprendre
FIA_UAU.2	Authentification d'un utilisateur préalablement à toute action
FIA_UID.2	Identification d'un utilisateur préalablement à toute action
FMT_MSA.1	Gestion des attributs de sécurité
FMT_MSA.2	Attributs de sécurité sûrs
FMT_MSA.3	Initialisation statique d'attribut
FMT_SMR.1	Rôles de sécurité
FPT_FLS.1	Défaillance avec préservation d'un état sûr
FTA_SSL.2	Verrouillage d'une session, initié par l'utilisateur

Reproduction, copie, prêt, diffusion interdits sans accord préalable.

FCS_CKM.1 - Génération de clés cryptographiques

La TSF doit générer les clés cryptographiques conformément à un algorithme de génération de clés cryptographiques spécifié [spécification: algorithme de génération de clés cryptographiques] **génération de nombres pseudo-aléatoires et d'exposants Diffie-Hellman** et à des tailles de clés cryptographiques spécifiées [spécification: tailles des clés cryptographiques] **de 40 à 256 bits pour les algorithmes symétriques et de 512 à 4096 bits pour les algorithmes asymétriques** qui satisfont à ce qui suit: [spécification: liste des normes] **PKCS#11 v2.11 de RSA Laboratories**

FCS_CKM.3 - Accès aux clés cryptographiques

La TSF doit réaliser [spécification : type d'accès aux clés cryptographiques] **l'utilisation, l'importation ou l'exportation de clés** conformément à une méthode d'accès aux clés cryptographiques spécifiée [spécification: méthode d'accès aux clés cryptographiques] **par contrôles des attributs de clés** qui satisfait à ce qui suit: [spécification: liste des normes] **PKCS#11 v2.11 de RSA Laboratories**.

FCS_CKM.4 - Destruction de clés cryptographiques

La TSF doit détruire les clés cryptographiques conformément à une méthode de destruction spécifiée de clés cryptographiques [spécification: méthode de destruction de clés cryptographiques] **par réécriture de motifs aléatoires** qui satisfait à ce qui suit: [spécification: liste des normes] **PKCS#11 v2.11 de RSA Laboratories**

FCS_COP.1 - Opération cryptographique

La TSF doit exécuter [spécification : liste des opérations cryptographiques] **le hashage, le chiffrement, le déchiffrement, la signature, la vérification de signature, la génération de clés, le wrapping de clés et la dérivation de clés** conformément à un algorithme cryptographique spécifié [spécification: algorithme cryptographique] **RSA, DSA, DH, RC2, RC4, RC5, DES, 3DES, MD2, MD5, SHA1, RIPEMB-160 et AES** et avec des tailles de clés cryptographiques [spécification : tailles de clés cryptographiques] **de 40 à 256 bits pour les algorithmes symétriques et de 512 à 4096 bits pour les algorithmes asymétriques** qui satisfont à ce qui suit: [spécification: liste des normes] **PKCS#11 v2.11 de RSA Laboratories**.

FDP_ACC.2 - Contrôle d'accès complet

La TSF doit appliquer la [spécification : SFP de contrôle d'accès] **politique ACCESS_OBJ** aux [spécification: liste des sujets et objets] **objets protégés par la TOE dans un «coffre-fort» individuel** et toutes les opérations sur les sujets et objets couverts par la SFP.

La TSF doit garantir que toutes les opérations entre tout sujet du TSC et tout objet du TSC sont couvertes par une SFP de contrôle d'accès.

FDP_ACF.1 - Contrôle d'accès basé sur les attributs de sécurité

La TSF doit appliquer la [spécification: SFP de contrôle d'accès] **politique ACCESS_OBJ** aux objets en fonction des [spécification: attributs de sécurité, groupes d'attributs de sécurité cités] **attributs de sécurité décrits dans le document PKCS#11 v2.11 de RSA Laboratories**

La TSF doit appliquer les règles suivantes pour déterminer si une opération entre des sujets contrôlés et des objets contrôlés est autorisée: [spécification: règles qui régissent les accès aux sujets contrôlés et aux objets contrôlés utilisant des opérations contrôlées sur des objets

Reproduction, copie, prêt, diffusion interdits sans accord préalable.

contrôlés] **règles décrites dans le document PKCS#11 v2.11 de RSA Laboratories**

La TSF doit autoriser explicitement l'accès de sujets à des objets en fonction des règles complémentaires suivantes : [spécification : règles basées sur les attributs de sécurité, qui autorisent explicitement l'accès de sujets à des objets] **règles décrites dans le document PKCS#11 v2.11 de RSA Laboratories**.

La TSF doit refuser explicitement l'accès de sujets à des objets en fonction de [spécification : règles basées sur les attributs de sécurité, qui interdisent explicitement l'accès de sujets à des objets] **règles décrites dans le document PKCS#11 v2.11 de RSA Laboratories**.

FDP_ETC.1 - Exportation de données de l'utilisateur sans attributs de sécurité

La TSF doit exporter les données de l'utilisateur sans les attributs de sécurité associés aux données de l'utilisateur.

La TSF doit appliquer la [spécification: les SFP de contrôle d'accès ou les SFP de contrôle de flux d'informations] **politique ACCESS_OBJ** lors de l'exportation de données de l'utilisateur, contrôlées par la ou les SFP, vers l'extérieur du TSC.

FDP_ITC.1 - Importation de données de l'utilisateur sans attributs de sécurité

La TSF doit appliquer les règles suivantes lors de l'importation de données de l'utilisateur contrôlées par la SFP en provenance de l'extérieur du TSC: [spécification: règles complémentaires de contrôle d'importation] **règles décrites dans le document PKCS#11 v2.11 de RSA Laboratories**.

La TSF doit appliquer la [spécification : SFP de contrôle d'accès ou SFP de contrôle de flux d'informations] **politique ACCESS_OBJ** lors de l'importation de données de l'utilisateur contrôlées par la SFP en provenance de l'extérieur du TSC.

La TSF doit ignorer tout attribut de sécurité associé aux données de l'utilisateur lorsqu'elles sont importées depuis l'extérieur du TSC.

FDP_RIP.2 - Protection totale des informations résiduelles

La TSF doit garantir que toute information contenue précédemment dans une ressource est rendue inaccessible lors de [sélection: l'allocation de la ressource à, désallocation de la ressource de] **désallocation de la ressource des** tous les objets.

FDP_SDI.2 - Contrôle de l'intégrité des données stockées et action à entreprendre

La TSF doit contrôler les données de l'utilisateur stockées au sein du TSC à la recherche des [spécification : erreurs d'intégrité] **erreurs d'intégrité** sur tous les objets, en fonction des attributs suivants [spécification: attributs des données de l'utilisateur] **attributs de sécurité « private » et « sensitive » décrits dans le document PKCS#11 v2.11 de RSA Laboratories**

En cas de détection d'une erreur d'intégrité, la TSF doit [spécification: action à entreprendre] **retourner un code d'erreur**.

FIA_UAU.2 - Authentification d'un utilisateur préalablement à toute action

La TSF doit exiger que chaque utilisateur soit authentifié avec succès avant d'autoriser toute autre action transitant par la TSF pour le compte de cet utilisateur.

Niveau minimum de résistance : **SOF-élevé**

FIA_UID.2 - Identification d'un utilisateur préalablement à toute action

La TSF doit exiger que chaque utilisateur soit identifié avec succès avant d'autoriser toute autre action transitant par la TSF pour le compte de cet utilisateur.

FMT_MSA.1 - Gestion des attributs de sécurité

La TSF doit mettre en œuvre une [spécification: SFP de contrôle d'accès, SFP de contrôle des flux d'information] **politique ACCESS_ROLES** pour restreindre aux [spécification : les rôles autorisés identifiés] **rôles USER et SO** la possibilité de [sélection: changer la valeur par défaut, interroger, modifier, supprimer, [spécification: autres opérations]] **changer la valeur par défaut, modifier ou supprimer** les attributs de sécurité [spécification: liste des attributs de sécurité] **attributs de sécurité décrits dans le document PKCS#11 v2.11 de RSA Laboratories**

FMT_MSA.2 - Attributs de sécurité sûrs

La TSF doit garantir que seules des valeurs sûres sont acceptées pour les attributs de sécurité.

FMT_MSA.3 - Initialisation statique d'attribut

La TSF doit permettre aux [spécification : les rôles autorisés identifiés] **rôles USER et SO** de spécifier des valeurs initiales alternatives pour remplacer les valeurs par défaut lorsqu'un objet ou une information est créé.

La TSF doit mettre en œuvre [spécification: SFP de contrôle d'accès, SFP de contrôle des flux d'information] **politique ACCESS_ROLES** afin de fournir des valeurs par défaut [sélection : restrictives, permissives, autres propriétés] **restrictives** pour les attributs de sécurité qui sont utilisés pour appliquer la SFP.

FMT_SMR.1 - Rôles de sécurité

La TSF doit tenir à jour les rôles [spécification: les rôles autorisés identifiés] **USER et SO décrits dans le document PKCS#11 v2.11 de RSA Laboratories**.

La TSF doit être capable d'associer les utilisateurs aux rôles.

FPT_FLS.1 - Défaillance avec préservation d'un état sûr

La TSF doit préserver un état sûr quand les types de défaillances suivants se produisent : spécification: liste des types de défaillances de la TSF] **arrêt brutal du PC (plantage ou coupure de courant)**.

FTA_SSL.2 - Verrouillage d'une session, initié par l'utilisateur

La TSF doit autoriser le verrouillage initié par l'utilisateur de sa propre session interactive:

a) en effaçant ou en écrasant le contenu des écrans d'affichage, les rendant ainsi illisibles;

b) en désactivant tout moyen d'accès aux données de l'utilisateur ou d'affichage de celles-ci, autrement qu'en déverrouillant la session.

La TSF doit exiger que les événements suivants interviennent avant le déverrouillage de la session: [spécification: événements devant se produire] **identification et authentification réussie de l'utilisateur connecté ou du SO**.

5.1.2. Exigences de sécurité d'assurance pour la TOE

Comme indiqué au paragraphe 3.2 du présent document, l'attaquant considéré a un profil stratégique, c'est-à-dire qu'il dispose de bonnes compétences (en informatique et en cryptographie), de ressources modérées (plusieurs hommes.mois) et d'une bonne motivation. La TOE doit donc être résistante aux attaques de pénétration effectuées par un attaquant ayant un potentiel d'attaque moyen.

Le niveau d'assurance visé par la TOE est le niveau :

EAL4 augmenté de AVA_VLA.3 avec une résistance SOF-élevé

Ce qui correspond à la sélection des composants d'assurance CC suivants :

Composants CC retenus	
ACM_AUT.1	Automatisation partielle de la CM
ACM_CAP.4	Aide à la génération et procédures de réception
ACM_SCP.2	Couverture du suivi des problèmes par la CM
ADO_DEL.2	Détection de modifications
ADO_IGS.1	Procédures d'installation, de génération et de démarrage
ADV_FSP.2	Définition exhaustive des interfaces externes
ADV_HLD.2	Conception de haut niveau de sécurité
ADV_IMP.1	Sous-ensemble de l'implémentation de la TSF
ADV_LLD.1	Conception de bas niveau descriptive
ADV_RCR.1	Démonstration de correspondance informelle
ADV_SPM.1	Modèle informel de politique de sécurité de la TOE
AGD_ADM.1	Guide de l'administrateur
AGD_USR.1	Guide de l'utilisateur
ALC_DVS.1	Identification des mesures de sécurité
ALC_LCD.1	Modèle de cycle de vie défini par le développeur
ALC_TAT.1	Outils de développement bien définis
ATE_COV.2	Analyse de la couverture
ATE_DPT.1	Tests: conception de haut niveau
ATE_FUN.1	Tests fonctionnels
ATE_IND.2	Tests indépendants - par échantillonnage
AVA_MSU.2	Validation de l'analyse
AVA_SOF.1	Evaluation de la résistance des fonctions de sécurité de la TOE
AVA_VLA.3	Résistance moyenne

Ce niveau d'assurance respecte les dépendances entre les composants d'assurance CC mentionnés dans la Partie 3 des Critères Communs.

Reproduction, copie, prêt, diffusion interdits sans accord préalable.

5.2. Exigences de sécurité pour l'environnement des TI

Aucune exigence particulière pour la TOE.

Reproduction, copie, prêt, diffusion interdits sans accord préalable.

6. Spécifications globales de la TOE

6.1. Fonctions de sécurité de la TOE

Les fonctions de sécurité de la TOE implémentées pour répondre aux composants fonctionnels CC sélectionnés pour la TOE sont les suivantes (leur spécification est conforme au standard PKCS#11 v2.11, à l'exception de quelques fonctions privées dont la spécification est indiquée ci-dessous) :

Fonctions privées

OpenKeyStore	Ouverture d'un « coffre-fort » individuel (en lecture seule ou lecture/écriture) pour consulter les données publiques.
OpenCardKeyStore	Ouverture d'un « coffre-fort » individuel associé à une carte à puce ou un token.
CloseKeyStore	Fermeture d'un « coffre-fort » individuel et des sessions cryptographiques associées.
CreateKeyStore	Création d'un « coffre-fort » individuel stocké sous la forme de fichier.
CreateCardKkeyStore	Création d'un « coffre-fort » individuel associé à une carte à puce ou un token USB
CreateKeyStoreForCard	Création d'un « coffre-fort » individuel associé à une carte à puce ou un token USB, dans une variante où c'est l'application qui calcule et fournit les éléments cryptographiques d'association (dont le mot de passe), et non le moteur crypto.
TransKeyStore	Transchiffrement d'un « coffre-fort » individuel par saisie du mot de passe SO (Niveau de résistance : **SOF-élevé)**.
OpenMemoryKeyStore	Ouverture d'un « coffre-fort » "mémoire" individuel par saisie du mot de passe utilisateur ou SO. Ce coffre-fort "mémoire" ne s'appuie sur aucun fichier ; son contenu est perdu lors de sa fermeture.
ComputePwdFromCardKey	Calcul du mot de passe d'un « coffre-fort » associé à une carte à puce ou un token USB Si la clé utilisée pour calculer le mot de passe n'est pas passée en entrée, alors la fonction la recherche dans la carte.
ComputePwdFromCard	Calcul du mot de passe d'un « coffre-fort » associé à une carte à puce ou un token USB. La clé utilisée pour calculer le mot de passe est passée en entrée.
EnterPrivateRandomSection	Initialise une instance de génération déterministe d'aléa.
LeavePrivateRandomSection	Termine une instance de génération déterministe d'aléa.

Reproduction, copie, prêt, diffusion interdits sans accord préalable.

Fonctions de sécurité PKCS#11

OpenSession	opens a session between an application and a token.
CloseSession	closes a session between an application and a token.
CloseAllSessions	closes all sessions with a token.
Login	logs a user into a token (Niveau de résistance : **SOF-élevé)**.
Logout	logs a user out from a token.
InitPIN	initializes the normal user's PIN.
SetPIN	modifies the PIN of the user who is logged in (Niveau de résistance : **SOF-élevé)**.
CreateObject	creates a new object.
CopyObject	copies an object, creating a new object for the copy.
DestroyObject	destroys an object.
SetAttributeValue	modifies the value of one or more object attributes
FindObjectsInit	initializes a search for token and session objects that match a template.
FindObjects	continues a search for token and session objects that match a template, obtaining additional object handles.
FindObjectsFinal	finishes a search for token and session objects.
EncryptInit	initializes an encryption operation.
Encrypt	encrypts single-part data.
EncryptUpdate	continues a multiple-part encryption operation.
EncryptFinal	finishes a multiple-part encryption operation.
DecryptInit	initializes a decryption operation.
Decrypt	decrypts encrypted data in a single part.
DecryptUpdate	continues a multiple-part decryption operation.
DecryptFinal	finishes a multiple-part decryption operation.
DigestInit	initializes a message-digesting operation.
Digest	digests data in a single part.
DigestUpdate	continues a multiple-part message-digesting operation.
DigestKey	continues a multi-part message-digesting operation, by digesting the value of a secret key as part of the data already digested.
DigestFinal	finishes a multiple-part message-digesting operation.
SignInit	initializes a signature (private key encryption) operation, where the signature is (will be) an appendix to the data, and plaintext cannot be recovered from the signature.
Sign	signs (encrypts with private key) data in a single part, where the signature is (will be) an appendix to the data, and plaintext cannot be recovered from the signature.
SignUpdate	continues a multiple-part signature operation, where the signature is (will be) an appendix to the data, and plaintext cannot be recovered from the signature.
SignFinal	finishes a multiple-part signature operation, returning the signature.

Reproduction, copie, prêt, diffusion interdits sans accord préalable.

SignRecoverInit	initializes a signature operation, where the data can be recovered from the signature.
SignRecover	signs data in a single operation, where the data can be recovered from the signature.
VerifyInit	initializes a verification operation, where the signature is an appendix to the data, and plaintext cannot be recovered from the signature (e.g. DSA).
Verify	verifies a signature in a single-part operation, where the signature is an appendix to the data, and plaintext cannot be recovered from the signature.
VerifyUpdate	continues a multiple-part verification operation, where the signature is an appendix to the data, and plaintext cannot be recovered from the signature.
VerifyFinal	finishes a multiple-part verification operation, checking the signature.
VerifyRecoverInit	initializes a signature verification operation, where the data is recovered from the signature.
VerifyRecover	verifies a signature in a single-part operation, where the data is recovered from the signature.
DigestEncryptUpdate	continues a multiple-part digesting and encryption operation.
DecryptDigestUpdate	continues a multiple-part decryption and digesting operation.
SignEncryptUpdate	continues a multiple-part signing and encryption operation.
DecryptVerifyUpdate	continues a multiple-part decryption and verify operation.
GenerateKey	generates a secret key, creating a new key object.
GenerateKeyPair	generates a public-key/private-key pair, creating new key objects.
WrapKey	wraps (i.e., encrypts) a key.
UnwrapKey	unwraps (decrypts) a wrapped key, creating a new key object.
DeriveKey	derives a key from a base key, creating a new key object.
SeedRandom	mixes additional seed material into the token's random number generator.
GenerateRandom	generates random data.

6.2. Mesures d'assurance

Les mesures intégrées au processus de développement de la société M.S.I. permettent de produire les fournitures attendues pour l'évaluation.

Reproduction, copie, prêt, diffusion interdits sans accord préalable.

7. Annonce de conformité à un Profil de Protection

Sans objet.

Reproduction, copie, prêt, diffusion interdits sans accord préalable.

8. Argumentaires

8.1. Argumentaire pour les objectifs de sécurité

Le tableau ci-dessous justifie la nécessité des objectifs de sécurité rédigés par rapport aux hypothèses et aux menaces retenues :

		Hyp.		Menaces												
		H.NON_OBSERV	H.NON_PIEG	M.ACCES	M.USURP_LOGIN	M.VOL_COFFRE	M.MODIF_SBOX	M.ABSENCE_TEMP	M.ARRET_PC	M.OUBLI_LOGIN	M.AUTRE_UTIL	M.RESIDUS	M.PLANTAGE	M.DESTRUCTION	M.MODIF_COFFRE	M.ALEAS
Objectifs de sécurité pour la TOE	O.CTL_ACCES			X												
	O.ALGO_STD					X										
	O.INTERF_TOKEN				X											
	O.VERROUILLAGE							X								
	O.CONFIDENTIALITE								X							
	O.ROLES									X						
	O.REINIT_LOGIN									X						
	O.GEST_SECRETS										X					
	O.EFF_RESIDUS											X				
	O.INTEGRITE												X			
	O.INTEG_COFFRE														X	
	O.ALEAS															X
Objectifs de sécurité pour l'env. de la TOE	OE.NON_OBSERV	X														
	OE.NON_PIEG		X													
	OE.ALGO_UTILISES					X										
	OE.ROBUST_MDP				X											
	OE.RENOUV_MDP				X											
	OE.LOGIN_OFFICIER				X											
	OE.VERIF_INTEGRITE						X									
	OE.VERR_AUTO							X								
	OE.SAUVEGARDE													X		
	OE.USAGE_ALEAS															X

Le tableau ci-dessous justifie la suffisance des objectifs de sécurité rédigés par rapport aux hypothèses et aux menaces retenues :

H.NON_OBSERV	*L'environnement physique de la TOE permet aux utilisateurs d'entrer leur mot de passe sans être observable directement ou interceptable par d'autres utilisateurs ou attaquants potentiels.*
OE.NON_OBSERV	L'environnement physique de la TOE empêche l'observation du mot de passe saisi par un utilisateur.
H.NON_PIEG	*Le PC de l'utilisateur supportant la TOE est considéré comme sain, c'est-à-dire qu'il ne comporte aucun dispositif matériel ou logiciel permettant d'accéder aux biens sensibles protégés par la TOE lorsqu'une session utilisateur est ouverte, en contournant les fonctions de sécurité de la TOE.*
OE.NON_PIEG	L'utilisation d'un logiciel anti-virus à jour permet de supprimer le risque de piégeage logiciel. L'inspection physique doit permettre de détecter tout piégeage matériel.
M.ACCES	*Un attaquant consulte de manière non autorisée les données sensibles d'un utilisateur protégées par la TOE, en exécutant la TOE directement.*
O.CTL_ACCES	L'identification et l'authentification réussies de l'utilisateur par la TOE sont obligatoires pour l'ouverture du « coffre-fort » individuel de cet utilisateur.
M.USURP_LOGIN	*Un attaquant désirant accéder aux données sensibles d'un utilisateur tente d'ouvrir une session en essayant systématiquement tous les mots de passe possibles.*
O.INTERF_TOKEN	A la place d'un mot de passe, la TOE admet la présentation d'une carte à puce ou d'un jeton USB et la saisie d'un code PIN.
OE.ROBUST_MDP	L'utilisateur dispose d'une aide pour la création de mots de passe, lui permettant d'estimer la robustesse de son mot de passe.
OE.RENOUV_MDP	L'utilisateur s'assure de la non divulgation de son mot de passe et de son renouvellement périodique.
OE.LOGIN_OFFICIER	L'utilisateur conserve le mot de passe de l'officier de sécurité associé à son "coffre-fort individuel" de manière sécurisée.
M.VOL_COFFRE	*Un attaquant récupère sur le PC d'un utilisateur le fichier contenant son « coffre-fort » individuel et essaie toutes les attaques cryptographiques connues pour accéder aux données sensibles de cet utilisateur.*
O.ALGO_STD	La TOE propose des algorithmes cryptographiques et des tailles de clés permettant de rendre impossible toute attaque par force brute.
OE.ALGO_UTILISES	Les applications s'appuyant sur la TOE doivent sélectionner des algorithmes cryptographiques pour lesquels aucune vulnérabilité n'est connue et des tailles de clés suffisamment longues.
M.MODIF_SBOX	*Un attaquant va modifier la TOE pour y insérer une fonction cachée de recouvrement, ce qui lui permettra d'accéder aux données sensibles d'un utilisateur.*
OE.VERIF_INTEGRITE	L'utilisateur a la possibilité de vérifier régulièrement l'intégrité de la TOE.
M.ABSENCE_TEMP	*Profitant de l'absence temporaire de l'utilisateur ayant ouvert une session avec la TOE, un attaquant accède à son PC et récupère les données sensibles de ce dernier protégées par la TOE.*
O.VERROUILLAGE	La TOE permet le verrouillage du « coffre-fort » individuel d'un utilisateur connecté.

Reproduction, copie, prêt, diffusion interdits sans accord préalable.

OE.VERR_AUTO	Le verrouillage par la TOE du « coffre-fort » individuel d'un utilisateur connecté est déclenché en cas d'inactivité prolongée dont la durée est paramétrable par l'utilisateur et/ou en cas de lancement du verrouillage d'écran du PC.
M.ARRET_PC	*Un arrêt brutal du PC empêche la fermeture «propre » de la session de l'utilisateur connecté, ce qui permet, après le redémarrage du PC, l'accès à certaines données sensibles qui ne sont plus protégées par la TOE.*
O.CONFIDENTIALITE	La confidentialité des données sensibles est assurée même suite à un arrêt brutal du PC qui a empêché la fermeture « propre » de la session des utilisateurs connectés.
M.OUBLI_LOGIN	*Un utilisateur autorisé oublie son mot de passe, ce qui lui interdit l'accès à ses données sensibles protégées par la TOE.*
O.ROLES	La TOE gère deux profils d'utilisateur : l'utilisateur du « coffre-fort » individuel et l'administrateur local.
O.REINIT_LOGIN	L'authentification en tant qu'officier de sécurité permet de réinitialiser le mot de passe de l'utilisateur
M.AUTRE_UTIL	*Un utilisateur malveillant possédant également un « coffre-fort » individuel sur le même PC ouvre une session avec la TOE pour accéder à son coffre-fort et profite de cette authentification réussie pour accéder aux données sensibles du « coffre-fort » individuel d'un autre utilisateur de ce PC.*
O.GEST_SECRETS	La TOE utilise des secrets différents pour chaque «coffre-fort » individuel.
M.RESIDUS	*Un attaquant désirant accéder aux données sensibles d'un utilisateur contenues dans son « coffre-fort » individuel récupère, sur le PC de celui-ci et après la fermeture de la session par la TOE, des traces, dans la mémoire ou sur le disque dur, contenant des fragments de données sensibles protégées par la TOE.*
O.EFF_RESIDUS	La TOE nettoie les fchiers temporaires sur le disque dur du PC ou les zones de mémoires allouées.
M.PLANTAGE	*Un arrêt brutal du PC pendant le chiffrement ou le déchiffrement d'un fichier d'un utilisateur connecté rend ce fichier inutilisable et irrécupérable par la TOE.*
O.INTEGRITE	La TOE assure l'intégrité des données sensibles, même suite à un arrêt brutal du PC.
M.DESTRUCTION	*Un attaquant désirant empêcher l'accès d'un utilisateur à ses données sensibles accède à son PC et détruit ou corrompt le fichier contenant son « coffre-fort » individuel utilisé par la TOE.*
OE.SAUVEGARDE	L'utilisateur sauvegarde régulièrement son « coffre-fort » individuel..
M.MODIF_COFFRE	*Un attaquant réussit de manière non autorisée à ajouter ou modifier un objet, ou l'attribut PKCS#11 d'un objet, contenu dans le «coffre-fort » individuel d'un utilisateur et protégé par la TOE.*
O.INTEG_COFFRE	La TOE contrôle l'intégrité des « coffres-forts » individuel des utilisateurs, pour éviter les ajouts ou modifications non autorisées.
M.ALEAS	*La génération des aléas réalisée par la TOE n'est pas assez aléatoire et peut être prévisible lors de la création des clés. Un attaquant connaissant cette faiblesse accède aux données sensibles des autres utilisateurs protégés par la TOE.*
O.ALEAS	La TOE doit offrir un mécanisme de génération de pseudo-aléas ou d'aléas vrais.
OE.USAGE_ALEAS	L'utilisateur doit générer suffisamment de bruit lors de la création des aléas pour assurer une bonne qualité d'aléas.

Reproduction, copie, prêt, diffusion interdits sans accord préalable.

8.2. Argumentaire pour les exigences de sécurité

Le tableau ci-dessous démontre la couverture des dépendances entre les composants fonctionnels sélectionnés :

Composant	Dépendances à respecter	Commentaires
FCS_CKM.1	[FCS_CKM.2 ou FCS_COP.1], FCS_CKM.4, FMT_MSA.2	Couvert
FCS_CKM.3	[FDP_ITC.1 ou FCS_CKM.1], FCS_CKM.4, FMT_MSA.2	Couvert
FCS_CKM.4	[FDP_ITC.1 ou FCS_CKM.1], FMT_MSA.2	Couvert
FCS_COP.1	[FDP_ITC.1 ou FCS_CKM.1], FCS_CKM.4, FMT_MSA.2	Couvert
FDP_ACC.2	FDP_ACF.1	Couvert
FDP_ACF.1	FDP_ACC.1, FMT_MSA.3	Couvert, car FDP_ACC.2 hiérarchiquement supérieur à FDP_ACC.1
FDP_ETC.1	[FDP_ACC.1 ou FDP_IFC.1]	Couvert
FDP_ITC.1	[FDP_ACC.1 ou FDP_IFC.1], FMT_MSA.3	Couvert
FDP_RIP.2		Pas de dépendances
FDP_SDI.2		Pas de dépendances
FIA_UAU.2	FIA_UID.1	Couvert, car FIA_UID.2 hiérarchiquement supérieur à FIA_UID.1
FIA_UID.2		Pas de dépendances
FMT_MSA.1	[FDP_ACC.1 ou FDP_IFC.1], FMT_SMR.1	Couvert
FMT_MSA.2	ADV_SPM.1, [FDP_ACC.1 ou FDP_IFC.1], FMT_MSA.1, FMT_SMR.1	Couvert
FMT_MSA.3	FMT_MSA.1, FMT_SMR.1	Couvert
FMT_SMR.1	FIA_UID.1	Couvert
FPT_FLS.1	ADV_SPM.1	Couvert
FTA_SSL.2	FIA_UAU.1	Couvert

Reproduction, copie, prêt, diffusion interdits sans accord préalable.

Le tableau ci-dessous justifie la nécessité des composants fonctionnels CC sélectionnés par rapport aux objectifs de sécurité rédigés :

		Objectifs de sécurité pour la TOE											
		O.CTL_ACCES	O.ALGO_STD	O.INTERF_TOKEN	O.VERROUILLAGE	O.CONFIDENTIALITE	O.ROLES	O.REINIT_LOGIN	O.GEST_SECRETS	O.EFF_RESIDUS	O.INTEGRITE	O.INTEG_COFFRE	O.ALEAS
Composants fonctionnels CC	FCS_CKM.1		X										X
	FCS_CKM.3		X										
	FCS_CKM.4		X										
	FCS_COP.1		X										X
	FDP_ACC.2	X				X			X				
	FDP_ACF.1	X				X			X				
	FDP_ETC.1			X									
	FDP_ITC.1			X									
	FDP_RIP.2					X				X			
	FDP_SDI.2										X	X	
	FIA_UAU.2	X		X				X					
	FIA_UID.2	X		X				X					
	FMT_MSA.1	X						X					
	FMT_MSA.2	X						X					
	FMT_MSA.3	X						X	X				
	FMT_SMR.1						X						
	FPT_FLS.1					X							
	FTA_SSL.2				X								

Reproduction, copie, prêt, diffusion interdits sans accord préalable.

Le tableau ci-dessous justifie la suffisance des composants fonctionnels CC sélectionnés par rapport aux objectifs de sécurité rédigés :

O.CTL_ACCES	*La TOE ne doit autoriser l'accès d'une donnée sensible contenue dans le " coffre-fort " individuel d'un utilisateur qu'après présentation du mot de passe associé et vérification des attributs PKCS#11 associés à cette donnée.*
FDP_ACC.2	Permet l'application d'une politique de contrôle d'accès au « coffre-fort » individuel
FDP_ACF.1	Permet de spécifier une politique de contrôle d'accès aux objets du « coffre-fort » individuel, basé sur les attributs de sécurité
FIA_UAU.2	Permet l'authentification d'un utilisateur préalablement à toute action
FIA_UID.2	Permet l'identification d'un utilisateur préalablement à toute action
FMT_MSA.1	Permet de décrire la gestion des attributs de sécurité des objets stockés
FMT_MSA.2	Permet de décrire ce que sont les attributs de sécurité sûrs
FMT_MSA.3	Demande l'initialisation statique des attributs de sécurité
O.ALGO_STD	*La TOE doit fournir un choix d'algorithmes cryptographiques et de tailles de clés conformes à l'état de l'art et aux standards de ce domaine.*
FCS_CKM.1	Décrit les algorithmes de génération de clés cryptographiques
FCS_CKM.3	Décrit les algorithmes d'accès aux clés cryptographiques
FCS_CKM.4	Décrit les algorithmes de destruction de clés cryptographiques
FCS_COP.1	Décrit les opérations cryptographiques possibles
O.INTERF_TOKEN	*La TOE doit autoriser l'identification et l'authentification d'un utilisateur par la présentation d'une carte à puce ou d'un jeton USB et la saisie d'un code PIN.*
FDP_ETC.1	Permet l'exportation de données de l'utilisateur sans attributs de sécurité vers une carte à puce ou un token USB
FDP_ITC.1	Permet l'importation de données de l'utilisateur sans attributs de sécurité depuis une carte à puce ou un token USB
FIA_UAU.2	Permet l'authentification d'un utilisateur préalablement à toute action par présentation d'une carte à puce ou d'un token USB
FIA_UID.2	Permet l'identification d'un utilisateur préalablement à toute action par présentation d'une carte à puce ou d'un token USB
O.VERROUILLAGE	*La TOE doit permettre le verrouillage du " coffre-fort " individuel d'un utilisateur connecté.*
FTA_SSL.2	Permet le verrouillage d'une session, initié par l'utilisateur
O.CONFIDENTIALITE	*La TOE doit assurer la confidentialité des données sensibles des utilisateurs contenues dans leur " coffre-fort " individuel respectif, même suite à un arrêt brutal du PC qui a empêché la fermeture " propre " de la session des utilisateurs connectés.*
FDP_ACC.2	Permet l'application d'une politique de contrôle d'accès au « coffre-fort » individuel
FDP_ACF.1	Permet de spécifier une politique de contrôle d'accès aux objets du « coffre-fort » individuel, basé sur les attributs de sécurité

FDP_RIP.2	Permet une protection totale des informations résiduelles par effacement
FPT_FLS.1	Permet une défaillance avec préservation d'un état sûr (pas de données récupérables en clair)
O.ROLES	*La TOE doit gérer deux profils d'utilisateur : l'utilisateur du " coffre-fort " individuel et l'administrateur local.*
FMT_SMR.1	Permet de gérer les rôles de sécurité USER et SO
O.REINIT_LOGIN	*La TOE doit offrir un mécanisme de secours qui permet à un utilisateur ayant oublier son mot de passe de réinitialiser celui-ci en s'authentifiant en tant qu'officier de sécurité associé à son "coffre-fort individuel".*
FIA_UAU.2	Permet l'authentification du SO
FIA_UID.2	Permet l'identification du SO
FMT_MSA.1	Permet de décrire la gestion des attributs de sécurité des objets stockés
FMT_MSA.2	Permet de décrire ce que sont les attributs de sécurité sûrs
FMT_MSA.3	Demande l'initialisation statique des attributs de sécurité
O.GEST_SECRETS	*La TOE doit utiliser des secrets différents pour protéger les "coffres-forts individuels" qu'elle gère.*
FDP_ACC.2	Permet l'application d'une politique de contrôle d'accès au « coffre-fort » individuel
FDP_ACF.1	Permet de spécifier une politique de contrôle d'accès aux objets du « coffre-fort » individuel, basé sur les attributs de sécurité
FMT_MSA.3	Demande l'initialisation statique des attributs de sécurité avec des valeurs différents
O.EFF_RESIDUS	*La TOE doit assurer le nettoyage par réécritures successives des fichiers temporaires sur le disque dur du PC ou des zones de mémoires allouées, dès la fin de traitement de données sensibles.*
FDP_RIP.2	Permet une protection totale des informations résiduelles par effacement
O.INTEGRITE	*La TOE doit assurer l'intégrité des données sensibles des utilisateurs connectés lors des traitements qu'elle réalise, même suite à un arrêt brutal du PC.*
FDP_SDI.2	Contrôle de l'intégrité des données traitées par la TOE et décrit les actions à entreprendre
O.INTEG_COFFRE	*La TOE doit assurer l'intégrité des " coffres-forts " individuel des utilisateurs.*
FDP_SDI.2	Contrôle de l'intégrité des données stockées dans un « coffre-fort » individuel et décrit les actions à entreprendre
O.ALEAS	*La TOE doit offrir un mécanisme de génération de pseudo-aléas ou d'aléas vrais.*
FCS_CKM.1	Décrit les algorithmes de génération de clés cryptographiques
FCS_COP.1	Décrit les opérations cryptographiques possibles

8.3. Argumentaire pour les spécifications globales de la TOE

Le tableau ci-dessous justifie la nécessité des fonctions de sécurité de la TOE par rapport aux composants fonctionnels CC sélectionnés :

		Composants fonctionnels de la TOE																	
		FCS_CKM.1	FCS_CKM.3	FCS_CKM.4	FCS_COP.1	FDP_ACC.2	FDP_ACF.1	FDP_ETC.1	FDP_ITC.1	FDP_RIP.2	FDP_SDI.2	FIA_UAU.2	FIA_UID.2	FMT_MSA.1	FMT_MSA.2	FMT_MSA.3	FMT_SMR.1	FPT_FLS.1	FTA_SSL.2
Fonctions de sécurité de la TOE	OpenKeyStore										X	X	X					X	
	OpenCardKeyStore										X	X	X					X	
	CloseKeyStore			X						X									X
	CreateKeyStore																X		
	CreateCardKkeyStore																X		
	CreateKeyStoreForCard																X		
	TransKeyStore										X						X		
	OpenMemoryKeyStore											X	X					X	
	ComputePwdFromCardKey											X					X		
	ComputePwdFromCard											X					X		
	EnterPrivateRandomSection	X																	
	LeavePrivateRandomSection	X																	
	OpenSession																	X	
	CloseSession									X									X
	CloseAllSessions									X									X
	Login											X	X						
	Logout									X									
	InitPIN											X					X		
	SetPIN											X					X		
	CreateObject	X				X	X		X					X	X	X			
	CopyObject	X				X	X		X					X	X	X			
	DestroyObject			X		X	X			X									
	SetAttributeValue	X				X	X		X					X	X			X	
	FindObjectsInit		X			X	X	X											
	FindObjects		X			X	X	X											
	FindObjectsFinal		X			X	X	X											
	EncryptInit				X														
	Encrypt				X														
	EncryptUpdate				X														
	EncryptFinal				X														
	DecryptInit				X														
	Decrypt				X														
	DecryptUpdate				X														
	DecryptFinal				X														
	DigestInit				X														
	Digest				X														
	DigestUpdate				X														
	DigestKey				X														
	DigestFinal				X														
	SignInit				X														
	Sign				X														
	SignUpdate				X														
	SignFinal				X														

Reproduction, copie, prêt, diffusion interdits sans accord préalable.

		Composants fonctionnels de la TOE																	
		FCS_CKM.1	FCS_CKM.3	FCS_CKM.4	FCS_COP.1	FDP_ACC.2	FDP_ACF.1	FDP_ETC.1	FDP_ITC.1	FDP_RIP.2	FDP_SDI.2	FIA_UAU.2	FIA_UID.2	FMT_MSA.1	FMT_MSA.2	FMT_MSA.3	FMT_SMR.1	FPT_FLS.1	FTA_SSL.2
Fonctions de sécurité de la TOE	SignRecoverInit				X														
	SignRecover				X														
	VerifyInit				X														
	Verify				X														
	VerifyUpdate				X														
	VerifyFinal				X														
	VerifyRecoverInit				X														
	VerifyRecover				X														
	DigestEncryptUpdate				X														
	DecryptDigestUpdate				X														
	SignEncryptUpdate				X														
	DecryptVerifyUpdate				X														
	GenerateKey	X																	
	GenerateKeyPair	X																	
	WrapKey				X														
	UnwrapKey				X														
	DeriveKey	X																	
	SeedRandom	X																	
	GenerateRandom	X																	

Le tableau ci-dessous justifie la suffisance des fonctions de sécurité de la TOE par rapport aux composants fonctionnels CC sélectionnés :

FCS_CKM.1	*Génération de clés cryptographiques*
CreateObject	Permet de créer un nouvel objet dans le “coffre-fort”, pour stocker la nouvelle clé.
CopyObject	Permet de créer un nouvel objet dans le “coffre-fort”, par duplication d’une clé existante.
SetAttributeValue	Permet de modifier l’un des attributs de la nouvelle clé.
GenerateKey	Permet de générer une clé secrète.
GenerateKeyPair	Permet de générer un bi-clé.
DeriveKey	Permet de dériver une clé secrète à partir d’une clé de base.
SeedRandom	Permet d’ajouter du bruit supplémentaire pour la génération d’aléas.
GenerateRandom	Permet de générer de l’aléa.
EnterPrivateRandomSection	Initialise une instance de génération déterministe d'aléa.
LeavePrivateRandomSection	Termine une instance de génération déterministe d'aléa.
FCS_CKM.3	*Accès aux clés cryptographiques*
FindObjectsInit	Initialise la recherche de clés dans un “coffre-fort”, en fonction de leurs attributs.
FindObjects	Continue la recherche de clés pour obtenir les clés suivantes, en fonction de leurs attributs.
FindObjectsFinal	Termine la recherche de clés.

FCS_CKM.4	*Destruction de clés cryptographiques*
CloseKeyStore	Efface les contextes mémoire lors de la fermeture d'un « coffre-fort ».
DestroyObject	Efface une clé dans un "coffre-fort".
FCS_COP.1	*Opération cryptographique*
EncryptInit	Chiffrement de données
Encrypt	
EncryptUpdate	
EncryptFinal	
DecryptInit	Déchiffrement de données
Decrypt	
DecryptUpdate	
DecryptFinal	
DigestInit	Calcul de sceaux sur des données
Digest	
DigestUpdate	
DigestKey	
DigestFinal	
SignInit	Signature de données
Sign	
SignUpdate	
SignFinal	
SignRecoverInit	Récupération de données à partir de la signature
SignRecover	
VerifyInit	Vérification de la signature de données
Verify	
VerifyUpdate	
VerifyFinal	
VerifyRecoverInit	Vérification de la signature à partir des données récupérées de la signature
VerifyRecover	
DigestEncryptUpdate	Opération combinée de scellement puis de chiffrement.
DecryptDigestUpdate	Opération combinée de déchiffrement puis de scellement.
SignEncryptUpdate	Opération combinée de signature puis de chiffrement.
DecryptVerifyUpdate	Opération combinée de déchiffrement puis de vérification de signature.
WrapKey	Chiffrement d'une clé.
UnwrapKey	Déchiffrement d'une clé chiffrée.
FDP_ACC.2	*Contrôle d'accès complet*
CreateObject	Permet de créer un nouvel objet dans le "coffre-fort".
CopyObject	Permet de créer un nouvel objet dans le "coffre-fort", par duplication d'un objet existant.

Reproduction, copie, prêt, diffusion interdits sans accord préalable.

DestroyObject	Efface un objet. dans un “coffre-fort”.
SetAttributeValue	Permet de modifier l’un des attributs d’un objet.
FindObjectsInit	Initialise la recherche d’objets dans un “coffre-fort”, en fonction de leurs attributs.
FindObjects	Continue la recherche d’objets pour obtenir les objets suivants, en fonction de leurs attributs.
FindObjectsFinal	Termine la recherche d’objets.
FDP_ACF.1	*Contrôle d'accès basé sur les attributs de séc urité*
CreateObject	Permet de créer un nouvel objet dans le “coffre-fort”.
CopyObject	Permet de créer un nouvel objet dans le “coffre-fort”, par duplication d’un objet existant.
DestroyObject	Efface un objet. dans un “coffre-fort”.
SetAttributeValue	Permet de modifier l’un des attributs d’un objet.
FindObjectsInit	Initialise la recherche d’objets dans un “coffre-fort”, en fonction de leurs attributs.
FindObjects	Continue la recherche d’objets pour obtenir les objets suivants, en fonction de leurs attributs.
FindObjectsFinal	Termine la recherche d’objets.
FDP_ETC.1	*Exportation de données de l'utilisateur sans attributs de sécurité*
FindObjectsInit	Initialise la recherche d’objets dans un “coffre-fort”, en fonction de leurs attributs.
FindObjects	Continue la recherche d’objets pour obtenir les objets suivants, en fonction de leurs attributs.
FindObjectsFinal	Termine la recherche d’objets.
FDP_ITC.1	*Importation de données de l'utilisateur sans attributs de sécurité*
CreateObject	Permet de créer un nouvel objet dans le “coffre-fort”.
CopyObject	Permet de créer un nouvel objet dans le “coffre-fort”, par duplication d’un objet existant.
SetAttributeValue	Permet de modifier l’un des attributs d’un objet.
FDP_RIP.2	*Protection totale des informations résiduelles*
CloseKeyStore	Efface les contextes mémoire lors de la fermeture d’un « coffre-fort ».
CloseSession	Fermeture de la session entre une application et un token.
CloseAllSessions	Fermeture de toutes les sessions d’un token.
Logout	Déconnexion d’un utilisateur d’un token.
DestroyObject	Efface un objet dans un “coffre-fort”.
FDP_SDI.2	*Contrôle de l'intégrité des données stockées et action à entreprendre*
OpenKeyStore	Contrôle d’intégrité lors de l’ouverture d’un « coffre-fort » individuel (en lecture seule ou lecture/écriture).
OpenCardKeyStore	Contrôle d’intégrité lors de l’ouverture d’un « coffre-fort » individuel associé à une carte à puce ou un token (en lecture seule ou lecture/écriture)
TransKeyStore	Contrôle d’intégrité lors du transchiffrement d’un « coffre-fort » individuel (en lecture seule ou lecture/écriture).

Reproduction, copie, prêt, diffusion interdits sans accord préalable.

FIA_UAU.2	*Authentification d'un utilisateur préalablement à toute action*
OpenKeyStore	Ouverture d'un «coffre-fort » individuel (en lecture seule ou lecture/écriture) par saisie du mot de passe utilisateur ou SO.
OpenCardKeyStore	Ouverture d'un «coffre-fort » individuel associé à une carte à puce ou un token (en lecture seule ou lecture/écriture)
OpenMemoryKeyStore	Ouverture d'un «coffre-fort » "mémoire" individuel par saisie du mot de passe utilisateur ou SO. Ce coffre-fort "mémoire" ne s'appuie sur aucun fichier ; son contenu est perdu lors de sa fermeture.
ComputePwdFromCardKey	Calcul du mot de passe d'un «coffre-fort » associé à une carte à puce ou un token USB Si la clé utilisée pour calculer le mot de passe n'est pas passée en entrée, alors la fonction la recherche dans la carte.
ComputePwdFromCard	Calcul du mot de passe d'un «coffre-fort » associé à une carte à puce ou un token USB. La clé utilisée pour calculer le mot de passe est passée en entrée.
Login	Connexion d'un utilisateur sur un token.
InitPIN	Initialise le code PIN d'un utilisateur.
SetPIN	Modifie le code PIN d'un utilisateur déjà connecté.
FIA_UID.2	*Identification d'un utilisateur préalablement à toute action*
OpenKeyStore	Ouverture d'un «coffre-fort » individuel (en lecture seule ou lecture/écriture) par saisie du mot de passe utilisateur ou SO.
OpenCardKeyStore	Ouverture d'un «coffre-fort » individuel associé à une carte à puce ou un token (en lecture seule ou lecture/écriture)
OpenMemoryKeyStore	Ouverture d'un «coffre-fort » "mémoire" individuel par saisie du mot de passe utilisateur ou SO. Ce coffre-fort "mémoire" ne s'appuie sur aucun fichier ; son contenu est perdu lors de sa fermeture.
Login	Connexion d'un utilisateur sur un token.
FMT_MSA.1	*Gestion des attributs de sécurité*
CreateObject	Permet de créer un nouvel objet dans le "coffre-fort", avec des attributs par défaut.
CopyObject	Permet de créer un nouvel objet dans le "coffre-fort", par duplication d'un objet existant, avec les attributs de l'objet copié.
SetAttributeValue	Permet de modifier l'un des attributs d'un objet.
FMT_MSA.2	*Attributs de sécurité sûrs*
CreateObject	Permet de créer un nouvel objet dans le "coffre-fort", avec des attributs par défaut.
CopyObject	Permet de créer un nouvel objet dans le "coffre-fort", par duplication d'un objet existant, avec les attributs de l'objet copié.
SetAttributeValue	Vérifie la validité des nouvelles valeurs d'attribut pour l'objet.
FMT_MSA.3	*Initialisation statique d'attribut*
CreateObject	Permet de créer un nouvel objet dans le "coffre-fort", avec des attributs par défaut.
CopyObject	Permet de créer un nouvel objet dans le "coffre-fort", par

	duplication d'un objet existant, avec les attributs de l'objet copié.
FMT_SMR.1	*Rôles de sécurité*
CreateKeyStore	Création d'un « coffre-fort » individuel stocké sous la forme de fichier.
CreateCardKkeyStore	Création d'un «coffre-fort » individuel associé à une carte à puce ou un token USB
CreateKeyStoreForCard	Création d'un « coffre-fort » individuel associé à une carte à puce ou un token USB, dans une variante où c'est l'application qui calcule et fournit les éléments cryptographiques d'association (dont le mot de passe), et non le moteur crypto.
TransKeyStore	Transchiffrement d'un « coffre-fort » individuel (en lecture seule ou lecture/écriture) par saisie du mot de passe SO.
ComputePwdFromCardKey	Calcul du mot de passe d'un «coffre-fort » associé à une carte à puce ou un token USB Si la clé utilisée pour calculer le mot de passe n'est pas passée en entrée, alors la fonction la recherche dans la carte.
ComputePwdFromCard	Calcul du mot de passe d'un «coffre-fort » associé à une carte à puce ou un token USB. La clé utilisée pour calculer le mot de passe est passée en entrée.
InitPIN	Initialise le code PIN d'un utilisateur.
SetPIN	Modifie le code PIN d'un utilisateur déjà connecté.
FPT_FLS.1	*Défaillance avec préservation d'un état sûr*
OpenKeyStore	Contrôle d'intégrité lors de l'ouverture d'un « coffre-fort » individuel (en lecture seule ou lecture/écriture).
OpenCardKeyStore	Contrôle d'intégrité lors de l'ouverture d'un « coffre-fort » individuel associé à une carte à puce ou un token (en lecture seule ou lecture/écriture)
OpenMemoryKeyStore	Ouverture d'un «coffre-fort » "mémoire" qui ne s'appuie sur aucun fichier ; son contenu est perdu lors de sa fermeture.
OpenSession	Ouverture de la session entre une application et un token.
SetAttributeValue	Permet de modifier l'un des attributs d'un objet.
FTA_SSL.2	*Verrouillage d'une session, initié par l'utilisateur*
CloseKeyStore	Fermeture d'un « coffre-fort » individuel et des sessions cryptographiques associées.
CloseSession	Fermeture de la session entre une application et un token.
CloseAllSessions	Fermeture de toutes les sessions d'un token.

8.4. Argumentaire pour les annonces de conformité à un Profil de Protection

Sans objet.